Generative Patch Priors for Practical Compressive Image Recovery

Rushil Anirudh
Lawrence Livermore National Laboratory,
Livermore, CA
anirudh1@llnl.gov

Suhas Lohit
Misubishi Electric Research Laboratories
Cambridge, MA
slohit@merl.com

Pavan Turaga
Arizona State University
Tempe, AZ.
pturaga@asu.edu

Abstract

In this paper, we propose the generative patch prior (GPP) that defines a generative prior for compressive image recovery, based on patch-manifold models. Unlike learned, image-level priors that are restricted to the range space of a pre-trained generator, GPP can recover a wide variety of natural images using a pre-trained patch generator. Additionally, GPP retains the benefits of generative priors like high reconstruction quality at extremely low sensing rates, while also being much more generally applicable. We show that GPP outperforms several unsupervised and supervised techniques on three different sensing models – linear compressive sensing with known, and unknown calibration settings, and the non-linear phase retrieval problem. Finally, we propose an alternating optimization strategy using GPP for joint calibration-and-reconstruction which performs favorably against several baselines on a real world, un-calibrated compressive sensing dataset.

1. Introduction

Deep generative priors have proven to be an effective alternative to supervised methods in a variety of inverse problems from compressive sensing [6], in-painting [59], to blind recovery [20, 2]. These priors are expressed using Generative Adversarial Networks (GAN) [17] or Variational Autoencoders (VAE) [29], which are pre-trained to approximate the image manifold. The solution to the inverse problem is then constrained to lie on this manifold via optimizing in the latent space of the generative model. Due to their ability to characterize the image manifold better than more traditional priors like sparsity [9] or total variation [32], these learned priors work well in recovering the true signal, particularly under extreme sensing scenarios with very few observations. However, they are not applicable when the image to be recovered lies far away from the modes of the training distribution. As a result, generative priors today are only applicable for specific domains on which high quality GANs (and therefore better generative priors) can be trained, such as digits or faces.

As such, a generic generative prior that is able to aid in recovering high quality, and high resolution natural images does not exist today. This is partly owing to the fact that training an unconditional GAN on a dataset like ImageNet [50] remains a challenge due to its complexity. Even if such a generative model were trained, it is still likely to be limited in the resolution and aspect ratios of images that can be recovered. Accordingly, we hypothesize that there exists a trade-off between *generalizability* of the prior (i.e., ability to recover a wide variety of images) versus its *efficiency*, (i.e. effectiveness in recovering accurate solutions with few observations). As an example, traditional priors like sparsity and more recently untrained network priors [55] generalize better because they make fewer assumptions on the properties of the image manifold, but they suffer from poor quality of image reconstruction

under few observations. In contrast, high quality generative priors have been shown to be highly efficient [6, 51], while being applicable only to images from the training distribution. Another issue with most existing priors, including generative priors, is that they require knowledge of the *exact* measurement operator, which is unrealistic in a real world application where there are bound to be issues like sensor drift or environmental and hardware artifacts that may not be explained away by noisy observations. These are typically corrected by sensor calibration, requiring human intervention that is expensive and impractical. On the other hand, not accounting for these calibration artifacts can severely degrade the performance of existing techniques.

In this paper, we address these issues for the problem of compressive image recovery [9, 56] using a deep generative *patch* model that relaxes the prior from the image level to the patch level. We call this the Generative Patch Prior (GPP), and it exploits the relatively simpler parameterization of the patch manifold, instead of the image manifold, using a generative adversarial network. Since GPP is defined at the patch level, it generalizes better than existing generative image priors, to a variety of natural images as well as images of arbitrary size and aspect ratio. In terms of the generalizability vs efficiency trade-off, GPP is significantly more generalizable (but less efficient) than a generative *image* prior, while also being more efficient (but less generalizable) than existing unsupervised priors, making it much more practically applicable compared to both. Figure 1 demonstrates the recovery process using GPP on a previously unseen, high resolution image.

Within compressive imaging, the idea of using patch-manifolds can be further motivated by practical hardware acquisition conditions since measurement matrices for very high-resolution images can become impractically large. As a result, many practical compressive imagers are actually implemented as patch-level or block-compressive imagers (c.f. [27]). This means compressive measurements are indeed taken at the patch level, thus recovering the patches and then composing them into a picture requires a proper patch-manifold prior with a compositional prior. Further, practical CS imagers need an additional layer of calibration due to slight misalignments between the photosensor and the compressive optical elements. Usually, this calibration is done as a pre-processing step and fixed, but is prone to error and drift over time. To overcome this drawback, we additionally propose a self-calibrating (SC) mechanism using generative priors, i.e., a recovery process that is able to adjust automatically to the required calibration setting. We assume a scale and shift calibration model, which accounts for common calibration issues like sensor gain or other physical changes in the measurement process. We demonstrate empirically that GPP outperforms several unsupervised priors, including recent untrained network-based priors [55, 26].

Main contributions:

1. We propose the generative patch prior (GPP)– which enhances the applicability of generative priors to the entire set of natural images, as opposed to only being restricted to the training distribution.

2. We show that the patch manifold can be approximated well by standard datasets (like CIFAR), for high quality CS image recovery, compared to existing unsupervised priors.

3. We extend the idea of self-calibration for generative priors, which minimizes the need to intervene for manual sensor-level calibration in compressive imaging systems.

4. Finally, we demonstrate that the proposed prior and calibration model achieves significantly better reconstruction quality on *un-calibrated* measurements obtained from a real block compressive imager validating our approach under real unknown sensor noise and calibration.

2. Background

2.1. Preliminaries

Consider a vectorized square block of an image $\mathbf{x} \in \mathcal{X} \subset \mathbb{R}^n$ which we want to sense, and denote by $\mathbf{y} \in \mathbb{R}^m$ the compressive measurements obtained by the sensor. Given a measurement matrix $\Phi \in \mathbb{R}^{m \times n}$, with $m < n$ and $\Phi_{i,j} \sim \mathcal{N}(0, 1)$, the compressive recovery problem is to estimate $\mathbf{x}$ accurately from $\mathbf{y}$. We consider three different sensing scenarios in this paper to benchmark the performance of the proposed patch-based prior–(a) Compressive sensing with
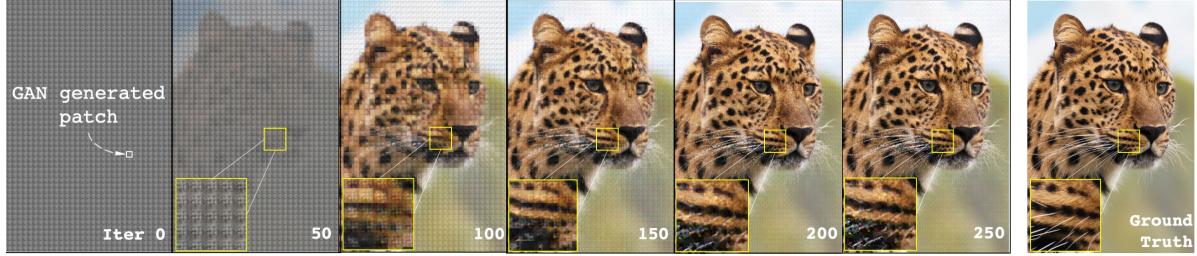
Figure 1: **CS Recovery with GPP:** The generative patch prior is able to capture the diversity of the natural image manifold with a single pre-trained GAN irrespective of the size or aspect ratio, unlike existing generative priors that do not generalize outside of their training distribution. Here, we illustrate the recovery process (without any post processing) at a measurement rate of 10%. The original image is of size 1536×1024, and is recovered by a GAN trained on CIFAR-C [23] of size 32×32.

known calibration (i.e. the ideal setting): $\mathbf{y} = \Phi\mathbf{x}$; (b) Compressive sensing with unknown calibration: $\mathbf{y} = (a\Phi + b\mathbf{1})\mathbf{x}$, where $a, b \in \mathbb{R}^1$ are unknown calibration parameters and have to be estimated, and $\mathbf{1} \in \mathbb{R}^{m \times n}$ is a matrix of the same size as Φ with 1s; and finally (c) Compressive phase retrieval, which has non-linear forward process defined by $\mathbf{y} = |\Phi\mathbf{x}|$.

As $m < n$, reconstructing $\mathbf{x}$ from $\mathbf{y}$ in any of the above models is an ill-posed inverse problem and therefore has infinite number of feasible solutions. We define the Measurement Rate (MR) to be the ratio $\frac{m}{n}$. However, if we have additional knowledge about the structure of the solution space $\mathcal{X}$ (in this paper, implicitly via the set of natural patches), it is possible to recover $\mathbf{x}$ nearly perfectly given m is sufficiently high and with graceful degradation as m is reduced.

2.2. Related Work

Priors for Inverse Problems: Some of the most effective early work on designing priors for inverse problems came in the form of sparsity/compressibility in the wavelet and gradient domains [9, 8, 42, 7, 57] and hard-crafted prior models that were constructed based on such observations. Subsequently, more complex models were proposed such as wavelet trees and block sparsity [3, 25], non-local low-rank regularization [13] and denoiser-based regularization [39]. These models are ineffective at low measurement rates (< 0.1) and extremely slow in terms of computational complexity. For a given Φ, supervised deep learning based methods have addressed these issues, like denoising autoencoders [41], or convolutional neural networks (CNNs) [31, 5, 43, 49]. These ideas have also been extended to the phase retrieval setting in applications such as Fourier Ptychography [5]. While these trained models do not work with new Φs straight away, there has

been work to make them more adaptable [37]. Another class of algorithms is obtained by unrolling/unfolding iterative algorithms which combines the advantages of the model-based and data-driven approaches, where a neural network serves as a sparsifying transform [18, 54, 61], a learned denoiser [38, 40], or a projector onto the image manifold [48]. These methods overcome the drawbacks of earlier methods. However, they still rely on paired data for training. Some of these requirements are addressed by [58], where the image reconstruction algorithm is trained entirely using only the measurements for a given measurement operator. The limitation of paired training data can be overcome by using deep trained/untrained priors, which have the representational power of deep neural networks, while not requiring paired training like classical approaches. Trained priors like generative models [6, 51, 21, 53], and untrained network priors like deep image prior [55, 22, 26] have been empirically shown to be useful compared to traditional priors for solving ill-posed problems in imaging. However, existing generative priors don't generalize out of distribution, and untrained network priors are extremely slow and fail under low sensing regimes; both of which are addressed by our proposed method, GPP.

Self Calibration: In compressive sensing, the idea of self-calibration has been extensively studied as the problem of basis mismatch [10, 33], although it is studied in the context of more traditional image priors such as sparsity in some spectral basis, whereas our focus is on more recent generative priors. More generally, the self-calibration problem has been studied in different contexts like medical imaging [19], camera self-calibration [46] and in other bi-linear inverse problems [34]. The SC problem is also indirectly

3

related to several recent efforts in solving problems 'blindly', with unknown corruptions [1], filters [58, 2], or demodulation [20].

Patch-based models: Since smaller patches are statistically simpler and easier to model than entire image, priors designed from on image patches have been used in computer vision and image processing for a long time [15, 45, 44, 11]. GPP is also related to patch-based dictionary learning approaches used for denoising problems like KSVD [14] where we expect the patch generator to act as a more powerful, non-linear basis than the one learned using KSVD. More recently, deep learning based on patches from a single image have emerged as powerful regularizers like in untrained network priors [60], and generative models [52], and inverse imaging [16]. These approaches do not focus on approximating the space of the entire patch manifold that can serve as a generic image prior, instead focusing on space of patches related to a single image.

3. A patch-manifold based generative prior for natural images

While generative image priors have been reasonably successful in solving under-constrained inverse problems for low-resolution images (128×128), they do not generalize to new domains that are not represented by the training distribution and are difficult to scale up to higher resolution natural images partly because the measurement operators for such images can become impractically large. Further, it remains extremely challenging to train a generative model to approximate the entire natural image manifold of complex, high resolution images. Instead, we propose a patch-based generative model which addresses these problems. As patches are assumed to be 'small' and with much simpler spatial statistics than full images, the so-called "patch manifold" [45] is therefore easier to parameterize when compared to the true image manifold. This has been the driving factor behind using patch-based models in computer vision for the last two decades [15, 45, 11].

In this paper, we are interested in using a generative adversarial network (GAN) [17] to approximate the patch manifold. As a result, we are able to leverage the representational power of GANs, and provide a *differentiable* way to project onto the true image manifold.

To the best of our knowledge, there does not exist such a patch-based generative prior for inverse problems. Given an image $\mathbf{x} \in \mathcal{X}$, the generative prior for images is implemented by constraining the solution to the inverse problem, $\mathbf{x}$ to the range space of a pre-trained generative model, $\{\mathcal{G}(\mathbf{z})\} \ \forall \ \mathbf{z}$, where the $\mathcal{G}$ is the generator, and $\mathbf{z} \in \mathbb{R}^d$ is sampled from a known probability distribution like the uniform distribution $\mathcal{U}(0, 1)$. By definition, this generative image prior can only recover images that are expressed by the $\mathcal{G}$. Accordingly, the solution to the inverse problem is given by $\mathbf{x}^* = \mathcal{G}(\mathbf{z}^*)$, where for a loss function $\mathcal{L}$ like the mean squared error, $\mathbf{z}^* = \arg\min_{\mathbf{z} \in [0,1]^d} \mathcal{L}(\mathbf{y}, \Phi \, \mathcal{G}(\mathbf{z}))$.

Generative Patch Prior: Formally, a patch is is defined as $x_p = \mathcal{F}(\mathbf{x}, c, \tau)$, where c is the center of the patch with width τ, on an image $\mathbf{x}$ defined on uniform grid $[0, 1]^2$, and $\mathcal{F}$ is the patch transform [11] which decomposes an image into N_p equally sized non-overlapping patches. First, we train a GAN on a dataset of patches until convergence. This dataset can be obtained either by using the patch transform on a dataset with larger images like ImageNet [50], or directly using a dataset with smaller images like CIFAR [30]. While traditional patch-based methods assume much smaller patches (of size 4×4 or 8×8) to be texture-features with mostly low frequency spatial information, we find that datasets like CIFAR with relatively larger patches can form effective learned priors. Once we have the pre-trained GAN, the solution to the inverse problem with the generative image prior is reformulated in terms of a patch-based generator as follows:

$$
\mathbf{z}_i^* = \arg\min_{\mathbf{z}_i \in [0,1]^d} \mathcal{L}(\mathbf{y}_i, \Phi \, \mathcal{G}(\mathbf{z}_i)), \forall i = \{1, \dots, N_p\}
$$
$$
\text{and } \mathbf{x}^* = \mathcal{F}^{-1}(\mathcal{G}(\mathbf{z}_1^*), \dots, \mathcal{G}(\mathbf{z}_{N_p}^*)), \quad (1)
$$

where $\mathbf{y}_i$ corresponds to the measurements obtained under the measurement operator Φ for each individual patch, and $\mathcal{F}^{-1} : \mathcal{P} \times \mathcal{P} \cdots \times \mathcal{P} \mapsto \mathcal{X}$ is the inverse patch transform, which we also refer to as a compositional prior, and takes a set of patches to produce the final image. Since we treat the patches independently, under very few measurements this can lead to blocking artifacts. This can be alleviated using a filtering mechanism like BM3D [12]. For fairness, we compare all baselines with the same filtering on the recov-

ered image. Figure 1 shows the optimization process for a high resolution image of size 1536×1024, of non-standard aspect ratio recovered with a measurement rate of 10%.

Relationship to generative *image* priors: GPP relaxes the generative image prior to the level of a patch, and as a result is far more applicable. The GPP can effectively recover any natural image with nearly zero representation error unlike GAN priors. The two become equivalent when the size of the patch is equal to the size of the image, i.e., $N_p = 1$. A generic unconditional GAN that approximates the natural image manifold does not exist today, however, in cases where GANs can be trained well like faces, animals, cars etc. it is expected that for an image from that specific distribution, the GAN prior will outperform GPP at low measurement rates. We explore these properties further in the experiments section.

4. Self Calibration with GPP

Most existing deep prior formulations in literature assume access to the exact measurement matrix Φ in order to solve the inverse problem. Any changes to Φ that are unknown can render such techniques useless due to errors in the optimization objective. This creates practical issues since there are likely going to be differences in the measurement operator realized using hardware sensors compared with the operator used during training. The exact nature of change in the hardware is hard to determine in advance and this issue is corrected by a manual calibration step requiring human intervention. Instead, here we pursue a self-calibrating approach that reduces the need for such manual interventions.

Calibration model: We propose a form of self-calibration that relaxes the need to know the exact Φ. We assume a simple calibration model using two parameters: $\tilde{\Phi} = \tilde{a}(\Phi + \tilde{b}\mathbf{1})$, where $\tilde{a} \in \mathbb{R}$ accounts for changes in sensor gain (mapping maximum intensity color white to 1.0), $\tilde{b} \in \mathbb{R}$ models unknown parameters in the measurement operator with a simple bias term and $\mathbf{1} \in \mathbb{R}^{m \times n}$ denotes a matrix of ones. For notational convenience, we rewrite the calibration model as $\tilde{\Phi} = a\Phi + b\mathbf{1}$. The compressive sensing problem is re-formulated as $\mathbf{y} = (a\Phi + b\mathbf{1})\mathbf{x}$, with three unknowns: $a, b, \mathbf{x}$. For known a, b, i.e. a manually cali-

brated measurement, $\mathbf{x}$ can be readily estimated using (1) using the proposed GPP prior, or any other image-prior. Note that the loss function for patch i now becomes:

$$\mathcal{L} = \|\mathbf{y}_i - (a\Phi + b\mathbf{1})\mathbf{x}_i)\|^2 \qquad (2)$$

where $\mathbf{x}_i = \mathcal{G}(\mathbf{z}_i)$ for some $\mathbf{z}_i$.

Solving for a and b: We use alternating minimization to solve for the unknowns $a, b, \mathbf{x}^*$. First, given an $\mathbf{x}$ from the GPP model we estimate a, b assuming the current estimate of $\mathbf{x}$ is the true solution. As our calibration model is linear in a, b these can be solved exactly to minimize the cost in Eqn. (2):

$$a^* = \frac{c_\mathbf{1}\lambda - c_\Phi\theta_\mathbf{1}}{\lambda^2 - \theta_\Phi\theta_\mathbf{1}} \text{ and } b^* = \frac{c_\mathbf{1} - a^*\lambda}{\theta_\mathbf{1}}, \qquad (3)$$

where we define scalar quantities for notational convenience: $c_\Phi = \mathbf{y}^\intercal\Phi\mathbf{x}, c_\mathbf{1} = \mathbf{y}^\intercal\mathbf{1}\mathbf{x}, \theta_\Phi = (\Phi\mathbf{x})^\intercal(\Phi\mathbf{x}), \theta_\mathbf{1} = (\mathbf{1}\mathbf{x})^\intercal(\mathbf{1}\mathbf{x}), \lambda = (\Phi\mathbf{x})^\intercal(\mathbf{1}\mathbf{x})$. It is easy to check that, in the ideal case when $\mathbf{y} = \Phi\mathbf{x}$ (i.e. no calibration is necessary), $c_\mathbf{1} = \lambda; c_\Phi = \theta_\Phi$ and as a result we obtain $a^* = 1, b^* = 0$ as expected. The details of complete derivation are available in the appendix. The algorithm for self-calibration with GPP is shown in Algorithm 1.

Algorithm 1 Self-calibration with the generative patch prior

Input: Patch-wise observations $\mathbf{y_i}$, $i = 1 \ldots N_p$, Measurement operator Φ, Pre-trained patch generator $\mathcal{G}$.
Output: Recovered full Image $\mathbf{x}^* \in \mathbb{R}^{nN_p}$, calibration parameters a^*, b^*.
Input: Initialize: $\mathbf{z}_i^{(0)} \sim \mathcal{U}(\mathbf{0}, I)$ for patches $i = 1 \ldots N_p$
1: **while** not converged **do**
2:     $\mathbf{x}_i^{(t)} = \mathcal{G}(\mathbf{z}_i) \; \forall \; i = 1, \ldots, N_p$
3:     Given $\mathbf{x}_i^{(t)}$, compute $a_i^{(t+1)}, b_i^{(t+1)}$ from (3) $\forall i$
4:     $a^{(t+1)} \leftarrow \frac{1}{N_p}\sum_i^{N_p} a_i^{(t+1)}$; and $b^{(t+1)} \leftarrow \frac{1}{N_p}\sum_i^{N_p} b_i^{(t+1)}$
5:     Compute cost $\mathcal{L}$ from (2)
6:     Latent space optimization: $\mathbf{z}_i^{(t+1)} \leftarrow \mathbf{z}_i^{(t)} - \gamma_g \nabla_\mathbf{z}\mathcal{L} \quad \forall i$
7: **end while**
8: $\mathbf{x}^* = \mathcal{F}^{-1}(\mathcal{G}(\mathbf{z}_1^*), \ldots, \mathcal{G}(\mathbf{z}_{N_p}^*))$

Since our generative model is defined at a patch level, we estimate a, b for each individual patch i sep-

arately and assign the mean values of all the patch-estimates as the single a, b for the entire image: $a = \frac{1}{N} \sum_{i=1}^{N} a_i$; and $b = \frac{1}{N} \sum_{i=1}^{N} b_i$. Finally, by plugging in the estimates for a^*, b^* in the cost function (2), we can solve for $\mathbf{x}^*$ using the optimization outlined in (1). We continue with this alternating minimization until the loss in (2) converges. In practice, we find that the algorithm converges within 1500 iterations, and finding recovery and convergence properties of this algorithm remain part of our future work.

5. Experiments

In this section, we benchmark the performance of the proposed GPP and other commonly used unsupervised priors for compressive image recovery. We show that GPP serves as a highly effective prior under very low sensing scenarios ($1 - 10\%$ measurement). We also demonstrate the robustness of the proposed self-calibration by introducing artificial calibration errors in the form of gain (a) and shift (b). Finally, we demonstrate that the combined model GPP with self-calibration can recover very high quality images directly from un-calibrated measurements obtained using a real compressive imaging camera [31].

Deep generative patch model training In our experiments we use non-overlapping patches of size of 32×32, and train a DCGAN [47] for 100K iterations in TensorFlow. We experiment with several different datasets to approximate the patch manifold—tinyimagenet [1], Imagenette[2], CIFAR[30], CIFAR-C [23], and MIT Places [62]. In all the following experiments we report results from the GAN trained on CIFAR-C as the default.

5.1. Compressive Image recovery with GPP

We first study the problem of compressive sensing, where we are provided with n measurements of an unknown image obtained as a linear projection of a measurement operator Φ, which is typically a Gaussian random matrix. Our test image set consists of 7 standard images used commonly to benchmark compressive sensing reconstruction algorithms. These images are grayscale, and of size 256×256, therefore using

---

[1] https://tiny-imagenet.herokuapp.com/

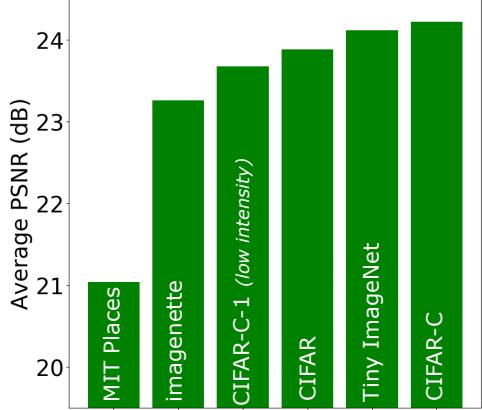[2] https://github.com/fastai/imagenette

Figure 2: Approximating the patch-manifold with commonly used image datasets. Here we train a patch-generator with each dataset shown and use it as the backend for GPP. We evaluate it's effectiveness in the compressive image recovery problem at 10% measurement rate and report average PSNR values.

non-overlapping patches of size 32×32, every image is represented as a batch of $\frac{256^2}{32^2} = 64$ patches. The measurements are taken by vectorizing each patch so our $n = 32^2 = 1024$, and $\Phi \in \mathbb{R}^{m \times 1024}$. We perform reconstruction using the two methods proposed here: (i) GPP and (ii) GPP + SC, which also includes the self-calibration step in addition to projected gradient descent (PGD) over the range of the patch-generator. We study the performance in two settings – 1% measurement rate, where $m = 10$, and 10% measurement rate with $m = 102$. We use the Adam optimizer [28] in all our experiments. For GPP , we use a learning rate (LR) of $5e - 3$, and run PGD for 1500 steps; for GPP +SC we use LR = $5e - 2$ for the first 200 iterations, and LR = $5e - 3$ for the next 1800.

Baselines: In addition to popular unsupervised benchmarks such as TVAL3[32], NLR-CS[13], and D-AMP[39], we implement a patch version of deep image prior (DIP) [55] or untrained network priors [26]. Here, we train a randomly initialized network to predict a batch of image patches, instead of a single image. We do this because it becomes impractical to implement a dense measurement operator for the full image of size 256×256, and we also observe that the patch-wise version converges much quicker. We use the same patch wise for comparison as in GPP with a LR = $5e - 4$ and run it for 10000 iterations for each image.

Method	Barbara	Parrot	Lena	Foreman	Cameraman	House	Monarch	Avg.
			Measurement Rate 10%					
TVAL3 [32]	**22.21**	23.16	<u>24.16</u>	<u>28.74</u>	**21.92**	26.32	<u>21.16</u>	<u>23.95</u>
NLR-CS [13]	14.67	14.16	15.33	13.56	14.22	14.80	14.67	14.49
D-AMP [39]	21.23	21.64	22.47	25.58	20.35	24.71	19.00	22.14
DIP [55, 26]	21.20	22.88	23.02	26.92	21.74	24.47	20.68	22.98
GPP +SC (ours)	22.18	<u>23.27</u>	21.95	28.11	21.18	<u>26.41</u>	19.46	23.22
GPP (ours)	<u>22.19</u>	**23.78**	**24.21**	**29.05**	<u>21.61</u>	**26.99**	**21.72**	**24.22**
			Measurement Rate 1%					
TVAL3	11.96	11.46	11.89	11.01	12.00	11.90	11.11	11.61
NLR-CS	5.86	5.44	6.27	4.25	6.31	5.29	6.71	5.73
D-AMP	5.48	5.09	5.96	3.83	5.64	5.02	6.20	5.31
DIP	14.07	12.83	14.68	13.53	12.42	14.89	13.64	13.72
GPP +SC (ours)	<u>17.16</u>	<u>17.09</u>	<u>16.81</u>	<u>18.30</u>	<u>15.62</u>	<u>16.65</u>	<u>14.57</u>	<u>16.59</u>
GPP (ours)	**17.45**	**17.46**	**17.31**	**20.40**	**16.63**	**18.89**	**14.58**	**17.53**

Table 1: PSNR (dB) for compressive image recovery under ideal operating conditions, i.e. the exact measurement operator Φ is known. We observe that GPP performs favourably compared to several commonly used unsupervised priors, particularly with extremely few (1%) measurements. Best performing method is shown in **bold**, and the second best is <u>underlined</u>.

Results: We report the results for 1% and 10% measurement scenarios in table 1. In all the methods, we report PSNR after post-processing with BM3D [12]. For DIP, GPP , and GPP +SC we use $\sigma = 0.25$ for the 10% measurement and $\sigma = 0.5$ for the 1% scenario, for the other baselines we report numbers on 33×33 sized patches from [31]. We see that the two proposed methods, GPP and GPP +SC perform very similarly. As expected, using self-calibration in addition to PGD optimization suffers a little when operating in ideal conditions. Yet, GPP +SC performs on par with many unsupervised baseline methods. Next, we observe that in both the cases GPP outperforms all the competing unsupervised methods. It is particularly noteworthy that for the 10% measurement case, the performance of GPP is higher than patch-based *supervised* methods like ReconNet [31] and SDA [41] (not shown in table) which have an average PSNR of 24.09dB and 23.49dB, respectively. In the 1% measurement case in table 1, we observe that GPP and GPP +SC significantly outperform the state-of-the-art unsupervised baselines considered by nearly $4dB$ indicating the effectiveness of patch-based priors in extreme sensing scenarios. An ablation on the choice of dataset to construct GPP is shown in figure 2. We observe that CIFAR/ImageNet datasets form better priors when compared to MIT Places, perhaps because MIT Places mostly contains low frequency, smooth patches of natural locations, whereas CIFAR contains a lot more high frequency spatial information within each patch.

5.2. Properties of GPP

We examine the components of GPP in order to better understand GPP's behaviour. (a) **Patch sizes:** First, we study the effect of the size of the patches – in a patch sensing setup, as the size of patches become smaller, the number of measurements available in each patch become correspondingly fewer. On the other hand, a larger patch implies fewer patches to compose an image, thereby limiting the details that can be resolved. In the extreme, when the size of the patch is equal to the size of the image, the GPP becomes equivalent to generative priors that have been used recently in inverse problems. In Figure 3(A), we show the average PSNR with varying patch size, as expected we see that a patch size of 32 seems to be optimal, with performance degradations with increasing or decreasing sizes. (b) **Effect of training DCGAN:** Next, we study the importance of training the DCGAN in figure 3(B), where we see that even a partially trained patch-generator can give an improvement over stan-
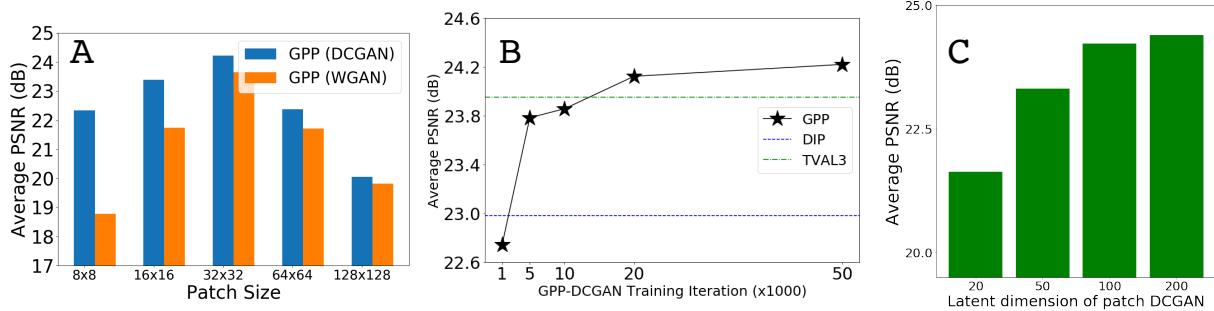
Figure 3: **(A)** Ablation on Patch Size in GPP. PSNR (dB) is shown across two GAN variants DCGAN and Wasserstein GAN (WGAN). **(B)** Comparing quality of reconstruction vs GAN training iterations. We see that even with a partially trained GAN backbone, GPP provides a benefit over the next best performing methods. **(C)** Ablating the affect of latent dimension of the patch DCGAN.

Figure 4: Comparing GPP's performance on CelebA faces with a DCGAN trained on CelebA Faces for compressive image recovery with 50% measurement rate. The GAN prior is effective when low number of measurements are available, but only when the true solution is from the same distribution on which the GAN is trained. On the other hand, GPP applies to any natural image and with more measurements, outperforms the GAN prior because it is able to recover the true image with a much lower projection error. See table 2 for quantitative comparisons. Here we use the same GPP model as before, trained using CIFAR-C images with patches of size 16×16 to recover CelebA faces.

dard baselines. (c) **Latent dimension of the patch generator:** Finally, in figure 3(C) we show how the performance varies with varying dimensionality of the patch-DCGAN. As expected, we observe quality increases with an increased capacity of the generator, indicating that better patch-generators can further improve image recovery, however this comes at a cost of more complexity in training the generator. In all our experiments we use a patch size of 32, and a latent dimensionality of $d = 100$.

GPP and generative image priors As noted earlier, GPP is related to the GAN prior when the number of patches used $N_p = 1$. Here, we study how this affects the performance in compressive image recovery. Since GAN priors, by design, are restricted in their use on specific datasets on which they are trained, we perform this experiment on the commonly used CelebA Faces dataset [35]. Here we train a standard DCGAN on a training set and perform CS recovery on 100 images

from the test set using different number of measurements. We also use the same GPP as before, trained on CIFAR-C images as a comparison on the same CelebA test images. These results are reported in table 2 and we remark on a few observations. First, since these images are faces we report both PSNR and SSIM evaluation metrics. Next, as expected, we find that under very few observations the Image-level generative prior works better simply because it can find a similar face, whereas GPP cannot. On the otherhand, once more observations are available, GPP outperforms the GAN prior on both metrics considered here because it is able to find the *exact* face better than the GAN prior. A few recovered examples are shown in Figure 4.

5.3. Self-Calibrated image recovery from a physical compressive camera

Here we study a real world use case of the self-calibrating mechanism in recovering high quality images from patch-based measurements that obtained us-

8

Meas	GAN Prior	PSNR (dB)	SSIM
5%	Image-level [6, 51]	**19.170**	**0.4752**
	Patch-level *(ours)*	16.770	0.2597
15%	Image-level	**21.367**	**0.5826**
	Patch-level	20.663	0.5294
50%	Image-level	22.913	0.6478
	Patch-level	**26.813**	**0.8261**

Table 2: CS reconstruction on the 100 test images from the CelebA Faces dataset [35]. We compare our patch-based prior trained on CIFAR-C with an image-based prior which is a DCGAN trained on CelebA. As expected, under very few observations the image prior is better because it can find a similar face but this only works when the images are from the same distribution on which the GAN is trained. Yet with more observations GPP outperforms even the GAN prior because it is able to find the exact face.

ing a real world compressive imaging system [27]. We use the measurements collected under a measurement rate of 10%, provided by the authors of [31], [36]. Following [31], the patches are 33×33, and so we resize the output of the generator from 32×32 to match this patch size. As expected, this setup requires manual calibration in order to use existing algorithms out of the box, without which they produce poor reconstructions. We show results over 7 different *un-calibrated* measurements and their corresponding reconstructions in table 3. In order to compute PSNR, we use the recovered image from every algorithm and resize it to 256×256 so it is comparable to the original, and scale it to have a maximum intensity of 1.0, which itself acts as a simple calibration for the baselines. This normalized image is compared to the ground truth to evaluate fidelity. Finally, we process the reconstructions from all the methods using BM3D [12], using $\sigma = 0.25$. In addition to the baselines described previously, we also report results from ISTA-Net[61] which is a supervised, state-of-the-art CS reconstruction technique that also uses 33×33 patches. We also add a "manual calibration" (MC) setting with GPP , where we use these estimates for a and b obtained using the GPP +SC, and use only GPP (without SC) as if these calibration parameters were known. Finally, for this setting we train DIP for 50000 steps before observing convergence, for both the un-calibrated and manually calibrated settings.

Results Table 3 shows performance in terms of PSNR. In these experiments, we estimate the calibration parameters to be $a = 0.075 - 0.085$; $b = 0.0$. The sensor gain a varies for each image, which is expected since every image may require slightly different calibration settings. In the manual calibration setting, we use an average value of $a = 0.08$, $b = 0.0$. We find that GPP +SC is able to successfully recover a high quality image from the un-calibrated measurements, unlike all baseline approaches, resulting in a gain of 4dB over the next best unsupervised method. We also observe that GPP +SC provides better reconstructions when compared to ISTA-Net [61], which is a supervised technique. For reference, we show the manual calibration settings for both GPP , and DIP, and observe that they are only marginally better than GPP +SC. A few qualitative comparisons are shown in figure 5.

5.4. Compressive Phase Retrieval

Finally, we evaluate GPP on the non-linear inverse problem of phase recovery in compressive sensing. We use the same experimental settings as in the linear compressive sensing case. We implement the alternating phase projected gradient descent described (APPGD) in [24], which is shown to have better performance compared to other gradient descent based optimization methods. We compare against the deep image prior (DIP) method, recently used in [26] for phase retrieval and show favourable performance for the 10% measurement case shown in table 4.

6. Discussion

We presented a new kind of deep prior called the generative patch prior (GPP) which uses a GAN trained on patches instead of full sized images as a learned prior for inverse problems. Using GPP, we address the problem of compressive image recovery and show that GPP has many favourable properties compared to existing generative priors – it is more broadly applicable to a wide variety of images not seen during training, or images or arbitrary sizes and aspect ratios. We also see that GPP outperforms several commonly used learned and model priors in compressive sensing and compressive phase retrieval tasks. We also propose a self-calibration mechanism that enables the model to estimate and adjust to calibration artifacts au-

Method	Barbara	Parrot	Lena	Foreman	Cameraman	House	Boats	Avg.
TVAL3 [32]	17.65	12.55	18.05	13.16	15.36	14.29	14.82	15.13
DIP [55, 26]	6.75	7.87	9.13	4.52	8.22	6.15	6.06	6.96
ISTA-Net[61]	**18.95**	16.23	19.19	15.94	**17.11**	18.01	16.14	17.37
GPP (ours)	12.78	12.66	13.50	10.73	13.32	12.77	11.83	12.54
DIP+SC	17.97	18.46	15.86	20.51	19.58	17.64	18.66	18.39
GPP +SC (ours)	18.21	**20.01**	**20.32**	**20.68**	16.33	**20.47**	**18.29**	**19.20**
DIP+Manual calib.	18.88	19.67	19.97	19.59	19.64	19.74	17.57	19.29
GPP +Manual calib.	19.06	20.23	20.43	19.97	19.52	20.22	17.87	19.61

Table 3: **Real CS dataset:** PSNR (dB) is shown for 7 test images obtained using *real, un-calibrated* measurements [36] from a patch-based compressive camera at MR = 10%. The proposed self-calibration improves the quality of recovered images significantly compared with baselines that require manual calibration.
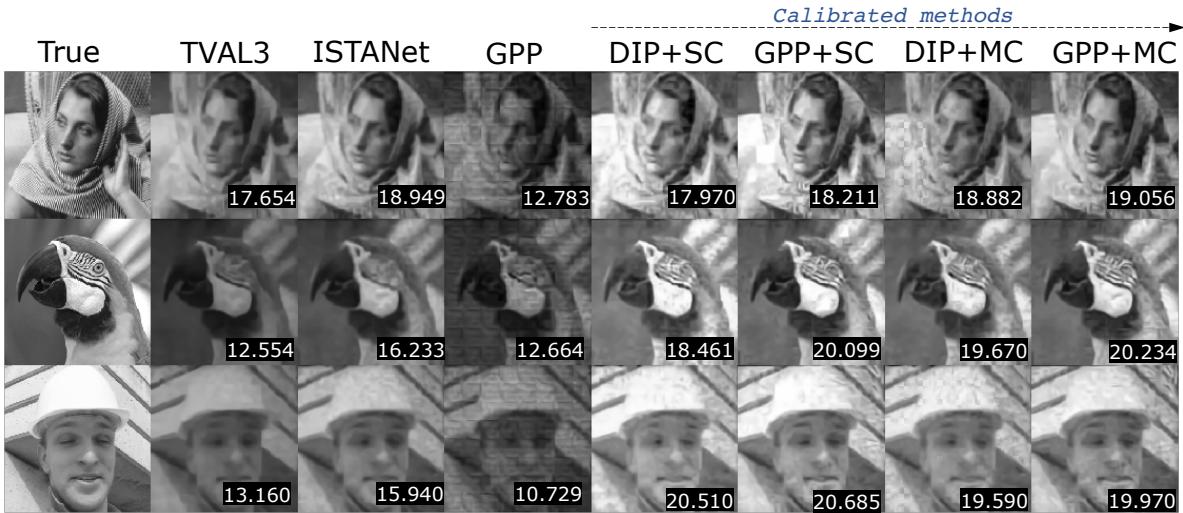
Figure 5: Sample reconstructions on un-calibrated measurements from a real compressive imaging system. PSNR (dB) is shown at the bottom right of each image. MC refers to manual calibration, using parameters obtained using GPP +SC. For the real data, we estimate the unknown calibration parameters to be around $a = 0.08$, $b = 0.0$. We don't show DIP results here because it performs very poorly compared to other baselines.

Method	Avg. PSNR (dB)
DIP [55, 26]	18.439
GPP (ours)	**22.616**

Table 4: Phase Retrieval at a measurement rate of 10%.

tomatically. Finally, we validate this mechanism and the GPP on a real, un-calibrated compressive sensing dataset and demonstrate that the proposed performs well without requiring manual calibration unlike existing methods.

Broader Impact

Applications: This work applies to the compressive sensing (CS) problem, which has been shown to result in faster, cheaper, bandwidth and energy-efficient sensing. As a result, CS methodologies have found applications in a diverse set of applications ranging from MRI, to privacy-preserving sensing, and even for energy efficient visual analytics. While the primary focus of this work is in advancing algorithmic aspects which are broadly applicable, it is applicable to the full spectrum of problems by a simultaneous effort in domain-specific hardware advances.

Implications: As indicated above, the societal im-

plications are strongly tied to the end-application integrated with the specifics of the hardware. There is already existing work on using similar techniques in faster MRI in the healthcare sector. Security and surveillance technology stands to benefit by making analytics robust to under-sampling of visual information; making cameras portable etc. We acknowledge that these approaches, if pushed further, can expand the surveillance capabilities to the point that they may open many ethical questions. While we don't have a solution to prevent misuse, we are aware of its potential. We are, however, motivated by its significant potential for applications in health, basic scientific inquiry, and more broadly energy efficient sensing.

References

[1] R. Anirudh, J. J. Thiagarajan, B. Kailkhura, and P.-T. Bremer. Mimicgan: Robust projection onto image manifolds with corruption mimicking. *International Journal of Computer Vision*, pages 1–19, 2020.

[2] M. Asim, F. Shamshad, and A. Ahmed. Blind image deconvolution using deep generative priors. *arXiv preprint arXiv:1802.04073*, 2018.

[3] R. G. Baraniuk, V. Cevher, M. F. Duarte, and C. Hegde. Model-based compressive sensing. *IEEE Transactions on information theory*, 56(4):1982–2001, 2010.

[4] A. Beck. On the convergence of alternating minimization for convex programming with applications to iteratively reweighted least squares and decomposition schemes. *SIAM Journal on Optimization*, 25(1):185–209, 2015.

[5] L. Boominathan, M. Maniparambil, H. Gupta, R. Baburajan, and K. Mitra. Phase retrieval for fourier ptychography under varying amount of measurements. *arXiv preprint arXiv:1805.03593*, 2018.

[6] A. Bora, A. Jalal, E. Price, and A. G. Dimakis. Compressed sensing using generative models. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 537–546. JMLR. org, 2017.

[7] T. T. Cai, X. Li, Z. Ma, et al. Optimal rates of convergence for noisy sparse phase retrieval via thresholded wirtinger flow. *The Annals of Statistics*, 44(5):2221–2251, 2016.

[8] E. J. Candès, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on information theory*, 52(2):489–509, 2006.

[9] E. J. Candès and M. B. Wakin. An introduction to compressive sampling. *IEEE signal processing magazine*, 25(2):21–30, 2008.

[10] Y. Chi, L. L. Scharf, A. Pezeshki, and A. R. Calderbank. Sensitivity to basis mismatch in compressed sensing. *IEEE Transactions on Signal Processing*, 59(5):2182–2195, 2011.

[11] T. S. Cho, M. Butman, S. Avidan, and W. T. Freeman. The patch transform and its applications to image editing. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008.

[12] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image denoising with block-matching and 3d filtering. In *Image Processing: Algorithms and Systems, Neural Networks, and Machine Learning*, volume 6064, page 606414. International Society for Optics and Photonics, 2006.

[13] W. Dong, G. Shi, X. Li, Y. Ma, and F. Huang. Compressive sensing via nonlocal low-rank regularization. *IEEE Transactions on Image Processing*, 23(8):3618–3632, 2014.

[14] M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image processing*, 15(12):3736–3745, 2006.

[15] W. T. Freeman, E. C. Pasztor, and O. T. Carmichael. Learning low-level vision. *International journal of computer vision*, 40(1):25–47, 2000.

[16] D. Gilton, G. Ongie, and R. Willett. Learned patch-based regularization for inverse problems in imaging. In *2019 IEEE 8th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, pages 211–215, 2019.

[17] 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 (NIPS)*, pages 2672–2680, 2014.

[18] K. Gregor and Y. LeCun. Learning fast approximations of sparse coding. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, pages 399–406, 2010.

[19] M. A. Griswold, P. M. Jakob, R. M. Heidemann, M. Nittka, V. Jellus, J. Wang, B. Kiefer, and A. Haase. Generalized autocalibrating partially parallel acquisitions (grappa). *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine*, 47(6):1202–1210, 2002.

[20] P. Hand and B. Joshi. Global guarantees for blind demodulation with generative priors. In *Advances in Neural Information Processing Systems*, pages 11531–11541, 2019.

[21] P. Hand, O. Leong, and V. Voroninski. Phase retrieval under a generative prior. In *Advances in Neural Information Processing Systems*, pages 9136–9146, 2018.

[22] R. Heckel and P. Hand. Deep decoder: Concise image representations from untrained non-convolutional networks. *arXiv preprint arXiv:1810.03982*, 2018.

[23] D. Hendrycks and T. Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *Proceedings of the International Conference on Learning Representations*, 2019.

[24] R. Hyder, V. Shah, C. Hegde, and M. S. Asif. Alternating phase projected gradient descent with generative priors for solving compressive phase retrieval. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7705–7709. IEEE, 2019.

[25] G. Jagatap and C. Hegde. Fast, sample-efficient algorithms for structured phase retrieval. In *Advances in Neural Information Processing Systems*, pages 4917–4927, 2017.

[26] G. Jagatap and C. Hegde. Algorithmic guarantees for inverse imaging with untrained network priors. 2019.

[27] R. Kerviche, N. Zhu, and A. Ashok. Information-optimal scalable compressive imaging system. In *Computational Optical Sensing and Imaging*, pages CM2D–2. Optical Society of America, 2014.

[28] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations ICLR*, 2014.

[29] D. P. Kingma and M. Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[30] A. Krizhevsky, G. Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[31] K. Kulkarni, S. Lohit, P. Turaga, R. Kerviche, and A. Ashok. Reconnet: Non-iterative reconstruction of images from compressively sensed measurements. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 449–458, 2016.

[32] C. Li, W. Yin, H. Jiang, and Y. Zhang. An efficient augmented lagrangian method with applications to total variation minimization. *Computational Optimization and Applications*, 56(3):507–530, 2013.

[33] S. Ling and T. Strohmer. Self-calibration and biconvex compressive sensing. *Inverse Problems*, 31(11):115002, 2015.

[34] S. Ling and T. Strohmer. Self-calibration and bilinear inverse problems via linear least squares. *SIAM Journal on Imaging Sciences*, 11(1):252–292, 2018.

[35] Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

[36] S. Lohit, K. Kulkarni, R. Kerviche, P. Turaga, and A. Ashok. Convolutional neural networks for noniterative reconstruction of compressively sensed images. *IEEE Transactions on Computational Imaging*, 4(3):326–340, 2018.

[37] S. Lohit, R. Singh, K. Kulkarni, and P. Turaga. Rate-adaptive neural networks for spatial multiplexers. *arXiv preprint arXiv:1809.02850*, 2018.

[38] C. Metzler, A. Mousavi, and R. Baraniuk. Learned d-amp: Principled neural network based compressive image recovery. In *Advances in Neural Information Processing Systems*, pages 1772–1783, 2017.

[39] C. A. Metzler, A. Maleki, and R. G. Baraniuk. From denoising to compressed sensing. *IEEE Transactions on Information Theory*, 62(9):5117–5144, 2016.

[40] C. A. Metzler, P. Schniter, A. Veeraraghavan, and R. G. Baraniuk. prdeep: Robust phase retrieval with a flexible deep network. *arXiv preprint arXiv:1803.00212*, 2018.

[41] A. Mousavi, A. B. Patel, and R. G. Baraniuk. A deep learning approach to structured signal recovery. In *2015 53rd annual allerton conference on communication, control, and computing (Allerton)*, pages 1336–1343. IEEE.

[42] P. Netrapalli, P. Jain, and S. Sanghavi. Phase retrieval using alternating minimization. In *Advances in Neural Information Processing Systems*, pages 2796–2804, 2013.

[43] T. Nguyen, Y. Xue, Y. Li, L. Tian, and G. Nehmetallah. Deep learning approach for fourier ptychography microscopy. *Optics express*, 26(20):26470–26484, 2018.

[44] J. Ni, P. Turaga, V. M. Patel, and R. Chellappa. Example-driven manifold priors for image deconvolution. *IEEE Transactions on Image Processing*, 20(11):3086–3096, 2011.

[45] G. Peyré. Manifold models for signals and images. *Computer vision and image understanding*, 113(2):249–260, 2009.

[46] M. Pollefeys, R. Koch, and L. Van Gool. Self-calibration and metric reconstruction inspite of varying and unknown intrinsic camera parameters. *International Journal of Computer Vision*, 32(1):7–25, 1999.

[47] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *International Conference on Learning Representations (ICLR)*, 2016.

[48] J. Rick Chang, C.-L. Li, B. Poczos, B. Vijaya Kumar, and A. C. Sankaranarayanan. One network to solve them all–solving linear inverse problems using deep projection models. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5888–5897, 2017.

[49] Y. Rivenson, Y. Zhang, H. Günaydın, D. Teng, and A. Ozcan. Phase recovery and holographic image reconstruction using deep learning in neural networks. *Light: Science & Applications*, 7(2):17141–17141, 2018.

[50] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

[51] V. Shah and C. 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)*, pages 4609–4613. IEEE, 2018.

[52] T. R. Shaham, T. Dekel, and T. Michaeli. Singan: Learning a generative model from a single natural image. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4570–4580, 2019.

[53] F. Shamshad, F. Abbas, and A. Ahmed. Deep ptych: Subsampled fourier ptychography using generative priors. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7720–7724. IEEE, 2019.

[54] J. Sun, H. Li, Z. Xu, et al. Deep admm-net for compressive sensing mri. In *Advances in neural information processing systems*, pages 10–18, 2016.

[55] D. Ulyanov, A. Vedaldi, and V. Lempitsky. Deep image prior. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9446–9454, 2018.

[56] M. B. Wakin, J. N. Laska, M. F. Duarte, D. Baron, S. Sarvotham, D. Takhar, K. F. Kelly, and R. G. Baraniuk. An architecture for compressive imaging. In *2006 International Conference on Image Processing*, pages 1273–1276. IEEE, 2006.

[57] G. Wang, L. Zhang, G. B. Giannakis, M. Akçakaya, and J. Chen. Sparse phase retrieval via truncated amplitude flow. *IEEE Transactions on Signal Processing*, 66(2):479–491, 2017.

[58] Z. Xia and A. Chakrabarti. Training image estimators without image ground truth. In *Advances in Neural Information Processing Systems*, pages 2436–2446, 2019.

[59] R. A. Yeh, C. Chen, T. Y. Lim, A. G. Schwing, M. Hasegawa-Johnson, and M. N. Do. Semantic image inpainting with deep generative models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5485–5493, 2017.

[60] T. Yokota, H. Hontani, Q. Zhao, and A. Cichocki. Manifold modeling in embedded space: A perspective for interpreting" deep image prior". *arXiv preprint arXiv:1908.02995*, 2019.

[61] J. Zhang and B. Ghanem. Ista-net: Interpretable optimization-inspired deep network for image compressive sensing. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1828–1837, 2018.

[62] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning deep features for scene recognition using places database. In *Advances in neural information processing systems*, pages 487–495, 2014.

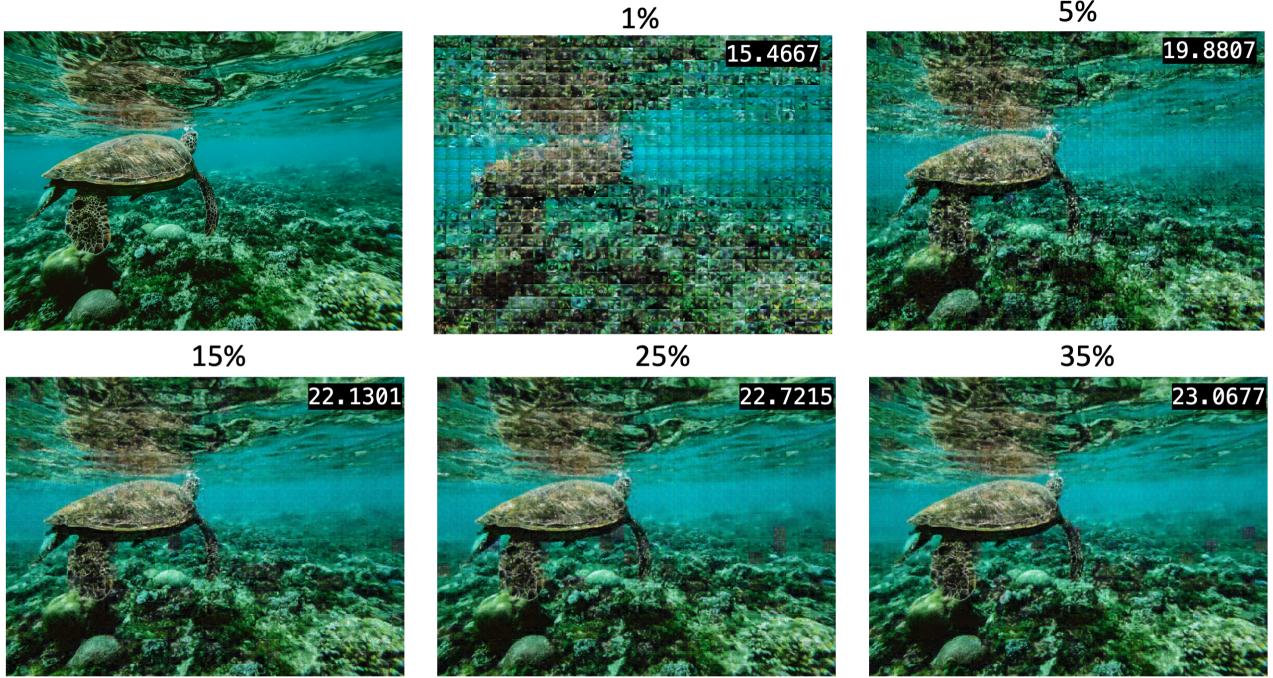## Acknowledgements

Disclaimer

Appendix A. Additional results

Figure 6: GPP degrades gracefully as the number of available observations are reduced. Results shown here are for an image of size 1024×768, recovered using patches of size 32×32. We do not use BM3D here to illustrate the patch artifacts under very few observations (1%). The PSNR (dB) is also shown along with the reconstruction, compared to the ground truth which is shown in the top left.

In figure 7, we show sample reconstructions for the phase retrieval task at a measurement rate of 10%.

Phase Retrieval at a Measurement Rate of 10%

Figure 7: Compressive phase retrieval sensing at a measurement rate of 10%.

A.1. Self calibration under unknown sensor shift

In figure 8 we illustrate how reconstruction methods can easily fail to recover the solution when there is even a small shift in the operator. We simulate this using $b = -0.25$ and compare the proposed self calibration approach against no calibration and the untrained network prior (DIP) [55]. We observe that the self calibration is able to successfully correct for the unknown shift, compared to the models that do not account for it.
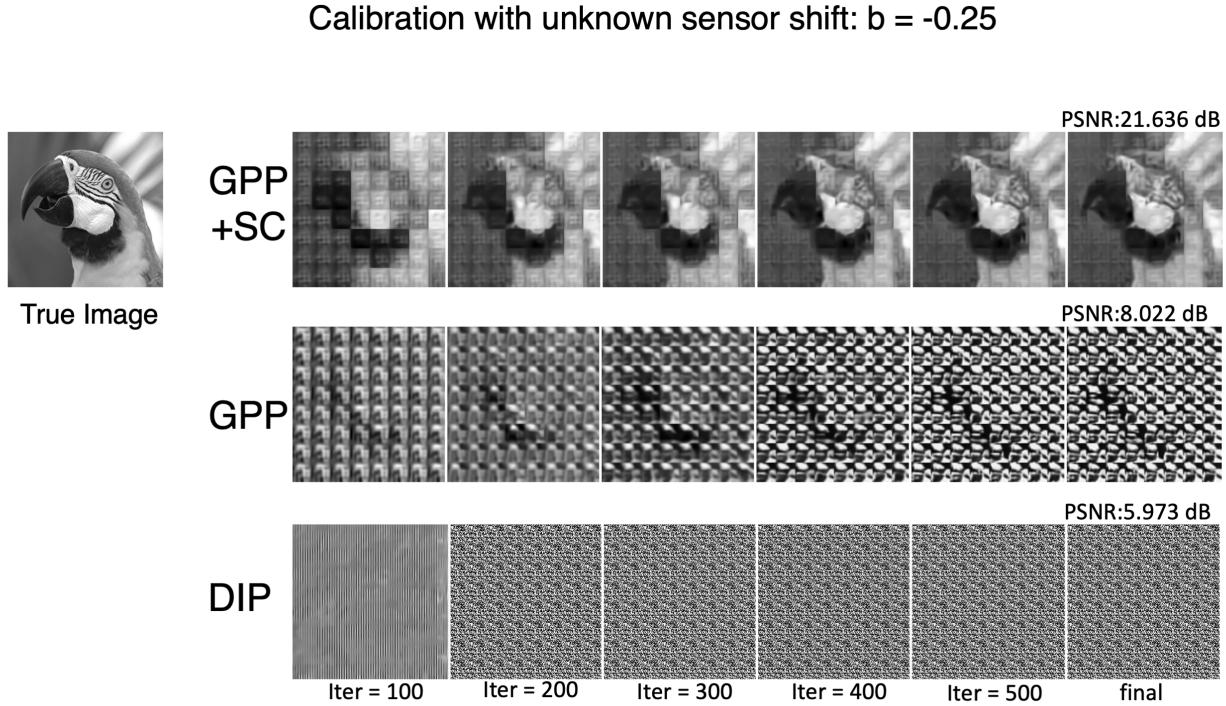

Figure 8: Compressive sensing at a measurement rate of 10% under unknown sensor shift (b). We see that methods that do not account for this shift can easily break. Note the iterations in DIP are scaled since we run it for 10000 iterations compared to 1000 iterations on GPP and GPP + SC.

Appendix B. Self-Calibrated Compressive Image Recovery

We evaluate the robustness of GPP using the proposed self-calibration (SC) step. In this experiment, we perturb the measurement operator using the perturbation model described in section 4 of the main paper, $\tilde{\Phi} = a * \Phi + b$ using different values for a and b. The measurements are then obtained by $\mathbf{y} = \tilde{\Phi}\mathbf{x}$, but all the reconstruction algorithms, including ours, are given access to only Φ. We study the average PSNR for the seven test images used earlier, for different values of a, and b. In figure 9a, we vary the gain coefficient a, while keeping b fixed at 0.0. We observe that GPP +SC remains robust to a wide variation of a, while the un-calibrated setup completely fails. We repeat these experiments for the sensor shift b coefficient in figure 9b where $a = 1.0$, and we vary b. We observe similarly that GPP +SC is significantly more robust than GPP , or DIP alone. Finally, in figure 9c, we study the convergence of the calibration algorithm for a mixed case with $a = 0.85, b = 0.5$. We see that the self-calibration step converges quickly to a value very close to the true values, and correspondingly improving the PSNR of the reconstruction.
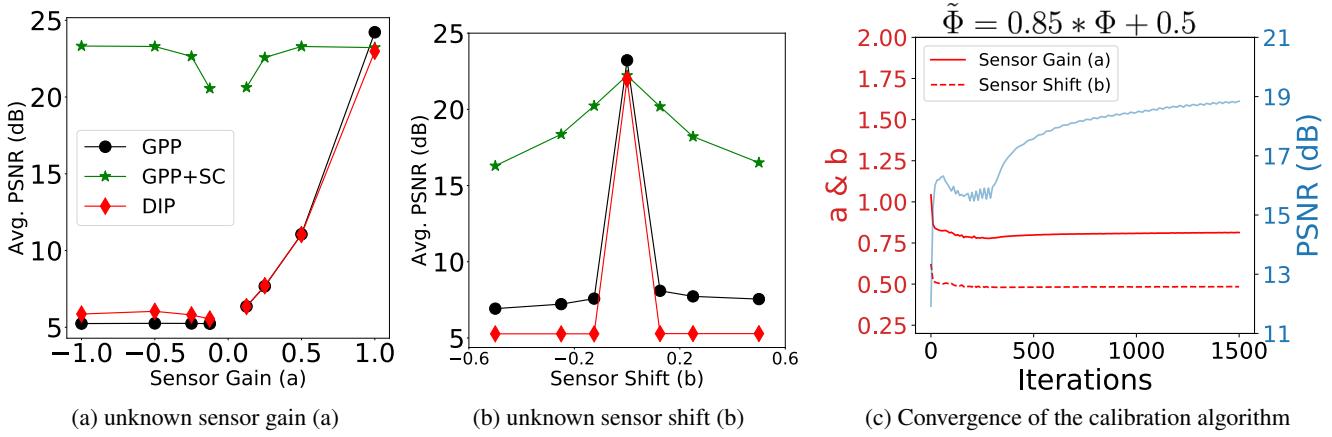
16

Figure 9: Calibration experiments

Appendix C. Derivation for a^* and b^*

Consider a vectorized square block of an image $\mathbf{x} \in \mathcal{X} \subset \mathbb{R}^n$ which we want to sense, and denote by $\mathbf{y} \in \mathbb{R}^m$ the compressive measurements obtained by the sensor. Given a measurement matrix $\Phi \in \mathbb{R}^{m \times n}$, with $m < n$ and $\Phi_{i,j} \sim \mathcal{N}(0,1)$, the compressive recovery problem is to estimate $\mathbf{x}$ accurately from $\mathbf{y}$. In the ideal setting, i.e., compressive sensing with known calibration the sensing model is given by $\mathbf{y} = \Phi\mathbf{x}$. Instead we consider a simple calibration model— $\mathbf{y} = (a\Phi + b\mathbf{1})\mathbf{x}$, where $a, b \in \mathbb{R}^1$ are unknown calibration parameters and have to be estimated, and $\mathbf{1} \in \mathbb{R}^{m \times n}$ is a matrix of the same size as Φ with 1s.

In order to derive a and b, we assume we have a current estimate of the solution $\mathbf{x}$ from a pre-trained generator $\mathcal{G}(\mathbf{z})$ for a latent vector $\mathbf{z}$. This is be randomly initialized at the beginning. Under this calibration model, let us define mean squared error loss function as follows:

$$\mathcal{L} = (\mathbf{y} - (a\Phi + b\mathbf{1})\mathbf{x})^\mathsf{T}(\mathbf{y} - (a\Phi + b\mathbf{1})\mathbf{x})$$
$$\implies \mathcal{L} = \mathbf{y}^\mathsf{T}\mathbf{y} - \mathbf{y}^\mathsf{T}(a\Phi + b\mathbf{1})\mathbf{x} - \mathbf{x}^\mathsf{T}(a\Phi + b\mathbf{1})^\mathsf{T}\mathbf{y} + \mathbf{x}^\mathsf{T}(a\Phi + b\mathbf{1})^\mathsf{T}(a\Phi + b\mathbf{1})\mathbf{x} \quad (4)$$

As a result, the derivatives with respect to each unknown a, b are:

$$\frac{\partial \mathcal{L}}{\partial a} = -\mathbf{y}^\mathsf{T}\Phi\mathbf{x} - \mathbf{x}^T\Phi^\mathsf{T}\mathbf{y} + \mathbf{x}^\mathsf{T}\left[2a\Phi^\mathsf{T}\Phi + b\Phi^\mathsf{T}\mathbf{1} + b\mathbf{1}^\mathsf{T}\Phi\right]\mathbf{x}$$
$$\text{Similarly,} \frac{\partial \mathcal{L}}{\partial b} = -\mathbf{y}^\mathsf{T}\mathbf{1}\mathbf{x} - \mathbf{x}^\mathsf{T}\mathbf{1}^\mathsf{T}\mathbf{y} + \mathbf{x}^\mathsf{T}\left[a\Phi^\mathsf{T}\mathbf{1} + \mathbf{1}^\mathsf{T}\Phi + 2b\mathbf{1}^\mathsf{T}\mathbf{1}\right]\mathbf{x} \quad (5)$$

By setting these derivatives to zero, we get:

$$\frac{\partial \mathcal{L}}{\partial a} = 0 \implies -\mathbf{y}^\mathsf{T}\Phi\mathbf{x} - \mathbf{x}^\mathsf{T}\Phi^\mathsf{T}\mathbf{y} + 2a\mathbf{x}^\mathsf{T}\Phi^\mathsf{T}\Phi\mathbf{x} + b\mathbf{x}^\mathsf{T}\Phi^\mathsf{T}\mathbf{1}\mathbf{x} + b\mathbf{x}^\mathsf{T}\mathbf{1}^\mathsf{T}\Phi\mathbf{x} = 0. \quad (6)$$

$$\implies -2\mathbf{y}^\mathsf{T}\Phi\mathbf{x} + 2a\mathbf{x}^\mathsf{T}\Phi^\mathsf{T}\Phi\mathbf{x} + 2b\mathbf{x}^\mathsf{T}\Phi^\mathsf{T}\mathbf{1}\mathbf{x} = 0. \quad (7)$$

$$\implies b = \frac{\mathbf{y}^\mathsf{T}\Phi\mathbf{x} - a\mathbf{x}^\mathsf{T}\Phi^\mathsf{T}\Phi\mathbf{x}}{\mathbf{x}^\mathsf{T}\Phi^\mathsf{T}\mathbf{1}\mathbf{x}} \quad (8)$$

Note, in (7) all the terms are scalars and therefore $\mathbf{y}^\mathsf{T}\Phi\mathbf{x} = \mathbf{x}^\mathsf{T}\Phi^\mathsf{T}\mathbf{y}$ etc. Next, we take the partial derivative with

17

respect to b.

$$\frac{\partial \mathcal{L}}{\partial b} = 0 \implies \mathbf{y}^\mathsf{T}\mathbf{1x} - \mathbf{x}^\mathsf{T}\mathbf{1}^\mathsf{T}\mathbf{y} + a\mathbf{x}^\mathsf{T}\Phi^\mathsf{T}\mathbf{1x} + a\mathbf{x}^\mathsf{T}\mathbf{1}^\mathsf{T}\Phi\mathbf{x} + 2b\mathbf{x}^\mathsf{T}\mathbf{1}^\mathsf{T}\mathbf{1x} = 0. \tag{9}$$

$$\implies -2\mathbf{y}^\mathsf{T}\mathbf{1x} + 2a\mathbf{x}^\mathsf{T}\Phi^\mathsf{T}\mathbf{1x} + 2b\mathbf{x}^\mathsf{T}\mathbf{1}^\mathsf{T}\mathbf{1x} = 0. \tag{10}$$

$$\implies b = \frac{\mathbf{y}^T\mathbf{1x} - a\mathbf{x}^\mathsf{T}\Phi^\mathsf{T}\mathbf{1x}}{\mathbf{x}^\mathsf{T}\mathbf{1}^\mathsf{T}\mathbf{1x}} \tag{11}$$

Combining equations (11) and (8), we get the following:

$$\left(\mathbf{y}^\mathsf{T}\Phi\mathbf{x} - a\mathbf{x}^\mathsf{T}\Phi^\mathsf{T}\Phi\mathbf{x}\right)\mathbf{x}^\mathsf{T}\mathbf{1}^\mathsf{T}\mathbf{1x} = \left(\mathbf{y}^T\mathbf{1x} - a\mathbf{x}^\mathsf{T}\Phi^\mathsf{T}\mathbf{1x}\right)\mathbf{x}^\mathsf{T}\Phi^\mathsf{T}\mathbf{1x} \tag{12}$$

As in the paper, let us define scalar quantities for notational convenience: $c_\Phi = \mathbf{y}^T\Phi\mathbf{x}, c_1 = \mathbf{y}^T\mathbf{1x}, \theta_\Phi = (\Phi\mathbf{x})^T(\Phi\mathbf{x}), \theta_1 = (\mathbf{1x})^T(\mathbf{1x}), \lambda = (\Phi\mathbf{x})^T(\mathbf{1x})$. This implies, (12) is now reformulated as:

$$(c_\Phi - a\theta_\Phi)\theta_1 = (c_1 - a\lambda)\lambda \tag{13}$$

$$\implies a = \frac{c_1\lambda - c_\Phi\theta_1}{\lambda^2 - \theta_\Phi\theta_1} \tag{14}$$

$$\text{and } b = \frac{c_1 - a^*\lambda}{\theta_1} \tag{15}$$

In each step of the alternating minimization, we use the estimates from (14), and (15) and update the latent vector $\mathbf{z}$, which is repeated until convergence in $a, b, \mathbf{z}$. Since our generative model is defined at a patch level, we estimate a, b for each individual patch i separately and assign the mean values of all the patch-estimates as the single a, b for the entire image: $a = \frac{1}{N}\sum_{i=1}^{N} a_i$; and $b = \frac{1}{N}\sum_{i=1}^{N} b_i$. We continue with this alternating minimization until the loss converges. In practice, we find that the algorithm converges within 1500 iterations, and finding recovery and convergence properties of this algorithm remain part of our future work. We empirically study convergence properties of this self-calibrating mechanism under different settings in the supplement.

Note that the latent space optimization is itself a nonconvex optimization problem which is solved only approximately using a gradient-descent type of optimization, yielding a local minimum at each iteration. Therefore the overall optimization problem is nonconvex even though the calibration parameters can be estimated exactly at each step. Using the result in Theorem 3.1 and Lemma 3.2 in the paper by beck [4], we can see that the alternating procedure converges to a stationary point. A stronger result is deferred for future work.

C.1. GPP for image inpainting

GPP is a generic prior to constrain solutions to the natural image manifold. We show an example here of how it can be used in other challenging inverse problems. In figure 10, we illustrate the efficiency of GPP for a for inpainting, where only a small number random pixels are shown, and the task is to recover the original image. Unlike most existing methods, we see that GPP's solution degrades more gracefully than DIP, even recovering some signal when 99.5% of the pixels are missing.
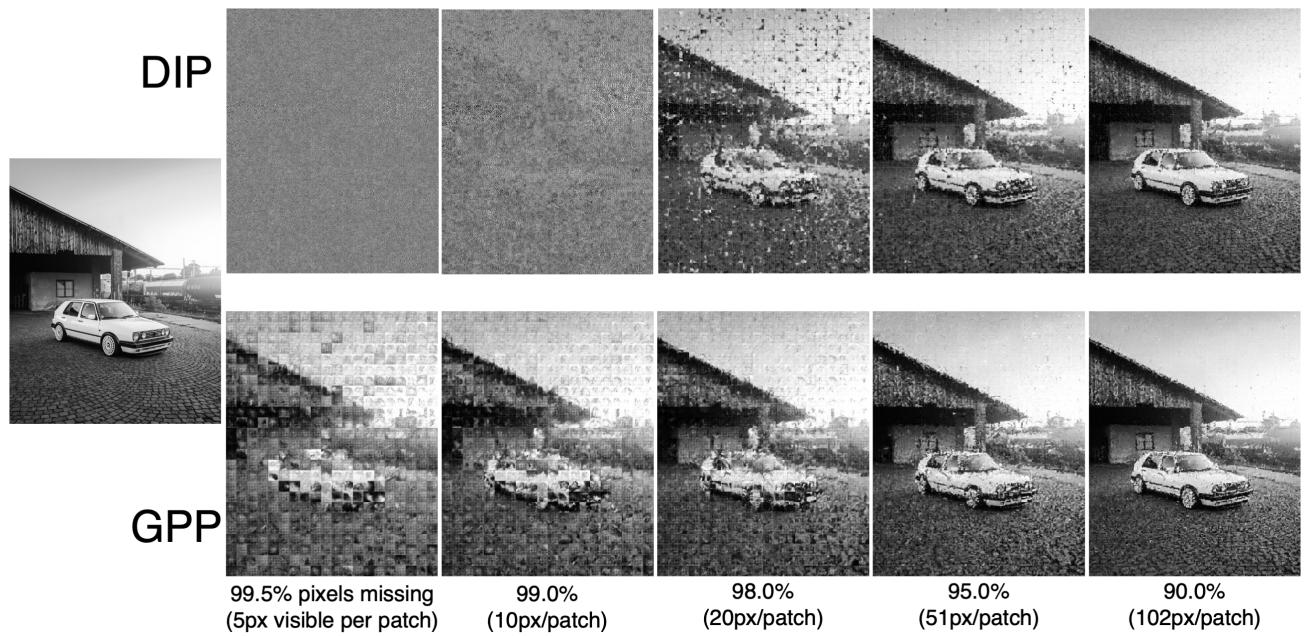
Figure 10: **GPP for image inpainting:** GPP is more efficient than DIP, being able to recover parts of original image even when 99.5 % of the pixels are missing. The original image is of size 800×640.