2020 Asilomar Conference on Signals, Systems, and Computers

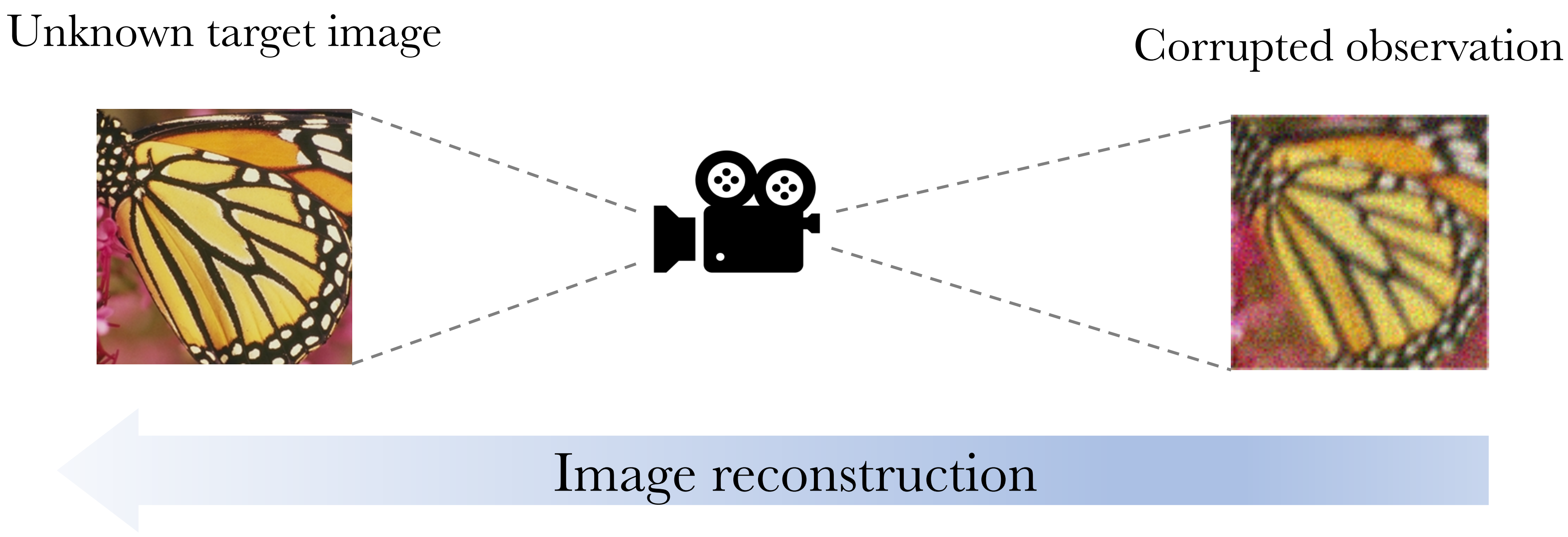
Boosting the Performance of Plug-and-Play via Denoiser Scaling

Xiaojian Xu[1], Jiaming Liu[2], Yu Sun[1], Brendt Wohlberg[3], Ulugbek S. Kamilov[1,2]

1. Department of Computer Science & Engineering, Washington University in St. Louis
2. Department of Electrical & Systems Engineering, Washington University in St. Louis.
3. Theoretical Division, Los Alamos National Laboratory

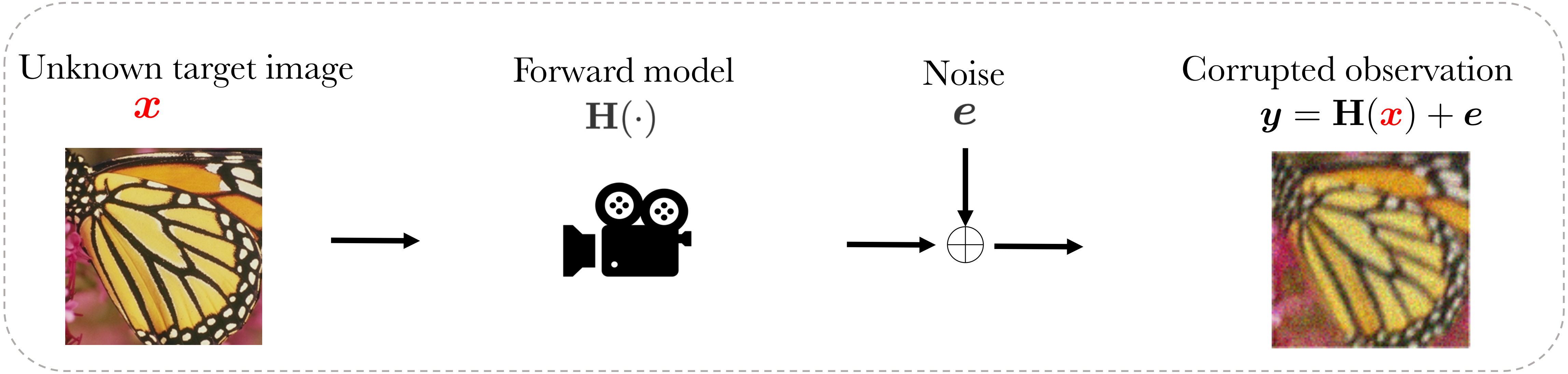
Image acquisition

- The acquisition of high-quality images is important but hard.



Imaging as an inverse problem

Acquisition procedure: generate $\boldsymbol{y}$ from $\boldsymbol{x}$



Inverse problem: recover $\boldsymbol{x}$ from $\boldsymbol{y}$

Imaging as a regularized optimization task

- Formulating it as a regularized optimization task

$$\widehat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \{ \underbrace{g(\boldsymbol{x})}_{\text{data-fidelity}} + \underbrace{h(\boldsymbol{x})}_{\text{prior/regularizer}} \}$$

Example: Fast iterative shrinkage/thresholding algorithm (FISTA) [Nesterov'13]& Alternating direction method of multipliers (ADMM) [Boyd'10]

FISTA

$$\boldsymbol{z}^k \leftarrow \boldsymbol{s}^{k-1} - \gamma \nabla g(\boldsymbol{s}^{k-1})$$
$$\boldsymbol{x}^k \leftarrow \mathsf{prox}_{\gamma h}(\boldsymbol{z}^k)$$
$$\boldsymbol{s}^k \leftarrow \boldsymbol{x}^k + ((q_{k-1} - 1)/q_k)(\boldsymbol{x}^k - \boldsymbol{x}^{k-1})$$

ADMM

$$\boldsymbol{z}^k \leftarrow \mathsf{prox}_{\gamma g}(\boldsymbol{x}^{k-1} - \boldsymbol{s}^{k-1})$$
$$\boldsymbol{x}^k \leftarrow \mathsf{prox}_{\gamma h}(\boldsymbol{z}^k + \boldsymbol{s}^{k-1})$$
$$\boldsymbol{s}^k \leftarrow \boldsymbol{s}^{k-1} + (\boldsymbol{z}^k - \boldsymbol{x}^k)$$

Proximal algorithms

model prior

$\widehat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \{g(\boldsymbol{x}) + h(\boldsymbol{x})\}$

- Let's take a closer look at these two proximal algorithms

FISTA		ADMM
$\boldsymbol{z}^k \leftarrow \boldsymbol{s}^{k-1} - \gamma \nabla g(\boldsymbol{s}^{k-1})$	increase data consistency	$\boldsymbol{z}^k \leftarrow \mathsf{prox}_{\gamma g}(\boldsymbol{x}^{k-1} - \boldsymbol{s}^{k-1})$
$\boldsymbol{x}^k \leftarrow \mathsf{prox}_{\gamma h}(\boldsymbol{z}^k)$	reduce noise	$\boldsymbol{x}^k \leftarrow \mathsf{prox}_{\gamma h}(\boldsymbol{z}^k + \boldsymbol{s}^{k-1})$
$\boldsymbol{s}^k \leftarrow \boldsymbol{x}^k + ((q_{k-1} - 1)/q_k)(\boldsymbol{x}^k - \boldsymbol{x}^{k-1})$		$\boldsymbol{s}^k \leftarrow \boldsymbol{s}^{k-1} + (\boldsymbol{z}^k - \boldsymbol{x}^k)$

Proximal algorithms

Washington University in St. Louis

model prior
$$\widehat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \{g(\boldsymbol{x}) + h(\boldsymbol{x})\}$$

- Let's take a closer look at these two proximal algorithms

FISTA		ADMM
$\boldsymbol{z}^k \leftarrow \boldsymbol{s}^{k-1} - \gamma \nabla g(\boldsymbol{s}^{k-1})$	increase data consistency	$\boldsymbol{z}^k \leftarrow \mathsf{prox}_{\gamma g}(\boldsymbol{x}^{k-1} - \boldsymbol{s}^{k-1})$
$\boldsymbol{x}^k \leftarrow \mathsf{prox}_{\gamma h}(\boldsymbol{z}^k)$	reduce noise	$\boldsymbol{x}^k \leftarrow \mathsf{prox}_{\gamma h}(\boldsymbol{z}^k + \boldsymbol{s}^{k-1})$
$\boldsymbol{s}^k \leftarrow \boldsymbol{x}^k + ((q_{k-1} - 1)/q_k)(\boldsymbol{x}^k - \boldsymbol{x}^{k-1})$		$\boldsymbol{s}^k \leftarrow \boldsymbol{s}^{k-1} + (\boldsymbol{z}^k - \boldsymbol{x}^k)$

Plug and Play Prior (PnP) [Venkat'13]:
simply replace the proximal map with other denoisers D_σ!

$$\mathsf{prox}_{\gamma h} \Rightarrow \mathsf{D}_\sigma$$

where $\sigma \geq 0$ refers to denoising strength.

PnP: Incorporating a denoiser in the optimization

Washington University in St. Louis

- Plug-and-Play (PnP) embraces off-the-shelf image denoisers

PnP-FISTA

$$\boldsymbol{z}^k = \boldsymbol{s}^{k-1} - \gamma \nabla g(\boldsymbol{s}^{k-1})$$

$$\boldsymbol{x}^k = \mathsf{D}_\sigma(\boldsymbol{z}^k)$$

$$\boldsymbol{s}^k = \boldsymbol{x}^k + ((q_{k-1} - 1)/q_k)(\boldsymbol{x}^k - \boldsymbol{x}^{k-1})$$

any off-the-shelf image denoiser

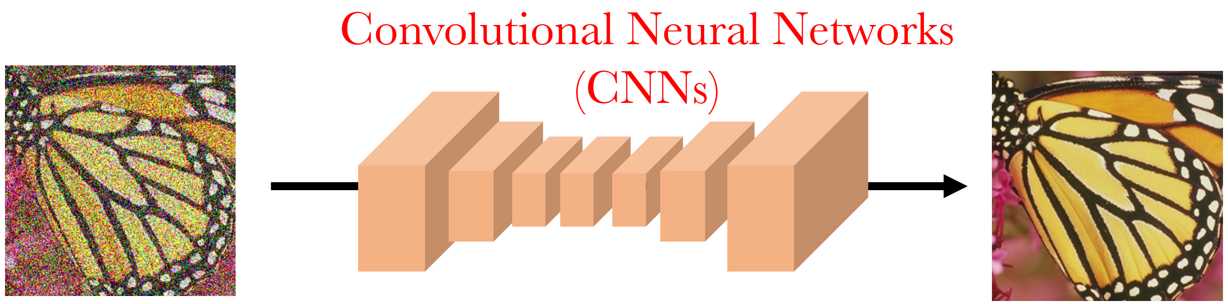
PnP-ADMM

$$\boldsymbol{z}^k = \mathsf{prox}_{\gamma g}(\boldsymbol{x}^{k-1} - \boldsymbol{s}^{k-1})$$

$$\boldsymbol{x}^k = \mathsf{D}_\sigma(\boldsymbol{z}^k + \boldsymbol{s}^{k-1})$$

$$\boldsymbol{s}^k = \boldsymbol{s}^{k-1} + (\boldsymbol{z}^k - \boldsymbol{x}^k)$$

Example: D_σ could be a neural network



PnP: Incorporating a denoiser in the optimization

Washington University in St. Louis

- Plug-and-Play (PnP) embraces off-the-shelf image denoisers

PnP-FISTA

$$\boldsymbol{z}^k = \boldsymbol{s}^{k-1} - \gamma \nabla g(\boldsymbol{s}^{k-1})$$
$$\boldsymbol{x}^k = \mathsf{D}_\sigma(\boldsymbol{z}^k)$$
$$\boldsymbol{s}^k = \boldsymbol{x}^k + ((q_{k-1} - 1)/q_k)(\boldsymbol{x}^k - \boldsymbol{x}^{k-1})$$

$\sigma = ?$

PnP-ADMM

$$\boldsymbol{z}^k = \mathsf{prox}_{\gamma g}(\boldsymbol{x}^{k-1} - \boldsymbol{s}^{k-1})$$
$$\boldsymbol{x}^k = \mathsf{D}_\sigma(\boldsymbol{z}^k + \boldsymbol{s}^{k-1})$$
$$\boldsymbol{s}^k = \boldsymbol{s}^{k-1} + (\boldsymbol{z}^k - \boldsymbol{x}^k)$$

Many CNNs denoisers do not have a tunable parameter for the noise standard deviation!

PnP: Incorporating a denoiser in the optimization

Washington University in St. Louis

- Plug-and-Play (PnP) embraces off-the-shelf image denoisers

PnP-FISTA

$$\boldsymbol{z}^k = \boldsymbol{s}^{k-1} - \gamma \nabla g(\boldsymbol{s}^{k-1})$$

$$\boldsymbol{x}^k = \mathsf{D}_\sigma(\boldsymbol{z}^k)$$

$$\boldsymbol{s}^k = \boldsymbol{x}^k + ((q_{k-1} - 1)/q_k)(\boldsymbol{x}^k - \boldsymbol{x}^{k-1})$$

$\sigma = ?$

PnP-ADMM

$$\boldsymbol{z}^k = \mathsf{prox}_{\gamma g}(\boldsymbol{x}^{k-1} - \boldsymbol{s}^{k-1})$$

$$\boldsymbol{x}^k = \mathsf{D}_\sigma(\boldsymbol{z}^k + \boldsymbol{s}^{k-1})$$

$$\boldsymbol{s}^k = \boldsymbol{s}^{k-1} + (\boldsymbol{z}^k - \boldsymbol{x}^k)$$

- Previous solution : denoiser selection
  - ✪ Idea: Training multiple CNN instances and select the one that works best.
  - ✪ Issues: Requires training multiple CNN instances and leads to suboptimal performance.

Proposed denoiser scaling technique

- Plug-and-Play (PnP) embraces off-the-shelf image denoisers

PnP-FISTA

$$\boldsymbol{z}^k = \boldsymbol{s}^{k-1} - \gamma \nabla g(\boldsymbol{s}^{k-1})$$

$$\boldsymbol{x}^k = \mathsf{D}_\sigma(\boldsymbol{z}^k)$$

$$\boldsymbol{s}^k = \boldsymbol{x}^k + ((q_{k-1} - 1)/q_k)(\boldsymbol{x}^k - \boldsymbol{x}^{k-1})$$

$\sigma = ?$

PnP-ADMM

$$\boldsymbol{z}^k = \mathsf{prox}_{\gamma g}(\boldsymbol{x}^{k-1} - \boldsymbol{s}^{k-1})$$

$$\boldsymbol{x}^k = \mathsf{D}_\sigma(\boldsymbol{z}^k + \boldsymbol{s}^{k-1})$$

$$\boldsymbol{s}^k = \boldsymbol{s}^{k-1} + (\boldsymbol{z}^k - \boldsymbol{x}^k)$$

- Our proposal : denoiser scaling
  - Introduce a tunable parameter μ to adjust the denoising strength of a pre-trained CNN.

Without scaling: $\widehat{\boldsymbol{z}} = \mathsf{D}_\sigma(\boldsymbol{z})$

Denoiser scaling: $\widehat{\boldsymbol{z}} = \mu^{-1}\mathsf{D}_\sigma(\mu \boldsymbol{z}), \quad \mu > 0$

Performance of denoiser scaling

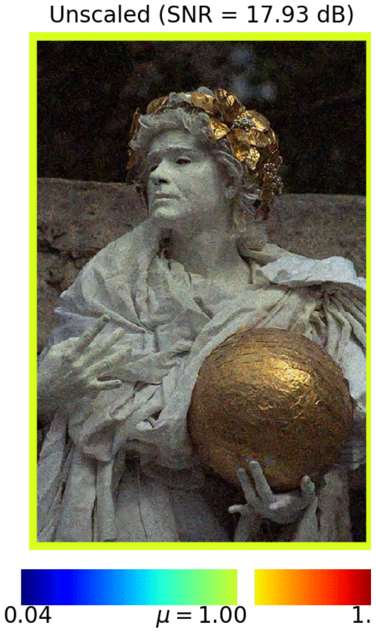
Washington University in St. Louis

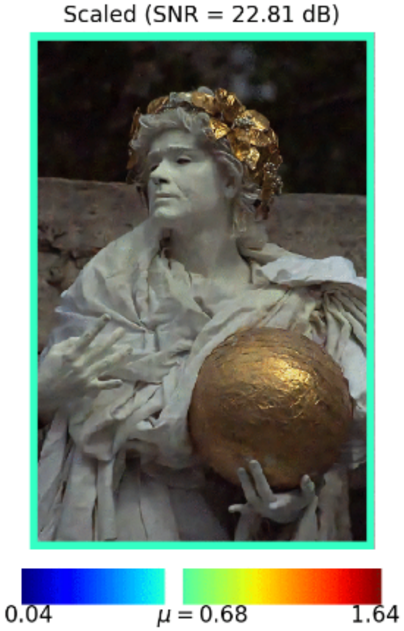
- CNN trained on noise level $\sigma = 20$, applied on noise level $\sigma = 30$, difference $\Delta_\sigma = 10$.

Noisy image:
z

Without scaling:
$\hat{z} = \mathsf{D}_\sigma(z)$

With scaling:
$\hat{z} = \mu^{-1}\mathsf{D}_\sigma(\mu z)$

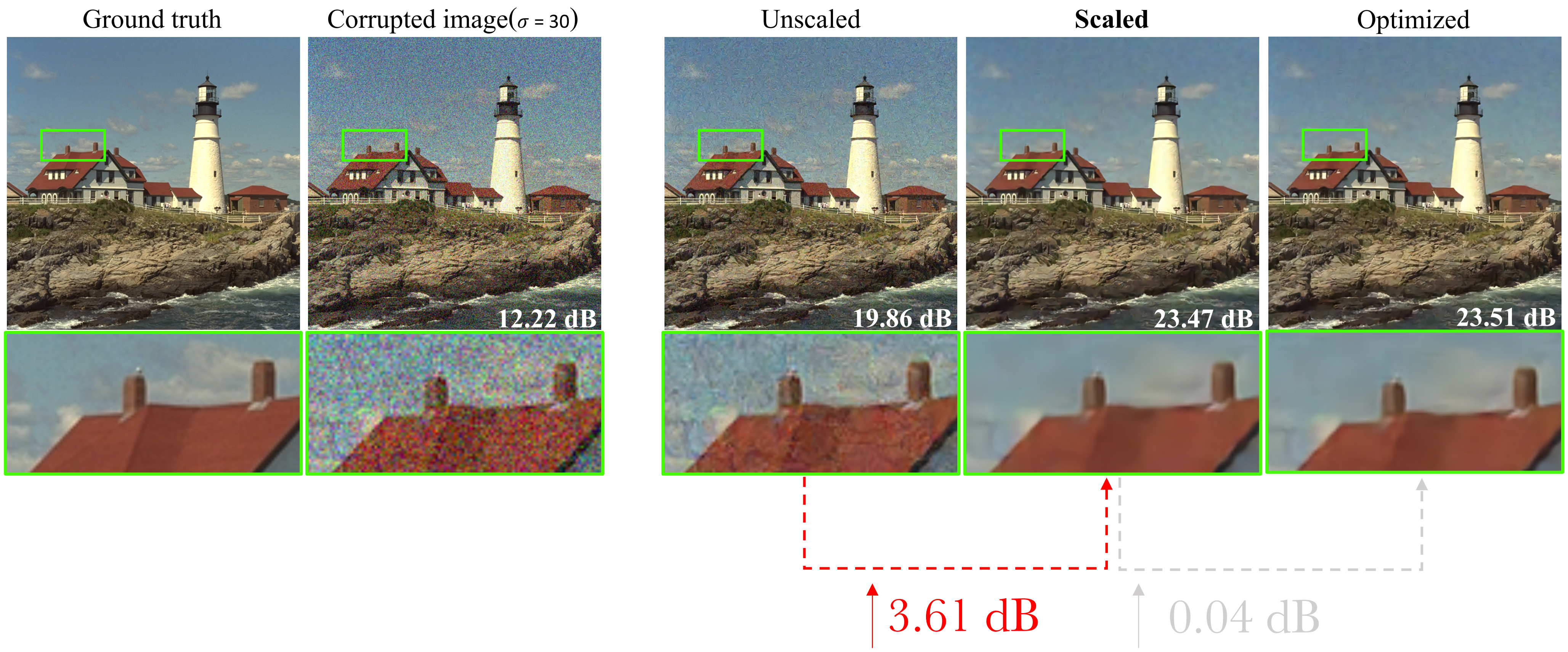




Performance of denoiser scaling

Washington University in St. Louis

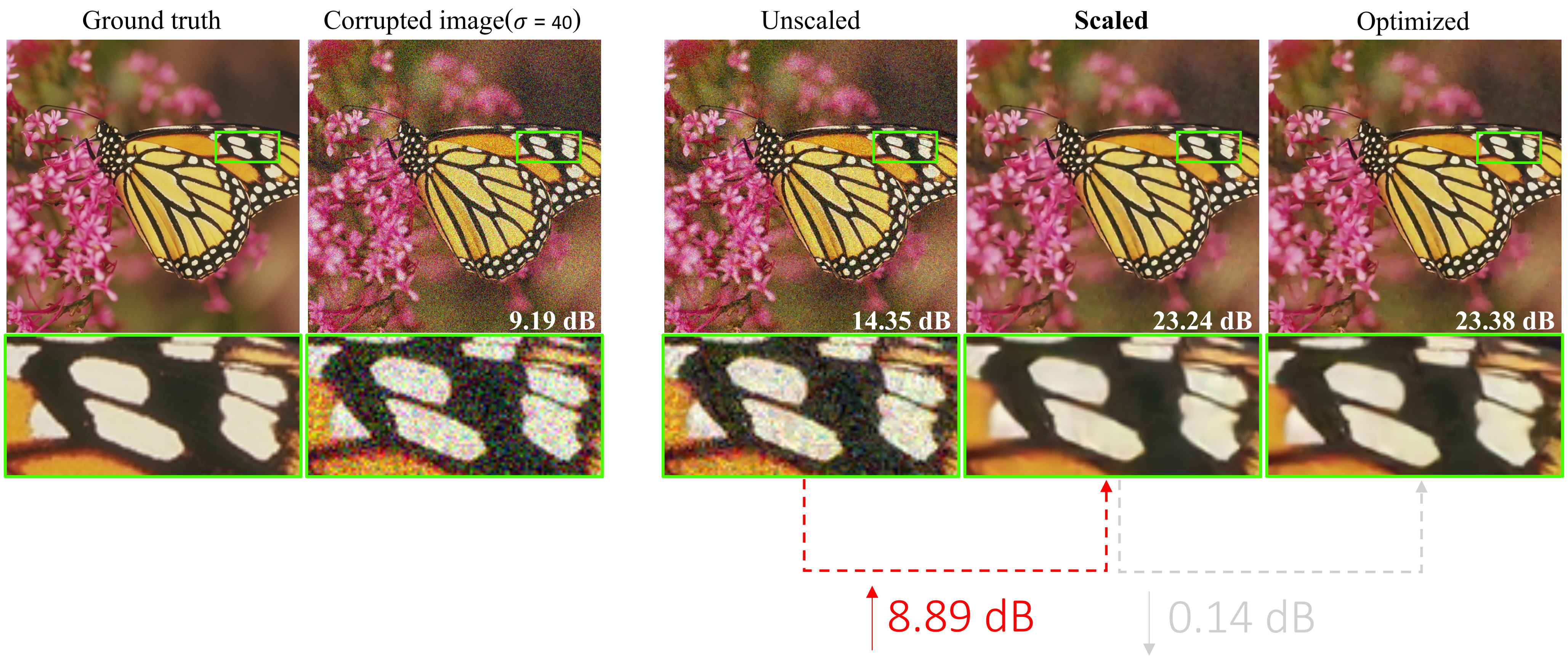
- CNN trained on noise level $\sigma = 20$, applied on noise level $\sigma = 30$, difference $\Delta_\sigma = 10$.



*Number written to image is signal-to-noise ratio (SNR)

Performance of denoiser scaling

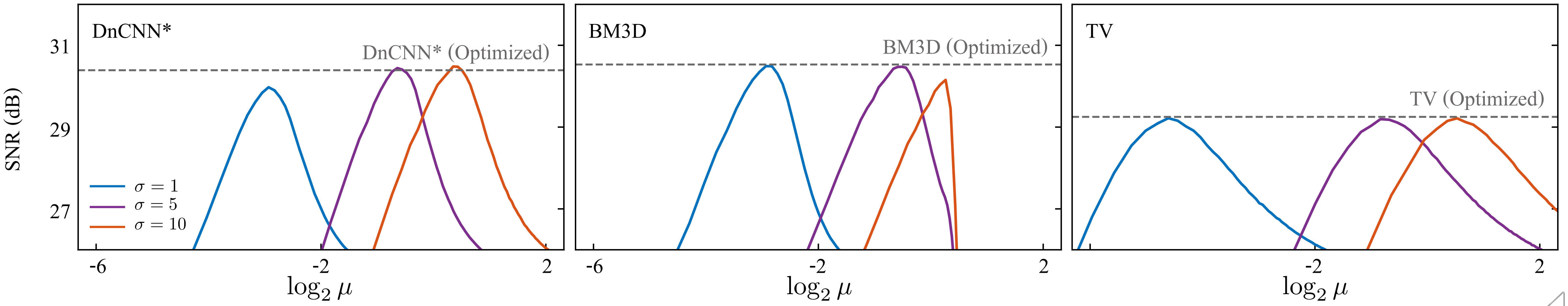
- CNN trained on noise level $\sigma = 20$, applied on noise level $\sigma = 40$, difference $\Delta_\sigma = 20$.



Theoretical analysis of denoiser scaling

Washington University in St. Louis

- Denoiser scaling is proved to have the following properties:
  - ✪ When the denoiser is a minimum mean-squared error (MMSE) denoiser, adjusting μ is equivalent to scale the variance of AWGN by μ^{-2} in the MMSE estimation.
  - ✪ When denoiser is a proximal map $\mathrm{prox}_{\lambda h}(\boldsymbol{z}) := \arg\min_{\boldsymbol{x}} \{\frac{1}{2}\|\boldsymbol{x}-\boldsymbol{z}\|_2^2 + \lambda h(\boldsymbol{x})\}$, where regularizer $h(\cdot)$ is 1-homogeneous with $h(\mu\,\cdot) = \mu\, h(\cdot)$, adjusting μ is equivalent to adjusting the weighting parameter of h.



*Plots are generated with noisy image at noise level σ = 7.23

PnP algorithms with denoiser scaling

- PnP algorithms with denoiser scaling

PnP-FISTA

$$\boldsymbol{z}^k = \boldsymbol{s}^{k-1} - \gamma \nabla g(\boldsymbol{s}^{k-1})$$
$$\boldsymbol{x}^k = \mathsf{D}_\sigma(\boldsymbol{z}^k)$$
$$\boldsymbol{s}^k = \boldsymbol{x}^k + ((q_{k-1} - 1)/q_k)(\boldsymbol{x}^k - \boldsymbol{x}^{k-1})$$

$\Downarrow$

Scaled PnP-FISTA

$$\boldsymbol{z}^k = \boldsymbol{s}^{k-1} - \gamma \nabla g(\boldsymbol{s}^{k-1})$$
$$\boldsymbol{x}^k = \mu^{-1}\mathsf{D}_\sigma(\mu \boldsymbol{z}^k)$$
$$\boldsymbol{s}^k = \boldsymbol{x}^k + ((q_{k-1} - 1)/q_k)(\boldsymbol{x}^k - \boldsymbol{x}^{k-1})$$

PnP-ADMM

$$\boldsymbol{z}^k = \mathsf{prox}_{\gamma g}(\boldsymbol{x}^{k-1} - \boldsymbol{s}^{k-1})$$
$$\boldsymbol{x}^k = \mathsf{D}_\sigma((\boldsymbol{z}^k + \boldsymbol{s}^{k-1}))$$
$$\boldsymbol{s}^k = \boldsymbol{s}^{k-1} + (\boldsymbol{z}^k - \boldsymbol{x}^k)$$

$\Downarrow$

Scaled PnP-ADMM

$$\boldsymbol{z}^k = \mathsf{prox}_{\gamma g}(\boldsymbol{x}^{k-1} - \boldsymbol{s}^{k-1})$$
$$\boldsymbol{x}^k = \mu^{-1}\mathsf{D}_\sigma(\mu(\boldsymbol{z}^k + \boldsymbol{s}^{k-1}))$$
$$\boldsymbol{s}^k = \boldsymbol{s}^{k-1} + (\boldsymbol{z}^k - \boldsymbol{x}^k)$$

Inverse problem examples

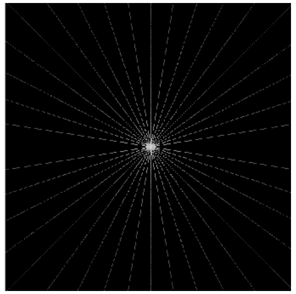
- Image Super-resolution (SR) and Magnetic resonance imaging (MRI) problem

Low-resolution image

SR inverse problem

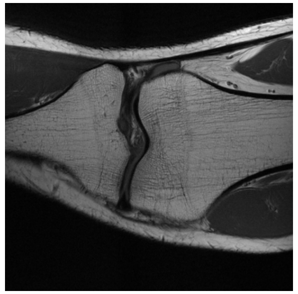
High-resolution image

Under-sampled frequencies



MRI inverse problem

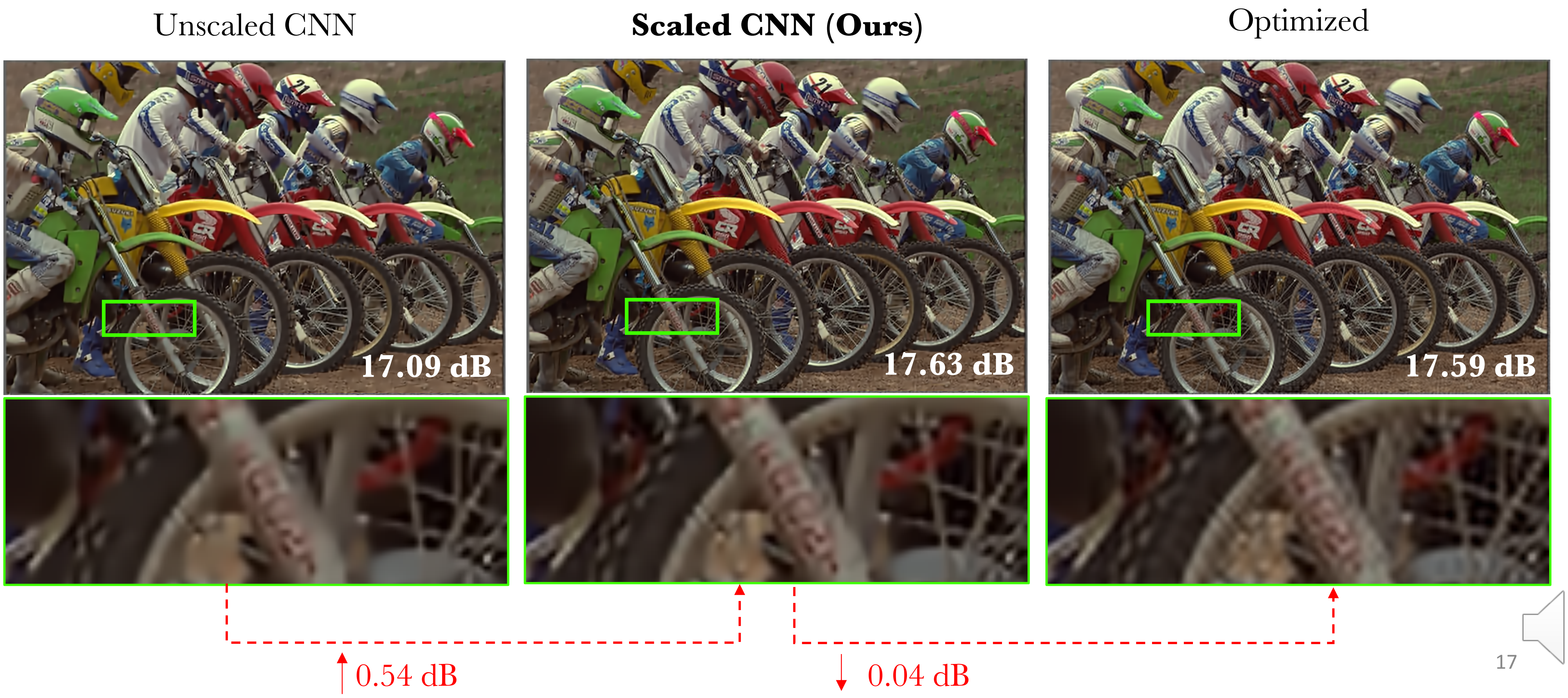
Clean image



Scaling performance in image SR problem

Washington University in St. Louis

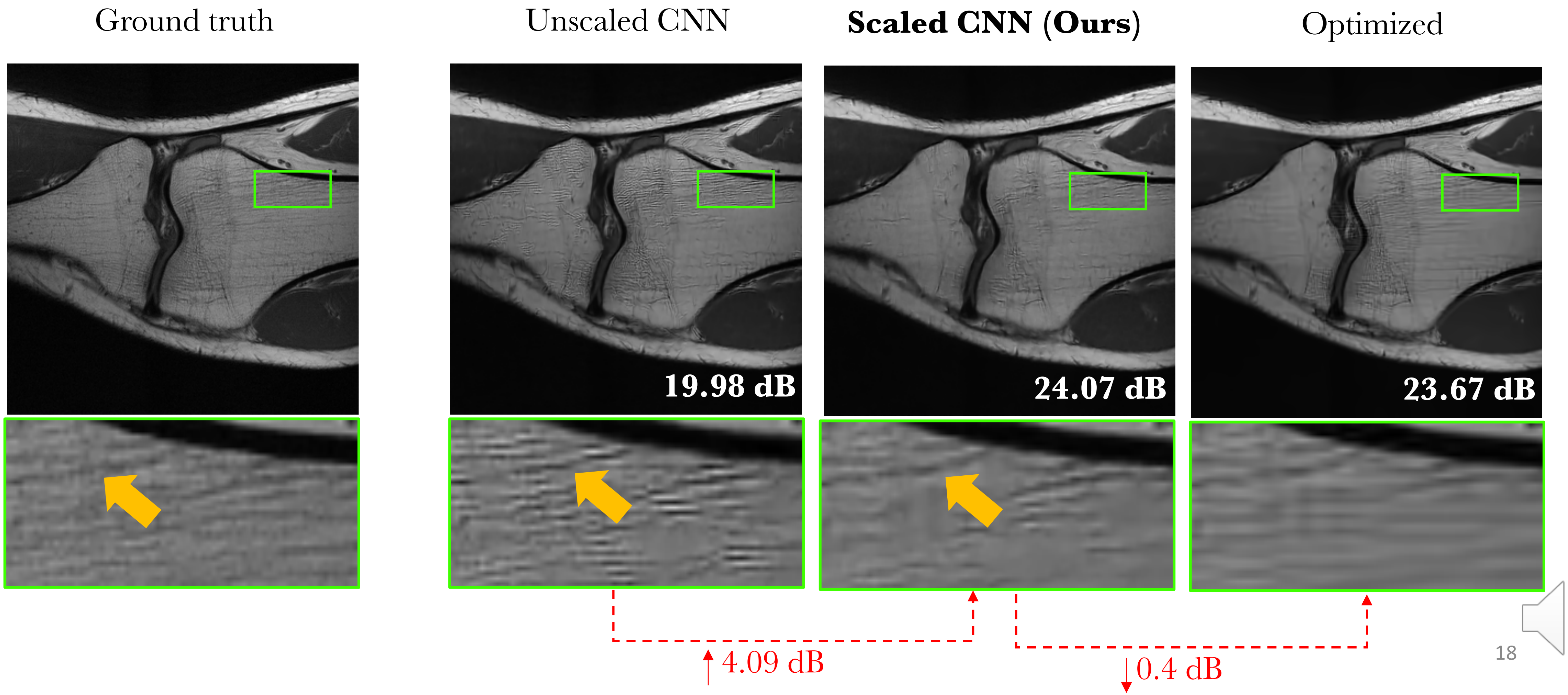
- Scaling technique can sharpen the blurry edges caused by the suboptimal denoiser.



Scaling performance in MRI problem

Washington University in St.Louis

- Scaling technique can alleviate the artifacts caused by the suboptimal denoiser.



Conclusion

Washington University in St. Louis

- Summary of our talk
  - We proposed a denoiser scaling technique that can help with the denoising strength tuning especially for CNN type of denoisers.
  - We showed that denoiser scaling can effectively boost the performance of PnP algorithms and achieve the optimal results.

Thanks!