

Deep Bayesian Inversion

Computational uncertainty quantification for large scale inverse problems

Jonas Adler

Department of Mathematics
KTH - Royal Institute of Technology
jonasadl@kth.se
Research and Physics, Elekta

Ozan Öktem

Department of Mathematics
KTH - Royal Institute of Technology
ozan@kth.se

Abstract

Characterizing statistical properties of solutions of inverse problems is essential for decision making. Bayesian inversion offers a tractable framework for this purpose, but current approaches are computationally unfeasible for most realistic imaging applications in the clinic. We introduce two novel deep learning based methods for solving large-scale inverse problems using Bayesian inversion: a sampling based method using a Wasserstein GAN with a novel mini-discriminator and a direct approach that trains a neural network using a novel loss function. The performance of both methods is demonstrated on image reconstruction in ultra low dose 3D helical CT. We compute the posterior mean and standard deviation of the 3D images followed by a hypothesis test to assess whether a “dark spot” in the liver of a cancer stricken patient is present. Both methods are computationally efficient and our evaluation shows very promising performance that clearly supports the claim that Bayesian inversion is usable for 3D imaging in time critical applications.

1 Introduction

In several areas of science and industry there is a need to reliably recover a hidden multidimensional model parameter from noisy indirect observations. A typical example is when imaging/sensing technologies are used in medicine, engineering, astronomy, and geophysics.

These inverse problems are often ill-posed, meaning that small errors in data may lead to large errors in the model parameter and there are several possible model parameter values that are consistent with observations. Addressing ill-posedness is critical in applications where decision making is based on the recovered model parameter, like in image guided medical diagnostics. Furthermore, many highly relevant inverse problems are large-scale; they involve large amounts of data and high-dimensional model parameter spaces.

Bayesian inversion Bayesian inversion is a framework for assigning probabilities to a model parameter given data (posterior) by combining a *data model* with a *prior model* (section 2). The former describes how measured data is

generated from a model parameter whereas the latter accounts for information about the unknown model parameter that is known beforehand. Exploring the posterior not only allows for recovering the model parameter in a reliable manner by computing suitable estimators, it also opens up for a complete statistical analysis including quantification of the uncertainty.

A key part of Bayesian inversion is to express the posterior using Bayes' theorem, which in turn requires access to the data likelihood, a prior, and a probability measure for data. The data likelihood is often given from insight into the physics of how data is generated (simulator). The choice of prior (section 2.1) is less obvious but important since it accounts for a priori information about the true model parameter. It is also very difficult to specify a probability distribution for data, which is required by many estimators. Finally, the computational burden associated with exploring the posterior (section 2.2) prevents usage of Bayesian inversion in most imaging applications.

To exemplify the above, consider clinical 3D computed tomography (CT) imaging where the model parameter represents the interior anatomy and data is x-ray radiographs taken from various directions. A natural prior in this context is that the object (model parameter) being imaged is a human being, but explicitly handcrafting such a prior is yet to be done. Instead, current priors prescribe roughness or sparsity, which suppresses unwanted oscillatory behavior at the expense of finer details. Next, the model parameter is typically 512³-dimensional and data is of at least same order of magnitude. Hence, exploring the posterior in a timely manner is challenging, e.g., uncertainty quantification in Bayesian inversion remains intractable for such large-scale inverse problems.

2 Statistical Approach to Inverse Problems

Uncertainty refers in general to the accuracy by which one can determine a model parameter. In an inverse problems, this rests upon the ability to explore the statistical distribution of model parameters given measured data. More precisely, the posterior probability of the model parameter conditioned on observed data describes all possible solutions to the inverse problem along with their probabilities [21, 19] and it is essential for uncertainty quantification.

Bayesian inversion uses Bayes' theorem [19, Theorem 14] to characterize the posterior:

$$p(x | y) = \frac{p(x)p(y | x)}{p(y)}.$$

Here, $p(y | x)$ is given by the data model that is usually derived from knowledge about how data is generated and $p(x)$ is given by the prior model that represents information known beforehand about the true (unknown) model parameter.

A tractable property of Bayesian inversion is that small changes in data lead to small changes in the posterior even when the inverse problem is ill-posed in the classical sense [19, Theorem 16], so Bayesian inversion is stable. Different reconstructions can be obtained by computing different estimators from the posterior and there is also a natural framework for uncertainty quantification, e.g., by computing Bayesian credible sets.

The posterior is however quite complicated with no closed form expression, so much of the contemporary research focuses on realizing the aforementioned advantages with Bayesian inversion without requiring access to the full posterior,

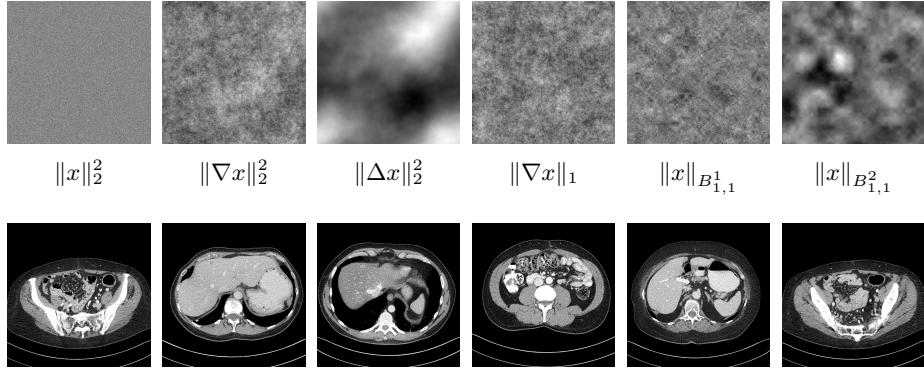


Fig. 1: Top row shows a single random sample generated by a Gibbs type of roughness priors that are common in inverse problems in imaging (appendix E). Such a prior is proportional to $e^{-S(x)}$ and images show samples for different choices of S . Bottom row shows typical samples of normal dose CT images of humans. Ideally, a prior generates samples similar to those in the bottom row.

see [19] for a nice survey. Some related key challenges were mentioned earlier in the introduction; choosing a “good” *prior*, specifying the *probability distribution of data*, and to explore the posterior in a *computationally feasible* manner.

2.1 Choosing a prior model

The difficulty in selecting a prior model lies in capturing the relevant a priori information. Bayesian non-parametric theory [25] provides a large class of handcrafted priors, but these only capture a fraction of the a priori information that is available. Figure 1 illustrates this by showing random samples generated from priors commonly used by state-of-the-art approaches in image recovery [37, 15] as well as samples from typical clinical CT images. The handcrafted priors primarily encode regularity properties, like roughness or sparsity, and it would clearly be stretching our imagination to claim that corresponding samples represent natural images.

2.2 Computational feasibility

Exploring the posterior for inverse problems in imaging often leads to large scale numerics since this mounts to sampling from a high dimensional probability distribution. Most approaches, see section 6, are either not fast enough or rely on simplifying assumptions that does not hold in many applications. For the above reasons, in large scale inverse problems one tends to reconstruct a *single point estimate* of the posterior distribution, the most common being the maximum a posteriori (MAP) estimator that corresponds to the most likely reconstruction given the data. A drawback that comes with working with single estimators is that these cannot include all the information present in the posterior distribution. It is clear that knowledge about the full posterior would have dramatic impact upon how solutions to inverse problems are intertwined into decision making. As an example, in medical imaging, practitioners would be able to

compute the probability of a tumor being an image artifact, which in turn is necessary for image guided hypothesis testing.

3 Contribution

Our overall contribution is to suggest two generic, yet adaptable, frameworks for uncertainty quantification in inverse problems that are computationally feasible and where both the prior and probability distribution of data are given implicitly through supervised examples instead of being handcrafted. The approach is based on recent advances in generative adversarial networks (GANs) from deep learning and we demonstrate its performance on ultra low dose 3D helical CT.

Our main contribution is *Deep Posterior Sampling* (section 4.1) where generative models from machine learning are used to sample from a high-dimensional unknown posterior distribution in the context of Bayesian inversion. This is made possible by a novel *conditional Wasserstein GAN (WGAN)* discriminator (appendix C.2). The approach is generic and applies in principle to any inverse problem assuming there is relevant training data. It can be used for performing statistical analysis of the posterior on X , e.g., by computing various estimators.

Independently, we also introduce *Deep Direct Estimation* (section 4.2) where one directly computes an estimator using a deep neural network trained using a cleverly chosen loss (appendix C.3). Deep direct estimation is faster than posterior sampling, but it mainly applies to statistical analysis that is based on evaluating a pre-determined estimator. Both approaches should give similar quantitative results when used for evaluating the same estimator.

We demonstrate the performance and computational feasibility for ultra low dose CT imaging in a clinical setting by computing some estimators and performing a hypothesis test (section 5).

4 Deep Bayesian Inversion

As already stated, in Bayesian inversion both the model parameter x and measured data y are assumed to be generated by random variables x and y , respectively. The ultimate goal is to recover the posterior $\pi(x | y)$, which describes all possible solutions $x = x$ along with their probabilities given data $y = y$.

We here outline two approaches that can be used to perform various statistical analysis on the posterior. Deep Posterior Sampling is a technique for learning how to sample from the posterior whereas Deep Direct Estimation learns various estimators directly.

4.1 Deep Posterior Sampling

The idea is to explore the posterior by sampling from a generator that is defined by a WGAN, which has been trained using a conditional WGAN discriminator.

To describe how a WGAN can be used for this purpose, let data $y \in Y$ be fixed and assume that $\pi(x | y)$, the posterior of x at $y = y$, can be approximated by elements in a parametrized family $\{\mathcal{G}_\theta(y)\}_{\theta \in \Theta}$ of probability measures on X . The best such approximation is defined as $\mathcal{G}_{\theta^*}(y)$ where $\theta^* \in \Theta$ solves

$$\theta^* \in \arg \min_{\theta \in \Theta} \ell(\mathcal{G}_\theta(y), \pi(x | y)). \quad (1)$$

\mathcal{G} is not a generator, but probability measure

Here, ℓ quantifies the “distance” between two probability measures on X . We are however interested in the best approximation for “all data”, so we extend (1) by including an averaging over all possible data. The next step is to choose a distance notion ℓ that is desirable from both a theoretical and a computational point of view. As an example, the distance should be finite and computational feasibility requires using it to be differentiable almost everywhere, since this opens up for using stochastic gradient descent (SGD) type of schemes. The Wasserstein 1-distance \mathcal{W} (appendix A) has these properties [8] and sampling from the posterior $\pi(x | y)$ can then be replaced by sampling from the probability distribution $\mathcal{G}_{\theta^*}(y)$ where θ^* solves

$$\theta^* \in \arg \min_{\theta \in \Theta} \mathbb{E}_{y \sim \sigma} [\mathcal{W}(\mathcal{G}_\theta(y), \pi(x | y))]. \quad (2)$$

In the above, σ is the probability distribution for data and $y \sim \sigma$ generates data.

Observe now that evaluating the objective in (2) requires access to the very posterior that we seek to approximate. Furthermore, the distribution σ of data is often unknown, so an approach based on (2) is essentially useless if the purpose is to sample from an unknown posterior. Finally, evaluating the Wasserstein 1-distance directly from its definition is not computationally feasible.

On the other hand, as shown in appendix C.1, all of these drawbacks can be circumvented by rewriting (2) as an expectation over the joint law $(x, y) \sim \mu$. This makes use of specific properties of the Wasserstein 1-distance (Kantorovich-Rubenstein duality) and one obtains the following approximate version of (2):

$$\theta^* \in \arg \min_{\theta \in \Theta} \left\{ \sup_{\phi \in \Phi} \mathbb{E}_{(x, y) \sim \mu} \left[D_\phi(x, y) - \mathbb{E}_{z \sim \eta} [D_\phi(G_\theta(z, y), y)] \right] \right\}. \quad (3)$$

In the above, $G_\theta: Z \times Y \rightarrow X$ (generator) is a deterministic mapping such that $G_\theta(z, y) \sim \mathcal{G}_\theta(y)$, where $z \sim \eta$ is a ‘simple’ Z -valued random variable in the sense that it can be sampled in a computationally feasible manner. Next, the mapping $D_\phi: X \times Y \rightarrow \mathbb{R}$ (discriminator) is a measurable mapping that is 1-Lipschitz in the X -variable.

On a first sight, it might be unclear why (3) is better than (2) if the aim is to sample from the posterior, especially since the joint law μ in (3) is unknown. The advantage becomes clear when one has access to supervised training data for the inverse problem, i.e., i.i.d. samples $(x_1, y_1), \dots, (x_m, y_m)$ generated by the random variable $(x, y) \sim \mu$. The μ -expectation in (3) can then be replaced by an averaging over training data.

To summarize, solving (3) given supervised training data in $X \times Y$ amounts to learning a generator $G_{\theta^*}(z, \cdot): Y \rightarrow X$ such that $G_{\theta^*}(z, y)$ with $z \sim \eta$ is approximately distributed as the posterior $\pi(x | y)$. In particular, for given $y \in Y$ we can sample from $\pi(x | y)$ by generating values of $z \mapsto G_{\theta^*}(z, y) \in X$ in which $z \in Z$ is generated by sampling from $z \sim \eta$.

An important part of the implementation is the concrete parameterizations of the generator and discriminator:

$$G_\theta: Z \times Y \rightarrow X \quad \text{and} \quad D_\phi: X \times Y \rightarrow \mathbb{R}.$$

We here use deep neural networks for this purpose and following [27], we softly enforce the 1-Lipschitz condition on the discriminator by including a gradient

approximate!?!?!

only X!!

penalty term to the training objective in (3). Furthermore, if (3) is implemented as is, then in practice \mathbf{z} is not used by the generator (so called mode-collapse). To solve this problem, we introduce a novel conditional mini-batch discriminator that can be used with conditional WGAN without impairing upon its analytical properties (claim 1), see appendix C.2 for more details.

4.2 Deep Direct Estimation

The idea here is to train a deep neural network to directly approximate an estimator of interest without resorting to generating samples from the posterior as in posterior sampling (section 4.1).

Deep direct estimation relies on the well known result:

$$\mathbb{E}_w[w | y = \cdot] = \min_{h: Y \rightarrow W} \mathbb{E}_{(y,w)} [\|h(y) - w\|_W^2]. \quad (4)$$

The solution of regression problem gives

conditional mean

In the above, w is *any* random variable taking values in some measurable Banach space W and the minimization is over all W -valued measurable maps on Y . See proposition 2 in appendix C.3 for a precise statement. This is useful since many estimators relevant for uncertainty quantification are expressible using terms of this form for appropriate choices of w .

Specifically, appendix C.3 considers two (deep) neural networks $\mathcal{T}_{\theta^*}^\dagger: Y \rightarrow X$ and $h_{\phi^*}: Y \rightarrow X$ with appropriate architectures that are trained according to

$$\begin{aligned} \theta^* &\in \arg \min_{\theta} \left\{ \mathbb{E}_{(x,y) \sim \mu} [\|x - \mathcal{T}_{\theta}^\dagger(y)\|_X^2] \right\} \quad \text{Theta is fixed} \\ \phi^* &\in \arg \min_{\phi} \left\{ \mathbb{E}_{(x,y) \sim \mu} [\|h_{\phi}(y) - (x - \mathcal{T}_{\theta^*}^\dagger(y))^2\|_X^2] \right\}. \end{aligned}$$

In my approach,
I predicted
 $\langle x \rangle$ and $\langle x^2 \rangle$,
and this gave
me residual.
But here a sequential
approach is
proposed

The resulting networks will then approximate the conditional mean and the conditional point-wise variance, respectively. Finally, if one has supervised training data (x_i, y_i) , then the joint law μ above can be replaced by its empirical counterpart and the μ -expectation is replaced by an averaging over training data.

As already indicated, by using (4) it is possible to re-write many estimators as minimizers of an expectation. Such estimators can then be approximated using the direct estimation approach outlined here. This should coincide with computing the same estimator by posterior sampling (section 4.1). Direct estimation is however significantly faster, but not as flexible as posterior sampling since each estimator requires a new neural network that specifically trained for that estimator. Section 5 compares outcome from both approaches.

5 Numerical Experiments

We evaluate the feasibility of posterior sampling (section 4.1) to sample from the posterior and Direct Estimation (section 4.2) to compute mean and point-wise variances for clinical 3D CT imaging.

5.1 Set-up

Our supervised data consists of pairs of 3D CT images (x_i, y_i) generated by (x, y) where x_i is a normal dose 3D image that serves as the ‘ground truth’ and

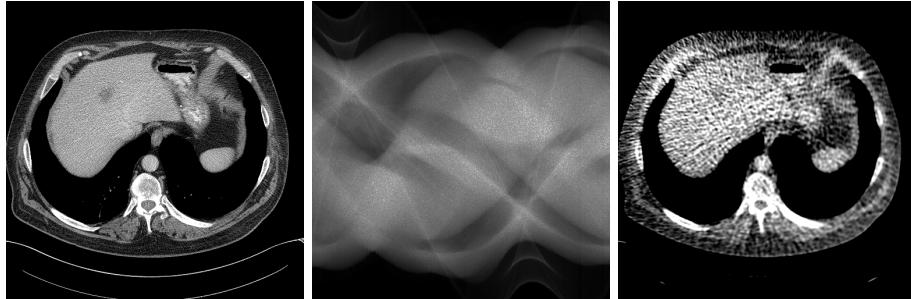


Fig. 2: Test data: Normal dose image (left), subset of CT data from a ultra-low dose 3D helical scan (middle), and corresponding FBP reconstruction (right). Images are shown using a display window set to $[-150, 200]$ HU.

y_i is the filtered back-projection (FBP) 3D reconstruction computed from **ultra low dose CT data associated with x_i** .

One could here let y_i be the ultra low dose CT data itself, which results in more complex architectures of the neural networks. On the other hand, using FBP as a pre-processing step (i.e., y_i is FBP reconstruction from ultra low dose data) simplifies the choice of architectures and poses no limitation in the **theoretical setting with infinite data** (see [4, section 8]).

Training data We used training data from the Mayo Clinic Low Dose CT challenge [47]. This data consists of **ten CT scans**, of which we use **nine for training and one for evaluation**. Each 3D image x_i has a corresponding ultra low dose data that is generated by using **only $\approx 10\%$ of the full data** and adding **additional Poisson noise** so that **the dose corresponds to 2% of normal dose**. Applying FBP on this data yields the ultra low dose CT images, see appendix D.1 for a detailed description.

An example of normal dose CT reconstruction, tomographic data, and the ultra low dose FBP reconstruction is shown in fig. 2.

Network architecture and training The operators

$$\begin{array}{ll} \text{generator} & G_\theta: Z \times Y \rightarrow X \\ \text{discriminator} & D_\phi: X \times Y \rightarrow \mathbb{R} \end{array} \quad \begin{array}{ll} \mathcal{T}_{\theta^*}^\dagger: Y \rightarrow X & \text{conditional mean estimate} \\ h_{\phi^*}: Y \rightarrow X & \text{conditional var estimate} \end{array}$$

are represented by **multi-scale residual neural networks**. For computational reasons, we applied the method **slice-wise**, see appendix D.2 for details regarding the exact choice of architecture and training procedure.

The parts related to the inverse problem (tomography) were implemented using the ODL framework [2] with ASTRA [62] as back-end for computing the ray-transform and its adjoint. The learning components were implemented in TensorFlow [1].

5.2 Results

Estimators A **typical use-case of Bayesian inversion** is to **compute estimators from the posterior**. In our case, **we are interested in the conditional mean and point-wise standard deviation (square root of variance)**.

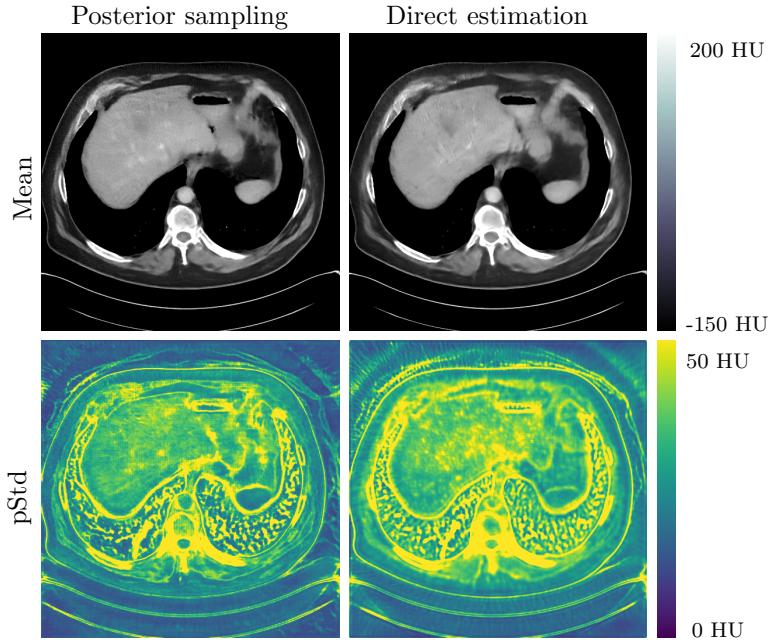


Fig. 3: Conditional mean and point-wise standard deviation (pStd) computed from test data (fig. 2) using posterior sampling (section 4.1) and direct estimation (section 4.2).

When using **posterior sampling**, we compute the conditional mean and point-wise standard deviations based on **1 000 images sampled from the posterior**, see appendix B for some examples of such images. For direct estimation we simply evaluated the associated trained networks. Both approaches are computationally feasible, the time needed per slice to compute these estimators is **40 s** using posterior sampling based on 1 000 samples and **80 ms** for direct estimation.

The mean and standard-deviations that were computed using both methods are shown in fig. 3. We note that results from the methods **agree very well with each other**, indicating that the posterior samples follow the posterior quite well, or at least that the methods have similar bias. The **posterior mean looks** as one would expect, **with highly smoothed features** due to the high noise level. Likewise, the standard deviation is also as one would expect, **with high uncertainties around the boundaries of the high contrast objects**. We also note that the standard deviation at the white “blobs” that appear in some samples (see appendix B) is quite high, indicating that the model is uncertain about their presence. There is also a background uncertainty at ≈ 20 HU due to point-wise noise in the reference normal-dose scans that we take as ground truth.

Uncertainty quantification We here show how to use Bayesian credible sets for clinical image guided decision making. One computes a reconstruction from ultra low dose data (middle image in fig. 2), identifies one or more features, and then seeks to estimate the likelihood for the presence of these features.

Formalizing the above, let Δ denote the difference in mean intensity in the reconstructed image between a region encircling the feature and the surrounding

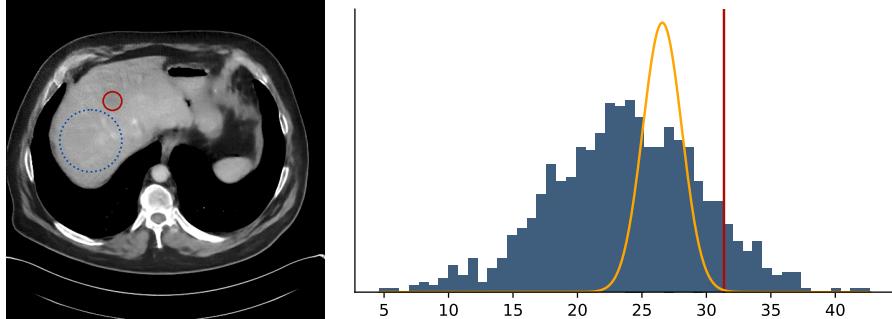


Fig. 4: The suspected tumor (red) and the reference region (blue) shown in the sample posterior mean image. Right plot shows average contrast differences between the tumor and reference region. The histogram is computed by posterior sampling applied to test data (fig. 2), the yellow curve is from direct estimation, and the true value is the red threshold.

organ, which in our example is the liver. The feature is said to “exist” whenever Δ is bigger than a certain threshold, say 10 HU.

To use posterior sampling, start by computing the conditional mean image (top left in fig. 3) by sampling from the posterior using the conditional WGAN approach in section 4.1. There is a dark “spot” in the liver (possible tumor) and a natural clinical question is to statistically test for the presence of this feature. To do this, compute Δ for a number of samples generated by posterior sampling, which here is the same 1 000 samples used for computing the conditional mean. We estimate $p := \text{Prob}(\Delta > 10 \text{ HU})$ from the resulting histogram in fig. 4 and clearly $p > 0.95$, indicating that the “dark spot” feature exists with at least 95% significance. This is confirmed by the ground truth image (left image in fig. 2). The conditional mean image also under-estimates Δ , whose true value is the vertical line in fig. 4. This is to be expected since the prior introduces a bias towards homogeneous regions, a bias that decreases as noise level decreases.

To perform the above analysis using direct estimation, start with computing the conditional mean image from the same ultra-low dose data using direct estimation. As expected, the resulting image (top right in fig. 3) shows a dark “spot” in the liver. Now, designing and training a neural network that directly estimates the distribution of Δ is unfeasible in a general setting. However, as shown in section 4.2, this is possible if one assumes pixels are independent of each other. The estimated distribution of Δ is the curve in fig. 4 and we get $p > 0.95$, which is consistent with the result obtained using posterior sampling. The direct estimation approach is based on assuming independent pixels, so it will significantly underestimate the variance. In contrast, the approach based on posterior sampling seems to give a more realistic estimate of the variance.

6 Related Work

Deep learning based methods are increasingly used for medical image reconstruction, either by using deep learning for post-processing [38, 35] or by integrating deep learning into the image reconstruction [66, 5, 6, 29, 16, 28, 30]. These

papers start by specifying the loss and then use a deep neural network to minimize the expected loss. This essentially amounts to directly computing a Bayes estimator with a risk given by the loss. The loss is often the squared L_2 -distance, which implicitly implies that one approximates the conditional mean. Hence, the above approaches could be seen as examples of deep direct estimation. There is however an important difference, in deep direct estimation one starts by explicitly specify the estimator, which then implies the appropriate loss function.

There has also been intense research in selecting a loss function different from the L_2 -loss [36, 7] and specifically GAN-like methods have been applied to image post-processing in CT [64, 67] and image reconstruction in magnetic resonance imaging (MRI) (Fourier inversion) [65, 46]. However, in these papers the authors discard providing any randomness to the generator, instead only giving it the prior. They have thus not fully realized the potential of using GANs for sampling from the posterior in Bayesian inversion.

Regarding sampling from a posterior, conditional generative models [48, 49] have been widely used in the machine learning literature for this purpose. Typical use cases is to sample from a posterior where an image is conditioned on a text, like “the bird is yellow” [32, 20], but also for simple applications in imaging, including image super-resolution and in-painting [44, 52, 51]. These approaches do not consider sampling from the posterior for more elaborate inverse problems that involve a physics driven data likelihood. An approach in this direction is presented in [41] where variational auto-encoders are used to sample from the posterior of possible segmentations (model parameter) given CT images (data).

An entirely different class of methods for exploring the posterior are based on Markov chain Monte Carlo (MCMC) techniques, which have revolutionized mathematical computation and enabled statistical inference within many previously intractable models. Most of the techniques are rooted in solid mathematical theory, but they are limited to cases where the prior model is known in closed form, see surveys in [19, 15, 11]. Furthermore, these MCMC techniques are still computationally unfeasible for large-scale inverse problems, like 3D clinical CT.

A computationally feasible alternative to MCMC for uncertainty quantification is to consider asymptotic characterizations of the posterior. For many inverse problems, it is possible to prove Bernstein–von Mises type of theorems that characterizes the posterior using analytic expressions assuming the prior is asymptotically uninformative [50]. Such characterizations do not hold for finite data, but assuming a Gaussian process model (data likelihood and prior are both Gaussian) allows for using numerical methods for linear inverse problems [55]. Gaussian process models are however still computationally demanding and it can be hard to design appropriate priors, so [23, 24] introduces (conditional) neural processes that incorporate deep neural networks into Gaussian process models for learning more general priors.

Finally, another computationally feasible approach for uncertainty quantification is to approximate Bayesian credible sets for the MAP estimator by solving a convex optimization problem [57, 54]. The approach is however restricted to the MAP estimator and furthermore, it requires access to a handcrafted prior.

7 Conclusions

Bayesian inversion is an elegant framework for recovering model parameters along with uncertainties that applies to a wide range of inverse problems. The traditional approach requires specifying a prior and, depending on the choice of estimator, also the probability of data. Furthermore, exploring the posterior remains a computational challenge. Hence, despite significant progress in theory and algorithms, Bayesian inversion remains unfeasible for most large scale inverse problems, like those arising in imaging.

This paper addresses *all* these issues, thereby opening up for the possibility to perform Bayesian inversion on large scale inverse problems. Capitalizing on recent advances in deep learning, we present two approaches for performing Bayesian inversion: *Deep Posterior Sampling* (section 4.1), which uses a GAN to sample from the posterior, and *Deep Direct Estimation* (section 4.2) that computes an estimator directly using a deep neural network.

The performance of both approaches is demonstrated in the context of ultra low dose (2% of normal dose) clinical 3D helical CT imaging (section 5). We show how to compute basic Bayesian estimators, like the posterior mean and point-wise standard deviation. We also compute Bayesian credible sets and use this for testing whether a suspected “dark spot” in the liver, which is visible in the posterior mean image, is real. The quality of the posterior mean reconstruction is also quite promising, especially bearing in mind that it is computed from CT data that corresponds to 2% of normal dose.

To the best of our knowledge, *this is the first time one can perform such computations on large scale inverse problems in a timely manner, like clinical 3D helical CT image reconstruction*. On the other hand, using such a radically different way to perform image reconstruction in clinical practice quickly gets complicated since it must be preceded by clinical trials in the context of image guided decision making. However, there are many advantages that comes with using our proposed approach, which for medical imaging means integrating imaging with clinical decision making while accounting for the uncertainty.

To conclude, the posterior sampling approach allows one to perform Bayesian inversion on large scale inverse problems that goes beyond computing specific estimators, such as the MAP or conditional mean. The framework is not specific to tomography, it applies to essentially any inverse problem assuming access to sufficient amount of “good” supervised training data. Furthermore, the possibility to efficiently sample from the posterior opens up for new ways to integrate decision making with reconstruction.

8 Discussion and Outlook

There are several open research topics related to using GANs as generative models for the posterior in inverse problems.

One natural topic is to have a precise notion of “good” supervised training data. Specifically, it is desirable to estimate the amount of supervised training data necessary for “resolving” the posterior/estimator up to some accuracy. Unfortunately, most of the current theory for Bayesian inference does not apply directly to this setting. Its emphasis is on characterizing the posterior in the asymptotic regime where information content in data increases indefinitely and

the prior is asymptotically non-informative, like when a Gaussian prior is used.

Another research topic is to study whether there are theoretical guarantees that ensure the conditional WGAN generator given by (3) converges towards the posterior. In [26] one proves that given infinite capacity of the discriminator, the optimal generator minimizes the Jensen–Shannon divergence w.r.t. the target distribution. For the case with WGAN, [56, Lemma 6] shows that one can learn the posterior in the sense of [60, Definition 3.1], i.e. solving (10) arbitrarily well, given enough training data. But this does not settle the question of what happens with realistic sample and model capacities. This is part of a more general research theme for investigating the theoretical basis for using GANs trained on supervised data to sample from high dimensional probability distributions [9].

Yet another topic relates to including explicit knowledge about the data likelihood, which in contrast to the prior, can be successfully handcrafted for many inverse problems. This is essential for large-scale inverse problems where the amount of supervised training data is little and there are few opportunities for re-training when data the acquisition protocol changes. In this work, this knowledge was implicitly accounted for by our choice to use a FBP reconstruction as the data. While it can be proven that this is in theory sufficient for generating samples from the posterior [4, section 8], [5, 6] clearly shows that working directly from measured data gives better results. We therefore expect further improvements to our results by using a conditional WGANs based on convolutional neural network (CNN) architectures that integrate a handcrafted data likelihood, such as those provided by learned iterative reconstruction.

Finally, our deep direct estimators were very easy to train with no major complications, but training the generative models for posterior sampling is still complicated and involves quite a bit of fine tuning and “tricks”. We hope that future research in generative models will improve upon this situation.

Acknowledgments*

The work was supported by the Swedish Foundation of Strategic Research grant AM13-0049, Industrial PhD grant ID14-0055 and by Elekta. The authors also thank Dr. Cynthia McCollough, the Mayo Clinic, and the American Association of Physicists in Medicine, and acknowledge funding from grants EB017095 and EB017185 from the National Institute of Biomedical Imaging and Bioengineering, for providing the data.

Appendices

A The Wasserstein 1-distance

Let X be a measurable separable Banach Space and \mathcal{P}_X the space of probability measures on X . The Wasserstein 1-distance $\mathcal{W}: \mathcal{P}_X \times \mathcal{P}_X \rightarrow \mathbb{R}$ is a metric on \mathcal{P}_X that can be defined as [63, Definition 6.1]

$$\mathcal{W}(p, q) := \inf_{\mu \in \Pi(p, q)} \mathbb{E}_{(x, v) \sim \mu} [\|x - v\|_X] \quad \text{for } p, q \in \mathcal{P}_X. \quad (5)$$

In the above, $\Pi(p, q) \subset \mathcal{P}_{X \times X}$ denotes the family of joint probability measures on $X \times X$ that has p and q as marginals. Note also that we assume \mathcal{P}_X only contains measures where the Wasserstein distance takes finite values (Wasserstein space), see [63, Definition 6.4] for the formal definition.

The Wasserstein 1-distance in (5) can be rewritten using the Kantorovich-Rubinstein dual characterization [63, Remark 6.5 on p. 95], resulting in

$$\mathcal{W}(p, q) = \sup_{\substack{D: X \rightarrow \mathbb{R} \\ D \in \text{Lip}(X)}} \left\{ \mathbb{E}_{x \sim q} [D(x)] - \mathbb{E}_{v \sim p} [D(v)] \right\} \quad \text{for } p, q \in \mathcal{P}_X. \quad (6)$$

Here, $\text{Lip}(X)$ denotes real-valued 1-Lipschitz maps on X , i.e.,

$$D \in \text{Lip}(X) \iff |D(x_1) - D(x_2)| \leq \|x_1 - x_2\|_X \quad \text{for all } x_1, x_2 \in X.$$

The above constraint can be hard to enforce in (6) as is, so following [27, 3] we prefer the *gradient characterization*:

$$D \in \text{Lip}(X) \iff \|\partial D(x)\|_{X^*} \leq 1 \quad \text{for all } x \in X,$$

where ∂ indicates the Fréchet derivative and X^* is the dual space of X . In our setting, X is an L_2 space, which is a Hilbert space so $X^* = X$ and the Fréchet derivative becomes the (Hilbert space) gradient of D .

B Individual Posterior Samples

It is instructive to visually inspect individual random samples of the posterior obtained from the conditional Wasserstein GAN (WGAN) generator.

Generating one such sample is fast, taking approximately 40 ms on a desktop “gaming” PC. Furthermore, as seen in fig. 5, the generated samples look realistic, practically indistinguishable from the ground truth to the *untrained observer*. With that said, some anatomical features are clearly misplaced, e.g., there are white “blobs” (blood vessels) in the liver. These are present because the supervised training set contained images from patients that were given contrast (see bottom row in fig. 1), which has influenced the anatomical prior that is learned from the supervised data.

C Theory of Deep Bayesian Inversion

This section contains the theoretical foundations needed for Deep Bayesian Inversion and derivations of the expressions used in the main article.

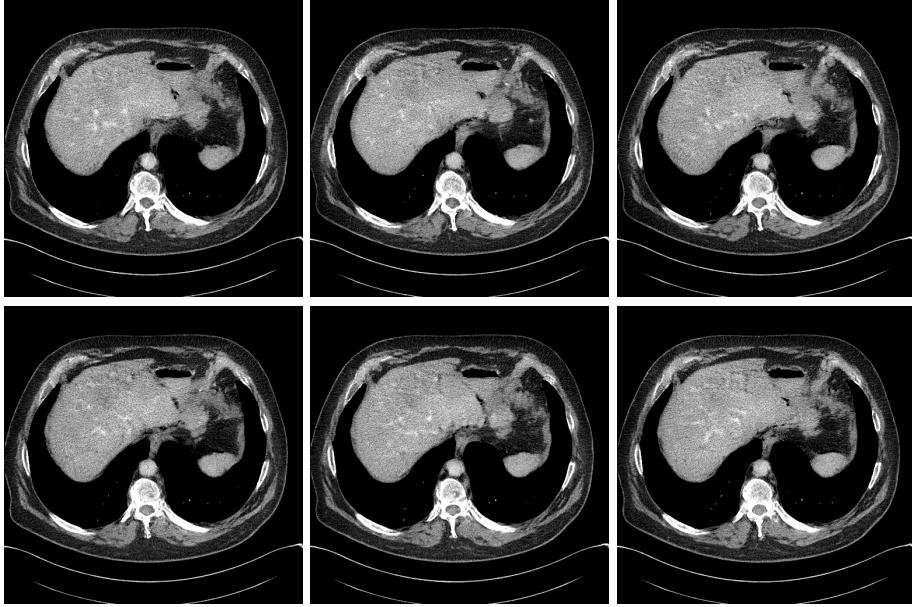


Fig. 5: Deep posterior samples (section 4.1) on test data (fig. 2) shown using a display window set to [-150, 200] HU.

C.1 Derivation of conditional WGAN

This section provides the mathematical details for deriving (3) from (2). Such a reformulation is well-known, but the derivation given here seems to be novel.

The overall aim with WGAN is to approximate the posterior $y \mapsto \pi(x | y)$ that is inaccessible. The approach is to construct a mapping that associates a probability measure on X to each data $y \in Y$. A family of such mappings can be explicitly constructed and trained against supervised training data in order to approximate the posterior. To proceed we need to specify the statistical setting and our starting point is to introduce a “distance” between two probability measures on X :

$$\ell: \mathcal{P}_X \times \mathcal{P}_X \rightarrow \mathbb{R}_+. \quad (7)$$

In the above, \mathcal{P}_X denotes the class of all probability measures on X , so $\pi(x | y) \in \mathcal{P}_X$ whenever the posterior exists, which holds under fairly general assumptions where both X and Y can be infinite-dimensional separable Banach spaces [19, Theorem 14]. Next, let \mathfrak{G} denote a fixed family of *generators*, which are mappings

$$\mathcal{G}: Y \rightarrow \mathcal{P}_X \quad (8)$$

that associate each $y \in Y$ to a probability measure on X . Note here that $y \mapsto \pi(x | y)$ is not necessarily contained in \mathfrak{G} . A generator in \mathfrak{G} is an “optimal” approximation of the posterior $y \mapsto \pi(x | y)$ if it minimizes the expected ℓ -distance, i.e., it solves

$$\inf_{\mathcal{G} \in \mathfrak{G}} \mathbb{E}_{y \sim \sigma} [\ell(\mathcal{G}(y), \pi(x | y))]. \quad (9)$$

Here, $y \sim \sigma$ is the Y -valued random variable generating data.

There are three issues that arise if a solution to (9) is to be used as a proxy for the posterior: (i) Evaluating the objective requires access to the very posterior, which we assumed was inaccessible, (ii) the distribution σ of data is almost always unknown, so the expectation cannot be computed, and finally (iii) the computational feasibility requires access to an explicit finite dimensional parametrization for constructing generators in \mathfrak{G} that one searches over in (9).

As we shall see next, choosing (7) as the Wasserstein 1-distance allows us to address the first two issues. With this choice one can re-write the objective in (9) as an expectation over the joint law $(x, y) \sim \mu$, thereby avoiding expressions that explicitly depend on the unknown posterior $y \mapsto \pi(x | y)$ and distribution of data σ . This joint law is also unknown, but it can often be replaced by its empirical counterpart derived from a suitable supervised training data set.

More precisely, choosing the Wasserstein 1-distance $\mathcal{W}: \mathcal{P}_X \times \mathcal{P}_X \rightarrow \mathbb{R}_+$ as ℓ in (9) yields

$$\inf_{\mathcal{G} \in \mathfrak{G}} \mathbb{E}_{y \sim \sigma} [\mathcal{W}(\mathcal{G}(y), \pi(x | y))]. \quad (10)$$

Note here that $y \mapsto \mathcal{W}(\mathcal{G}(y), \pi(x | y))$ is assumed to be a measurable real-valued function on Y . The Kantorovich-Rubinstein dual characterization in (6) yields

$$\mathcal{W}(\mathcal{G}(y), \pi(x | y)) = \sup_{D_y \in \text{Lip}(X)} \left\{ \mathbb{E}_{x \sim \pi(x | y)} [D_y(x) - D_y(v)] \right\} \quad \text{for } y \in Y. \quad (11)$$

Here, $\text{Lip}(X)$ denotes the set of real-valued mappings on X that are 1-Lipschitz. Hence, (10) can be written as

$$\inf_{\mathcal{G} \in \mathfrak{G}} \mathbb{E}_{y \sim \sigma} \left[\sup_{D_y \in \text{Lip}(X)} \left\{ \mathbb{E}_{x \sim \pi(x | y)} [D_y(x) - D_y(v)] \right\} \right]. \quad (12)$$

Next, in this case the supremum commutes with the σ -expectation, i.e.,

$$\begin{aligned} & \mathbb{E}_{y \sim \sigma} \left[\sup_{D_y \in \text{Lip}(X)} \left\{ \mathbb{E}_{x \sim \pi(x | y)} [D_y(x) - D_y(v)] \right\} \right] \\ &= \sup_{D \in \mathcal{D}(X \times Y)} \left\{ \mathbb{E}_{(x,y) \sim \mu} [D(x, y) - D(v, y)] \right\}, \end{aligned} \quad (13)$$

where $\mathcal{D}(X \times Y)$ is the space of measurable real-valued mappings on $X \times Y$ that are 1-Lipschitz in the X -variable for every $y \in Y$. The proof of (13) is given on p. 17 and combining it with (12) gives

$$\inf_{\mathcal{G} \in \mathfrak{G}} \left\{ \sup_{D \in \mathcal{D}(X \times Y)} \mathbb{E}_{(x,y) \sim \mu} [D(x, y) - D(v, y)] \right\}. \quad (14)$$

Note that there are no approximations involved in going from (10) to (14), the derivation is solely based on properties of the Wasserstein 1-distance. Furthermore, the advantage of (14) over (10) is that the latter neither involves the posterior nor σ (the probability measure of data). It does involve the joint law $(x, y) \sim \mu$, which is of course unknown. On the other hand, if we have access to supervised training data:

$$(x_1, y_1), \dots, (x_m, y_m) \in X \times Y \quad \text{are i.i.d. samples of } (x, y) \sim \mu, \quad (15)$$

then we can replace the joint law μ in (14) with its empirical counterpart and the μ -expectation is replaced by an averaging over training data.

The final steps concern computational feasibility. We start by considering parameterizations of the generators in \mathfrak{G} that enables one to solve (14) in a computational feasible manner. A key aspect is to evaluate the $\mathcal{G}(y)$ -expectation for any $y \in Y$ without impairing upon the ability to approximate the posterior with elements from \mathfrak{G} . We will assume that each generator $\mathcal{G} \in \mathfrak{G}$ corresponds to a measurable map $G: Z \times Y \rightarrow X$ such that the following holds:

$$v \sim \mathcal{G}(y) \iff v = G(z, y) \quad \text{for some } Z\text{-valued random variable } z \sim \eta. \quad (16)$$

In the above, Z is some fixed set and η is a “simple” probability measure on Z meaning that there are computationally efficient means for generating samples of $z \sim \eta$. It is then possible to express (14) as

$$\inf_{G \in \mathfrak{G}} \left\{ \sup_{D \in \mathcal{D}(X \times Y)} \mathbb{E}_{(x,y) \sim \mu} \left[D(x, y) - D(G(z, y), y) \right] \right\} \quad (17)$$

where \mathfrak{G} is the class of X -valued measurable maps on $Z \times Y$ that corresponds to \mathfrak{G} by (16).

The formulation in (17) involves taking the infimum over \mathfrak{G} and supremum over \mathcal{D} , which is clearly computationally unfeasible. Hence, one option is to consider a parametrization of these spaces using deep neural networks with appropriately chosen architectures:

$$\mathfrak{G} := \{G_\theta\}_{\theta \in \Theta} \quad \text{where } G_\theta: Z \times Y \rightarrow X \quad (18)$$

$$\mathcal{D} := \{D_\phi\}_{\phi \in \Phi} \quad \text{where } D_\phi: X \times Y \rightarrow \mathbb{R}. \quad (19)$$

Inserting the above parametrizations into (17) results in

$$\theta^* \in \arg \min_{\theta \in \Theta} \left\{ \sup_{\phi \in \Phi} \mathbb{E}_{(x,y) \sim \mu} \left[D_\phi(x, y) - D_\phi(G_\theta(z, y), y) \right] \right\}. \quad (20)$$

Note again that the unknown joint law μ in (20) is replaced by its empirical counterpart given from the training data in (15).

To summarize, solving the training problem in (20) given training data (15) and the parametrizations in (18) and (19) yields a mapping $G_{\theta^*}: Z \times Y \rightarrow X$ that approximates the posterior in the sense that the distribution of $G_{\theta^*}(z, y)$ with $z \sim \eta$ is closest to $\pi(x | y)$ in expected Wasserstein 1-distance. Hence, we can sample $z \in Z$ from $z \sim \eta$ and $G_{\theta^*}(z, y) \in X$ will approximate a sample of the conditional random variable $(x | y = y) \sim \pi(x | y)$. The formulation in (20) is also suitable for stochastic gradient descent (SGD), so computational techniques from deep neural networks can be used for solving the empirical expected minimization problem. We conclude with providing a proof of (13).

Proof of (13) To simplify the notational burden, define $f_D: Y \rightarrow \mathbb{R}$ as

$$f_D(y) := \mathbb{E}_{\substack{x \sim \pi(x|y) \\ v \sim \mathcal{G}(y)}} [D(x) - D(v)] \quad \text{for } D \in \text{Lip}(X).$$

Next, $(x, y) \sim \mu$ with $\mu = \pi(x | y) \otimes \sigma$, so by the law of total expectation we can re-write the objective in the right-hand side of (13) as

$$\mathbb{E}_{\substack{(x,y) \sim \mu \\ v \sim \mathcal{G}(y)}} [D(x, y) - D(v, y)] = \mathbb{E}_{y \sim \sigma} [f_D(\cdot, y)].$$

Hence, proving (13) is equivalent to proving

$$\mathbb{E}_{y \sim \sigma} \left[\sup_{D_y \in \text{Lip}(X)} f_{D_y}(y) \right] = \sup_{D \in \mathcal{D}(X \times Y)} \mathbb{E}_{y \sim \sigma} [f_{D(\cdot, y)}(y)]. \quad (21)$$

To prove (21), note first that the claim clearly holds when equality is replaced with “ \geq ” since $D(\cdot, y) \in \text{Lip}(X)$ for any $D \in \mathcal{D}(X \times Y)$. It remains to prove that strict inequality in (21) cannot hold. In the following, we use a proof by contradiction approach, so assume strict inequality holds:

$$\mathbb{E}_{y \sim \sigma} \left[\sup_{D_y \in \text{Lip}(X)} f_{D_y}(y) \right] > \sup_{D \in \mathcal{D}(X \times Y)} \mathbb{E}_{y \sim \sigma} [f_{D(\cdot, y)}(y)]. \quad (22)$$

From (22), there exists $\varepsilon > 0$ such that

$$\mathbb{E}_{y \sim \sigma} \left[\sup_{D_y \in \text{Lip}(X)} f_{D_y}(y) \right] - \varepsilon > \sup_{D \in \mathcal{D}(X \times Y)} \mathbb{E}_{y \sim \sigma} [f_{D(\cdot, y)}(y)]. \quad (23)$$

Next, for any $y \in Y$ and $\varepsilon > 0$, there exists $\widehat{D}_y \in \text{Lip}(X)$ such that

$$f_{\widehat{D}_y}(y) > \sup_{D_y \in \text{Lip}(X)} f_{D_y}(y) - \varepsilon \quad \text{holds for any } y \in Y. \quad (24)$$

Assume next that it is possible to choose \widehat{D}_y so that $(x, y) \mapsto \widehat{D}_y(x)$ is measurable on $X \times Y$. This implies that $(x, y) \mapsto \widehat{D}_y(x) \in \mathcal{D}(X \times Y)$ since $\widehat{D}_y \in \text{Lip}(X)$ for all $y \in Y$. Hence, the σ -expectation of $f_{\widehat{D}_y}(y)$ exists and (24) combined with the monotonicity of the expectation gives

$$\begin{aligned} \mathbb{E}_{y \sim \sigma} [f_{\widehat{D}_y}(y)] &> \mathbb{E}_{y \sim \sigma} \left[\sup_{D_y \in \text{Lip}(X)} f_{D_y}(y) - \varepsilon \right] \\ &= \mathbb{E}_{y \sim \sigma} \left[\sup_{D_y \in \text{Lip}(X)} f_{D_y}(y) \right] - \varepsilon. \end{aligned}$$

Insert the above into (23) gives

$$\mathbb{E}_{y \sim \sigma} [f_{\widehat{D}_y}(y)] > \sup_{D \in \mathcal{D}(X \times Y)} \mathbb{E}_{y \sim \sigma} [f_{D(\cdot, y)}(y)]. \quad (25)$$

Since $(x, y) \mapsto \widehat{D}_y(x) \in \mathcal{D}(X \times Y)$, the statement in (25) contradicts the definition of the supremum, i.e., (22) leads to a contradiction implying that (21) is true. This concludes the proof.

C.2 A novel discriminator for conditional WGAN

A generator trained using the formulation in (20) *as is* will typically learn to ignore the randomness from $\mathbf{z} \sim \eta$. This can be seen in figs. 6 and 7 that replicate the tests performed in figs. 3 and 5 but with a generator trained using (20) *as is*. Observe that the inter-sample variance is very low, e.g., the conditional mean image in fig. 6 is still very noisy as compared to corresponding images in fig. 3.

An explanation to this phenomena can be found in statistical learning theory. Note that, regardless of the number of supervised training data points (x_i, y_i) , the training data only provides a single X -sample x_i of the probability measure

is not this "corresponds" an overfitting?

Does it mean that MSE will be lower than in regression problem?

$\pi(x | y_i)$, which is the posterior at y_i . Since training data only provides a *single sample* from $\pi(x | y_i)$, training by (20) will result in a generator that only learns how to generate the corresponding single sample thereby generating the same sample repeatedly (mode collapse) [68].

The importance of addressing mode collapse is clearly illustrated in figs. 6 and 7. One approach to avoid mode collapse is to let the discriminator in (12) see multiple samples from $\pi(x | y_i)$, which leads to the idea of mini-batch discriminators [59, 39]. Such an approach is not possible in Bayes inversion since training data only provides access to a *single* model parameter x for each data y . In the following we describe a new *conditional mini-batch discriminator* that is better at avoiding mode collapse in the Bayesian inversion setting.

Conditional WGAN discriminator The idea is to let the discriminator distinguish between unordered pairs in X containing either the model parameter or random samples generated by the generative model. To formalize this, the generative model is trained using the following generalization of (2):

$$\inf_{\mathcal{G} \in \mathfrak{G}} \mathbb{E}_{y \sim \sigma} \left[\mathcal{W} \left(\mathcal{G}(y) \otimes \mathcal{G}(y), \frac{1}{2} (\pi(x | y) \otimes \mathcal{G}(y)) \oplus \frac{1}{2} (\mathcal{G}(y) \otimes \pi(x | y)) \right) \right] \quad (26)$$

where \oplus denotes usual summation of measures. Next, we show that one may train a generative model based on (26) instead of (2). The former lets the discriminator see more than a single sample from the posterior, so the resulting learned generator is much less likely to suffer from mode collapse, see fig. 7 for an empirically confirmation of this.

Claim 1. A generative model $\mathcal{G}: Y \rightarrow \mathcal{P}_X$ solves (26) iff it solves (2).

Proof. Let $y \in Y$ be fixed and consider the objective in (26):

$$\begin{aligned} & \mathcal{W} \left(\mathcal{G}(y) \otimes \mathcal{G}(y), \frac{1}{2} (\pi(x | y) \otimes \mathcal{G}(y)) \oplus \frac{1}{2} (\mathcal{G}(y) \otimes \pi(x | y)) \right) \\ &= \mathcal{W} \left(\frac{1}{2} (\mathcal{G}(y) \otimes \mathcal{G}(y), \pi(x | y) \oplus \mathcal{G}(y)) \otimes \frac{1}{2} (\pi(x | y) \oplus \mathcal{G}(y)) \right) \\ &\propto \mathcal{W} \left(\frac{1}{2} (\mathcal{G}(y) \otimes \mathcal{G}(y)), \frac{1}{2} (\pi(x | y) \otimes \pi(x | y)) \right). \end{aligned}$$

The last equality above follows from subtracting the measure $\frac{1}{2} (\mathcal{G}(y) \otimes \mathcal{G}(y))$ from both arguments in the Wasserstein metric and utilizing its translation invariance (which is easiest to see in the Kantorovich-Rubenstein characterization). Next,

$$\mathcal{W} \left(\frac{1}{2} (\mathcal{G}(y) \otimes \mathcal{G}(y)), \frac{1}{2} (\pi(x | y) \otimes \pi(x | y)) \right) \propto \mathcal{W}(\mathcal{G}(y), \pi(x | y)),$$

so a generative model solves (26) if and only if it solves (2). \square

Note that in the proof of claim 1, we implicitly assume the Wasserstein distance can be defined on any pair of positive Radon measures with equal mass. This is a trivial extension of the original definition of the Wasserstein distance, which assumes the domain is a pair of probability measures. It is worth noting that one can define “optimal transportation”-like distances between arbitrary positive Radon measures [17, 18].

To proceed, we need to rewrite the training in (26) so that it becomes more tractable, e.g., by removing the explicit appearance of the unknown posterior and probability measure for data. To do that, we yet again resort to the Kantorovich-Rubenstein duality (6). When applied to (26), it yields

$$\inf_{\mathcal{G} \in \mathfrak{G}} \mathbb{E}_{y \sim \sigma} \left[\sup_{D \in \mathcal{D}} \mathbb{E}_{\substack{(x_1, x_2) \sim \rho(y) \\ (v_1, v_2) \sim \mathcal{G}(y) \otimes \mathcal{G}(y)}} \left[D((x_1, x_2), y) - D((v_1, v_2), y) \right] \right]. \quad (27)$$

Here, $\rho(y) := \frac{1}{2}(\pi(x|y) \otimes \mathcal{G}(y)) \oplus \frac{1}{2}(\mathcal{G}(y) \otimes \pi(x|y))$ is a probability measure on $X \times X$ and \mathcal{D} are measurable maps $D: (X \times X) \times Y \rightarrow \mathbb{R}$ that are 1-Lipschitz w.r.t. its $(X \times X)$ -variable. Next, the same arguments used to rewrite (12) as (14) can also be used to rewrite (27) as

$$\inf_{\mathcal{G} \in \mathfrak{G}} \left\{ \sup_{D \in \mathcal{D}} \mathbb{E}_{(x,y) \sim \mu} \left[\frac{1}{2} \left(D((x, v_2), y) + D((v_1, x), y) \right) - D((v_1, v_2), y) \right] \right\}. \quad (28)$$

In contrast to (27), the formulation in (28) makes no reference to the posterior $\pi(x|y)$ nor the probability measure σ for data. Instead, it involves an expectation w.r.t. the joint law μ , which in a practical setting can be replaced by its empirical counterpart given from supervised training data in (15).

The final step is to introduce parameterizations for the generator and discriminator. The generator is parametrized as in (18), whereas the parametrized family $\mathcal{D} := \{D_\phi\}_{\phi \in \Phi}$ of discriminators are measurable mappings of the type $D_\phi: (X \times X) \times Y \rightarrow \mathbb{R}$ that are 1-Lipschitz in the $(X \times X)$ -variable. Inserting these parameterizations into (28) results in

$$(\theta^*, \phi^*) \in \arg \min_{\theta \in \Theta} \left\{ \sup_{\phi \in \Phi} \mathbb{E}_{\substack{(x,y) \sim \mu \\ z_1, z_2 \sim \eta}} \left[\frac{1}{2} \left(D_\phi((x, G_\theta(z_2, y)), y) + D_\phi((G_\theta(z_1, y), x), y) \right) - D_\phi((G_\theta(z_1, y), G_\theta(z_2, y)), y) \right] \right\}. \quad (29)$$

Note again that the unknown joint law μ in (29) is replaced by its empirical counterpart given from the training data in (15).

C.3 Deep Direct Estimation

The aim here is to show how an appropriately trained deep neural network can be used for approximating a wide range of non-randomized decision rules (estimators) associated, e.g., with uncertainty quantification. This differs from the posterior sampling approach (section 4.1 and appendix C.1) where such estimators are computed empirically by sampling from a trained WGAN generator.

The idea is to extend the approach in [5, 6] for learning estimators that minimizes Bayes risk so that it applies to a wider class of estimators. Our starting point is a well known proposition from probability theory that characterizes the minimizer of the mean squared error loss.

Proposition 2. *Assume that Y be a measurable space, W is a measurable Hilbert space, and y and w are Y - and W -valued random variables, respectively. Then, the conditional expectation $h^*(y) := \mathbb{E}[w | y = y]$ solves*

$$\min_{h: Y \rightarrow W} \mathbb{E} \left[\|h(y) - w\|_W^2 \right].$$

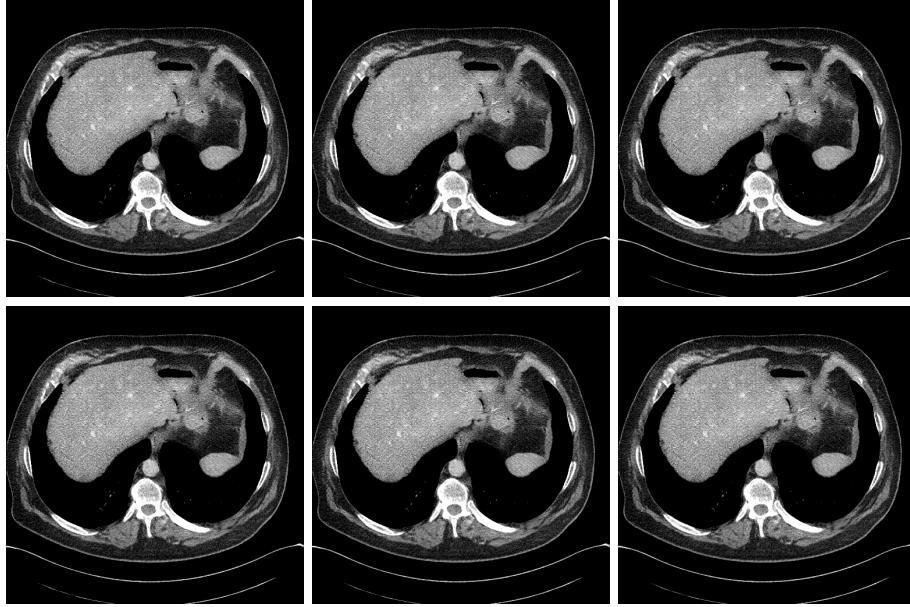


Fig. 6: Replication of fig. 5 without conditional WGAN discriminator shown using the same intensity window. Observe that there is practically no inter-sample variability due to mode collapse, confirming that the conditional WGAN discriminator is essential for posterior sampling.

The minimization above is taken over all W -valued measurable functions on Y .

Proof. Let $h: Y \rightarrow W$ be any measurable function so

$$\mathbb{E}[\|h(y) - w\|_W^2] = \mathbb{E}[\mathbb{E}[\|h(y) - w\|_W^2 | y]].$$

Next, W is a Hilbert space so we can expand the squared norm:

$$\begin{aligned} \|h(y) - w\|_W^2 &= \|h(y) - \mathbb{E}[w | y] + \mathbb{E}[w | y] - w\|_W^2 \\ &= \|h(y) - \mathbb{E}[w | y]\|_W^2 \\ &\quad + 2\langle h(y) - \mathbb{E}[w | y], \mathbb{E}[w | y] - w \rangle_W + \|w - \mathbb{E}[w | y]\|_W^2. \end{aligned}$$

By the law of total expectation and the linearity of the inner product, we get

$$\begin{aligned} \mathbb{E}[2\langle h(y) - \mathbb{E}[w | y], \mathbb{E}[w | y] - w \rangle_W | y] \\ &= 2\langle h(y) - \mathbb{E}[w | y], \mathbb{E}[w | y] - \mathbb{E}[w | y] \rangle_W \\ &= 2\langle h(y) - \mathbb{E}[w | y], 0 \rangle_W = 0 \end{aligned}$$

and $\|w - \mathbb{E}[w | y]\|_W^2$ is independent of $h(y)$. Combining all of this gives

$$\arg \min_{h: Y \rightarrow W} \mathbb{E}[\|h(y) - w\|_W^2] = \arg \min_{h: Y \rightarrow W} \mathbb{E}[\|h(y) - \mathbb{E}[w | y]\|_W^2]$$

where $h^*(y) = \mathbb{E}[w | y]$ is the solution to the right hand side. \square

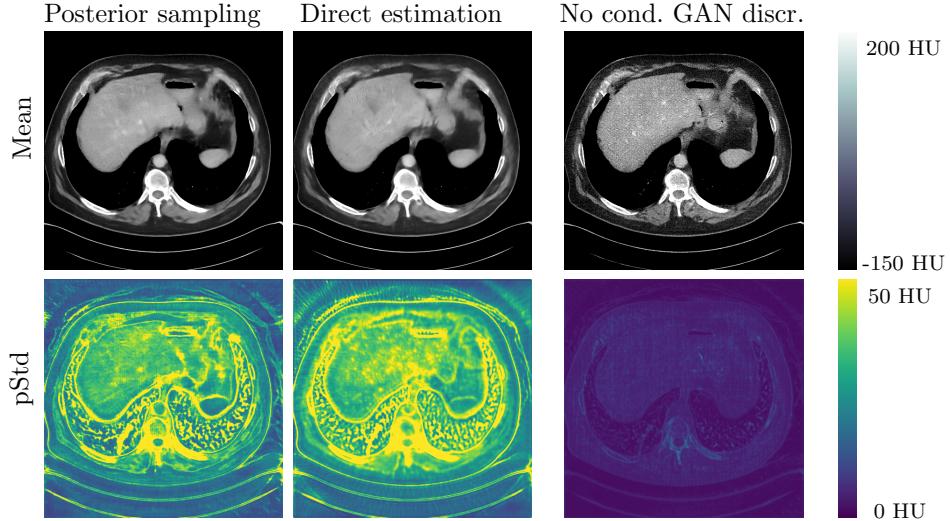


Fig. 7: Replication of fig. 3 also showing (right most column) the sample mean and sample point-wise standard deviation (pStd) when the conditional WGAN discriminator is not used. The standard deviation grossly underestimate due to mode collapse.

Proposition 2 implies in particular that minimizing Bayes risk with a loss given by the mean squared error amounts to computing the conditional mean. This result *does not* hold when the loss is the 1-norm, which would give the conditional *median* instead of the conditional mean. In a finite dimensional setting, proposition 2 holds also when the loss is any functional that is the Bregman distance of a convex functional [10].

In the context of Bayesian inversion, x and y are the X and Y -valued random variables generating the model parameter and data, respectively. Proposition 2 is then the starting point for studying the relation between the maximum a posteriori (MAP) and conditional mean estimates [14]. In our setting, if $h^*: Y \rightarrow X$ is the estimator that minimizes Bayes risk using squared loss, then proposition 2 (with $w := x$) implies that

$$h^* \in \arg \min_{h: Y \rightarrow X} \mathbb{E} \left[\|h(y) - w\|_W^2 \right] \implies h^*(y) = \mathbb{E}[x | y = y] \quad \text{for } y \in Y. \quad (30)$$

Since neural networks are universal function approximators, training a neural network using the mean squared error as loss yields an approximation of the conditional mean.

By selecting some other regression target w we can approximate estimators other than the conditional mean. As an example, let us consider the point-wise conditional variance which is defined as

$$\text{pVar}[x | y = y] := \mathbb{E} \left[(x - \mathbb{E}[x | y = y])^2 | y = y \right] \quad (31)$$

In the following, we show how the (point-wise) conditional variance can be estimated directly using a neural network trained against supervised data, similar

to how we estimate the conditional mean. The key step is to re-write the conditional variance as a minimizer of the expectation of some scalar objective w.r.t. the joint law of (x, y) .

Proposition 3. *Assume that Y, X are measurable spaces and that X is a Hilbert space. The point-wise variance is then characterized by*

$$\text{pVar}[x | y = y] \in \arg \min_{h: Y \rightarrow X} \mathbb{E} \left[\left\| h(y) - (x - \mathbb{E}[x | y = y])^2 \right\|_X^2 \right] \quad (32)$$

where the minimization is taken over all X -valued measurable functions on Y .

The proof follows by applying proposition 2 with $w := (x - \mathbb{E}[x | y = y])^2$, which yields

$$\arg \min_{h: Y \rightarrow X} \mathbb{E} \left[\left\| h(y) - (x - \mathbb{E}[x | y = y])^2 \right\|_X^2 \right] = \mathbb{E} \left[(x - \mathbb{E}[x | y = y])^2 | y = \cdot \right].$$

In practice we don't have direct access to samples from $(x - \mathbb{E}[x | y = y])^2$, so this cannot be applied as is since we cannot compute the expectation in (32). However, if there is access to supervised training data as in (15), then the conditional expectation in (32) can be approximated by a deep neural network trained according to (30), $\mathbb{E}[x | y = y] \approx \mathcal{T}_\theta^\dagger(y)$. From this training data one can then generate "new" training data of the form

$$\left((x_i - \mathcal{T}_\theta^\dagger(y_i))^2, y_i \right) \in X \times Y \quad \text{where } (x_i, y_i) \in X \times Y \text{ is from (15).}$$

This training data is random samples from a $(X \times Y)$ -valued random variable that approximately has required distribution.

Finally, the minimization in (32) can be restricted to X -valued measurable functions on Y that are parametrized by another deep neural network architecture $h_\phi: Y \rightarrow X$. Hence, the conditional point-wise variance can be estimated as $\text{pVar}[x | y = y] \approx h_{\phi^*}(y)$ where ϕ^* is obtained from solving the following training problems:

$$\begin{aligned} \theta^* &\in \arg \min_{\theta} \left\{ \mathbb{E}_{(x,y)} \left[\|x - \mathcal{T}_\theta^\dagger(y)\|_X^2 \right] \right\} \\ \phi^* &\in \arg \min_{\phi} \left\{ \mathbb{E}_{(x,y)} \left[\left\| h_\phi(y) - (x - \mathcal{T}_{\theta^*}^\dagger(y))^2 \right\|_X^2 \right] \right\}. \end{aligned}$$

Direct estimation is a sample free method that has several advantages against posterior sampling. First, they are much easier to train, generative adversarial networks (GANs) that are used for posterior sampling are known for being notoriously hard to train whereas learned iterative methods that underly direct estimation can be trained using standard approaches. Next, they are much faster. Evaluating a trained deep neural network for direct estimation requires roughly as much computational power as generating a single sample of the posterior in posterior sampling. Since posterior sampling requires several samples to get sufficient statistics, they will require an order of magnitude more time.

A downside with direct estimation is that a separate neural network has to be constructed and trained for each estimator. This is especially problematic

in cases where we need to answer patient-specific questions that are perhaps unknown during training. Another is that direct estimation as introduced here can only be used for estimators that can be re-written as a minimizer of the expectation of some scalar objective w.r.t. the joint law of (x, y) . It is well known that conditional distributions can be approximated by Edgeworth expansions that in turn contain such terms [53], so in principle any posterior can be approximated in this manner by a series of direct estimations. However, the computations quickly get complicated and the computational and training related advantages of direct estimation quickly diminishes.

Finally, results and corresponding proofs as stated in this section are not fully rigorous in the function space setting. As an example, proposition 3 would in such a setting involve the theory of higher moments of Banach space valued random variables [34], which quickly involves elaborate measure theory. On the other hand, the proofs are straightforward in finite dimensional spaces.

D Implementation Details

D.1 Training data

Training data is clinical 3D helical computed tomography (CT) scans from the Mayo Clinic Low Dose CT challenge [47]. The data was obtained using a Siemens SOMATOM Definition AS+ scanner and consists of ten abdomen CT scans of patients with predominantly liver and lung cancer obtained at normal dose. The scanner is a 64-slice cone beam helical CT that further enhances longitudinal resolution by a periodic motion of the focal spot in the z-direction (z-flying focal spot acquisition) [22]. The x-ray tube peak voltage (kVp) was 100–120 kV, depending on patient size, the exposure time was 500 ms and the tube current was 230–430 mA, again depending on patient size.

The normal dose reconstructions are obtained by applying a filtered back-projection (FBP)-type of reconstruction scheme, provided by the manufacturer of the scanner, on the full data. To obtain the low dose images, we first sub-sampled data and then added noise. The original data is acquired using a 3-PI acquisition geometry [13], meaning that the helical pitch is chosen to oversample each integration line by a factor of three. We sub-sampled the data by excluding the “upper” and “lower” pitch, which corresponds to data from 1-PI acquisition geometry. This results in a sub-sampling of 33%. Furthermore, we split each dataset into three independent datasets by using every third angle. This gives a further sub-sampling by 33%, for a total subsampling of ≈ 10 . In addition, we added Poisson noise to the data according to [47] until they corresponded to 2% normal dose scans, i.e. roughly 1 000 photons per pixel. While electron noise is significant at these dose levels, we chose not to model it.

Standard FBP was applied to the above ultra low dose data with a Hann filter with cutoff 0.4 and the filter frequency was chosen to maximize the peak signal to noise ratio (PSNR) of the ultra low dose reconstructions. The 2D slice size was set to 512×512 pixels with a reconstruction diameter of 370–440 mm (depending on patient size) and a slice-thickness of 3 mm. Note that the FBP reconstruction operator is formally not information conserving when using a cutoff (information is irreversibly lost), which technically invalidates the claim in section 5.1 that FBP may be used as a pre-processing step without any

information loss. However, we did not observe any adverse effects in letting y represent FBP reconstructions rather than CT data.

Finally, in order to (approximately) center the images, they were linearly scaled so that zero corresponds to 0 HU and -1 to $-1\,000$ HU. In total, supervised training data consisted of 6 498 pairs of semi-independent 2D images at normal and ultra low dose. To further augment the training data during training, we applied random flips (left-right), rotations ($\pm 10^\circ$), adding pixel-wise dequantization noise distributed according to $\mathcal{U}(0, 1)$ HU, and a random mean-value offset distributed according to $\mathcal{N}(0, 10)$ HU.

D.2 Neural networks

For simplicity, all networks are based on a similar convolutional neural network (CNN) architecture that consists of the following three building blocks:

- Averagepooling. Mapping an $2n \times 2n$ image to a $n \times n$ image by taking the average over 2×2 pixel blocks.
- Pixelshuffle (also “space to depth”) [61]. Mapping a $n \times n$ image with $4c$ channels to a $2n \times 2n$ image with c channels by spatially spreading the channels into a 2×2 block.
- Residual blocks [31]. A single residual block consists of applying batch normalization to the input, followed by a nonlinearity, convolution, batch normalization, nonlinearity and finally a convolution. This is added to a 1×1 convolution of the result of the first batch normalization. Such a block is shown in fig. 9b.

Furthermore, unless otherwise stated, the CNN uses 3×3 convolutions and leaky ReLU ($\alpha = 0.2$) non-linearities [45].

For the generator $G_\theta: Z \times Y \rightarrow X$, direct mean estimator $\mathcal{T}_\theta^\dagger: Y \rightarrow X$, and direct variance estimator $h_\phi: Y \rightarrow X$, we used an architecture similar to U-Net [58] combining down-sampling followed by a residual block until the image is 8×8 pixels. At this point we performed up-samplings combined with concatenating skip-connections until we reach the original 512×512 pixel resolution. The network architecture is illustrated in fig. 8. For $\mathcal{T}_\theta^\dagger$ and h_ϕ the input was simply the data y . Regarding the generator, we let the random noise z be white noise on $Z := X$, so $G_\theta: X \times X \rightarrow X$. For the generator and direct mean estimator we also added an additive skip-connection from y to result [35].

Finally, the discriminator D_ϕ is parametrized using a similar network architecture but stopped at the lowest resolution (8×8 pixels) and finished with two fully connected layers (fig. 9a).

Noise is the 512^2 image

Discriminator - half U-net + Fully connected

D.3 Training

First, all training procedures involved applying a small L_2 regularization (weight decay) with constant 10^{-4} to complement the expected loss. Furthermore, $\hat{\mu}$ will denote the empirical probability measure derived from the supervised training data (15) that has undergone data augmentation (appendix D.1).

Direct estimation Training the networks in the direct estimation approach (appendix C.3) amounts to solving

$$\begin{aligned}\theta^* &\in \arg \min_{\theta} \left\{ \mathbb{E}_{(x,y) \sim \hat{\mu}} \left[\|x - \mathcal{T}_\theta^\dagger(y)\|_X^2 \right] + 10^{-4} \|\theta\|^2 \right\} \quad \text{weight decay} \\ \phi^* &\in \arg \min_{\phi} \left\{ \mathbb{E}_{(x,y) \sim \hat{\mu}} \left[\left\| h_\phi(y) - (x - \mathcal{T}_{\theta^*}^\dagger(y))^2 \right\|_X^2 \right] + 10^{-4} \|\phi\|^2 \right\}.\end{aligned}$$

Posterior sampling The WGAN loss with the conditional WGAN discriminator (appendix C.2) is the objective in (29), i.e., it is given by

$$\begin{aligned}L_W(\theta, \phi) := \mathbb{E}_{\substack{(x,y) \sim \hat{\mu} \\ z_1, z_2 \sim \eta}} \left[\frac{1}{2} \left(D_\phi((x, G_\theta(z_1, y)), y) + D_\phi((G_\theta(z_1, y), x), y) \right) \right. \\ \left. - D_\phi((G_\theta(z_1, y), G_\theta(z_2, y)), y) \right]\end{aligned}$$

The set-up in (29) indicates that the discriminator should always be fully trained. Following best practice, instead of minimizing $(\theta, \phi) \mapsto L_W(\theta, \phi)$ jointly, we set-up an intertwined scheme where we take one step to minimize a generator loss $\theta \mapsto L_G(\theta)$ keeping ϕ fixed, then we take five steps to minimize a discriminator loss $\phi \mapsto L_D(\phi)$ keeping θ fixed. In the following, we explain how to construct these generator and discriminator losses.

For training the discriminator, note that $\phi \mapsto L_W(\theta, \phi)$ is invariant w.r.t. adding an arbitrary constant to the discriminator. This causes the training to become unstable since the discriminator can drift [39]. We levitate this by adding a small penalization

$$L_{\text{drift}}(\phi) := \mathbb{E}_{(x,y) \sim \hat{\mu}} [D_\phi(x, y)^2].$$

Next, as in [27], we enforce the 1-Lipschitz condition for the discriminator (see appendix A) by adding the following gradient penalty term:

$$L_{\text{grad}}(\theta, \phi) := \mathbb{E}_{\substack{(x,y) \sim \hat{\mu} \\ \varepsilon \sim \mathcal{U}(0,1) \\ z_1, z_2 \sim \eta}} \left[\left(\|\Gamma_{\theta, \phi}(x, y, z_1, z_2, \varepsilon)\|_{X^*} - 1 \right)^2 \right]$$

where $\Gamma_{\theta, \phi}: X \times Y \times Z \times Z \times [0, 1] \rightarrow X^*$ is given as

Gradient penalty
only along X variable,
as as stated

$$\begin{aligned}\Gamma_{\theta, \phi}(x, y, z_1, z_2, \varepsilon) & \quad \text{no penalty} \\ &:= \frac{1}{2} \left\{ \partial_1 D_\phi \left(\varepsilon(x, G_\theta(z_1, y)) + (1 - \varepsilon)(G_\theta(z_1, y), G_\theta(z_2, y)), y \right) \right. \\ & \quad \left. + \partial_1 D_\phi \left(\varepsilon(G_\theta(z_1, y), x) + (1 - \varepsilon)(G_\theta(z_1, y), G_\theta(z_2, y)), y \right) \right\}\end{aligned}$$

with $\partial_1 D_\phi$ denoting the first order partial (Banach space) derivative w.r.t. the $(X \times X)$ -variable of $D_\phi: (X \times X) \times Y \rightarrow \mathbb{R}$. Then, the loss $\phi \mapsto L_D(\phi)$ for training the discriminator (for fixed generator θ) becomes

$$L_D(\phi) := -L_W(\theta, \phi) + 10 L_{\text{grad}}(\theta, \phi) + 10^{-3} L_{\text{drift}}(\theta, \phi) + 10^{-4} \|\phi\|^2, \quad (33)$$

where the scalings 10 and 10^{-3} were chosen according to best practice [27, 39] and not hand-tuned by us.

The loss $\theta \mapsto L_G(\theta)$ for training the generator (for fixed discriminator ϕ) is

$$L_G(\theta) := L_W(\theta, \phi) + 10^{-4} \|\theta\|^2. \quad (34)$$

Optimization for training We used the same optimization method to train all networks (both for direct and posterior sampling), which was the ADAM optimizer [40] with $\beta_1 = 0.5$, $\beta_2 = 0.9$ and 50 000 training steps (≈ 8 epochs). For the batch normalization [33], we used decay 0.9 and $\varepsilon = 10^{-5}$. Moreover, we reduced the learning rate following Noisy Linear Cosine Decay [12] with default parameters, starting with a learning rate of $2 \cdot 10^{-4}$.

Despite our data-augmentation and regularization, we observed some overfitting during training, and expect that better results than ours could be obtained with more data.

E Handcrafted Priors

The samples were generated from Gibbs priors of the form $e^{-S(x)}$ where the regularization functional $S: X \rightarrow \mathbb{R}$ is chosen as indicated by the caption text for the images in the top row of fig. 1.

An interesting feature is that many of the samples from the priors shown in fig. 1 appear to be generated by a Gaussian random field prior. This may contradict the conventional wisdom that the choice of prior (regularizer) has a significant impact on the end result. However, a closer consideration shows that this behavior is to be expected from theory. It turns out that several priors, including the total variation (TV)-prior $S(x) := \|\nabla x\|_1$, converge weakly to a standard Gaussian free field as the discretization becomes finer as shown in [43, Theorem 5.3] for the TV-prior and in [42] for Besov space priors. The differences in the regularized solution provided by the MAP estimator are largely due to a small set of relatively unlikely images. In conclusion, using such priors in Bayesian inversion of large scale inverse problems has very little effect over, e.g., using a Gaussian random field prior.

References

- [1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mane, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viegas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. TensorFlow: Large-scale machine learning on heterogeneous distributed systems. *ArXiv*, cs.DC(1603.04467), 2016.
- [2] J. Adler, H. Kohr, and O. Öktem. Operator discretization library (ODL), January 2017. Software available from github.com/odlgroup/odl.
- [3] J. Adler and S. Lunz. Banach wasserstein GAN. In *Advances in Neural Information Processing Systems 32 (NIPS 2018)*, 2018. ArXiv version at <http://arxiv.org/abs/1806.06621>.

- [4] J. Adler, S. Lunz, O. Verdier, C.-B. Schönlieb, and O. Öktem. Task adapted reconstruction for inverse problems. *ArXiv*, cs.CV(1809.00948), 2018.
- [5] J. Adler and O. Öktem. Solving ill-posed inverse problems using iterative deep neural networks. *Inverse Problems*, 2017.
- [6] J. Adler and O. Öktem. Learned primal-dual reconstruction. *IEEE Transactions on Medical Imaging*, 2018.
- [7] J. Adler, A. Ringh, O. Öktem, and J. Karlsson. Learning to solve inverse problems using Wasserstein loss. *NIPS Workshop on Optimal Transport*, 2017.
- [8] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein GAN. *ArXiv*, stat.ML(1701.07875), 2017.
- [9] S. Arora, A. Risteski, and Y. Zhang. Do GANs learn the distribution? some theory and empirics. In *The Seventh International Conference on Machine Learning (ILCR 2018)*, 2018. Also available at ArXiv: <http://arxiv.org/abs/1706.08224>.
- [10] A. Banerjee, X. Guo, and H. Wang. On the optimality of conditional expectation as a Bregman predictor. *IEEE Transactions on Information Theory*, 51(7):2664–2669, 2005.
- [11] A. Barp, F.-X. Briol, A. D. Kennedy, and M. Girolami. Geometry and dynamics for Markov Chain Monte Carlo. *Annual Review of Statistics and Its Application*, 5:451–471, 2018. ArXiv version at <http://arxiv.org/abs/1705.02891>.
- [12] I. Bello, B. Zoph, V. Vasudevan, and Q. V. Le. Neural optimizer search with reinforcement learning. In D. Precup and Y. W. Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 459–468, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR.
- [13] C. Bontus and T. Köhler. Reconstruction algorithms for computed tomography. *Advances in Imaging and Electron Physics*, 151:1–63, 2009.
- [14] M. Burger and F. Lucka. Maximum a posteriori estimates in linear inverse problems with log-concave priors are proper Bayes estimators. *Inverse Problems*, 30:114004 (21pp), 2014.
- [15] D. Calvetti and E. Somersalo. Inverse problems: From regularization to Bayesian inference. *WIREs Computational Statistics*, 2017.
- [16] H. Chen, Y. Zhang, Y. Chen, J. Zhang, W. Zhang, H. Sun, Y. Lv, P. Liao, J. Zhou, and G. Wang. LEARN: Learned Experts’ Assessment-based Reconstruction Network for Sparse-data CT. *ArXiv*, physics.med-ph(1707.09636), 2017.
- [17] L. Chizat. *Unbalanced Optimal Transport: Models, Numerical Methods, Applications*. Phd thesis, Université Paris-Dauphine, 2017.

- [18] L. Chizat, G. Peyré, B. Schmitzer, and F.-X. Vialard. Unbalanced optimal transport: Geometry and Kantorovich formulation. *ArXiv*, math.OC(1508.05216), 2015. Accepted for publication in Journal of Functional Analysis.
- [19] M. Dashti and A.M. Stuart. The Bayesian approach to inverse problems. In R. Ghanem, D. Higdon, and H. Owhadi, editors, *Handbook of Uncertainty Quantification*, chapter 10. Springer-Verlag, New York, 2016.
- [20] Z. Deng, H. Zhang, X. Liang, L. Yang, S. Xu, J. Zhu, and E. P. Xing. Structured generative adversarial networks. *ArXiv*, cs.LG(1711.00889), 2017.
- [21] S. N. Evans and P. B. Stark. Inverse problems as statistics. *Inverse Problems*, 2002.
- [22] T.G. Flohr, K. Stierstorfer, S. Ulzheimer, H Bruder, A.N. Primak, and C. H. McCollough. Image reconstruction and image quality evaluation for a 64-slice CT scanner with z-flying focal spot. *Medical Physics*, 32(8):2536–2347, 2005.
- [23] M. Garnelo, D. Rosenbaum, C. J. Maddison, T. Ramalho, D. Saxton, M. Shanahan, Y. W. Teh, D. J. Rezende, and A. S. M. Eslami. Conditional neural processes. *ArXiv*, cs.LG(1807.01613), 2018.
- [24] M. Garnelo, J. Schwarz, D. Rosenbaum, F. Viola, D. J. Rezende, A. S. M. Eslami, and Y. W. Teh. Neural processes. *ArXiv*, cs.LG(1807.01622), 2018.
- [25] S. Ghosal and A. W. van der Vaart. *Fundamentals of Nonparametric Bayesian Inference*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2017.
- [26] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, 2014.
- [27] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville. Improved training of wasserstein GANs. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5767–5777. Curran Associates, Inc., 2017.
- [28] S. J. Hamilton and A. Hauptmann. Deep D-bar: Real time electrical impedance tomography imaging with deep neural networks. *IEEE Transactions on Medical Imaging*, pages 1–1, 2018.
- [29] K. Hammernik, T. Klatzer, E. Kobler, M. P. Recht, D. K. Sodickson, T. Pock, and F. Knoll. Learning a variational network for reconstruction of accelerated MRI data. *Magnetic Resonance in Medicine*, 79(6):3055–3071, 2017.
- [30] A. Hauptmann, F. Lucka, M. Betcke, N. Huynh, J. Adler, B. Cox, P. Beard, S. Ourselin, and S. Arridge. Model-based learning for accelerated, limited-view 3-D photoacoustic tomography. *IEEE Transactions on Medical Imaging*, 37(6):1382–1393, June 2018.

- [31] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, June 2016.
- [32] Z. Hu, Z. Yang, X. Liang, R. Salakhutdinov, and E. P. Xing. Toward controlled generation of text. *ArXiv*, cs.LG(1703.00955), 2017.
- [33] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *ArXiv*, cs.LG(1502.03167), 2015.
- [34] S. Janson and S. Kaijser. Higher moments of Banach space valued random variables. *Memoirs of the American Mathematical Society*, 238(1127):1–110, 2015.
- [35] K. H. Jin, M. T. McCann, E. Froustey, and M. Unser. Deep convolutional neural network for inverse problems in imaging. *IEEE Transactions on Image Processing*, 26(9):4509–4522, Sept 2017.
- [36] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In B. Leibe, J. Matas, N. Sebe, and Welling. M., editors, *European Conference on Computer Vision (ECCV 2016): 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II*, volume 9906 of *Lecture Notes in Computer Science*, pages 694–711. Springer-Verlag, 2016.
- [37] J. P. Kaipio and E. Somersalo. *Statistical and Computational Inverse Problems*, volume 160 of *Applied Mathematical Sciences*. Springer Verlag, 2005.
- [38] E. Kang, Min; J., and J. C. Ye. WaveNet: a deep convolutional neural network using directional wavelets for low-dose x-ray CT reconstruction. *ArXiv*, cs.CV(1610.09736), 2016.
- [39] T. Karras, T. Aila, S. Laine, and J. Lehtinen. Progressive growing of GANs for improved quality, stability, and variation. In *International Conference on Learning Representations*, 2018.
- [40] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *ArXiv*, cs.LG(1412.6980), 2014.
- [41] S. A. A. Kohl, B. Romera-Paredes, C. Meyer, J. De Fauw, J. R. Ledsam, K. H. Maier-Hein, S. M. A. Eslami, D. J. Rezende, and O. Ronneberger. A probabilistic U-Net for segmentation of ambiguous images. In *Advances in Neural Information Processing Systems (NIPS 2018)*, 2018. ArXiv version at <http://arxiv.org/abs/1806.05034>.
- [42] M. Lassas, E. Saksman, and S. Siltanen. Discretization-invariant Bayesian inversion and Besov space priors. *Inverse Problems & Imaging*, 3(1):87–122, 2009.
- [43] M. Lassas and S. Siltanen. Can one use total variation prior for edge-preserving Bayesian inversion? *Inverse Problems*, 20(5):1537–1563, 2004.

- [44] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi. Photo-realistic single image super-resolution using a generative adversarial network. *ArXiv*, cs.CV(1609.04802), 2016.
- [45] A. L. Maas, A. Y. Hannun, and A. Y. Ng. Rectifier nonlinearities improve neural network acoustic models. *Proceedings of the International Conference on Machine Learning (ICML)*, 2013.
- [46] M. Mardani, E. Gong, J. Y. Cheng, S. Vasanawala, G. Zaharchuk, M. Alley, N. Thakur, S. Han, W. Dally, J. M. Pauly, and L. Xing. Deep generative adversarial networks for compressed sensing automates MRI. *ArXiv*, cs.CV(1706.00051), 2017.
- [47] C. McCollough. TFG-207A-04: Overview of the low dose CT grand challenge. *Medical Physics*, 43(6):3759–3760, 2016.
- [48] M. Mirza and S. Osindero. Conditional generative adversarial nets. *ArXiv*, cs.LG(1411.1784), 2014.
- [49] A. Nguyen, J. Clune, Y. Bengio, Y. Dosovitskiy, and J. Yosinski. Plug & play generative networks: Conditional iterative generation of images in latent space. *ArXiv*, cs.CV(1612.00005), 2016.
- [50] R. Nickl. On Bayesian inference for some statistical inverse problems with partial differential equations. *Bernoulli News*, 24(2):5–9, 2017.
- [51] N. Parmar, A. Vaswani, J. Uszkoreit, L. Kaiser, N. Shazeer, and A. Ku. Image transformer. *ArXiv*, cs.CV(1802.05751), 2018.
- [52] N. Parmar, A. Vaswani, J. Uszkoreit, L. Kaiser, N. Shazeer, A. Ku, and D. Tran. Image transformer. *ArXiv*, cs.CV(1802.05751), 2018.
- [53] B. V. Pedersen. Approximating conditional distributions by the mixed Edgeworth-saddlepoint expansion. *Biometrika*, 66(3):597–604, 1979.
- [54] M. Pereyra. Maximum-a-posteriori estimation with Bayesian confidence regions. *SIAM Journal of Imaging Sciences*, 10(1):285–302, 2017.
- [55] Z. Purisha, C. Jidling, N. Wahlström, S. Särkkä, and T. B. Schön. Probabilistic approach to limited-data computed tomography reconstruction. *ArXiv*, cs.CV(1809.03779), 2018.
- [56] G.-L. Qi. Loss-sensitive generative adversarial networks on Lipschitz densities. *ArXiv*, cs.CV(1701.06264), 2017.
- [57] A. Repetti, M. Pereyra, and Y. Wiaux. Scalable Bayesian uncertainty quantification in imaging inverse problems via convex optimization. *ArXiv*, stat.ME(1803.00889), 2018.
- [58] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, volume 9351 of *Lecture Notes in Computer Science*, pages 234–241. Springer, 2015. ArXiv version at <http://arxiv.org/abs/1505.04597>.

- [59] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen. Improved techniques for training GANs. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 2234–2242. Curran Associates, Inc., 2016.
- [60] S. Shalev-Shwartz and S. Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014.
- [61] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang. Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network. *ArXiv*, cs.CV(1609.05158), 2016.
- [62] W. van Aarle, W. J. Palenstijn, J. Cant, E. Janssens, F. Bleichrodt, A. Dabravolski, J. De Beenhouwer, J. K. Batenburg, and J. Sijbers. Fast and flexible x-ray tomography using the astra toolbox. *Opt. Express*, 2016.
- [63] C. Villani. *Optimal Transport: Old and New*. Number 338 in Grundlehren der mathematischen Wissenschaften. Springer Verlag, 2009.
- [64] J. M. Wolterink, T. Leiner, M. A. Viergever, and I. Išgum. Generative adversarial networks for noise reduction in low-dose CT. *IEEE Transactions on Medical Imaging*, 2017.
- [65] G. Yang, S. Yu, H. Dong, G. Slabaugh, P. L. Dragotti, X. Ye, F. Liu, S. Arridge, J. Keegan, Y. Guo, and D. Firmin. DAGAN: Deep De-Aliasing Generative Adversarial Networks for fast compressed sensing MRI reconstruction. *IEEE Transactions on Medical Imaging*, 2018.
- [66] Y. Yang, J. Sun, H. Li, and Z. Xu. Deep ADMM-Net for compressive sensing MRI. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29 (NIPS 2016)*, pages 10–18. Curran Associates, Inc., 2016.
- [67] X. Yi and P. Babyn. Sharpness-aware low dose CT denoising using conditional generative adversarial network. *ArXiv*, cs.CV(1708.06453), 2017.
- [68] M. Zaheer, S. Kottur, S. Ravanbakhsh, B. Poczos, R. Salakhutdinov, and A. Smola. Deep sets. *ArXiv*, cs.LG(1703.06114), 2017.

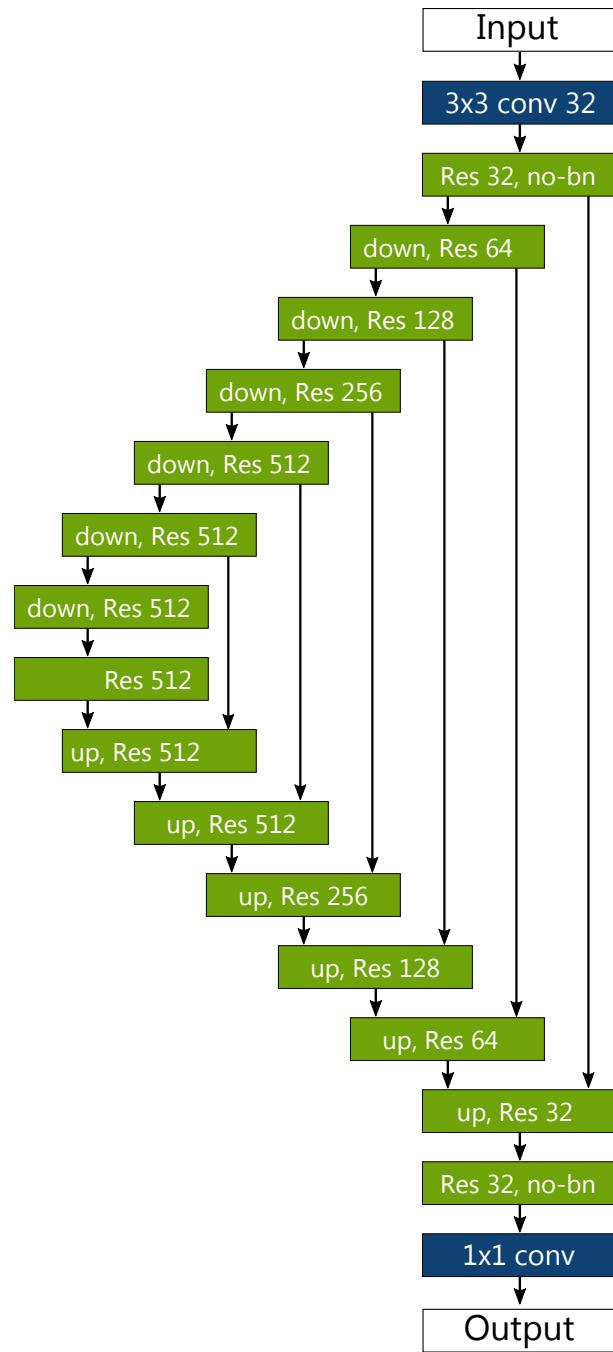
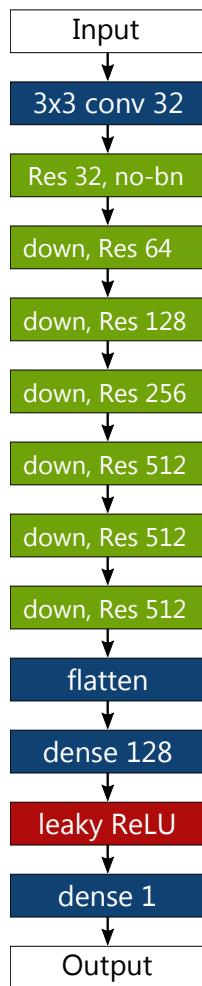
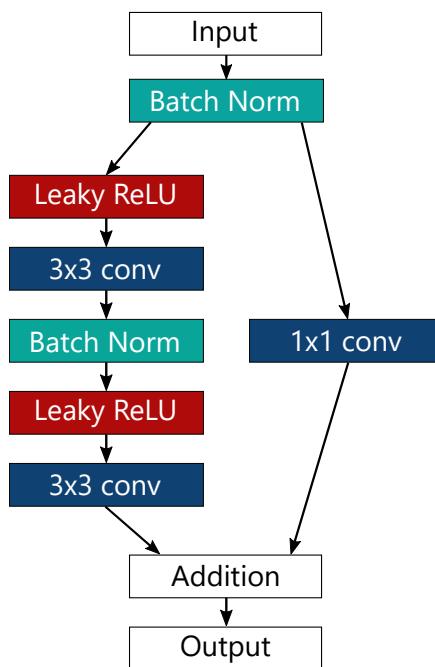


Fig. 8: Residual U-Net network architecture. “down” indicates that a down-sampling is done before the resblock. “up” indicates that an upsampling and a concatenation is done before the residual block. “no-bn” indicates that batch normalization was not used in that residual block. The input has resolution 512^2 while the smallest images have size 8^2 .



(a) Discriminator network.



(b) Residual unit used in the network. All convolutions have the same number of output channels.