GAN-based Projector for Faster Recovery with Convergence Guarantees in Linear Inverse Problems

Ankit Raj* Yuqi Li* Yoram Bresler
University of Illinois at Urbana-Champaign, USA
{ankitr3, yuqil3, ybresler}@illinois.edu

Abstract

A Generative Adversarial Network (GAN) with generator G trained to model the prior of images has been shown to perform better than sparsity-based regularizers in ill-posed inverse problems. Here, we propose a new method of deploying a GAN-based prior to solve linear inverse problems using projected gradient descent (PGD). Our method learns a network-based projector for use in the PGD algorithm, eliminating expensive computation of the Jacobian of G. Experiments show that our approach provides a speed-up of 60-80$\times$ over earlier GAN-based recovery methods along with better accuracy. Our main theoretical result is that if the measurement matrix is moderately conditioned on the manifold range(G) and the projector is δ-approximate, then the algorithm is guaranteed to reach $O(\delta)$ reconstruction error in $O(log(1/\delta))$ steps in the low noise regime. Additionally, we propose a fast method to design such measurement matrices for a given G. Extensive experiments demonstrate the efficacy of this method by requiring 5-10$\times$ fewer measurements than random Gaussian measurement matrices for comparable recovery performance. Because the learning of the GAN and projector is decoupled from the measurement operator, our GAN-based projector and recovery algorithm are applicable without retraining to all linear inverse problems, as confirmed by experiments on compressed sensing, super-resolution, and inpainting.

1. Introduction

Many application such as computational imaging, and remote sensing fall in the compressive sensing (CS) paradigm. CS [9, 5] refers to projecting a high dimensional, sparse or sparsifiable signal $x \in \mathbb{R}^n$ to a lower dimensional measurement $y \in \mathbb{R}^m, m \ll n$, using a small set of linear, non-adaptive frames. The noisy measurement model is:

$$y = Ax + v, A \in \mathbb{R}^{m \times n}, v \sim \mathcal{N}(0, \sigma^2 I) \quad (1)$$

where the measurement matrix A is often a random matrix. In this work, we are interested in the problem of recovering the unknown natural signal x, from the compressed measurement y, given the measurement matrix A. Traditionally, for signal priors, natural images are considered sparse in some fixed or learnable basis [11, 8, 36, 22, 7, 38, 10, 21]. Instead of the sparse prior commonly adopted by CS literature, we turn to a learned prior. Neural network-based inverse problem solvers have been explored recently [14, 35, 31, 1, 12, 15, 25, 32, 22, 37, 26]. However, [1, 12, 15, 25] use information about the measurement matrix A while training the network. Thus, their algorithms are limited to a particular set-up to solve specific inverse-problem and usually cannot solve other problems without retraining. Another line of work, [28, 29] jointly optimizes the measurement matrix and recovery algorithm, again resulting in algorithm limited to a particular inverse problem and measurement matrix. Instead, in this paper the network is trained independently of A and can be generalized across different inverse problems. This aspect is shared by two other neural-network-based solvers [35, 31], however, they model the image prior only implicitly by training a denoiser

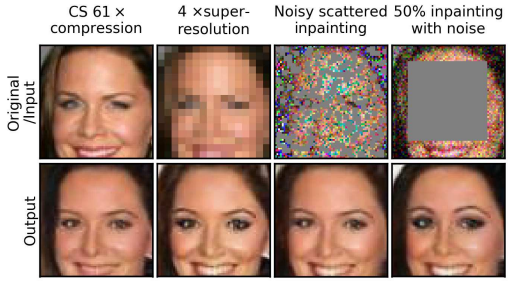


Figure 1: Our network-based PGD solves the following inverse problems: compressive sensing with 61$\times$ compression, 4$\times$ super-resolution, scatterd inpaining with high noise ($\sigma = 40$) and 50% blocked inpainting with high noise.

*Equal contribution. Ankit Raj and Yoram Bresler's research work was supported in part by the National Science Foundation under Grant IIS 14-47879 . Yuqi Li and Yoram Bresler's reseach work was supported in part by Sandia National Laboratories under Grant ID: AE056, IP: 00371547

or a proximal map, and perhaps for this reason appear to require massive quantity of training samples. Importantly, very little is known about why and when they perform well, as even if the learned proximal map is assumed to be exact, there is no theoretical convergence guarantee or bound on the recovery error.

In this work, we leverage the success of generative adversarial network (GAN) [13, 6, 42, 39, 3, 20] in modeling the distribution of data. Indeed, GAN-based priors for natural images have been successfully employed to solve linear inverse problems [24, 4, 33]. However, in [24], the operator A is integrated into training the GAN, limiting it to a particular inverse problem. We therefore focus on the recent papers [4, 33] closest to our work, for extensive comparisons.

Bora *et al*. [4] do not have a guarantee on the convergence of their algorithm for solving the non-convex optimization problem, requiring several random initializations. Similarly, in [33], the inner loop uses a gradient descent algorithm to solve a non-convex optimization problem with no guarantee of convergence to a global optimum. Furthermore, the conditions imposed in [33] on the random Gaussian measurement matrix for convergence of their outer iterative loop are unnecessarily stringent and cannot be achieved with a moderate number of measurements. Importantly, both these methods require expensive computation of the Jacobian $\nabla_z G$ of the differentiable generator G with respect to the latent input z. Since computing $\nabla_z G$ involves back-propagation through G at every iteration, these reconstruction algorithms are computationally expensive and even when implemented on a GPU they are slow.

We propose a GAN-based projection network to solve compressed sensing recovery problems using projected gradient descent (PGD). We are able to reconstruct the image even with $61\times$ compression ratio (*i.e*., with less than 1.6% of a full measurement set) using a random Gaussian measurement matrix. The proposed approach provides superior recovery accuracy over existing methods, simultaneously with a 60-80$\times$ speed-up, making the algorithm useful for practical applications. We also provide theoretical results on the convergence of the reconstruction error, given that the measurement matrix A satisfies certain conditions when restricted to the range $R(G)$ of the generator. We complement the theory by proposing a method to design a measurement matrix that satisfies these sufficient conditions for guaranteed convergence. We assess these sufficient conditions for both the random Gaussian measurement matrix and the designed matrix for a given dataset. Both our analysis and experiments show that with the designed matrix, 5-10$\times$ fewer measurements suffice for robust recovery. Because the training of the GAN and projector is decoupled from the measurement operator, we demonstrate that other linear inverse problems like super-resolution and inpainting can also be solved using our algorithm without retraining.

2. Problem Formulation

Let $x^* \in \mathbb{R}^n$ denote a ground truth image, A a fixed measurement matrix, and $y = Ax^* + v \in \mathbb{R}^m$ the noisy measurement, with noise $v \sim \mathcal{N}(0, \sigma^2 I)$. We assume that the ground truth images lie in a non-convex set $S = R(G)$, the range of generator G. The maximum likelihood estimator (MLE) of x^*, $\hat{x}_{MLE}$, can be formulated as follows:

$$\hat{x}_{MLE} = \underset{x \in R(G)}{\arg\min} - \log p(y|x) \quad = \underset{x \in R(G)}{\arg\min} \|y - Ax\|_2^2$$

Bora *et al*. [4] (whose algorithm we denote by CSGM) solve the optimization problem $\hat{z} = \arg\min_{z \in \mathbb{R}^k} \|y - AG(z)\|^2 + \lambda \|z\|^2$ in the latent space (z), and set $\hat{x} = G(\hat{z})$. Their gradient descent algorithm often gets stuck at local optima. Since the problem is non-convex, the reconstruction is strongly dependent on the initialization of z and requires several random initializations to converge to a good point. To resolve this problem, Shah and Hegde [33] proposed a projected gradient descent (PGD)-based method (which we call PGD-GAN) to solve (2), shown in fig.2(a). They perform gradient descent in the ambient (x)-space and project the updated term onto $R(G)$. This projection involves solving another non-convex minimization problem (shown in the second box in fig.2(a)) using the Adam optimizer [17] for 100 iterations from a random initialization. No convergence result is given for this iterative algorithm to perform the non-linear projection, and the convergence analysis for the PGD-GAN algorithm [33] only holds if one assumes that the inner loop succeeds in finding the optimum projection.

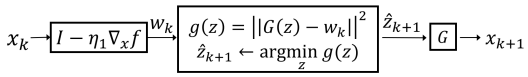
Our main idea in this paper is to replace this iterative scheme in the inner-loop with a learning-based approach, as it often performs better and does not fall into local optima [42]. Another important benefit is that both earlier approaches require expensive computation of the Jacobian of G, which is eliminated in the proposed approach.

3. Proposed Method

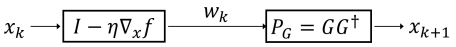
In this section, we introduce our methodology and architecture to train a projector using a pre-trained generator G and how we use this projector to obtain the optimizer in (2).

3.1. Inner-Loop-Free Scheme

We show that by carefully designing a network architecture with a suitable training strategy, we can train a projector onto $R(G)$, the range of the generator G, thereby removing the inner-loop required in the earlier approach. The resulting iterative updates of our network-based PGD (NPGD) algorithm are shown in fig.2(b). This approach eliminates the need to solve the non-convex optimization problem in the inner-loop, which depends on initialization and requires



(a) PGD with inner-loop



(b) Network-based PGD (NPGD)

Figure 2: (a) Block diagram of PGD using inner-loop [33]. k represents the outer loop iterators and $\hat{z}_{k+1}$ is the optimizer of $\|G(z) - w_k\|^2$ obtained by solving the inner-loop using Adam optimizer. (b) Block diagram of our network-based PGD (NPGD) with $P_G = GG^\dagger$ as a network based projector onto $R(G)$. $f(x) = \|Ax - y\|^2$ is the cost function defined in (2)

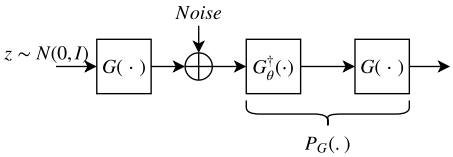


Figure 3: Architecture to train a projector onto range(G)

several restarts. Furthermore, our method provides a significant speed-up by a factor of 30-40$\times$ on the CelebA dataset for two major reasons: (i) since there is no inner-loop, the total number of iterations required for convergence is significantly reduced, (ii) doesn't require computation of ∇G_z *i.e.* the Jacobian of the generator with respect to the input, z. This expensive operation repeats back-propagation through the network for $T_{out} \times \#_{restarts}$(for [4]) or $T_{out} \times T_{in}$ (for [33]) times, where $\#_{restarts}, T_{out}$ and T_{in} are number of restarts, outer and inner iterations respectively.

3.2. Generator-based Projector

A GAN consists of two networks, generator and discriminator, which follow an adversarial training strategy to learn the data distribution. A well-trained generator $G : \mathbb{R}^k \to R(G) \subset \mathbb{R}^n, k \ll n$ takes in a random latent variable $z \sim \mathcal{N}(0, I_k)$ and produces sharp looking images imitating the training data distribution in $\mathbb{R}^n$. The goal is to train a network that projects an image $x \in \mathbb{R}^n$ onto $R(G)$. The projector, P_S onto a set S should satisfy two main properties: (i) *Idempotence*, for any point x, $P_S(P_S(x)) = P_S(x)$, (ii) *Least distance*, for a point $\tilde{x}$, $P_S(\tilde{x}) = \arg\min_{x \in S} \|x - \tilde{x}\|^2$. Figure 3 shows the network structure we used to train a projector using a GAN. We define the multi-task loss to be:

$$\begin{aligned} \mathcal{L}(\theta) = \mathbb{E}_{z,\nu}\left[\left\|G\left(G_\theta^\dagger\left(G(z)+\nu\right)\right) - G(z)\right\|^2\right] \\ + \mathbb{E}_{z,\nu}\left[\lambda\left\|G_\theta^\dagger\left(G(z)+\nu\right) - z\right\|^2\right] \end{aligned} \tag{2}$$

where G is a generator obtained from the GAN trained on a particular dataset. Operator $G_\theta^\dagger : \mathbb{R}^n \to \mathbb{R}^k$, parameterized by θ, approximates a non-linear least squares pseudo-inverse of G and $\nu \sim \mathcal{N}(0, I_n)$ indicates noise added to the generator's output for different $z \sim \mathcal{N}(0, I_k)$ so that the projector network denoted by $P_G = GG_\theta^\dagger$ is trained on points outside the range(G) and learns to project them onto $R(G)$. The objective function consists of two parts. The first is similar to standard *Encoder-Decoder* framework, however, the loss function is minimized over θ – the parameters of $G^\dagger$, while keeping the parameters of G (obtained by standard GAN training) fixed. This ensures that $R(G)$ doesn't change and $P_G = GG^\dagger$ is a mapping onto $R(G)$. The second part is used to keep $G^\dagger(G(z))$ close to true z used to generate training image $G(z)$. This second term can be considered a regularizer for training the projector with λ being the regularization constant.

Algorithm 1 Network-based Projected Gradient Descent

Input: loss function f, $A, y, G, G^\dagger$
Parameter: step size $\eta(= \frac{1}{\beta})$
Output: an estimate $\hat{x} \in R(G)$

Let $t = 0, x_0 = A^T y$.
while $t < T$ **do**
  $w_t := x_t - \eta A^T(Ax_t - y)$
  $x_{t+1} := G(G^\dagger(w_t))$
end while
return $\hat{x} = x_T$

4. Theoretical Study

4.1. Convergence Analysis

Let $f(x) = \|Ax - y\|_2^2$ denote the loss function of projected gradient descent. Algorithm (1) describes the proposed network-based projected gradient descent (NPGD) to solve equation (2).

Definition 1 (Restricted Eigenvalue Constraint (REC)) *Let $S \subset \mathbb{R}^n$. For some parameters $0 < \alpha < \beta$, matrix $A \in \mathbb{R}^{m \times n}$ is said to satisfy the $REC(S, \alpha, \beta)$ if the following holds for all $x_1, x_2 \in S$.*

$$\alpha\|x_1 - x_2\|^2 \leq \|A(x_1 - x_2)\|^2 \leq \beta\|x_1 - x_2\|^2. \tag{3}$$

Definition 2 (Approximate Projection using GAN) *A concatenated network $G(G^\dagger(\cdot)) : \mathbb{R}^n \to R(G)$ is a δ-approximate projector, if the following holds for all $x \in \mathbb{R}^n$:*

$$\|x - G(G^\dagger(x))\|^2 \leq \min_{z \in \mathbb{R}^k} \|x - G(z)\|^2 + \delta \tag{4}$$

Theorem 1 provides upper bounds on the cost function and reconstruction error of our NPGD algorithm after n iterations.

Theorem 1 *Let matrix $A \in \mathbb{R}^{m\times n}$ satisfy the $REC(S,\alpha,\beta)$ with $\beta/\alpha < 2$, and let the concatenated network $G(G^\dagger(\cdot))$ be a δ-approximate projector. Then for every $x^* \in R(G)$ and measurement $y = Ax^*$, executing algorithm 1 with step size $\eta = 1/\beta$, will yield $f(x_n) \leq (\frac{\beta}{\alpha} - 1)^n f(x_0) + \frac{\beta\delta}{2-\beta/\alpha}$. Furthermore, the algorithm achieves $\|x_n - x^*\|^2 \leq \left(C + \frac{1}{2\alpha/\beta - 1}\right)\delta$ after $\frac{1}{2-\beta/\alpha}\log\left(\frac{f(x_0)}{C\alpha\delta}\right)$ steps. When $n \to \infty$, $\|x^* - x_\infty\|^2 \leq \frac{\delta}{2\alpha/\beta-1}$.*

Proof 1 *Please refer to the appendix.*

From theorem 1, one important factor is the ratio β/α. This ratio largely determines the speed of linear ("geometric") convergence, as well as the reconstruction error $\|x^* - x_\infty\|^2$ at convergence. We would like β/α ratio as close to 1 as possible and must have $\beta/\alpha < 2$ for convergence. It has been shown in [2] that a random matrix A with orthonormal rows will satisfy this condition with high probability for m roughly linear in dimension k with log factors dependent on the properties of the manifold, in this case, $R(G)$. However, as we demonstrate later (see figure 4), a random matrix often will not satisfy the desired condition $\beta/\alpha < 2$ for small or moderate m. To extend into such regimes, we propose next a fast heuristic method to find a relatively good measurement matrix for an image set S, given a fixed m.

4.2. Generator-based Measurement Matrix Design

There have been a few attempts to optimize the measurement matrix based on the specific data distribution. Hegde *et al.* [16] find a deterministic measurement matrix that satisfies $REC(S, 1-\delta_S, 1+\delta_S)$ for a given finite set S of size $|S|$, but their time complexity is $O(n^3 + |S|^2n^2)$. Because the secant set S (defined later) would be of cardinality $|S| = O(M^2)$ for a training set of size M, with $M \gg n$, the time complexity would be infeasible even for fairly small n-pixel images. Furthermore, the final number of required measurements m, which is determined by the algorithm, depends on the isometry constant δ_S, and cannot be specified in advance. Kvinge *et al.* [18] introduced a heuristic iterative algorithm to find a measurement matrix with orthonormal rows that satisfies the REC with small β/α ratio, but their time complexity is $O\left(n^5\right)$ and the space complexity is $O(n^3)$, which is infeasible for a high-dimensional image dataset. Instead, our method, based on sampling from the secant set, has time complexity $O(Mn^2 + n^3)$, and space complexity $O(n^2)$, where M is a tiny fraction of $|S|$.

Definition 3 (Secant Set) *The normalized secant set of G is defined as follows:*

$$\mathcal{S}(G) = \left\{\frac{x_1 - x_2}{\|x_1 - x_2\|} : x_1, x_2 \in R(G)\right\} \tag{5}$$

and the associated distribution is denoted as Π_S, where

$$z_1, z_2 \sim \mathcal{N}(0, I_k), s = \frac{G(z_1) - G(z_2)}{\|G(z_1) - G(z_2)\|} \sim \Pi_S \tag{6}$$

Given $\mathcal{S}(G)$, the optimization over A is as follows:

$$\min_{A\in\mathbb{R}^{m\times n}} \frac{\beta}{\alpha} = \min_{A\in\mathbb{R}^{m\times n}} \frac{\max_{s\in\mathcal{S}(G)} \|As\|^2}{\min_{s\in\mathcal{S}(G)} \|As\|^2} \tag{7}$$

$$\leq \min_{AA^T=I_m} \frac{1}{\min_{s\in\mathcal{S}(G)} \|As\|^2} = \left(\max_{AA^T=I_m} \min_{s\in\mathcal{S}(G)} \|As\|^2\right)^{-1}$$

The inequality is due to an additional constraint on A : $AA^T = I_m$. This results in the largest singular value of A being 1 and hence the numerator term, $\max_{s\in\mathcal{S}(G)} \|As\|^2$, is at most 1. As the minimization in (7) requires iterating through the set S, we use the expected value over $s \sim \Pi_S$ as a surrogate objective

$$A = \arg\max_{AA^T=I_m} E_{s\sim\Pi_S}\left[\|As\|^2\right] \approx \arg\max_{AA^T=I_m} \frac{1}{M}\sum_{j=1}^{M} \|As_j\|^2 \tag{8}$$

The last approximation replaces the surrogate objective by its empirical estimate obtained by sampling $M \gg n$ secants $(s_j)_{j=1}^M$ according to Π_S. For m and M large enough, this designed measurement matrix would satisfy the condition $\beta/\alpha < 2$ for most of the secants in $R(G)$. Constructing an $n \times M$ matrix $D = [s_1|s_2|\ldots|s_M]$, (8) reduces to:

$$A^* = \arg\max_{A} \|AD\|_F^2 \text{ s.t. } AA^T = I_m \tag{9}$$

The optimal A^* in (9) has rows equal to the m leading eigenvectors DD^T. We compute $DD^T = \sum_{j=1}^M s_j s_j^T$ and its eigenvalue decomposition at time complexity $O(Mn^2 + n^3)$ and space complexity $O(n^2)$.

Our approach to the design of A is related to one of the steps described by [18], however by using the sampling-based estimates per (6) and (8) rather than the secant set for the entire training set, we reduce the computational cost by orders of magnitude to a modest level.

4.2.1 REC Histogram for A

We analyze the REC conditions by plotting the histogram of $\|As\|$ values for different measurement matrices $A \in R^{m\times n}$ in figure 4 where $s \in S$, the secant set of the samples from G trained on MNIST dataset. The left column shows the histograms for the random and G-based designed matrix. For random A, the spread of $\|As\|$ is clearly wider for few measurements m, resulting in $\beta/\alpha \not< 2$. For the designed A, the histogram is more concentrated. Even with as few as $m = 20$ measurements (for MNIST), the designed A satisfies the sufficient condition $\beta/\alpha < 2$ for convergence of the PGD algorithm, thus ensuring stable recovery.

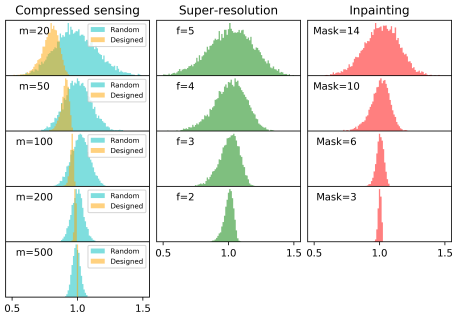


Figure 4: Distribution of $\|As\|$ with different A. Left: Random (cyan) and Designed matrix (orange) with different m. Middle: Downsampling matrix (green) with different f. Right: Inpainting matrix (red) with different mask size.

The middle columns shows the histograms corresponding to the downsampling A that takes the spatial averages of $f \times f$, $f = 2, 3, 4, 5$, pixel values to generate low-resolution images. The right column shows the histograms for the inpainting A that masks out a centered square of various sizes. As expected, with more difficult recovery problems the spread increases. However, for each inverse problem (defined by a matrix A), the ratio β/α can be estimated for *e.g.*, 99.9% of the samples, providing, in combination with Theorem 1, an explicit quantitative guarantee.

5. Experiments

Network Architecture: We implement two GAN architectures: (i) Deep convolutional GAN (DCGAN) [30] for MNIST and CelebA, (ii) Self-attention GAN (SAGAN) [41] for LSUN church-outdoor dataset. DCGAN builds on multiple convolution, transpose convolution, and ReLU layers, and uses batch normalization and dropout for better generalization, whereas SAGAN combines convolutions with self-attention mechanisms in both, generator and discriminator, allowing for long-range dependency modeling to generate images with high-resolution details. For DCGAN, we have used standard objective function of the adversarial loss, whereas for SAGAN, we minimized the hinge version of the adversarial loss [27]. The architecture of the model $G^\dagger$ is similar to that of the discriminator D in the GAN and only differs in the final layer, where we add a fully-connected layer with output size same as the latent variables dimension k. For training $G^\dagger$, we used the architecture shown in Fig. 3 and objective defined in (2), while keeping the pre-trained G fixed. We found that using $\lambda = 0.1$, in (2), gave the best performance. The noise ν used for perturbing the training images $G(z)$ follows $\mathcal{N}(0, \sigma^2 I)$. We observed that training with low σ results in a projector similar to an identity operator and hence only projecting close-by points onto $R(G)$, whereas for large σ the projector violates idempotence. We empirically set $\sigma = 1$. We

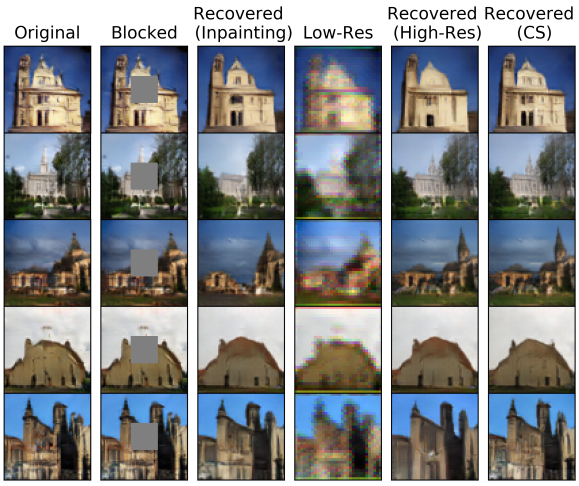


Figure 5: Recovery of LSUN church-outdoor images in inpainting (mask size = 20), super-resolution ($4\times$) and Compressed Sensing (CS, $m = 1000$) tasks.

then obtain a projection network $P_G = GG^\dagger$ that approximately projects images lying outside $R(G)$ onto $R(G)$. We empirically pick latent variable dimension $k = 100$.

MNIST dataset [19] consists of 28×28 greyscale images of digits with $50,000$ training and $10,000$ test samples. We pre-train the GAN consisting of 4 transposed convolution layers for G and 4 convolution layers in the discriminator D using rescaled images lying between $[-1, 1]$. We use $z \sim \mathcal{N}(0, I_k)$ as the G's input. The GAN is trained using the Adam optimizer with learning rate 0.0001, mini-batch size of 128 for 40 epochs. For training the pseudo-inverse of G *i.e.* $G^\dagger$, we minimize the objective (2), using samples generated from $G(z)$, and with the same hyper-parameters used for the GAN.

CelebA dataset [23] consists of more than $200,000$ celebrity images. We use the aligned and cropped version, which preprocesses each image to a size of $64 \times 64 \times 3$ and scaled between $[-1, 1]$. We randomly pick $160,000$ images for training the GAN. Images from the $40,000$ held-out set are used for evaluation. The GAN consists of 5 transposed convolution layers in the G and 5 convolution layers in D. GAN is trained for 35 epochs using Adam optimizer with learning rate 0.00015 and mini-batch size 128. $G^\dagger$ is trained in the same way as for the MNIST dataset.

LSUN church-outdoor dataset [40] consists of more than $126,000$ cropped and aligned images of size $64 \times 64 \times 3$ scaled between $[-1, 1]$. DCGAN generates high-resolution details using spatially local points in lower-resolution feature maps, whereas in SAGAN, details can be generated using information from many feature locations making it a natural choice for diverse dataset such as LSUN. The SAGAN consists of 4 transposed convolution layers and 2 self-attention modules at different scales in G and 4 con-

volution layers and 2 self-attention modules in D. Each self-attention module consists of 3 convolution layers and are added at the $3rd$ and $4th$ layers of the two networks. SAGAN uses conditional batch normalization in G and projection in D. Spectral normalization is used for the layers in both G and D. We use ADAM optimizer with $\beta_1 = 0$ and $\beta_2 = 0.9$, learning rate 0.0001 and mini-batch size 64 for the GAN training. $G^\dagger$, consisting of self-attention mechanism similar to D, is trained using the objective 2 using the ADAM optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.999$, learning rate 0.001 and mini-batch size of 64 for 100 epochs.

We compare the performance of our algorithm on MNIST and CelebA with other GAN-prior solvers ([4, 33]) and sparsity-based methods, Lasso with discrete cosine transform (DCT) basis [34] and total variation minimization method (TVAL3) [21] for linear inverse problems, namely compressed sensing (CS), super-resolution and inpainting. For CS, we extensively evaluate the reconstruction performance with the random Gaussian and designed measurement matrices. Furthermore, we demonstrate the recovery of LSUN church-outdoor dataset images using the proposed method for the different problems in Fig. 5.

5.1. Compressed Sensing

5.1.1 Recovery with random Gaussian matrix

In this set-up, we use the same measurement matrix A as ([4, 33]) *i.e.* $A_{i,j} \sim N(0, 1/m)$ where m is the number of measurements. For MNIST, the measurement matrix $A \in R^{m\times 784}$, with $m = 20, 50, 100, 200$, whereas for CelebA, $A \in R^{m\times 12288}$, with $m = 200, 500, 1000, 2000$. Figure 6 shows the recovery results for MNIST images from the test set. Our NPGD algorithm performs better than others and avoids local optima. Figure 7 shows the reconstruction of eight test images from CelebA. Our algorithm outperforms the other three methods visually as it is able to preserve detailed facial features such as sunglasses, hair and has accurate color tones. Figures 8a and 8c provide a quantitative comparison for MNIST and CelebA, respectively.

5.1.2 Recovery with the designed matrix

In this set-up, we use the G-based designed A described in the section 4.2. We observe that recovery with the designed A is possible for much fewer measurements m. This corroborates our assessment based on Figure 4 that the designed matrix satisfies the desired REC condition with high probability for most of the secants, for smaller m. Figures 8a, 8c show that our algorithm consistently outperforms other approaches in terms of reconstruction error and structural similarity index (SSIM) for a random A. Furthermore, with the designed A, we are able to get performance on-par with the random matrix using 5-10$\times$ smaller m. Figures 8b,8d show the recovered images with the designed and a random A using our algorithm for different m. Clearly, recovery with the random A requires much bigger m than the designed one to achieve similar performance.

[1] Code of Shah *et al.* (PGD-GAN) for MNIST not available

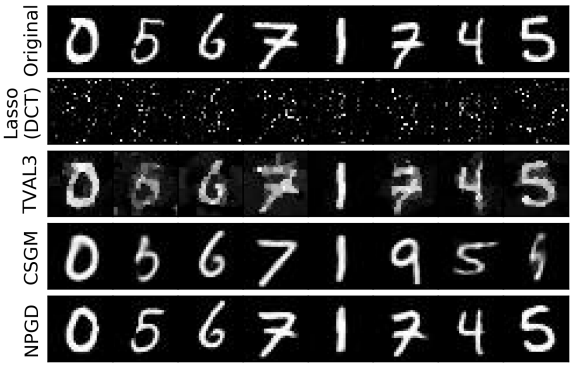


Figure 6: Reconstruction using Gaussian matrix with $m = 100$. [1]

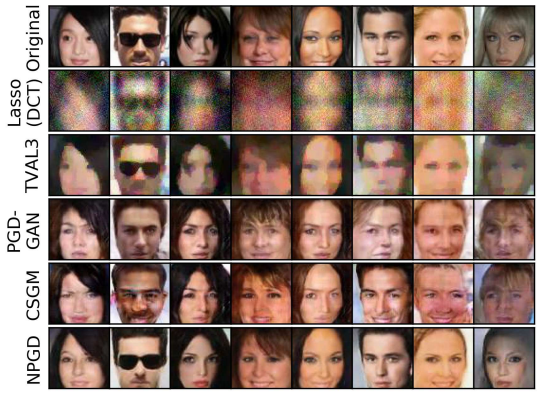


Figure 7: Reconstruction using Gaussian matrix with $m = 1000$.

5.2. Super-resolution

Super-resolution refers to recovering the high-resolution image from a single low-resolution image, often modeled as a blurred and downsampled image of the original. This super-resolution problem is just a special case in our framework of linear measurements. We simulate the blurring+downsampling by taking the spatial averages of $f \times f$ pixel values (in RGB color space), where f is the ratio of downsampling. This corresponds to blurring by an $f \times f$ box impulse response, followed by downsampling. We test our algorithm with $f = 2, 3, 4$, corresponding to $4\times$, $9\times$ and $16\times$-smaller image sizes, respectively. We note that for higher f, the measurement matrix A may not satisfy the desired $REC(S, \alpha, \beta)$ with $\frac{\beta}{\alpha} < 2$ (see figure 4) required for convergence of our algorithm and, consequently, our theorem might not be applicable. Results for MNIST in figure 9a-9c shows that recovery performance indeed degrades with increasing f, however, our NPGD algorithm,

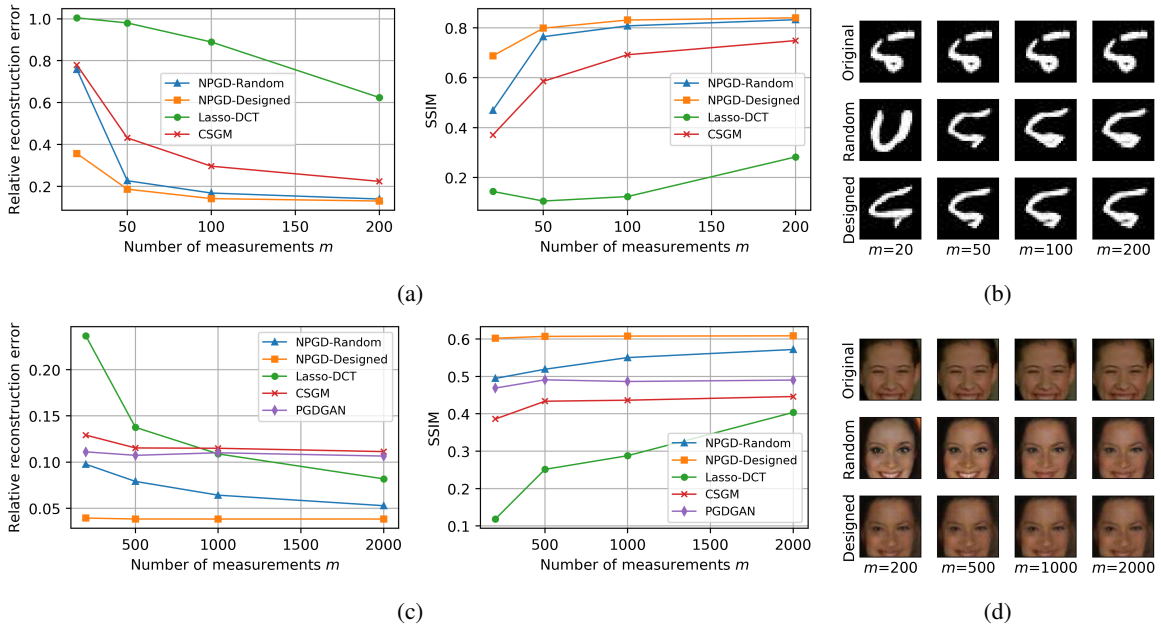
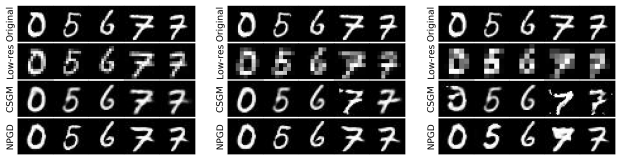
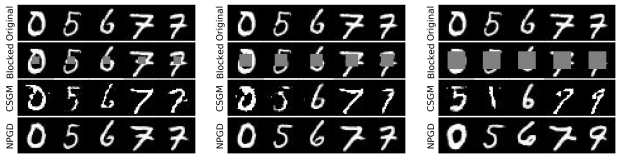


Figure 8: (a) Relative error $\|x^* - \hat{x}\|^2/\|x^*\|^2$ and SSIM of reconstruction algorithms for MNIST dataset with $m = 20, 50, 100, 200$ measurements. (b) MNIST reconstructions with a random Gaussian (middle row) and the designed matrix with orthonormal rows based on G (bottom row) using different m. (c) Relative error and SSIM for CelebA dataset with $m = 200, 500, 1000, 2000$ measurements. (d) CelebA reconstructions, as in (b).



(a) 4× low-res (b) 9× low-res (c) 16× low-res

Figure 9: Super-resolution on MNIST dataset. Row 1: original image x. Row 2: low-resolution images y, upsampled using constant padding, Row 3: high resolution image recovered by [4]. Row 4: high-resolution image recovered by our method.



(a) Mask size = 6 (b) Mask size = 10 (c) Mask size = 14

Figure 10: Inpainting in MNIST dataset. Row 1: original image x. Row 2: image y with center block missing. Row 3: image recovered by [4]. Row 4: image recovered by our method.

gives better reconstructions than Bora *et al.* [4].

5.3. Inpainting

Inpainting refers to recovering the entire image from a partly occluded version. In this case, y is an image with masked regions and A is the linear operation applying a pixel-wise mask to the original image x. Again, this is a special case of linear measurements where each measurement corresponds to an observed pixel. For experiments on the MNIST dataset, we apply a centered square mask of size $6, 10, 14$. Recovery results in figure 10a-10c show that our method consistently outperforms [4] and recovers almost perfectly for mask-size less than 10. The results align with the REC histogram for inpainting (figure 4), which shows that for higher mask-size, the desired REC condition for guaranteed convergence may not be satisfied.

5.4. Comparison of Run-time for Recovery

Table 1 compares the run times of our network-based algorithm NPGD and other recovery algorithms. We record the average run time to recover a single image from its compressed sensing measurements over 10 different images. All three algorithms were run on the same workstation with i7-4770K CPU, 32GB RAM and GeForce Titan X GPU.

5.5. Analysis: Error in Projector

Figure 11 illustrates the idempotence error of the projector for different k. Three different categories of images are tested, namely, MNIST training samples, MNIST test

[2]Run time includes 2 initializations, as implemented by the authors, for CelebA. The same number of initializations for CelebA (and 10 for MNIST) has been used to produce results in figures 6, 7, 8, and 9. Our NPGD algorithm uses only one, deterministic initialization, $x_0 = A^T y$.

m	CSGM [2]	PGD-GAN	NPGD
200	5.8	66	0.09 (64x)
500	6.6	60	0.10 (66x)
1000	8.0	63	0.11 (72x)
2000	11.2	61	0.14 (80x)

Table 1: Comparison of execution time ([sec.]) of recovery algorithms on the CelebA dataset. The relative speedup of our NPGD over the CSGM algorithm of Bora *et al*. is shown in parenthesis.

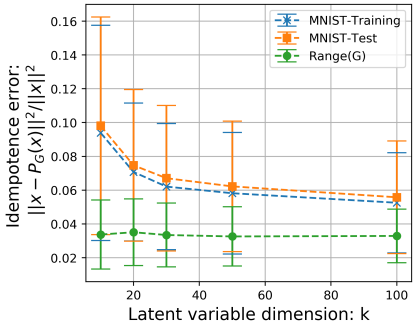


Figure 11: Idempotence Error

samples, and samples $G(z)$ generated using the pre-trained G. We use clean images from the three sources and plot the relative idempotence error $\|x - P_G(x)\|^2/\|x\|^2$. The error decreases with increasing k and saturates around $k = 100$. The idemopotence errors for MNIST training and test samples are very close, indicating negligible generalization error. On the other hand, samples generated by $G(z)$ give much lower errors, which indicates representation error in the GAN. Thus we expect that a more flexible generator (deeper network) will lead to a better projector on the actual dataset and hence improve performance.

6. Conclusion

In this work, we propose a GAN based projection network for faster recovery in linear inverse problems. Our method demonstrates superior performance and also provides a speed-up of 60-80$\times$ over existing GAN-based methods, eliminating the expensive computation of the Jacobian matrix every iteration. We provide a theoretical bound on the reconstruction error for a moderately-conditioned measurement matrix. To help design such a matrix for compressed sensing, we propose a method which enables recovery using 5-10$\times$ fewer measurements than using a random Gaussian matrix. Our experiments on compressed sensing, super-resolution, and inpainting demonstrate that generic linear inverse problems can be solved with the proposed method without requiring retraining. In the future, deriving a bound for the projection error δ and an associated performance guarantee is a interesting direction.

A. Appendix: Proof of Theorem 1

By the assumption of δ-approximate projection,

$$\|w_t - x_{t+1}\|^2 = \|w_t - G(G^\dagger(w_t))\|^2 \leq \|x^* - w_t\|^2 + \delta \tag{10}$$

where from the gradient update step, we have

$$w_t = x_t - \eta A^T(Ax_t - y) = x_t - \eta A^T A(x_t - x^*)$$

Substituting w_t into (10) yields

$$\begin{aligned}&\|x_{t+1} - x_t\|^2 - 2\eta\left\langle x_{t+1} - x_t, A^TA\left(x^* - x_t\right)\right\rangle \\ &\leq \|x^* - x_t\|^2 - 2\eta\|A(x^* - x_t)\|^2 + \delta\end{aligned}$$

Rearranging the terms we have

$$\begin{aligned}&2\left\langle x_t - x_{t+1}, A^TA\left(x^* - x_t\right)\right\rangle \\ &\leq \frac{1}{\eta}\|x^* - x_t\|^2 - 2f\left(x_t\right) - \frac{1}{\eta}\|x_{t+1} - x_t\|^2 + \frac{\delta}{\eta} \\ &\leq \left(\frac{1}{\eta\alpha} - 2\right)f(x_t) - \frac{1}{\eta}\|x_{t+1} - x_t\|^2 + \frac{\delta}{\eta} \\ &\leq \left(\frac{1}{\eta\alpha} - 2\right)f\left(x_t\right) - \frac{1}{\eta\beta}\|Ax_{t+1} - Ax_t\|^2 + \frac{\delta}{\eta}\end{aligned} \tag{11}$$

where the last two inequalities follow from $REC(S, \alpha, \beta)$. Now the LHS can be rewritten as:

$$\begin{aligned}&2\left\langle x_t - x_{t+1}, A^TA\left(x^* - x_t\right)\right\rangle \\ &= \|Ax^* - Ax_{t+1}\|^2 - \|Ax^* - Ax_t\|^2 - \|Ax_{t+1} - Ax_t\|^2 \\ &= f(x_{t+1}) - f(x_t) - \|Ax_{t+1} - Ax_t\|^2\end{aligned} \tag{12}$$

Combining (11) and (12), and rearranging the terms, we have:

$$f(x_{t+1}) \leq \left(\frac{1}{\eta\alpha} - 1\right)f(x_t) + \left(1 - \frac{1}{\eta\beta}\right)\|Ax_{t+1} - Ax_t\|_2^2 + \frac{\delta}{\eta}$$

and since $\eta = 1/\beta$,

$$f(x_{t+1}) \leq \left(\frac{\beta}{\alpha} - 1\right)f(x_t) + \beta\delta$$

For simplicity, we substitute $\kappa = \beta/\alpha$ in the following:

$$\begin{aligned}f(x_n) &\leq (\kappa - 1)^n f\left(x_0\right) + \beta\delta\sum_{k=0}^{n-1}(\kappa - 1)^k \\ &= (\kappa - 1)^n f\left(x_0\right) + \frac{\beta\left(1 - (\kappa - 1)^n\right)}{2 - \kappa}\delta\end{aligned}$$

For convergence, we require $1 \leq \kappa = \beta/\alpha < 2$. When n reaches $\frac{1}{2-\kappa}\log\left(\frac{f(x_0)}{C\alpha\delta}\right)$, we have

$$\begin{aligned}\|x_n - x^*\|^2 &\leq \frac{\|Ax_n - Ax^*\|^2}{\alpha} = \frac{f(x_n)}{\alpha} \\ &\leq (\kappa - 1)^n\frac{f(x_0)}{\alpha} + \frac{\beta\left(1 - (\kappa - 1)^n\right)}{\alpha(2 - \kappa)}\delta \\ &\leq (\kappa - 1)^n\frac{f(x_0)}{\alpha} + \frac{\delta}{2/\kappa - 1} \leq \left(C + \frac{1}{2/\kappa - 1}\right)\delta\end{aligned}$$

Finally, when $n \to \infty$, we have $(\kappa - 1)^n\frac{f(x_0)}{\alpha} \to 0$

$$\|x^* - x_\infty\|^2 \leq \frac{\delta}{2/\kappa - 1} = \frac{\delta}{2\alpha/\beta - 1}$$

References

[1] Jonas Adler and Ozan Öktem. Solving ill-posed inverse problems using iterative deep neural networks. *Inverse Problems*, 33(12):124007, 2017. 1

[2] Richard G Baraniuk and Michael B Wakin. Random projections of smooth manifolds. *Foundations of computational mathematics*, 9(1):51–77, 2009. 4

[3] David Berthelot, Thomas Schumm, and Luke Metz. Began: Boundary equilibrium generative adversarial networks. *arXiv preprint arXiv:1703.10717*, 2017. 2

[4] Ashish Bora, Ajil Jalal, Eric Price, and Alexandros G Dimakis. Compressed sensing using generative models. *arXiv preprint arXiv:1703.03208*, 2017. 2, 3, 6, 7

[5] Emmanuel J Candes, Justin K Romberg, and Terence Tao. Stable signal recovery from incomplete and inaccurate measurements. *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, 59(8):1207–1223, 2006. 1

[6] Antonia Creswell, Tom White, Vincent Dumoulin, Kai Arulkumaran, Biswa Sengupta, and Anil A Bharath. Generative adversarial networks: An overview. *IEEE Signal Processing Magazine*, 35(1):53–65, 2018. 2

[7] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Bm3d image denoising with shape-adaptive principal component analysis. In *SPARS'09-Signal Processing with Adaptive Sparse Structured Representations*, 2009. 1

[8] Weisheng Dong, Lei Zhang, Guangming Shi, and Xiaolin Wu. Image deblurring and super-resolution by adaptive sparse domain selection and adaptive regularization. *IEEE Transactions on Image Processing*, 20(7):1838–1857, 2011. 1

[9] David L Donoho. Compressed sensing. *IEEE Transactions on information theory*, 52(4):1289–1306, 2006. 1

[10] Michael Elad. *Sparse and redundant representations: from theory to applications in signal and image processing*. Springer Science & Business Media, 2010. 1

[11] Michael Elad and Michal Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image processing*, 15(12):3736–3745, 2006. 1

[12] Kai Fan, Qi Wei, Lawrence Carin, and Katherine A Heller. An inner-loop free solution to inverse problems using deep neural networks. In *Advances in Neural Information Processing Systems*, pages 2370–2380, 2017. 1

[13] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014. 2

[14] Karol Gregor and Yann LeCun. Learning fast approximations of sparse coding. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, pages 399–406. Omnipress, 2010. 1

[15] Harshit Gupta, Kyong Hwan Jin, Ha Q Nguyen, Michael T McCann, and Michael Unser. Cnn-based projected gradient descent for consistent ct image reconstruction. *IEEE transactions on medical imaging*, 37(6):1440–1453, 2018. 1

[16] C. Hegde, A. C. Sankaranarayanan, W. Yin, and R. G. Baraniuk. Numax: A convex approach for learning near-isometric linear embeddings. *IEEE Transactions on Signal Processing*, 63(22):6109–6121, Nov 2015. 4

[17] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 2

[18] Henry Kvinge, Elin Farnell, Michael Kirby, and Chris Peterson. A gpu-oriented algorithm design for secant-based dimensionality reduction. In *2018 17th International Symposium on Parallel and Distributed Computing (ISPDC)*, pages 69–76. IEEE, 2018. 4

[19] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. 5

[20] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4681–4690, 2017. 2

[21] Chengbo Li, Wotao Yin, and Yin Zhang. User's guide for tval3: Tv minimization by augmented lagrangian and alternating direction algorithms. *CAAM report*, 20(46-47):4, 2009. 1, 6

[22] Ding Liu, Bihan Wen, Xianming Liu, Zhangyang Wang, and Thomas S Huang. When image denoising meets high-level vision tasks: A deep learning approach. *arXiv preprint arXiv:1706.04284*, 2017. 1

[23] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2015. 5

[24] Morteza Mardani, Enhao Gong, Joseph Y Cheng, Shreyas S Vasanawala, Greg Zaharchuk, Lei Xing, and John M Pauly. Deep generative adversarial neural networks for compressive sensing mri. *IEEE transactions on medical imaging*, 38(1):167–179, 2019. 2

[25] Morteza Mardani, Qingyun Sun, David Donoho, Vardan Papyan, Hatef Monajemi, Shreyas Vasanawala, and John Pauly. Neural proximal gradient descent for compressive imaging. In *Advances in Neural Information Processing Systems*, pages 9596–9606, 2018. 1

[26] Chris Metzler, Ali Mousavi, and Richard Baraniuk. Learned d-amp: Principled neural network based compressive image recovery. In *Advances in Neural Information Processing Systems*, pages 1772–1783, 2017. 1

[27] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*, 2018. 5

[28] Ali Mousavi, Gautam Dasarathy, and Richard G Baraniuk. Deepcodec: Adaptive sensing and recovery via deep convolutional neural networks. *arXiv preprint arXiv:1707.03386*, 2017. 1

[29] Ali Mousavi, Gautam Dasarathy, and Richard G Baraniuk. A data-driven and distributed approach to sparse signal representation and recovery. 2018. 1

[30] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015. 5

[31] JH Rick Chang, Chun-Liang Li, Barnabas Poczos, BVK Vijaya Kumar, and Aswin C Sankaranarayanan. One network to solve them all–solving linear inverse problems using deep projection models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5888–5897, 2017. 1

[32] Yaniv Romano, Michael Elad, and Peyman Milanfar. The little engine that could: Regularization by denoising (red). *SIAM Journal on Imaging Sciences*, 10(4):1804–1844, 2017. 1

[33] Viraj Shah and Chinmay Hegde. Solving linear inverse problems using gan priors: An algorithm with provable guarantees. *arXiv preprint arXiv:1802.08406*, 2018. 2, 3, 6

[34] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996. 6

[35] Singanallur V Venkatakrishnan, Charles A Bouman, and Brendt Wohlberg. Plug-and-play priors for model based reconstruction. In *Global Conference on Signal and Information Processing (GlobalSIP), 2013 IEEE*, pages 945–948. IEEE, 2013. 1

[36] Bihan Wen, Saiprasad Ravishankar, and Yoram Bresler. Structured overcomplete sparsifying transform learning with convergence guarantees and applications. *International Journal of Computer Vision*, 114(2-3):137–167, 2015. 1

[37] Bihan Wen, Saiprasad Ravishankar, Luke Pfister, and Yoram Bresler. Transform learning for magnetic resonance image reconstruction: From model-based learning to building neural networks. *arXiv preprint arXiv:1903.11431*, 2019. 1

[38] Jianchao Yang, John Wright, Thomas S Huang, and Yi Ma. Image super-resolution via sparse representation. *IEEE transactions on image processing*, 19(11):2861–2873, 2010. 1

[39] Raymond A Yeh, Chen Chen, Teck Yian Lim, Alexander G Schwing, Mark Hasegawa-Johnson, and Minh N Do. Semantic image inpainting with deep generative models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5485–5493, 2017. 2

[40] Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, and Jianxiong Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015. 5

[41] Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-attention generative adversarial networks. *arXiv preprint arXiv:1805.08318*, 2018. 5

[42] Jun-Yan Zhu, Philipp Krähenbühl, Eli Shechtman, and Alexei A Efros. Generative visual manipulation on the natural image manifold. In *European Conference on Computer Vision*, pages 597–613. Springer, 2016. 2