QUANTIFYING UNCERTAINTY WITH GAN-BASED PRIORS

Anonymous authors
Paper under double-blind review

ABSTRACT

Bayesian inference is used extensively to quantify the uncertainty in an inferred field given the measurement of a related field when the two are linked by a mathematical model. Despite its many applications, Bayesian inference faces challenges when inferring fields that have discrete representations of large dimension, and/or have prior distributions that are difficult to characterize mathematically. In this work we demonstrate how the approximate distribution learned by a generative adversarial network (GAN) may be used as a prior in a Bayesian update to address both these challenges. We demonstrate the efficacy of this approach by inferring and quantifying uncertainty in inference problems arising in computer vision and physics-based applications. In both instances we highlight the role of computing uncertainty in providing a measure of confidence in the solution, and in designing successive measurements to improve this confidence.

1 INTRODUCTION

Quantifying uncertainty in an inference problem amounts to making a prediction and quantifying the confidence in that prediction. In the context of an image recovery problem, this may be understood as follows. A typical computer vision algorithm uses a noisy version of an image and prior knowledge to produce the recovered image which can be interpreted as the "best guess" of the original image. Quantifying uncertainty in this context involves generating an estimate of the level of confidence in the best guess, in addition to the guess itself.

Bayesian inference provides a principled approach for quantifying uncertainty. As shown in the following section, it treats the inferred vector as a multivariate stochastic vector and leads to an expression for its distribution. This expression can be used to estimate the most likely solution (the maximum a-posteriori estimate, or the MAP), the mean, the variance, or any other population parameter of interest. Thus providing a recipe for thoroughly quantifying the uncertainty in an inference problem. For the image recovery problems considered in this paper, Bayesian inference not only provides the best guess of the true image, but also a means to estimate measures of uncertainty such as the pixel-wise variance.

The knowledge of uncertainty in a prediction can directly influence the downstream action that depends on the inference. Consider an image recovery problem where two distinct inputs lead to similar recovered images: those of a traffic sign with a high speed limit. However, for the first input the predicted variance is small, while for the second input it is large. Further, the set of likely images in the second set also includes images of a Stop Sign. Then the appropriate action for the two inputs, determined after solving the inference problem and quantifying uncertainty, is very different. For the first input, the appropriate action is one of continued motion, whereas for the second input it is to slow down. Similar examples can be drawn from other areas as well, like medical imaging, high frequency trading, and autonomous systems where critical decisions are made based on the output of AI system (Gal (2016); Begoli et al. (2019)).

The knowledge of uncertainty can also be useful in determining the optimal location of a sensor. Consider an image recovery problem, where the goal is to infer the signal, and associated uncertainty, using limited amount of measurement data. In this problem a user can leverage information about the spatial distribution of uncertainty to choose the location with maximum uncertainty as next measurement location. This task falls within the fields of active learning and/or design of ex-

periments (DeGroot et al. (1962); Houlsby et al. (2011)) and is particularly useful in applications like satellite imaging, where each measurement requires significant time and/or resources.

In Figure 1, we demonstrate how the proposed GAN-based Bayesian inference algorithm can be useful in both scenarios explained above: (i) in quantifying uncertainty - "*pixel-wise variance*" - our quantitative measure of uncertainty in the inferred field, and (ii) in determining the optimal sensor placement location in an iterative fashion. We return to these applications in greater detail in Section 4.

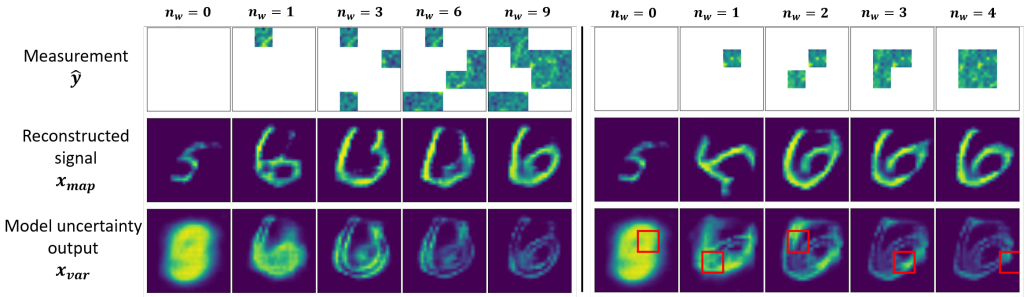


Figure 1: Estimate of the MAP (2nd row) and pixel-wise variance (3rd row) from the limited view of a noisy image (1st row) using the proposed method. The first five columns correspond to a strategy where the next window is randomly selected, while the other columns correspond to a strategy where the next window is selected in the region with the maximum estimated variance. For equivalent accuracy, the variance-driven selection strategy uses fewer sampling windows (4 versus 9). In both cases variance reduces with increasing measurement.

1.1 Bayesian inference

Bayesian inference is a well-established technique for quantifying uncertainties in inference problems (Kaipio & Somersalo (2006); Dashti & Stuart (2016); Polpo et al. (2018)). It has found applications in diverse fields such as geophysics (Gouveia & Scales (1997); Malinverno (2002)), climate modeling (Jackson et al. (2004)), chemical kinetics (Najm et al. (2009)), heat conduction (Wang & Zabaras (2004)), and the detection and diagnosis of disease (Siltanen et al. (2003); Kolehmainen et al. (2006)). The two critical ingredients of a Bayesian inference problem are - an informative prior representing the prior belief about the parameters to be inferred and an efficient method for sampling from the posterior distribution. In this manuscript we describe how deep generative adversarial networks (GANs) can be effectively used in these roles.

Consider the setting where we wish to infer a vector of parameters $\boldsymbol{x} \in \mathbb{R}^N$ from the measurement of a related vector $\boldsymbol{y} \in \mathbb{R}^P$, where the two are related through a forward model $\boldsymbol{y} = \boldsymbol{f}(\boldsymbol{x})$. A noisy measurement of $\boldsymbol{y}$ is denoted by $\hat{\boldsymbol{y}} = \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{\eta}$, where the vector $\boldsymbol{\eta} \in \mathbb{R}^P$ represents noise. While the forward map $\boldsymbol{f}$ is typically well-posed, its inverse is not, and hence to infer $\boldsymbol{x}$ from the measurement $\hat{\boldsymbol{y}}$ requires techniques that account for this ill-posedness. Classical techniques based on regularization tackle this ill-posedness by using additional information about the sought solution field explicitly or implicitly (Tarantola (2005)). Bayesian inference offers a different approach to this problem by modeling the unknown solution as well as the measurements as random variables. This framework addresses the ill-posedness of the inverse problem, and allows for the characterization of the uncertainty in the inferred solution.

The notion of a prior distribution plays a key role in Bayesian inference. Through multiple observations of the field $\boldsymbol{x}$, denoted by the set $\mathcal{S} = \{\boldsymbol{x}^{(1)}, \cdots, \boldsymbol{x}^{(S)}\}$, we have some prior knowledge of $\boldsymbol{x}$ that can be utilized when inferring $\boldsymbol{x}$ from $\hat{\boldsymbol{y}}$. This is used to build, or intuit, a prior distribution for $\boldsymbol{x}$, denoted by $p_X^{\text{prior}}(\boldsymbol{x})$. Some typical examples include Gaussian process prior with specified co-variance kernels, Gaussian Markov random fields (Fahrmeir & Lang (2001)), Gaussian priors defined through differential operators (Stuart (2010)), and hierarchical Gaussian priors (Marzouk & Najm (2009); Calvetti & Somersalo (2008)). These priors promote smoothness and/or structure in the inferred solution and importantly, can be expressed explicitly in an analytical form.

Another key component of Bayesian inference is a distribution that represents the likelihood of $\boldsymbol{y}$ given an instance of $\boldsymbol{x}$, denoted by $p^{\mathrm{l}}(\boldsymbol{y}|\boldsymbol{x})$. This is often determined by the distribution of the error in the model, denoted by p_η, which captures both model and measurement errors. Given this, and an additive model for noise, the posterior distribution of $\boldsymbol{x}$, determined using Bayes' theorem after accounting for the observation $\hat{\boldsymbol{y}}$ is given by,

$$p_X^{\mathrm{post}}(\boldsymbol{x}|\boldsymbol{y}) = \frac{1}{\mathbb{Z}} p^{\mathrm{l}}(\boldsymbol{y}|\boldsymbol{x}) p_X^{\mathrm{prior}}(\boldsymbol{x}) = \frac{1}{\mathbb{Z}} p_\eta(\hat{\boldsymbol{y}} - \boldsymbol{f}(\boldsymbol{x})) p_X^{\mathrm{prior}}(\boldsymbol{x}). \tag{1}$$

Here, $\mathbb{Z}$ is the prior-predictive distribution of $\boldsymbol{y}$ and ensures that the posterior integrates to one.

The posterior distribution characterizes the uncertainty in $\boldsymbol{x}$; however for vectors of large dimension characterizing this distribution explicitly is a challenging task. Consequently the expression above is used to perform tasks that are more manageable. These include determining estimates such as the maximum a-posteriori estimate (MAP), expanding the posterior distribution in terms of other distributions that are simpler to work with (Bui-Thanh et al. (2012)), or using techniques like Markov Chain Monte-Carlo (MCMC) to generate samples that are "close" to the samples generated by the true posterior distribution (Han & Carlin (2001); Parno & Marzouk (2018)).

Despite its numerous applications in solving inverse problems, Bayesian inference faces significant challenges. These include defining a reliable and informative prior distribution for $\boldsymbol{x}$ when the set $\mathcal{S} = \{\boldsymbol{x}^{(1)}, \cdots, \boldsymbol{x}^{(S)}\}$ is difficult to characterize analytically, and efficiently sampling from the posterior distribution when the dimension of $\boldsymbol{x}$ is large; a typical situation in many practical science and engineering applications.

1.2 RELATED WORK

The main idea developed in this paper tackles the above mentioned challenges by training a generative adversarial network (GAN) using the sample set $\mathcal{S}$, and then using the distribution learned by the GAN as the prior distribution in Bayesian inference.

The use of sample-based priors for solving an inverse problem has a rich history (Vauhkonen et al. (1997); Calvetti & Somersalo (2005)). As does the idea of reducing the dimension of the parameter space by mapping it to a lower-dimensional space (Marzouk & Najm (2009); Lieberman et al. (2010)). However, the use of learning-based deep generative models like GANs in these tasks is novel.

Recently, several authors have considered the use of learning-based methods for solving inverse problems arising in different domains. These include the use of deep convolutional neural networks (CNNs), recurrent neural networks (RNNs) or variational autoencoders (VAEs) to solve physics-driven inverse problems (Adler & Öktem (2017); Patel et al. (2019); Pesah et al. (2018); McCarthy et al. (2017)), use of VAEs or GANs to solve inverse problems in computer vision (Yeh et al. (2016); Ham et al.; Chang et al.; Kupyn et al. (2018); Ledig et al.; Zhu et al. (2017)). There is also a growing body of work dedicated to using GANs to learn regularizers in solving inverse problems (Lunz et al. (2018) and in compressed sensing (Bora et al. (2017; 2018); Kabkab et al. (2018); Wu et al. (2019); Shah & Hegde (2018)). However, these approaches differs from ours in that they solve the inverse problem as an optimization problem and do not rely on Bayesian inference; as a result, they add regularization in an ad-hoc manner and do not attempt to quantify the uncertainty in the inferred field.

More recently, the approach described in Adler & Öktem (2018) utilizes GANs in a Bayesian setting; however the GAN is trained to approximate the posterior distribution (and not the prior, as in our case), and training is done in a supervised fashion. That is, paired samples of the measurement $\hat{\boldsymbol{y}}$ and the corresponding true solution $\boldsymbol{x}$ are required. We note that deep learning based Bayesian networks, where the network weights are stochastic parameters that are determined using Bayesian inference, are another means of quantifying uncertainty (MacKay (1992); Kingma & Welling (2013); Gal & Ghahramani (2016)), and have recently been applied to semantic image-segmentation and super-resolution (Kendall & Gal (2017); Kendall et al. (2019); Kohl et al. (2018); Hu et al. (2019); Tanno et al. (2019)) . However, these method also rely on supervised learning, whereas in contrast, our approach is un-supervised, and requires only samples of the true solution $\boldsymbol{x}$ to train the GAN prior.

1.3 OUR CONTRIBUTION

The main contribution of this paper can be summarized as follows:

1. A novel method for performing Bayesian inference involving complex priors and high dimensional posterior. In this method we utilize the distribution learned by a GAN as a surrogate for the prior distribution and reformulate the inference problem in the low-dimensional latent space of the GAN. Furthermore, we provide a theoretical analysis of the weak convergence of the posterior density learned by the proposed method to the true posterior density.
2. A novel unsupervised image denoising and inpainting algorithms with quantitative measures of uncertainty through pixel-wise variance.
3. Application of the proposed method to physics-based inference problems.
4. Demonstration of the utility of quantifying uncertainty to facilitate active learning.

2 PROBLEM FORMULATION

Let $\mathcal{S}$ denote the set of instances of vector $\boldsymbol{x}$ sampled from the true distribution, $p_X^{\text{true}}(\boldsymbol{x})$. Further, let $\boldsymbol{z} \sim p_Z(\boldsymbol{z})$ characterize the latent vector space and $\boldsymbol{g}(\boldsymbol{z})$ be the generator of a GAN trained using $\mathcal{S}$. Then according to Goodfellow et al. (2014), with infinite capacity and sufficient data, the generator learns the true distribution. That is,

$$p_X^{\text{gen}}(\boldsymbol{x}) = p_X^{\text{true}}(\boldsymbol{x}). \tag{2}$$

The distribution $p_X^{\text{gen}}(\boldsymbol{x})$ is defined as

$$\boldsymbol{x} \sim p_X^{\text{gen}}(\boldsymbol{x}) \Rightarrow \boldsymbol{x} = \boldsymbol{g}(\boldsymbol{z}), \boldsymbol{z} \sim p_Z(\boldsymbol{z}). \tag{3}$$

Here p_Z is the multivariate distribution of the latent vector whose components are iid and typically conform to a Gaussian or a uniform distribution. The equation above implies that the GAN generates samples of $\boldsymbol{x}$ by sampling $\boldsymbol{z}$ from p_Z and then passing these through the generator.

Now consider a measurement $\hat{\boldsymbol{y}}$ from which we would like to infer the posterior distribution of $\boldsymbol{x}$. For this we use (1) and set the prior distribution equal to the true distribution, that is $p_X^{\text{prior}} = p_X^{\text{true}}$. Then from (2) this is the same as $p_X^{\text{prior}} = p_X^{\text{gen}}$. Therefore,

$$p_X^{\text{post}}(\boldsymbol{x}|\boldsymbol{y}) \quad = \quad \frac{1}{\mathbb{Z}} p_\eta(\hat{\boldsymbol{y}} - \boldsymbol{f}(\boldsymbol{x})) p_X^{\text{gen}}(\boldsymbol{x}). \tag{4}$$

Now for any $l(\boldsymbol{x})$, we have

$$\begin{aligned}
\mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_X^{\text{post}}}[l(\boldsymbol{x})] &= \frac{1}{\mathbb{Z}} \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_X^{\text{gen}}}[l(\boldsymbol{x}) p_\eta(\hat{\boldsymbol{y}} - \boldsymbol{f}(\boldsymbol{x}))], && \text{From (4)} \\
&= \frac{1}{\mathbb{Z}} \mathop{\mathbb{E}}_{\boldsymbol{z}\sim p_Z}[l(\boldsymbol{g}(\boldsymbol{z})) p_\eta(\hat{\boldsymbol{y}} - \boldsymbol{f}(\boldsymbol{g}(\boldsymbol{z})))], && \text{From (3)} \\
&= \mathop{\mathbb{E}}_{\boldsymbol{z}\sim p_Z^{\text{post}}}[l(\boldsymbol{g}(\boldsymbol{z}))], && (5)
\end{aligned}$$

where $\mathbb{E}$ is the expectation operator, and

$$p_Z^{\text{post}}(\boldsymbol{z}|\boldsymbol{y}) \equiv \frac{1}{\mathbb{Z}} p_\eta(\hat{\boldsymbol{y}} - \boldsymbol{f}(\boldsymbol{g}(\boldsymbol{z}))) p_Z(\boldsymbol{z}). \tag{6}$$

The distribution p_Z^{post} is the analog of p_X^{post} in the latent vector space. The measurement $\hat{\boldsymbol{y}}$ updates the prior distribution for $\boldsymbol{x}$ to the posterior distribution. Similarly, it updates the prior distribution for $\boldsymbol{z}$, p_Z, to the posterior distribution, p_Z^{post}, defined above.

Equation (5) implies that sampling from the posterior distribution of $\boldsymbol{x}$ is equivalent to sampling from the posterior distribution for $\boldsymbol{z}$ and passing the sample through the generator $\boldsymbol{g}$. That is,

$$\boldsymbol{x} \sim p_X^{\text{post}}(\boldsymbol{x}|\boldsymbol{y}) \Rightarrow \boldsymbol{x} = \boldsymbol{g}(\boldsymbol{z}), \boldsymbol{z} \sim p_Z^{\text{post}}(\boldsymbol{z}|\boldsymbol{y}). \tag{7}$$

Since the dimension of $\boldsymbol{z}$ is typically smaller than that of $\boldsymbol{x}$, this represents an efficient approach to sampling from the posterior of $\boldsymbol{x}$.

The left hand side of (5) is an expression for a population parameter of the posterior, defined by $\overline{l(\boldsymbol{x})} \equiv \mathbb{E}_{\boldsymbol{x}\sim p_X^{\text{post}}}[l(\boldsymbol{x})]$. The right hand sides of the last two lines of this equation describe how this parameter may be evaluated by sampling $\boldsymbol{z}$ (instead of $\boldsymbol{x}$) from either p_Z or p_Z^{post}.

The equality in (5) holds for a GAN with an infinite number of weights in the generator and the discriminator. In Appendix A of this manuscript, we consider the case of a Wasserstein GAN with a finite number of weights and prove the weak convergence of the posterior density obtained by using a GAN as a prior to the true posterior density as the number of weights is increased.

2.1 Sampling from the posterior distribution

We consider a scenario where we wish to infer and characterize the uncertainty in the vector of parameters $\boldsymbol{x}$ from a noisy measurement of $\boldsymbol{y}$, denoted by $\hat{\boldsymbol{y}}$, where $\boldsymbol{f}$ is a known map that connects $\boldsymbol{x}$ and $\boldsymbol{y}$. And we have several prior realizations of plausible $\boldsymbol{x}$, contained in the set $\mathcal{S}$. For this problem we propose the following algorithm that accounts for the prior information in $\mathcal{S}$ and the "new" measurement $\hat{\boldsymbol{y}}$ through a Bayesian update:

1. Train a GAN with a generator $\boldsymbol{g}(\boldsymbol{z})$ on $\mathcal{S}$.
2. Sample $\boldsymbol{x}$ from $p_X^{\text{post}}(\boldsymbol{x}|\boldsymbol{y})$ given in (7).

With sufficient capacity in the GAN and with sufficient training, the posterior obtained using this algorithm will converge to the true posterior (see eq. (5) above and Appendix A). Since GANs can be used to represent complex distributions efficiently, this algorithm provides a means of including complex priors that are solely defined by samples within a Bayesian update. Further, it provides an efficient approach to sampling from the posterior since the dimension of $\boldsymbol{z}$ is typically much smaller (10^1 - 10^2) than that of $\boldsymbol{x}$ (10^4 - 10^7). Finally, in contrast to other methods that attempt to quantify uncertainty in image recovery tasks, this algorithm falls within the class of unsupervised learning algorithms, and does not required paired data for training. We now describe two approaches for estimating population parameters of the posterior that make use of this observation.

Monte-Carlo (MC) approximation The first approach is based on a Monte-Carlo approximation of a population parameter of the posterior distribution. This integral, which is defined in the second line of (5), may be approximated as,

$$\overline{l(\boldsymbol{x})} \equiv \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_X^{\text{post}}}[l(\boldsymbol{x})] \approx \frac{\sum_{n=1}^{N_{\text{samp}}} l(\boldsymbol{g}(\boldsymbol{z}))p_\eta(\hat{\boldsymbol{y}} - \boldsymbol{f}(\boldsymbol{g}(\boldsymbol{z})))}{\sum_{n=1}^{N_{\text{samp}}} p_\eta(\hat{\boldsymbol{y}} - \boldsymbol{f}(\boldsymbol{g}(\boldsymbol{z})))}, \qquad \boldsymbol{z} \sim p_Z(\boldsymbol{z}). \tag{8}$$

In the equation above, the numerator is obtained from a MC approximation of the integral in (5), and the denominator is obtained from a MC approximation of the scaling parameter $\mathbb{Z}$. Sampling within this approach is rather simple since in a typical GAN, the z_is belong to a simple distribution like a Gaussian or a uniform distribution. In many applications we anticipate that the likelihood will tend to concentrate the distribution of latent vector $\boldsymbol{z}$ to a small region within Ω_z. Thus the MC sampling described above may be inefficient.

Markov-Chain Monte-Carlo (MCMC) approximation A more efficient approach is to generate an MCMC approximation $p_Z^{\text{mcmc}}(\boldsymbol{z}|\boldsymbol{y}) \approx p_Z^{\text{post}}(\boldsymbol{z}|\boldsymbol{y})$ using the definition in (6), and thereafter sample $\boldsymbol{z}$ from this distribution. Then from the third line of (5), any desired population parameter may be approximated as

$$\overline{l(\boldsymbol{x})} \equiv \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_X^{\text{post}}}[l(\boldsymbol{x})] \approx \frac{1}{N_{\text{samp}}} \sum_{n=1}^{N_{\text{samp}}} l(\boldsymbol{g}(\boldsymbol{z})), \qquad \boldsymbol{z} \sim p_Z^{\text{mcmc}}(\boldsymbol{z}|\boldsymbol{y}). \tag{9}$$

We note that for all the numerical experiments in this paper we have used MCMC because of its better sample efficiency.

Summary We have described two methods for probing the posterior distribution when the prior is defined by a GAN. These include an MC (8) and an MCMC estimate (9) of a given population parameter and a MAP estimate that is applicable to additive Gaussian noise with a Gaussian prior for the latent vector (see Section B in the Appendix). In the following section we apply these algorithms to inverse problems drawn from physics-based and computer vision applications.

3 EXPERIMENTS

We evaluate our method empirically on practical probabilistic field inference tasks in the domain of computer vision and physical science. In computer vision, we consider image denoising and inpainting tasks and evaluate our algorithm on MNIST and CelebA datasets. Here we attempt to infer the true image and quantify the uncertainty in this inference. We also demonstrate the utility of quantifying uncertainty by using it in an active learning setup. In the physical domain, we consider the problem of recovering the initial temperature distribution from a measurement at later time, and use this problem to demonstrate the statistics inferred using our method converge to their "true" values.

In all cases we use a Wasserstein GAN-GP (Gulrajani et al. (2017)) to learn the prior density (architecture described in the Appendix D). We also ensure that the target images are not chosen from the set used to train the GAN. We sample from the posterior using Hamiltonian Monte Carlo (Brooks et al. (2012)) and implement it using Tensorflow-probability (Dillon et al. (2017)) library. We use initial step size of 1.0 for HMC and adapt it following (Andrieu & Thoms (2008)) based on the target acceptance probability. We use 64k samples with burn-in period of 0.5. We select these parameters to ensure convergence of chains. Using the HMC sampler we compute the MAP, which is our "best guess" of the true image, the pixel-wise mean, and the pixel-wise variance, which is our quantitative and spatially-varying estimate of uncertainty.

3.1 IMAGE RECOVERY USING THE MNIST & CELEBA DATABASES

We first consider the MNIST database of hand-written digits and use 55k images to train the GAN. We use a latent vector of dimension 100 with Gaussian distribution. For the image denoising task, we add Gaussian noise with zero mean and specified variance (σ_y) to the test image and use it as measurement to recover the distribution of likely images using the MCMC approach. For this problem the forward operator is the identity map, and the likelihood distribution is Gaussian. In Figure 2, we have plotted the noisy input image, the MAP estimate, and the pixel-wise mean and variance. We observe that for low and medium noise levels ($\sigma_y = 0.1, 1$), we are able to recover the original image with good accuracy, the pixel-wise variance is small overall, and is largest around the boundary of the recovered digit; this represents the variability in the different realizations of the recovered digit within the GAN prior. For the highest noise level ($\sigma_y = 10$), however, the image recovered by the MAP is incorrect in 2/3 cases, and would be misleading if viewed by itself. However, when viewed in conjunction with the estimated variance, which is large, it is clear that the confidence in the inference is small and the inferred image ought not be trusted for downstream tasks. The dependence of the average per-pixel variance in the recovered image on the variance of noise in the measured image is shown in Figure 4 (a) with 95% confidence interval. As expected, the per-pixel variance increases with increasing noise.

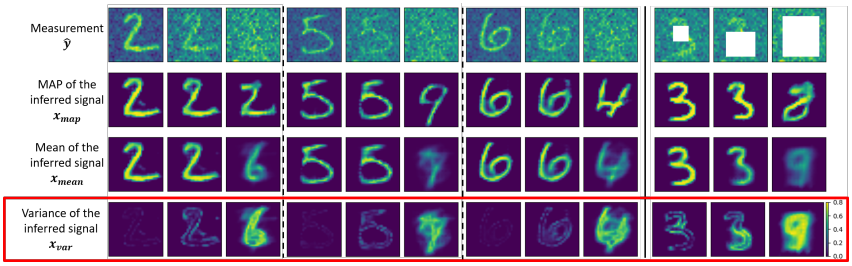


Figure 2: Estimate of the MAP, mean and pixel-wise variance from a noisy image using the proposed method. In the first three panels $\sigma_y = 0.1, 1, \&10$, when moving from left to right. In the fourth panel $\sigma_y = 1$, and the size of the occluded region is increased.

In the right-most panel of Figure 2, we show image inpainting + denoising results for an image of digit 3 with $\sigma_y = 1$. Here the forward map is the indicator function set to zero on the occluded pixels. We note that for the small and intermediate occluded regions, the MAP solution is close to the true solution. However, when most of the image is occluded, the MAP is incorrect. Once again, the variance image, which is small for the low and medium occluded regions, and large for the large

occlusion, is a reliable indicator of the confidence in the recovered MAP image. Furthermore, the variance is peaked within the occluded region demonstrating lower confidence in the image reconstructed in this region. This is likely to be useful in applications like autonomous cars and medical imaging, where partial measurements are be common and the spatial distribution of uncertainty can inform downstream tasks. More image denoising and inpainting examples are provided in Appendix C.

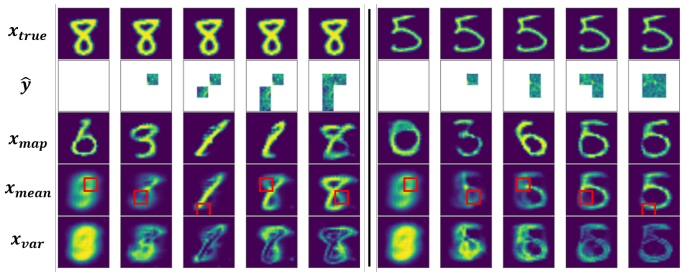


Figure 3: Estimate of the MAP, mean and variance from the limited view of a noisy image (2nd row) using the proposed method for the digits 8 & 5 (left and right panels). The window to be revealed at a given iteration (shown in red box) is selected using a variance-driven strategy.

In Figure 3, we demonstrate how uncertainty may be used in active learning/design of experiment, where the goal is to determine the optimal location for a measurement. We begin with an input where the entire image is occluded and in every subsequent step, we allow for a small 7×7 pixel window to be revealed. We select the window with the largest average pixel-wise variance. As the iterations progress, the MAP estimate converges to the true digit, and the variance decreases. In about 4 iterations we arrive at a very good guess for the digit. The performance of this approach is quantified in Figure 4(b), where we have plotted reconstruction error versus the number windows for this strategy, and a strategy where the subsequent window is selected randomly. The variance-driven strategy consistently performs better. We note that we are not aware of any other methods for computing uncertainty in recovered images that have been applied to drive an active learning task in image inpainting. While methods based on dropout (Kendall & Gal (2017); Kendall et al. (2019)) or variational inference (Kohl et al. (2018)) could be extended to accomplish this, this has not been done thus far.

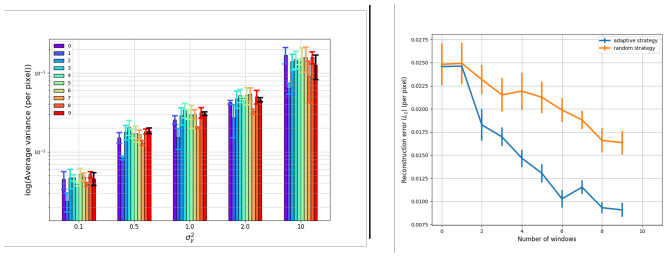


Figure 4: (a) Average variance per pixel in a reconstructed image as a function of variance in noise for 10 digits along with 95% confidence interval. (5 different digits from test set were used at each noise level to compute CI) – should we include this here? (b) Average reconstruction error as a function of number of windows for a variance-driven (adaptive learning) and a random sampling strategy.

Results for the variance-based window selection strategy applied to the CelebA dataset are shown in Figure 5. We observe that the algorithm produces realistic images at each iteration; however, the initial variance is large indicating large uncertainty. As more windows are sampled using the variance driven active learning strategy, the variance reduces and by the 7th iteration a good approximation of the true image is obtained, even though only a small, noisy portion is revealed. Additional results along with implementation details for this dataset is discussed in Appendix C.2.

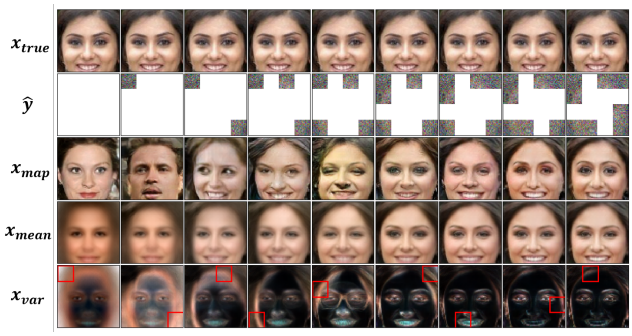


Figure 5: Estimate of the MAP, mean and variance from the limited view of a noisy image (2nd row) of a true image (1st row) using the variance-driven adaptive learning strategy.

3.2 A PHYSICS-DRIVEN INFERENCE PROBLEM

We consider the problem of inverse heat conduction, where the goal is to infer the initial temperature distribution (at $t = 0$) in a domain given a noisy measurement of temperature at later time $t = 1$ and the thermal conductivity of the material. The forward map is the solution of the time-dependent heat conduction problem with uniform conductivity, $\kappa = 0.64$ and domain length $L = 2\pi$ with Dirichlet boundary conditions. This operator maps the initial temperature. The discrete version of this operator is obtained by discretizing the time-dependent linear heat conduction equation using central difference scheme in space and backward difference scheme in time. Much like a blurring kernel, the forward operator smooths the initial temperature distribution, and the extent of smoothing increases with the product of conductivity and time.

We consider a family of initial temperatures where the background is zero, and the temperature is non-zero on a rectangular sub-domain, where it varies linearly from 2 units on the left edge to 4 units on the right edge. This distribution is parameterized by four parameters ($\{\xi_i\}_{i=1}^4$) the lower left and upper right coordinates of the rectangular region. A sample set $\mathcal{S}$ is created by sampling each parameter from a uniform distribution ($\{\xi_i\}_{i=1}^4 \sim Unif(0.25, 0.75) \times L$) and is used to train the GAN prior. The posterior distribution is sampled using the HMC sampler.

In the top two rows of Figure 6, we have plotted the true initial condition, the noise-free temperature at $t = 1$, and the noisy temperature measurement ($\sigma_y = 1$) used as input in the GAN-based prior approach. The corresponding MAP, mean and pixel-wise variance estimated by the MCMC approximation is shown next. We observe that the MAP is very close to the true initial temperature distribution and the variance is concentrated along the edges of the rectangle where the uncertainty is the largest. In the following columns we have plotted the MAP estimate obtained assuming L_2 and H^1 Gaussian priors, which are much less accurate.

For this problem the "true" posterior can be reduced to the 4-dimensional space of parameters, and sampled by generating initial conditions corresponding to the values of these parameters. A simple MC approximation can be performed to compute statistics - the mean and the variance for the true posterior (last two columns of Figure 6). By comparing these with the mean and the point-wise variance (columns 5 & 6) for the GAN-based prior, we note that GAN-based posterior has converged to the true posterior.

In the bottom rows of Figure 6, we plot similar results for initial conditions and GAN-based priors generated from the MNIST database with $t = 0.2$. Since the "true" distribution for this set is not known the true mean and variance are not plotted.

4 CONCLUSIONS

The ability to quantify the uncertainty in the prediction to an inference problem is useful in developing confidence in that prediction, and in designing strategies to improve the confidence. In this paper we have described how this may be accomplished when solving a Bayesian inference problem by using GANs as priors. Since GANs can be used to learn complex distributions of a wide variety of

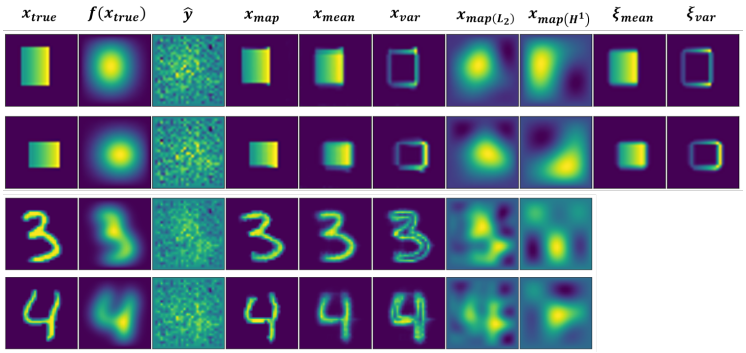


Figure 6: From left to right: (1) true initial temperature, (2) temperature at $t = 1$, (3) noisy version temperature used as measurement, (4), (5), (6) MAP, mean and pixel-wise variance estimates using GAN priors, (7) and (8) MAP estimates using L_2 and H^1 Gaussian priors, (9) & (10) true MAP and variance obtained by sampling over the true parameter space.

fields from their samples, this approach can be applied to a range of problems in computer vision and physics-driven inference. It derives its efficiency by mapping the posterior distribution to the latent space, whose dimension is often much smaller than that of the inferred field. We have applied this approach to image recovery tasks and demonstrated how the knowledge of uncertainty in the prediction can be used to assess confidence in a prediction, and via active learning to design a strategy to improve the confidence. We have also applied this approach to a physics-based problem, where we have verified its accuracy and robustness. While for the results reported in this manuscript we do assume knowledge of forward map, we note that the proposed algorithm can easily be extended to regime where the forward map (likelihood) is also unknown by incorporating likelihood-free inference methods like ABC or meta-learning approaches. We leave this as an interesting direction for future research.

REFERENCES

Jonas Adler and Ozan Öktem. Solving ill-posed inverse problems using iterative deep neural networks. *Inverse Problems*, 33(12):124007, dec 2017. ISSN 0266-5611. doi: 10.1088/1361-6420/aa9581. URL http://stacks.iop.org/0266-5611/33/i=12/a=124007?key=crossref.65c4fa88a47e07d4789aa10592f2090c.

Jonas Adler and Ozan Öktem. Deep bayesian inversion. *arXiv preprint arXiv:1811.05910*, 2018.

Christophe Andrieu and Johannes Thoms. A tutorial on adaptive MCMC. *Statistics and Computing*, 18(4):343–373, dec 2008. ISSN 09603174. doi: 10.1007/s11222-008-9110-y.

Edmon Begoli, Tanmoy Bhattacharya, and Dimitri Kusnezov. The need for uncertainty quantification in machine-assisted medical decision making. *Nature Machine Intelligence*, 1(1):20, 2019.

Ashish Bora, Ajil Jalal, Eric Price, and Alexandros G Dimakis. Compressed sensing using generative models. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 537–546. JMLR. org, 2017.

Ashish Bora, Eric Price, and Alexandros G Dimakis. Ambientgan: Generative models from lossy measurements. *ICLR*, 2:5, 2018.

Steve Brooks, Andrew Gelman, Galin Jones, Xiao-Li Meng, and Radford M Neal. MCMC using Hamiltonian dynamics. Technical report, 2012. URL https://arxiv.org/pdf/1206.1901.pdf.

Tan Bui-Thanh, Carsten Burstedde, Omar Ghattas, James Martin, Georg Stadler, and Lucas C. Wilcox. Extreme-scale UQ for Bayesian inverse problems governed by PDEs. In *2012 International Conference for High Performance Computing, Networking, Storage and Analysis*, pp. 1–11. IEEE, nov 2012. ISBN 978-1-4673-0805-2. doi: 10.1109/SC.2012.56. URL http://ieeexplore.ieee.org/document/6468442/.

Daniela Calvetti and Erkki Somersalo. Priorconditioners for linear systems. *Inverse Problems*, 21(4):1397–1418, aug 2005. ISSN 0266-5611. doi: 10.1088/0266-5611/21/4/014. URL http://stacks.iop.org/0266-5611/21/i=4/a=014?key=crossref.ab419ffb66111e3db21bf3d9fd3836f7.

Daniela Calvetti and Erkki Somersalo. Hypermodels in the Bayesian imaging framework. *Inverse Problems*, 24(3):034013, jun 2008. ISSN 0266-5611. doi: 10.1088/0266-5611/24/3/034013. URL http://stacks.iop.org/0266-5611/24/i=3/a=034013?key=crossref.10e6728aeff596e7626030876ce5e2a2.

J H Rick Chang, Chun-Liang Li, Barnab Barnabás, P Oczos, B V K Vijaya Kumar, and Aswin C Sankaranarayanan. One Network to Solve Them All-Solving Linear Inverse Problems using Deep Projection Models. Technical report. URL https://arxiv.org/pdf/1703.09912.pdf.

Masoumeh Dashti and Andrew M Stuart. The bayesian approach to inverse problems. *Handbook of Uncertainty Quantification*, pp. 1–118, 2016.

Morris H DeGroot et al. Uncertainty, information, and sequential experiments. *The Annals of Mathematical Statistics*, 33(2):404–419, 1962.

Joshua V. Dillon, Ian Langmore, Dustin Tran, Eugene Brevdo, Srinivas Vasudevan, Dave Moore, Brian Patton, Alex Alemi, Matt Hoffman, and Rif A. Saurous. TensorFlow Distributions. nov 2017. URL http://arxiv.org/abs/1711.10604.

Ludwig Fahrmeir and Stefan Lang. Bayesian inference for generalized additive mixed models based on Markov random field priors. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 50(2):201–220, jan 2001. ISSN 0035-9254. doi: 10.1111/1467-9876.00229. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/1467-9876.00229.

Yarin Gal. *Uncertainty in deep learning*. PhD thesis, PhD thesis, University of Cambridge, 2016.

Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pp. 1050–1059, 2016.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pp. 2672–2680, 2014.

Wences P Gouveia and John A Scales. Resolution of seismic waveform inversion: Bayes versus Occam. *Inverse Problems*, 13(2):323–349, apr 1997. ISSN 0266-5611. doi: 10.1088/0266-5611/13/2/009. URL http://stacks.iop.org/0266-5611/13/i=2/a=009?key=crossref.c3e937d46f2de4adfe9aa2de3c226f3e.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In *Advances in neural information processing systems*, pp. 5767–5777, 2017.

Cusuh Ham, Amit Raj, Vincent Cartillier, and Irfan Essa. Variational Image Inpainting. Technical report.

Cong Han and Bradley P Carlin. Markov chain monte carlo methods for computing bayes factors: A comparative review. *Journal of the American Statistical Association*, 96(455):1122–1132, 2001.

Neil Houlsby, Ferenc Huszár, Zoubin Ghahramani, and Máté Lengyel. Bayesian active learning for classification and preference learning. *arXiv preprint arXiv:1112.5745*, 2011.

Shi Hu, Daniel Worrall, Stefan Knegt, Bas Veeling, Henkjan Huisman, and Max Welling. Supervised Uncertainty Quantification for Segmentation with Multiple Annotations. jul 2019. URL http://arxiv.org/abs/1907.01949.

Charles Jackson, Mrinal K. Sen, Paul L. Stoffa, Charles Jackson, Mrinal K. Sen, and Paul L. Stoffa. An Efficient Stochastic Bayesian Approach to Optimal Parameter and Uncertainty Estimation for Climate Model Predictions. *Journal of Climate*, 17(14): 2828–2841, jul 2004. ISSN 0894-8755. doi: 10.1175/1520-0442(2004)017⟨2828:AESBAT⟩2.0.CO;2. URL http://journals.ametsoc.org/doi/abs/10.1175/1520-0442{%}282004{%}29017{%}3C2828{%}3AAESBAT{%}3E2.0.CO{%}3B2.

Maya Kabkab, Pouya Samangouei, and Rama Chellappa. Task-aware compressed sensing with generative adversarial networks. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

Jari Kaipio and Erkki Somersalo. *Statistical and computational inverse problems*, volume 160. Springer Science & Business Media, 2006.

Alex Kendall and Yarin Gal. What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision? mar 2017. URL http://arxiv.org/abs/1703.04977.

Alex Kendall, Vijay Badrinarayanan, and Roberto Cipolla. Bayesian SegNet: Model Uncertainty in Deep Convolutional Encoder-Decoder Architectures for Scene Understanding. British Machine Vision Association and Society for Pattern Recognition, may 2019. doi: 10.5244/c.31.57.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Simon A. A. Kohl, Bernardino Romera-Paredes, Clemens Meyer, Jeffrey De Fauw, Joseph R. Ledsam, Klaus H. Maier-Hein, S. M. Ali Eslami, Danilo Jimenez Rezende, and Olaf Ronneberger. A Probabilistic U-Net for Segmentation of Ambiguous Images. jun 2018. URL http://arxiv.org/abs/1806.05034.

V. Kolehmainen, A. Vanne, S. Siltanen, S. Jarvenpaa, J.P. Kaipio, M. Lassas, and M. Kalke. Parallelized Bayesian inversion for three-dimensional dental X-ray imaging. *IEEE Transactions on Medical Imaging*, 25(2):218–228, feb 2006. ISSN 0278-0062. doi: 10.1109/TMI.2005.862662. URL http://ieeexplore.ieee.org/document/1583768/.

Orest Kupyn, Volodymyr Budzan, Mykola Mykhailych, Dmytro Mishkin, and Jií Matas. DeblurGAN: Blind Motion Deblurring Using Conditional Adversarial Networks, 2018. URL http://openaccess.thecvf.com/content{_}cvpr{_}2018/html/Kupyn{_}DeblurGAN{_}Blind{_}Motion{_}CVPR{_}2018{_}paper.html.

Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe Shi Twitter. Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network. Technical report. URL http://openaccess.thecvf.com/content{_}cvpr{_}2017/papers/Ledig{_}Photo-Realistic{_}Single{_}Image{_}CVPR{_}2017{_}paper.pdf.

Chad Lieberman, Karen Willcox, and Omar Ghattas. Parameter and State Model Reduction for Large-Scale Statistical Inverse Problems. *SIAM Journal on Scientific Computing*, 32(5):2523–2542, jan 2010. ISSN 1064-8275. doi: 10.1137/090775622. URL http://epubs.siam.org/doi/10.1137/090775622.

Sebastian Lunz, Ozan Öktem, and Carola-Bibiane Schönlieb. Adversarial regularizers in inverse problems. In *Advances in Neural Information Processing Systems*, pp. 8507–8516, 2018.

David JC MacKay. A practical bayesian framework for backpropagation networks. *Neural computation*, 4(3):448–472, 1992.

A. Malinverno. Parsimonious Bayesian Markov chain Monte Carlo inversion in a nonlinear geophysical problem. *Geophysical Journal International*, 151(3):675–688, dec 2002. ISSN 0956-540X. doi: 10.1046/j.1365-246X.2002.01847.x. URL https://academic.oup.com/gji/article-lookup/doi/10.1046/j.1365-246X.2002.01847.x.

Youssef M. Marzouk and Habib N. Najm. Dimensionality reduction and polynomial chaos acceleration of Bayesian inference in inverse problems. *Journal of Computational Physics*, 228 (6):1862–1902, apr 2009. ISSN 0021-9991. doi: 10.1016/J.JCP.2008.11.024. URL https://www.sciencedirect.com/science/article/pii/S0021999108006062.

Adam McCarthy, Blanca Rodriguez, and Ana Minchole. Variational Inference over Non-differentiable Cardiac Simulators using Bayesian Optimization. dec 2017. URL http://arxiv.org/abs/1712.03353.

H. N. Najm, B. J. Debusschere, Y. M. Marzouk, S. Widmer, and O. P. Le MaÃ®tre. Uncertainty quantification in chemical systems. *International Journal for Numerical Methods in Engineering*, 80(6â7):789–814, nov 2009. ISSN 00295981. doi: 10.1002/nme.2551. URL http://doi.wiley.com/10.1002/nme.2551.

Matthew D Parno and Youssef M Marzouk. Transport map accelerated markov chain monte carlo. *SIAM/ASA Journal on Uncertainty Quantification*, 6(2):645–682, 2018.

Dhruv Patel, Raghav Tibrewala, Adriana Vega, Li Dong, Nicholas Hugenberg, and Assad A Oberai. Circumventing the solution of inverse problems in mechanics through deep learning: Application to elasticity imaging. *Computer Methods in Applied Mechanics and Engineering*, 353:448–466, 2019.

Arthur Pesah, Antoine Wehenkel, and Gilles Louppe. Recurrent machines for likelihood-free inference. nov 2018. URL http://arxiv.org/abs/1811.12932.

Adriano Polpo, Julio Stern, Francisco Louzada, Rafael Izbicki, and Hellinton Takada (eds.). *Bayesian Inference and Maximum Entropy Methods in Science and Engineering*, volume 239 of *Springer Proceedings in Mathematics & Statistics*. Springer International Publishing, Cham, 2018. ISBN 978-3-319-91142-7. doi: 10.1007/978-3-319-91143-4. URL http://link.springer.com/10.1007/978-3-319-91143-4.

Viraj Shah and Chinmay Hegde. Solving linear inverse problems using gan priors: An algorithm with provable guarantees. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4609–4613. IEEE, 2018.

S Siltanen, V Kolehmainen, S J rvenp, J P Kaipio, P Koistinen, M Lassas, J Pirttil, and E Somersalo. Statistical inversion for medical x-ray tomography with few radiographs: I. General theory. *Physics in Medicine and Biology*, 48(10):1437–1463, may 2003. ISSN 0031-9155. doi: 10.1088/0031-9155/48/10/314. URL http://stacks.iop.org/0031-9155/48/i=10/a=314?key=crossref.5fce2d21d49cf69f7c7b946fb1945c85.

A M Stuart. Inverse problems: A Bayesian perspective. *Acta Numerica*, 19:451–559, 2010. doi: 10.1017/S0962492910000061. URL http://journals.cambridge.org/abstract{_}S0962492910000061.

Ryutaro Tanno, Daniel Worrall, Enrico Kaden, Aurobrata Ghosh, Francesco Grussu, Alberto Bizzi, Stamatios N. Sotiropoulos, Antonio Criminisi, and Daniel C. Alexander. Uncertainty Quantification in Deep Learning for Safer Neuroimage Enhancement. jul 2019. URL http://arxiv.org/abs/1907.13418.

Albert Tarantola. *Inverse problem theory and methods for model parameter estimation*, volume 89. siam, 2005.

M Vauhkonen, J P Kaipio, E Somersalo, and P A Karjalainen. Electrical impedance tomography with basis constraints. *Inverse Problems*, 13(2):523–530, apr 1997. ISSN 0266-5611. doi: 10.1088/0266-5611/13/2/020. URL http://stacks.iop.org/0266-5611/13/i=2/a=020?key=crossref.46559bf45aab26a8302acc14e8db4c89.

Jingbo Wang and Nicholas Zabaras. Hierarchical bayesian models for inverse problems in heat conduction. *Inverse Problems*, 21(1):183–206, dec 2004. doi: 10.1088/0266-5611/21/1/012. URL https://doi.org/10.1088%2F0266-5611%2F21%2F1%2F012.

Yan Wu, Mihaela Rosca, and Timothy Lillicrap. Deep compressed sensing. *arXiv preprint arXiv:1905.06723*, 2019.

Raymond A. Yeh, Chen Chen, Teck-Yian Lim, Alexander G. Schwing, Mark Hasegawa-Johnson, and Minh N. Do. Semantic image inpainting with deep generative models. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6882–6890, 2016.

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, 2017.

A WEAK CONVERGENCE OF THE POSTERIOR DENSITY

In this section we prove the weak convergence of the posterior density obtained by using a GAN as a prior to the true posterior density as the number of weights in the generator and discriminator are increased.

Let the generator of the GAN be given by $\boldsymbol{g}(\boldsymbol{z};\boldsymbol{\theta})$, where $\boldsymbol{z} \in \mathbb{R}^M$ is the latent vector, and $\boldsymbol{\theta} \in \mathbb{R}^{N_\theta}$ be the vector of weights. The vector $\boldsymbol{z}$ is selected from the distribution $p_Z(\boldsymbol{z})$. Note that $\boldsymbol{g} : \mathbb{R}^M \to \mathbb{R}^N$.

Let the discriminator of the GAN be given by $d(\boldsymbol{x};\boldsymbol{\phi})$, where $\boldsymbol{x} \in \mathbb{R}^N$, and $\boldsymbol{\phi} \in \mathbb{R}^{N_\phi}$ be the vector of weights. Note that $d : \mathbb{R}^N \to \mathbb{R}(0,1)$.

The GAN is trained using a set of samples of $\boldsymbol{x}$, drawn from $p_X^{\text{prior}}(\boldsymbol{x})$. For the loss function, consider

$$L(\boldsymbol{\theta},\boldsymbol{\phi}) = \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_X^{\text{prior}}}[\rho(1-d(\boldsymbol{x};\boldsymbol{\phi}))] + \mathop{\mathbb{E}}_{\boldsymbol{z}\sim p_Z}[\rho(d(\boldsymbol{g}(\boldsymbol{z};\boldsymbol{\theta});\boldsymbol{\phi}))]. \tag{10}$$

Here ρ is a monotone real-valued function which defines the GAN family being analyzed. For example, for the Wasserstein GAN, $\rho(\xi) = \xi$.

The optimal values of the weights are given by

$$\boldsymbol{\theta}^*, \boldsymbol{\phi}^* = \mathop{\text{argmax}}_{\boldsymbol{\theta}}(\mathop{\text{argmin}}_{\boldsymbol{\phi}}(L(\boldsymbol{\theta},\boldsymbol{\phi}))). \tag{11}$$

A.1 STATIONARITY CONDITIONS

The necessary conditions for these optimal values are

$$\frac{\partial L(\boldsymbol{\theta}^*,\boldsymbol{\phi}^*)}{\partial \boldsymbol{\phi}} = \mathbf{0} \tag{12}$$

$$\frac{\partial L(\boldsymbol{\theta}^*,\boldsymbol{\phi}^*)}{\partial \boldsymbol{\theta}} = \mathbf{0}. \tag{13}$$

Using the definition of the loss function (10) in (12), we have

$$\mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_X^{\text{prior}}}[\rho'(1-d(\boldsymbol{x};\boldsymbol{\phi}^*))\frac{\partial d}{\partial \boldsymbol{\phi}}(\boldsymbol{x};\boldsymbol{\phi}^*)] = \mathop{\mathbb{E}}_{\boldsymbol{z}\sim p_Z}[\rho'(d(\boldsymbol{g}(\boldsymbol{z};\boldsymbol{\theta}^*);\boldsymbol{\phi}^*))\frac{\partial d}{\partial \boldsymbol{\phi}}(\boldsymbol{g}(\boldsymbol{z};\boldsymbol{\theta}^*);\boldsymbol{\phi}^*)]. \tag{14}$$

Similarly, using (10) in (13), we have

$$\mathop{\mathbb{E}}_{\boldsymbol{z}\sim p_Z}[\rho'(d(\boldsymbol{g}(\boldsymbol{z};\boldsymbol{\theta}^*);\boldsymbol{\phi}^*))\frac{\partial d}{\partial \boldsymbol{x}}(\boldsymbol{g}(\boldsymbol{z};\boldsymbol{\theta}^*);\boldsymbol{\phi}^*)\cdot\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{\theta}}(\boldsymbol{z};\boldsymbol{\theta}^*)] = \mathbf{0}. \tag{15}$$

A.2 WASSERSTEIN GAN

For the Wasserstein GAN, $\rho(\xi) = \xi$ and $\rho'(\xi) = 1$. As a result (14) and (15) reduce to,

$$\mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_X^{\text{prior}}}[\frac{\partial d}{\partial \boldsymbol{\phi}}(\boldsymbol{x};\boldsymbol{\phi}^*)] = \mathop{\mathbb{E}}_{\boldsymbol{z}\sim p_Z}[\frac{\partial d}{\partial \boldsymbol{\phi}}(\boldsymbol{g}(\boldsymbol{z};\boldsymbol{\theta}^*);\boldsymbol{\phi}^*)] \tag{16}$$

$$\mathop{\mathbb{E}}_{\boldsymbol{z}\sim p_Z}[\frac{\partial d}{\partial \boldsymbol{x}}(\boldsymbol{g}(\boldsymbol{z};\boldsymbol{\theta}^*);\boldsymbol{\phi}^*)\cdot\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{\theta}}(\boldsymbol{z};\boldsymbol{\theta}^*)] = \mathbf{0}. \tag{17}$$

Let $w_a(\boldsymbol{x}) \equiv \frac{\partial d}{\partial \phi_a}(\boldsymbol{x};\boldsymbol{\phi}^*), a = 1,\cdots,N_\phi$. Then (16) implies

$$\mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_X^{\text{prior}}}[w_a(\boldsymbol{x})] = \mathop{\mathbb{E}}_{\boldsymbol{z}\sim p_Z}[w_a(\boldsymbol{g}(\boldsymbol{z};\boldsymbol{\theta}^*))], a = 1,\cdots,N_\phi. \tag{18}$$

As $N_\phi \to \infty$, this is a weak statement of the equivalence of p_X^{prior} and p_X^{gen}, where the latter is defined in (3). In particular this says that the push forward of the measure in the latent space under

the function $\boldsymbol{g}(\boldsymbol{z})$ weakly converges to the measure associated with distribution of $\boldsymbol{x}$. We note with increasing number of weights in the discriminator, the relation above is required to hold for an increasing number of test functions, w_a. In addition, we have implicitly assumed that the generator is rich enough, that is it has enough weights/layers, such that this relation can actually be satisfied. To make this clear consider the extreme case of a generator with a single weight; in this case there is no way that (18) will be satisfied for a large number N_ϕ. Thus in order for this relation to hold for a large N_ϕ, we must also provide the generator with a large N_θ.

Now consider a sufficiently smooth function $m(\boldsymbol{x})$ which defines the point estimate we wish to compute. Expand it using $w_a(\boldsymbol{x})$ as basis such that

$$m(\boldsymbol{x}) - \sum_{a=1}^{N_\phi} \alpha_a w_a(\boldsymbol{x}) = \epsilon(\boldsymbol{x}), \tag{19}$$

and the coefficients are selected to minimize

$$\bar{\epsilon} = \|\epsilon(\boldsymbol{x})\|_\infty. \tag{20}$$

As we increase N_ϕ the dimension of the basis used to represent m increases, and therefore $\bar{\epsilon}$ tends to zero.

Given this,

$$\begin{aligned}
\mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_X^{\text{prior}}}[m(\boldsymbol{x})] &= \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_X^{\text{prior}}}[\sum_{a=1}^{N_\phi} \alpha_a w_a(\boldsymbol{x}) + \epsilon(\boldsymbol{x})], \qquad \text{from}(19) \\
&= \sum_{a=1}^{N_\phi} \alpha_a \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_X^{\text{prior}}}[w_a(\boldsymbol{x})] + \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_X^{\text{prior}}}[\epsilon(\boldsymbol{x})] \\
&= \sum_{a=1}^{N_\phi} \alpha_a \mathop{\mathbb{E}}_{\boldsymbol{z}\sim p_Z}[w_a(\boldsymbol{g}(\boldsymbol{z};\boldsymbol{\theta}^*))] + \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_X^{\text{prior}}}[\epsilon(\boldsymbol{x})], \qquad \text{from}(18) \\
&= \mathop{\mathbb{E}}_{\boldsymbol{z}\sim p_Z}[\sum_{a=1}^{N_\phi} \alpha_a w_a(\boldsymbol{g}(\boldsymbol{z};\boldsymbol{\theta}^*))] + \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_X^{\text{prior}}}[\epsilon(\boldsymbol{x})] \\
&= \mathop{\mathbb{E}}_{\boldsymbol{z}\sim p_Z}[m(\boldsymbol{g}(\boldsymbol{z};\boldsymbol{\theta}^*)) - \epsilon(\boldsymbol{g}(\boldsymbol{z};\boldsymbol{\theta}^*))] + \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_X^{\text{prior}}}[\epsilon(\boldsymbol{x})], \qquad \text{from}(19) \\
&= \mathop{\mathbb{E}}_{\boldsymbol{z}\sim p_Z}[m(\boldsymbol{g}(\boldsymbol{z};\boldsymbol{\theta}^*))] - \mathop{\mathbb{E}}_{\boldsymbol{z}\sim p_Z}[\epsilon(\boldsymbol{g}(\boldsymbol{z};\boldsymbol{\theta}^*))] + \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_X^{\text{prior}}}[\epsilon(\boldsymbol{x})]. \qquad (21)
\end{aligned}$$

This yields,

$$\begin{aligned}
|\mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_X^{\text{prior}}}[m(\boldsymbol{x})] - \mathop{\mathbb{E}}_{\boldsymbol{z}\sim p_Z}[m(\boldsymbol{g}(\boldsymbol{z};\boldsymbol{\theta}^*))]| &\leq |-\mathop{\mathbb{E}}_{\boldsymbol{z}\sim p_Z}[\epsilon(\boldsymbol{g}(\boldsymbol{z};\boldsymbol{\theta}^*))] + \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_X^{\text{prior}}}[\epsilon(\boldsymbol{x})]| \\
&\leq |\mathop{\mathbb{E}}_{\boldsymbol{z}\sim p_Z}[\epsilon(\boldsymbol{g}(\boldsymbol{z};\boldsymbol{\theta}^*))]| + |\mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_X^{\text{prior}}}[\epsilon(\boldsymbol{x})]| \\
&\leq \mathop{\mathbb{E}}_{\boldsymbol{z}\sim p_Z}[\|\epsilon(\boldsymbol{g}(\boldsymbol{z};\boldsymbol{\theta}^*))\|_\infty] + \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_X^{\text{prior}}}[\|\epsilon(\boldsymbol{x})\|_\infty] \\
&\leq \mathop{\mathbb{E}}_{\boldsymbol{z}\sim p_Z}[\bar{\epsilon}] + \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_X}[\bar{\epsilon}], \qquad \text{from}(20) \\
&= 2\bar{\epsilon}. \qquad (22)
\end{aligned}$$

Note that in deriving this estimate, in the third line we have assumed that $\boldsymbol{g}(\boldsymbol{z};\boldsymbol{\theta}^*) \in \Omega_x$. That is, the generator does not land outside the domain of $\boldsymbol{x}$. The statement above bounds the error in computing a population parameter for the prior using a GAN.

A.3 Convergence to the posterior density

In order to turn this estimate into an estimate for the error in approximating a population parameter for the posterior density, we simply choose $m(\boldsymbol{x}) = \frac{l(\boldsymbol{x})p_\eta(\hat{\boldsymbol{y}}-\boldsymbol{f}(\boldsymbol{x}))}{\mathbb{Z}}$. With this choice,

$$\mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_X^{\text{prior}}}[m(\boldsymbol{x})] = \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_X^{\text{post}}}[l(\boldsymbol{x})], \qquad \text{from (1)} \tag{23}$$

and

$$\mathop{\mathbb{E}}_{\boldsymbol{z}\sim p_Z}[m(\boldsymbol{g}(\boldsymbol{z};\boldsymbol{\theta}^*))] = \mathop{\mathbb{E}}_{\boldsymbol{z}\sim p_Z^{\rm post}}[l(\boldsymbol{g}(\boldsymbol{z};\boldsymbol{\theta}^*))], \qquad \text{from (6).} \tag{24}$$

Using these in (22) we have the desired result,

$$|\mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_X^{\rm post}}[l(\boldsymbol{x})] - \mathop{\mathbb{E}}_{\boldsymbol{z}\sim p_Z^{\rm post}}[l(\boldsymbol{g}(\boldsymbol{z};\boldsymbol{\theta}^*))]| \quad \leq \quad 2\bar{\epsilon}. \tag{25}$$

This result demonstrates that the difference between the true population parameter for the posterior and its approximation obtained after using a GAN as a prior (the method proposed in the paper) reduces as the number of weights in the generator and the discriminator are increased. In the limit of infinite weights, $\bar{\epsilon} \to 0$, and the equation above is a statement of weak convergence of the posterior obtained by using the GAN and the true posterior density.

B EXPRESSION FOR THE MAXIMUM A-POSTERIORI ESTIMATE

The techniques described in Section 2.1 focus on sampling from the posterior distribution and computing approximations to population parameters. These techniques can be applied in conjunction with any distribution used to model noise and the latent space vector; that is, any choice of p_η (likelihood) and p_Z (prior). In this section we consider the special case when Gaussian models are used for noise and the latent vector. In this case, we can derive a simple optimization algorithm to determine the maximum a-posteriori estimate (MAP) for $p_Z^{\rm post}(\boldsymbol{z}|\boldsymbol{y})$. This point is denoted by $\boldsymbol{z}^{\rm map}$ in the latent vector space and represents the most likely value of the latent vector in the posterior distribution. It is likely that the operation of the generator on $\boldsymbol{z}^{\rm map}$, that is $\boldsymbol{g}(\boldsymbol{z}^{\rm map})$, will yield a value that is close to $\boldsymbol{x}^{\rm map}$, and may be considered as a likely solution to the inference problem.

We consider the case when the components of the latent vector are iid with a normal distribution with zero mean and unit variance. This is often the case in many typical applications of GANs. Further, we assume that the components of noise vector are defined by a normal distribution with zero mean and a covariance matrix $\boldsymbol{\Sigma}$. Using these assumptions in (6), we have

$$p_Z^{\rm post}(\boldsymbol{z}|\boldsymbol{y}) \propto \exp\Big(-\frac{1}{2}\overbrace{\big(|\boldsymbol{\Sigma}^{-1/2}(\hat{\boldsymbol{y}} - \boldsymbol{f}(\boldsymbol{g}(\boldsymbol{z})))|^2 + |\boldsymbol{z}|^2\big)}^{\equiv r(\boldsymbol{z})}\Big). \tag{26}$$

The MAP estimate for this distribution is obtained by minimizing the negative of the argument of the exponential. That is

$$\boldsymbol{z}^{\rm map} = \arg\min_{\boldsymbol{z}} r(\boldsymbol{z}). \tag{27}$$

This minimization problem may be solved using any gradient-based optimization algorithm. The input to this algorithm is the gradient of the functional r with respect to $\boldsymbol{z}$, which is given by

$$\frac{\partial r}{\partial \boldsymbol{z}} = \boldsymbol{H}^T(\boldsymbol{z})\boldsymbol{\Sigma}^{-1}(\boldsymbol{f}(\boldsymbol{g}(\boldsymbol{z})) - \hat{\boldsymbol{y}}) + \boldsymbol{z}, \tag{28}$$

where the matrix $\boldsymbol{H}$ is defined as

$$\boldsymbol{H} \equiv \frac{\partial \boldsymbol{f}(\boldsymbol{g}(\boldsymbol{z}))}{\partial \boldsymbol{z}} = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{z}}. \tag{29}$$

Here $\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}$ is the derivative of the forward map $\boldsymbol{f}$ with respect to its input $\boldsymbol{x}$, and $\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{z}}$ is the derivative of the generator output with respect to the latent vector. In evaluating the gradient above we need to evaluate the operation of the matrices $\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}$ and $\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{z}}$ on a vector, and not the matrices themselves. The operation of $\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{z}}$ on a vector can be determined using a back-propagation algorithm with the GAN; while the operation of $\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}$ can be determined by making use of the adjoint of the linearization of the forward operator.

Once $\boldsymbol{z}^{\rm map}$ is determined, one may evaluate $\boldsymbol{g}(\boldsymbol{z}^{\rm map})$ by using the GAN generator. This represents the value of the field we wish to infer at the most likely value value of latent vector. Note that this is not the same as the MAP estimate of $p_X^{\rm post}(\boldsymbol{x}|\boldsymbol{y})$.

C MORE RESULTS

In this section we provide additional results for both MNIST and CelebA dataset for different tasks discussed in the main paper.

C.1 MNIST

First we provide additional examples in figure 7 for variance-based adaptive measurement window selection procedure described in section 3.1,

Figure 8 shows additional results for the inpainting task, where different MNIST digits are occluded with mask of different size and at different location.

Note that the variance is high where the occlusion mask is located indicating lower confidence in reconstructed image in that location. This could be very useful in applications like self-driving cars or medical imaging, where often times the goal is to recover original image from occluded (and possibly noisy) measurements and this quantified uncertainty information can help in high-impact down stream decision making process.

C.2 CELEBA

For CelebA dataset, we first trained a GAN model (WGAN-GP) using more than 200,000 celebrity facial images. The input images were cropped to a 64x64 RGB image and were normalized between [-1, 1]. We use latent space dimension of 100. The architecture for this dataset was slightly different than that for MNIST and heat conduction problem and is explained in Appendix D.

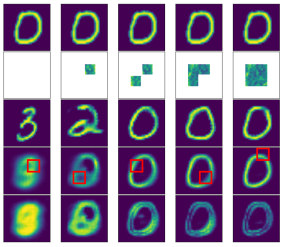
Once the GAN was trained, we use the MCMC algorithm proposed in section 2.1 for posterior sampling and inference on complimentary set of images (the one that were not used for training). In figure 9 we show some additional results for variance-based adaptive measurement window selection procedure for CelebA dataset.

Next, in figure 10 we show some additional results for image recovery task for CelebA dataset.

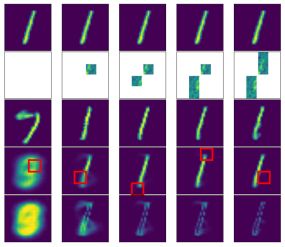
D ARCHITECTURE DETAILS

We used the same generator and discriminator architecture for MNIST and synthetic dataset used in physics-based inference problem, whereas for the CelebA dataset we used slightly different architecture.. The layout of both these architecture is shown in figure 11. Some notes regarding nomenclature used in the figure 11.

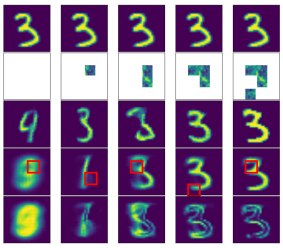
- Conv (HxWxC s=n) indicates convolutional layer with filer size of HxW and number of filters=C with stride=n.
- BN = Batch norm, LN = Layer norm.
- TrConv = Transposed Convolution



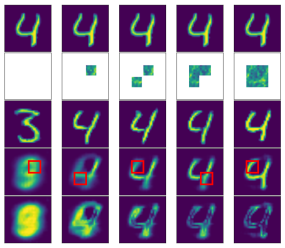
(a) Digit 0



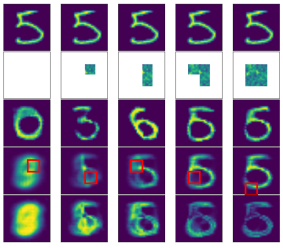
(b) Digit 1



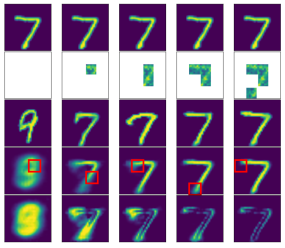
(c) Digit 3



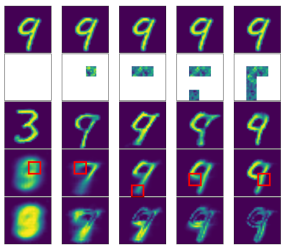
(d) Digit 4



(e) Digit 5



(f) Digit 7



(g) Digit 9

Figure 7: Estimate of the MAP (3rd row), mean (4th row) and variance (5th row) from the limited view of a noisy image (2nd row) using the proposed method. The window to be revealed at a given iteration (shown in red box) is selected using a variance-driven strategy. Top row indicates ground truth. For all the digits noise variance of 1 is used.

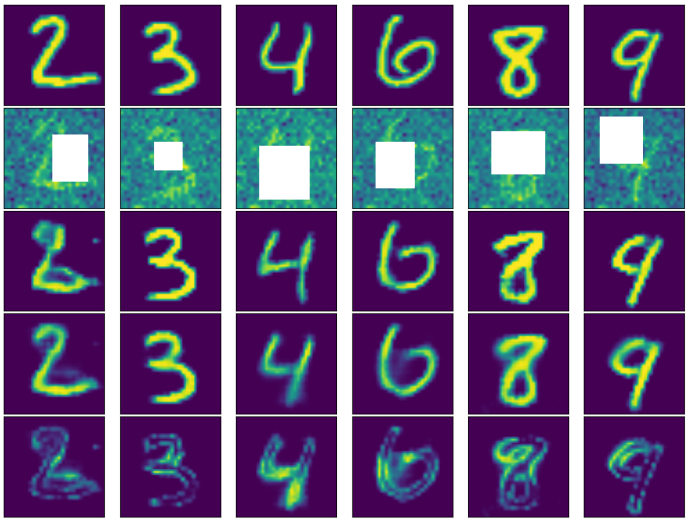
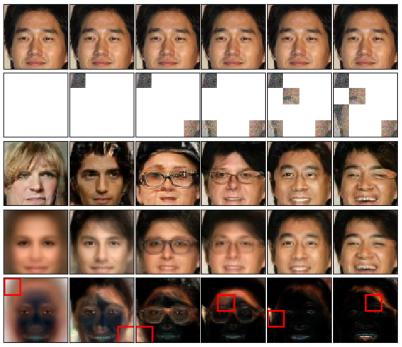
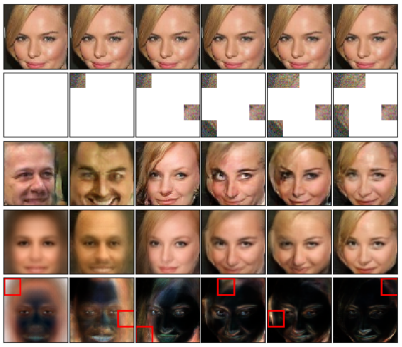


Figure 8: Estimate of the MAP (3rd row), mean (4th row) and variance (5th row) from a noisy image (2nd row) using the proposed method. Top row shows ground truth. For all the examples noise variance of 1 is used.

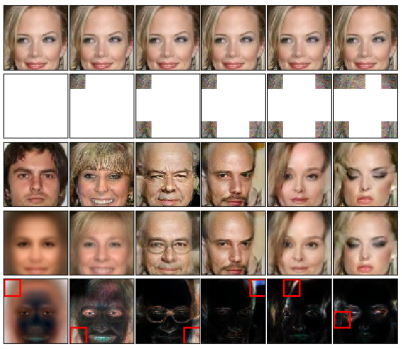


(a)

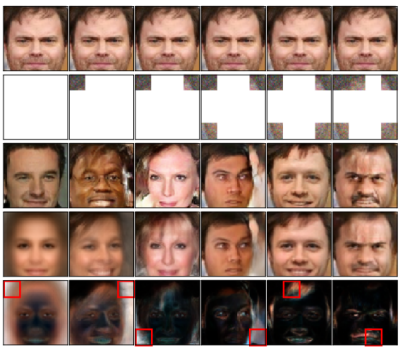


(b)

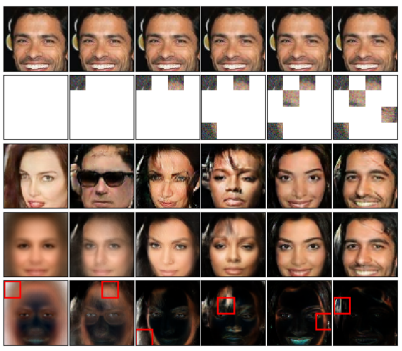
(c)



(d)



(e)



(f)

Figure 9: Estimate of the MAP (3rd row), mean (4th row) and variance (5th row) from the limited view of a noisy image (2nd row) using the proposed adaptive method. The window to be revealed at a given iteration (shown in red box) is selected using a variance-driven strategy. Top row indicates ground truth. For all the images fixed noise variance of 0.1 is used.

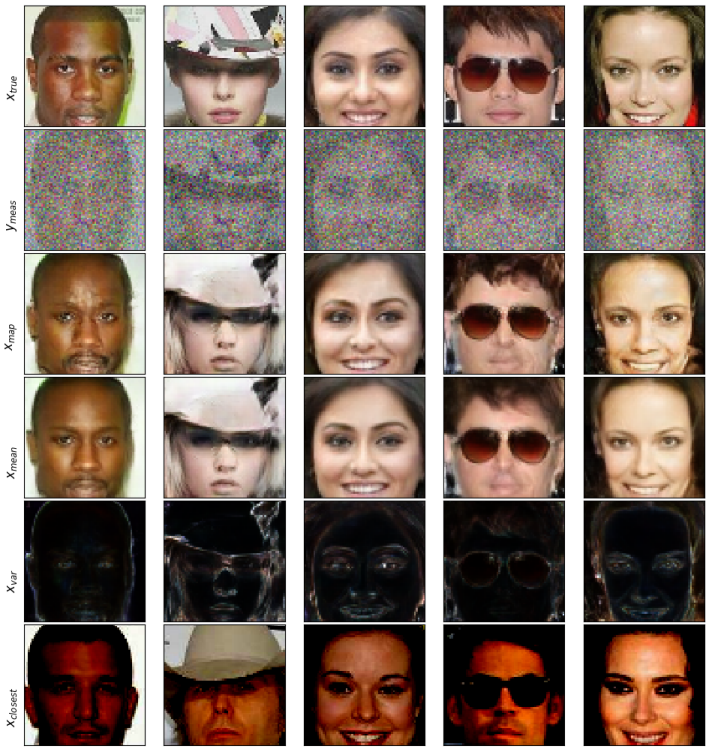
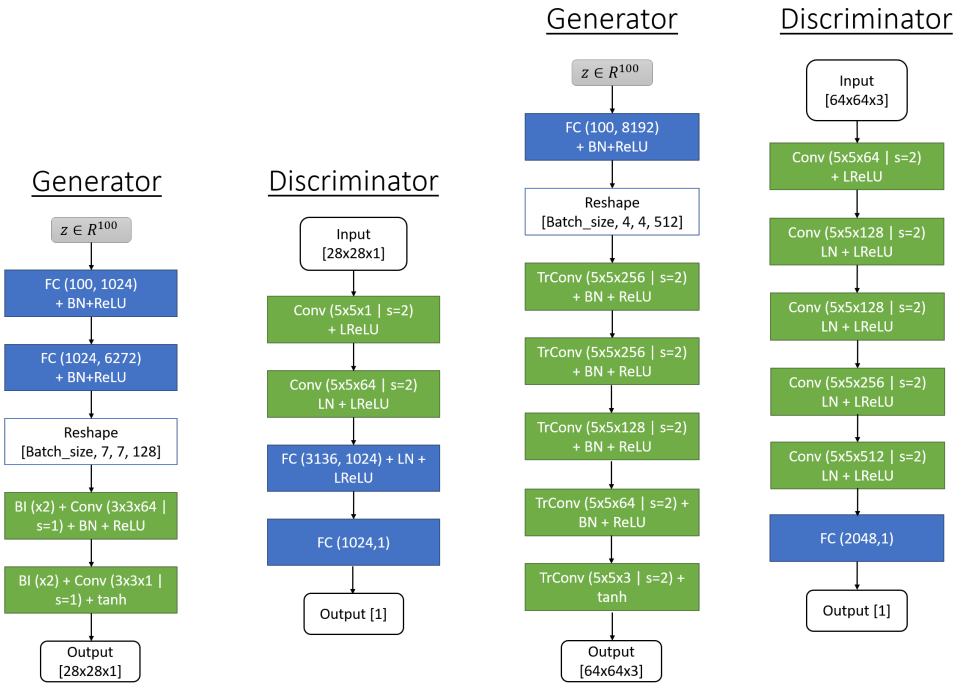


Figure 10: Estimate of the MAP (3rd row), mean (4th row) and variance (5th row) from a noisy image (2nd row) using the proposed method. Top row shows ground truth. The last row shows the closest example in training set (in L_2 sense). For all the examples noise variance of 1 is used.



(a) Architecture for MNIST and synthetic dataset (used in physics-based inference problem)

(b) Architecture for CelebA dataset

Figure 11: Generator and discriminator architectures,