Learning to Bound: A Generative Cramér-Rao Bound

Hai Victor Habi, *Graduate Student Member, IEEE*, Hagit Messer, *Life Fellow, IEEE*, and Yoram Bresler, *Life Fellow, IEEE*

***Abstract*—The Cramér-Rao bound (CRB), a well-known lower bound on the performance of any unbiased parameter estimator, has been used to study a wide variety of problems. However, to obtain the CRB, requires an analytical expression for the likelihood of the measurements given the parameters, or equivalently a precise and explicit statistical model for the data. In many applications, such a model is not available. Instead, this work introduces a novel approach to approximate the CRB using data-driven methods, which removes the requirement for an analytical statistical model. This approach is based on the recent success of deep generative models in modeling complex, high-dimensional distributions. Using a learned normalizing flow model, we model the distribution of the measurements and obtain an approximation of the CRB, which we call Generative Cramér-Rao Bound (GCRB). Numerical experiments on simple problems validate this approach, and experiments on two image processing tasks of image denoising and edge detection with a learned camera noise model demonstrate its power and benefits.**

***Index Terms*—Generative models, normalizing flows, CRB, parameter estimation.**

I. Introduction

The Cramér-Rao Bound (CRB) is a lower bound on the variance of any unbiased parameter estimator [2], [3], [4]. It has been used in a wide variety of estimation problems such as DOA [5], TDOA [6], etc. The CRB enables to understand the fundamental limits in a given parameter estimation problem, regardless of the algorithm used. However, to obtain an applicable CRB, it is required to have an analytical expression for the likelihood of the measurements given the parameters, or equivalently a precise and explicit statistical model for the measurements. In many applications, such a model is not available. Examples include device-specific noise statistics, such as in image sensors [7], or radio frequency communications with jamming [8], or unknown channel characteristics [9].

Recently, generative models have shown state-of-the-art results in modeling complex, high-dimensional data distribution from images [10], [11], voice [12], image noise [7] and communications channels [9]. In this work, we suggest to use generative models to learn the measurement distribution from data. Then, using this generative model, we obtain an approximation to the CRB. We call this approach a *Generative Cramér-Rao Bound (GCRB)* and show conditions under which the GCRB accurately approximates the CRB. Specifically, we use a normalizing flow [13], [14] to learn a generative model for the measurement distribution. This is used, in turn, to generate samples of the gradient of the log-likelihood and obtain, as an empirical mean, an estimate of the Fisher Information Matrix (FIM). We refer to this estimate as a *Generative Fisher Information Matrix (GFIM)*. Finally, by inverting the GFIM, we obtain the GCRB.

The GCRB enables approximation of the CRB in cases when the measurement distribution is completely unknown.[1] To assess the approximation quality we provide three theoretical bounds: i) a bound on the GFIM error due to imperfect learning in terms of two well-known measures of the discrepancy between probability distributions (Total Variation Distance and Fisher Relative Information); ii) a bound on the error in the GCRB due to the use of an empirical mean to estimate the GFIM from a finite number of samples generated by the normalizing flow model;; and iii) a bound on the relative error of the GCRB, combining the effects learning and sampling errors.

To validate the GCRB, we examine two simple examples of parameter estimation with Gaussian and non-Gaussian measurement distributions, respectively. First, we show analytically that the GCRB and the CRB produce the same results under optimal conditions, i.e., assuming an invertible generative model that produces the exact measurement distribution. Second, we illustrate a realistic setup where we train a standard normalizing flow on each of the two measurement distributions to evaluate its GCRB and compare it to the corresponding CRB.

Then, to demonstrate the value of the GCRB, we use two examples from image processing: image denoising, and edge position detection, in the presence of realistic, camera-specific noise. We model the camera noise using a recently published normalizing flow model NoiseFlow [7]. With these examples,

Manuscript received 12 March 2022; revised 3 October 2022 and 12 January 2023; accepted 25 February 2023. Date of publication 13 March 2023; date of current version 20 April 2023. An earlier version of this paper is available as [DOI: 10.48550/arXiv.2203.03695]. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Yao Xie. The work of Yoram Bresler was supported in part by ARO under Grant W911NF-15-1-0479. *(Corresponding author: Yoram Bresler.)*

Hai Victor Habi and Hagit Messer are with the School of Electrical Engineering, Tel Aviv University, Tel Aviv 6139001, Israel (e-mail: haivictorh@mail.tau.ac.il; messer@eng.tau.ac.il).

Yoram Bresler is with the Department of Electrical and Computer Engineering and the Coordinated Science Laboratory, University of Illinois at, Urbana-Champaign, Urbana, IL 61801 USA (e-mail: ybresler@illinois.edu).

This article has supplementary downloadable material available at https://doi.org/10.1109/TSP.2023.3255546, provided by the authors.

Digital Object Identifier 10.1109/TSP.2023.3255546

[1] Our approach is somewhat related to the misspecified Cramér -Rao bound (MSCRB) [15] in that the MSCRB too can be evaluated without knowledge of the underlying true distribution by using data samples. However, the MSCRB provides a bound on the accuracy of estimating parameters in an assumed (mispecified) model, using measurements taken from an actual unknown distribution. Instead, we aim to determine, from data, the true model and the bound on parameter estimates in the true model.

we show two main benefits of the GCRB: (1) a lower bound for image denoising for several cameras, which provides a device-specific lower bound; and (2) we compare the GCRB lower bound on the estimation of the position and width parameters of an edge in an image corrupted by camera noise to the CRB that would be obtained using two popular noise models: white Gaussian, and Noise Level Function (NLF) noises. This experiment demonstrates that the analytical CRB with specific assumed noise models (such as the white Gaussian or even the refined NLF noise models) cannot capture the complex actual noise of image sensors and its effect on image processing performance, which is, however, successfully captured by the proposed GCRB.

The main contributions of this paper are the following.

- We introduce a Generative Cramér-Rao Bound - a data-driven approach to approximate the CRB, eliminating the need for an analytical statistical data model.
- We demonstrate the benefit of the GCRB on two real-world problems of image denoising and edge detection.
- We evaluate the approximation quality (between the CRB and the GCRB) using two simple measurement distributions.
- We provide a theoretical bound on the GCRB error due to empirical sampling and learning error.

In the spirit of reproducible research, we make the code and trained models of the generative Cramér-Rao bound available online [16].

The paper is organized as follows: the Generative Cramér-Rao Bound is developed in Section II followed by an analysis of its theoretical properties in Section III. A brief overview of normalizing flows is in Section IV. In Section V we present a set of parameter estimation examples, including simple parameter estimation in Gaussian and Non-Gaussian noise, and image processing with device-specific noise. The experimental results for the GCRB are described in Section VI, and Section VII provides discussion and conclusions. Section VIII provides detailed proofs of the theoretical results of this paper. Appendices are included in the online Supplementary Material.

II. Generative Cramer-Rao Bound

We introduce the Generative Cramér-Rao Bound (GCRB), a data-driven approach to approximate the Cramér-Rao Bound (CRB). We begin with the measurements model, the classical CRB, and problem statement. Then, we introduce our method to obtain the Generative Fisher Information Matrix (GFIM) and the GCRB using an invertiable generative model.

A. Notation

Lower-case italics a and boldface $\boldsymbol{a}$ indicate a scalar and a vector, respectively, with $\|\boldsymbol{a}\|_2$ denoting the l_2 norm. The i-th element of vector $\boldsymbol{a}$ is indicated by $[\boldsymbol{a}]_i$. Upper-case boldface $\mathbf{A}$ indicates a matrix, with its trace, determinant, transpose, Frobenius norm and spectral norm (largest singular value) denoted by $\mathrm{Tr}(\mathbf{A})$, $\det \mathbf{A}$, $\mathbf{A}^T$, $\|\mathbf{A}\|_{\mathrm{F}}$, and $\|\mathbf{A}\|$, respectively. An identity matrix of size $k \times k$ is denoted by $\mathbf{I}_k$. For symmetric matrix $\mathbf{A}$ the notations $\mathbf{A} \succ 0$ (or $\mathbf{A} \succeq 0$) mean that $\mathbf{A}$ is positive-definite (or positive semi-definite). For symmetric $\mathbf{A}$ and $\mathbf{B}$ the inequality $A \succ B$ mean that $A - B \succ 0$.

B. Data Model and Problem Statement

Consider a data model described by a random mapping, also known as a "channel," producing a random measurement $\mathrm{R}(\boldsymbol{\theta})$ from a deterministic input $\boldsymbol{\theta}$. The channel is fully characterized by the probability density function (PDF) $p_{\mathrm{R}}(\boldsymbol{r};\boldsymbol{\theta})$. Let $\boldsymbol{\theta} \in \mathbb{R}^k$ be a parameter vector, $\mathrm{R} \in \mathbb{R}^d$ the measurement vector, and $p_{\mathrm{R}}(\cdot;\boldsymbol{\theta}) : \mathbb{R}^d \rightarrow \mathbb{R}^+$ the probability density function of R for a given parameter value $\boldsymbol{\theta}$. The CRB is specified in terms of the log-likelihood (NLL) of R given $\boldsymbol{\theta}$

$$\mathrm{L}_{\mathrm{R}}\left(\boldsymbol{\theta}\right) \triangleq \log p_{\mathrm{R}}\left(\boldsymbol{r};\boldsymbol{\theta}\right)$$

and the corresponding Fisher information matrix (FIM)

$$\mathrm{F}_{\mathrm{R}}\left(\boldsymbol{\theta}\right) \triangleq \mathbb{E}_{\mathrm{R}}\left[\nabla_{\boldsymbol{\theta}}\mathrm{L}_{\mathrm{R}}\left(\boldsymbol{\theta}\right)\nabla_{\boldsymbol{\theta}}\mathrm{L}_{\mathrm{R}}\left(\boldsymbol{\theta}\right)^T\right], \tag{1}$$

where $\mathbb{E}_{\mathrm{R}}[]$ denotes the expectation with respect to R. For the CRB to apply, we assume that appropriate regularity conditions [3], [17] hold. We list below those to which we appeal in this paper explicitly, with the understanding that the remaining regularity conditions hold too.

Assumptions II.1: $p_{\mathrm{R}}(\boldsymbol{r};\boldsymbol{\theta})$ satisfies the following conditions:

A.1 For all $\boldsymbol{\theta} \in \Theta$, where Θ is an open set, the densities $p_{\mathrm{R}}(\boldsymbol{r};\boldsymbol{\theta})$ have a common support $\Upsilon = \{\boldsymbol{r} : p_{\mathrm{R}}(\boldsymbol{r};\boldsymbol{\theta}) > 0\} \subseteq \mathbb{R}^d$ that is independent of $\boldsymbol{\theta}$.
A.2 For any $\boldsymbol{r} \in \Upsilon$ and $\boldsymbol{\theta} \in \Theta$ the derivative (gradient) $\nabla_{\boldsymbol{\theta}} p_{\mathrm{R}}(\boldsymbol{r};\boldsymbol{\theta})$ with respect to $\boldsymbol{\theta}$ exists and is finite.
A.3 For all $\boldsymbol{\theta} \in \Theta$, the FIM is positive definite, $\mathrm{F}_{\mathrm{R}} \succ 0$.

Let $\hat{\boldsymbol{\theta}}(\mathrm{R})$ be an unbiased estimator of $\boldsymbol{\theta}$ from the measurement $\mathrm{R}(\boldsymbol{\theta})$ that satisfies $\mathbb{E}_{\mathrm{R}}[\|\hat{\boldsymbol{\theta}}(\mathrm{R})\|_2^2] < \infty$. Then the covariance matrix of the estimation error of any such estimator of $\boldsymbol{\theta}$ satisfies the so-called *information inequality*

$$\mathbb{E}_{\mathrm{R}}\left[\left(\hat{\boldsymbol{\theta}}(\mathrm{R}) - \boldsymbol{\theta}\right)\left(\hat{\boldsymbol{\theta}}(\mathrm{R}) - \boldsymbol{\theta}\right)^T\right] \succeq \mathrm{CRB}_{\mathrm{R}}\left(\boldsymbol{\theta}\right) \triangleq \left[\mathrm{F}_{\mathrm{R}}\left(\boldsymbol{\theta}\right)\right]^{-1}. \tag{2}$$

We wish to determine $\mathrm{CRB}_{\mathrm{R}}(\boldsymbol{\theta})$ when the channel pdf $p_{\mathrm{R}}(\boldsymbol{r};\boldsymbol{\theta})$ is unknown, and we are instead given representative data samples. We define this problem as follows.

Problem 1: Let $\Theta \subseteq \mathbb{R}^k$ be an open set. Assume that $p_{\mathrm{R}}(\boldsymbol{r};\boldsymbol{\theta})$ and $p(\boldsymbol{\theta})$ satisfy Assumptions II.1 and II.2. Given a data set $\mathcal{D} = \{\boldsymbol{\theta}_i, \boldsymbol{r}_i\}_{i=1}^{l}$ of l channel input-output samples that are independent and identically-distributed (i.i.d) as $\boldsymbol{r}_i \sim p_{\mathrm{R}}(\boldsymbol{r}_i;\boldsymbol{\theta}_i), \boldsymbol{\theta}_i \sim (\boldsymbol{\theta})$, obtain an approximation to the Cramér-Rao lower bound on the estimation of parameter $\boldsymbol{\theta} \in \Theta$ from the measurement $\mathrm{R}(\boldsymbol{\theta})$:

$$\mathrm{CRB}_{\mathrm{R}}\left(\boldsymbol{\theta}\right) \quad \forall \boldsymbol{\theta} \in \Theta.$$

The additional assumptions indicated above in Problem 1 are the following.

Assumptions II.2:

A.4 Θ is bounded set.
A.5 $p(\boldsymbol{\theta}) > \epsilon_{\Theta} > 0 \quad \forall \boldsymbol{\theta} \in \Theta$
A.6 Υ is connected set.

Assumptions II.1 are required in Problem 1 for the validity of the information inequality. Assumptions II.2 facilitate the training of the normalizing flow and generator. Specifically, A.6 facilitates the universal approximation by the generator; and A.4 and A.5 enable all $\boldsymbol{\theta} \in \Theta$ to be present in the training set with some non-vanishing probability, and limit the degree of generalization to unseen $\boldsymbol{\theta}$ required of the generator. Note that while $\boldsymbol{\theta}$ is a deterministic unknown parameter for the purposes of the CRB, $p(\boldsymbol{\theta})$ describes the *sampling* distribution of the training set $\mathcal{D}$. We will address these assumptions where relevant.

C. Method

We address Problem 1 with a two-stage approach. In the first stage, we train a conditional normalizing flow (invertible neural network) that learns the distribution of the measurements. Training of normalizing flows is a well-studied subject, and we only provide a short overview in Section IV. In the second stage, we obtain an approximation of the CRB from the trained conditional normalizing flow.

In the rest of this section, we describe how to approximate the CRB using a trained conditional normalizing flow. Let $\nu(\boldsymbol{\gamma};\boldsymbol{\theta})$ be a trained conditional normalizing flow with conditioning input $\boldsymbol{\theta}$ and data input $\boldsymbol{\gamma}$. Then $\mathrm{G}(\boldsymbol{z};\boldsymbol{\theta})$, the inverse of ν with respect to $\boldsymbol{\gamma}$, is a conditional generator with conditioning input $\boldsymbol{\theta}$ and random input $\boldsymbol{Z}$ with known and tractable distribution (usually $\boldsymbol{Z} \sim \mathcal{N}(0,\mathbf{I})$), producing the output:

$$\Gamma(\boldsymbol{\theta}) = \mathrm{G}(\boldsymbol{Z};\boldsymbol{\theta}). \tag{3}$$

While G is usually obtained directly from ν by a simple transformation and does not require separate training (see Section IV), we refer to G as a *trained generator* because it is obtained from the trained normalizing flow ν. We assume (in a sense soon to be made precise) that the trained generator simulates the random measurement process $\mathrm{R}(\boldsymbol{\theta})$ accurately, i.e. $\Gamma(\boldsymbol{\theta})$ has the same distribution as $\mathrm{R}(\boldsymbol{\theta})$.

We make the standard assumption that for each $\boldsymbol{\theta}$, $\mathrm{G}(\cdot;\boldsymbol{\theta}) : \mathbb{R}^d \mapsto \mathbb{R}^d$ is a bijection, i.e., it has an inverse $\nu(\cdot;\boldsymbol{\theta})$, and that both are differentiable functions, that is, for each $\boldsymbol{\theta}$, the mapping $\mathrm{G}(\cdot;\boldsymbol{\theta})$ is a diffeomorphism. Furthermore, for reasons explained below, we strengthen the differntiability assumption to $\mathrm{G} \in C^2$, that is the first and second order derivatives, including the mixed derivative of G w.r.t $\boldsymbol{Z}$ and $\boldsymbol{\theta}$ exist and are continuous. A trained G is a deterministic function of $\boldsymbol{\theta}$ and $\boldsymbol{Z}$, implemented as a neural network $\mathrm{G}(\cdot;\boldsymbol{\theta})$ that is invertible in its first parameter. Thanks to the randomness of $\boldsymbol{Z}$, the generative model (3) is a random mapping from $\boldsymbol{\theta}$ to $\Gamma(\boldsymbol{\theta})$.

It is important to note that when the measurement distribution is not continuous (e.g., quantized measurement), a different approach is needed for learning the CNF. The most straightforward approach is to add a prepossessing stage such dequantization [13], [18] making the measurement distribution continuous. We used this approach to apply the GCRB to the problem of frequency estimation from quantized measurements [19]. An alternative approach can be to use a CNF built for discrete data distribution [20]. However, the focus of this work is on continuous measurements, leaving extensions to discrete distributions for future work.

It follows, using the standard formula of transformation of random variables, that the probability density function of $\Gamma(\boldsymbol{\theta})$ is

$$p_{\Gamma}(\boldsymbol{\gamma};\boldsymbol{\theta}) = p_{\boldsymbol{Z}}(\nu(\boldsymbol{\gamma};\boldsymbol{\theta})) \left|\det \mathbf{J}_{\nu}(\boldsymbol{\gamma};\boldsymbol{\theta})\right|, \tag{4}$$

where $\mathbf{J}_{\nu}(\boldsymbol{\gamma};\boldsymbol{\theta}) = \frac{\partial \nu(\boldsymbol{\gamma};\boldsymbol{\theta})}{\partial \boldsymbol{\gamma}}$ is the Jacobian matrix of the transformation $\nu(\boldsymbol{\gamma};\boldsymbol{\theta})$ with respect to $\boldsymbol{\gamma}$. Since both G and ν are known functions and the pdf of $\boldsymbol{Z}$ is known (standard normal), in principle, the pdf $p_{\Gamma}(\boldsymbol{\gamma};\boldsymbol{\theta})$ can be determined.

Given the trained normalizing flow ν and the corresponding generator G, we compute the GCRB as follows. First using (4) we determine (as detailed in Appendix A1) the so-called score vector

$$\begin{aligned} \boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{z}) &\triangleq \nabla_{\boldsymbol{\theta}} \log p_{\Gamma}(\boldsymbol{\gamma};\boldsymbol{\theta})\big|_{\boldsymbol{\gamma}=\mathrm{G}(\boldsymbol{z};\boldsymbol{\theta})} \\ &= \nabla_{\boldsymbol{\theta}} \log\left[p_{\boldsymbol{Z}}(\nu(\boldsymbol{\gamma};\boldsymbol{\theta})) \left|\det \mathbf{J}_{\nu}(\boldsymbol{\gamma};\boldsymbol{\theta})\right|\right]\big|_{\boldsymbol{\gamma}=\mathrm{G}(\boldsymbol{z};\boldsymbol{\theta})} \qquad (5) \\ &= \left.\frac{\partial \nu(\boldsymbol{\gamma};\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}^{\mathrm{T}}\right|_{\boldsymbol{\gamma}=\mathrm{G}(\boldsymbol{z};\boldsymbol{\theta})} \times \nabla_{\boldsymbol{z}} \log p_{\boldsymbol{Z}}(\boldsymbol{z}) + \boldsymbol{k}(\boldsymbol{\gamma},\boldsymbol{\theta})\big|_{\boldsymbol{\gamma}=\mathrm{G}(\boldsymbol{z};\boldsymbol{\theta})}, \qquad (6) \end{aligned}$$

$$\text{where} \quad [\boldsymbol{k}(\boldsymbol{\gamma},\boldsymbol{\theta})]_i = \mathrm{Tr}\left(\mathbf{J}_{\nu}^{-1}(\boldsymbol{\gamma};\boldsymbol{\theta}) \frac{\partial \mathbf{J}_{\nu}(\boldsymbol{\gamma};\boldsymbol{\theta})}{\partial [\boldsymbol{\theta}]_i}\right), \tag{7}$$

where $\frac{\partial \nu(\boldsymbol{\gamma};\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}$ is the Jacobian matrix of ν w.r.t $\boldsymbol{\theta}$, and in (7) $\frac{\partial \mathbf{J}_{\nu}(\boldsymbol{\gamma};\boldsymbol{\theta})}{\partial [\boldsymbol{\theta}]_i}$ is a derivative matrix of the Jacobian matrix $\mathbf{J}_{\nu}$ w.r.t the i-th component of $\boldsymbol{\theta}$. Note that to evaluate the score vector for a given $\boldsymbol{z}$ both G and ν are used. We therefore refer to (6) as a hybrid score vector. As an alternative, we show in Appendix A2 an equivalent form that only uses the generator G.

To perform the computation in (6)–(7) we need to be able to evaluate the following derivatives: ν w.r.t $\boldsymbol{\theta}$, ν w.r.t $\boldsymbol{\gamma}$, and a mixed second derivative of ν w.r.t to $\boldsymbol{\gamma}$ and $\boldsymbol{\theta}$. As elaborated in Appendix A2, this motivates the differentiability conditions imposed on G above. We also require that the log-likelihood of the base distribution $p_{\boldsymbol{z}}(\boldsymbol{z})$ be differentiable, which is satisfied by the Gaussian distribution. Moreover, $\mathbf{J}_{\nu}$ should be invertible, which is usually guaranteed by common layer structures in the CNF literate by combining design of the layer and training loss objective (23). The invertibility condition $\det \mathbf{J}_{\nu}(\boldsymbol{\gamma};\boldsymbol{\theta}) \neq 0$ also enables stable training.

As a practical matter, because ν and G are implemented as a neural networks, the required derivatives of ν and G w.r.t to their respective inputs $\boldsymbol{\gamma}$, $\boldsymbol{z}$ and $\boldsymbol{\theta}$ can be easily evaluated in common deep learning frameworks such as PyTorch [21], TensorFlow [48], etc.

Given the score vector, we compute the Generative Fisher Information Matrix (GFIM):

$$\mathrm{F}_{\mathrm{G}}(\boldsymbol{\theta}) \triangleq \mathbb{E}_{\boldsymbol{Z}}\left[\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{Z})\, \boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{Z})^{T}\right]. \tag{8}$$

In practice, to avoid integration in (8), the expected value with respect to $\boldsymbol{Z}$ is estimated as an empirical mean by sampling from

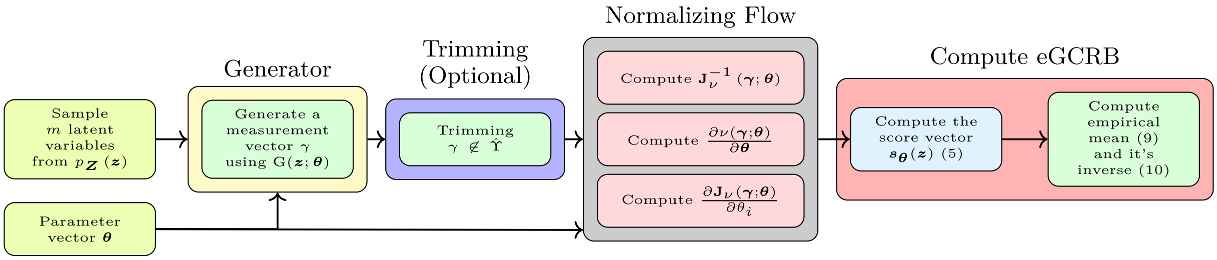

Normalizing Flow
Generator
Trimming (Optional)
Compute eGCRB
Sample m latent variables from $p_{\boldsymbol{Z}}(\boldsymbol{z})$
Generate a measurement vector γ using $\mathrm{G}(\boldsymbol{z};\boldsymbol{\theta})$
Trimming $\gamma\notin\hat{\Upsilon}$
Compute $\mathbf{J}_{\nu}^{-1}(\boldsymbol{\gamma};\boldsymbol{\theta})$
Compute $\frac{\partial\nu(\boldsymbol{\gamma};\boldsymbol{\theta})}{\partial\boldsymbol{\theta}}$
Compute $\frac{\partial\mathbf{J}_{\nu}(\boldsymbol{\gamma};\boldsymbol{\theta})}{\partial\theta_i}$
Compute the score vector $\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{z})$ (5)
Compute empirical mean (9) and it's inverse (10)
Parameter vector $\boldsymbol{\theta}$

Fig. 1. Generative Cramér Rao bound using normalizing flow.

$p_{\boldsymbol{Z}}$. The result is an empirical Generative Fisher Information Matrix (eGFIM) that is computed as

$$\overline{\mathrm{F}_{\mathrm{G}}}\left(\boldsymbol{\theta}\right)\triangleq\frac{1}{m}\sum_{i=1}^{m}\boldsymbol{s}_{\boldsymbol{\theta}}\left(\boldsymbol{z}_i\right)\boldsymbol{s}_{\boldsymbol{\theta}}\left(\boldsymbol{z}_i\right)^T,\tag{9}$$

using m samples $\boldsymbol{z}_i\sim p_{\boldsymbol{Z}}$. Finally, we approximate the CRB using the empirical estimate of the GCRB (eGCRB) $\overline{\mathrm{GCRB}_{\mathrm{G}}}$ associated with generator G by

$$\overline{\mathrm{GCRB}_{\mathrm{G}}}\left(\boldsymbol{\theta}\right)=\overline{\mathrm{F}_{\mathrm{G}}}\left(\boldsymbol{\theta}\right)^{-1}.\tag{10}$$

Given the trained neural networks G and ν, the computation of $\overline{\mathrm{GCRB}_{\mathrm{G}}}$ for a given value of $\boldsymbol{\theta}$ is illustrated in Fig. 1. It involves m uses of the neural networks G (to generate $\boldsymbol{\gamma}$) and ν (to generate the various derivatives) and the simple computations in (6)–(7), (9), and (10), so can be computationally cheap.

In the rest of this subsection, we address a modification of the GCRB to improve the learning of the generator in the practical situation of a finite training data set. Because some regions of the measurement space Υ may have few or no training samples, we need to bound the region $\hat{\Upsilon}$ where generated samples can be trusted. We define this region by its assumed properties.

Assumptions II.3. (Trusted Region):

A.7 $\hat{\Upsilon}\subseteq\Upsilon$ is a connected and closed and bounded (hence compact) set.

A.8 $\hat{\Upsilon}$ is large enough that for some chosen $\epsilon_r\geq 0$

$$\int_{\boldsymbol{r}\notin\hat{\Upsilon}}p_{\mathrm{R}}\left(\boldsymbol{r};\boldsymbol{\theta}\right)d\boldsymbol{r}\leq\epsilon_r\quad\forall\boldsymbol{\theta}\in\Theta.$$

A.9 $p_{\mathrm{R}}(\boldsymbol{r};\boldsymbol{\theta})>\epsilon>0\quad\forall\boldsymbol{r}\in\hat{\Upsilon}.$

To ensure that the computation of the GCRB is performed using a sample generated on the trusted region, we add an optional trimming step that removes un-trusted generated samples $\boldsymbol{\gamma}=\mathrm{G}(\boldsymbol{z})\notin\hat{\Upsilon}$. The trimming step ensures that only values of $\boldsymbol{z}$ that correspond to trusted $\boldsymbol{\gamma}\in\hat{\Upsilon}$ are used in the computation of GCRB. By Assumptions II.3 the effect of this trimming on the approximation quality should be a negligible. Furthermore, Assumption A.9 enables all $\boldsymbol{r}\in\hat{\Upsilon}$ to be present in the training set with some non-vanishing probability. Algorithm 1 describes the evaluation of the eGCRB, with the trimming step included.

Algorithm 1: eGCRB Sampling.

```
Require: G, $\nu$, $B$, $\hat{\boldsymbol{r}}$, $m$, $\boldsymbol{\theta}$
  $\mathrm{S}\leftarrow\emptyset$
  while $|\mathrm{S}|<m$ do
  $\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})$
    $\boldsymbol{\gamma}=\mathrm{G}(\boldsymbol{z};\boldsymbol{\theta})$          ▷ Generator Step
    if $\boldsymbol{\gamma}\in\hat{\Upsilon}$ then          ▷ Timming Step
    $\hat{\boldsymbol{s}}=\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{z})$          ▷ Compute score vector (5).
      $\mathrm{S}\leftarrow\mathrm{S}\cup\{\hat{\boldsymbol{s}}\}$          ▷ Append to $\hat{\boldsymbol{s}}$ to score set.
    end if
  end while
  $\overline{\mathrm{F}_{\mathrm{G}}}(\boldsymbol{\theta})=\frac{1}{m}\sum_{\hat{\boldsymbol{s}}\in\mathrm{S}}\hat{\boldsymbol{s}}\hat{\boldsymbol{s}}^T$.          ▷ Compute eGFIM
  $\overline{\mathrm{GCRB}_{\mathrm{G}}}(\boldsymbol{\theta})=\overline{\mathrm{F}_{\mathrm{G}}}(\boldsymbol{\theta})^{-1}$          ▷ Invert eGFIM to obtain
  eGCRB
```

The trimming step, in the spirit of standard trimmed mean computation in robust statistics [22], is a kind of outlier removal step, which is a well-studied but also active field of research (cf. [23], [24] and the references therein). We propose a simple heuristic trimming criterion; a more refined criterion may improve the eGCRB accuracy when only limited training data is available. The trimming process consist of two steps. First, we evaluate the mean $\bar{\boldsymbol{r}}$ and an upper bound B on the spread of $\boldsymbol{r}$ in the training set $\mathcal{D}$:

$$\bar{\boldsymbol{r}}=\frac{1}{|\mathcal{D}|}\sum_i\boldsymbol{r}_i,$$

$$B=\max_i\|\boldsymbol{r}_i-\bar{\boldsymbol{r}}\|.$$

Then, the trusted set is defined as

$$\hat{\Upsilon}=\{\gamma\in\Upsilon:\|\gamma-\bar{\boldsymbol{r}}\|\leq B\}.$$

This trimming is designed to exclude samples $\boldsymbol{\gamma}=\mathrm{G}(\boldsymbol{z})$ generated in regions where no training samples were available to train the normalizing flow. This will reduce the requirement of the normalizing flow and generator to extrapolate during inference outside the coverage of the training set. Note that thanks to the adaptivity of $\hat{\Upsilon}$ to the training set, as the size of the training set $|\mathcal{D}|\to\infty$, the "unrepresented probability" vanishes: $\epsilon_r\to 0$.

III. Theoretical Properties

This section addresses three questions: (i) What are the errors introduced into the GFIM by learning the measurement distribution? (ii) What is the error introduced by using an empirical mean eGFIM to estimate the GFIM? (iii) When does the approximation eGCRB to the CRB computed using the learned generative model in the proposed approach converge to the correct CRB? Our key assumption in (iii) will be that the generative model is expressive enough and the training data set has sufficient size and diversity of values of $\boldsymbol{\theta}$ and R that the training is successful, resulting in a generative model that simulates the random mapping $\mathrm{R}(\boldsymbol{\theta})$.

A. Learning Error

In this part we address the error induced by replacing the true measurement distribution p_R by the learned distribution p_Γ with trimming of the generator, meaning that $\hat{\Upsilon}$ is a strict subset of Υ. We account for the deviation between p_Γ and p_R on their common support $\hat{\Upsilon} \bigcap \Upsilon$, as well as on the truncated region $\Upsilon \setminus \hat{\Upsilon}$ where $p_\Gamma = 0$.

Define

$$\hat{\mathrm{F}}_{\mathrm{G}}(\boldsymbol{\theta}) \triangleq \int_{\hat{\mathcal{Z}}} \boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{z})\, \boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{z})^T p_{\boldsymbol{Z}}(\boldsymbol{z})\, d\boldsymbol{z}, \tag{11}$$

as the result of the GFIM calculation over the trimmed latent variable set $\hat{\mathcal{Z}} = \{\boldsymbol{z} : \mathrm{G}(\boldsymbol{z};\boldsymbol{\theta}) \in \hat{\Upsilon}\}$. We begin by introducing bounds on the generated and the true score vectors.

Lemma III.1: Let $\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{z})$ be a score vector computed using a trimmed and differentiable $\mathrm{G} \in C^2$ generator G and it's inverse ν. Then $\|\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{z})\|_2 \leq C_s(\boldsymbol{\theta}) < \infty\ \forall \boldsymbol{\theta} \in \Theta, \forall \boldsymbol{z} \in \hat{\mathcal{Z}}$.

This result (proved in Section VIII-A) shows that the score vector is bounded in 2-norm. Next, we introduce an additional assumption, that the true measurement distribution too has a bounded score vector.

Assumption III.1. (Bounded True Score Vector):

$$\|\nabla_{\boldsymbol{\theta}} \mathrm{L}_{\mathrm{R}}(\boldsymbol{\theta})\|_2 \leq C_{\mathrm{R}}(\boldsymbol{\theta}) < \infty \quad \forall \boldsymbol{\theta} \in \Theta, \forall \boldsymbol{r} \in \Upsilon. \tag{12}$$

Note that this assumption is a slightly more restrictive version of Assumption A.2.

Then we have our main results.

Theorem III.2. (GFIM Learning Errors): Let G be a normalizing flow trained on $\mathrm{R} \sim p_{\mathrm{R}}$, where p_{R} has a bounded score vector (Assumption III.1). Then

$$\left\|\mathrm{F}_{\mathrm{R}}(\boldsymbol{\theta}) - \hat{\mathrm{F}}_{\mathrm{G}}(\boldsymbol{\theta})\right\| \leq \eta(\boldsymbol{\theta}) \qquad \forall \boldsymbol{\theta} \in \Theta, \tag{13a}$$

$$\eta(\boldsymbol{\theta}) \triangleq 2C_{\mathrm{R}}^2(\boldsymbol{\theta})\, \mathrm{TV}(p_{\mathrm{R}}, p_\Gamma; \boldsymbol{\theta}) + 2\left(\left\|\hat{\mathrm{F}}_{\mathrm{G}}(\boldsymbol{\theta})\right\| \mathrm{I}_{\mathrm{F}}(p_\Gamma, p_{\mathrm{R}}; \boldsymbol{\theta})\right)^{1/2} + \mathrm{I}_{\mathrm{F}}(p_\Gamma, p_{\mathrm{R}}; \boldsymbol{\theta}) \tag{13b}$$

where

$$\mathrm{TV}(p_\Gamma, p_{\mathrm{R}}; \boldsymbol{\theta}) \triangleq \frac{1}{2}\int_{\Upsilon} |p_\Gamma(\boldsymbol{r};\boldsymbol{\theta}) - p_{\mathrm{R}}(\boldsymbol{r};\boldsymbol{\theta})|\, d\boldsymbol{r}$$

is the total variation distance [25] between the PDFs p_Γ and p_{R}, and

$$\mathrm{I}_{\mathrm{F}}(p_\Gamma, p_{\mathrm{R}}; \boldsymbol{\theta}) \triangleq \int_{\Upsilon} p_\Gamma(\boldsymbol{r};\boldsymbol{\theta}) \left\|\nabla_{\boldsymbol{\theta}} \log\left(\frac{p_\Gamma(\boldsymbol{r};\boldsymbol{\theta})}{p_{\mathrm{R}}(\boldsymbol{r};\boldsymbol{\theta})}\right)\right\|_2^2 d\boldsymbol{r}$$

is the Fisher relative information [26], [27] between p_Γ and p_{R}.

Theorem III.2 (which is proved in Section VIII-B) bounds the error in learning the FIM in term of the total variation (TV) distance and the Fisher relative information between the true and the learned measurement distributions, p_{R} and p_Γ. The TV distance term captures both the trimming error and sample generation error, whereas the Fisher relative information term accounts for the errors in learning the derivative of $\log p_{\mathrm{R}}$, namely the score vector. Both $\mathrm{TV}(p_\Gamma, p_{\mathrm{R}}; \boldsymbol{\theta})$ and $\mathrm{I}_{\mathrm{F}}(p_\Gamma, p_{\mathrm{R}}; \boldsymbol{\theta})$ are non-negative and vanish if and only if $p_\Gamma(\boldsymbol{r};\boldsymbol{\theta}) = p_{\mathrm{R}}(\boldsymbol{r};\boldsymbol{\theta}) \quad \forall \boldsymbol{r} \in \Upsilon$. Furthermore, both metrics are bounded; the TV distance by definition, and the Fisher relative information is bounded as a direct consequence of Lemma III.1 and Assumption III.1

The impact of the learning error on the GCRB is given by the following corollary, where we use Assumption A.3, that the FIM is positive definite, to provide conditions in terms of its strictly positive smallest eigenvalue $\lambda_{\min}(\mathrm{F}_{\mathrm{R}}(\boldsymbol{\theta})) > 0$.

Corollary III.2.1: Suppose that in addition to the assumptions in Theorem III.2, we have $\eta(\boldsymbol{\theta}) < \lambda_{\min}(\mathrm{F}_{\mathrm{R}}(\boldsymbol{\theta}))$. Then

$$\|\hat{\mathrm{F}}_{\mathrm{G}}(\boldsymbol{\theta})^{-1}\| \leq [\lambda_{\min}(\mathrm{F}_{\mathrm{R}}(\boldsymbol{\theta})) - \eta(\boldsymbol{\theta})]^{-1} \tag{14a}$$

$$\begin{aligned}\|\mathrm{CRB}_{\mathrm{R}}(\boldsymbol{\theta}) - \mathrm{GCRB}(\boldsymbol{\theta})\| &= \|\mathrm{F}_{\mathrm{R}}(\boldsymbol{\theta})^{-1} - \hat{\mathrm{F}}_{\mathrm{G}}(\boldsymbol{\theta})^{-1}\| \\ &\leq \|\mathrm{F}_{\mathrm{R}}(\boldsymbol{\theta})^{-1}\| \cdot \|\hat{\mathrm{F}}_{\mathrm{G}}(\boldsymbol{\theta})^{-1}\| \cdot \eta(\boldsymbol{\theta})\end{aligned} \tag{14b}$$

Note that (14a) in Corollary III.2.1 (which is proved in Section VIII-C) is a guarantee that the GFIM is positive definite, i.e., the GCRB is finite, if the condition of the Corollary is satisfied. The second result, bounds the deviation of the GCRB from the CRB in terms of the FIM learning error.

To help further interpret Corollary III.2.1, consider the relative (normalized) learning error in the FIM, $\frac{\eta(\boldsymbol{\theta})}{\|\mathrm{F}_{\mathrm{R}}(\boldsymbol{\theta})\|}$. The condition of the Corollary then becomes $\kappa(F_{\mathrm{R}}(\boldsymbol{\theta}))\frac{\eta(\boldsymbol{\theta})}{\|\mathrm{F}_{\mathrm{R}}(\boldsymbol{\theta})\|} < 1$, where $\kappa(F_{\mathrm{R}}(\boldsymbol{\theta})) \triangleq \|F_{\mathrm{R}}(\boldsymbol{\theta})\| \cdot \|F_{\mathrm{R}}(\boldsymbol{\theta})^{-1}\|$ is the 2-norm condition number of $F_{\mathrm{R}}(\boldsymbol{\theta})$ (which is also equal to the condition number of $\mathrm{CRB}_{\mathrm{R}}(\boldsymbol{\theta})$). Hence, the requirement of the corollary on the learning error $\eta(\boldsymbol{\theta})$ is easy to satisfy for a well-conditioned FIM (or CRB), and becomes more demanding with increasing condition number.

Next, consider the case of small learning error, $\frac{\eta(\boldsymbol{\theta})}{\|\mathrm{F}_{\mathrm{R}}(\boldsymbol{\theta})\|} \ll 1$. Then, by standard arguments, (14b) yields

$$\frac{\|\mathrm{CRB}_{\mathrm{R}}(\boldsymbol{\theta}) - \mathrm{GCRB}(\boldsymbol{\theta})\|}{\|\mathrm{CRB}_{\mathrm{R}}(\boldsymbol{\theta})\|} \leq \kappa(F_{\mathrm{R}}(\boldsymbol{\theta}))\frac{\eta(\boldsymbol{\theta})}{\|\mathrm{F}_{\mathrm{R}}(\boldsymbol{\theta})\|} + \epsilon_3 \tag{15}$$

where $\epsilon_3 = O((\frac{\eta(\boldsymbol{\theta})}{\|\mathrm{F}_{\mathrm{R}}(\boldsymbol{\theta})\|})^2)$, that is, the inequality is dominated by the first term on the right hand side, with the remainder ϵ_3 of second order in the relative FIM learning error $\frac{\eta(\boldsymbol{\theta})}{\|\mathrm{F}_{\mathrm{R}}(\boldsymbol{\theta})\|}$, and hence negligible. By (15), the relative error in the GCRB is bounded by the relative GFIM learning error, scaled by the condition number of the FIM. Again, the better the conditioning of the FIM, the lower the sensitivity of the GCRB to the GFIM learning error.

Furthermore, using (13b) to express the relative error in learning the GFIM for the case of small learning error yields the simplified expression

$$\frac{\eta(\boldsymbol{\theta})}{\|\mathrm{F}_{\mathrm{R}}(\boldsymbol{\theta})\|} \approx 2\frac{C_{\mathrm{R}}^2(\boldsymbol{\theta})}{\|\mathrm{F}_{\mathrm{R}}(\boldsymbol{\theta})\|}\mathrm{TV}(p_{\mathrm{R}}, p_\Gamma; \boldsymbol{\theta}) + 2\sqrt{\tilde{\mathrm{I}}_{\mathrm{F}}(p_\Gamma, p_{\mathrm{R}}; \boldsymbol{\theta})} \tag{16}$$

where $\tilde{\mathrm{I}}_{\mathrm{F}}(p_\Gamma, p_{\mathrm{R}}; \boldsymbol{\theta}) \triangleq \mathrm{I}_{\mathrm{F}}(p_\Gamma, p_{\mathrm{R}}; \boldsymbol{\theta})/\|\mathrm{F}_{\mathrm{R}}(\boldsymbol{\theta})\|$ is the Fisher relative information between p_Γ and p_{R} normalized by the Fisher information for p_{R}. Now all terms in (16) are dimensionless and naturally normalized. (Recall that $C_{\mathrm{R}}^2(\boldsymbol{\theta})$ and

$\|\mathrm{F}_{\mathrm{R}}(\boldsymbol{\theta})\|$ scale similarly with the magnitude of the true score vector.)

B. Sampling Error

Here, we study the effects of finite number of samples in (9) on the accuracy of the estimation of $\hat{\mathrm{F}}_{\mathrm{G}}(\boldsymbol{\theta})$, by deriving an upper bound on the error.

Theorem III.3. (Sampling Error): Let $\overline{\mathrm{F}_{\mathrm{G}}}(\boldsymbol{\theta})^{-1}$ be the eGCRB computed using a trimmed generator $\mathrm{G} \in \mathcal{C}^2$ and its inverse ν – the corresponding normalizing flow, trained on R. Assume that Assumptions II.3, III.1 hold, and $\hat{\mathrm{F}}_{\mathrm{G}}(\boldsymbol{\theta}) \succ 0 \;\; \forall \boldsymbol{\theta} \in \Theta$, which implies that $\|\hat{\mathrm{F}}_{\mathrm{G}}(\boldsymbol{\theta})^{-1}\| \le \mathrm{C}_{\mathrm{G}}(\boldsymbol{\theta})$. Then there exist absolute constants $C_1, C_2 > 0$ such that provided that $m > C_1(1+u)\mathrm{C}_{\mathrm{G}}(\boldsymbol{\theta})^2$, for any $u > 0$ we have, with probability at least $1 - \exp(-u)$:

$$\left\|\overline{\mathrm{GCRB}_{\mathrm{G}}}\left(\boldsymbol{\theta}\right) - \hat{\mathrm{F}}_{\mathrm{G}}\left(\boldsymbol{\theta}\right)^{-1}\right\|_{\mathrm{F}} \le \mathrm{B}_{\mathrm{s}}\left(\boldsymbol{\theta}\right), \tag{17}$$

where $\mathrm{B}_{\mathrm{s}}(\boldsymbol{\theta}) \triangleq C_2\|\hat{\mathrm{F}}_{\mathrm{G}}(\boldsymbol{\theta})^{-1}\|^2\mathrm{C}_{\mathrm{s}}(\boldsymbol{\theta})^2\sqrt{\frac{1+u}{m}}$.

Theorem III.3 (proved in Section VIII-D) is based on a bound for the precision matrix [28] and properties of the score vector. This result shows that the deviation of the eGCRB from the GCRB is bounded in terms of the norm of the generator score vector the GCRB itself, and the number of samples. Importantly, Theorem III.3 shows that the error decreases (at the best possible rate) as the number m of samples increases.

C. Convergence of the eGCRB

To study the convergence of the eGCRB to the true CRB, we first bound the relative error of approximating $\mathrm{CRB}_{\mathrm{R}}$ by $\overline{\mathrm{GCRB}_{\mathrm{G}}}(\boldsymbol{\theta})$ due to both empirical mean and learning errors using Theorems III.3 and III.2 and Corollary III.2.1. Then we discuss the conditions under which the eGCRB convergence to the true CRB.

Corollary III.3.1: Suppose the assumptions in Theorem III.2 and Assumptions II.3, III.1 hold, and $\kappa(\mathrm{CRB}_{\mathrm{R}}(\boldsymbol{\theta}))\frac{\eta(\theta)}{\|\mathrm{F}_{\mathrm{R}}(\boldsymbol{\theta})\|} < 1$. Then $\|\hat{\mathrm{F}}_{\mathrm{G}}(\boldsymbol{\theta})^{-1}\| \le \mathrm{C}_{\mathrm{G}}(\boldsymbol{\theta})$ and there exist absolute constants $C_1, C_2 > 0$ such that provided that $m > C_1(1+u)\mathrm{C}_{\mathrm{G}}(\boldsymbol{\theta})^2$, for any $u > 0$ we have, with probability at least $1 - \exp(-u)$:

$$\begin{aligned}\mathrm{RE}\left(\boldsymbol{\theta}\right) &\triangleq \frac{\left\|\overline{\mathrm{GCRB}_{\mathrm{G}}}\left(\boldsymbol{\theta}\right) - \mathrm{CRB}_{\mathrm{R}}\left(\boldsymbol{\theta}\right)\right\|}{\left\|\mathrm{CRB}_{\mathrm{R}}\left(\boldsymbol{\theta}\right)\right\|} \\ &\le \left\|\hat{\mathrm{F}}_{\mathrm{G}}\left(\boldsymbol{\theta}\right)^{-1}\right\|\left[\tilde{\mathrm{B}}_{\mathrm{s}} + \eta\left(\boldsymbol{\theta}\right)\right],\end{aligned} \tag{18}$$

where

$$\mathrm{C}_{\mathrm{G}}\left(\boldsymbol{\theta}\right) \triangleq \frac{\left\|\mathrm{CRB}_{\mathrm{R}}\left(\boldsymbol{\theta}\right)\right\|}{1 - \kappa(\mathrm{CRB}_{\mathrm{R}}\left(\boldsymbol{\theta}\right))\frac{\eta(\theta)}{\|\mathrm{F}_{\mathrm{R}}(\boldsymbol{\theta})\|}}, \tag{19a}$$

$$\tilde{\mathrm{B}}_{s}\left(\boldsymbol{\theta}\right) \triangleq C_2\frac{\left\|\hat{\mathrm{F}}_{\mathrm{G}}\left(\boldsymbol{\theta}\right)^{-1}\right\|}{\left\|\mathrm{F}_{\mathrm{R}}\left(\boldsymbol{\theta}\right)^{-1}\right\|}\mathrm{C}_{\mathrm{s}}\left(\boldsymbol{\theta}\right)^2\sqrt{\frac{1+u}{m}}. \tag{19b}$$

In Corollary III.3.1 (proved in Section VIII-E), we observe that the relative error in approximating the CRB using the proposed approach decreases with decreasing FIM learning error and increasing number of samples used to compute the empirical mean in the evaluation of the GFIM. Similar to the case of Corollary III.2.1, the interpretation of the result is facilitated by considering the case of normalized learning error bounded by $\kappa_{\mathrm{R}}\frac{\eta(\theta)}{\|\mathrm{F}_{\mathrm{R}}(\boldsymbol{\theta})\|} < 0.5$, where $\kappa_{\mathrm{R}} \triangleq \kappa(\mathrm{CRB}_{\mathrm{R}}(\boldsymbol{\theta}))$. (This is only slightly more stringent than the requirement in Corollary III.3.1.) Then, as shown in Appendix D.1, the following exact bound holds.

$$\mathrm{RE}\left(\boldsymbol{\theta}\right) \le \kappa_{\mathrm{R}}\left(4C_2\frac{C_{\mathrm{R}}^2\left(\boldsymbol{\theta}\right)}{\left\|\mathrm{F}_{\mathrm{R}}\left(\boldsymbol{\theta}\right)\right\|}\sqrt{\frac{1+u}{m}} + \frac{\eta(\theta)}{\left\|\mathrm{F}_{\mathrm{R}}\left(\boldsymbol{\theta}\right)\right\|}\right) \tag{20}$$

The bound on the relative error in the eGCRB in (20) (which is proved in Appendix D1), is dimensionless, and shows clearly the effect of the condition number κ_{R} of the CRB, the number of samples used to compute the empirical mean, and the normalized FIM learning error.

The eGCRB relative error (20) consists of two terms. The first is the sampling error, which can be made arbitrarily small by using a large enough m. The second term is the learning error, which we address next.

Assumption III.2: [Existence of well-trained generator] Let $\mathcal{G}$ be the set of all generators representable by the chosen architecture of the normalizing flow network, and define G^* to be an optimal generator in the sense that if $\Gamma^*(\boldsymbol{\theta}) = \mathrm{G}^*(\boldsymbol{Z};\boldsymbol{\theta})$ then $\Gamma^*(\boldsymbol{\theta})$ is distributed the same as the measurement distribution R, that is, $\Gamma^*(\boldsymbol{\theta}) \sim p_{\mathrm{R}}(\gamma;\boldsymbol{\theta}) \;\; \forall \boldsymbol{\theta} \in \Theta$. Then we assume that:

$$\mathrm{G}^* \in \mathcal{G}, \tag{21}$$

and the dataset $\mathcal{D}$ is rich enough such that the training is successful and results in $\mathrm{G} = \mathrm{G}^*$ which yields:

$$\mathrm{TV}\left(p_{\mathrm{R}}, p_{\Gamma};\boldsymbol{\theta}\right) = 0, \tag{22a}$$

$$\mathrm{I}_{\mathrm{F}}\left(p_{\Gamma}, p_{\mathrm{R}};\boldsymbol{\theta}\right) = 0. \tag{22b}$$

Two conditions are required for a well-trained generator (Assumption III.2) to be realizable: (i) the set of generators $\mathcal{G}$ representable by the chosen architecture of the normalizing flow network contains the optimal generator G^* (21); and (ii) the generator can be trained to achieve this approximation using the training data.

Assuming that Condition (i) holds, then Condition (ii) can be satisfied, i.e., a well-trained generator is realizable on a trusted region (Assumption II.3) in the limit of infinite training data set if Assumptions A.1 and A.6 on the measurement distribution $p_{\mathrm{R}}(\boldsymbol{r};\boldsymbol{\theta})$ and Assumption A.4 on the training set distribution are satisfied. Recall that $\mathrm{TV}(p_{\mathrm{R}}, p_{\Gamma};\boldsymbol{\theta})$ includes the trusted region truncation error, which must vanish for $\mathrm{TV}(p_{\mathrm{R}}, p_{\Gamma};\boldsymbol{\theta}) = 0$. This happens automatically when the true measurement distribution is bounded $\hat{\Upsilon} = \Upsilon$, or thanks to the proposed adaptive trimming criterion in the limit of an infinite training set, $\epsilon_r \to 0$ as $|\mathcal{D}| \to \infty$.

Moreover, Condition (i) can be addressed in several ways. Available prior knowledge of the problem can be incorporated into the chosen architecture of the normalizing flow (e.g., Noise-Flow [7], SineFlow [19]) to help satisfy Condition (i). Because the very notion of parameter estimation requires some modeling of the measurements, such prior knowledge is typically available in parameter estimation problems. Furthermore, following the standard practice in deep learning, one can increase the representation power of the network by increasing its size and number of trainable parameters. In the extreme case of no domain knowledge, this involves reliance on the ability of NF to provide a universal approximation.

However, the question of universal approximations is an active research area, with recent results [14], [29], [30], [31], [32] showing that for certain architectural choices and under some additional assumptions, normalizing flows can provide universal approximations with arbitrarily small error. As the currently available universal approximation conditions are sufficient conditions, we expect that ongoing research will result in a further relaxation of the conditions and a larger variety of architectural choices.

Finally, we combine Corollary III.3.1 (equivalently, (20)) with Assumption III.2 to state that if G is well-trained, then the eGCRB converges almost surely to the data CRB.

Theorem III.4:

$$\overline{\mathrm{GCRB}_{\mathrm{G}}}\left(\boldsymbol{\theta}\right) \xrightarrow{m\to\infty} \mathrm{CRB}_{\mathrm{R}}\left(\boldsymbol{\theta}\right) \quad \text{a.s}$$

Proof: By Assumption III.2 $\eta(\boldsymbol{\theta})=0$, so that by Corollary III.3.1 the eGCRB converges to the CRB as $m\to\infty$. To establish the type of convergence, note that $\mathbb{E}[\overline{\mathrm{F}_{\mathrm{G}}}]=\hat{\mathrm{F}}_{\mathrm{G}}(\boldsymbol{\theta})$ By the strong Law of Large numbers $\overline{\mathrm{F}_{\mathrm{G}}}(\boldsymbol{\theta})$ in (9) converges to its expected value, $\lim_{m\to\infty}\overline{\mathrm{F}_{\mathrm{G}}}(\boldsymbol{\theta})=\hat{\mathrm{F}}_{\mathrm{G}}(\boldsymbol{\theta})$ a.s. Finally, by Theorem III.2 for $\eta(\boldsymbol{\theta})=0$ we have $\hat{\mathrm{F}}_{\mathrm{G}}(\boldsymbol{\theta})=\mathrm{F}_{\mathrm{R}}(\boldsymbol{\theta})$. Inverting yields the result.

It follows that if Assumptions III.1 and III.2 hold then the eGCRB converges to the CRB almost surely as $m\to\infty$.

IV. Normalizing Flows

We use a normalizing flow [13], [14], a class of (invertiable) neural networks to obtain G and ν. Here, we give a brief overview of the normalizing flows utilized in this paper. Specifically, we will present conditional normalizing flow (CNF) where the normalizing flow is conditioned on the input parameter $\boldsymbol{\theta}$. A CNF transforms a random variable with a known distribution (typically Normal) through a sequence of differentiable, invertible mappings. Let $\boldsymbol{Z}_1,\ldots,\boldsymbol{Z}_{n_l}$ be a sequence of random variables that are related as $\boldsymbol{Z}_i=G_i(\boldsymbol{Z}_{i-1};\boldsymbol{\theta})$, where for each $\boldsymbol{\theta}\in\Theta$ the function $G_i(\cdot;\boldsymbol{\theta}):\mathbb{R}^d\to\mathbb{R}^d$ is a differentiable and bijective, n_l is the number of flow layers, and $\boldsymbol{Z}=\boldsymbol{Z}_0$ a random variable with a known and tractable probability density function $p_{\boldsymbol{Z}}:\mathbb{R}^d\to\mathbb{R}$. Then defining $\Gamma\triangleq\mathrm{G}(\boldsymbol{z}_0;\boldsymbol{\theta})=G_{n_l}\circ G_{n_l-1}\circ\ldots\circ G_1(\boldsymbol{z}_0;\boldsymbol{\theta})$ as a composition of the G_i, the transformation of a random variables formula says that the probability density function for Γ is

$$\begin{aligned} p_{\Gamma}\left(\boldsymbol{\gamma};\boldsymbol{\theta}\right) &= p_{\boldsymbol{Z}}\left(\nu\left(\boldsymbol{\gamma};\boldsymbol{\theta}\right)\right)\left|\det\mathbf{J}_{\nu}\left(\boldsymbol{\gamma};\boldsymbol{\theta}\right)\right|, \\ &= p_{\boldsymbol{Z}}\left(\nu\left(\boldsymbol{\gamma};\boldsymbol{\theta}\right)\right)\prod_{j=1}^{n_l}\left|\det\mathbf{J}_j\left(\boldsymbol{\gamma}_j;\boldsymbol{\theta}\right)\right|, \end{aligned} \tag{23}$$

where for each fixed $\boldsymbol{\theta}$, $\nu=\nu_1\circ\nu_2\circ\ldots\circ\nu_{n_l}$ and ν_i are the inverses of G and of G_i with respect to their first argument and $\mathbf{J}_j(\boldsymbol{\gamma};\boldsymbol{\theta})=\frac{\partial\nu_j(\boldsymbol{\gamma};\boldsymbol{\theta})}{\partial\boldsymbol{\gamma}}$ is the Jacobian of the $j\mathrm{th}$ transformation ν_j with respect to its input. We denote the value of the $j\mathrm{th}$ intermediate flow as $\boldsymbol{\gamma}_j\triangleq\mathrm{G}_j\circ\cdots\circ\mathrm{G}_1(\boldsymbol{z}_0;\boldsymbol{\theta})=\nu_{j+1}\circ\cdots\circ\nu_{n_l}(\boldsymbol{\gamma};\boldsymbol{\theta})$ and $\boldsymbol{\gamma}=\boldsymbol{\gamma}_{n_l}$.

Density Learning: A CNF can be used directly for density learning by finding parameters that minimize the negative log-likelihood (NLL) over a set of samples where the likelihood is given by (23). Given a dataset $\mathcal{D}$ (see Problem 1) and the transformations $G_1,\ldots,G_{n_l}$ parameterized by $\Omega=(\omega_1,\ldots,\omega_{n_l})$ respectively, the negative log-likelihood is given by:

$$\begin{aligned} L\left(\Omega\right) = &-\sum_i\log\left(p_{\boldsymbol{Z}}\left(\nu\left(\boldsymbol{r}_i;\boldsymbol{\theta}_i|\Omega\right)\right)\right) \\ &-\sum_i\sum_j^{n_l}\log\left(\left|\det\mathbf{J}_j\left(\boldsymbol{u}_{ij};\boldsymbol{\theta}_i|\Omega\right)\right|\right). \end{aligned} \tag{24}$$

where $\boldsymbol{u}_{ij}=\nu_{j+1}\circ\cdots\circ\nu_{n_l}(\boldsymbol{r}_i;\boldsymbol{\theta})$ denotes the intermediate flow of the $i\mathrm{th}$ sample and the $j\mathrm{th}$ layer. Note that the first term is the negative log-likelihood of the sample under the base measure (latent distribution) and the second term is a differential volume correction, which accounts for the change of differential volume induced by the transformations.

We use a CNF based on the Glow [11] architecture, which includes the following flow steps: Activation Normalization, Affine Coupling, and so-called 1x1 convolution (an invertible matrix operation). These flow steps transport the base distribution into the target distribution. However, we take the SRFlow approach [33] for the insertion of the conditioning parameter using the Affine Inject flow step that modifies the transformation according to the conditional parameter $\boldsymbol{\theta}$. Furthermore, in some cases (e.g., in the non-Gaussian measurement example of Section V-A2), a more complex modification of the base distribution is required, and this is achieved by replacing the Affine Coupling with a Cubic Spline Coupling flow [34]. The flow steps mentioned above are detailed in Appendix B.

V. Measurements Model Examples

First, we present two simple examples in which we can compute both the CRB and GCRB analytically and obtain an optimal generator. Note that by "optimal" we mean that the generator distribution $p_\Gamma(\boldsymbol{r};\boldsymbol{\theta})$ is identical to the data distribution $p_{\mathrm{R}}(\boldsymbol{r};\boldsymbol{\theta})$, meaning that Assumption III.2 holds with $\mathrm{G}=\mathrm{G}^*$. Then in the second part, we present a real-world measurement model of cameras, which will be used to demonstrate some of the benefits of the GCRB.

A. Simple Measurement Models

1) Linear Gaussian: Let

$$\mathrm{R}\left(\boldsymbol{\theta}\right)=\mathbf{A}\boldsymbol{\theta}+\boldsymbol{V}, \tag{25}$$

where matrix $\mathbf{A}\in\mathbb{R}^{d\times k}$ with $d>k$ and $\boldsymbol{V}\sim\mathcal{N}(0,\mathbf{C}_{vv})$ is an additive zero-mean Gaussian noise with positive-definite covariance $\mathbf{C}_{vv}\in\mathbb{R}^{d\times d}$. Then the measurement is distributed as $\mathrm{R}(\boldsymbol{\theta})\sim\mathcal{N}(\mathbf{A}\boldsymbol{\theta},\mathbf{C}_{vv})$, which provides a complete description of the measurement channel. The CRB for $\boldsymbol{\theta}$ coincides with the expression for the covariance of the linear unbiased minimum variance estimator and is given by [3] : $\mathrm{CRB}_{\mathrm{R}}(\boldsymbol{\theta})=[\mathbf{A}^T\mathbf{C}_{vv}^{-1}\mathbf{A}]^{-1}$.

Now we present an optimal generator for this example. Let $\mathrm{G}(\boldsymbol{Z};\boldsymbol{\theta}) = \mathbf{A}\boldsymbol{\theta} + \mathbf{L}\boldsymbol{Z}$, where $\boldsymbol{Z}_1 \sim N(0, \mathbf{I})$ and $\mathbf{L}$ is a square root (e.g, the Cholesky) factor of $\mathbf{C}_{vv}$, that is, $\mathbf{L}\mathbf{L}^T = \mathbf{C}_{vv}$. Then, as easily verified, G is an optimal generator, because for any $\boldsymbol{\theta}$ the distribution $\mathrm{G}(\boldsymbol{Z}) \sim \mathcal{N}(\mathbf{A}\boldsymbol{\theta}, \mathbf{C}_{vv})$ coincides with that of $\mathrm{R}(\boldsymbol{\theta})$. The inverse function of G, the normalizing flow, is $\nu(\boldsymbol{\gamma};\boldsymbol{\theta}) = \mathbf{L}^{-1}(\boldsymbol{\gamma} - \mathbf{A}\boldsymbol{\theta})$. We compute the score vector of the optimal generator and normalizing flow in Appendix E, which yields

$$\boldsymbol{s}_{\boldsymbol{\theta}}\left(\boldsymbol{z}\right) = -\mathbf{A}^T \left(\mathbf{L}^{-1}\right)^T \boldsymbol{z}, \tag{26}$$

and the GFIM corresponding to the optimal generator obtained using (8) is $\mathrm{F}_{\mathrm{G}}(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{Z}}[\mathbf{A}^T(\mathbf{L}^{-1})^T\boldsymbol{Z}\boldsymbol{Z}^T\mathbf{L}^{-1}\mathbf{A}]$. Simplifying and taking the inverse results in $\mathrm{F}_{\mathrm{G}}(\boldsymbol{\theta})^{-1} = \mathrm{GCRB}_{\mathrm{G}}(\boldsymbol{\theta}) = \mathrm{CRB}_{R}(\boldsymbol{\theta})$. This confirms that, as expected, an optimal generator will yield the same CRB on the parameter vector $\boldsymbol{\theta}$ as the correct distribution.

2) Scale Non-Gaussian: Here, we consider a scale model with a non-Gaussian distribution. Consider the data model

$$r = y\theta, \tag{27}$$

where $\theta \in \mathbb{R}^+$ is the desired parameter and y is a random variable with the PDF

$$p_y(y) = \frac{1}{\sqrt{2\pi\sigma^2}} 3y^2 \cdot \exp\left(-\frac{1}{2\sigma^2}y^6\right). \tag{28}$$

Then, as shown in Appendix F, the FIM of θ is $\mathrm{F}_{\mathrm{R}}(\boldsymbol{\theta}) = 18\theta^{-2}$ and $\mathrm{CRB}_{R}(\theta) = \frac{\theta^2}{18}$. We show in Appendix F1 that $\Gamma(\theta) = \mathrm{G}(z;\theta) = \theta z^{\frac{1}{3}}$ is an optimal generator, that is $\Gamma(\theta) \sim p_{R,\theta}(r;\theta)$. The inverse function of the optimal generator is the normalizing flow $\nu(\gamma;\theta) = (\frac{\gamma}{\theta})^3$. We compute the score vector of the optimal generator and normalizing flow in Appendix F2, which yields

$$\boldsymbol{s}_{\boldsymbol{\theta}}\left(\boldsymbol{z}\right) = -\theta^{-1}3\left(1 - z^2\right). \tag{29}$$

Finally, the FIM of the optimal generator obtained using (8) is $\mathrm{F}_{\mathrm{G}}(\boldsymbol{\theta}) = \mathbb{E}_{Z}[(\theta^{-1}3(1 - Z^2))^2] = 18\theta^{-2}$, where the last equality follows by the Gaussian moment property. Hence $\mathrm{GCRB}_{\mathrm{G}}(\theta) = \mathrm{CRB}_{R}(\theta)$. This example demonstrates that an optimal generator yields the correct CRB on the parameter in a non-Gaussian case.

B. Image Processing

We consider two classical image processing problems, however with a real-world learned measurement model of 4-channel (RGGB) color image sensors using NoiseFlow [7], a normalizing flow that models camera noise. Using this learned model, we obtain the GCRB for these two problems.

1) Image Denoising: Image denoising is a well-known problem in signal processing, however modeling camera noise is a challenging task [35], [36]. Due to the difficulty of noise modeling, it is impossible to compute an analytical lower bound for the denoising performance on a realistic model.

The denoising problem is defined as follows. Denoting by $\mathbf{H}$ a clean 4-channel (RGGB) image patch and by $\mathbf{V}$ the camera noise, the noisy image tensor is defined as

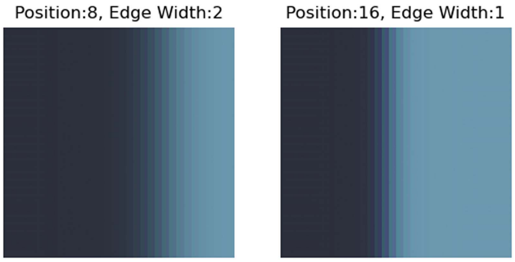


Fig. 2. Edges in a clean image.

$$\tilde{\mathbf{H}} = \mathbf{H} + \mathbf{V}. \tag{30}$$

Our goal is to provide a lower bound on the performance of any unbiased estimator that estimates the clean image $\mathbf{H}$ from the noisy image $\tilde{\mathbf{H}}$. Under this model, denoting by $\mathrm{Vec}(\mathbf{T})$ the vectorization of tensor $\mathbf{T}$, $\mathrm{R} = \mathrm{Vec}(\tilde{\mathbf{H}})$ is the measurement vector and $\boldsymbol{\theta} = \mathrm{Vec}(\mathbf{H})$ is the parameter vector.

2) Edge Detection: Another interesting image processing task is edge detection. Here we describe the edge model used in this work. Consider a c-channel image of width h pixels, and let $\mathbf{H}_{ijc} = f_{ijc}(\boldsymbol{\theta})$ be a vertical edge function that maps a continuous-parameter vector $\boldsymbol{\theta} = [\theta_p, \theta_w]$ of edge position $\theta_p \in [0, h-1]$ and width $\theta_w \in \mathbb{R}^+$, to the image color values at horizontal and vertical pixel position i, j and image channel c. The edge function is specified in terms of a horizontal color scaling function $s_i(\boldsymbol{\theta}) : ([0, h-1], \mathbb{R}^+) \to [0, 1]$ as

$$f_{ijc}\left(\boldsymbol{\theta}\right) = \left(p_c^h - p_c^l\right) \cdot s_i\left(\boldsymbol{\theta}\right) + p_c^l,$$

where p^h and p^l are the vectors of RGGB pixel values for high and low intensities, respectively. The color scaling function is defined as

$$s_i\left(\boldsymbol{\theta}\right) = \phi\left(\frac{\theta_p - i}{\theta_w}\right),$$

where ϕ is the Sigmoid function $\phi(x) = \frac{1}{1+\exp{(x)}}$. Images with edges of different position and width following the model above are shown in Fig. 2.

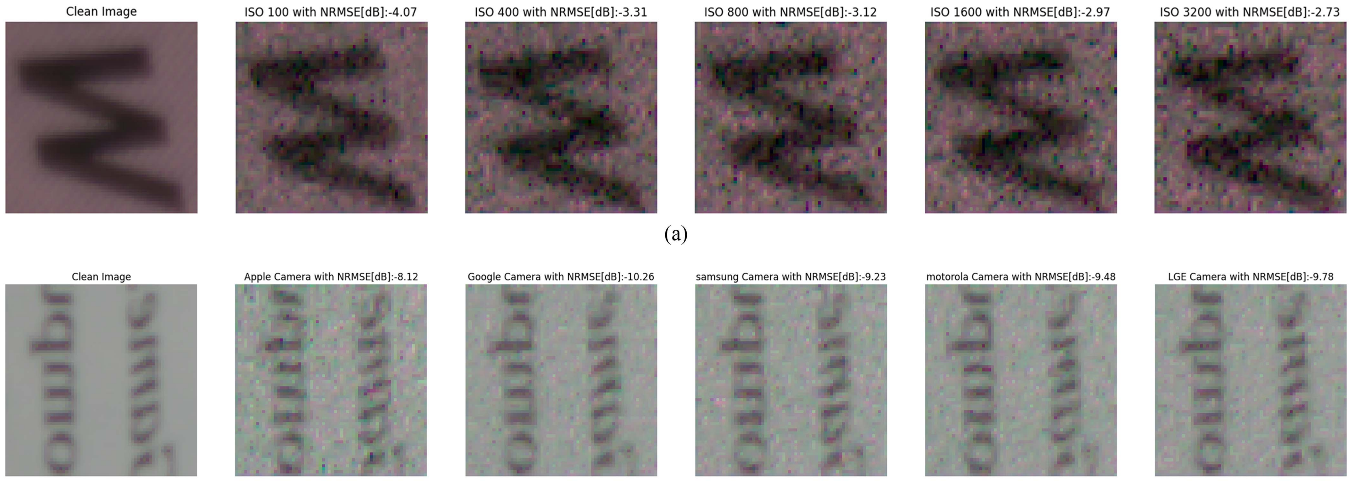
In this example we want to estimate the edge position θ_p from a noisy image $\tilde{\mathbf{H}}$. We compare the GCRB with NoiseFlow to the CRB derived for two well-known analytical Gaussian noise models: (i) i.i.d, or white Gaussian noise (WGN); and (ii) independent noise with image-dependent intensity - the so-called noise level function (NLF) noise model. The WGN model ($\mathbf{V}_{ijc} \sim \mathcal{N}(0, \sigma^2)$) with i.i.d noise in each channel of each pixel and variance σ^2 has CRB

$$\mathrm{CRB}_{\mathrm{W}}\left(\boldsymbol{\theta}\right) = \frac{\sigma^2\theta_w^2\left(\sum_i \mathbf{M}_i\left(\boldsymbol{\theta}\right)\right)^{-1}}{h\left\|\boldsymbol{p}^h - \boldsymbol{p}^l\right\|_2^2}, \tag{31}$$

where $\mathbf{M}_i(\boldsymbol{\theta}) = s_i^2(\boldsymbol{\theta})(1 - s_i(\boldsymbol{\theta}))^2 \begin{bmatrix} 1 & -\frac{\theta_p - i}{\theta_w} \\ -\frac{\theta_p - i}{\theta_w} & \frac{(\theta_p - i)^2}{\theta_w^2} \end{bmatrix}$.

The NLF model with $\mathbf{V}_{ijc} \sim \mathcal{N}(0, \alpha^2 f_{ijc}(\boldsymbol{\theta}) + \delta^2)$, where α and δ are the noise parameters, has CRB

$\mathrm{CRB}_{\mathrm{NLF}}\left(\boldsymbol{\theta}\right)$


Clean Image
ISO 100 with NRMSE[dB]:-4.07
ISO 400 with NRMSE[dB]:-3.31
ISO 800 with NRMSE[dB]:-3.12
ISO 1600 with NRMSE[dB]:-2.97
ISO 3200 with NRMSE[dB]:-2.73
Clean Image
Apple Camera with NRMSE[dB]:-8.12
Google Camera with NRMSE[dB]:-10.26
samsung Camera with NRMSE[dB]:-9.23
motorola Camera with NRMSE[dB]:-9.48
LGE Camera with NRMSE[dB]:-9.78

(a)

(b)

Fig. 3. NoiseFlow output: clean and noisy images. (a) Different ISO levels, for Camera Type=zero (Apple). (b) Different cameras at ISO level 100.

$$= \left(\sum_{i,j,c} \frac{\left(p_c^h - p_c^l\right)^2 \mathbf{M}_i\left(\boldsymbol{\theta}\right)}{\left(\alpha^2 f_{ijc}\left(\boldsymbol{\theta}\right) + \delta^2\right)^2 \theta_w^2} \left(\alpha^2 f_{ijc}\left(\boldsymbol{\theta}\right) + \delta^2 + \frac{\alpha^2}{2}\right)\right)^{-1}. \qquad (32)$$

A detailed calculation of CRBs is given in Appendix G.

3) Camera Noise Model: Several recent works [7], [37] have used a data-driven approach to model camera noise. We use NoiseFlow [7] to model a realistic camera noise $\mathbf{V} \sim p_{\mathbf{V}}(\boldsymbol{v}; \mathbf{H})$ and similarly to model the noisy image $\tilde{\mathbf{H}} \sim p_{\tilde{\mathbf{H}}}(\tilde{\boldsymbol{h}}; \mathbf{H})$. To obtain a noisy image flow, we cascade to NoiseFlow an AdditiveNoise Flow layer corresponding to (30), defined as

$$\boldsymbol{z}_{n+1} = \mathbf{H} + \boldsymbol{z}_n. \qquad (33)$$

The inverse of (33) is given by $\boldsymbol{z}_n = \boldsymbol{z}_{n+1} - \mathbf{H}$ and the log-determinant term is zero. Note that the ability to incorporate a signal model into the normalizing flow is a well-known advantage, which has also been exploited in NoiseFlow [7].

NoiseFlow is trained using the Smartphone Image Denoising Dataset (SIDD) [38]. The SSID dataset consists of 150 noisy and corresponding clean images captured in ten different scenes, with five smartphone cameras of different brands, under several lighting conditions and ISO (sensitivity) levels. Specifically, NoiseFlow is trained on $h_p = 32 \times w_p = 32$ pixel RGGB patches of clean image and noise $\mathbf{H}, \mathbf{V} \in \mathbb{R}^{h_p \times w_p \times 4}$.

Fig. 3 shows examples of clean images and the corresponding noisy images generated by NoiseFlow at different ISO levels and for different camera devices. They illustrate the strong ISO, device, and image dependence of the noise, which cannot be captured by an analytical model, thus precluding traditional calculation of estimation bounds. Instead, using this learned model, we obtain the GCRB for the two problems of image denoising and edge position detection.

VI. Experimental Results

This section presents a set of numerical experiments for assessing, analyzing, and demonstrating the GCRB. In the first set of experiments, we determine the quality of the approximation provided by the GCRB by evaluating the eGCRB on the examples in Section V-A and comparing to the true, analytically derived CRB. In the second set of experiments, we study, for the linear estimation problem, the approximation error of the eGCRB due to imperfect training, and due to the use of the sample mean to estimate the expected value. For the last two experiments, we present the usage of GCRB on the real-world examples of image denoising and edge detection in a device-dependent noise. Unless stated otherwise, we evaluate the eGCRB using $m = 64\ K$ generated samples in the sample mean in all experiments. In all experiments, the computation is done using Nvidia 1080Ti GPU running white PyTorch [21].

A. Approximation Quality

We evaluate the accuracy of the approximation to the CRB provided by the GCRB using two kinds of normalizing flows: (i) the optimal flow, which satisfies the condition of perfectly matched distribution as G^* in Assumption III.2; and (ii) a standard/learned normalizing flow (see Section IV), which is trained using the dataset $\mathcal{D}$.

Unless stated otherwise, we use the following parameters in the training process of all experiments For training a normalizing flow, we use the conditional negative log-likelihood (NLL) of the training set (24) as the loss function. We train each normalizing flow using a dataset of 200 k samples for 90 epochs with batch size 64. We use the Adam optimizer [39] with learning rate $1e-4$ and parameters $\beta_1 = 0.9$ and $\beta_2 = 0.999$. At the end of training, we obtain the learned normalizing flow ν and it's inverse (the generator) G and evaluate the eGCRB at several $\boldsymbol{\theta}$ values using Algorithm 1. The latent variable is chosen to be $\boldsymbol{z} \sim \mathcal{N}(0, \mathbf{I})$ in all examples. We begin by showing the approximation quality in the two examples presented in Section V-A, and then investigate the source of approximation error.

Linear Example: For the linear model (25) we use $d = 8, \sigma_v = 2.0$ and $k = 2$. Hence $\boldsymbol{\theta} \in \mathbb{R}^2$ and $R(\boldsymbol{\theta}_i) \in \mathbb{R}^8$. We generate the training dataset $\mathcal{D}$ in the following manner. First, we generate

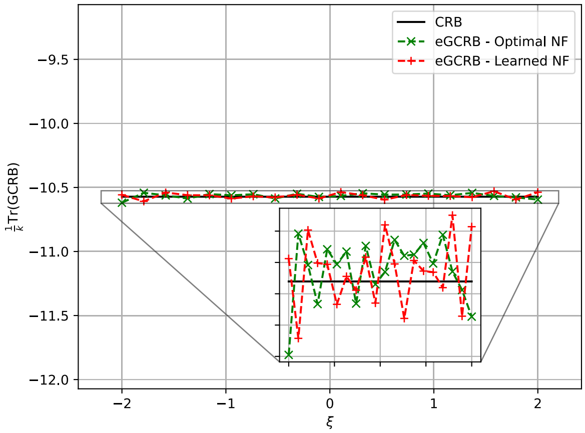


Fig. 4. Trace of CRB and eGCRB for the linear measurement model (25).

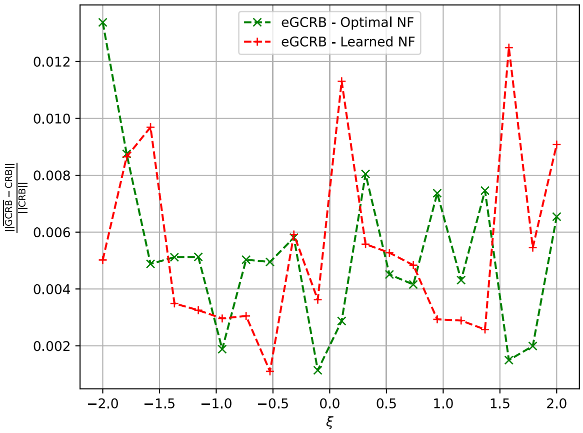


Fig. 5. Relative error between the eGCRB and CRB for the linear measurement model (25) at $\boldsymbol{\theta} = (\xi, \xi)^T$ as a function of ξ. The eGCRBs obtained using an optimal and learned flow are compared.

matrices $\mathbf{A}$ and $\mathbf{L}$ using a standard normal distribution, and use the same two matrices to generate all the samples in $\mathcal{D}$. For each sample $(\boldsymbol{\theta}_i, R(\boldsymbol{\theta}_i)) \in \mathcal{D}$, the parameter vector $\boldsymbol{\theta}_i \in \mathbb{R}^2$ is drawn i.i.d from a uniform distribution $\boldsymbol{\theta}_i \sim U[-2,2]^2$, $\boldsymbol{V}_i \in \mathbb{R}^8$ is drawn i.i.d Normal $\boldsymbol{V}_i \sim \mathcal{N}(0, \mathbf{C}_{vv})$ with $\mathbf{C}_{\boldsymbol{vv}} = \mathbf{L}\mathbf{L}^T$, and $R(\boldsymbol{\theta}_i) \in \mathbb{R}^8$ is computed using (25). Using this dataset, we train a normalizing flow with the architecture shown in Appendix C3, obtaining ν and G.

We chose an architecture with invertible $1x1$ convolution and affine inject since it can represent an optimal generator and satisfies the G $\in C^2$ condition.

Then, for each value of $\boldsymbol{\theta}$ of interest, we use the trained generator to generate samples of the score vector, and compute the eGFIM using (9), which yields, upon inversion, the eGCRB. For comparison, we repeat the generation of the score vector using the optimal normalizing flow and generator instead of the learned flow and generator.

In Fig. 4 we display the traces of the two eGCRBs, as well as that of the analytical CRB, for $\boldsymbol{\theta} = (\xi, \xi)^T$, with ξ on a uniform grid on the interval [−2,2]. Fig. 4 shows that a learned normalizing flow can estimate the true CRB to a good accuracy. Because (as we verifed) the specific parameter values $\boldsymbol{\theta}$ shown in Fig. 4 are not present in the training set, this also demonstrates that the GCRB works well for unseen examples. As expected, on the average the eGCRB using the optimal flow has slightly better accuracy than the one using the learned flow, because the former only suffers from the finite sampling error in estimating the GFIM using an empirical mean, whereas the latter is also subject to the imperfectly learned flow model. The relative error (18) of the eGCRB is displayed in Fig. 5 for both learned and optimal flows, showing that both have comparable accuracy, of within $\approx 0.5\%$ from the true CRB.

Scale Example: In this example we generate the training dataset $\mathcal{D}$ in the following manner. For each sample $(\theta_i, R(\theta_i)) \in \mathcal{D}$, the parameter θ_i is drawn i.i.d from a uniform distribution $\theta_i \sim U[3,6]$, $\boldsymbol{Y}$ is drawn i.i.d $\boldsymbol{Y} \sim p_Y$, where p_Y is given by (28) and $R(\theta_i) = y\theta_i$, per (27). To produce a vector input to the normalizing flow, as needed for the application of affine coupling, we define a vector measurement of length 2, composed of two i.i.d samples with the same parameter θ_i.[2] Note that since this vector measurement corresponds to two i.i.d measurements, by the additivity property of the FIM, this only scales the resulting GFIM by a factor of 2. Using this dataset $\mathcal{D}$ we train a normalizing flow with the architecture shown in Appendix C. We chose an architecture with cubic-spline and affine inject since it can locally represent an optimal generator and satisfies the G $\in C^2$ condition.

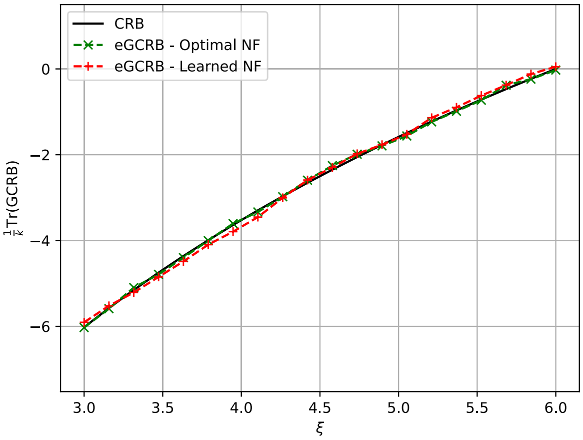


Fig. 6. Analytical CRB and eGCRBs for the scale model (27),(28), using the optimal flow and a learned flow.

Then, we follow the same procedure as for the linear measurement model to produce the eGCRB using both the learned flow and the optimal flow. The two eGCRB values and the true CRB are compared in Fig. 6. Fig. 6 demonstrates that a learned normalizing flow can estimate the true CRB in the non-Gaussian case, with accuracy comparable to that of the optimal flow. Similar to the linear example, because (as we verified) the parameter values θ used to plot Fig 6 are not present in the training set, this again demonstrates the interpolation capability of the GCRB to provide a good approximation for unseen examples.

[2] We use this form, rather than padding with an unrelated standard normal random variable, to mitigate issues of exploding condition number [29].

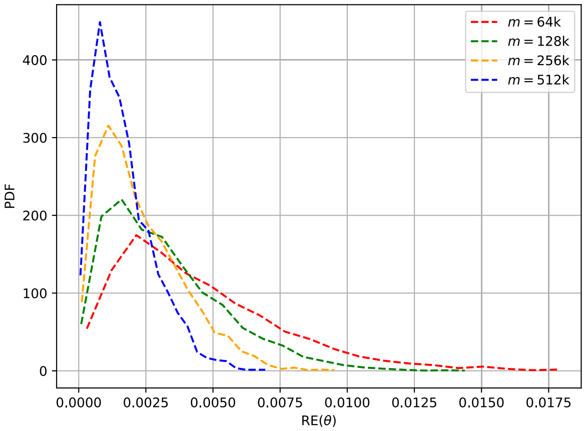


Fig. 7. Histogram of $\mathrm{RE}(\boldsymbol{\theta})$ for the linear example using different number of samples $m \in \{64\ k, 128\ k, 256\ k, 512k\}$.

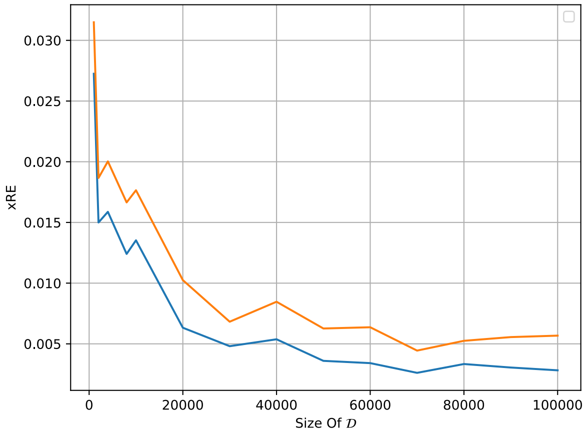


Fig. 8. MRE and $\overline{\mathrm{MRE}}$ for the linear measurement model example vs. training dataset size.

To summarize, Figs. 6 and 4 demonstrate Theorem III.2, which states that given a well-trained generative model (Assumption III.2) the GCRB approximates well the CRB. Both figures show random deviations of the eGCRB from the CRB due to two reasons: (i) imperfectly trained generative model; and (ii) a finite number of samples used to calcuate empirical mean, as stated in Theorem III.3. Moreover, the eGCRBs in Figs. 4 and 6 are evaluated at points that are not present in the training set, which shows the ability of GCRB to interpolate the CRB values to those points.

B. Error Analysis

Here, we study further the approximation error due to the empirical mean and imperfect training. We use two metrics for the error:

$$\mathrm{MRE} = \max_{\boldsymbol{\theta}\in\Theta_T} \mathrm{RE}\left(\boldsymbol{\theta}\right),$$

$$\overline{\mathrm{MRE}} = \frac{1}{|\Theta_T|}\sum_{\boldsymbol{\theta}\in\Theta_T} \mathrm{RE}\left(\boldsymbol{\theta}\right),$$

where MRE and $\overline{\mathrm{MRE}}$ are the maximal and mean relative norm error, respectively, $\Theta_T \subset \Theta$ is the set of $\boldsymbol{\theta}$ value used in the validation process, and $|\Theta_T|$ is the cardinality of the set Θ_T. In this experiment, we verify Theorem III.3 using the linear optimal model with the same parameters as above and evaluate eGCRB with different number of samples $m \in \{64\ K, 128\ K, 256\ K, 512K\}$. We repeat the evaluation for each m 2000 times and and present the histogram of the relative norm error $\mathrm{RE}(\boldsymbol{\theta})$ in Fig. 7. In all the trials we use the same parameter vector $\boldsymbol{\theta} = (0.2, 0.2)^T$.

We see in Fig. 7 the effect of different m values on the distribution of the relative error. This confirms, that as predicted by Theorem III.3, for the optimal generator, we can make the eGCRB error arbitrary small by increasing the number of samples m to calculate the eGFIM (9). This addresses the error due to sampling assuming a well-trained normalizing flow.

In the next experiment, we address the error due to imperfect training. To focus on this aspect, we set $m = 512\ k$, so that the error due to the empirical mean is negligible. We train a normalizing flow on the linear problem using various dataset sizes, and report the maximal and mean relative error. To train a normalizing flow with a small dataset size, we adjust the number of epochs to have a constant number of gradient updates, by setting the number of epochs to $\left\lceil 90\frac{200e^3}{|\mathcal{D}|}\right\rceil$ where $|\mathcal{D}|$ is the size of the dataset and $\lceil x \rceil$ denotes the ceiling of x. In Fig. 8 we present the MRE for different dataset size and validation parameter set Θ consisting of 20 points $\boldsymbol{\theta} = (\xi, \xi)^T$ with $\xi \in [-2, 2]$ uniformly spaced.

Fig. 8 shows relative error decreasing with increasing training set size, highlighting the importance of training to obtain a well-trained generative model. However, increasing the dataset size beyond some point (in this case, above 50 K samples) doesn't improve the results. To interpret this saturation effect, recall that the normalizing flow used in this problem can represent the optimal generator, hence a limitation in representation capacity by the normalizing flow is not the culprit. Instead, comparison with Fig. 7 suggests that for $|\mathcal{D}| > 5 \cdot 10^4$ the error in this learned generator experiment is dominated by the empirical mean error - with similar values as when using an optimal generator.

C. Image Denoising

Here, we study the GCRB for the Image denoising problem, using the NoiseFlow [7] to model the image noise. Noise Flow is composed of affine coupling, invertible 1×1 convolution, gain layer (which is a constant affine transformation), and signal-dependent layer (which is similar to affine inject with a predefined function). First, we replace the ReLU activation function in the original NoiseFlow to Sigmoid Linear Unit (SiLU) [40], to satisfy the requirement that $\mathrm{G} \in C^2$. Then, we train the modified Noise-Flow from scratch using the authors' original code [41]. We obtained NNL of -3.528 which is a similarl to that in the original NoiseFlow. Then, we add to NoiseFlow an AdditiveNoise layer (see Section V-B3), which provides a noisy image. We generate all the required derivatives using the PyTorch built-in symbolic differentiation invoked using standard PyTorch commands, and utilize our eGCRB computing forumulas to provide an approximation to the CRB. We compute the eGCRB using (9) with $m = 64\ k$.

Note that for image visualization purposes only, we render RGGB images through a color processing pipeline into sRGB color space. Moreover, to visualize the eGCRB, we extract the diagonal of the eGCRB matrix and present the lower bound for each channels (R, G, G, B) separately. In all experiments we use an image size of $h = w = 32$ and batch-size 32.

We present the bounds on the denoising performance in normalized form, by computing the following metrics

$$\mathrm{NPRMSE}_{ijc} \triangleq \frac{\sqrt{g\left(\mathrm{diag}\left(\overline{\mathrm{GCRB}}\right)\right)_{ijc}}}{I_{ijc}}, \tag{35a}$$

$$\mathrm{NRMSE} \triangleq \sqrt{\frac{1}{hwc}\sum\nolimits_{i,j,c} \mathrm{NPMSE}^2_{ijc}}. \tag{35b}$$

The NPRMSE_{ijc} is the normalized per pixel bound on the error standard deviation error, $\mathrm{diag}(\mathbf{A})$ denotes the diagonal vector of matrix $\mathbf{A}$, $g() : \mathbb{R}^{h \cdot w \cdot c} \to \mathbb{R}^{h \times w \times c}$ is the reshaping of vector to RGGB image, and i, j, c are the vertical, horizontal and channel indices, respectively. In turn, NRMSE is the square of the per pixel NPRMSE, averaged over the entire image followed by a square root.

Note that the metrics in (35) take into account only the diagonal elements of the eGCRB. However, both the eGFIM and the eGCRB have off-diagonal elements thanks to the ability of NoiseFlow to generate correlated noise that models the sensor. This correlation is evident in the non-zero off-diagonals in the eGFIM, and affects the diagonal elements of the eGCRB.

In this study, we perform several experiments. First, we present several visual examples in Fig. 9. The top row shows three clean images to which we refer, from left to right, as scene one, two, and three. The second row shows the corresponding noisy versions for Camera 0 (Apple) at ISO $=$ 100. The next four rows display (as images color-coded by magnitude) the normalized lower bound on the denoising error standard deviation for each pixel (i.e., the $NPRMSE_{ijc}$): one row for each of the four channels $c = 1, 2, 3, 4$. We observe that pixels with different colors have distinct lower bounds, showing the effect of the Signal Depend Layer in NoiseFlow [7]. An analogous behavior is seen in the NFL model which scales the noise by the clean image values. It is also seen that brighter pixels have a better (smaller) normalized lower bound than darker ones.

Next, in Fig. 10(a) we plot the normalized lower bound on the denoising performance using Device=0 (Apple) on the same three scenes as in Fig. 9. It is seen that a lower ISO level allows better denoising than a higher one, and that different scenes have different denoising lower bounds. For an insight as to whether the difference is due to color level, scene structure, or both, we refer to Fig. 3. It reveals that the noise level increases with ISO level, and is relatively higher in darker (lower color level) areas. The first property clearly accounts for the increase in the bound vs. ISO level seen in Fig. 10(a), whereas the second property explains the relative ranking of the bounds for the three scenes, with increasing normalized bound for darker images.

Fig. 10(b) shows, for Scene 1, the effect of different measurement devices. The relative ranking in terms of the denoising bound cannot be inferred from the visual impression of the noise for the different devices in Fig. 3. However, it remains the same for Scene 2 and Scene 3, showing consistency of the bounds for each device. These results demonstrates a unique advantage of the GCRB, which can provide a bound specific to a measurement device.

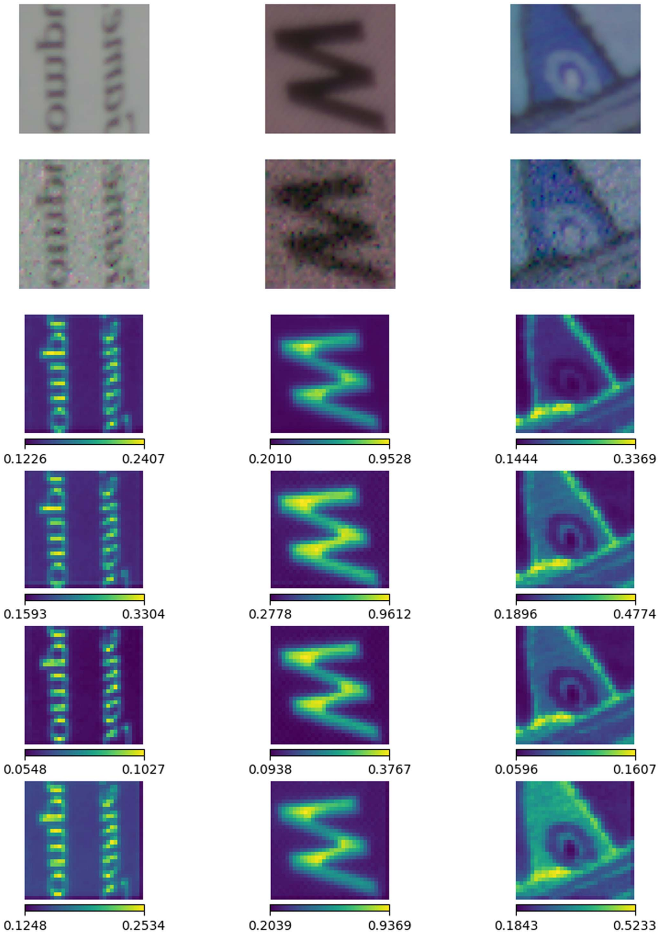
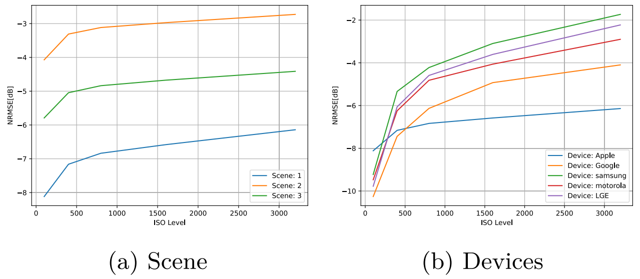


Fig. 9. GCRB for the denoising problem on three scenes (one per column). 1st and 2nd rows: clean and noisy image, respectively. Last four last rows: NPMSE of the R, G, G, B channels. Camera type zero (Apple).



(a) Scene (b) Devices

Fig. 10. Lower bound on the denoising error vs. ISO levels. In Fig. 10(a) the lines correspond to Scene 1 - Scene 3 on the first row of Fig. 9 and in Fig 10(b) the lines represent different devices in Scene 1 of Fig. 9.

D. Edge Detection

We use the same parameters as used in the image denoising problem. First, in Fig. 11, we present a lower bound on edge position estimate vs. the position of the edge in the image, for several different edge widths, using Device=0 (Apple) and

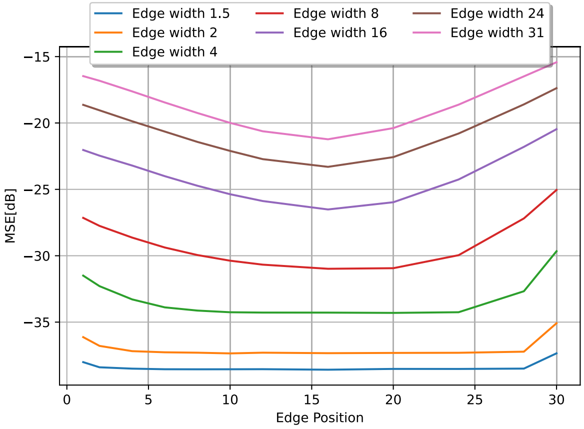
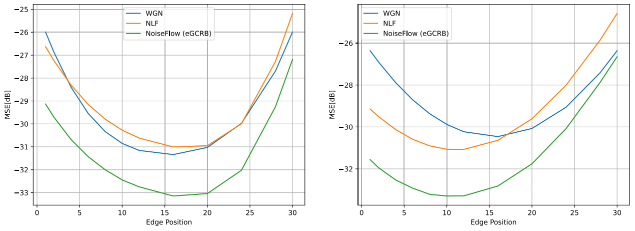


Fig. 11. Lower bound on edge position with different edge widths using Device=0 (Apple) at ISO 100.



(a) Position (b) Width

Fig. 12. Lower bound on variances of the edge detection parameters (Position 12(a) and Width 12(b)) over different edge positions, using different measurement noise models for Device=2 at ISO 100 and edge width 8 pixels.

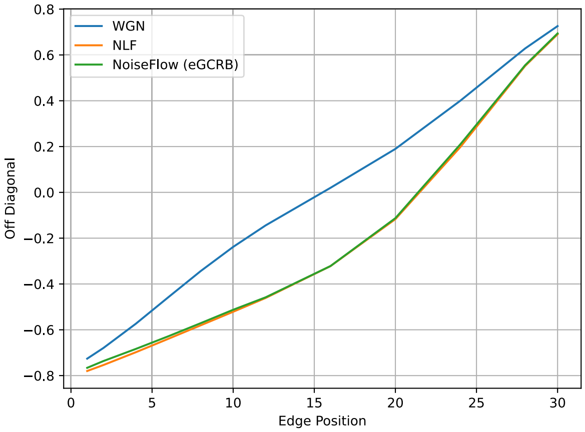


Fig. 13. Normalized off-diagonal elements of the eGCRB (correlation of position and width estimation errors) for the edge detection problem over different edge positions, using Device=2 at ISO 100.

ISO level 100. Fig. 11 reveals different behavior of the edge localization bound for different edge widths. First, the bound increases with increasing edge width. This is not surprising, since a smooth edge can be expected to be harder localize in the presence of noise than a sharp edge, and this also agrees with the dependence of the CRB (31) on edge width for standard Gaussian noise. Second, for small edge width, the bound shows little dependence on position at the center of the image, but increases slightly when the edge approaches the boundaries of the image. This can be attributed to the truncation of some of the edge transition when the edge approaches the image boundaries. Third, for larger edge widths, the bound shows an asymmetric dependence on the edge position relative to the image center. This too can be explained by truncation of one side of the edge transition: however, because the noise is signal-dependent, the effect of truncating the bright side of the edge is opposite to that of truncating the dark side. Moreover, a similar effect of edge position is observed in the analytical CRB (32) for the NLF noise model, which also has signal-dependent noise level.

To further demonstrate the advantages of the GCRB, we investigate in the next experiment the ability of a generative model to capture the complex measurements distribution and produce an accurate lower bound. To this end, we compare, in the context of the edge detection problem, the three noise models: WGN, NLF, and Noise-Flow. We do so for Device=2 (Samsung) (Fig. 3 at ISO level 100. For a quantitatively meaningful comparison, we set the parameters σ^2, α, and δ of the analytical noise models to the maximum likelihood estimates obtained from the noisy images that were used to train NoiseFlow [7]. These noisy images are taken from the base SSID dataset [38], and were preprocessed as in NoiseFlow.

Fig. 12 shows that (i) the WGN model misses altogether the asymmetric behavior of the bound with respect to edge position; and (ii) both Gaussian noise models have CRBs larger than the eGCRB. Both (i) and (ii) are to be expected, since the WGN model misses the signal-dependence of the noise, and independent Gaussian noise yields the largest CRB for given noise variance [42]. Finally, note the subtantial difference between the eGCRB for Device=2 (in Fig. 12) and the eGCRB for Device=0 in Fig. 11 for the same edge width of 8 and ISO 100. This again demonstrates the unique ability of the GCRB to provide device-dependent bounds. Overall, these results illustrate the importance of a learned model to capture the complexity of the measurement distribution and obtain an accurate lower bound.

In addition, in Fig. 13 we illustrate the ability of the GCRB to study the correlation between estimation errors of different parameters. Specifically, we present the normalized off-diagonal of the eGCRB, namely the Pearson correlation. We observe that whenever the edge is located at the boundaries of the image, $\theta_p = 0$ or $\theta_p = 31$, there is a high correlation between the position and width parameter estimates. This correlation diminishes for edge position at the center of the image. Moreover, in the center region, the NLF and NoiseFlow have a different crossing point, due to signal depend noise, the point at which the dark and light pixels have the same SNR is shifted.

VII. Conclusions and Future Work

In this paper, we use for the first time a generative model to obtain a data-driven estimate of the Cramer-Rao bound, which does not require access to an analytical model of the measurement probability distribution. Specifically, we used a normalizing flow and showed that this generative model provides the same CRB as the measurement distribution if the generator is well-trained. Moreover, we provided an error analysis bounding the inaccuracy due to the use of an empirical mean for the well-trained case, and the error of the GCRB due to imperfect learning. We validated the performance of this approach on two simple signal models with known ground-truth CRBs. We also studied the GCRB on two image processing tasks with a complex learned measurement model. The results demonstrate two advantages of the GCRB: the ability to obtain a highly accurate performance bound for complex measurement distributions without an analytical model; and the ability to obtain a device-specific bound.

Questions for future research include quantifying the impacts of limited representation power of the generative model and a limited training data set on the accuracy of the GCRB. Another direction is to ensure that GCRB is a valid lower bound (rather than a good approximation to it) by utilizing methods for error estimation and model selection [43]. On the practical side, it will be interesting to study some of the many real-world applications that can benefit from this approach, such as direction-of-arrival estimation in sensor arrays with poorly characterized propagation models.

VIII. Proofs

A. *Proof of Lemma III.1*

Proof: The generated samples $\boldsymbol{\gamma} = \mathrm{G}(\boldsymbol{z};\boldsymbol{\theta}) \in \hat{\Upsilon}$ retained after the trimming process correspond to $\boldsymbol{z} \in \mathcal{Z}$, where the set $\mathcal{Z} \triangleq \mathrm{G}^{-1}(\hat{\Upsilon};\boldsymbol{\theta})$ is the pre-image of $\hat{\Upsilon}$ under G. Because $\hat{\Upsilon}$ is a compact set in metric space $\mathbb{R}^d$ (Assumption A.7) and $\mathrm{G}^{-1} = \nu : \mathbb{R}^d \rightarrow \mathbb{R}^d$ is a continuous mapping, it follows that as the image of a compact set by a continuous mapping, $\mathcal{Z}$ is a compact set. Next, because $\mathrm{G} \in C^2$ and G is a diffeomorphism wrt to $\boldsymbol{z}$, it follows that each of the components $\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{z})$ is a continuous function of $\boldsymbol{z}$ on the compact set $\mathcal{Z}$. Hence (by pseudocompactness) $\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{z})$ is bounded componentwise, and thus also in norm: $\|\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{z})\| \leq \mathrm{C_s}(\boldsymbol{\theta})$. □

B. *Proof of Theorem III.2*

We use the following result, proved in Section D4 of the Appendix.

Lemma VIII.1. (Matrix Cauchy-Schwartz Inequality): Let $\boldsymbol{X}, \boldsymbol{Y} \in \mathbb{R}^n$ be random vectors with correlation matrices $\mathbf{R}_X$ and $\mathbf{R}_Y$ and cross correlation $\mathbf{R}_{XY} = \mathbb{E}[\boldsymbol{X}\boldsymbol{Y}^T]$. Then,

$$\|\mathbf{R}_{XY}\| \leq \left(\|\mathbf{R}_X\| \, \|\mathbf{R}_Y\|\right)^{1/2}. \tag{36}$$

For conciseness, in the proof of Theorem III.2 below we omit the integration variable $\boldsymbol{r}$ and the parameter vector $\boldsymbol{\theta}$ from integrals. Thus, the PDFs of the true and learned measurement distributions are abbreviated as $p_{\mathrm{R}} = p_{\mathrm{R}}(\boldsymbol{r};\boldsymbol{\theta})$ and $p_{\Gamma} = p_{\Gamma}(\boldsymbol{r};\boldsymbol{\theta})$, respectively, and the corresponding score vectors $\boldsymbol{s}_{\mathrm{R}} = \boldsymbol{s}_{\mathrm{R}}(\boldsymbol{r};\boldsymbol{\theta}) \triangleq \nabla_{\boldsymbol{\theta}}\mathrm{L}_{\mathrm{R}}(\boldsymbol{r};\boldsymbol{\theta})$ and $\boldsymbol{s}_{\Gamma} = \boldsymbol{s}_{\Gamma}(\boldsymbol{r};\boldsymbol{\theta}) \triangleq \nabla_{\boldsymbol{\theta}}\mathrm{L}_{\Gamma}(\boldsymbol{r};\boldsymbol{\theta})$, where $\mathrm{L}_{\mathrm{R}}(\boldsymbol{r};\boldsymbol{\theta})$ $\mathrm{L}_{\Gamma}(\boldsymbol{r};\boldsymbol{\theta})$ are the corresponding negative log-likelihoods.

Proof:

$$\begin{aligned}
\mathrm{F}_{\mathrm{R}}(\boldsymbol{\theta}) - \hat{\mathrm{F}}_{\mathrm{G}}(\boldsymbol{\theta}) &= \mathrm{F}_{\mathrm{R}}(\boldsymbol{\theta}) - \mathrm{F}_{\Gamma}(\boldsymbol{\theta}) \\
&= \int_{\Upsilon\setminus\hat{\Upsilon}} \boldsymbol{s}_{\mathrm{R}}\boldsymbol{s}_{\mathrm{R}}^T p_{\mathrm{R}} d\boldsymbol{r} + \int_{\hat{\Upsilon}} \boldsymbol{s}_{\mathrm{R}}\boldsymbol{s}_{\mathrm{R}}^T p_{\mathrm{R}} d\boldsymbol{r} - \mathrm{F}_{\Gamma}(\boldsymbol{\theta}) \\
&= \mathbf{P}_1 + \mathbf{P}_2 \\
\mathbf{P}_1 &\triangleq \int_{\Upsilon\setminus\hat{\Upsilon}} \boldsymbol{s}_{\mathrm{R}}\boldsymbol{s}_{\mathrm{R}}^T p_{\mathrm{R}} d\boldsymbol{r} + \int_{\hat{\Upsilon}} \boldsymbol{s}_{\mathrm{R}}\boldsymbol{s}_{\mathrm{R}}^T \Delta_p d\boldsymbol{r} \\
\mathbf{P}_2 &\triangleq \int_{\hat{\Upsilon}} \boldsymbol{s}_{\mathrm{R}}\boldsymbol{s}_{\mathrm{R}}^T p_{\Gamma} d\boldsymbol{r} - \mathrm{F}_{\Gamma}(\boldsymbol{\theta}) && (37) \\
\Delta_p &\triangleq p_{\mathrm{R}} - p_{\Gamma} && (38)
\end{aligned}$$

Because $p_{\Gamma}(\boldsymbol{r};\boldsymbol{\theta}) = 0 \quad \forall \boldsymbol{r} \in \Upsilon \setminus \hat{\Upsilon}, \boldsymbol{\theta} \in \Theta$,

$$\mathbf{P}_1 = \int_{\Upsilon} \boldsymbol{s}_{\mathrm{R}}\boldsymbol{s}_{\mathrm{R}}^T \Delta_p d\boldsymbol{r} + \int_{\Upsilon\setminus\hat{\Upsilon}} \boldsymbol{s}_{\mathrm{R}}\boldsymbol{s}_{\mathrm{R}}^T p_{\Gamma} d\boldsymbol{r} = \int_{\Upsilon} \boldsymbol{s}_{\mathrm{R}}\boldsymbol{s}_{\mathrm{R}}^T \Delta_p d\boldsymbol{r}$$

which, by Assumption III.1, is bounded in terms of the total variation distance as

$$\|\mathbf{P}_1\| \leq 2C_{\mathrm{R}}^2(\boldsymbol{\theta})\,\mathrm{TV}\left(p_{\mathrm{R}}, p_{\Gamma}; \boldsymbol{\theta}\right). \tag{39}$$

Turning to $\mathbf{P}_2$, we have

$$\begin{aligned}
\mathbf{P}_2 &= \mathbb{E}_{\Gamma}\left[\boldsymbol{s}_{\mathrm{R}}\boldsymbol{s}_{\mathrm{R}}^T - \boldsymbol{s}_{\Gamma}\boldsymbol{s}_{\Gamma}^T\right] \\
&= \mathbb{E}_{\Gamma}\left[\left(\boldsymbol{s}_{\Gamma} - \boldsymbol{\Delta}_s\right)\left(\boldsymbol{s}_{\Gamma} - \boldsymbol{\Delta}_s\right)^T - \boldsymbol{s}_{\Gamma}\boldsymbol{s}_{\Gamma}^T\right] \\
&= \mathbb{E}_{\Gamma}\left[\boldsymbol{\Delta}_s\boldsymbol{\Delta}_s^T\right] - \mathbb{E}_{\Gamma}\left[\boldsymbol{s}_{\Gamma}\boldsymbol{\Delta}_s^T + \boldsymbol{\Delta}_s\boldsymbol{s}_{\Gamma}^T\right], && (40)
\end{aligned}$$

where $\boldsymbol{\Delta}_s \triangleq \boldsymbol{s}_{\mathrm{R}} - \boldsymbol{s}_{\Gamma}$ is the score difference vector. Considering the first term in (40):

$$\begin{aligned}
&\left\|\mathbb{E}_{\Gamma}\left[\boldsymbol{\Delta}_s\boldsymbol{\Delta}_s^T\right]\right\| \leq \mathrm{Tr}\left(\mathbb{E}_{\Gamma}\left[\boldsymbol{\Delta}_s\boldsymbol{\Delta}_s^T\right]\right) = \mathbb{E}_{\Gamma}\left[\boldsymbol{\Delta}_s^T\boldsymbol{\Delta}_s\right] \\
&= \mathbb{E}_{\Gamma}\left[\|\boldsymbol{\Delta}_s\|^2\right] \leq \int_{\Upsilon} p_{\Gamma}\|\boldsymbol{\Delta}_s\|^2 d\boldsymbol{r} \triangleq \mathrm{I_F}\left(p_{\Gamma}, p_{\mathrm{R}}; \boldsymbol{\theta}\right). && (41)
\end{aligned}$$

Next, applying Lemma VIII.1 to the norm of the second term in (40), yields

$$\begin{aligned}
\left\|\mathbb{E}_{\Gamma}\left[\boldsymbol{s}_{\Gamma}\boldsymbol{\Delta}_s^T + \boldsymbol{\Delta}_s\boldsymbol{s}_{\Gamma}^T\right]\right\| &\leq 2\left\|\mathbb{E}_{\Gamma}\left[\boldsymbol{\Delta}_s\boldsymbol{s}_{\Gamma}^T\right]\right\| \\
&\leq 2\left(\left\|\mathbb{E}_{\Gamma}\left[\boldsymbol{s}_{\Gamma}\boldsymbol{s}_{\Gamma}^T\right]\right\| \left\|\mathbb{E}_{\Gamma}\left[\boldsymbol{\Delta}_s\boldsymbol{\Delta}_s^T\right]\right\|\right)^{1/2} \\
&= 2\left(\left\|\hat{\mathrm{F}}_{\mathrm{G}}(\boldsymbol{\theta})\right\| \mathrm{I_F}\left(p_{\Gamma}, p_{\mathrm{R}}; \boldsymbol{\theta}\right)\right)^{1/2}. && (42)
\end{aligned}$$

Now combining (41) and (42) yields:

$$\|\mathbf{P}_2\| \leq 2\left(\left\|\hat{\mathrm{F}}_{\mathrm{G}}(\boldsymbol{\theta})\right\| \mathrm{I_F}\left(p_{\Gamma}, p_{\mathrm{R}}; \boldsymbol{\theta}\right)\right)^{1/2} + \mathrm{I_F}\left(p_{\Gamma}, p_{\mathrm{R}}; \boldsymbol{\theta}\right). \tag{43}$$

In the last step we combine (39) and (43), which yields Theorem III.2. □

C. *Proof of Corollary III.2.1*

Proof: By definition (11), we have $\hat{\mathrm{F}}_{\mathrm{G}}(\boldsymbol{\theta}) \succeq 0$. Combining with (13a), we have $\lambda_{\min}(\hat{\mathrm{F}}_{\mathrm{G}}(\boldsymbol{\theta})) \geq \lambda_{\min}(\mathrm{F}_{\mathrm{R}}(\boldsymbol{\theta})) - \eta(\boldsymbol{\theta}) > 0$, where the positivity is by the assumption of the Corollary. Hence $\hat{\mathrm{F}}_{\mathrm{G}}(\boldsymbol{\theta}) \succ 0$. It then follows that

$$\begin{aligned}
\|\hat{\mathrm{F}}_{\mathrm{G}}(\boldsymbol{\theta})^{-1}\| &= \lambda_{\max}\left(\hat{\mathrm{F}}_{\mathrm{G}}(\boldsymbol{\theta})^{-1}\right) = 1/\lambda_{\min}\left(\hat{\mathrm{F}}_{\mathrm{G}}(\boldsymbol{\theta})\right) \\
&\leq \left[\lambda_{\min}(\mathrm{F}_{\mathrm{R}}(\boldsymbol{\theta})) - \eta(\boldsymbol{\theta})\right]^{-1}
\end{aligned}$$

which, establishes (14a). To establish (14b), we have

$$\begin{aligned}&\|\mathrm{F}_{\mathrm{R}}\left(\boldsymbol{\theta}\right)^{-1}-\hat{\mathrm{F}}_{\mathrm{G}}\left(\boldsymbol{\theta}\right)^{-1}\|\\&\quad=\|\mathrm{F}_{\mathrm{R}}\left(\boldsymbol{\theta}\right)^{-1}\left(\mathrm{F}_{\mathrm{R}}\left(\boldsymbol{\theta}\right)-\hat{\mathrm{F}}_{\mathrm{G}}\left(\boldsymbol{\theta}\right)\right)\hat{\mathrm{F}}_{\mathrm{G}}\left(\boldsymbol{\theta}\right)^{-1}\|\\&\quad\leq\|\mathrm{F}_{\mathrm{R}}\left(\boldsymbol{\theta}\right)^{-1}\|\cdot\|\hat{\mathrm{F}}_{\mathrm{G}}\left(\boldsymbol{\theta}\right)^{-1}\|\cdot\|\mathrm{F}_{\mathrm{R}}\left(\boldsymbol{\theta}\right)-\hat{\mathrm{F}}_{\mathrm{G}}\left(\boldsymbol{\theta}\right)\|\end{aligned}$$

and the result follows by applying (13a) to the last factor in the product. □

D. Proof of Theorem III.3

First we establish that the generated score vector has zero mean.

Lemma VIII.2. Let $\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{z})$ be a score vector computed using a trimmed and differentiable $\mathrm{G}\in C^2$ generator G and it's inverse ν. Then $\mathbb{E}_{\boldsymbol{Z}}[\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{Z})]=0$.

The proof of Lemma VIII.2 is given in Section D2 of the Appendix. Now we present a bound on the estimation of the precision matrix (inverse of a covariance matrix).

Theorem VIII.3. [Theorem 13 in [28], specialized for $\mathbb{E}[\boldsymbol{x}]=0$.] Let $\boldsymbol{x}\in\mathbb{R}^D$ be a random vector with $\mathbb{E}[\boldsymbol{x}]=0$ and covariance matrix $\Sigma=\mathbb{E}[\boldsymbol{x}\boldsymbol{x}^T]$. Assume $\|\mathbf{A}\boldsymbol{\Sigma}^{-1}\boldsymbol{x}\|_2\leq C_A$, $\|\mathbf{B}\boldsymbol{\Sigma}^{-1}\boldsymbol{x}\|_2\leq C_B$, $\|\sqrt{\boldsymbol{\Sigma}^{-1}}\boldsymbol{x}\|_2\leq C_x$ almost surely, where $\mathbf{A}\in\mathbb{R}^{d_1\times D}$, $\mathbf{B}\in\mathbb{R}^{d_2\times D}$ are known matrices. Let $\boldsymbol{x}_1,..,\boldsymbol{x}_m$ be a set of m independent copies of $\boldsymbol{x}$ with $\hat{\Sigma}=\frac{1}{m}\sum_{j=1}^m\boldsymbol{x}_j\boldsymbol{x}_j^T$ the finite sample estimator of Σ. Then there exist absolute constants $C_1>0$ and $C_2>0$ such that provided $m>C_1(1+u)C_x^2$, we have with probability at least $1-\exp(-u)$ for any $u>0$ that

$$\left\|\mathbf{A}\left(\hat{\Sigma}^{-1}-\Sigma^{-1}\right)\mathbf{B}^T\right\|_F\leq C_2C_AC_B\sqrt{\frac{1+u}{m}}.$$

The proof of Theorem VIII.3 is given in Section D3 of the Appendix.

Proof of Theorem III.3: By Lemma III.1 $\|\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{z})\|\leq\mathrm{C}_{\mathrm{s}}(\boldsymbol{\theta})$ and by Lemma VIII.2 we have $\mathbb{E}_{\boldsymbol{z}}[\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{z})]=0$. It follows that Theorem VIII.3 is applicable to the score vector $\boldsymbol{x}=\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{z})$ satisfying $\mathbb{E}[\boldsymbol{x}]=0$ and $\|\boldsymbol{x}\|\leq\mathrm{C}_{\mathrm{s}}(\boldsymbol{\theta})$. Making the identifications $\Sigma=\hat{\mathrm{F}}_{\mathrm{G}}$, and $\hat{\Sigma}=\overline{\mathrm{F}_{\mathrm{G}}}(\boldsymbol{\theta})$ and setting $\mathbf{A}=\mathbf{B}=\mathbf{I}$, and $C_A=C_B=\|\boldsymbol{\Sigma}^{-1}\|\mathrm{C}_{\mathrm{s}}(\boldsymbol{\theta})$ results in:

$$\begin{aligned}&\left\|\overline{\mathrm{F}_{\mathrm{G}}}\left(\boldsymbol{\theta}\right)^{-1}-\hat{\mathrm{F}}_{\mathrm{G}}\left(\boldsymbol{\theta}\right)^{-1}\right\|_F\\&\quad\leq C_2\left\|\hat{\mathrm{F}}_{\mathrm{G}}\left(\boldsymbol{\theta}\right)^{-1}\right\|^2\mathrm{C}_{\mathrm{s}}{}^2\left(\boldsymbol{\theta}\right)\sqrt{\frac{1+u}{m}}.\end{aligned}$$

□

E. Proof of Corollary III.3.1

Proof:

$$\begin{aligned}\mathrm{E}\left(\boldsymbol{\theta}\right)&\triangleq\left\|\overline{\mathrm{F}}_{\mathrm{G}}\left(\boldsymbol{\theta}\right)^{-1}-\mathrm{F}_{\mathrm{R}}\left(\boldsymbol{\theta}\right)^{-1}\right\|\\&\leq\left\|\overline{\mathrm{F}}_{\mathrm{G}}\left(\boldsymbol{\theta}\right)^{-1}-\hat{\mathrm{F}}_{\mathrm{G}}\left(\boldsymbol{\theta}\right)^{-1}\right\|+\left\|\hat{\mathrm{F}}_{\mathrm{G}}\left(\boldsymbol{\theta}\right)^{-1}-\mathrm{F}_{\mathrm{R}}\left(\boldsymbol{\theta}\right)^{-1}\right\|\\&\leq\mathrm{B}_{\mathrm{s}}\left(\boldsymbol{\theta}\right)+\left\|\mathrm{F}_{\mathrm{R}}\left(\boldsymbol{\theta}\right)^{-1}\right\|\left\|\hat{\mathrm{F}}_{\mathrm{G}}\left(\boldsymbol{\theta}\right)^{-1}\right\|\eta\left(\boldsymbol{\theta}\right)\end{aligned}\tag{44}$$

The first step follows by the triangle inequality, and the second by applying Theorem III.3 to the first term and upperbounding the spectral norm by the Frobenius norm, and applying Corollary III.2.1 to the second term on the second line in (44). Finally, dividing (44) by $\|\mathrm{F}_{\mathrm{R}}(\boldsymbol{\theta})^{-1}\|$ yields the Corollary. ■

Acknowledgment

The authors would like to thank the anonymous reviewers for their valuable criticism and comment, which helped to improve this manuscript.

References

[1] H. V. Habi, H. Messer, and Y. Bresler, "Learning to bound: A generative Cramér-Rao bound," 2022, *arXiv:2203.03695*.

[2] C. R. Rao, "Information and the accuracy attainable in the estimation of statistical parameters," *Reson. J. Sci. Educ.*, vol. 20, pp. 78–90, 1945.

[3] S. M. Kay, *Fundamentals of Statistical Signal Processing*. Hoboken, NJ, USA: Prentice Hall, 1993.

[4] H. L. Van Trees, *Detection, Estimation, and Modulation Theory, Part I: Detection, Estimation, and Linear Modulation Theory*. Hoboken, NJ, USA: Wiley, 2004.

[5] H. L. Van Trees, *Optimum Array Processing: Part IV of Detection, Estimation, and Modulation Theory*. Hoboken, NJ, USA: Wiley, 2004.

[6] A. Catovic and Z. Sahinoglu, "The Cramér-Rao bounds of hybrid TOA/RSS and TDOA/RSS location estimation schemes," *IEEE Commun. Lett.*, vol. 8, no. 10, pp. 626–628, Oct. 2004.

[7] A. Abdelhamed, M. A. Brubaker, and M. S. Brown, "Noise flow: Noise modeling with conditional normalizing flows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 3165–3173.

[8] J. Carmack, A. Bhatia, J. Robinson, J. Majewski, and S. Kuzdeba, "Neural network generative models for radiofrequency data," in *Proc. IEEE 12th Annu. Ubiquitous Comput. Electron. Mobile Commun. Conf.*, 2021, pp. 0577–0582.

[9] Y. Yang, Y. Li, W. Zhang, F. Qin, P. Zhu, and C.-X. Wang, "Generative-adversarial-network-based wireless channel modeling: Challenges and opportunities," *IEEE Commun. Mag.*, vol. 57, no. 3, pp. 22–27, Mar. 2019.

[10] I. Goodfellow et al., "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[11] D. P. Kingma et al., "Glow: Generative flow with invertible 1x1 convolutions," in *Adv. Neural Inf. Process. Syst.*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., pp. 1–10, 2018. [Online]. Available: https://proceedings.neurips.cc/paper/2018/file/d139db6a236200b21cc7f752979132d0-Paper.pdf

[12] A. Oord et al., "Parallel wavenet: Fast high-fidelity speech synthesis," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 3918–3926.

[13] I. Kobyzev, S. J. D. Prince, and M. A. Brubaker, "Normalizing flows: An introduction and review of current methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 11, pp. 3964–3979, Nov. 2021.

[14] G. Papamakarios, E. Nalisnick, D. J. Rezende, S. Mohamed, and B. Lakshminarayanan, "Normalizing flows for probabilistic modeling and inference," *J. Mach. Learn. Res.*, vol. 22, no. 57, pp. 1–64, 2021.

[15] S. Fortunati, F. Gini, M. S. Greco, and C. D. Richmond, "Performance bounds for parameter estimation under misspecified models: Fundamental findings and applications," *IEEE Signal Process. Mag.*, vol. 34, no. 6, pp. 142–157, Nov. 2017.

[16] H. V. Habi, "Generative Cramer Rao bound," 2022. [Online]. Available: https://github.com/haihabi/GenerativeCRB

[17] E. L. Lehmann and G. Casella, *Theory of Point Estimation*. Berlin, Germany: Springer, 2006.

[18] E. Hoogeboom, T. Cohen, and J. M. Tomczak, "Learning discrete distributions by dequantization," in *Proc. 3rd Symp. Adv. Approx. Bayesian Inference*, pp. 1–16, 2021. [Online]. Available: https://openreview.net/forum?id=a0EpGhKt_R

[19] H. V. Habi, H. Messer, and Y. Bresler, "A generative Cramér-Rao bound on frequency estimation with learned measurement distribution," in *Proc. IEEE 12th Sensor Array Multichannel Signal Process. Workshop*, 2022, pp. 176–180.

[20] D. Nielsen and O. Winther, "Closing the dequantization gap: PixelCNN as a single-layer flow," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 3724–3734.

[21] A. Paszke et al., "Pytorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 8026–8037.

[22] S. A. Kassam and H. V. Poor, “Robust techniques for signal processing: A survey,” *Proc. IEEE*, vol. 73, no. 3, pp. 433–481, Mar. 1985.

[23] G. Lugosi and S. Mendelson, “Robust multivariate mean estimation: The optimality of trimmed mean,” *Ann. Statist.*, vol. 49, no. 1, pp. 393–410, 2021.

[24] E. Yang, A. C. Lozano, and A. Aravkin, “A general family of trimmed estimators for robust high-dimensional data analysis,” *Electron. J. Statist.*, vol. 12, no. 2, pp. 3519–3553, 2018.

[25] A. B. Tsybakov, *Introduction to Nonparametric Estimation*, Berlin, Germany: Springer, 2009, doi: 10.1007/b13794.

[26] P. Hammad, “Mesure d’ordre α de l’information au sens de Fisher,” *Revue de Statistique Appliquée*, vol. 26, no. 1, pp. 73–84, 1978.

[27] P. Zegers, “Fisher information properties,” *Entropy*, vol. 17, no. 7, pp. 4918–4939, 2015.

[28] Ž. Kereta and T. Klock, “Estimating covariance and precision matrices along subspaces,” *Electron. J. Statist.*, vol. 15, no. 1, pp. 554–588, 2021.

[29] H. Lee, C. Pabbaraju, A. P. Sevekari, and A. Risteski, “Universal approximation using well-conditioned normalizing flows,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 12700–12711.

[30] A. Verine, B. Negrevergne, F. Rossi, and Y. Chevaleyre, “On the expressivity of bi-Lipschitz normalizing flows,” *ICML Workshop Invertible Neural Netw., Normalizing Flows, and Explicit Likelihood Models*, 2021. [Online]. Available: https://openreview.net/forum?id=URKYsI2TFl

[31] Z. Kong and K. Chaudhuri, “The expressive power of a class of normalizing flow models,” in *Proc. Int. Conf. Artif. Intell. Statist.*, 2020, pp. 3599–3609.

[32] Z. Kong and K. Chaudhuri, “Universal approximation of residual flows in maximum mean discrepancy,” in *Proc. Workshop Invertible Neural Netw. Normalizing Flows, Explicit Likelihood Models*, 2021, pp. 1–8. [Online]. Available: https://openreview.net/forum?id=-g3Ae5tWZfm

[33] A. Lugmayr, M. Danelljan, L. Van Gool, and R. Timofte, “Srflow: Learning the super-resolution space with normalizing flow,” in *Proc. 16th Eur. Conf. Comput. Vis.*, 2020, pp. 715–732.

[34] C. Durkan, A. Bekasov, I. Murray, and G. Papamakarios, “Cubic-spline flows,” *ICML Workshop Invertible Neural Nets Normalizing Flows*, 2019. [Online]. Available: https://invertibleworkshop.github.io/INNF_2019/accepted_papers/pdfs/INNF_2019_paper_15.pdf

[35] A. Foi, M. Trimeche, V. Katkovnik, and K. Egiazarian, “Practical poissonian-Gaussian noise modeling and fitting for single-image raw-data,” *IEEE Trans. Image Process.*, vol. 17, no. 10, pp. 1737–1754, Oct. 2008.

[36] J. Zhang and K. Hirakawa, “Improved denoising via Poisson mixture modeling of image sensor noise,” *IEEE Trans. Image Process.*, vol. 26, no. 4, pp. 1565–1578, Apr. 2017.

[37] J. Chen, J. Chen, H. Chao, and M. Yang, “Image blind denoising with generative adversarial network based noise modeling,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3155–3164.

[38] A. Abdelhamed, S. Lin, and M. S. Brown, “A high-quality denoising dataset for smartphone cameras,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 1692–1700.

[39] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *Proc. 3rd Int. Conf. Learn. Representations*, 2015. [Online]. Available: http://arxiv.org/abs/1412.6980

[40] S. Elfwing, E. Uchibe, and K. Doya, “Sigmoid-weighted linear units for neural network function approximation in reinforcement learning),” *Neural Netw.*, vol. 107, pp. 3–11, 2018.

[41] Abdelrahman Abdelhamed, Marcus A. Brubaker, and Michael S. Brown, “Noise flow: Noise modeling with conditional normalizing flows,” *GitHub Repository*, 2019. [Online]. Available: https://github.com/BorealisAI/noise_flow

[42] P. Stoica and P. Babu, “The Gaussian data assumption leads to the largest Cramér-rao bound [lecture notes],” *IEEE Signal Process. Mag.*, vol. 28, no. 3, pp. 132–133, May 2011.

[43] L. J. M. Aslett, “Statistical machine learning,” 2021. [Online]. Available: https://www.louisaslett.com/StatML/notes/

[44] K. B. Petersen and M. S. Pedersen, “The matrix cookbook,” Tech. Univ. Denmark, Nov. 2012. [Online]. Available: http://www2.compute.dtu.dk/pubdb/pubs/3274-full.html

[45] L. Dinh, J. Sohl-Dickstein, and S. Bengio, “Density estimation using real NVP,” in *Proc. 5th Int. Conf. Learn. Representations*, 2017. [Online]. Available: https://openreview.net/forum?id=HkpbnH9lx

[46] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[47] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 2012.

[48] M. Abadi et al., “Tensorflow: A system for large-scale machine learning,” in *Proc. 12th USENIX Symp. Oper. Syst. Design Implement.*, 2016, vol. 16, no. 2016, pp. 265–283.

Hai Victor Habi (Graduate Student Member, IEEE) received the B.Sc. degree (*cum laude*) in electrical engineering from the Holon Institute of Technology, Holon, Israel, in 2014, and the M.Sc. degree in electrical and computer engineering from Tel Aviv University, Tel Aviv, Israel, in 2020. He is currently working toward the Ph.D. degree in electrical and computer engineering. His research interests include machine learning, deep learning, and statistical signal processing. He also leads a Research and Algorithm Team with Sony Semiconductors Israel Ltd. (Formerly Altair Semiconductor), Hod Hasharon, Israel.

Hagit Messer (Life Fellow, IEEE) received the Ph.D. degree in electrical engineering from Tel Aviv University (TAU), Tel Aviv, Israel. After completing a Post Doctoral Fellowship with Yale University, New Haven, CT, USA, she joined the Faculty of Engineering, TAU, in 1986, where she is currently the Kranzberg Chair Professor of signal processing with the School of Electrical Engineering. From 2000 to 2003, while on leave from TAU, she was the Chief Scientist with the Ministry of Science, Israel. On returning to TAU, she became the Head of the Porter School of Environmental Studies from 2004 to 2006, and the Vice President of research and development from 2006 to 2008. Then, she was the President of the Open University, Ra’anana, Israel, and Vice Chairperson of the Council of Higher Education, Israel, from 2013 to 2016. In 2016, she also become a Co-founder of ClimaCell, Boston, MA, USA. She is an Expert in statistical signal processing with applications to source localization, communication, and environmental monitoring. She has authored numerous journal and conference papers, and has supervised tens of graduate students. Since 1993, Dr. Messer has been a member of technical committees of the Signal Processing Society. She is on the Editorial Boards of the IEEE TRANSACTIONS ON SIGNAL PROCESSING, IEEE SIGNAL PROCESSING LETTERS, and IEEE JOURNAL OF SELECTED TOPICS IN SIGNAL PROCESSING, and on the Overview Editorial Board of the Signal Processing Society journals.

Yoram Bresler (Life Fellow, IEEE) received the B.Sc. and M.Sc. degrees in electrical engineering from the Technion–Israel Institute of Technology, Haifa, Israel, and the Ph.D. degree in electrical engineering from Stanford University, Stanford, CA, USA. He is currently a Professor with the Coordinated Science Laboratory, and the Department of Bioengineering, and Founder Professor Emeritus with the Department of Electrical and Computer Engineering, University of Illinois at Urbana– Champaign. He is also the Founder of InstaRecon, Inc., commercializing breakthrough technology for tomographic reconstruction developed in his academic research. His research interests include machine learning and statistical signal processing and their applications to inverse problems in imaging, including compressed sensing, computed tomography, magnetic resonance imaging, and ultrasound. Dr. Bresler is a Fellow of the IAMBE and AIMBE. His papers received four IEEE best journal paper awards, two of these with his students. During 2016–2017, he was an IEEE SPS Distinguished Lecturer. He was the recipient of the 1991 NSF Presidential Young Investigator Award, Technion Faculty Fellowship in 1995, and Xerox Senior Award for Faculty Research in 1998. He was named University of Illinois Scholar in 1999, and was appointed as an Associate with the Center for Advanced Study of the University during 2001–2002. He was a Faculty Fellow with the National Center for Super Computing Applications in 2006. He has served on the editorial board of several journals, including the IEEE TRANSACTIONS ON SIGNAL PROCESSING, the IEEE JOURNAL OF SELECTED TOPICS IN SIGNAL PROCESSING, *Machine Vision and Applications*, and *SIAM Journal on Imaging Science*, and on various committees of the IEEE.