IMAGE RESTORATION WITH DEEP GENERATIVE MODELS

Raymond A. Yeh[*] *Teck Yian Lim*[*], *Chen Chen*
Alexander G. Schwing, Mark Hasegawa-Johnson, Minh N. Do

University of Illinois at Urbana Champaign, Champaign, IL, USA
Department of Electrical and Computer Engineering
{yeh17, tlim11, cchen156, aschwing, jhasegaw, minhdo}@illinois.edu

ABSTRACT

Many image restoration problems are ill-posed in nature, hence, beyond the input image, most existing methods rely on a carefully engineered image prior, which enforces some local image consistency in the recovered image. How tightly the prior assumptions are fulfilled has a big impact on the resulting task performance. To obtain more flexibility, in this work, we proposed to design the image prior in a data-driven manner. Instead of explicitly defining the prior, we learn it using deep generative models. We demonstrate that this learned prior can be applied to many image restoration problems using an unified framework.

***Index Terms*—** deep generative models, generative adversarial networks, image restoration.

1. INTRODUCTION

Image restoration is the task of restoring the missing or damaged information from a corrupted image. Here, we let the term image restoration subsume tasks such as image inpainting, image colorization, super-resolution, denoising, and inverse quantization to name a few. Due to the ill-posed nature of the problem, restoration algorithms require a parametric image generation model and a choice of image prior. For instance, existing methods, such as, total variation (TV) [1] uses the smoothness property of natural images. Other methods, *e.g.*, low rank (LR) [2] and Sparse methods [3, 4] use prior knowledge from image patches.

To this end, existing methods, generally employ a data term to enforce consistency with observations, and a regularization term to encourage prior assumptions. Investigating all of the aforementioned formulations more closely, we observe that the data term and the prior term are often chosen independently of each other. This independent choice may result in suboptimal performance for the respective task.

Therefore, we proposed to use deep generative models to learn the prior in a data-driven manner.

In particular, we leverage the recent advances in Generative Adversarial Nets (GANs) [5]. Given a GAN trained for a particular task, we develop and demonstrate a general framework to adapt the GAN for image restoration, including image inpainting, colorization, super-resolution, denoising, and inverse quantization. Our approach is based on the maximum a posteriori (MAP) estimation framework. Similar to MAP, we reconstruct the image by minimize a loss function and a regularization term. But instead of minimizing in the high dimensional image space, we operate in the latent space defined by the GAN.

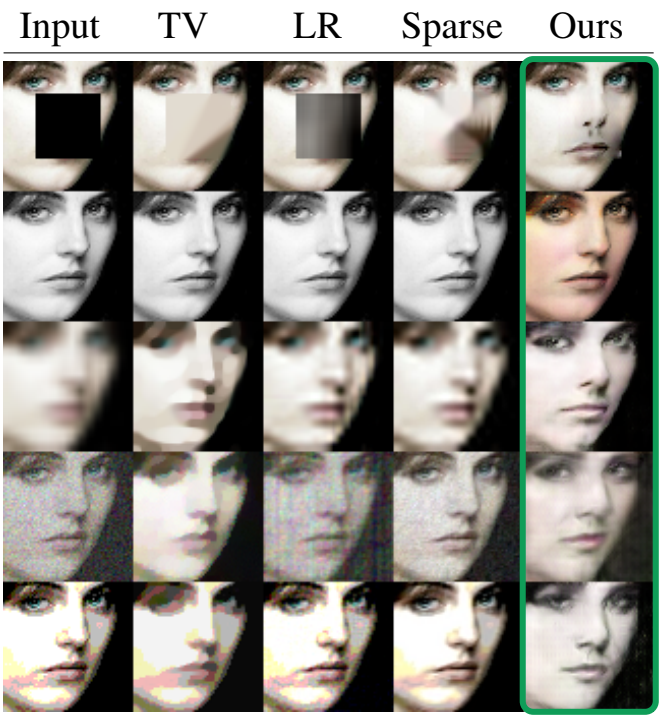


Fig. 1. Image restoration results of different methods. Each row corresponds to a restoration task in the following order: semantic inpainting, colorization, super-resolution, denoising, and intensity quantization.

We evaluate our framework on the CelebA dataset [6] and demonstrate promising reconstruction on different tasks when compared to the aforementioned approaches.

[*]Indicating Equal Contribution. Image samples and software will be released on author's website at tlim11.web.engr.illinois.edu.

2. RELATED WORK

2.1. Deep neural network for image restoration

Recently, deep convolutional neural nets have shown remarkable results on image restoration tasks. For example, single image super-resolution [7, 8, 9, 10], image inpainting [11, 12, 13], and denoising [14, 12]. These works typically treat image restoration as a regression problem (*i.e.*, given the corrupted image predict the clean image). Therefore, each of these models requires training of a deep net for each type of data corruption. In [15], Yeh *et al*. take a different approach. Using a trained generative adversarial network, they demonstrate promising results on the task of image inpainting with various noise distributions and without the need of retraining. In this work, we extend this method to general image restoration and provide a more formal justification of the approach.

2.2. Maximum a posterior (MAP) Estimation

Let x refer to the original, unobserved image, let y denote the observation, and let A be a generative operator which models the measurement process. The image generation model is then given by $y = Ax + \epsilon$, with independent and identically distributed noise. To recover an estimate $\hat{x}$ of the original, unobserved image x, we consider the classical program:

$$\hat{x} = \arg\min_x \|y - Ax\|_p - \lambda \log(\tilde{p}_{\mathbf{X}}(x)). \tag{1}$$

The squared norm term of the cost function is interpreted as the likelihood function and the second term is the prior term. Depending on the task, different choices of generative operators A and priors $\tilde{p}_{\mathbf{X}}(x)$ can be chosen. Our choice for the prior is motivated by recently introduced generative adversarial nets (GANs) [5], which are introduced next.

2.3. Generative Adversarial Networks (GANs)

Generative adversarial nets (GANs) are a framework for training generative models through an adversarial process. The framework jointly trains a generator, G, which attempts to synthesize data from the data distribution, $p_{\mathbf{X}}$, and a discriminator D, to estimate the probability that a given sample originates from the data distribution rather than being synthesized from G. The G and D networks are trained by optimizing the following cost function, which pits the generator against the discriminator in a non-cooperative two-player game:

$$\min_G \max_D V(G, D) := \mathbb{E}_{x \sim p_{\mathbf{X}}}[\log(D(x))] + \mathbb{E}_{z \sim p_{\mathbf{Z}}}[\log(1 - D(G(z))]. \tag{2}$$

Hereby, x is a sample from the data distribution $p_{\mathbf{X}}$, and z is a random encoding on the latent space sampled from a prior $p_{\mathbf{Z}}$. We'll next describe how to combine GANs with MAP estimation to obtain a unified framework for image restoration tasks.

3. IMAGE RESTORATION BY CONSTRAINED IMAGE GENERATION

The MAP formulation in Eq. (1), without the regularization, is considered ill-posed because the degradation operator A is under-determined, *i.e.*, several images x can map to an observation y (surjective mapping). Therefore, in a MAP framework, the prior term plays a crucial role in regularizing the ill-posed problem of image restoration. The ideal prior is the data distribution $p_{\mathbf{X}}$, *i.e.*, the probability density function (pdf) of the data x. However, this prior is typically not available and for data such as images, it is also difficult to learn or define analytically.

In recent years, GANs have been used successfully to generate realistic samples for imaging data. However, unlike generative models such as variational auto-encoders [16], GANs are not a probabilistic model, even though one can prove that upon convergence, GANs generate samples from the data distribution. This means that the data distribution $p_{\mathbf{X}}$ is captured implicitly.

Since the data distribution is not readily available, we subsequently develop a surrogate mechanism which utilizes this implicit prior for image restoration. More specifically, we develop a technique which concurrently employs the trained generator net G and the trained discriminator net D.

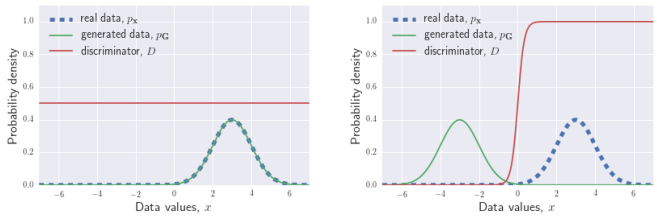
Our approach follows in spirit the aforementioned MAP technique, *i.e.*, we also minimize a loss and a regularization term. But instead of optimizing in the data domain, *i.e.*, instead of searching for images x as done in Eq. (1), we optimize in the embedding space to find an encoding z which can then be used to reconstruct the image $\hat{x} = G(\hat{z})$. To find this embedding we use the following program:

$$\hat{z} = \arg\min_z \|y - A(G(z))\|_p + \lambda \Big(\log(1 - D(G(z))) - \log(D(G(z)) + \log(p_{\mathbf{z}}(z)) \Big), \tag{3}$$

where the first term $\|y - A(G(z))\|$ denotes the reconstruction loss, and the second term is the regularizer. Again, A is the measurement operator, *e.g.*, the corruption mask, the downsampling operator, the quantization operator, *etc*. Empirically, we found that $p = 1$ performs well in practice.

Intuitively, the loss in Eq. (3) encourages the corrupted reconstructed image, $A(G(z))$, to closely fit the observation y. Instead of maximizing the unavailable log-probability directly, we suggest to maximize the surrogate logit $\log D(G(z))$, which corresponds to minimizing $\log(1 - D(G(z)))$. To help understand this surrogate regularizer, we visually illustrate, in Fig. 2, the generator G, the discriminator D, and the data distribution $p_{\mathbf{X}}$, assuming, for illustration purposes only, that G and $p_{\mathbf{X}}$ follow a Gaussian distribution. We let $p_{\mathbf{G}}$ denote the distribution of the samples $G(z)$ when drawing embeddings z from some distribution $z \sim p_{\mathbf{Z}}$.

Assume that the discriminator D and the generator G are both trained to optimality. It can be shown [5], that, in



(a) (b)

Fig. 2. One dimensional illustration of generator, discriminator, and the ground-truth distribution. (a) When generator and discriminator are trained to optimality. (b) When generator is not optimal, but discriminator is optimal.

this case, $p_{\mathbf{G}} = p_{\mathbf{X}}$ and $D = 0.5$ for all samples; where $p_{\mathbf{G}}$ denotes the distribution induced on the data space by the transformation of the generator G. We illustrate this case in Fig. 2 (a). Note that, in this case, the regularizer has no effect since its derivative is zero. This is intuitive since the artificial samples $G(z)$ are guaranteed to follow the data distribution and we refrain from biasing to highly likely data configurations.

When G is not trained to optimality, as shown in Fig. 2 (b), it can be seen that, when $\hat{x}$ is sampled from $p_{\mathbf{G}}$, then the regularizer pushes $\hat{x}$ in the direction of $p_{\mathbf{X}}$. The strength of the push is determined by the hyperparameter λ.

To solve the program given in Eq. (3), we use gradient descent based methods to back-propagate to the latent space z, while choosing the initial encoding z randomly from a uniform distribution.

3.1. Justification of regularizer

In the ideal case, we would like to use $p_{\mathbf{X}}(x)$ as the prior term. From [5], for a fixed G, the optimal discriminator D for a given generator G is

$$D^*(x) = \frac{p_{\mathbf{X}}(x)}{p_{\mathbf{X}}(x) + p_{\mathbf{G}}(x)}, \tag{4}$$

Rearranging terms,

$$\begin{aligned}\log(p_{\mathbf{X}}(x)) = \log(D(x)) - \log(1 - D(x)) \\ + \log(p_{\mathbf{Z}}(z)) + \log\left(\left|\frac{\partial z}{\partial x}\right|\right),\end{aligned} \tag{5}$$

where $p_{\mathbf{G}}(x) = p_{\mathbf{Z}}(z)\left|\frac{\partial z}{\partial x}\right|$. Since $\left|\frac{\partial z}{\partial x}\right|$ is intractable to compute, we assume it to be constant.

If we choose $p_{\mathbf{z}}$ to be uniform, then the regularizer depends only on $\log(D(x))$ and $-\log(1 - D(x))$. Both terms have the same gradient direction with respect to x. To optimize the program given in Eq. (3) we use a gradient based approach. Hence, our regularizer provides the correct prior guidance whenever the scalar projection of $\nabla_x p_{\mathbf{G}}(x)$ onto $\nabla_x p_{\mathbf{X}}(x)$ is greater than 0.

To build some intuition, consider the one dimensional case in Fig. 2 (b), where both $p_{\mathbf{G}}$ and $p_{\mathbf{X}}$ are unit variance Gaussian with mean at -3 and 3 respectively. In this case, if $x \in (-\infty, 3)$, then our regularizer is useful; note that x being in this range is likely as our reconstructed image is sampled from G.

3.2. Applications and Implementation Details

Ideally, we would like A to reflect the function that generates the corrupted data. However, in order to solve Eq. (3) with gradient based methods, the corruption operator A also needs be sub-differentiable. We discuss the choice of A for each type of corruption next.

Image Inpainting: We chose A to be the masking function. Given a mask M indicating whether a pixel is corrupted, A is an element-wise multiplication of M and $G(z)$, as also used in [15].

Image Colorization: We chose A to be the function which converts RGB to HSV, then selects the V channel. Using this A, we encourage the gray-scale version of the reconstruction image to be close to the corrupted input.

Image Super Resolution: We chose A to be the downsampling operation, meaning a low pass filter followed by a decimation.

Image Denoising: We chose A to be the identity function.

Image Quantization: We chose A to be the identity function. Ideally, we should choose A to be the step function used in the quantization, however this function provides no meaningful gradients. We therefore approximate it with an identity function.

Post processing: For the task of inpainting, we follow [15] and apply Poisson blending [17] to combine the raw generated image with the original image. For all other tasks, blending techniques could further improve SSIM or raise PSNR. Nonetheless we chose to not perform any form of post-processing as the generated output already performs competitively in terms of PSNR and SSIM, while greatly surpassing traditional methods in visual quality.

GAN archiecture: We use the popular DCGAN model architecture [18] in all our experiments. At each layer in G this architecture halves the number of channels and doubles the spatial dimension. The discriminator D is constructed in reverse order. To train this GAN, we use Adam [19] as the optimizer and augment the data with random horizontal flips.

4. EXPERIMENTS

4.1. Dataset & Corruption Process

We evaluated our method on the CelebFaces Attributes Dataset (CelebA) [6]. The dataset contains 202,599 images of celebrity faces of varying viewpoints, expressions, accessories and hairstyles, see samples in Fig. 4. The images are aligned algorithmically such that faces are centered and of similar sizes. These images were then cropped to 64×64 patches that consists mainly of the face of the subject. From the 202,599 images, we keep 2000 images for testing.

Applications Metric	Inpainting SSIM PSNR	Colorization SSIM PSNR	Super Res SSIM PSNR	Denoising SSIM PSNR	Quantization SSIM PSNR
TV	0.7647 23.10	- -	0.6648 21.05	0.7373 21.97	0.6312 20.77
LR	0.6644 16.98	- -	**0.6754 21.45**	0.6178 18.69	0.6754 20.65
Sparse	0.7528 20.67	- -	0.6075 20.82	**0.8092 23.63**	**0.7869 22.67**
Ours	**0.8121 23.60**	0.8876 20.85	0.5626 19.58	0.6161 19.31	0.6061 19.77

Table 1. Quantitative comparison on image restoration tasks using SSIM and PSNR(dB).

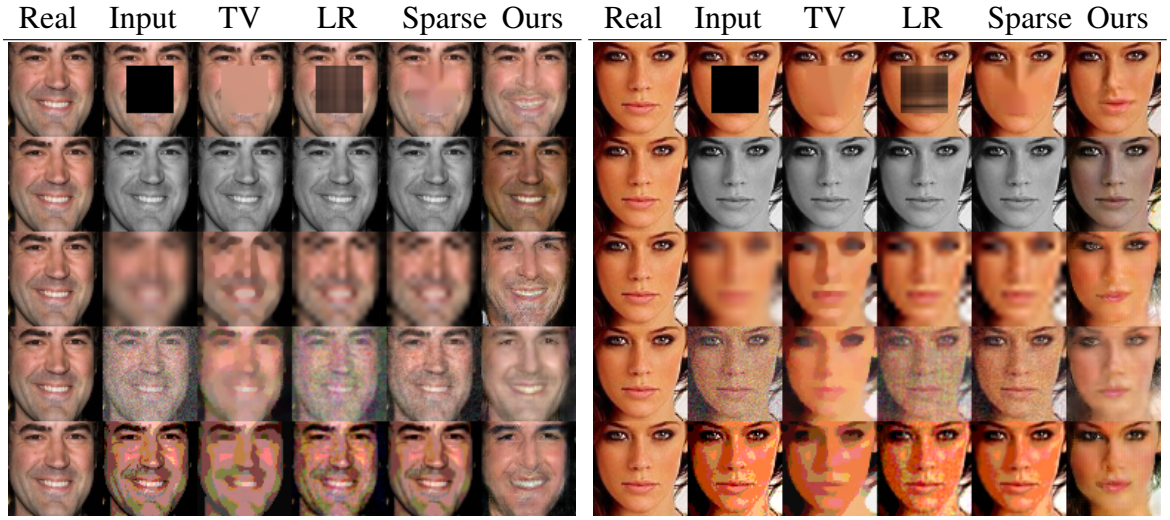


Fig. 3. Qualitative comparison on image restoration tasks. Each row corresponds to a task in the following order: semantic inpainting, colorization, super-resolution, denoising, and quantization.

Fig. 4. Samples from the CelebA dataset illustrating the variety of face images.

For both the training and test dataset, we generated pairs of clean and corrupted examples using the following corruption process (see Fig. 3 for examples): **Semantic Inpainting:** The corruption method is a missing center patch of 32×32; **Colorization:** The corruption is the standard grayscale conversion; **Super Resolution:** The corruption corresponds to downsampling by a factor of 4; **Denoising:** The corruption applies additive Gaussian noise, with standard deviation of 0.1 (pixel intensities from 0 to 1); **Quantization:** The corruption quantizes with 5 discrete levels per channel.

4.2. Comparisons to baseline approaches

We compare the SSIM [20] and PSNR values of our results to the traditional approaches. Using these metrics, it would appear that our method performs poorly. However, visual inspection of the generated images demonstrates that our method generates a far more realistic image when compared to other approaches. Furthermore, traditional methods are not suitable for the colorization task without significant modification. Therefore we left out the comparison for this restoration task. We also provide visual comparisons to TV, sparse coding, LR, and our method in Fig. 3. Our method recovers more realistic and natural images in all the presented restoration tasks.

Our method performs particularly well when significant amount of information needs to be generated. Although we do not perform well on the pixel difference based metrics, we generate images that are closer conceptually, visually more pleasing and more natural looking.

4.3. Discussions

The empirical results looks promising, however the performance of our method is limited by the quality of the trained GAN. On the other hand, improvements in generative models naturally transfers to our approach. State-of-the-art GANs can generate images with simple structures, *e.g.* faces or cars, but cannot handle general natural images.

5. CONCLUSION

In this paper, we proposed a unified framework for image restoration based on MAP and generative adversarial nets. We evaluated our approach on five different image restoration tasks. Experiments demonstrate that our approach generates more natural and realistic reconstructions; we hope to explore the theoretical aspect of our method in future works.

6. REFERENCES

[1] Manya V Afonso, José M Bioucas-Dias, and Mário AT Figueiredo, "An augmented lagrangian approach to the constrained optimization formulation of imaging inverse problems," *IEEE TIP*, 2011.

[2] Yao Hu, Debing Zhang, Jieping Ye, Xuelong Li, and Xiaofei He, "Fast and accurate matrix completion via truncated nuclear norm regularization," *IEEE PAMI*, 2013.

[3] Michael Elad and Michal Aharon, "Image denoising via learned dictionaries and sparse representation," in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*. IEEE, 2006, vol. 1, pp. 895–900.

[4] Jianchao Yang, John Wright, Thomas S Huang, and Yi Ma, "Image super-resolution via sparse representation," *IEEE transactions on image processing*, vol. 19, no. 11, pp. 2861–2873, 2010.

[5] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative adversarial nets," in *NIPS*, 2014.

[6] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang, "Deep learning face attributes in the wild," in *ICCV*, 2015.

[7] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang, "Deep laplacian pyramid networks for fast and accurate super-resolution," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[8] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee, "Accurate image super-resolution using very deep convolutional networks," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR Oral)*, June 2016.

[9] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang, "Image super-resolution using deep convolutional networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 2, pp. 295–307, Feb. 2016.

[10] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe Shi, "Photo-realistic single image super-resolution using a generative adversarial network," *arXiv preprint arXiv:1609.04802*, 2016.

[11] Deepak Pathak, Philipp Krähenbühl, Jeff Donahue, Trevor Darrell, and Alexei Efros, "Context encoders: Feature learning by inpainting," 2016.

[12] Junyuan Xie, Linli Xu, and Enhong Chen, "Image denoising and inpainting with deep neural networks," in *NIPS*, 2012.

[13] Jeremias Sulam and Michael Elad, "Large inpainting of face images with trainlets," *IEEE signal processing letters*, vol. 23, no. 12, pp. 1839–1843, 2016.

[14] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang, "Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising," *IEEE Transactions on Image Processing*, 2017.

[15] Raymond A. Yeh*, Chen Chen*, Teck Yian Lim, Schwing Alexander G., 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*, 2017, * equal contribution.

[16] Diederik Kingma and Max Welling, "Auto-encoding variational bayes," in *ICLR*, 2014.

[17] Patrick Pérez, Michel Gangnet, and Andrew Blake, "Poisson image editing," in *ACM TOG*, 2003.

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

[19] Diederik Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," in *ICLR*, 2015.

[20] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE TIP*, 2004.