Matching Plug-and-Play Algorithms to the Denoiser

Saurav K. Shastri, Rizwan Ahmad, Christopher A. Metzler, Philip Schniter

The Ohio State University · University of Maryland

NeurIPS 2021 Deep Inverse Workshop
(Supported by NSF 1955587, NIH 135489, NIH 029957)

Plug-and-Play (PnP) Image Recovery

- Goal: Recover N-pixel image $\boldsymbol{x}_0$ from $M \ll N$ noisy linear measurements

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}_0 + \boldsymbol{w}, \text{ with } \begin{cases} \boldsymbol{x}_0 : \text{true image} \\ \boldsymbol{A} : \text{linear measurement operator} \\ \boldsymbol{w} : \text{AWGN with precision } \gamma_w. \end{cases}$$

- Although deep nets can be trained to predict $\boldsymbol{x}_0$ from $\boldsymbol{y}$, they
  - require a huge number of $(\boldsymbol{x}_0, \boldsymbol{y})$ pairs for training
  - may not generalize well to a different $\boldsymbol{A}$ operators
- Plug-and-play (PnP) algorithms iteratively call a deep-net image denoiser, which can be trained ...
  - from very few images, using patches
  - independently of $\boldsymbol{A}$, facilitating generalization to any $\boldsymbol{A}$
- Challenge: In PnP, the denoiser input-error statistics are iteration-dependent and difficult to characterize. For example, they are generally non-white and non-Gaussian
- Thus, it's not clear how to train the denoiser for optimal performance in PnP!
  - Typically the denoiser is trained with AWGN
  - Gilton et al. recently proposed to train the denoiser at the PnP equilibrium point, but it's $\boldsymbol{A}$-dependent and thus may not generalize

Approximate Message Passing (AMP) Algorithms

- AMP is a family of PnP algorithms that have remarkable properties for large random $\boldsymbol{A}$:
  - The denoiser input-error is white and Gaussian with predictable variance
  - When used with an MMSE denoiser, AMP algs converge to the MMSE estimate of $\boldsymbol{x}_0$ from $\boldsymbol{y}$
- Challenge: In most image recovery problems, $\boldsymbol{A}$ does not satisfy AMP's randomness assumptions

AMP for Fourier-Structured Matrix $\boldsymbol{A} = \boldsymbol{M}\boldsymbol{F}$

- Idea: Recover the wavelet coefficients $\boldsymbol{c}_0$, not pixels $\boldsymbol{x}_0$
  - Why? The resulting model becomes $\boldsymbol{y} = \boldsymbol{B}\boldsymbol{c}_0 + \boldsymbol{w}$, where the masked Fourier-wavelet $\boldsymbol{B} = \boldsymbol{M}\boldsymbol{F}\boldsymbol{\Psi}^{\mathsf{T}}$ is approximately block-diagonal with sufficiently randomizing blocks
- With appropriate algorithm design, the denoiser input-error will be white and Gaussian *in each wavelet subband*
- Prior work includes Whitened VAMP [PS et al. '17], Variable-Density (VD)-AMP [Millard et al. '20], based on wavelet thresholding, & Denoising-VD-AMP [Metzler et al. '21]
- Note: These algorithms provide well-characterized errors, but a non-standard denoiser is required to exploit them!

Proposed Algorithm: Denoising GEC (D-GEC)

Our approach builds on the Generalized Expectation Consistent (GEC) algorithm from Fletcher et al. '16:

require: $\boldsymbol{f}_1(\cdot),\ \boldsymbol{f}_2(\cdot)$, and $\text{gdiag}(\cdot)$
initialize: $\boldsymbol{r}_1, \boldsymbol{\gamma}_1$
for $t = 0, 1, 2, \dots$

$\widehat{\boldsymbol{x}}_1 \leftarrow \boldsymbol{f}_1(\boldsymbol{r}_1, \boldsymbol{\gamma}_1)$ — linear estimation
$\boldsymbol{\eta}_1 \leftarrow \text{Diag}(\text{gdiag}(\nabla \boldsymbol{f}_1(\boldsymbol{r}_1, \boldsymbol{\gamma}_1)))^{-1}\boldsymbol{\gamma}_1$
$\boldsymbol{\gamma}_2 \leftarrow \boldsymbol{\eta}_1 - \boldsymbol{\gamma}_1$
$\boldsymbol{r}_2 \leftarrow \text{Diag}(\boldsymbol{\gamma}_2)^{-1}(\text{Diag}(\boldsymbol{\eta}_1)\widehat{\boldsymbol{x}}_1 - \text{Diag}(\boldsymbol{\gamma}_1)\boldsymbol{r}_1)$ — Onsager

$\widehat{\boldsymbol{x}}_2 \leftarrow \boldsymbol{f}_2(\boldsymbol{r}_2, \boldsymbol{\gamma}_2)$ — denoising
$\boldsymbol{\eta}_2 \leftarrow \text{Diag}(\text{gdiag}(\nabla \boldsymbol{f}_2(\boldsymbol{r}_2, \boldsymbol{\gamma}_2)))^{-1}\boldsymbol{\gamma}_2$
$\boldsymbol{\gamma}_1 \leftarrow \boldsymbol{\eta}_2 - \boldsymbol{\gamma}_2$
$\boldsymbol{r}_1 \leftarrow \text{Diag}(\boldsymbol{\gamma}_1)^{-1}(\text{Diag}(\boldsymbol{\eta}_2)\widehat{\boldsymbol{x}}_2 - \text{Diag}(\boldsymbol{\gamma}_2)\boldsymbol{r}_2)$ — Onsager

- GEC is essentially Peaceman-Rachford ADMM with adaptive vector-valued stepsizes $\boldsymbol{\gamma}_1$ and $\boldsymbol{\gamma}_2$
- The GEC linear estimation stage is preconditioned LS:

$$\boldsymbol{f}_1(\boldsymbol{r}, \boldsymbol{\gamma}) = \big(\gamma_w \boldsymbol{B}^{\mathsf{H}}\boldsymbol{B} + \text{Diag}(\boldsymbol{\gamma})\big)^{-1}\big(\gamma_w \boldsymbol{B}^{\mathsf{H}}\boldsymbol{y} + \text{Diag}(\boldsymbol{\gamma})\boldsymbol{r}\big)$$

which can be implemented using the conjugate gradient method

- $\nabla \boldsymbol{f}_i$ denotes the Jacobian, and $\text{gdiag}(\cdot)$ averages its diagonal across different wavelet subbands. D-GEC approximates the Jacobian using a Monte-Carlo approach [Ramani et al. '08]

Proposed Denoiser: corr+corr

- In the wavelet domain, the denoiser input-error is white and Gaussian in each subband, but with subband-dependent inverse-variances $\boldsymbol{\gamma}$ that change with the iterations
  - Thus, in the pixel-domain, the error is correlated Gaussian with known covariance matrix $\boldsymbol{\Psi}\,\text{Diag}(\boldsymbol{\gamma})^{-1}\boldsymbol{\Psi}^{\mathsf{T}}$
  - How should we inform the denoiser about $(\boldsymbol{\Psi}, \boldsymbol{\gamma})$?
- We propose to add an extra input channel to an arbitrary denoiser (e.g., DnCNN) and feed it with an independent realization of $\mathcal{N}(\boldsymbol{0}, \boldsymbol{\Psi}\,\text{Diag}(\boldsymbol{\gamma})^{-1}\boldsymbol{\Psi}^{\mathsf{T}})$
  - The denoiser learns to extract the statistics $(\boldsymbol{\Psi}, \boldsymbol{\gamma})$ from $\boldsymbol{e}$ and use them productively for denoising
  - We call it "corr+corr"
- Example PSNRs for depth-1 2D wavelet transform:

$\sqrt{\boldsymbol{\gamma}^{-1}}$	white DnCNN	corr+corr DnCNN	genie DnCNN
[48,47,6,19]	25.54	32.32	32.79
[10,40,23,14]	33.08	35.84	36.47
[13,7,8,10]	36.93	37.53	37.90
[10,10,10,10]	38.03	37.92	38.21
uniform [0-50,0-50,0-50,0-50]	32.18	35.34	—

White trained unif [0-50] and & corr+corr unif [0-50,0-50,0-50,0-50]

MRI Image Recovery Experiments

- We consider both single coil measurements $\boldsymbol{y} = \boldsymbol{M}\boldsymbol{F}\boldsymbol{x}_0 + \boldsymbol{w}$ as well as multi-coil measurements

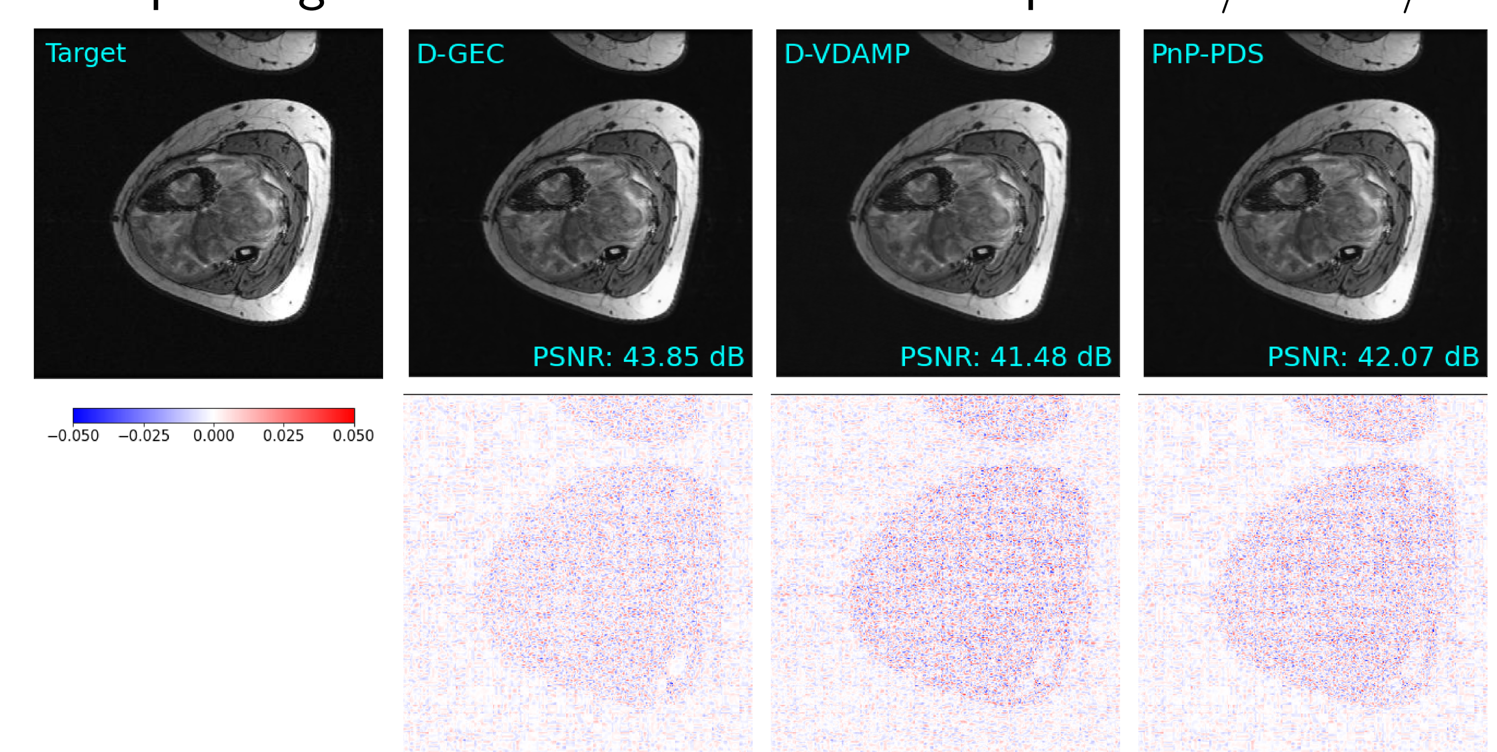
$$\boldsymbol{y} = [\boldsymbol{A}_1^{\mathsf{T}}, \dots, \boldsymbol{A}_C^{\mathsf{T}}]^{\mathsf{T}}\boldsymbol{x} + \boldsymbol{w} \quad \text{with} \quad \boldsymbol{A}_c = \boldsymbol{M}\boldsymbol{F}\,\text{Diag}(\boldsymbol{s}_c)$$

where $\{\boldsymbol{s}_c\}$ are Biot-Savart-law coil-sensitivity maps

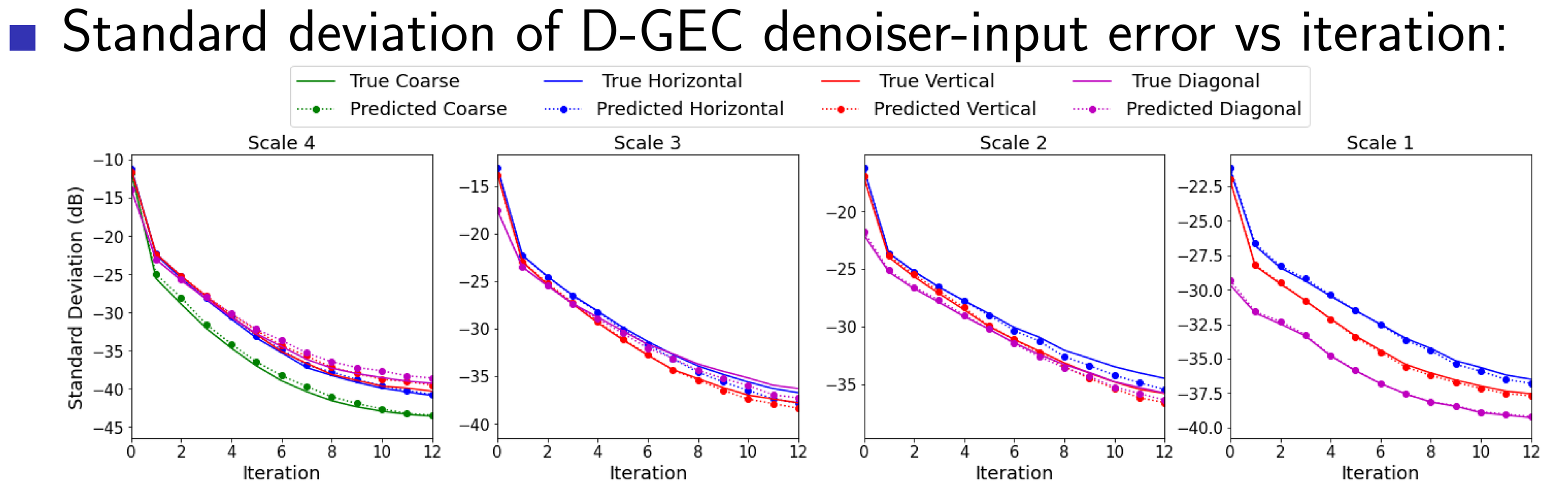
- Experimental setup:
  - $\boldsymbol{M}$ is a variable density mask
  - $\boldsymbol{w}$ is AWGN giving pre-mask SNR = 40 dB
  - $\boldsymbol{\Psi}$ is 2D Haar wavelet transform with $D = 4$ levels $\Rightarrow$ 13 subbands
  - PnP-PDS uses bias-free white-noise DnCNN and careful tuning
  - D-VDAMP uses the modified DnCNN denoiser from that paper
  - D-GEC uses bias-free corr+corr DnCNN
  - training data: 62 000 48x48 patches from 70 training images of the Stanford 2D FSE dataset
- Avg performance on 10 Stanford 2D FSE 352×352 test images:

	$C=1$ coil				$C=4$ coils			
	$M/N=1/4$		$M/N=1/8$		$M/N=1/4$		$M/N=1/8$	
method	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
PnP-PDS	45.97	0.978	41.28	0.957	47.98	0.992	43.81	0.977
D-VDAMP	44.61	0.974	38.43	0.901	n/a	n/a	n/a	n/a
D-GEC	47.64	0.982	42.42	0.959	50.80	0.997	46.67	0.991

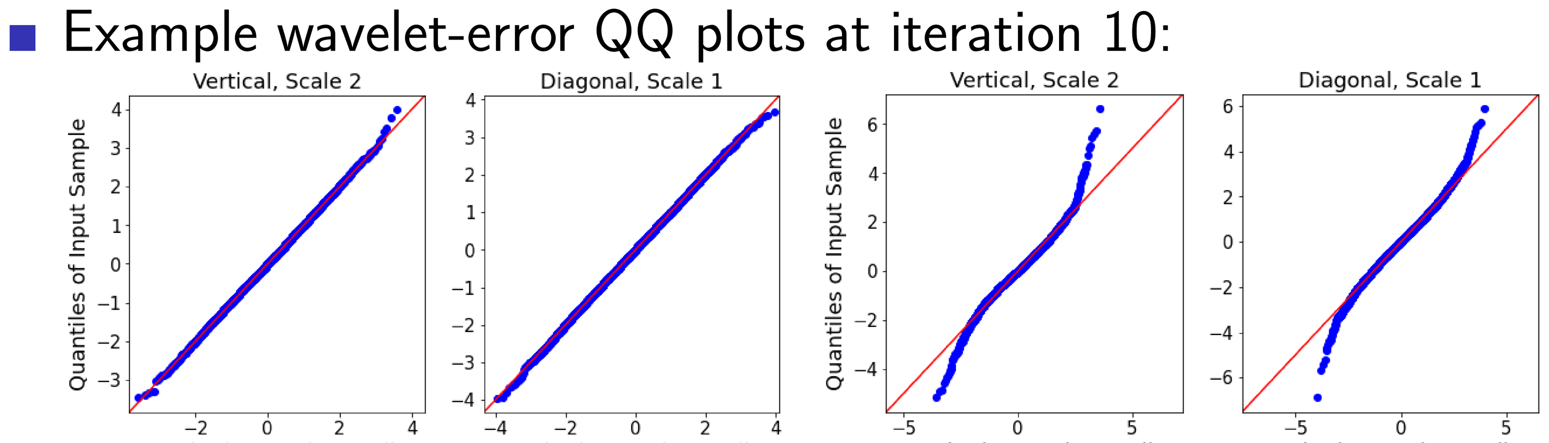
- Example single-coil recoveries and error maps at $M/N = 1/4$:



- Standard deviation of D-GEC denoiser-input error vs iteration:



- Example wavelet-error QQ plots at iteration 10:



D-GEC PnP-PDS