
Intermediate Layer Optimization for Inverse Problems using Deep Generative Models

Giannis Daras ^{*1} Joseph Dean ^{*1} Ajil Jalal ¹ Alexandros G. Dimakis ¹

Abstract

We propose Intermediate Layer Optimization (**ILO**), a novel optimization algorithm for solving inverse problems with deep generative models. Instead of optimizing only over the initial latent code, we progressively change the input layer obtaining successively more expressive generators. To explore the higher dimensional spaces, our method searches for latent codes that lie within a small l_1 ball around the manifold induced by the previous layer. Our theoretical analysis shows that by keeping the radius of the ball relatively small, we can improve the established error bound for compressed sensing with deep generative models. We empirically show that our approach outperforms state-of-the-art methods introduced in StyleGAN-2 and PULSE for a wide range of inverse problems including inpainting, denoising, super-resolution and compressed sensing.

1. Introduction

We study how deep generators can be used as priors to solve inverse problems like inpainting, super-resolution, denoising and compressed sensing from random projections. Image reconstruction methods can be either supervised (Pathak et al., 2016; Richardson et al., 2020; Yu et al., 2018) or unsupervised (Menon et al., 2020; Bora et al., 2017; Pajot et al., 2019), see the recent survey (Ongie et al., 2020) for a unified presentation. Such inverse problems naturally appear in many applications including medical imaging, single pixel reconstruction and other domains (Lustig et al., 2007; 2008; Chen et al., 2008; Duarte et al., 2008; Qaisar et al., 2013; Hegde et al., 2009).

We focus on unsupervised image reconstruction techniques

^{*}Equal contribution ¹The University of Texas at Austin. Correspondence to: Giannis Daras <giannisdaras@utexas.edu>, Joseph Dean <josephdean98@utexas.edu>, Ajil Jalal <ajiljalal@utexas.edu>, Alexandros G. Dimakis <dimakis@austin.utexas.edu>.

that rely on a pre-trained generator, building on the general framework introduced in CSGM (Bora et al., 2017). The central optimization problem that appears in unsupervised image reconstruction is the inversion of a deep generative model, i.e. finding a latent code that explains the measurements. This can be performed for different generators, e.g. DCGAN or more recently the powerful StyleGAN-2 (Karras et al., 2019; 2020) as shown in the excellent results obtained by PULSE (Menon et al., 2020). Unfortunately, inverting a generator with even 4 layers is NP-hard (Lei et al., 2019) so approximate inversion methods are needed.

The CSGM framework (Bora et al., 2017) used gradient descent to minimize the measurement mean squared error (MSE) and showed good empirical performance for numerous inverse problems including inpainting and compressed sensing with random Gaussian measurements using DC-GAN. However, this *does not* work as well for deeper generators e.g. BigGAN as discussed in Daras et al. (2020). PULSE (Menon et al., 2020) improved the CSGM framework focusing specifically on super-resolution, by refining the latent space optimization and using the StyleGAN-2 (Karras et al., 2019; 2020) generator.

We propose a novel optimization method for solving general inverse problems using a technique we call **Intermediate Layer Optimization** (**ILO**). Our method adaptively changes which layer is optimized, moving from the initial latent code to intermediate layers closer to the pixels. By optimizing intermediate layers we expand the range of the generator to better satisfy the measurements. This has to be done carefully since intermediate layers can produce non-realistic images and therefore inversion must be regularized.

1.1. Our Contributions

1. We propose a novel optimization method for solving general inverse problems by adaptively changing which layer variables are optimized. Our method extends PULSE (Menon et al., 2020) beyond super-resolution, to all inverse problems with differentiable forward operators.
2. To avoid over-expanding the range of the generator to non-realistic images, we only search for latent codes within a small l_1 ball around the manifold induced by the previous layer. Conceptually, our method generalizes the framework

introduced in Dhar et al. (2018); instead of allowing sparse deviations only in the image space, we allow small deviations from the manifold of any layer of the generator.

3. We theoretically analyze our framework by establishing sample complexity and error bounds. We show that by restricting the radius of the latent searches, we can improve the established error bound of CSGM (Bora et al., 2017).
 4. Experimentally, our method significantly outperforms the previous state-of-the-art techniques for solving inverse problems with deep generative models for a wide range of tasks including inpainting, denoising and super-resolution.
 5. To illustrate the power of inverse problems with general differentiable forward operators, we use a classifier as a measurement process. Specifically, we show how we can use a classifier to bias generators to produce human images that look like ImageNet classes like frogs, corals and gold-fishes. Our method uses gradients from classifiers trained to achieve robustness to adversarial attacks as proposed in (Santurkar et al., 2019), but guiding generative latent codes as opposed to pixels directly.

2. Algorithm

2.1. Setting

The key step in our approach is to decompose pre-trained generative models as compositions of feed-forward neural networks. Given a (pre-trained) generative model $G(z) \in \mathbb{R}^n$ that produces images from latent codes $z \in \mathbb{R}^k$, we decompose it as a $G = G_2 \circ G_1$ where $G_1 : \mathbb{R}^k \rightarrow \mathbb{R}^p$ and $G_2 : \mathbb{R}^p \rightarrow \mathbb{R}^n$. As usual, the latent vectors $z^k \in \mathbb{R}^k$ were sampled according to a simple distribution P_z , typically Gaussian and independent.

Our observations are formed by a known measurement matrix

$$y = Ax + \text{noise}, \quad (1)$$

where $A : \mathbb{R}^{m \times n}$ and $x \in \mathbb{R}^n$ is the real image we want to recover. We emphasize that our algorithm can be applied when the measurement process is a general differentiable operator $y = \mathcal{A}(x)$ but our theory only applies to linear inverse problems. Since we will be working with latent vectors in different layers we indicate the dimension as a superscript, so z^k denotes an initial latent vector in \mathbb{R}^k and z^p an intermediate vector in \mathbb{R}^p .

2.2. Approach

Our approach is described in Algorithm 1. The first step of our method is the same as in CSGM (Bora et al., 2017); we optimize over a k -dimensional latent code, z^k , which is the input of the first layer of the generator. In practice, to obtain the solution of line 1 of Algorithm 1, we pick an initial z^k from the latent distribution of the generator

and we optimize the loss function $\|AG(z^k) - Ax\|$ using gradient descent. Once we solve this optimization problem, we obtain a solution, \hat{z}^k , that we map to the p -dimensional space using G_1 . By doing that, we get an intermediate latent representation, $\hat{z}^p = G_1(\hat{z}^k)$.

From that point onwards, our algorithm proceeds in rounds. At the beginning of each round, we optimize on the p -dimensional input space of G_2 but we only allow solutions that lie within an l_1 ball centered at \hat{z}^p . Intuitively, we allow deviations from the range of G_1 to increase the expressivity of the model, but we restrict those deviations to avoid overfitting on the measurements (see Experiments section).

Once we obtain the solution of line 4 of Algorithm 1, i.e. once we find the latent code, \tilde{z}^p , that best explains the measurements and lies inside an l_1 ball of the previous latent, we project this solution back to the range of the generator. To do that, we search for the latent code z^k such that $G_1(z^k)$ is as close as possible to \tilde{z}^p (line 5 of Algorithm 1). This problem is solved by initializing a latent vector z^p to \tilde{z}^p and then minimizing using gradient descent the loss $\|G_1(z^k) - \tilde{z}^p\|$. The solution of this problem forms a new \hat{z}^k vector which is in turn projected again to the intermediate code $\hat{z}^p = G_1(\hat{z}^k)$. Our algorithm attempts to explore the set we call the *extended range*: the range of vectors realizable by the previous layer, dilated by an l_1 ball of sparse deviations. Within this set we would like to find the latent vector that best explains the measurements.

We emphasize that our theoretical analysis provides performance bounds for the global optimum in this extended range, while our algorithm is based on projected gradient descent for a non-convex problem and therefore can be stuck in local optima. It may be possible to prove that such local optimization algorithms obtain global minima under generator weight assumptions as achieved in the pioneering work of Hand & Voroninski (2018); Hand et al. (2018) for CSGM, but this remains open for future work.

3. Theoretical Analysis

3.1. Preliminaries

We begin our theoretical discussion by revisiting some important elements of the theory of compressed sensing with deep generative models.

Definition 1 (S-REC (Bora et al., 2017)). *Let $S \subseteq \mathbb{R}^n$. For some parameters $\gamma, \delta > 0$, a matrix $A \in \mathbb{R}^{m \times n}$ is said to satisfy S-REC(S, γ, δ) if $\forall x_1, x_2 \in S$, we have that:*

$$\|A(x_1 - x_2)\|_2 \geq \gamma \|x_1 - x_2\|_2 - \delta. \quad (2)$$

The S-REC condition, introduced in CSGM (Bora et al., 2017), guarantees that if two vectors, $x_1, x_2 \in \mathbb{R}^n$, are very different (right side of the equation), then their measure-

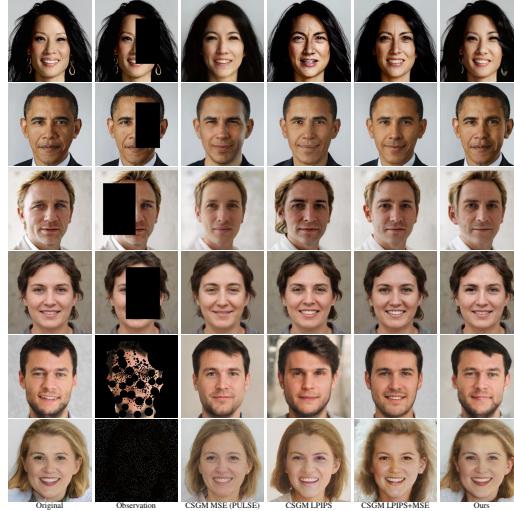


Figure 1. Results on the inpainting task. Rows 1, 2, 3 and 5 are real images (outside of the test set, collected from the web) while rows 4, 6 are StyleGAN-2 generated images. Column 2: the first five images have masks that were chosen to remove important facial features. The last row is an example of randomized inpainting, i.e. a random 1% of the total pixels is observed. Columns 3-5: reconstructions using the CSGM (Bora et al., 2017) algorithm with the StyleGAN-2 generator and the optimization setting described in PULSE (Menon et al., 2020). While PULSE only applies to super-resolution, we extend it using MSE, LPIPS and jointly MSE+LPIPS loss. The experiments of Columns 3-5 form an ablation study of the benefits of each loss function. Column 6: reconstructions with ILO (ours). As shown, ILO consistently gives better reconstructions of the original image. Also, many biased reconstructions can be corrected by our method. In the last two rows, recovery of the image is still possible from very few pixel observations using our method.

ments will be significantly different as well (left side of the equation). In CSGM, the set S of interest is the range of the generator. Therefore, S-REC is a key property for proving small reconstruction error when observing Ax . Bora et al. (2017) show that, for any L -Lipschitz generator $G : \mathbb{R}^k \rightarrow \mathbb{R}^n$, if 1) A is a matrix with i.i.d. Gaussian entries drawn from $\mathcal{N}(0, \frac{1}{m})$ and 2) $m = \frac{1}{a^2} \Omega(k \log(\frac{Lr_1}{\delta}))$, then with probability $1 - e^{-a^2 \Omega(m)}$, $\text{S-REC}(G(B_1^k(r_1)), 1 - a, \delta)$ is satisfied.

Algorithm 1 ILO for one layer of the generator

```

// CSGM solution
1  $\hat{z}^k \leftarrow \operatorname{argmin}_{z^k \in B_2^k(r_1)} \|AG(z^k) - Ax\|_2$ 
2  $\hat{z}^p \leftarrow G_1(\hat{z}^k)$ 
3 for  $t \leftarrow 0$  to  $r$  do
    // Best solution within an  $l_1$  ball
    // centered around the prev. solution
4      $\hat{z}^p \leftarrow \operatorname{argmin}_{z^p \in \hat{z}^p \oplus B_1^p(r_2)} \|AG_2(z^p) - Ax\|$ 
    // Projection back to the range
5      $\hat{z}^k \leftarrow \operatorname{argmin}_{z^k \in B_2^k(r_1)} \|G_1(z^k) - \hat{z}^p\|$ 
6      $\hat{z}^p \leftarrow G_1(\hat{z}^k)$ 
end
// Return the best solution within an  $l_1$ 
// ball of some point in the range
7 return  $G_2(\hat{z}^p)$ 

```

3.2. Intermediate Layer Optimization

Our theoretical result is a sample complexity bound for the reconstruction algorithm that optimizes in the full extended range of the generative model, similar in style to the CSGM (Bora et al., 2017) result.

Let $B_q^k(r_1)$ denote a ball of radius r_1 measured in l_q norm and \oplus denote the Minkowski sum operation, i.e. given sets S_1, S_2 , the set

$$S_1 \oplus S_2 = \{x + y | x \in S_1, y \in S_2\}.$$

If the initial vector z^k lies in a ball of radius r_1 , denoted as $B_2^k(r_1)$, the range of the first generator is $G_1(B_2^k(r_1))$. We are expanding this set to create the *extended range*:

$$G_1(B_2^k(r_1)) \oplus B_1^p(r_2).$$

Our result is showing that minimizing the measurements in this extended range gives a reconstruction that is close to the best reconstruction that the extended generator G_2 can produce. This result is obtained with high probability over the random measurement matrix A , if the number of measurements is sufficiently large:

Theorem 1. Let $G = G_2 \circ G_1$ with $G_1 : \mathbb{R}^k \rightarrow \mathbb{R}^p$ be an L_1 -Lipschitz function and $G_2 : \mathbb{R}^p \rightarrow \mathbb{R}^n$ be an L_2 -Lipschitz function. Let $A \in \mathbb{R}^{m \times n}$ be the measurements matrix with $A_{ij} \sim \mathcal{N}(0, 1/m)$ i.i.d. entries.

Let K be a parameter of our choice where $K \leq \sqrt{p}$, and $r_2 = \frac{K\delta}{L_2}$. Consider the true optimum in the extended range

$$\bar{z}^p = \operatorname{argmin}_{z^p \in G_1(B_2^k(r_1)) \oplus B_1^p(r_2)} \|x - G_2(z^p)\|, \quad (3)$$

and the measurements optimum in the extended range

$$\tilde{z}^p = \operatorname{argmin}_{z^p \in G_1(B_2^k(r_1)) \oplus B_1^p(r_2)} \|Ax - AG_2(z^p)\|. \quad (4)$$

Then, if the number of measurements is sufficiently large:

$$m = \frac{1}{(1-\gamma)^2} \Omega \left(k \log \frac{L_1 L_2 r_1}{\delta} + K^2 \log p \right), \quad (5)$$

then with probability at least $1 - e^{-\Omega((1-\gamma)^2 \cdot m)}$, we have the following error bound:

$$\begin{aligned} \|x - G_2(\tilde{z}^p)\| &\leq \left(1 + \frac{4}{\gamma}\right) \|x - G_2(\bar{z}^p)\| \\ &+ \delta \cdot \frac{\log(4K)}{\gamma} \cdot \frac{\sqrt{p}}{K} \log \frac{\sqrt{p}}{K}. \end{aligned} \quad (6)$$

We will now try to develop intuition about the theorem. We begin by explaining the sets involved in Equations (3), (4). We consider $B_2^k(r_1)$ to be a set containing all the latent codes of the first layer of the generator that could be potentially pre-images of any sensed signal x . We refer to $B_2^k(r_1)$ as the domain of G and to $G_1(B_2^k(r_1))$ as the range of G_1 . The *extended range* contains all vectors that lie within an l_1 ball of radius r_2 from some point in the range of G_1 . This is the set $G_1(B_2^k(r_1)) \oplus B_1^p(r_2)$ that we optimize over.

Let's now consider the error bound of (6). First, \bar{z}^p is the latent code in the extended range that best explains the image x . We refer to this as the true optimum latent code. Next, \tilde{z}^p is the measurements optimum, i.e. the latent code in the extended range that best explains the measurements Ax . It is important to realize that a reconstruction algorithm only has access to this measurement error and can never compute \bar{z}^p . Our goal is to show that \tilde{z}^p produces an image close to the one produced by \bar{z}^p .

Our theorem states that given enough measurements m , the measurements optimum is nearly as good as the true optimum (see (6) and Remark 3).

Remark 1 (Choice of K). *The size of the extended range affects the required number of measurements ((4)) and our error bound (see (6)). Observe that the size of the extended range is directly controlled by K , since, for any fixed δ , we set $r_2 = \frac{K\delta}{L_2}$. As K increases, we explore a bigger set and both terms on the right side of (6) become smaller. However, measurements scale quadratically with K . We can set K to scale approximately as \sqrt{k} (see Remark 3 for details on*

how all the quantities can scale). For that choice of K , observe that our result requires measurements that scale linearly on k (and only logarithmic in p) while the CSGM result requires measurements that scale linearly on p . The costs for the small increase in the measurements, are 1) the additive error scales with \sqrt{p} , 2) we are restricted to exploring a small radius.

In practice, these can be tuned as hyperparameters and our experiments show that even small expansions significantly outperform CSGM in numerous inverse problems.

Remark 2 (CSGM sample bound applied directly on the intermediate layer). *We compare to the result we obtain by applying CSGM to the intermediate layer generator. That would yield measurements that scale as:*

$$m = \Omega \left(k \log \left(\frac{L_1 L_2 r_1}{\delta} \right) + p \log \left(\frac{L_2 r_2}{\delta} \right) \right).$$

These many measurements result in an additive error term of $O(\delta)$. Our new bound requires fewer measurements when the free parameter K is smaller than \sqrt{p} .

Remark 3 (Parameter Scaling). *There are various ways to set the parameters in our bounds, depending on the scaling of sizes of the intermediate layers and the Lipschitz constants. For typical piecewise linear networks with d layers and maximum n neurons in each layer, we know that the end-to-end Lipschitz constant $L \leq L_1 \cdot L_2$ might scale as n^d for bounded maximum weights. Hence, as in CSGM, we may set r_1 to scale as n^d . The error term $\|x - G_2(\bar{z}^p)\|$ scales linearly with n . Hence, we need to choose δ, K such that the additive term in inequality (6) scales sublinearly. We may set δ to scale as $\frac{1}{\sqrt{p}}$. To get the same order of measurements as CSGM, we may set K to scale as \sqrt{k} . For that choice of parameters, the radius for the intermediate search, i.e. r_2 scales as $\sqrt{\frac{k}{p}} n^{-d_2}$, where d_2 is the depth of G_2 .*

Remark 4 (Model expressivity). *As we optimize over deeper layers, the model expressivity increases and hence the error $\|x - G_2(\bar{z}^p)\|$ decreases.*

3.3. Sketch of the proof

The central novelty of our proof is how we upper bound the metric entropy of the epsilon nets used to cover the extended range of the generator, i.e. the set $G_1(B_2^k(r_1)) \oplus B_1^p(r_2)$. First, we observe that if S_1 is a epsilon net for $G_1(B_2^k(r_1))$ and S_2 is an epsilon net for $B_1^p(r_2)$, then a simple bound for the size of an epsilon net on the extended range will have at most $|S_1| \cdot |S_2|$ elements.

CSGM uses a volumetric argument to upper bound the size of the epsilon net for S_1 . Our key idea is that using the same method to bound the size of the cover for the l_1 ball is sub-optimal for small radii. Instead, we use Maurey's empirical method or the related Sudakov's minoration inequality

(Pisier, 1986; Wainwright, 2019) yielding logarithmic (instead of linear) dependence on the dimension p . Maurey’s bound poses technical challenges that we need to address when extending the chaining argument of the CSGM proof. With Maurey’s method, successive nets in the chaining can have significantly higher metric entropy for large radii. To minimize the additive error in our bound during chaining, we switch from volumetric epsilon-nets to Maurey’s method at the right selected scale. The full proof of our Theorem can be found in the Appendix.

3.4. S-REC for partial circulant matrices

We extend the theory of matrices that satisfy the S-REC condition beyond i.i.d. Gaussian measurements. To establish that a family of random matrices satisfies this condition (and hence obtains sample complexity bounds), three conditions must be proved with high probability (Bora et al., 2017; Baraniuk et al., 2008): (1) The random matrix A should satisfy the Johnson-Lindenstrauss (JL) lemma on a suitable ϵ -net, (2) The matrix operator norm should be bounded: $\|A\|_{op} \leq \sqrt{n}$, and (3) for a fixed vector x , $\|Ax\| \leq 2\|x\|$.

Here we establish that randomly signed partial circulant matrices satisfy the S-REC condition for a number of measurements scaling similarly to Gaussian i.i.d. measurements.

Lemma 1. Consider the setting of Theorem 1. Let $g = [g_1, \dots, g_n]$ be a vector with i.i.d. Gaussian entries of variance $1/m$, let $F \in \mathbb{R}^{m \times n}$ be a partial circulant matrix that has g in its first row, and let $D \in \mathbb{R}^{n \times n}$ be a diagonal matrix with uniform ± 1 entries along its diagonal. Then for $m = \Omega\left(\frac{1}{(1-\gamma)^2}(k \log \frac{L_1 L_2 r_1}{\delta} + K^2 \log p) \log^4(n)\right)$, FD satisfies S-REC($G_2(G_1(B_2^k(r_1)) \oplus B_1^p(r_2)), 1 - \gamma, \delta \cdot \frac{\log(4K)}{\gamma} \cdot \frac{\sqrt{p}}{K} \log \frac{\sqrt{p}}{K}$) with probability $1 - e^{-\Omega(m)}$.

Our proof of this lemma can be found in the appendix and relies on previous results establishing JL properties for partial circulant matrices post-multiplied by random diagonal matrices (Krahmer & Ward, 2011; Hinrichs & Vybráil, 2011).

There is an important computational benefit in such structured measurement matrices. We are sensing high resolution images that are 1024×1024 for 3 color channels resulting in signal dimension n being 3 million. If measurements are at ten percent (a typically challenging compressed sensing regime), that results to $m \times n$ matrices that are $300k \times 3m$ which require gigabytes to store and hit GPU memory limitations. Therefore random Gaussian measurement matrices cannot be implemented for high resolution imaging. Partial circulant matrices require orders of magnitude less memory due to their structure and matrix-vector products can be computed much faster using FFT. We expect that these benefits will have a key role for future high-resolution imaging systems.

4. Experiments

4.1. Algorithmic adaptations to StyleGAN

Up to this point, we have presented and theoretically analyzed the ILO algorithm. Our method is not tied to any specific architecture and it only assumes access to a generative model and the underlying domain of the latent space of the initial layer. In this section, we present empirical innovations on how to use our framework with the state-of-the-art generative model StyleGAN-2 (Karras et al., 2020).

StyleGAN-2 has several peculiarities that need to be taken into account for the design of a compressed sensing algorithm. First, in StyleGAN-2 the initial latent code $z^k \in \mathbb{R}^k$ is not fed directly to the model. Instead, it is first mapped through a multilayer linear network, *the mapping network*, to an intermediate representation $w^k \in \mathbb{R}^k$. We refer to the domains of z^k, w^k as \mathcal{Z}, \mathcal{W} respectively. During training, a z^k is sampled according to a distribution on \mathcal{Z} , it gets transformed through the mapping network to a $w^k \in \mathcal{W}$ and one copy of w^k is fed to each one of the 18 layers of StyleGAN-2. Additionally, each one of the layers receives a noise vector u^k (unique for each layer).

4.1.1. OPTIMIZATION SETTING

The first thing to decide is which intermediate layer will be used to split the StyleGAN-2 generator. We observe that we obtain better results with multiple splits. We consider the generator of StyleGAN-2 as a composition of layers $G_1 \circ G_2 \circ \dots \circ G_{18}$ and we run Algorithm (1) in rounds, where in each round the initial layer is discarded.

To ensure that we stay in an l_1 ball around the manifold at each layer, we use Projected Gradient Descent (PGD) (Nesterov, 2003). To implement the projection to an l_1 ball around the current best solution (see line 4 of Algorithm (1)), we use the method of Duchi et al. (2008). Guided by our theory, we increase the maximum allowed deviation as we move to higher dimensional latent spaces. The radii of the balls are tuned separately as hyperparameters, for a full description see the Appendix.

For all inverse problems, it is helpful to allow the w^k vectors to deviate (Menon et al., 2020), i.e. we can optimize over a sequence $\{w_i^k\}_{i=1}^{18}$. The deviations are typically regularized with an additional term in the loss function, which captures the geodesic distance of the vectors. PULSE reports that optimizing only over the first five noise vectors, i.e. $\{u_i^k\}_{i=1}^5$, yields better reconstructions for super-resolution comparing to optimizing over the whole sequence. We show that this is not necessarily true if this optimization is performed sequentially. Our method starts by optimizing only the first five noise vectors (as in PULSE), but we gradually allow optimization of the rest of the latent vectors as we move to higher dimensional latent spaces.

4.2. Loss functions and adaptation to general inverse problems

Here we consider the effect of different loss functions in solving general inverse problems. It has been observed that LPIPS yields optimal performance with image size 256×256 (Karras et al., 2020). Therefore, we downsample images from 1024×1024 to 256×256 pixels. If the given image is inpainted, missing pixels are mixed with observed pixels during this downsampling. We observe that this blending leads to distorted reconstructions when using the LPIPS loss. Hence, for inpainting under scarce measurements we use only the MSE loss. We note that unlike the previously proposed methods, ILO can work for inpainting with extremely few observed pixels – even with less than 1% of the whole image. In presence of enough measurements, we use both LPIPS and MSE. To address these distortion issues, we minimize the perceptual distance between the generated image and a superimposed reconstruction, i.e. we replace the missing pixels of the observed image with the ones generated by StyleGAN prior to downsampling.

For super-resolution, we use a weighted average of LPIPS and MSE (as in inpainting with sufficient measurements). To compare the high-resolution and low-resolution images, we first downsample with cubic interpolation (Keys, 1981) as in PULSE. We also consider the problem of denoising, where Gaussian noise is added to the image. As usual, we assume knowledge to the forward operator $\mathcal{A}(x)$. Simply inverting a noisy high-resolution image creates grainy reconstructions due to the expressive power of StyleGAN-2. We address this in the optimization process by adding gaussian noise to the generated images before using them in the loss function. We call this new technique **Stochastic Noise Addition** (SNA).

4.3. Results

We show that ILO obtains state-of-the-art unsupervised performance for solving inverse problems with deep generative models in four different settings: inpainting, super-resolution, denoising and compressed sensing with circulant matrices. We compare with different variants of the CSGM algorithm using optimization and loss function innovations introduced in PULSE and StyleGAN. Unless stated otherwise, we will denote with CSGM + MSE the optimization procedure described in PULSE for the StyleGAN generator. Through a wide variety of experiments, we observe that ILO largely outperforms alternative techniques, both in terms of visual quality and in terms of true MSE error. We measure the latter on images sampled randomly from Celeba-HQ (Liu et al., 2018; Lee et al., 2020). Finally, to show the benefits of extending the range of the generator, we illustrate how one can use an adversarially robust classifier to guide the generation of human faces that look like objects from ImageNet (Deng et al., 2009).

Inpainting: For inpainting, the algorithm tries to complete missing pixels to a given image. The measurement process corresponds to a linear matrix that has rows that are a subset of the identity. Results for inpainting are shown in Figure 1. We perform two types of experiments. First, we mask important facial features from real images (collected from the web) and generated images from StyleGAN-2. Next, we do randomized inpainting, i.e. we inpaint pixels of a given image independently with a pre-defined probability. We experiment with observation probabilities up to 1%. This is a very challenging scenario: a human observer cannot distinguish face characteristics from such few pixels, e.g. see Figure 1 last row, second column. As shown in the Figure, ILO gives reconstructions that look much closer to the hidden image than the other methods. Our method is able to give surprisingly accurate reconstructions even under extreme scarce measurements (see last column, last row of Figure 1). To quantify the performance of the different methods we randomly select a few images from Celeba-HQ (Liu et al., 2018; Lee et al., 2020) and reconstruct at different levels of sparsity. Figure 2 column 1 shows that ILO is $2\times$ better in terms of reconstruction error anywhere between 5% – 100% observed pixels.

Denoising: Our next experiment is on denoising. To ablate the SNA framework we introduced, we show results with and without our technique on an image with additive noise of standard deviation $\sigma = 30$. Results are summarized in Table 1. Since SNA consistently improves reconstruction,

Algorithm	SNA	PSRN (dB)
CSGM	✗	19.89
	✓	21.38
ILO	✗	28.34
	✓	32.92

Table 1. Results with and without SNA for a noisy image ($\sigma = 30$).

we use it in all subsequent denoising experiments.

We compare with the CSGM framework using MSE, only LPIPS or a combination of both loss functions. For ILO, we only use a weighted combination of MSE and LPIPS. We also compare with a standard denoising method, the BM3D algorithm (Dabov et al., 2006). We vary the noise standard deviation from 5 to 256 and clip the perturbed values to the range $[0, 255]$ (RGB). Results are shown in Figure 2, second column. We observe that ILO outperforms all the previously proposed CSGM based methods by a large margin. For the typical setting of $\sigma = 25$, ILO is $1.8\times$ better than the best performing CSGM baseline. BM3D shows excellent performance, outperforming all other methods in the very low noise regime but rapidly deteriorates for harder settings. Figure 3 shows visual results for the task of denoising. As shown, ILO gives superior visual reconstructions and better

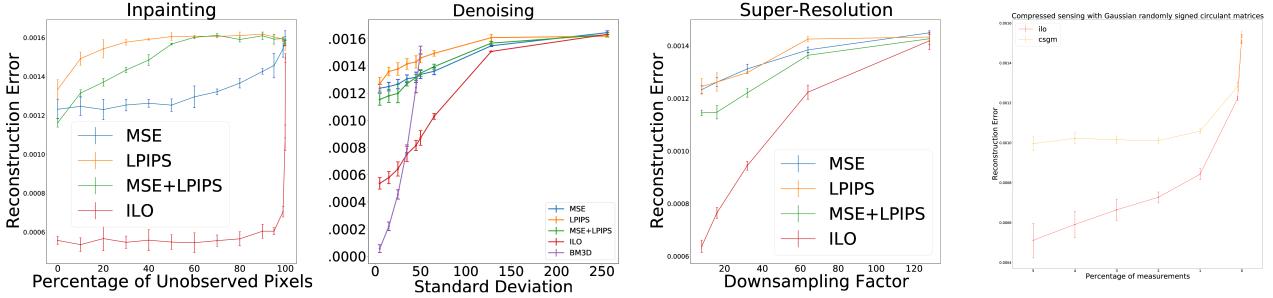


Figure 2. Plots showing the true MSE error on Celeba-HQ images, i.e. the MSE between the real image (that we never observe) and the reconstructed image from the measurements. From left to right: Inpainting, Denoising, Super-resolution and Compressed sensing with partial circulant matrices. As shown, ILO significantly outperforms all previous methods except in the very noisy regime.

actual performance comparing to the (adapted for denoising) PULSE and the classical BM3D method.

Super-resolution: We report results on super-resolution, the task PULSE was actually designed for. We sample images from Celeba-HQ, downsample using Bicubic Down-sampling and measure the reconstruction error. Three example reconstructions are shown in Figure 4. We also report reconstruction error on Celeba-HQ. Results are reported in Figure 2, third column. As shown, ILO outperforms significantly all the other methods, including PULSE (CSGM + MSE). To give some examples, when the image is down-scaled from 1024×1024 to 64×64 (scaling factor 16), ILO is $1.65 \times$ better than PULSE in terms of reconstruction error. For 32×32 images, ILO is $1.4 \times$ better than PULSE.

As shown, our method not only generates reconstructions that look much closer to the true image, but also appears to generate more racially diverse samples (Jain et al., 2020; Tan et al., 2020; Menon et al., 2020), e.g. see third row.

Compressed sensing with partial circulant matrices: For an experiment with observations of random projections we used partial circulant measurement matrices with random signs. Lemma 1 establishes that such matrices satisfy the conditions for Theorem 1. Figure 2, column 4, shows the reconstruction error when varying the number of measurement rows. When the number of measurements is 5% of the dimension n , ILO performs $2 \times$ better than CSGM in terms of reconstruction error.

Out of Distribution generation: Our method can generate images that lie outside of the range of the pre-trained generator. By choosing the radius of the l_1 ball for each layer, we control the trade-off between how natural (comparing to the dataset the model was trained on) these images look, and the out-of-distribution generation capability of our model.

To demonstrate this, we run the following experiment; we remove entirely the loss functions that relate the generated images with a reference image (i.e. MSE and LPIPS) and

we add a new classification loss term using an external classifier trained on a different domain. Essentially, we search for latent codes that lie in an l_1 ball around the range of intermediate layers and maximize the probability that the generated image belongs to a certain category. We consider a classifier trained on ImageNet (Deng et al., 2009). This optimization problem is one of the simplest methods to create adversarial examples (Xiao et al., 2018) and hence the generated images will not be visually interesting. However, if our classifier is adversarially robust, then even optimizing directly over the pixel space leads to an interesting generative process (Santurkar et al., 2019). We use the latent space of StyleGAN-2 to generate images of faces with fruit or animal characteristics. The radius of the l_1 projection at different layers controls the distance of the generated images to human faces. The results are shown in Figure 5.

Running time: Our algorithm runs CSGM as the first step and therefore initially seems to be strictly slower. Surprisingly, ILO can find better solutions than CSGM in fewer total steps. StyleGAN-2 typically requires 300 – 1000 optimization steps (on the first layer) for a good reconstruction (Karras et al., 2019; 2020). However, we observe that running 50 steps in each one of the first four layers outperforms CSGM. That said, ILO continues to improve with more iterations, also depending on task, number of measurements and hyperparameters. Figures 6, 7 show speed plots (with vertical axis being MSE and LPIPS correspondingly) for inversion with ILO. To obtain the plots, we use a single V100 GPU. We run 300 optimization steps per layer. The leftmost point corresponds to CSGM, i.e. we optimize over only the first layer. For each other point, we also optimize over the input space of a layer one level deeper compared to the previous point, e.g. the rightmost point corresponds to 1200 optimization steps (300 for each one of the first four layers).

Related Work: There has been significant recent work in unsupervised methods for inverse problems using pre-



Figure 3. Results on the task of denoising. Gaussian noise ($\sigma = 25$, known) is added to the original image and recovered with various methods. The MSE images indicate the reconstructed images obtained by inverting the noisy image.

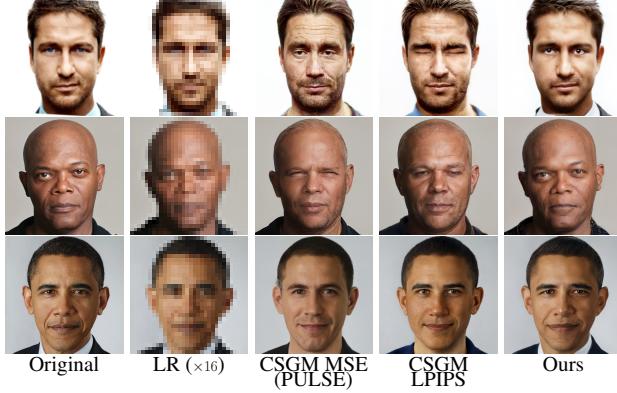


Figure 4. Results on the super-resolution task. ILO (ours) gives more accurate reconstructions comparing to PULSE (third column) and other baselines. Many biased reconstructions can be corrected by applying ILO on the weighted combination of MSE and LPIPS.



Figure 5. Illustration of using a classifier as a differentiable forward operator. Here we assume that the only observation is $y = \mathcal{A}(x|\text{class})$ where \mathcal{A} is an ImageNet classifier. The classes used in this Figure are (from top-left): Frog, Coral, Irish Wolf Dog, Goldfish, Boston Terrier Dog and Apple. We use a robust classifier as proposed by Santurkar et al. (2019) and solve the inverse problem to generate images that look like these classes. The difference with Santurkar et al. (2019) is that we perform the search using ILO in the StyleGAN-2 generator latent spaces as opposed to pixels and that keeps images closer to human faces.

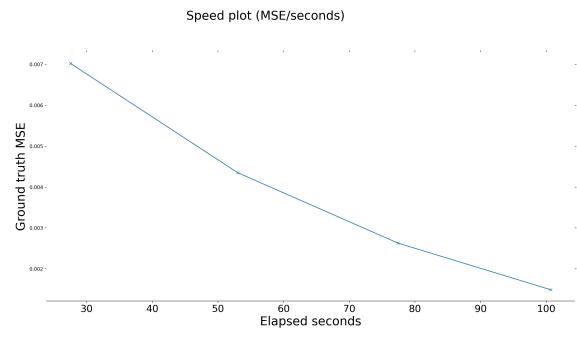


Figure 6. Speed plot (MSE / elapsed seconds) for the task of inversion. We use a single V100 GPU. We run 300 optimization steps per layer. The leftmost point corresponds to CSGM, i.e. we optimize over only the first layer. For each other point, we also optimize over the input space of a layer one level deeper compared to the previous point, e.g. the rightmost point corresponds to 1200 optimization steps (300 for each one of the first four layers).

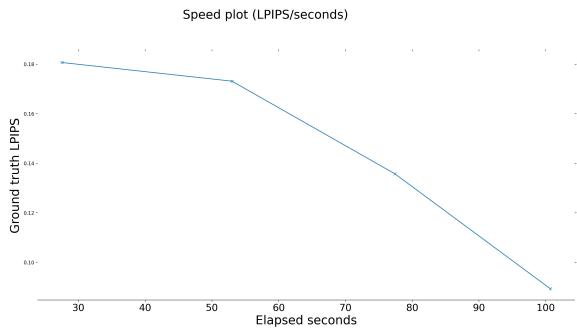


Figure 7. Speed plot (LPIPS / elapsed seconds) for the task of inversion. We use a single V100 GPU. We run 300 optimization steps per layer. The leftmost point corresponds to CSGM, i.e. we optimize over only the first layer. For each other point, we also optimize over the input space of a layer one level deeper compared to the previous point, e.g. the rightmost point corresponds to 1200 optimization steps (300 for each one of the first four layers).

trained generative models. Recently, Liu & Scarlett (2020) showed that the sample scaling of CSGM is near-optimal in the absence of further assumptions. Hand & Voroninski (2018) proved algorithmic convergence guarantees for solving non-convex linear inverse problems with deep generative priors under random weight assumptions. Faster recovery algorithms were proposed by Raj et al. (2019); Shah & Hegde (2018) while Pandit et al. (2019) analyzed approximate message passing (AMP) for inverse problems in the high-dimensional random limit. Beyond AMP, Regularization-by-Denoising (RED) methods have shown excellent recent performance in imaging, see e.g. Sun et al. (2019). Deep generative models have been developed for MRI (Mardani et al., 2018) and benefited from task-awareness (Kabkab et al., 2018), meta-learning (Wu et al., 2019) and specifically designed autoencoders (Mousavi et al., 2019). Solving inverse problems for out-of-distribution images has been studied in (Asim et al., 2020).

The theoretical framework we introduce is related to the ideas proposed by Dhar et al. (2018) on allowing additive sparse deviations in the generated images. In that case, the recovered signals have the form $G(z) + v$, where $G : \mathbb{R}^k \rightarrow \mathbb{R}^n$ is a deep generative model, $z \in \mathbb{R}^k$ is a latent variable and $v \in \mathbb{R}^n$ is an l -sparse vector. The additive term allows the recovery of signals that lie outside of the range. Our approach is a generalization of this framework since it allows sparse deviations anywhere in the latent space.

Intermediate Layer Optimization has also been proposed in the GAN Surgery paper (Park et al., 2020) and in Bau et al. (2019) as a method to identify modes of the data distribution that have been dropped from the generator. The main difference between ILO and this prior line of work is in the optimization procedure; we optimize sequentially and we regularize with a previous search in the lower dimensional space and by allowing only small deviations from the range. Our paper also benefits from the StyleGAN-2 architecture (Karras et al., 2019; 2020) and several key ideas from PULSE (Menon et al., 2020).

Finally, there is significant prior work on deep learning methods that do not rely on pre-trained generators, see e.g. (Lucas et al., 2018; Yu et al., 2019; Liu et al., 2019; Sun & Chen, 2020; Sun et al., 2020; Yang et al., 2019; Tian et al., 2020; Tripathi et al., 2018). Such methods can show excellent performance but require training a network specifically for each reconstruction task. This is in contrast with our framework that can solve all inverse problems universally, leveraging the same pre-trained network.

5. Conclusions and Future Work

We proposed a novel framework for solving inverse problems leveraging pre-trained generative models. Our method

expands the range of the generator by optimizing different intermediate layers and achieves excellent performance for several tasks. On the theory side, a central open problem would be to establish global convergence of ILO, possibly following the ideas of (Hand & Voroninski, 2018; Hand et al., 2018) or surfing (Song et al., 2019).

On the empirical side, a central open problem would be the application of our framework in other domains like medical imaging, but that would require pre-trained generative models e.g. for high-resolution MRI images. Another open direction that is particularly exciting is the use of classifiers to generate out-of-distribution samples. Our generated samples show the powerful modularity of combining pre-trained generators with differentiable forward operators that can guide image reconstruction in a data-driven way.

Acknowledgements

This research has been supported by NSF Grants CCF 1934932, AF 1901292, 2008710, 2019844 the NSF IFML 2019844 award as well as research gifts by Western Digital, WNCG and MLL, computing resources from TACC and the Archie Straiton Fellowship.

References

- Asim, M., Daniels, M., Leong, O., Ahmed, A., and Hand, P. Invertible generative models for inverse problems: mitigating representation error and dataset bias. In *International Conference on Machine Learning*, pp. 399–409. PMLR, 2020.
- Baraniuk, R., Davenport, M., DeVore, R., and Wakin, M. A simple proof of the restricted isometry property for random matrices. *Constructive Approximation*, 28(3): 253–263, 2008.
- Bau, D., Zhu, J.-Y., Wulff, J., Peebles, W., Strobel, H., Zhou, B., and Torralba, A. Seeing what a gan cannot generate, 2019.
- Bora, A., Jalal, A., Price, E., and Dimakis, A. G. Compressed sensing using generative models. In *International Conference on Machine Learning*, pp. 537–546. PMLR, 2017.
- Chen, G.-H., Tang, J., and Leng, S. Prior image constrained compressed sensing (piccs): a method to accurately reconstruct dynamic ct images from highly undersampled projection data sets. *Medical physics*, 35(2):660–663, 2008.
- Dabov, K., Foi, A., Katkovnik, V., and Egiazarian, K. Image denoising with block-matching and 3d filtering. In *Image Processing: Algorithms and Systems, Neural Networks, and Machine Learning*, volume 6064, pp. 606414. International Society for Optics and Photonics, 2006.
- Dasas, G., Odena, A., Zhang, H., and Dimakis, A. G. Your local gan: Designing two dimensional local attention mechanisms for generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.
- Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.
- Dhar, M., Grover, A., and Ermon, S. Modeling sparse deviations for compressed sensing using generative models. In *International Conference on Machine Learning*, pp. 1214–1223. PMLR, 2018.
- Duarte, M. F., Davenport, M. A., Takhar, D., Laska, J. N., Sun, T., Kelly, K. F., and Baraniuk, R. G. Single-pixel imaging via compressive sampling. *IEEE signal processing magazine*, 25(2):83–91, 2008.
- Duchi, J., Shalev-Shwartz, S., Singer, Y., and Chandra, T. Efficient projections onto the ℓ_1 -ball for learning in high dimensions. In *Proceedings of the 25th international conference on Machine learning*, pp. 272–279, 2008.
- Hand, P. and Voroninski, V. Global guarantees for enforcing deep generative priors by empirical risk. In *Conference On Learning Theory*, pp. 970–978. PMLR, 2018.
- Hand, P., Leong, O., and Voroninski, V. Phase retrieval under a generative prior. *arXiv preprint arXiv:1807.04261*, 2018.
- Hegde, C., Duarte, M. F., and Cevher, V. Compressive sensing recovery of spike trains using a structured sparsity model. In *SPARS'09-Signal Processing with Adaptive Sparse Structured Representations*, 2009.
- Hinrichs, A. and Vybíral, J. Johnson-lindenstrauss lemma for circulant matrices. *Random Structures & Algorithms*, 39(3):391–398, 2011.
- Jain, N., Olmo, A., Sengupta, S., Manikonda, L., and Kambhampati, S. Imperfect imagination: Implications of gans exacerbating biases on facial data augmentation and snapchat selfie lenses. *arXiv preprint arXiv:2001.09528*, 2020.
- Kabkab, M., Samangouei, P., and Chellappa, R. Task-aware compressed sensing with generative adversarial networks. In *AAAI*, 2018.
- Karras, T., Laine, S., and Aila, T. A style-based generator architecture for generative adversarial networks. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2019. doi: 10.1109/cvpr.2019.00453. URL <http://dx.doi.org/10.1109/CVPR.2019.00453>.
- Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., and Aila, T. Analyzing and improving the image quality of stylegan. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2020. doi: 10.1109/cvpr42600.2020.00813. URL <http://dx.doi.org/10.1109/cvpr42600.2020.00813>.
- Keys, R. Cubic convolution interpolation for digital image processing. *IEEE transactions on acoustics, speech, and signal processing*, 29(6):1153–1160, 1981.
- Krahmer, F. and Ward, R. New and improved johnson-lindenstrauss embeddings via the restricted isometry property. *SIAM Journal on Mathematical Analysis*, 43(3): 1269–1281, 2011.
- Lee, C.-H., Liu, Z., Wu, L., and Luo, P. Maskgan: Towards diverse and interactive facial image manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5549–5558, 2020.
- Lei, Q., Jalal, A., Dhillon, I. S., and Dimakis, A. G. Inverting deep generative models, one layer at a time. In *NeurIPS*, 2019.

- Liu, H., Jiang, B., Xiao, Y., and Yang, C. Coherent semantic attention for image inpainting. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct 2019. doi: 10.1109/iccv.2019.00427. URL <http://dx.doi.org/10.1109/ICCV.2019.00427>.
- Liu, Z. and Scarlett, J. Information-theoretic lower bounds for compressive sensing with generative models. *IEEE Journal on Selected Areas in Information Theory*, 1(1): 292–303, May 2020. ISSN 2641-8770. doi: 10.1109/jsait.2020.2980676. URL <http://dx.doi.org/10.1109/JSAIT.2020.2980676>.
- Liu, Z., Luo, P., Wang, X., and Tang, X. Large-scale celeb-faces attributes (celeba) dataset. *Retrieved August, 15 (2018):11*, 2018.
- Lucas, A., Iliadis, M., Molina, R., and Katsaggelos, A. K. Using deep neural networks for inverse problems in imaging: beyond analytical methods. *IEEE Signal Processing Magazine*, 35(1):20–36, 2018.
- Lustig, M., Donoho, D., and Pauly, J. M. Sparse mri: The application of compressed sensing for rapid mr imaging. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine*, 58(6):1182–1195, 2007.
- Lustig, M., Donoho, D. L., Santos, J. M., and Pauly, J. M. Compressed sensing mri. *IEEE signal processing magazine*, 25(2):72–82, 2008.
- Mardani, M., Gong, E., Cheng, J. Y., Vasanawala, S. S., Zaharchuk, G., Xing, L., and Pauly, J. M. Deep generative adversarial neural networks for compressive sensing mri. *IEEE transactions on medical imaging*, 38(1):167–179, 2018.
- Menon, S., Damian, A., Hu, S., Ravi, N., and Rudin, C. Pulse: Self-supervised photo upsampling via latent space exploration of generative models. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. doi: 10.1109/cvpr42600.2020.00251. URL <http://dx.doi.org/10.1109/cvpr42600.2020.00251>.
- Mousavi, A., Dasarathy, G., and Baraniuk, R. G. A data-driven and distributed approach to sparse signal representation and recovery. In *International Conference on Learning Representations*, 2019.
- Nesterov, Y. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media, 2003.
- Ongie, G., Jalal, A., Metzler, C. A., Baraniuk, R. G., Dimakis, A. G., and Willett, R. Deep learning techniques for inverse problems in imaging. *IEEE Journal on Selected Areas in Information Theory*, 1(1):39–56, 2020.
- Pajot, A., de Bezenac, E., and Gallinari, P. Unsupervised adversarial image reconstruction. In *International Conference on Learning Representations*, 2019. URL <https://openreview.net/forum?id=BJg4Z3RqF7>.
- Pandit, P., Sahraee, M., Rangan, S., and Fletcher, A. K. Asymptotics of map inference in deep networks. *2019 IEEE International Symposium on Information Theory (ISIT)*, Jul 2019. doi: 10.1109/isit.2019.8849316. URL <http://dx.doi.org/10.1109/ISIT.2019.8849316>.
- Park, J. Y., Smedemark-Margulies, N., Daniels, M., Yu, R., van de Meent, J.-W., and HAnd, P. Generator surgery for compressed sensing. In *NeurIPS 2020 Workshop on Deep Learning and Inverse Problems*, 2020. URL <https://openreview.net/forum?id=s2EucjZ6d2s>.
- Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., and Efros, A. A. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2536–2544, 2016.
- Pisier, G. Probabilistic methods in the geometry of banach spaces. In *Probability and analysis*, pp. 167–241. Springer, 1986.
- Qaisar, S., Bilal, R. M., Iqbal, W., Naureen, M., and Lee, S. Compressive sensing: From theory to applications, a survey. *Journal of Communications and networks*, 15(5): 443–456, 2013.
- Raj, A., Li, Y., and Bresler, Y. Gan-based projector for faster recovery with convergence guarantees in linear inverse problems. 2019.
- Richardson, E., Alaluf, Y., Patashnik, O., Nitzan, Y., Azar, Y., Shapiro, S., and Cohen-Or, D. Encoding in style: a stylegan encoder for image-to-image translation. *arXiv preprint arXiv:2008.00951*, 2020.
- Santurkar, S., Tsipras, D., Tran, B., Ilyas, A., Engstrom, L., and Madry, A. Image synthesis with a single (robust) classifier. *arXiv preprint arXiv:1906.09453*, 2019.
- Shah, V. and Hegde, C. Solving linear inverse problems using gan priors: An algorithm with provable guarantees. *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Apr 2018. doi: 10.1109/icassp.2018.8462233. URL <http://dx.doi.org/10.1109/ICASSP.2018.8462233>.
- Song, G., Fan, Z., and Lafferty, J. Surfing: Iterative optimization over incrementally trained deep networks. *NeurIPS*, 2019.

Sun, W. and Chen, Z. Learned image downscaling for upscaling using content adaptive resampler. *IEEE Transactions on Image Processing*, 29:4027–4040, 2020. ISSN 1941-0042. doi: 10.1109/tip.2020.2970248. URL <http://dx.doi.org/10.1109/TIP.2020.2970248>.

Sun, Y., Liu, J., and Kamilov, U. S. Block coordinate regularization by denoising. *NeurIPS*, 2019.

Sun, Y., Liu, J., and Kamilov, U. S. Block coordinate regularization by denoising. *IEEE Transactions on Computational Imaging*, 6:908–921, 2020. ISSN 2573-0436. doi: 10.1109/tci.2020.2996385. URL <http://dx.doi.org/10.1109/TCI.2020.2996385>.

Tan, S., Shen, Y., and Zhou, B. Improving the fairness of deep generative models without retraining, 2020.

Tian, C., Fei, L., Zheng, W., Xu, Y., Zuo, W., and Lin, C.-W. Deep learning on image denoising: An overview. *Neural Networks*, 131:251–275, Nov 2020. ISSN 0893-6080. doi: 10.1016/j.neunet.2020.07.025. URL <http://dx.doi.org/10.1016/j.neunet.2020.07.025>.

Tripathi, S., Lipton, Z. C., and Nguyen, T. Q. Correction by projection: Denoising images with generative adversarial networks. *arXiv preprint arXiv:1803.04477*, 2018.

Wainwright, M. J. *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48. Cambridge University Press, 2019.

Wu, Y., Rosca, M., and Lillicrap, T. Deep compressed sensing, 2019.

Xiao, C., Li, B., Zhu, J.-Y., He, W., Liu, M., and Song, D. Generating adversarial examples with adversarial networks. *arXiv preprint arXiv:1801.02610*, 2018.

Yang, W., Zhang, X., Tian, Y., Wang, W., Xue, J.-H., and Liao, Q. Deep learning for single image super-resolution: A brief review. *IEEE Transactions on Multimedia*, 21(12):3106–3121, 2019.

Yu, J., Lin, Z., Yang, J., Shen, X., Lu, X., and Huang, T. S. Generative image inpainting with contextual attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5505–5514, 2018.

Yu, J., Lin, Z., Yang, J., Shen, X., Lu, X., and Huang, T. Free-form image inpainting with gated convolution. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct 2019. doi: 10.1109/iccv.2019.00457. URL <http://dx.doi.org/10.1109/ICCV.2019.00457>.