Regularization by Denoising: Clarifications and New Interpretations

Phil Schniter and Ted Reehorst

The Ohio State University

Supported by NSF grants IIP-0968910 and CCF-1716388 and NIH grant R01HL135489

BASP Frontiers Workshop 2019

Inverse Problems in Imaging

- Consider the basic inverse problem in imaging:

  $$\text{Recover } \boldsymbol{x}^0 \text{ from measurements } \boldsymbol{y} = \text{corrupted}(\boldsymbol{A}\boldsymbol{x}^0),$$

  where $\boldsymbol{A}$ is a known linear operator.
- Corruptions include noise, quantization, loss of phase, Poisson photons, etc.
- The operator $\boldsymbol{A}$ depends on the application:
  - deblurring
  - super-resolution
  - compressive imaging
  - inpainting
  - etc

Optimization-Based Recovery and MAP Estimation

- A common approach to recovering the image $\boldsymbol{x}$ is through posing and solving an optimization problem:

  $$\widehat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \left\{ \ell(\boldsymbol{x};\boldsymbol{y}) + \lambda\rho(\boldsymbol{x}) \right\} \text{ with } \begin{cases} \ell(\boldsymbol{x};\boldsymbol{y}): & \text{loss function} \\ \rho(\boldsymbol{x}): & \text{regularization} \\ \lambda > 0: & \text{tuning parameter} \end{cases}$$

- This can be interpreted as Bayesian MAP estimation:

  $$\widehat{\boldsymbol{x}}_{\mathsf{map}} = \arg\min_{\boldsymbol{x}} \left\{ -\ln p(\boldsymbol{y}|\boldsymbol{x}) - \ln p(\boldsymbol{x}) \right\} \text{ with } \begin{cases} p(\boldsymbol{y}|\boldsymbol{x}): & \text{likelihood} \\ p(\boldsymbol{x}): & \text{prior} \end{cases}$$

- The loss function $\ell(\cdot;\boldsymbol{y})$ is usually straightforward to choose.
  But how do we choose the regularization $\rho(\cdot)$?

Plug-and-Play ADMM

- A common approach to convex optimization is ADMM: For some $\beta > 0$ and $k = 1, 2, 3, \dots$

  $$\boldsymbol{x}_k = \arg\min_{\boldsymbol{x}} \left\{ \ell(\boldsymbol{x};\boldsymbol{y}) + \tfrac{\beta}{2}\|\boldsymbol{x} - \boldsymbol{v}_{k-1} + \boldsymbol{u}_{k-1}\|^2 \right\}$$
  $$\boldsymbol{v}_k = \arg\min_{\boldsymbol{v}} \left\{ \rho(\boldsymbol{v}) + \tfrac{\beta}{2}\|\boldsymbol{v} - \boldsymbol{x}_k + \boldsymbol{u}_{k-1}\|^2 \right\} \triangleq \operatorname{prox}_{\rho/\beta}(\boldsymbol{x}_k - \boldsymbol{u}_{k-1})$$
  $$\boldsymbol{u}_k = \boldsymbol{u}_{k-1} + \boldsymbol{x}_k - \boldsymbol{v}_k$$

- The prox operation performs denoising (eg, soft-thresholding when $\rho(\boldsymbol{x}) = \|\boldsymbol{x}\|_1$).
- In 2013, Bouman et al. proposed plug-and-play (PnP) ADMM, where the prox is replaced by a sophisticated image denoiser $\boldsymbol{f}(\cdot)$, such as BM3D.

Regularization by Denoising (RED)

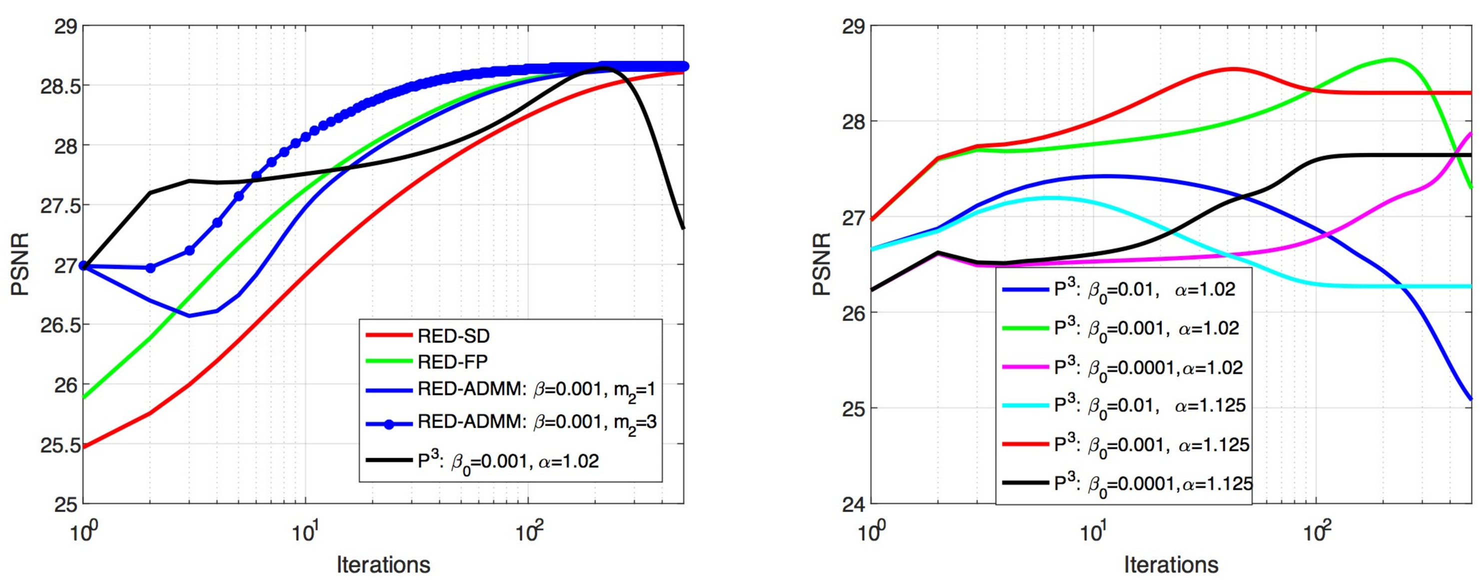
- In 2017, Romano, Elad, and Milanfar proposed a new family of PnP algorithms that find the image estimate $\widehat{\boldsymbol{x}}$ that obeys

  $$\boxed{\nabla\ell(\widehat{\boldsymbol{x}};\boldsymbol{y}) + \lambda\big(\widehat{\boldsymbol{x}} - \boldsymbol{f}(\widehat{\boldsymbol{x}})\big) = \boldsymbol{0}}$$

- They claimed these algorithms result from optimization under the regularizer

  $$\rho_{\mathsf{red}}(\boldsymbol{x}) \triangleq \frac{1}{2}\boldsymbol{x}^\top\big(\boldsymbol{x} - \boldsymbol{f}(\boldsymbol{x})\big)$$

  and thus coined the approach Regularization by Denoising (RED).
- They furthermore claimed that $\rho_{\mathsf{red}}(\cdot)$ was convex with practical image denoisers $\boldsymbol{f}(\cdot)$.
- Experiments in the RED paper suggest advantages for RED over PnP-ADMM:



Super-resolution recovery, averaged over 10 test images.

The RED algorithms are not explained by the RED regularization!

- Visualize by probing in two random directions:

  $$\boldsymbol{x}_{\alpha,\beta} = \widehat{\boldsymbol{x}} + \alpha\boldsymbol{r}_1 + \beta\boldsymbol{r}_2.$$

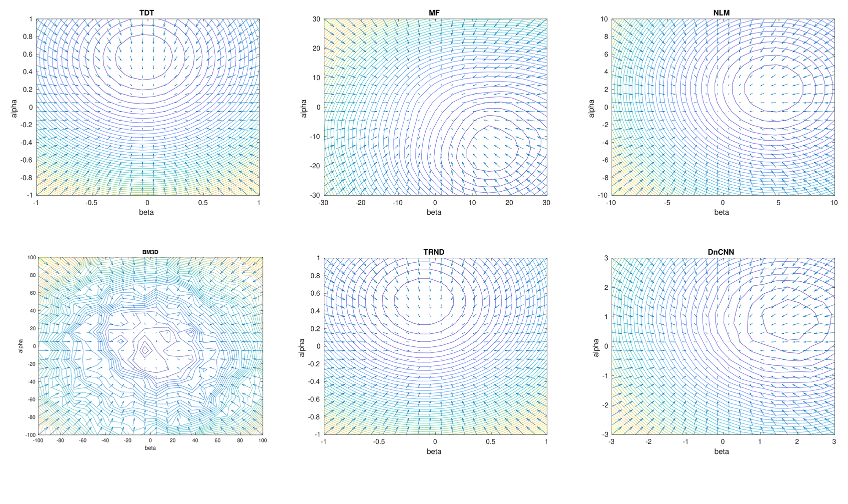
- Contours show cost:

  $$C_{\mathsf{red}}(\boldsymbol{x}_{\alpha,\beta}) \triangleq \frac{1}{2\sigma^2}\|\boldsymbol{y} - \boldsymbol{x}_{\alpha,\beta}\|^2 + \rho_{\mathsf{red}}(\boldsymbol{x}_{\alpha,\beta}).$$

- Arrows show claimed RED gradient:

  $$\frac{1}{\sigma^2}(\boldsymbol{x}_{\alpha,\beta} - \boldsymbol{y})^\top[\boldsymbol{r}_1\ \boldsymbol{r}_2] + \lambda(\boldsymbol{x}_{\alpha,\beta} - \boldsymbol{f}(\boldsymbol{x}_{\alpha,\beta}))^\top[\boldsymbol{r}_1\ \boldsymbol{r}_2].$$

- Figures show that 1) zero of gradient field is not at cost minimizer, and 2) cost may not be convex!



Clarifications on the RED Gradient

In the full paper, we established that…

- differentiability of $\boldsymbol{f}(\cdot)$ implies

  $$\nabla\rho_{\mathsf{red}}(\boldsymbol{x}) \overset{\mathsf{D}}{=} \boldsymbol{x} - \frac{1}{2}\boldsymbol{f}(\boldsymbol{x}) - \frac{1}{2}[J\boldsymbol{f}(\boldsymbol{x})]^\top\boldsymbol{x}.$$

- adding local-homogeneity (LH), i.e., $\boldsymbol{f}\big((1+\epsilon)\boldsymbol{x}\big) = (1+\epsilon)\boldsymbol{f}(\boldsymbol{x})$, gives

  $$\nabla\rho_{\mathsf{red}}(\boldsymbol{x}) \overset{\mathsf{D,LH}}{=} \boldsymbol{x} - \frac{1}{2}[J\boldsymbol{f}(\boldsymbol{x})]\boldsymbol{x} - \frac{1}{2}[J\boldsymbol{f}(\boldsymbol{x})]^\top\boldsymbol{x}.$$

- adding Jacobian symmetry (JS) finally leads to

  $$\nabla\rho_{\mathsf{red}}(\boldsymbol{x}) \overset{\mathsf{D,LH,JS}}{=} \boldsymbol{x} - \boldsymbol{f}(\boldsymbol{x}) \quad \dots\text{which yields the RED algorithms.}$$

But practical denoisers are not LH and JS! And there exists no regularizer ρ_{red} for a non-JS denoiser $\boldsymbol{f}$!

How can we explain the RED algorithms?

The RED algorithms solve $\boxed{\nabla\ell(\widehat{\boldsymbol{x}};\boldsymbol{y}) + \lambda\big(\widehat{\boldsymbol{x}} - \boldsymbol{f}(\widehat{\boldsymbol{x}})\big) = \boldsymbol{0}}$ and work well.

Can we justify this approach? Even when $\boldsymbol{f}(\cdot)$ is not locally homogeneous or Jacobian symmetric?

Yes! Using score matching, a framework first described by Hyvärinen in 2005. We explain this in 3 steps:

1. kernel density estimation,
2. Tweedie's formula,
3. score matching.

Kernel Density Estimation (KDE)

- Given training data $\{\boldsymbol{x}_t\}_{t=1}^T$, consider forming the empirical prior

  $$\widehat{p}_{\mathsf{x}}(\boldsymbol{x}) = \frac{1}{T}\sum_{t=1}^T \delta(\boldsymbol{x} - \boldsymbol{x}_t).$$

- A better match to the true p_{x} is obtained via Parzen windowing or KDE:

  $$\widetilde{p}_{\mathsf{x}}(\boldsymbol{x};\nu) = \frac{1}{T}\sum_{t=1}^T \mathcal{N}(\boldsymbol{x};\boldsymbol{x}_t,\nu\boldsymbol{I}) = \int_{\mathbb{R}^N}\mathcal{N}(\boldsymbol{r};\boldsymbol{x},\nu\boldsymbol{I})\,\widehat{p}_{\mathsf{x}}(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}. \qquad \text{"smoothed prior"}$$

- Using the smoothed prior $\widetilde{p}_{\mathsf{x}}$ for MAP image recovery, we get

  $$\widehat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \left\{ \ell(\boldsymbol{x};\boldsymbol{y}) - \ln\widetilde{p}_{\mathsf{x}}(\boldsymbol{x};\nu) \right\}.$$

Tweedie's Formula

- Assuming differentiable $\ell(\cdot;\boldsymbol{y})$, the MAP estimation problem is solved by

  $$\boldsymbol{0} = \nabla\ell(\boldsymbol{x};\boldsymbol{y}) - \nabla\ln\widetilde{p}_{\mathsf{x}}(\boldsymbol{x};\nu).$$

- Tweedie's formula (see [Robbins'56]) says that

  $$\nabla\ln\widetilde{p}_{\mathsf{x}}(\boldsymbol{x};\nu) = \tfrac{1}{\nu}\big(\boldsymbol{f}_{\mathsf{mmse},\nu}(\boldsymbol{x}) - \boldsymbol{x}\big),$$

  with $\boldsymbol{f}_{\mathsf{mmse},\nu}(\boldsymbol{r})$ the MMSE denoiser of $\boldsymbol{x} \sim \widehat{p}_{\mathsf{x}}$ from $\boldsymbol{r} = \boldsymbol{x} + \mathcal{N}(\boldsymbol{0},\nu\boldsymbol{I})$.
- Together, these results match the RED fixed-point equation

  $$\boldsymbol{0} = \nabla\ell(\boldsymbol{x};\boldsymbol{y}) + \lambda\big(\boldsymbol{x} - \boldsymbol{f}_{\mathsf{mmse},\nu}(\boldsymbol{x})\big) \text{ with } \lambda = \frac{1}{\nu}$$

  for the specific denoiser $\boldsymbol{f}_{\mathsf{mmse},\nu}$. What about generic denoisers $\boldsymbol{f}$?

Score-Matching by Denoising

- Recall $\boldsymbol{f}_{\mathsf{mmse},\nu} = \arg\min_{\boldsymbol{f}} \mathbb{E}\{\|\boldsymbol{x} - \boldsymbol{f}(\boldsymbol{r})\|^2\}$ for $\begin{cases} \boldsymbol{r} = \boldsymbol{x} + \mathcal{N}(\boldsymbol{0},\nu\boldsymbol{I}) \\ \boldsymbol{x} \sim \widehat{p}_{\mathsf{x}}. \end{cases}$
- Since $\boldsymbol{f}_{\mathsf{mmse},\nu}$ is expensive to implement, we typically use some approximation $\boldsymbol{f}_{\widehat{\boldsymbol{\theta}}}$ with

  $$\begin{aligned} \widehat{\boldsymbol{\theta}} &= \arg\min_{\boldsymbol{\theta}} \mathbb{E}\{\|\boldsymbol{x} - \boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{r})\|^2\} && \text{e.g., deep network} \\ &= \arg\min_{\boldsymbol{\theta}} \mathbb{E}\{\|\boldsymbol{x} - \boldsymbol{f}_{\mathsf{mmse},\nu}(\boldsymbol{r})\|^2\} + \mathbb{E}\left\{\left\|\boldsymbol{f}_{\mathsf{mmse},\nu}(\boldsymbol{r}) - \boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{r})\right\|^2\right\} && \text{via orthog principle} \\ &= \arg\min_{\boldsymbol{\theta}} \mathbb{E}\left\{\left\|\boldsymbol{f}_{\mathsf{mmse},\nu}(\boldsymbol{r}) - \boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{r})\right\|^2\right\} \\ &= \arg\min_{\boldsymbol{\theta}} \mathbb{E}\left\{\Big\|\underbrace{\nabla\ln\widetilde{p}_{\mathsf{x}}(\boldsymbol{r};\nu)}_{\text{"score"}} - \underbrace{\tfrac{1}{\nu}\big(\boldsymbol{r} - \boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{r})\big)}_{\text{RED log-prior}}\Big\|^2\right\} && \text{via Tweedie's formula} \end{aligned}$$

- Thus RED algorithm with general denoiser $\boldsymbol{f}_{\boldsymbol{\theta}}$ can be interpreted as "score matching."
- Key points:
  1. RED algs solve $\boldsymbol{0} = \nabla\ell(\boldsymbol{x};\boldsymbol{y}) + \lambda\big(\boldsymbol{x} - \boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x})\big)$ where $\lambda\big(\boldsymbol{x} - \boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x})\big)$ approximates the score $-\nabla\ln\widetilde{p}_{\mathsf{x}}(\boldsymbol{x};\nu)$.
  2. This SMD interpretation holds for any $\widehat{p}_{\mathsf{x}}$, any denoiser class $\boldsymbol{f}_{\boldsymbol{\theta}}$ (i.e., may be non-JS and/or non-LH), and any $\boldsymbol{\theta}$.
  3. SMD arises naturally via non-parametric estimation (i.e., KDE). Matches construction of *learned* denoisers liked TNRD and DnCNN.
- Related work:
  - In 2014, Alain and Bengio showed that learned auto-encoders are be explained by score-matching and *not* by minimization of an energy function.
  - In 2017, Bigdeli and Zwicker used Tweedie's formula to interpret autoencoding-based image priors.

Fast RED Algorithms

Until now we focused on how to explain the RED method, which solves

$$\boxed{\boldsymbol{0} = \nabla\ell(\widehat{\boldsymbol{x}};\boldsymbol{y}) + \lambda\big(\widehat{\boldsymbol{x}} - \boldsymbol{f}(\widehat{\boldsymbol{x}})\big).}$$

Now we focus on algorithms that try to *solve* this equation.

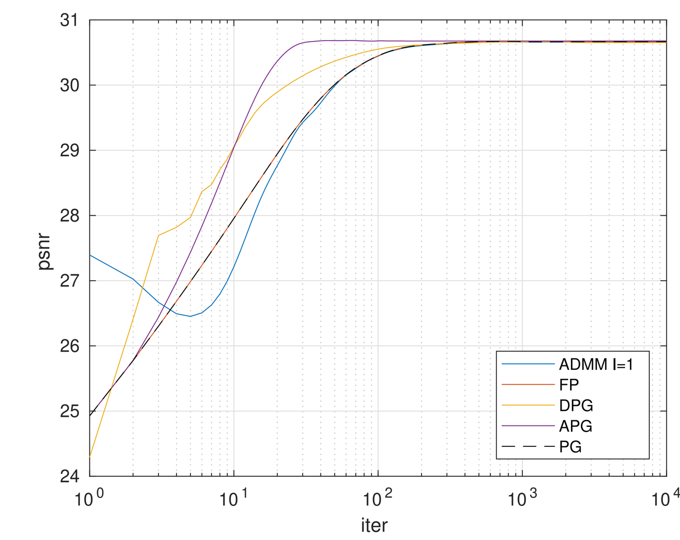
In the RED paper by Romano, Elad, and Milanfar, three algorithms were described:

1. steepest-descent
2. ADMM with I inner iters (to solve $\arg\min_{\boldsymbol{x}}\{\lambda\rho_{\mathsf{red}}(\boldsymbol{x}) + \frac{\beta}{2}\|\boldsymbol{x} - \boldsymbol{r}_t\|^2\}$)
3. a heuristic "fixed-point" method.

We proposed three new algorithms:

- PG: Proximal gradient with stepsize $L > 0$.
- DPG: "Dynamic" proximal gradient that schedules L with the iterations.
- APG: Accelerated proximal gradient, similar in spirit to FISTA.

In the de-blurring experiment on the right, APG is about $3\times$ faster than the "fixed-point" method.



Convergence to a Fixed Point

Theorem:
If $\ell(\cdot)$ is proper, convex, and continuous; $\boldsymbol{f}(\cdot)$ is non-expansive; $L > 1$; and RED-PG has at least one fixed point, then RED-PG converges to a fixed point.

Proof:
Uses α-averaged operators and the Mann iteration.

Conclusions

- RED algorithms seem to work well in practice.
- But, in practice, they are *not* minimizing any cost function.
  - Practical denoisers $\boldsymbol{f}(\cdot)$ are not LH and JS.
  - Non-JS $\boldsymbol{f}$ implies that there exists no regularizer ρ s.t. $\nabla\rho(\boldsymbol{x}) = \boldsymbol{x} - \boldsymbol{f}(\boldsymbol{x})$.
- The RED methodology can be explained as "score-matching by denoising".
- We proposed new RED algorithms with
  - faster recovery
  - guaranteed convergence to a fixed point.
- For more details (e.g., an equilibrium analysis), please see:

E.T. Reehorst and P. Schniter, "Regularization by denoising: Clarifications and new interpretations," *IEEE Trans. Computational Imaging*, vol. 5, no. 1, pp. 52-67, Mar 2019. `http://arxiv.org/abs/1806.02296`