Data-Driven Feasibility Algorithms

Howard Heaton[1], Samy Wu Fung[1], Aviv Gibali[2], Wotao Yin[1]

[1]UCLA, [2]ORT Braude College

August 2, 2021

Problem – Reconstruct a signal $u_d^\star$ from noisy measurements

$$d = Au_d^\star + \varepsilon, \tag{1}$$

where ε is noise.

Structure

- Matrix A is big, sparse, (probably) overdetermined
- Available prior data (measurements d and "clean" images $u_d^\star$)

Goal – Fuse fast feasibility algorithms with big data in practical and theoretical sound manner.

Feasibility approach defines hyperplane for equations in system:

$$\mathcal{C}_{d,i} \triangleq \left\{u : \left\langle a^i, u \right\rangle = d_i \right\}, \quad \text{for } i = 1, 2, \ldots, m. \tag{2}$$

When $\varepsilon = 0$, we can solve convex feasibility problem

$$\text{Find } \tilde{u}_d \in \mathcal{C}_d \triangleq \bigcap_{i=1}^{m} \mathcal{C}_{d,i}. \tag{CFP}$$

Simple successive projections yields updates $u^{k+1} = P_{\mathcal{C}_{d,i_k}}(u^k)$.

Note – Several fast projection algorithms exist for solving (CFP).

Note – No regularization in (CFP), $\tilde{u}_d$ might poorly approximate $u_d^\star$.

Problem 1 – When $\varepsilon \neq 0$, we often have $\mathcal{C}_d = \emptyset$.

Partial Solution – Relax projections, early stopping, *etc*

Problem 2 – How can we incorporate regularization? And data?

Partial Solution – Superiorization augments projection algorithms by efficiently steering iterates to feasibility while pushing down regularizer value.[1]

But what about using available data?

[1] See papers of workshop participants Yair Censor and Aviv Gibali.

No Data, No Noise – With $\varepsilon = 0$ and regularizer (*e.g.* TV), solve

$$\min_{u \in \mathcal{C}} g(u) \tag{3}$$

by projected gradient (assuming we can project onto $\mathcal{C}$)

$$u^{k+1} = P_{\mathcal{C}}\left(u^k - \alpha \nabla g(u^k)\right). \tag{4}$$

With Data, No Noise – Use nonexpansive (*i.e.* 1-Lipschitz) R_Θ with

$$u^{k+1} = P_{\mathcal{C}}(R_\Theta(u^k)). \tag{5}$$

Key Idea – Replace analytic term $(\mathrm{I} - \alpha \nabla g)$ with learned term R_Θ.

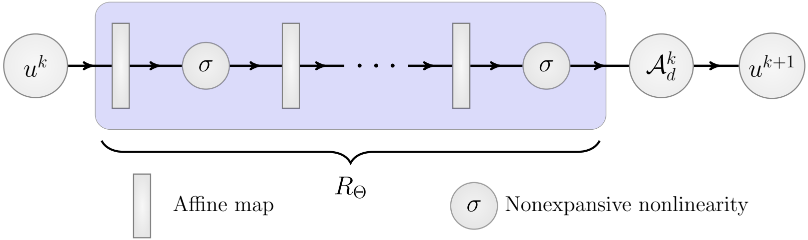


Figure 1: We replace feasibility algorithm $u^{k+1} = \mathcal{A}^k(u^k)$ with modified method $u^{k+1} = \mathcal{A}^k(R_\Theta(u^k))$. Regularization R_Θ takes form of neural network with parameters Θ.

Let $\{\mathcal{A}^k\}$ be sequence of update operators of a fast projection algorithm (*e.g.* DROP, CARP).

	Algorithm	Problem
Classic	$u^{k+1} = \mathcal{A}^k(u^k)$	$\mathcal{C}_d = \bigcap_{i=1}^m \mathcal{C}_{d,i}$
Data-Driven	$u^{k+1} = \mathcal{A}^k(R_\Theta(u^k))$	$\mathcal{C}_{d,\Theta} = \bigcap_{i=1}^m \mathrm{Fix}(\mathcal{A}^k \circ R_\Theta)$

Table 1: Comparison of Approaches

(Informal) Theorem: If R_Θ and $\{\mathcal{A}^k\}$ are 1-Lipschitz and $\mathcal{C}_{d,\Theta} \neq \emptyset$, then $u^k \to u_d^\Theta \in \mathcal{C}_{d,\Theta}$.

Key Idea – We can tune parameters Θ so that R_Θ gives

$$u_d^\star \approx u_d^\Theta = \lim_{k\to\infty} \mathcal{A}^k(R_\Theta(u^k)). \tag{6}$$

Training – Pick optimal $\Theta^\star$ that solves[2]

$$\min_\Theta \mathbb{E}_{d\sim\mathcal{D}}\left[\|u_d^\Theta - u_d^\star\|^2\right] \tag{7}$$

In practice, we consider $\{(d, u_d^\star)\}_{d\in\mathcal{D}}$ and minimize empirical risk

$$\min_\Theta \frac{1}{|\mathcal{D}|}\sum_{d\in\mathcal{D}} \|u_d^\Theta - u_d^\star\|^2. \tag{8}$$

[2] Training details outside scope of talk. See Fixed Point Networks paper. Note training can be unsupervised, *i.e.* we do not need measurement samples d and signals $u_d^\star$ to come in pairs.

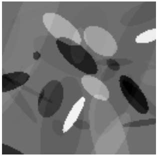
Setup – We give numerical examples for X-ray CT reconstruction using $A \in \mathbb{R}^{16384 \times 5490}$ and $\varepsilon = 1.5\%$ Gaussian noise.

Problem 1 – Dataset of random ellipses

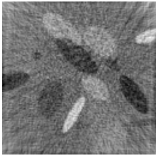
Problem 2 – Downscaled LoDoPab CT dataset (realistic)

Algorithm – Use $\mathcal{A}^k =$DROP and R_Θ = ResNet with Convolutions.

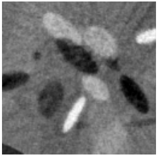
Code – All code is available and can be readily run on Google Colab (thus, from any device, including your phone).



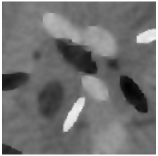
Ground Truth
SSIM: 1.000
PSNR: ∞



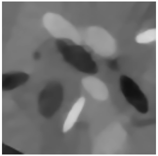
FBP
SSIM: 0.273
PSNR: 18.224



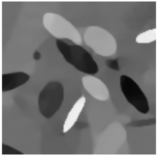
TVS
SSIM: 0.582
PSNR: 25.88



TVM
SSIM: 0.786
PSNR: 27.80

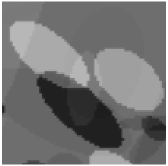


Unrolled
SSIM: 0.811
PSNR: 26.01

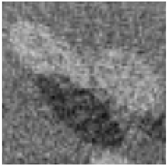


F-FPN
SSIM: 0.900
PSNR: 30.94

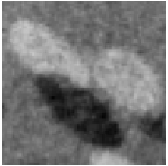
Figure 2: Ellipse reconstructions.



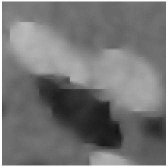
Ground Truth
SSIM: 1.000
PSNR: ∞



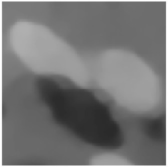
FBP
SSIM: 0.273
PSNR: 18.224



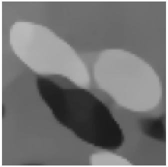
TVS
SSIM: 0.582
PSNR: 25.88



TVM
SSIM: 0.786
PSNR: 27.80



Unrolled
SSIM: 0.811
PSNR: 26.01

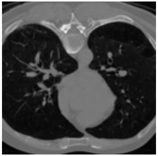


F-FPN
SSIM: 0.900
PSNR: 30.94

Figure 3: Zoomed-in ellipse reconstructions.

Method	Avg. PSNR (dB)	Avg. SSIM	# Parameters
Filtered Backprojection	17.79	0.211	1
TV Superiorization	27.35	0.721	2
TV Minimization	28.55	0.772	4
Unrolled Network	30.39	0.859	96,307
F-FPN (proposed)	**31.30**	**0.877**	96,307

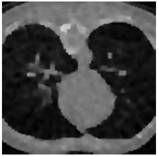
Table 2: Average PSNR and SSIM on the 1,000 image ellipse testing dataset.



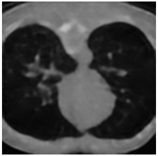
Ground Truth
SSIM: 1.000
PSNR: ∞

FBP
SSIM: 0.273
PSNR: 18.224

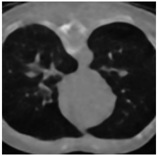
TVS
SSIM: 0.582
PSNR: 25.88



TVM
SSIM: 0.761
PSNR: 26.85

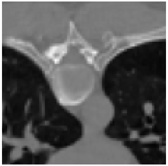


Unrolled
SSIM: 0.787
PSNR: 27.14

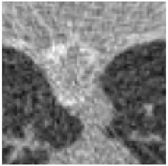


F-FPN
SSIM: 0.827
PSNR: 28.82

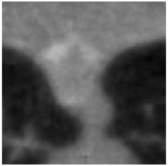
Figure 4: LoDoPab reconstructions.



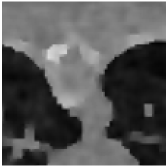
Ground Truth
SSIM: 1.000
PSNR: ∞



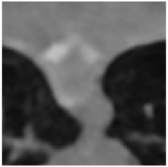
FBP
SSIM: 0.273
PSNR: 18.224



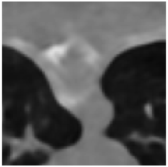
TVS
SSIM: 0.582
PSNR: 25.88



TVM
SSIM: 0.761
PSNR: 26.85



Unrolled
SSIM: 0.787
PSNR: 27.14



F-FPN
SSIM: 0.827
PSNR: 28.82

Figure 5: Zoomed-in LoDoPab reconstructions.

Method	Avg. PSNR (dB)	Avg. SSIM	# Parameters
Filtered Backprojection	19.27	0.354	1
TV Superiorization	26.65	0.697	2
TV Minimization	28.52	0.765	4
Unrolled Network	29.30	0.800	96,307
F-FPN (proposed)	**30.46**	**0.832**	96,307

Table 3: Average PSNR/SSIM on the 2,000 image LoDoPab testing dataset.

This talk explored a few ideas.

- CFP has limited handling of noise and regularization
- CFP can be augmented with regularization operator
- Regularizer parameterization can be tuned using data

Contact: hheaton@g.ucla.edu / howardheaton.tech

arXiv Preprint: Feasibility-based Fixed Point Networks