Intermediate Layer Optimization for Inverse Problems using Deep Generative Models

The University of Texas at Austin

Joseph Dean Giannis Daras Alexandros Dimakis

NEURAL INFORMATION PROCESSING SYSTEMS

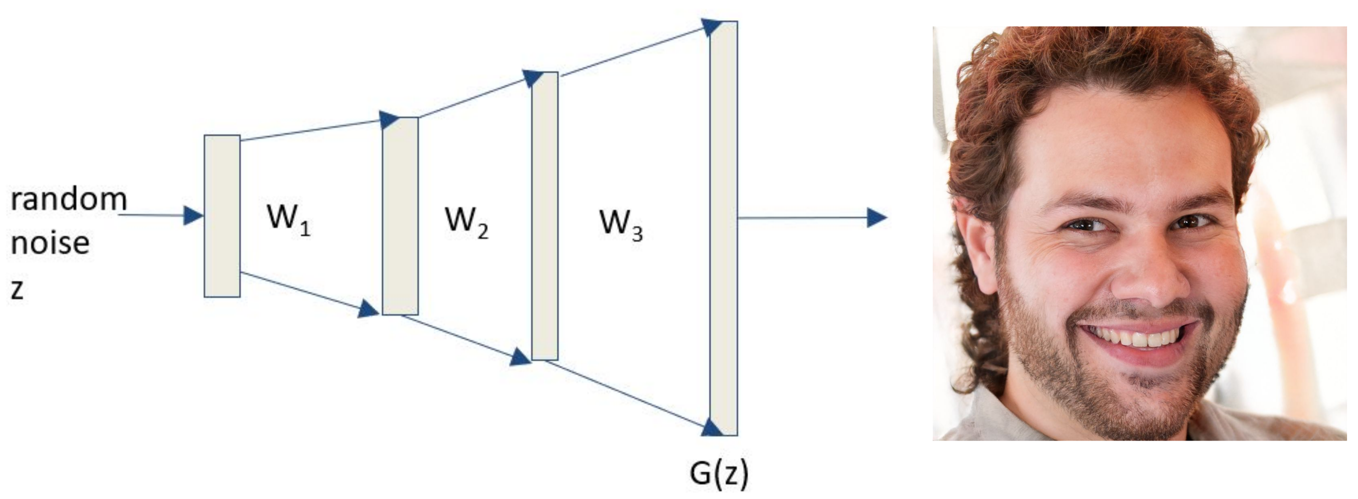
Inverting Deep Generative Models

$$\min_{z} \operatorname{dist}\left(f(G(z)), f(x)\right) + R(z)$$

- The goal of inversion is to use gradient descent to recover a latent code that approximates an image as well as possible.
- A degradation operator f can be used to perform unsupervised image recovery tasks such as inpainting, super-resolution, denoising, and compressed sensing.
- Inverting very deep generators (BigGAN or StyleGAN2) becomes increasingly difficult, as inverting a generator with 4 layers is NP-hard.

What if the image does not lie in the latent space?

Can the optimization process be further improved?



Perceptual loss function

- Prior work mainly focuses on MSE for loss functions. LPIPS has been shown to be a better perceptual-based loss function in image similarity
- We find that a weighted combination of MSE and LPIPS yields optimal reconstructions for all image recovery tasks
- We show that many of the biases inherent in previous inversion methods can be addressed by introducing LPIPS to the loss function

Original LPIPS MSE

Intermediate Layer Optimization (ILO)

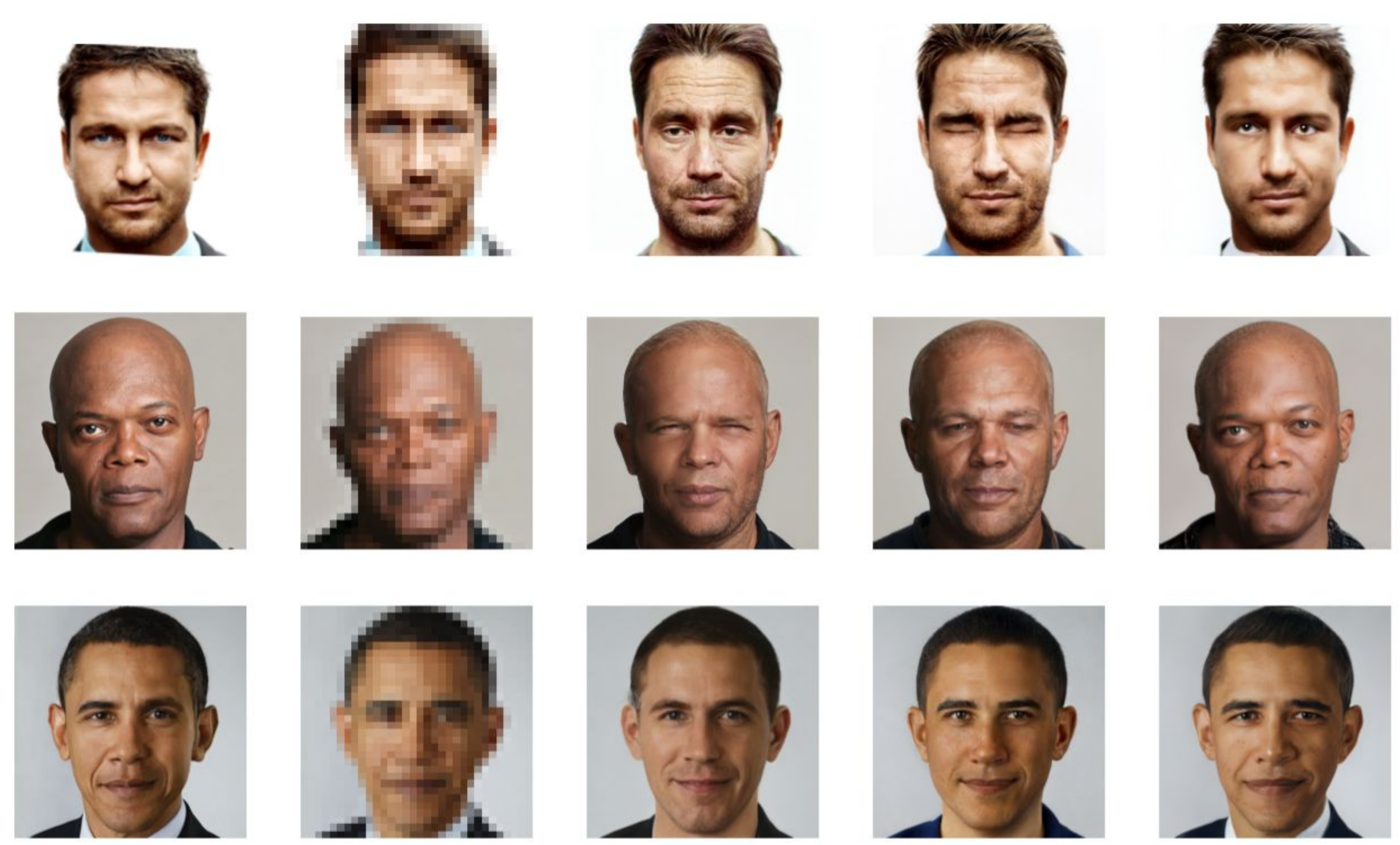
- Fine-grained image reconstructions are obtained through iterative optimization over the layers of the generator
- The algorithm runs in rounds, where in each round the previous layer is discarded and optimize over this newly defined generator
- By removing layers one at a time, we introduce flexibility in the generation of images that more easily match the observed image
- We switch to the next layer when the loss function flattens, which can be done in an unsupervised manner
- The algorithm consists of the following steps:
  I. Given a deep generator: $G = (g_n \circ g_{n-1} \dots \circ g_2 \circ g_1)(z_1)$ we run gradient descent to find the latent vector
  II. We then discard the previous layer and optimize over the generator defined as: $(g_n \circ g_{n-1} \dots \circ g_2)(z_2)$ while $z_2 = g_1(z_1)$ initializing
  III. This process repeats until the MSE loss becomes very small

Experiment Setup

- Experiments are performed using the StyleGAN2 generator
- We utilize a ramped-down learning scheduler with an initial learning rate of 0.1
- The regularization term R(z) is defined as the geodesic loss of all 18 latent vectors
- In each instantiation of a generator in any round of the ILO algorithm, we optimize over all the latent vectors and the next 5 noises

Super-resolution

- We show the effect of different loss functions and compare our results to PULSE
- We notice that ILO leads to less biased reconstructions since it expands the expressive range of the generators by optimizing intermediate layers



Original LR (×16) PULSE LPIPS Ours

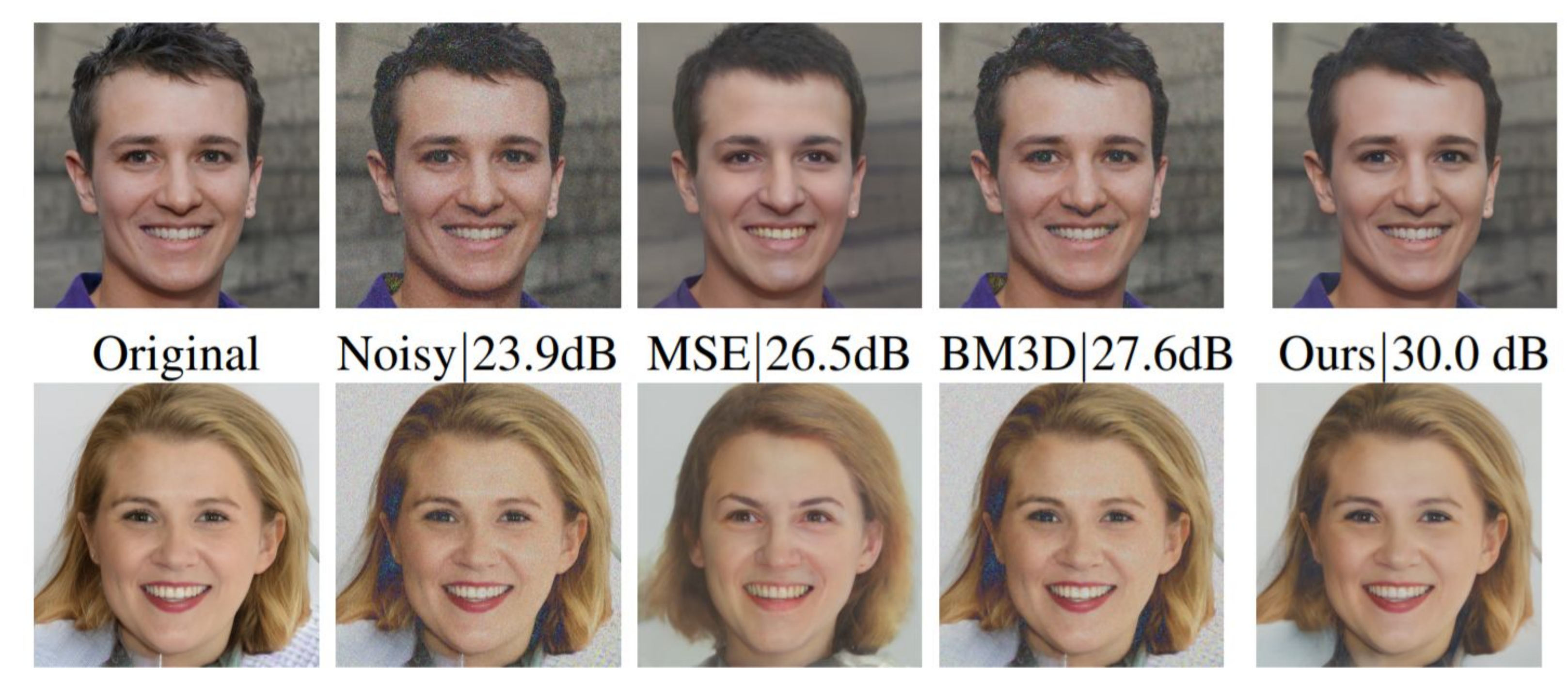
Inpainting

$$\min_{z} \|M \odot I - M \odot G(z)\|_2^2 + \alpha \cdot \mathrm{LPIPS}\left(M \odot I + M^C \odot G(z), G(z)\right) + \beta \cdot R(z)$$

Original Observation PULSE (MSE) LPIPS MSE+LPIPS Ours

Denoising

$$\min_{z} \|I - (G(z) + \mathcal{N}(0,\sigma^2))\|_2^2 + \alpha \cdot \mathrm{LPIPS}(I, G(z) + \mathcal{N}(0,\sigma^2)) + \beta \cdot R(z)$$



Original Noisy|23.9dB MSE|26.5dB BM3D|27.6dB Ours|30.0 dB

Original Noisy|21.8dB MSE|23.0dB BM3D|24.4dB Ours|27.1dB

Future Work

- Can theoretical results be established in the same setting as proposed by CSGM?
- Can we robustly determine when to move from layer to layer and how to adjust the learning rate