2022 IEEE International Conference on Robotics and Automation (ICRA)
May 23-27, 2022. Philadelphia, PA, USA

f-*Cal*: Aleatoric uncertainty quantification for robot perception via calibrated neural regression

Dhaivat Bhatt*[1], Kaustubh Mani*[1], Dishank Bansal[1], Krishna Murthy[1], Hanju Lee[2], and Liam Paull[1]

[1]Montreal Robotics and Embodied AI Lab, Mila, Université de Montréal, [2]DENSO CORP.

***Abstract*— While modern deep neural networks are performant perception modules, performance (accuracy) alone is insufficient, particularly for safety-critical robotic applications such as self-driving vehicles. Robot autonomy stacks also require these otherwise blackbox models to produce reliable and *calibrated* measures of confidence on their predictions. Existing approaches estimate uncertainty from these neural network perception stacks by modifying network architectures, inference procedure, or loss functions. However, in general, these methods lack *calibration*, meaning that the predictive uncertainties do not faithfully represent the true underlying uncertainties (process noise). Our key insight is that calibration is only achieved by imposing constraints across multiple examples, such as those in a mini-batch; as opposed to existing approaches which only impose constraints per-sample, often leading to overconfident (thus miscalibrated) uncertainty estimates. By enforcing the distribution of outputs of a neural network to resemble a target distribution by minimizing an f-divergence, we obtain significantly better-calibrated models compared to prior approaches. Our approach, f-Cal, outperforms existing uncertainty calibration approaches on robot perception tasks such as object detection and monocular depth estimation over multiple real-world benchmarks.**

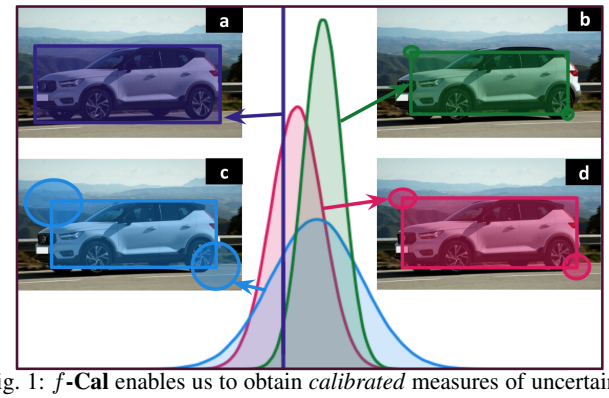

Fig. 1: **f-Cal** enables us to obtain *calibrated* measures of uncertainty from otherwise blackbox neural networks used in robot perception tasks. This didactic example demonstrates how f-Cal can estimate the *aleatoric* uncertainty from object detectors. (a) depicts a ground-truth bounding box, a single-sample (dirac-delta) distribution; (b) and (c) denote *uncalibrated* probabilistic outputs from a Bayesian neural network – (b) is overconfident and inconsistent, (c) is consistent but underconfident; (d) denotes a calibrated estimate, i.e., the error ellipses correspond to the *true* underlying aleatoric uncertainty.

I. Introduction

The *performance* of deep neural network-based visual perception systems has increased dramatically in recent years. However, for safety-critical *embodied* applications, such as autonomous driving, performance alone is not sufficient. The absence of reliable and *calibrated* uncertainty estimates in neural network predictions precludes us from incorporating these into downstream sensor fusion [1] or probabilistic planning [2], [3], [4] components.

The tools of probabilistic robotics require calibrated confidence/uncertainty measures, in the form of a *measurement model* $z = h(x, \nu)$. For a traditional sensor, this model h is specified by the designer's understanding of the physical sensing processes, and the noise distribution parameters ν are estimated by controlled calibration experiments with known ground truth states x^* and sensor observations z. However, for deep neural networks (DNNs) to be used as *sensors* in typical robotic perception stacks, estimating the noise distribution is a much more challenging task for several reasons. First, the domain of inputs is extremely high dimensional (e.g., RGB images) - generating a calibration setup for every possible input is infeasible. Second, the noise distribution is input dependent (heteroscedastic). Finally, neural networks typically transform the inputs via millions of nonlinear operations, preventing approximation by simpler (e.g., piecewise affine) models. We envision a **deep neural network (DNN) as a sensor** paradigm where a DNN outputs calibrated predictive distributions that may directly be used in probabilistic planning or sensor fusion. The challenge, however, is that these predictive distributions must be learned solely from training data, with neither additional postprocessing nor architectural modifications.

Our key insight is that **distributional calibration cannot be achieved by a loss function that operates over individual samples**. This motivates a new loss function that enforces calibration through a distributional constraint that is imposed upon uncertainty estimates across multiple (i.i.d.) samples. Specifically, our approach f-Cal, minimizes an f-divergence between a specified canonical distribution and an empirical distribution generated from neural network predictions, as shown in Fig. 2. Unlike prior approaches [5], [6], [7], we neither require a held-out calibration dataset nor impose any inference time overhead. For a given performance threshold, f-Cal achieves better calibration compared to current art. We demonstrate the effectiveness, scalability and widespread applicability of this approach on large-scale, real-world tasks such as object detection and depth estimation.

II. Related Work

The rapidly growing field of **Bayesian deep learning** has resulted in the development of models that estimate a *distribution* over the output space [8], [9], [10], [11], [12].

*Equal contribution. Project page: https://f-cal.github.io

2022 IEEE International Conference on Robotics and Automation (ICRA) | 978-1-7281-9681-7/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICRA46639.2022.9811903

There is a distinction between uncertainty that is due to the stochasticity of the underlying process (*aleatoric*) versus uncertainty that is due to the model being insufficiently trained (*epistemic*) [10] **Epistemic** uncertainty is often estimated by either using ensembles of neural networks or by stochastic regularization at inference time (Monte-Carlo dropout) [9], [11], [13]. **Distributional** uncertainty is also being extensively studied, to detect out of training-distribution examples [14], [11], [15], [16], [17], [18], [19], [20], [21]. However, there is no direct approach to address distributional uncertainty for regression settings.

In this work, we assume distributional and epistemic uncertainty to be low (i.e., in-distribution setting with reasonably well-trained models such as those common in robot perception), and focus specifically on calibrating **aleatoric uncertainty** estimates in regression problems. Such challenging settings have received far less attention in terms of uncertainty estimation [22], [5], [23], [7]. Existing calibration techniques are post-hoc and either require a large held-out calibration dataset [7] and/or add parameters to the model after training [7], [6]. Quantile regression methods [24], [13], [25], [26], [27] quantify uncertainty by the fraction of predictions in each quantile. Other methods, such as isotonic regression and temperature scaling, have also been extended to be the regression setting [22], [7]. Authors in [28] proposed an alternate architecture for aleatoric uncertainty estimation. However, f-Cal is completely architecture agnostic, and can be applied to any probabilistic neural regressors. More recently, a *calibration loss* is proposed in [7] that enforces the predicted variances to be equal to per-sample errors, thus grounding each prediction. However, this takes on a *local view* of the calibration problem, and while individual samples might appear well-calibrated, the overall distribution of the regressor errors exhibits a strong deviation from the expected target distribution (*cf.*Sec. V).

A recent approach that is somewhat similar to ours in spirit is Gaussian process beta calibration (GP-beta) [5]. It is a post-hoc approach that employs a Gaussian process model (with a beta-link function prior) to calibrate uncertainties during inference. This requires the computation of pairwise statistics, exacerbating inference time. In the maximum mean discrepancy (MMD) loss [29] distribution matching is performed to achieve calibration. This method was proposed for small datasets and does not scale well with input size. f-Cal is a superior performing loss function that requires the same inference time as typical Bayesian neural networks [30].

III. Problem Setup

A. Preliminaries

We assume a regression problem over an i.i.d. labelled training dataset $\mathcal{D} \triangleq \{(x_i, y_i)\}_{i=1...|\mathcal{D}|}$ with $x_i \in \mathcal{X}$ where $\mathcal{X}$ is the (n-dimensional) input space and $y_i \in \mathcal{Y}$ where $\mathcal{Y} \subseteq \mathbb{R}^n$ is the output space.

A *deterministic* model $f_d : \mathcal{X} \mapsto \mathcal{Y}$ [1] directly learns the mapping from the input to the output space by minimizing a loss function $\mathcal{L} : \mathcal{Y} \times \mathcal{Y} \mapsto \mathbb{R}$, for example through empirical risk minimization:

$$R_{emp}(f_d) = \frac{1}{N}\sum_{i=1}^{N} \mathcal{L}(f_d(x_i), y_i). \tag{1}$$

Equation 1 is typically estimated over a mini-batch of size $N << |\mathcal{D}|$ during stochastic gradient descent (SGD). Following the notation in [5], we desire a *probabilistic* model $f_p : \mathcal{X} \mapsto \mathcal{S}_\mathcal{Y}$ where $\mathcal{S}_\mathcal{Y}$ is the space of all probability density functions $s(y)$ over $\mathcal{Y}$ ($s : \mathcal{Y} \mapsto [0, \infty)$ and $\int s(y)dy = 1$). The probability density function (PDF) is defined through its cumulative density function (CDF): $S(y) = \int_{-\infty}^{y} s(y')dy'$.

B. Uncertainty Calibration

Calibrated uncertainty estimates are those where the output uncertainties can be exactly interpreted as confidence intervals of the underlying target label distribution. This allows uncertainty estimates across multiple samples (and models) to be compared. Intuitively, we understand the notion of uncertainty calibration to mean that if we repeated a stochastic experiment a large number of times, for example by asking many different people to label the same image, that the "label generating distribution" matches the predictive distribution of the model:

$$y_i \sim f_p(x_i) \tag{2}$$

However, it is impractical to label every piece of data multiple times. Instead, we aggregate the labels across many different inputs to produce calibrated predictive distributions. Using our definitions from Sec. III-A and adapting from [5], we can define what we desire in terms of calibration in the case of a deep neural regressor as follows:

*Definition 1 (**Uncertainty Calibration**):* A neural regressor f_p is calibrated if and only if [2]:

$$p(Y \leq y | s(y)) = \int_{-\infty}^{y} s(y')dy' \;\; \forall y \in \mathcal{Y} \tag{3}$$

In the above definition, Y is an instantiation of the random variable y. If we can assume that the noise is sampled from a parametric distribution $s(y; \phi)$, then the probabilistic model need only output the parameters associated with each sample. In this case, we can consider the model to be calibrated if and only if the aggregated error statistics over multiple outputs of a model align with the parameters predicted by the model.

C. Loss Attenuation (Negative Log-Likelihood - NLL)

The most widely used technique for estimating heteroscedastic aleatoric uncertainty is *loss attenuation* [10], [31], which performs maximum likelihood estimation by minimizing the negative log-likelihood loss:

$$\begin{aligned} R_{emp}(f_p) &= -\frac{1}{N}\sum_{i=1}^{N} \mathcal{L}_{\text{LA}}(f_p(x_i), y_i) \\ &= -\frac{1}{N}\sum_{i=1}^{N} \log s(y_i; f_p(x_i)) \end{aligned} \tag{4}$$

since $f_p(x_i)$ outputs the parameters of the distribution. For example, if the aleatoric uncertainty is characterized by a

[1] In practice these models are assumed to be neural networks with parameters θ but we omit the θ for clarity at this stage.

[2] Referring to Fig. 1, the requirement for calibration is more stringent than that of consistency, which is a one-way constraint at an arbitrary confidence bound c: $p(Y \leq y | s(y)) \leq c$

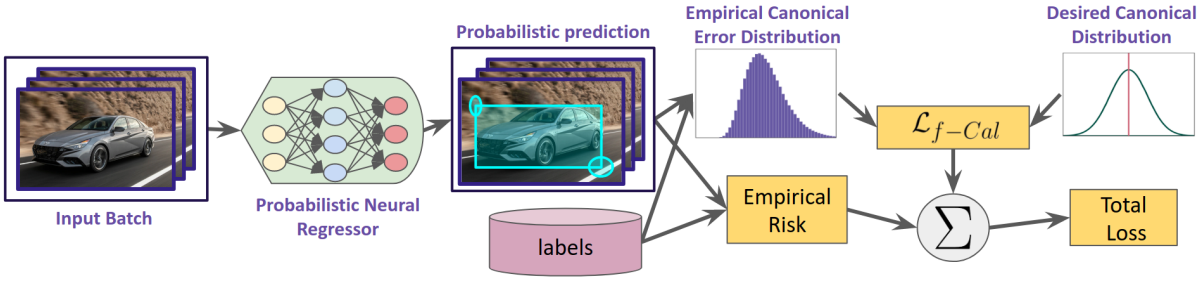

Fig. 2: **f-Cal pipeline**: We make a conceptually simple tweak to the loss function in a typical (deterministic) neural network training pipeline. In addition to the empirical risk (e.g., $L1$, $L2$, etc.) terms, we impose a distribution matching constraint ($\mathcal{L}_{f-\text{Cal}}$) over the error residuals across a mini-batch. By encouraging the distribution of these error residuals to match a target *calibrating distribution* (e.g., Gaussian), we ensure the neural network predictions are *calibrated*. Compared to prior approaches, most of which perform post-hoc calibration, or require large held-out calibration datasets, f-Cal does not impose an inference time overhead. f-Cal is task and architecture agnostic, and we apply it to robot perception problems such as object detection and depth estimation.

Gaussian random variable ($\phi \triangleq (\mu, \sigma)$), the above expression becomes

$$R_{emp}(f_p) = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{2}\left(\frac{(y_i-\mu_i)^2}{\sigma_i^2} + \log\sigma_i^2\right) \tag{5}$$

We refer to the loss in (5) as the **NLL** loss in the experiments. However, probabilistic neural regressors trained using this NLL objective typically lack **calibration** according to Def. 1.

IV. f-Cal: Variational Inference for Aleatoric Uncertainty Calibration

In this section we present f-Cal, a principled approach to obtain calibrated aleatoric uncertainty from neural nets.

A. Calibration as Distribution Matching

Following the definition of distributional calibration (Def. 1), f-Cal formulates a variational minimization objective to calibrate the uncertainty estimates from a deep network.

In the case of a traditional (non deep learning based) sensor, we would calibrate the noise distribution with the procedure:

1) Choose a distribution family for the noise
2) For a fixed and known input value, draw multiple samples of the output observations
3) Fit the output samples to the distribution family

In the DNNS case, we only have one sample for any given input and we have no knowledge of the ground truth (noise free) label. We can similarly choose a distribution family for our model, but we cannot assume that any of the parameters are fixed across samples. Our approach to overcome this problem will be to assume that there is some canonical element of the distribution family that we can transform each predictive distribution to. Specifically, we seek to approximate the empirical posterior over some canonical transformation of the target variables Y by a simpler (tractable) target distribution Q (modeling choice). This enables us to leverage an abundant class of distribution matching metrics, f-divergences, to formulate a loss function enforcing distributional calibration. For tractable inference, we assume i.i.d. mini-batches of training data and instead impose distribution matching losses over empirical error residuals across each batch.

We assume that we can transform each training sample output distribution to some canonical element of the distribution family. For instance, Gaussian random variables are canonicalized by centering the distribution (subtracting the output label), followed by normalization (scaling the result by the inverse variance). These canonical elements are used (in conjunction with the labels) to determine the *empirical* error distribution. f-Cal then performs distribution matching across this empirical and a target distribution.

B. f-Cal Algorithm

Given a mini-batch containing N inputs x_i, a probabilistic regressor predicts N sets of distributional parameters $f_p(x_i) = \phi_i$ ($\phi_i \in \Phi$) to the corresponding probability distribution $s(y_i; \phi_i)$. Define $g : \mathcal{Y} \times \Phi \mapsto \mathcal{Z}$ as the function that maps the target random variable y_i to a random variable z_i which follows a known canonical distribution. Since these residuals $\{z_1, z_2, \ldots, z_N\}$ must ideally follow a chosen calibrating (target) distribution Q:

$$z_i = g(y_i, \phi_i) \sim Q \tag{6}$$

The key difference between (2) and (6) is that (6) now applies **for all samples** in the dataset, as opposed to just a single sample. As a result, we can now follow the similar procedure that we would with a traditional sensor and compute the empirical statistics of the residuals of the z_i variables across the entire set (or in practice across a mini-batch) to fit a proposal distribution P_z, and minimize the distributional distance from the canonical distribution Q. This minimization can be performed with variational loss function that minimizes an f-divergence, $D_f(P_z||Q)$, between these two distributions. In summary, we propose a distribution matching loss function that augments typical supervised regression losses, and results in the neural regressor being calibrated to the target distribution:

$$\begin{aligned}\mathcal{L} &= (1-\lambda)R_{emp}(f_p) + \lambda\mathcal{L}_{f\text{-Cal}} \\ &= (1-\lambda)R_{emp}(f_p) + \lambda D_f(P_z||Q)\end{aligned} \tag{7}$$

where λ is a hyper-parameter to balance the two loss terms (we provide thorough analysis of the choice of λ in Sec. V. We experiment with a number of f-divergence choices, and identify KL-divergence and Wasserstein distance as viable choices. Importantly, f-Cal is agnostic to the choice of probabilistic deep neural regression model or task. In practice, it is a straightforward modification to the training loss function that can also be applied as a fine-tuning step to a previously partially trained model.

Algorithm 1: f-Cal for Gaussian uncertainties

Input : Dataset D, probabilistic neural regressor, f_p, degrees of freedom K, batch size N, number of samples for hyper-constraint H

for $i = 1 \dots N$ **do**
  $(\mu_i, \sigma_i) \leftarrow f_p(x_i)$
  $z_i \leftarrow \frac{y_i - \mu_i}{\sigma_i}$
end
$C = \emptyset$ // Samples from Chi-squared distribution
for $i = 1 \dots H$ **do**
  // Create a chi-squared hyper-constraint
  $q_i \leftarrow \sum_{j=1}^{K} z_{ij}^2, z_{ij} \sim \{z_1 \cdots z_N\}$
  C.append(q_i)
end
$P_z \leftarrow$ Fit-Chi-Squared-Distribution(C)
$\mathcal{L}_{f\text{-Cal}} \leftarrow D_f(P_z || \chi_K^2)$
return $\mathcal{L}_{f\text{-Cal}}$

C. f-Cal for Gaussian calibration

The f-Cal framework is generic and can be applied to arbitrary distributions. In this section we consider the case when the distribution $s(y_i; \phi_i)$ is Gaussian with $\phi_i \triangleq (\mu_i, \sigma_i)$. The variance σ_i^2 denotes the aleatoric uncertainty in this case. The error residuals are computed as $z_i = \frac{y_i - \mu_i}{\sigma_i}$, where μ_i and σ_i are predicted mean and the standard deviation of the ith Gaussian output from the neural network for each input x_i. So, $y_i \sim \mathcal{N}(\mu_i, \sigma_i^2)$, then $z_i \sim \mathcal{N}(0, 1)$.

Optionally, one may apply several transforms to the random variables y_i and impose distributional *hyper*-constraints over the transformed variables. In practice, we find that this can improve the stability of the training process and enforces more stable calibration. In this case we compute the sum-of-squared error residuals $q = \sum_{i=1}^{K} z_i^2$, and enforce the resulting distribution to be Chi-squared with parameter K i.e $q \sim \chi_K^2$, so in this case target distribution $Q = \chi_K^2$. Subsequently, we note that as the degrees of freedom K of a Chi-squared distribution increase, it can be approximated by a Gaussian of mean K and variance $2K$ through the application of the central limit theorem:

$$\lim_{K \to \infty} \frac{\chi_K^2 - K}{\sqrt{2K}} \to \mathcal{N}(0, 1) \implies \lim_{K \to \infty} \chi_K^2 \to \mathcal{N}(K, 2K)$$

In practice, this variation of the central limit theorem for Chi-squared random variables holds for moderate values of K (i.e., $K > 50$). This is practical to ensure, particularly in dense regression tasks such as bounding box object detection (where hundreds of proposals have to be scored per image) and per-pixel regression. We summarize the process for generating the calibration loss in Alg. 1. This loss is then combined with the typical empirical risk as given by (7).

V. Experiments

We conduct experiments on a number of large-scale perception tasks, on both synthetic and real-world datasets. We report the following key findings which we elaborate on in the remainder of this section.

1) f-Cal achieves significantly superior calibration compared to existing methods for calibrating aleatoric uncertainty.
2) These performance trends are consistently observed across multiple regression setups, neural network architectures, and dataset sizes.
3) We demonstrate that there is a trade-off between deterministic and calibration performance by varying the λ hyper-parameter. This trade-off has been established in previous literature [7], [17]. However, we further demonstrate empirically that this trade-off is inherently caused by a mismatch between the choice of the noise data distribution family and the true underlying noise distribution.

A. Regression tasks

We consider 3 regression tasks: a synthetic disc tracking dataset (Bokeh), KITTI depth estimation [32] and KITTI object detection [32]. These tasks are chosen to span the range of regression tasks relevant for robotics applications: sparse (one output per image in disc tracking), semi-dense (object detection), and pixelwise (fully) dense (depth estimation). Unless otherwise specified, we model aleatoric uncertainty using heteroscedastic Gaussian distributions.

B. Baselines

We compare f-Cal models with the following baselines: **NLL loss** [31], [8], **Temperature scaling** [7], **Isotonic regression** [6], **Calibration loss** [7] and **GP-beta** [5]. We report results for f-Cal, with KL-divergence (**f-Cal-KL**) and Wasserstein distance (**f-Cal-Wass**) as losses for distribution matching. We also experimented with a recently proposed maximum mean discrepancy based method [29]. Being designed for very low data regimes, it failed to solve any of our tasks considered. Similarly, GP-Beta [5] and isotonic regression [6] solve our synthetic tasks, but do not scale to large, real-world tasks.

C. Evaluation metrics

We evaluate the accuracy in calibration by means of the following widely used metrics. The **expected calibration error** (ECE) [34], [7] measures the discrete discrepancy between the predicted distribution of the neural regressor and that of the label distribution. We divide the predicted distribution into S intervals of size $\frac{1}{S}$. ECE is computed as the difference between the empirical bin frequency and the true frequency($\frac{1}{S}$). For total samples P and number of samples in bin s as B_s, $ECE = \sum_{s=1}^{S} \frac{|B_s|}{P} \left\| \frac{1}{S} - \frac{|B_s|}{P} \right\|$.

We report ECE scores for standard normal distribution and chi-squared distribution in this work, which we denote by ECE(z) and ECE(q) respectively. We also plot **reliability diagrams** which visually depict the amount of miscalibration over the support of the distribution. Perfectly calibrated distributions should have a diagonal reliability plot. Portions of a curve above the diagonal line are over-confident regions, while those below the curve are under-confident.

	Bokeh - synthetic dataset (a)					KITTI - depth estimation (b)					KITTI - Object detection (c)				Cityscapes - Object detection (d)			
Approach	**L1(GT)↓**	**L1↓**	**ECE(z)↓**	**ECE(q)↓**	**NLL↓**	**SiLog↓**	**RMSE↓**	**ECE(z)↓**	**ECE(q)↓**	**NLL↓**	**mAP↑**	**ECE(z)↓**	**ECE(q)↓**	**NLL↓**	**mAP↑**	**ECE(z)↓**	**ECE(q)↓**	**NLL↓**
NLL Loss[31]	1.44	1.54	1.73	91.83	-1.60	9.213	2.850	2.39	99.0	2.403	54.451	0.304	5.37	1.022	38.309	0.224	3.503	1.069
Calibration Loss[7]	1.46	1.57	1.13	76.11	-1.68	9.604	2.918	1.71	99.9	2.879	50.405	2.33	81.848	0.773	39.218	0.163	9.681	0.999
Temperature scaling[7]	1.44	1.54	0.82	9.22	-1.70	9.213	2.850	2.36	99.9	3.362	54.451	0.315	4.151	1.021	38.309	0.226	2.705	1.065
Isotonic regression[6]	1.38	1.49	2.05	9.38	-1.57	-	-		-	-	-	-	-	-	-	-	-	-
GP-Beta[5]	1.39	1.49	2.21	93.48	-1.54	-	-		-	-	-	-	-	-	-	-	-	-
f-Cal-KL (ours)	1.42	1.52	0.56	9.21	-1.76	9.679	2.911	0.074	22.5	2.004	51.874	0.162	4.126	0.846	38.481	0.126	1.686	0.929
f-Cal-Wass (Ours)	1.43	1.54	0.79	7.99	-1.75	9.509	3.202	0.156	67.9	2.157	48.04	0.115	0.768	0.914	37.220	0.104	0.832	1.007

TABLE I: **f-Cal - Results**: We evaluate f-Cal for a wide range of robot perception tasks and datasets. In each column group (a, b, c, d), we report an empirical risk (deterministic performance metric such as L1, SiLog, RMSE, mAP), expected calibration errors (ECE), and negative log-likelihood. f-Cal consistently outperforms all other calibration techniques considered (lower ECE values). (a) We develop Bokeh – a synthetic disc tracking benchmark that contains GT uncertainty values, useful for baseline comparisons. (b) Depth estimation on the KITTI benchmark [32]. (c) Object detection on the KITTI benchmark [32]. (d) Object detection on the Cityscapes dataset [33]. Notably, f-Cal improves calibration without sacrificing deterministic performance. (Note: L1 scores are scaled by a factor of 1000 and ECE scores by a factor 100 for improved readability. ↓: Lower is better, ↑: Higher is better, −: Method did not scale to task/dataset)

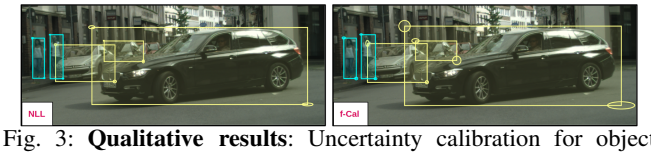

Fig. 3: **Qualitative results**: Uncertainty calibration for object detection models (Faster RCNN) over the KITTI [32] dataset. (*Left*) Models trained using an NLL loss term produce overconfident predictions (notice how the model outputs small, low uncertainty, ellipses for the occluded cars). (*Right*) f-Cal, on the other hand, produces calibrated uncertainty estimates (notice the large covariances for occluded cars, and the car in the foreground, whose endpoints are indeed uncertain).

D. Bokeh: A synthetic disc-tracking benchmark

Since ground-truth estimates of aleatoric uncertainty are extremely challenging to obtain from real-world datasets, we first validate our proposed approach in simulation.

Setup: We design a synthetic dataset akin to [35] for a *disc-tracking* task. The goal is to predict the 2D location of the centre of a unique red disc from an input image containing other distractor discs. All disc locations are sampled from a known data-generating distribution.

Models: We use a 3-layer ConvNet architecture with an uncertainty prediciton head. We train a model using the NLL loss [31] for our baseline probabilistic regressor. We then train two models using our proposed f-Cal loss (f-Cal-KL and f-Cal-Wass).

Results: Table I(a) compares f-Cal to the aforementioned baselines, evaluating *performance* (i.e., the accuracy of the estimated mean) and *calibration* quality.

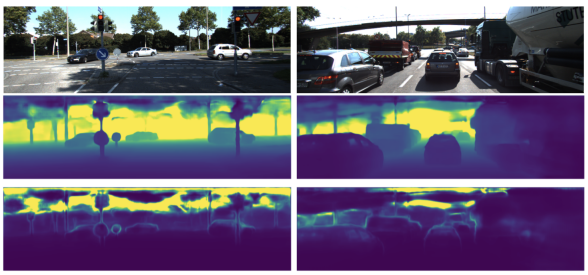

Fig. 4: **Qualitative results** for depth estimation models on the KITTI [32] benchmark. (*Top*) Input image; (*Middle*) Predicted depth; (*Bottom*) Predicted uncertainty.

We report the performance (Smooth-L1 error) denoted by L1 in Table I for both the *noise-free* ground-truth (in typical ML settings, we never have access to this variable. We only ever access the noisy ground-truth labels), and the *noisy* ground-truth (accounting for label generation error).

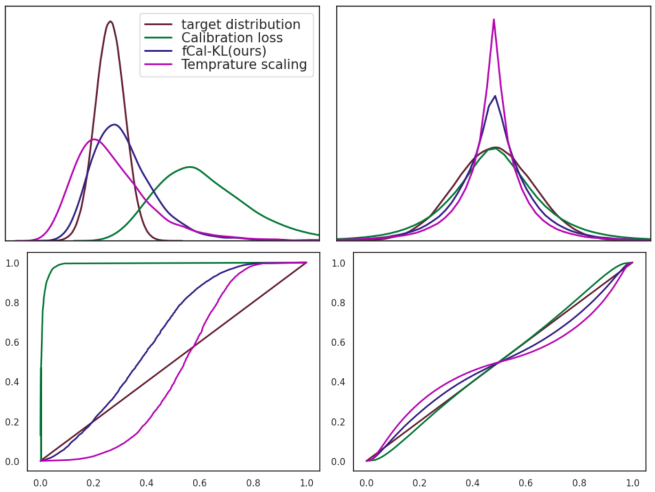

Fig. 5: **Calibration plots** on KITTI[32] object detection: *Top:* Predicted Chi-squared distributions (using hyper-constraints) and standard normal distributions from the residuals, *Bottom:* corresponding reliability diagrams for chi-square and standard normal space. f-Cal consistently yields superior calibration curves in both, chi-square and standard normal space. These curves correspond to results reported in Table I

We see in Table I that f-Cal outperforms all baselines considered. It is worth noting that we perform better than temperature scaling [7] despite this being a somewhat unfair comparison (temperature scaling leverages a large held-out calibration dataset, while we do not use any additional data). f-Cal gives well-calibrated uncertainty estimates without sacrificing the deterministic performance (more discussion of this point in Sec. VI).

E. KITTI Depth Estimation

Setup: We evaluate f-Cal on real-world robotics tasks like depth estimation and object detection (Sec. V-F). We train f-Cal and several baseline calibration techniques on the KITTI depth estimation benchmark dataset [32]. We modify the BTS model[36] for supervised depth estimation into a Bayesian Neural Network by adding a variance decoder. We evaluate the deterministic performance using SiLog and RMSE metrics and calibration using ECE and NLL.

Discussion: Through our experiments, we conclude that there is a trade-off between deterministic and calibration performances as shown in Fig. 6 (also established in [17], [7]). We can control this trade-off by varying the λ in (7). By plotting SiLog and ECE for different values of λ

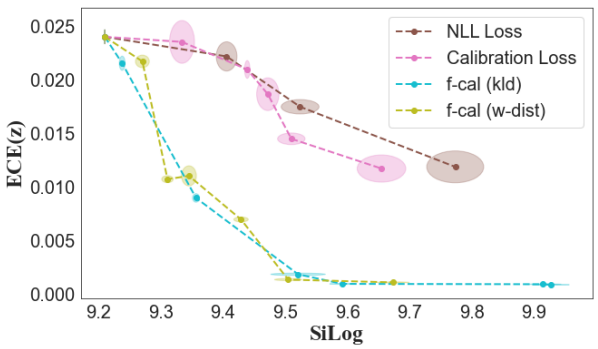

Fig. 6: **Calibration-vs-deterministic performance trade-off:** We see that this trade-off is observed for all the three calibration techniques. For similar deterministic performance f-Cal models are able to achieve smaller ECE values (i.e., better calibration).

we can analyze this trade-off for the baseline calibration techniques. We note that λ may be application dependent. To our knowledge our method is the first that enables this tradeoff to be made easily with one parameter. In Table. I-(b), for every method we select a λ which best balances between deterministic performance and calibration. For this fixed λ we run the experiment over multiple seeds and report mean scores We see that f-Cal outperforms all baselines on all calibration metrics. We also observe that unlike Bokeh (Table. I - (a)), temperature scaling struggles to calibrate uncertainties by tuning a single temperature parameter on such a large and complex task of depth estimation. We show qualitative results of depth estimation in figure 4.

F. Object detection

Setup: We now consider the task of object detection in an autonomous driving seting. We calibrate probabilistic object detectors trained on the KITTI [32] and Cityscapes [33] datasets.We use the popular Faster R-CNN [37] model with a feature pyramid network [38] and a Resnet-101 [39] backbone. We use the publicly available detectron2 [40] implementation and extend the model to output variances.

Discussion: We summarize the results of our object detection experiments in Table I-(c, d) and Fig. 5. As can be seen in Table I, we see that f-Cal variants, while having competitive regression performance (in terms of mAP), exhibit far superior calibration as reflected through ECE scores. In Fig 5, we can see through reliability plots that the baselines methods yield inferior calibration and are farther away from the ground-truth distribution. It is important to note that even though calibration loss ([7]) is able to predict a distribution which is close to being standard normal, it is still not as calibrated as the f-Cal estimates. This is reflected in the reliability diagram for the Chi-squared distribution which is much more contrastive than the curve for the standard normal distribution. Fig 5 also shows that loss attenuation yields very over-confident uncertainty predictions, which can be corroborated with qualitative results shown in Figure 3. By employing hyper-constraints over the proposed distribution, f-Cal enforces regularization at a batch level which leads to superior calibration performance.

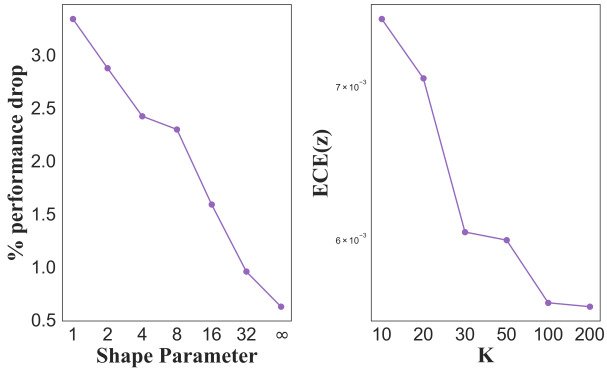

Fig. 7: **Ablation**: (left) We plot the % drop in deterministic performance compared to a deterministic model for different noise distributions. For large shape parameter, the Gamma distribution converges to a Gaussian, resulting in nearly identical performance to a deterministic model. (right) Effect of K on the performance of f-Cal, we see that as long as $K > 50$, the central limit theorem holds and we get good calibration.

VI. Discussion and Conclusion

Impact of modeling assumption: We postulate that for real-world datasets such as KITTI [32], the tradeoff in calibration and deterministic performance occurs due to poor modeling assumptions (i.e., modeling uncertainty using a distribution that is quite different from the underlying label error distribution). To investigate this, we introduce a mismatch between the true distribution, a Gamma distribution parameterized by γ, and the assumed distribution, a Gaussian distribution, on the synthetic (Bokeh) dataset (Fig. 7 (left)). For lower distributional mismatch, the performance gap between the calibrated and deterministic models is reduced. We attribute the deterministic performance drop for KITTI results to this phenomenon.

The impact of this facet of our approach is significant. This means that through experimenting with different modeling assumptions and looking at the resulting tradeoff, we may be able to infer something about the underlying noise distribution, something that is typically very hard to do.

Effect of degrees of freedom (K): We analyze how the number of degrees of freedom (K) would impact calibration performance. We train models with different values of K and measure the degree of calibration. In Fig. 7 (right), we can observe that for $K > 50$, the central limit theorem holds and we see superior calibration when compared with models trained for $K \leq 50$, when our approximation of a Gaussian distribution breaks, resulting in poor calibration. For Object detection(where thousands of proposals are being scored) and per pixel depth estimation, minibatch size $N >> K$, which allows us to effectively construct hyperconstraints.

Summary: In this work, we presented f-Cal, a principled variational inference approach to calibrate aleatoric uncertainty estimates from deep neural networks. This enables the deep neural network perceptual models to be treated as a sensor in a typical robot autonomy stack. Predicted uncertainties can be used in object-based state estimation or MPC loops. In future, we intend to extend this approach for epistemic uncertainty estimation or non-iid settings.

REFERENCES

[1] Y.-S. Shin, Y. S. Park, and A. Kim, "Direct visual slam using sparse depth for camera-lidar system," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 5144–5151. 1

[2] D. Bhatt, A. Garg, B. Gopalakrishnan, and K. M. Krishna, "Probabilistic obstacle avoidance and object following: An overlap of gaussians approach," in *2019 28th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, 2019, pp. 1–8. 1

[3] B. Gopalakrishnan, A. K. Singh, M. Kaushik, K. M. Krishna, and D. Manocha, "Prvo: Probabilistic reciprocal velocity obstacle for multi robot navigation under uncertainty," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 1089–1096. 1

[4] L. Blackmore, H. Li, and B. Williams, "A probabilistic approach to optimal robust path planning with obstacles," in *American Control Conference, 2006*. IEEE, 2006, pp. 7–pp. 1

[5] H. Song, T. Diethe, M. Kull, and P. Flach, "Distribution calibration for regression," in *International Conference on Machine Learning*. PMLR, 2019, pp. 5897–5906. 1, 2, 4, 5

[6] B. Zadrozny and C. Elkan, "Transforming classifier scores into accurate multiclass probability estimates," in *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2002, pp. 694–699. 1, 2, 4, 5

[7] D. Feng, L. Rosenbaum, C. Glaeser, F. Timm, and K. Dietmayer, "Can we trust you? on calibration of a probabilistic object detector for autonomous driving," *arXiv preprint arXiv:1909.12358*, 2019. 1, 2, 4, 5, 6

[8] Y. Gal, "Uncertainty in deep learning," *University of Cambridge*, vol. 1, no. 3, 2016. 1, 4

[9] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in *international conference on machine learning*. PMLR, 2016, pp. 1050–1059. 1, 2

[10] A. Kendall and Y. Gal, "What uncertainties do we need in bayesian deep learning for computer vision?" in *Advances in neural information processing systems*, 2017, pp. 5574–5584. 1, 2

[11] B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and scalable predictive uncertainty estimation using deep ensembles," *Advances in Neural Information Processing Systems*, vol. 30, 2017. 1, 2

[12] C. Blundell, J. Cornebise, K. Kavukcuoglu, and D. Wierstra, "Weight uncertainty in neural network," in *International Conference on Machine Learning*. PMLR, 2015, pp. 1613–1622. 1

[13] N. Tagasovska and D. Lopez-Paz, "Single-model uncertainties for deep learning," in *Advances in Neural Information Processing Systems*, 2019, pp. 6417–6428. 2

[14] D. Hendrycks and K. Gimpel, "A baseline for detecting misclassified and out-of-distribution examples in neural networks," *arXiv preprint arXiv:1610.02136*, 2016. 2

[15] M. Hein, M. Andriushchenko, and J. Bitterwolf, "Why relu networks yield high-confidence predictions far away from the training data and how to mitigate the problem," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 41–50. 2

[16] S. Liang, Y. Li, and R. Srikant, "Enhancing the reliability of out-of-distribution image detection in neural networks," in *International Conference on Learning Representations*, 2018. 2

[17] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in *International Conference on Machine Learning*. PMLR, 2017, pp. 1321–1330. 2, 4, 5

[18] D. Hendrycks, M. Mazeika, and T. Dietterich, "Deep anomaly detection with outlier exposure," in *International Conference on Learning Representations*, 2018. 2

[19] S. Mohseni, M. Pitale, J. Yadawa, and Z. Wang, "Self-supervised learning for generalizable out-of-distribution detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 04, 2020, pp. 5216–5223. 2

[20] A. Malinin and M. Gales, "Predictive uncertainty estimation via prior networks," in *Advances in Neural Information Processing Systems*, 2018, pp. 7047–7058. 2

[21] V. Sehwag, A. N. Bhagoji, L. Song, C. Sitawarin, D. Cullina, M. Chiang, and P. Mittal, "Analyzing the robustness of open-world machine learning," in *Proceedings of the 12th ACM Workshop on Artificial Intelligence and Security*, 2019, pp. 105–116. 2

[22] V. Kuleshov, N. Fenner, and S. Ermon, "Accurate uncertainties for deep learning using calibrated regression," in *International Conference on Machine Learning*. PMLR, 2018, pp. 2796–2804. 2

[23] D. Levi, L. Gispan, N. Giladi, and E. Fetaya, "Evaluating and calibrating uncertainty prediction in regression tasks," *arXiv preprint arXiv:1905.11659*, 2019. 2

[24] Y. Chung, W. Neiswanger, I. Char, and J. Schneider, "Beyond pinball loss: Quantile methods for calibrated uncertainty quantification," *arXiv preprint arXiv:2011.09588*, 2020. 2

[25] Y. H. Ho and S. M. Lee, "Calibrated interpolated confidence intervals for population quantiles," *Biometrika*, vol. 92, no. 1, pp. 234–241, 2005. 2

[26] M. Rueda, S. Martínez-Puertas, H. Martínez-Puertas, and A. Arcos, "Calibration methods for estimating quantiles," *Metrika*, vol. 66, no. 3, pp. 355–371, 2007. 2

[27] M. Taillardat, O. Mestre, M. Zamo, and P. Naveau, "Calibrated ensemble forecasts using quantile regression forests and ensemble model output statistics," *Monthly Weather Review*, vol. 144, no. 6, pp. 2375–2393, 2016. 2

[28] J. Gast and S. Roth, "Lightweight probabilistic deep networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3369–3378. 2

[29] P. Cui, W. Hu, and J. Zhu, "Calibrated reliable regression using maximum mean discrepancy," *Advances in Neural Information Processing Systems*, vol. 33, 2020. 2, 4

[30] A. Kendall, V. Badrinarayanan, and R. Cipolla, "Bayesian segnet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding," *arXiv preprint arXiv:1511.02680*, 2015. 2

[31] D. A. Nix and A. S. Weigend, "Estimating the mean and variance of the target probability distribution," in *Proceedings of 1994 ieee international conference on neural networks (ICNN'94)*, vol. 1. IEEE, 1994, pp. 55–60. 2, 4, 5

[32] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 3354–3361. 4, 5, 6

[33] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3213–3223. 5, 6

[34] M. P. Naeini, G. F. Cooper, and M. Hauskrecht, "Obtaining well calibrated probabilities using bayesian binning," in *Proceedings of the... AAAI Conference on Artificial Intelligence. AAAI Conference on Artificial Intelligence*, vol. 2015. NIH Public Access, 2015, p. 2901. 4

[35] T. Haarnoja, A. Ajay, S. Levine, and P. Abbeel, "Backprop kf: Learning discriminative deterministic state estimators," in *Advances in Neural Information Processing Systems*, 2016, pp. 4376–4384. 5

[36] J. H. Lee, M.-K. Han, D. W. Ko, and I. H. Suh, "From big to small: Multi-scale local planar guidance for monocular depth estimation," *arXiv preprint arXiv:1907.10326*, 2019. 5

[37] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99. 6

[38] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2117–2125. 6

[39] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778. 6

[40] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick, "Detectron2," https://github.com/facebookresearch/detectron2, 2019. 6