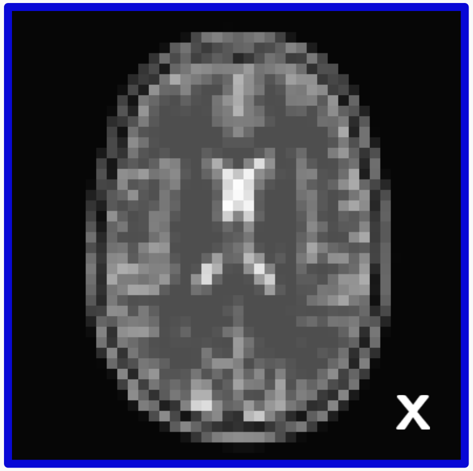
A Fast Algorithm for Structured Low-Rank Matrix Recovery with Applications to Undersampled MRI Reconstruction

Greg Ongie*, Mathews Jacob
Computational Biomedical Imaging Group (CBIG)
University of Iowa, Iowa City, Iowa.

April 14, 2016
ISBI 2016, Prague

THE UNIVERSITY OF IOWA

Compressed Sensing MRI Reconstruction



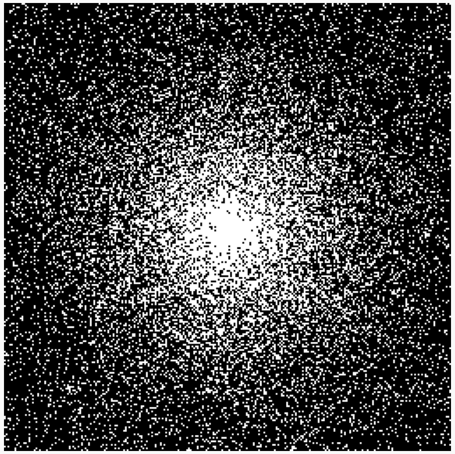
$$\min_{\mathbf{x}} ||\mathbf{A}\mathbf{x} - \mathbf{b}||^2 + \lambda\, \varphi(\mathbf{x})$$

recovery posed in discrete image domain

smoothness/sparsity regularization penalty

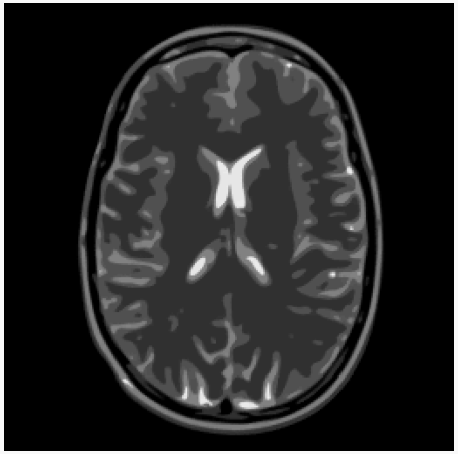
Example: TV-minimization

$$\min_{\mathbf{x}} ||\mathbf{A}\mathbf{x} - \mathbf{b}||^2 + \lambda\, ||\mathbf{x}||_{\mathbf{TV}}$$



25% k-space

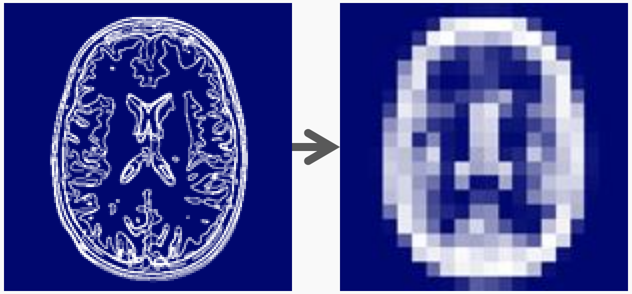
TV-min →



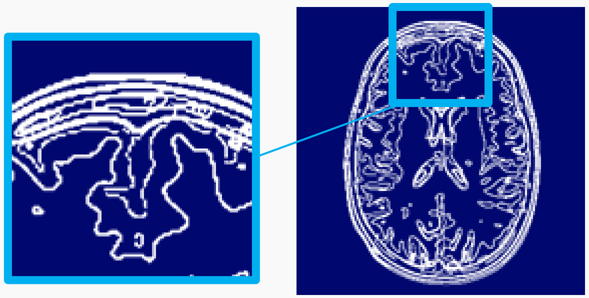
Rel. Error = 5%

Drawbacks to TV Minimization

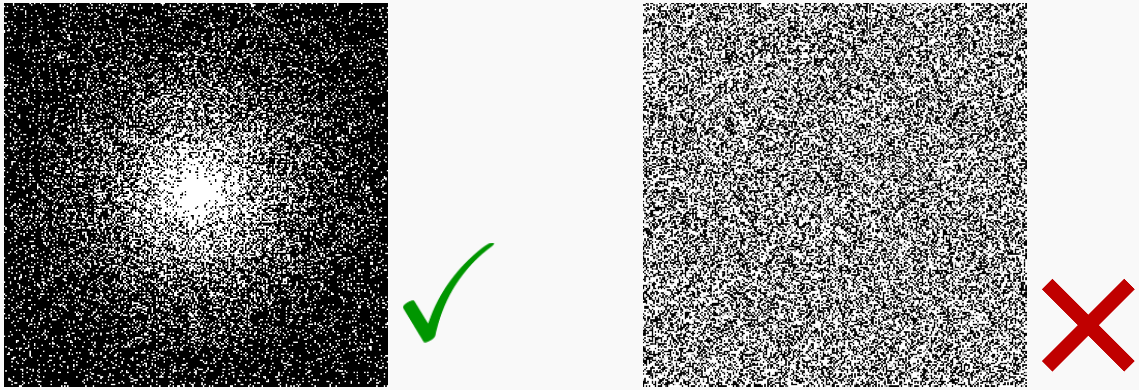
- **Discretization effects:**
  - Lose sparsity when discretizing to grid



- **Unable to exploit *structured* sparsity:**
  - Images have smooth, connected edges

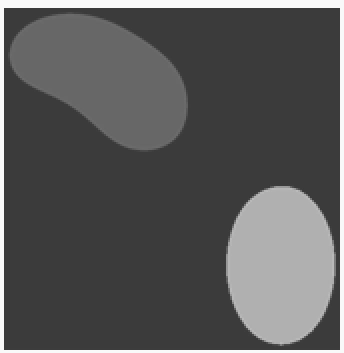


- **Sensitive to k-space sampling pattern** [Krahmer & Ward, 2014]

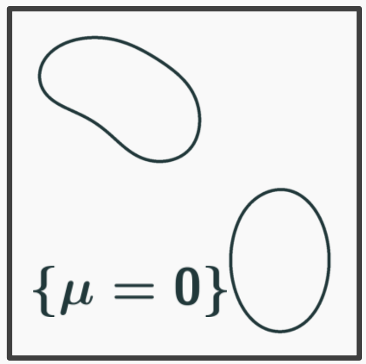


Off-the-Grid alternative to TV
[O. & Jacob, ISBI 2015], [O. & Jacob, SampTA 2015]

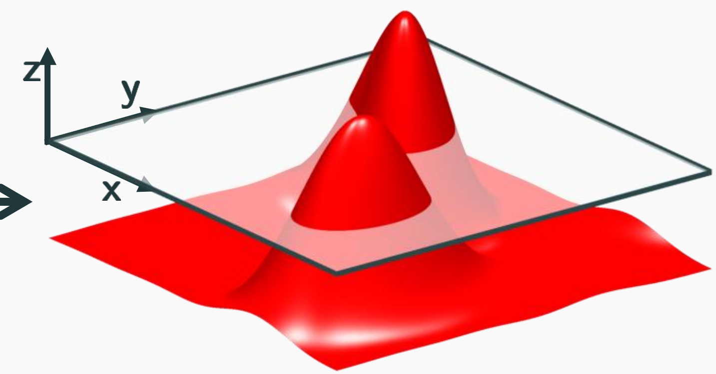
- Continuous domain piecewise constant image model
- Model edge set as zero-set of a 2-D band-limited function



image



edge set

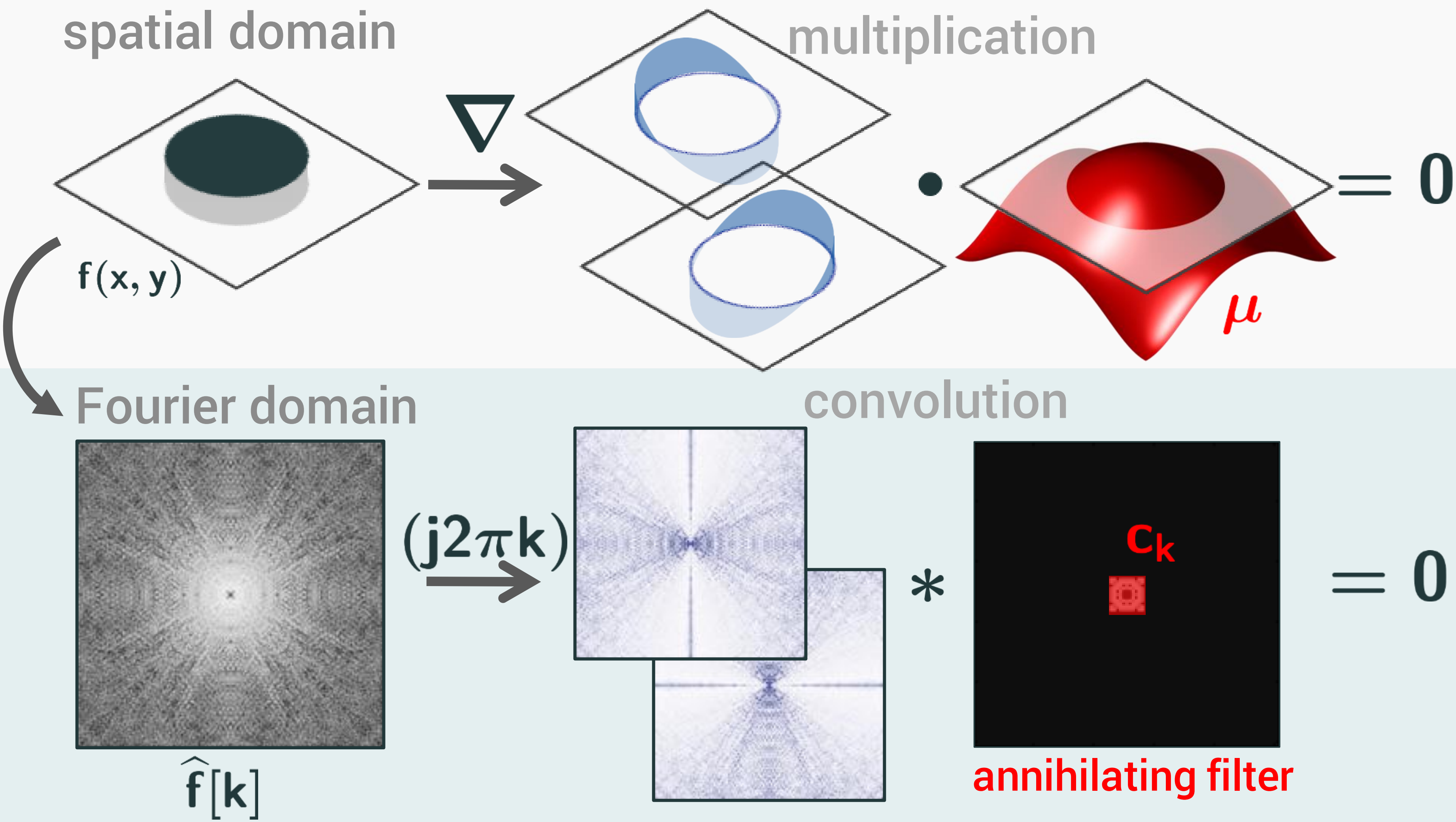


$$\mu(x, y) = \sum_{(k,l)\in\Lambda} c_{k,l}\, e^{j2\pi(kx+ly)}$$

"Finite-rate-of-innovation curve"

[Pan et al., IEEE TIP 2014]

2-D PWC functions satisfy an annihilation relation



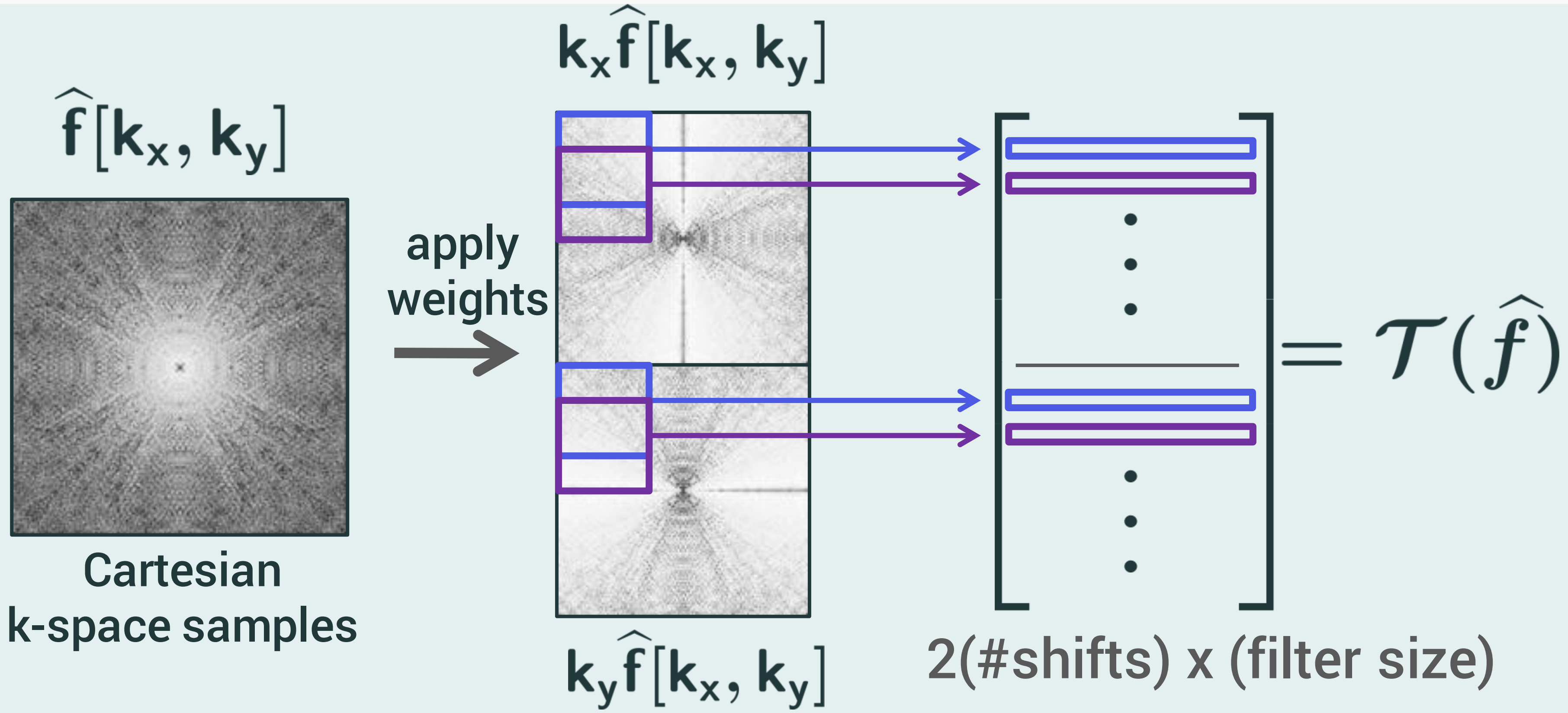
Annihilation relation: $\sum_k \widehat{\nabla f}[\ell - k] c_k = 0$

Matrix representation of annihilation

$$\mathcal{T}(\widehat{f})\,\color{red}{c} = 0$$

2-D convolution matrix
built from k-space samples

vector of filter coefficients



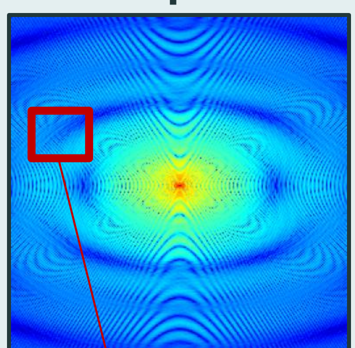
Matrix representation of annihilation

$$\boxed{\mathcal{T}(\widehat{f})}\,\boldsymbol{c} = 0$$

Basis of algorithms: Annihilation matrix is low-rank

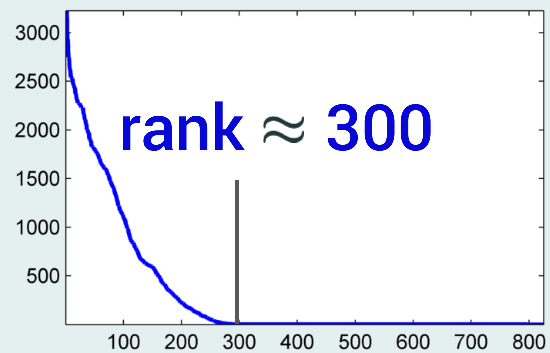
Ex: Shepp-Logan

k-space



Filter size: 33x25

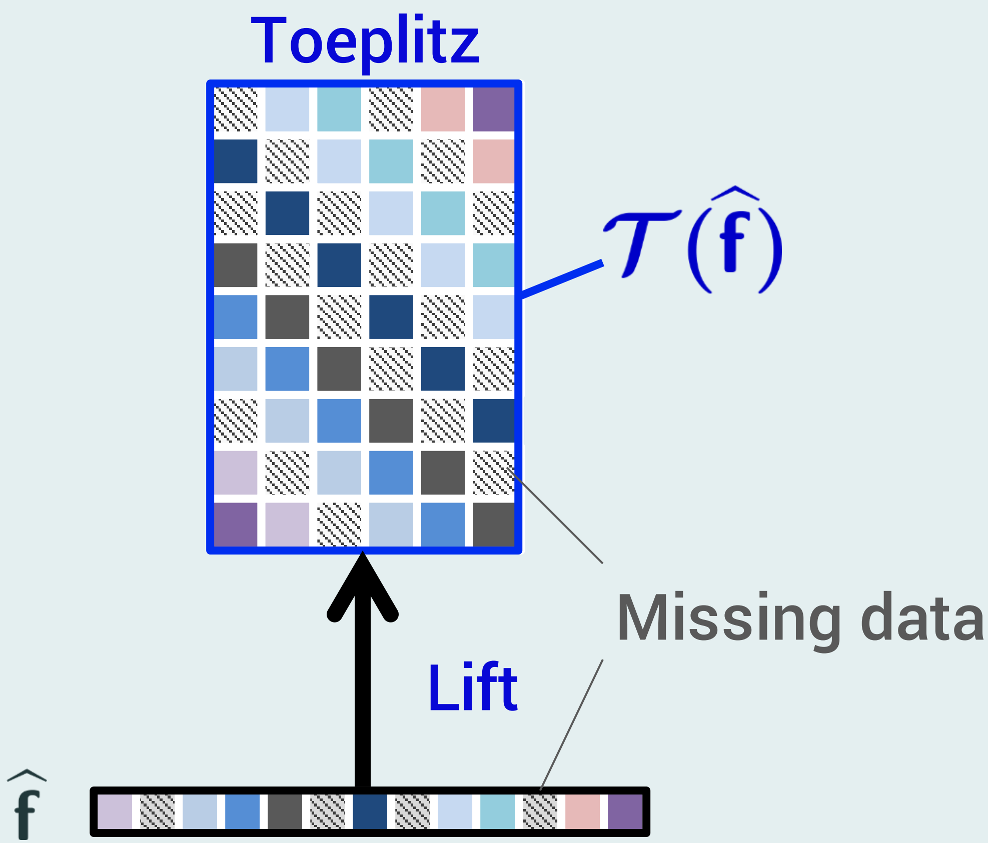
$\sigma(\mathcal{T}(\widehat{f}))$



Pose recovery from missing k-space data as **structured low-rank matrix completion** problem

$$\min_{\widehat{\mathbf{f}}} \quad \mathrm{rank}[\mathcal{T}(\widehat{\mathbf{f}})] \quad \textbf{s.t.} \quad \widehat{\mathbf{f}}[\mathbf{k}] = \widehat{\mathbf{f}}_0[\mathbf{k}], \mathbf{k} \in \boldsymbol{\Omega}$$

1-D Example:



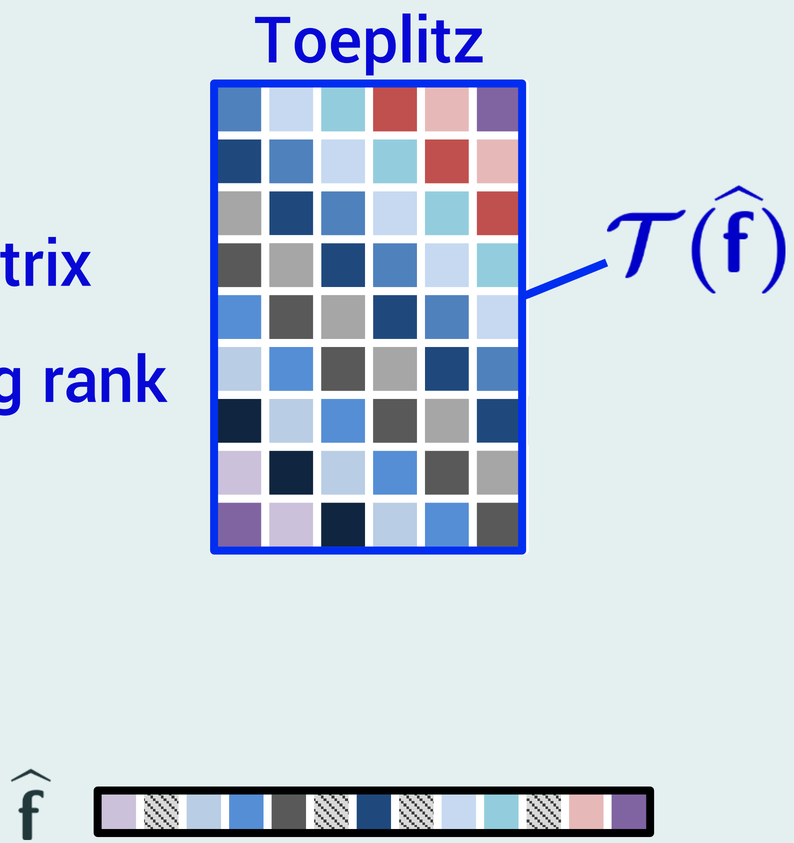
Pose recovery from missing k-space data as **structured low-rank matrix completion** problem

$$\min_{\widehat{\mathbf{f}}} \quad \mathrm{rank}[\mathcal{T}(\widehat{\mathbf{f}})] \quad \textbf{s.t.} \quad \widehat{\mathbf{f}}[\mathbf{k}] = \widehat{\mathbf{f}}_0[\mathbf{k}], \mathbf{k} \in \mathbf{\Omega}$$

1-D Example:

Complete matrix

By minimizing rank



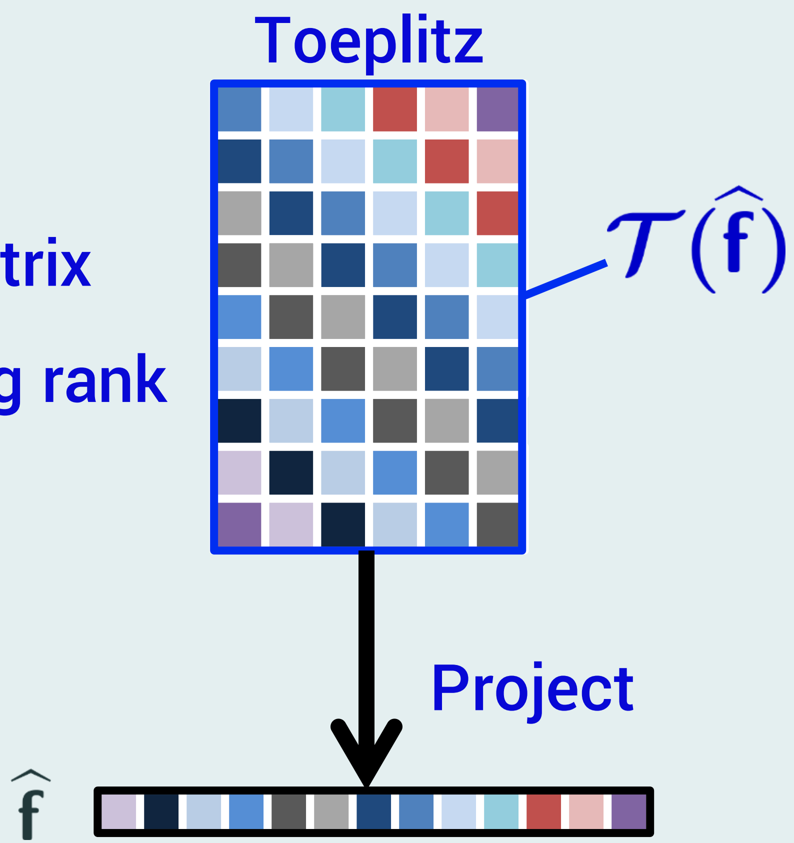
Pose recovery from missing k-space data as **structured low-rank matrix completion** problem

$$\min_{\widehat{\mathbf{f}}} \quad \mathrm{rank}[\mathcal{T}(\widehat{\mathbf{f}})] \quad \textbf{s.t.} \quad \widehat{\mathbf{f}}[\mathbf{k}] = \widehat{\mathbf{f}}_0[\mathbf{k}], \mathbf{k} \in \boldsymbol{\Omega}$$

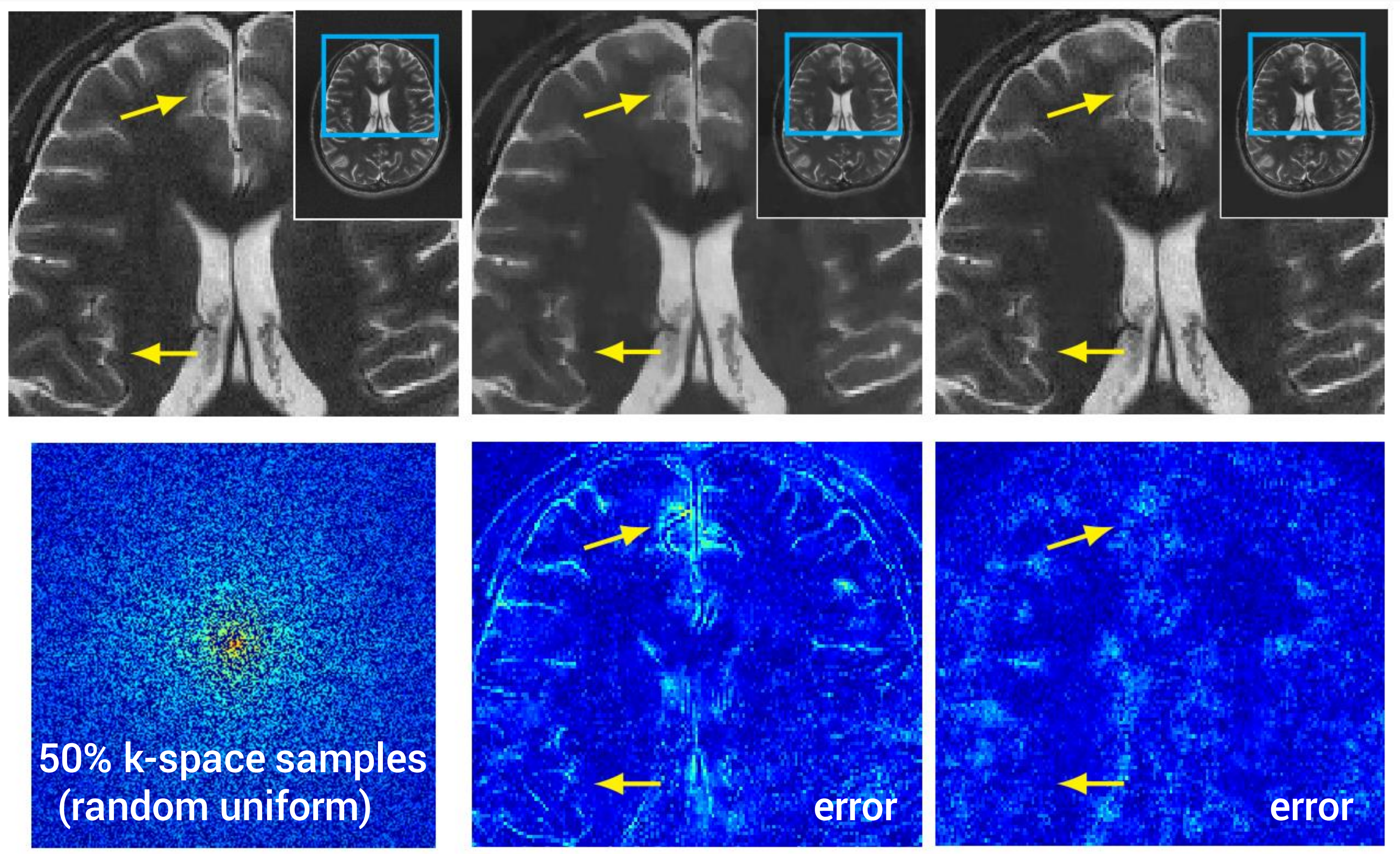
1-D Example:

Complete matrix

By minimizing rank



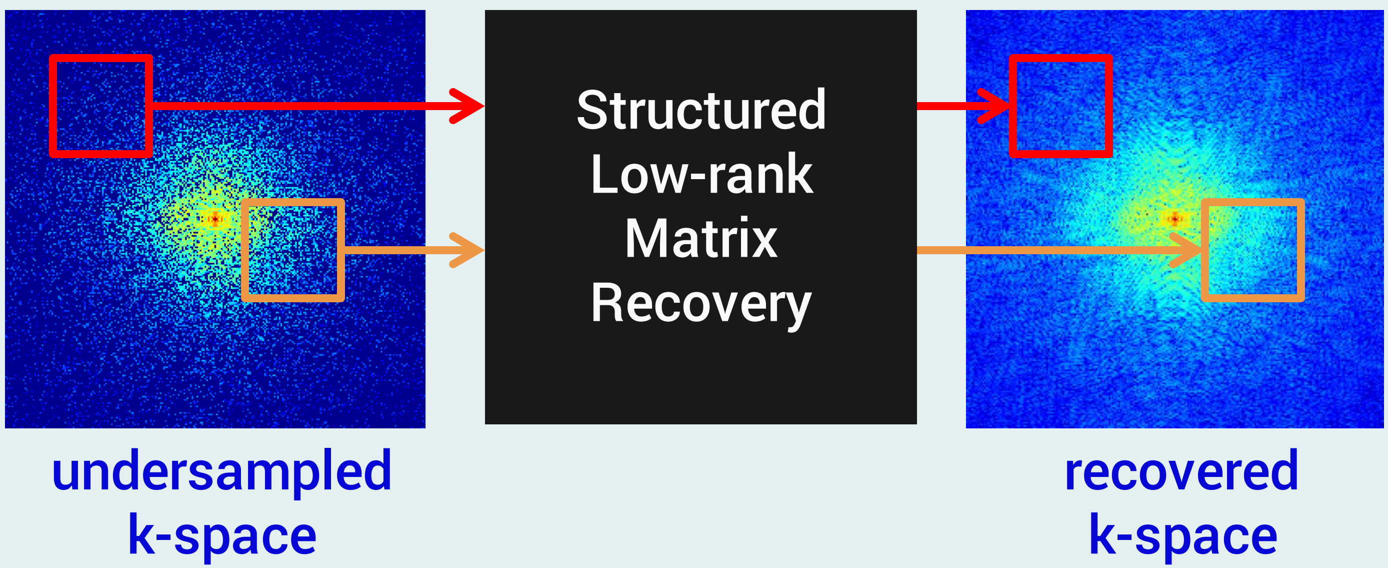
Fully sampled TV (SNR=17.8dB) Proposed (SNR=19.0dB)



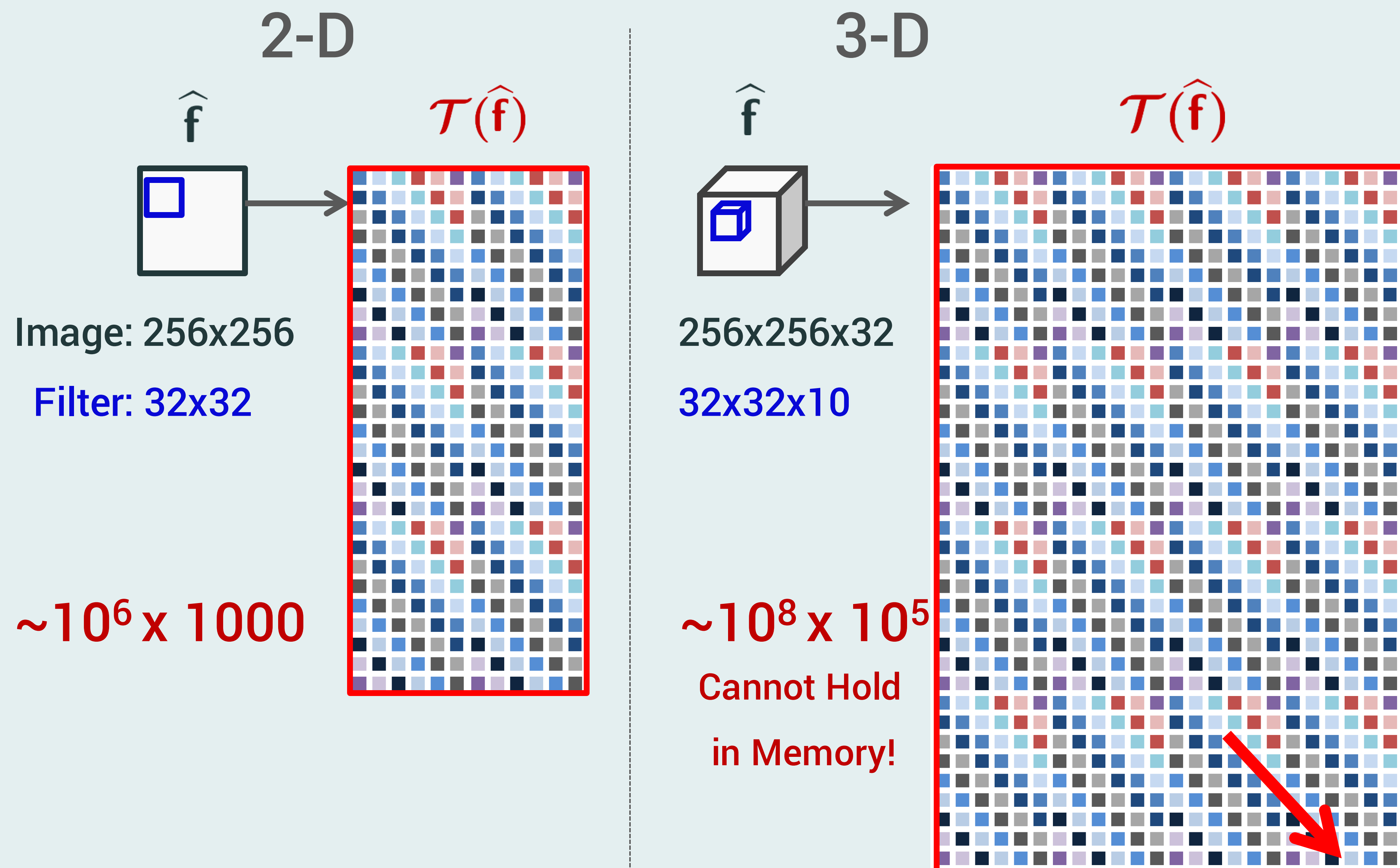
(Retrospective undersampled 4-coil data compressed to single virtual coil)

Emerging Trend: Fourier domain low-rank priors for MRI reconstruction

- **SAKE** [Shin et al., MRM 2014]
  – Image model: Smooth coil sensitivity maps (parallel imaging)
- **LORAKS** [Haldar, TMI 2014]
  – Image model: Support limited & smooth phase
- **ALOHA** [Jin et al., ISBI 2015]
  – Image model: Transform sparse/Finite-rate-of-innovation
- **Off-the-grid models** [O. & Jacob, ISBI 2015], [O. & Jacob, SampTA 2015]



Main challenge: Computational complexity



Outline

1. Prior Art
2. Proposed Algorithm
3. Applications

Cadzow methods/Alternating projections

["SAKE," Shin et al., 2014], ["LORAKS," Haldar, 2014]

$$\min_{\widehat{\mathbf{f}}} \|\mathbf{A}\widehat{\mathbf{f}} - \mathbf{b}\|^2 \quad \textbf{s.t.} \quad \mathbf{X} = \mathcal{T}(\widehat{\mathbf{f}})$$

$$\operatorname{rank} \mathbf{X} \leq \mathbf{r}$$

No convex relaxations. Use rank estimate.

Cadzow methods/Alternating projections

["SAKE," Shin et al., 2014], ["LORAKS," Haldar, 2014]

$$\min_{\widehat{\mathbf{f}}} \|\mathbf{A}\widehat{\mathbf{f}} - \mathbf{b}\|^2 \quad \text{s.t.} \quad \boxed{\mathbf{X} = \mathcal{T}(\widehat{\mathbf{f}})}$$

$$\boxed{\operatorname{rank} \mathbf{X} \leq \mathbf{r}}$$

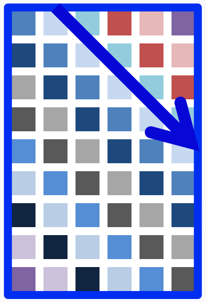
Alternating projection algorithm (Cadzow)

1. Project onto space of rank r matrices

    -Compute *truncated SVD:* $\mathbf{X}^{\star} = \mathbf{U}\boldsymbol{\Sigma}_r\mathbf{V}^H$

2. Project onto space of structured matrices

    -Average along "diagonals"



Cadzow methods/Alternating projections

["SAKE," Shin et al., 2014], ["LORAKS," Haldar, 2014]

$$\min_{\widehat{\mathbf{f}}} ||\mathbf{A}\widehat{\mathbf{f}} - \mathbf{b}||^2 \quad \mathbf{s.t.} \quad \mathbf{X} = \mathcal{T}(\widehat{\mathbf{f}})$$

$$\boxed{\mathrm{rank}\ \mathbf{X} \leq \mathbf{r}}$$

Drawbacks

- Highly non-convex
- Need estimate of rank bound r
- Complexity grows with r
- Naïve approach needs to store large matrix

Nuclear norm minimization

$$\min_{\widehat{\mathbf{f}}} ||\mathbf{A}\widehat{\mathbf{f}} - \mathbf{b}||^2 + \lambda ||\mathbf{X}||_* \quad \text{s.t.} \quad \mathbf{X} = \mathcal{T}(\widehat{\mathbf{f}})$$

ADMM = Singular value thresholding (SVT)

1. Singular value thresholding step

   -compute *full SVD* of X!

2. Solve linear least squares problem

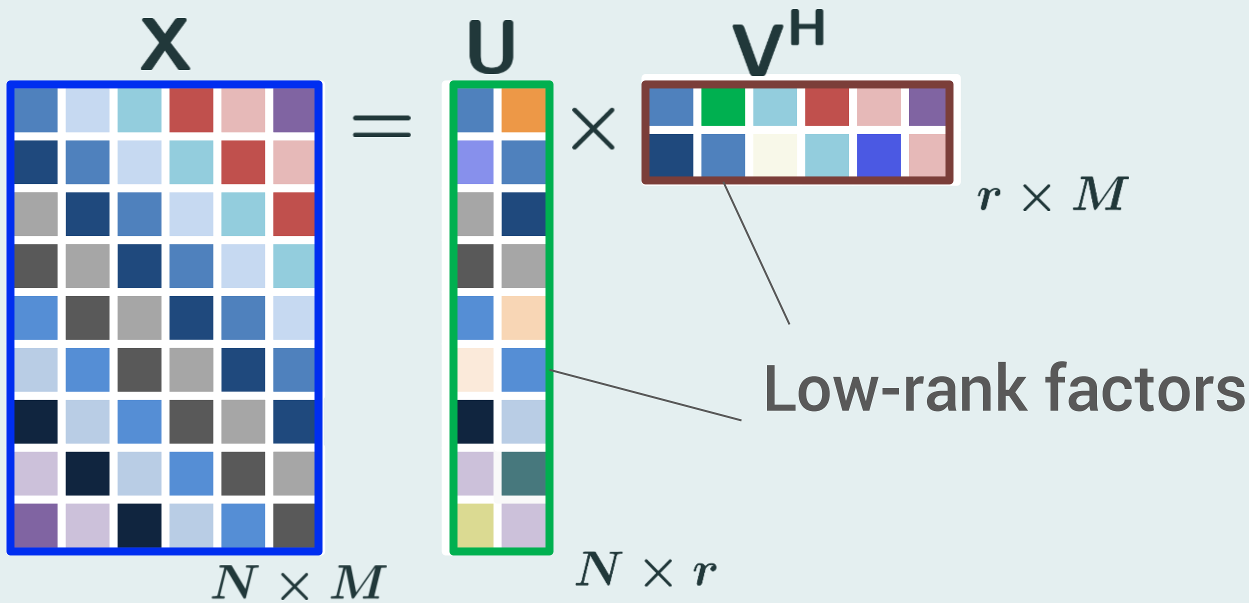
   -analytic solution or CG solve

Nuclear norm minimization

$$\min_{\widehat{\mathbf{f}}} \|\mathbf{A}\widehat{\mathbf{f}} - \mathbf{b}\|^2 + \lambda \|\mathbf{X}\|_* \quad \text{s.t.} \quad \mathbf{X} = \mathcal{T}(\widehat{\mathbf{f}})$$

"U,V factorization trick"

$$\|\mathbf{X}\|_* = \min_{\mathbf{X}=\mathbf{U}\mathbf{V}^H} \frac{1}{2}\left(\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2\right)$$



Nuclear norm minimization *with U,V factorization*

[O.& Jacob, SampTA 2015], ["ALOHA", Jin et al., ISBI 2015]

$$\min_{\widehat{\mathbf{f}},\mathbf{U},\mathbf{V}} \|\mathbf{A}\widehat{\mathbf{f}} - \mathbf{b}\|^2 + \frac{\lambda}{2}\left(\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2\right)$$

$$\text{s.t.} \quad \mathbf{U}\mathbf{V}^H = \mathcal{T}(\widehat{\mathbf{f}})$$

UV factorization approach

~~1. Singular value thresholding step~~

~~-compute *full SVD* of X!~~

SVD-free → fast matrix inversion steps

2. Solve linear least squares problem

-analytic solution or CG solve

Nuclear norm minimization *with U,V factorization*

[O.& Jacob, SampTA 2015], ["ALOHA", Jin et al., ISBI 2015]

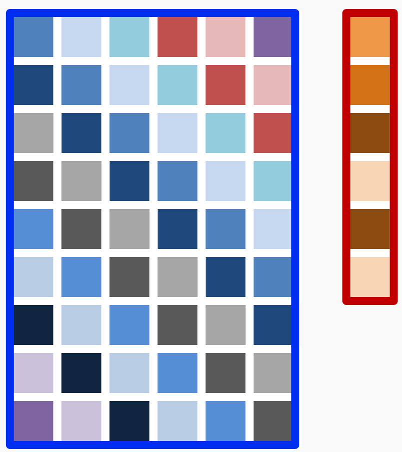
$$\min_{\widehat{\mathbf{f}},\mathbf{U},\mathbf{V}} \|\mathbf{A}\widehat{\mathbf{f}} - \mathbf{b}\|^2 + \frac{\lambda}{2}\left(\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2\right)$$

$$\text{s.t.} \quad \mathbf{U}\mathbf{V}^H = \mathcal{T}(\widehat{\mathbf{f}})$$

Drawbacks

- Big memory footprint—not feasible for 3-D
- U,V trick is non-convex

None of current approaches exploit structure of matrix liftings (e.g. Hankel/Toeplitz)



$$\mathcal{T}(\hat{\mathbf{f}})\,\mathbf{c} \approx \hat{\mathbf{f}} * \mathbf{c}$$

Can we exploit this structure to give a more efficient algorithm?

Outline

1. Prior Art
2. Proposed Algorithm
3. Applications

Proposed Approach:
Adapt IRLS algorithm for nuclear norm minimization

- IRLS: Iterative Reweighted Least Squares
- Proposed for low-rank matrix completion in
  [Fornasier, Rauhut, & Ward, 2011], [Mohan & Fazel, 2012]
- Solves:

$$\min_{\mathbf{X}} \|\mathbf{X}\|_* + \lambda \|\mathbf{A}\mathbf{X} - \mathbf{B}\|_F^2$$

- Idea:

$$\|\mathbf{X}\|_* = \|\mathbf{X}\mathbf{W}^{\frac{1}{2}}\|_F^2$$
$$\mathbf{W} = (\mathbf{X}^H\mathbf{X})^{-1/2}$$

$\updownarrow$ Alternate

- Original IRLS: To recover low-rank matrix X, iterate

$$\mathbf{W} \leftarrow (\mathbf{X}^{\mathbf{H}}\mathbf{X} + \epsilon \mathbf{I})^{-\frac{1}{2}}$$

$$\mathbf{X} \leftarrow \arg\min_{\mathbf{X}} \|\mathbf{X}\mathbf{W}^{\frac{1}{2}}\|_{\mathbf{F}}^{2} + \lambda \|\mathbf{A}\mathbf{X} - \mathbf{B}\|_{\mathbf{F}}^{2}$$

- Original IRLS: To recover low-rank matrix X, iterate

$$\mathbf{W} \leftarrow (\mathbf{X}^{\mathbf{H}}\mathbf{X} + \epsilon \mathbf{I})^{-\frac{1}{2}}$$

$$\mathbf{X} \leftarrow \arg\min_{\mathbf{X}} \|\mathbf{X}\mathbf{W}^{\frac{1}{2}}\|_{\mathbf{F}}^{2} + \lambda \|\mathbf{A}\mathbf{X} - \mathbf{B}\|_{\mathbf{F}}^{2}$$

- We adapt to structured case: $\mathbf{X} = \mathcal{T}(\widehat{f})$

$$\mathbf{W} \leftarrow (\mathcal{T}(\widehat{\mathbf{f}})^{\mathbf{H}}\mathcal{T}(\widehat{\mathbf{f}}) + \epsilon \mathbf{I})^{-\frac{1}{2}}$$

$$\widehat{\mathbf{f}} \leftarrow \arg\min_{\widehat{\mathbf{f}}} \|\mathcal{T}(\widehat{\mathbf{f}})\mathbf{W}^{\frac{1}{2}}\|_{\mathbf{F}}^{2} + \lambda \|\mathbf{A}\widehat{\mathbf{f}} - \mathbf{b}\|^{2}$$

- Original IRLS: To recover low-rank matrix X, iterate

$$\mathbf{W} \leftarrow (\mathbf{X}^H\mathbf{X} + \epsilon \mathbf{I})^{-\frac{1}{2}}$$

$$\mathbf{X} \leftarrow \arg\min_{\mathbf{X}} \|\mathbf{X}\mathbf{W}^{\frac{1}{2}}\|_F^2 + \lambda\|\mathbf{A}\mathbf{X} - \mathbf{B}\|_F^2$$

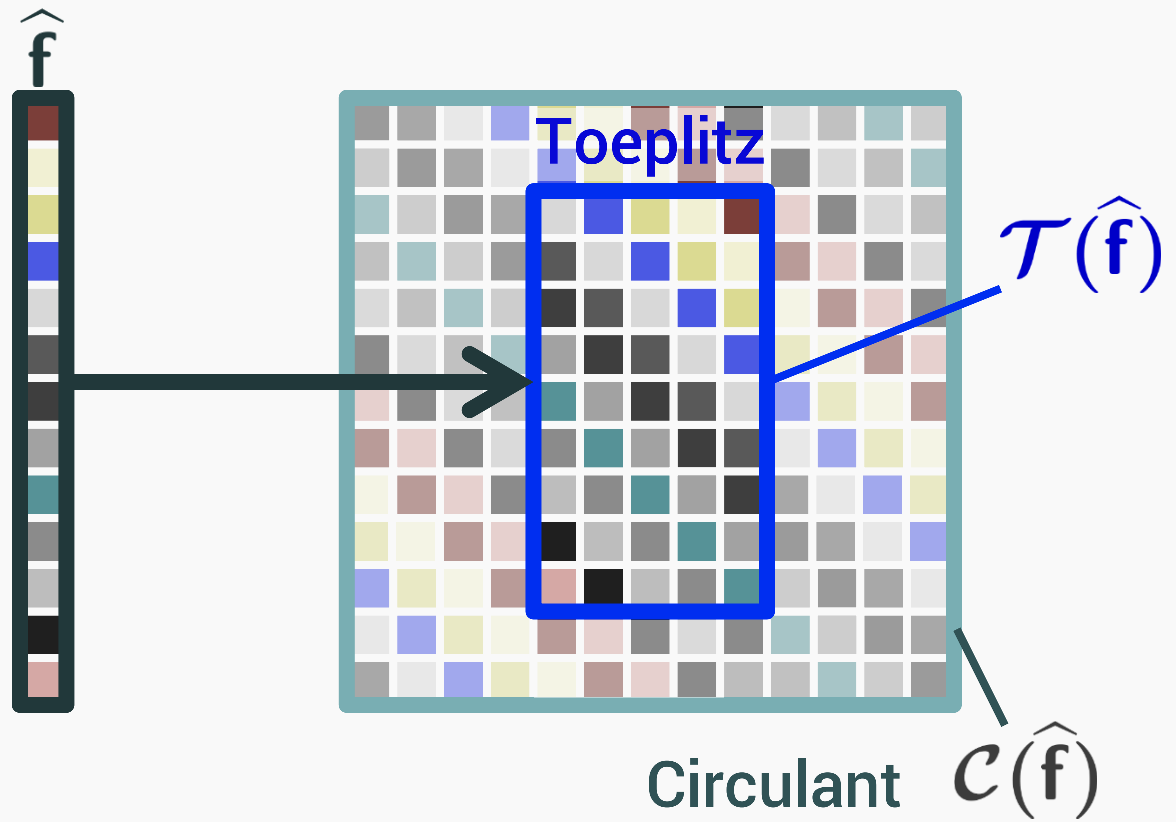
- We adapt to structured case: $\mathbf{X} = \mathcal{T}(\widehat{f})$

$$\mathbf{W} \leftarrow (\boxed{\mathcal{T}(\widehat{\mathbf{f}})^H\mathcal{T}(\widehat{\mathbf{f}})} + \epsilon \mathbf{I})^{-\frac{1}{2}}$$

$$\widehat{\mathbf{f}} \leftarrow \arg\min_{\widehat{\mathbf{f}}} \|\mathcal{T}(\widehat{\mathbf{f}})\mathbf{W}^{\frac{1}{2}}\|_F^2 + \lambda\|\mathbf{A}\widehat{\mathbf{f}} - \mathbf{b}\|^2$$

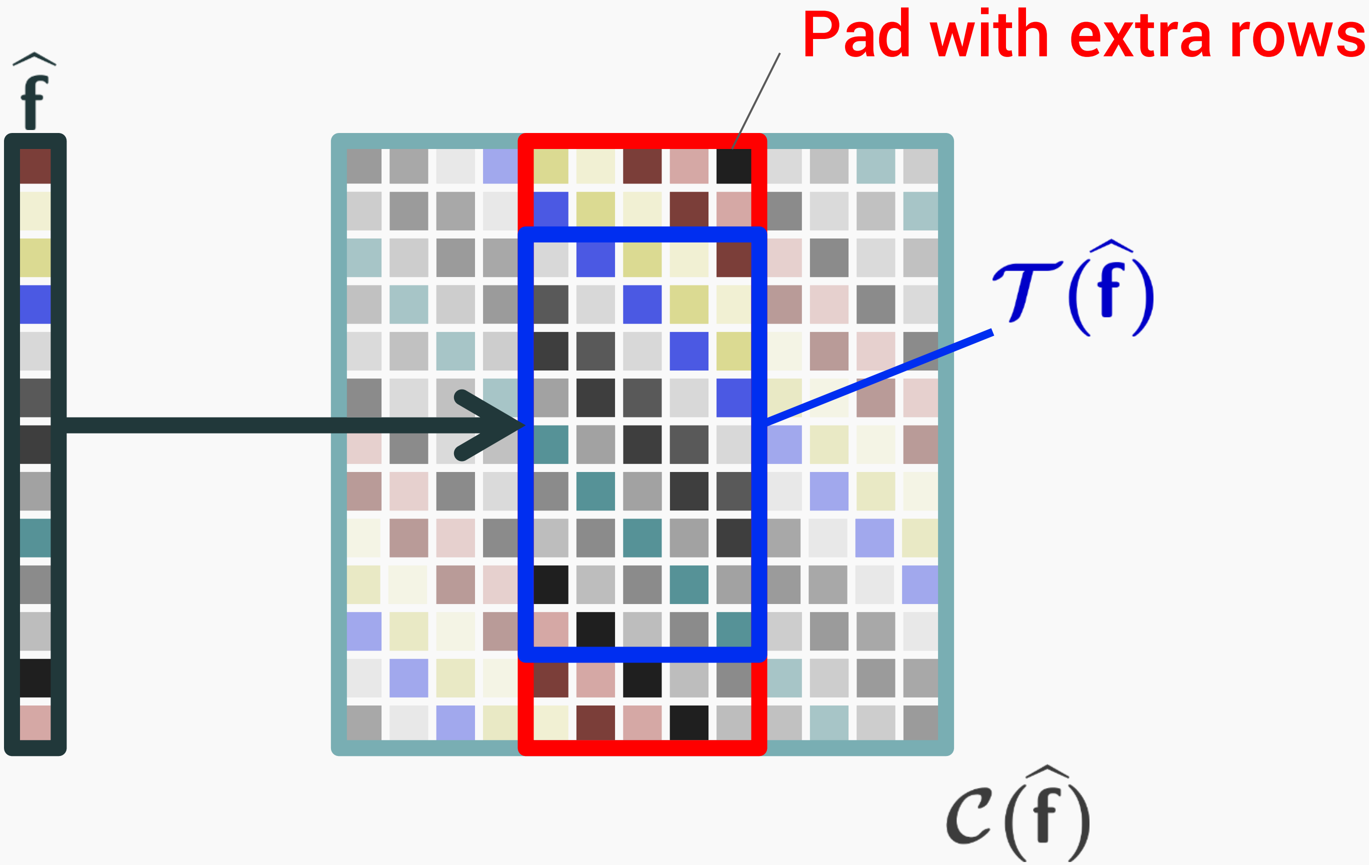
Without modification, this approach is still slow!

Idea 1: Embed Toeplitz lifting in circulant matrix



*Fast matrix-vector products with $\mathcal{T}(\widehat{\mathbf{f}})$ by FFTs

Idea 2: Approximate the matrix lifting



*Fast computation of $\mathcal{T}(\widehat{\mathbf{f}})^{\mathsf{H}}\mathcal{T}(\widehat{\mathbf{f}})$ by FFTs

Simplifications: Weight matrix update

$$\mathbf{W} \leftarrow \left(\mathcal{T}(\widehat{\mathbf{f}})^{\mathsf{H}}\mathcal{T}(\widehat{\mathbf{f}}) + \epsilon \mathbf{I}\right)^{-\frac{1}{2}}$$

Explicit form: $\mathbf{P}_{\Lambda}\mathbf{F}\,\mathrm{diag}(|\nabla \mathbf{f}|^2)\mathbf{F}^{\mathsf{H}}\mathbf{P}_{\Lambda}^{\mathsf{H}}$

- Build Gram matrix with two FFTs—no matrix product
- Computational cost:

  One eigen-decomposition of small Gram matrix

$$\cancel{\texttt{svd}\left(\overset{\mathcal{T}(\widehat{\mathbf{f}})}{\boxed{}}\right)} \longrightarrow \texttt{eig}\left(\overset{\mathcal{T}(\widehat{\mathbf{f}})^{\mathsf{H}}\mathcal{T}(\widehat{\mathbf{f}})}{\boxed{}}\right)$$

Simplifications: Least squares subproblem

$$\widehat{\mathbf{f}} \leftarrow \arg\min_{\widehat{\mathbf{f}}} \|\mathcal{T}(\widehat{\mathbf{f}})\mathbf{W}^{\frac{1}{2}}\|_F^2 + \lambda\|\mathbf{A}\widehat{\mathbf{f}} - \mathbf{b}\|^2$$

$$\downarrow$$

$$\|\widehat{\nabla \mathbf{f}} * \widehat{\mu}_{sos}\|^2$$

Convolution with *single filter*

Sum-of-squares average: $\mu_{sos} = \sqrt{\sum_{i=1}^{N} |\mathcal{F}^{-1}(\mathbf{w}_i)|^2}$

- Fast solution by CG iterations

Proposed GIRAF algorithm

- GIRAF = Generic Iterative Reweighted Annihilating Filter
- Adapt IRLS algorithm +simplifications based on structure

$$\min_{\widehat{\mathbf{f}}} \|\mathbf{A}\widehat{\mathbf{f}} - \mathbf{b}\|^2 + \lambda \|\mathbf{X}\|_* \quad \text{s.t.} \quad \mathbf{X} = \mathcal{T}(\widehat{\mathbf{f}})$$

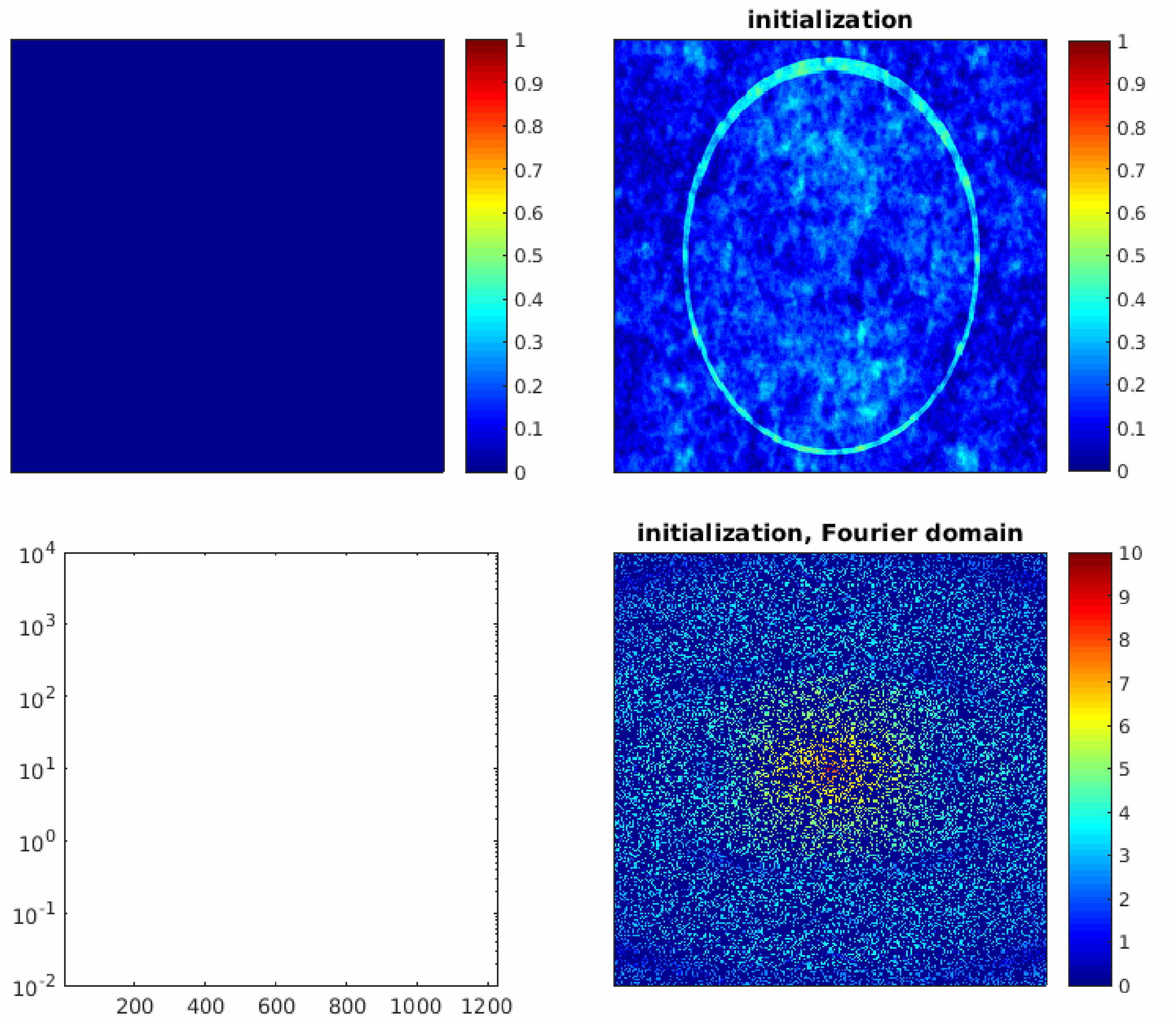
GIRAF algorithm

1. Update annihilating filter

   -Small eigen-decomposition $\mathcal{T}(\widehat{\mathbf{f}})^*\mathcal{T}(\widehat{\mathbf{f}})$

2. Least-squares annihilation

   -Solve with CG $\min \|\widehat{\nabla \mathbf{f}} * \widehat{\mu}_{\mathbf{sos}}\|^2$



1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
initialization
10^4
10^3
10^2
10^1
10^0
10^-1
10^-2
200
400
600
800
1000
1200
initialization, Fourier domain
10
9
8
7
6
5
4
3
2
1
0

GIRAF complexity similar to iterative reweighted TV minimization

IRLS TV-minimization

Edge weights

Image

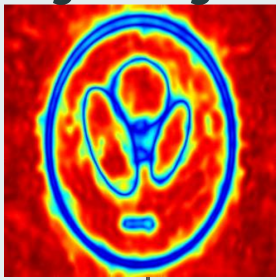
Local update:

$$w_{i,j} = \frac{1}{|(\nabla f)_{i,j}| + \epsilon}$$

Least-squares problem

GIRAF algorithm

Edge weights

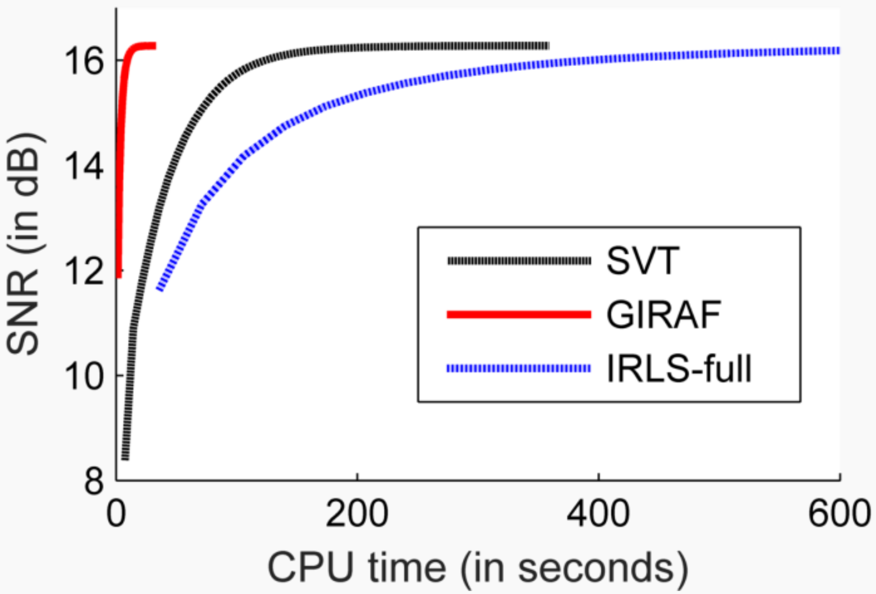


Image

Global update: w/small SVD

Least-squares problem

Convergence speed of GIRAF



CS-recovery from
50% random k-space samples
Image size: 256x256
Filter size: 15x15
(C-LORAKS spatial sparsity penalty)

Scaling with filter size

	15×15 filter		31×31 filter	
Algorithm	# iter	total	# iter	total
SVT	7	110s	11	790 s
GIRAF	6	20s	7	44 s

Table: iterations/CPU time to reach convergence tolerance of NMSE < 10^{-4}.

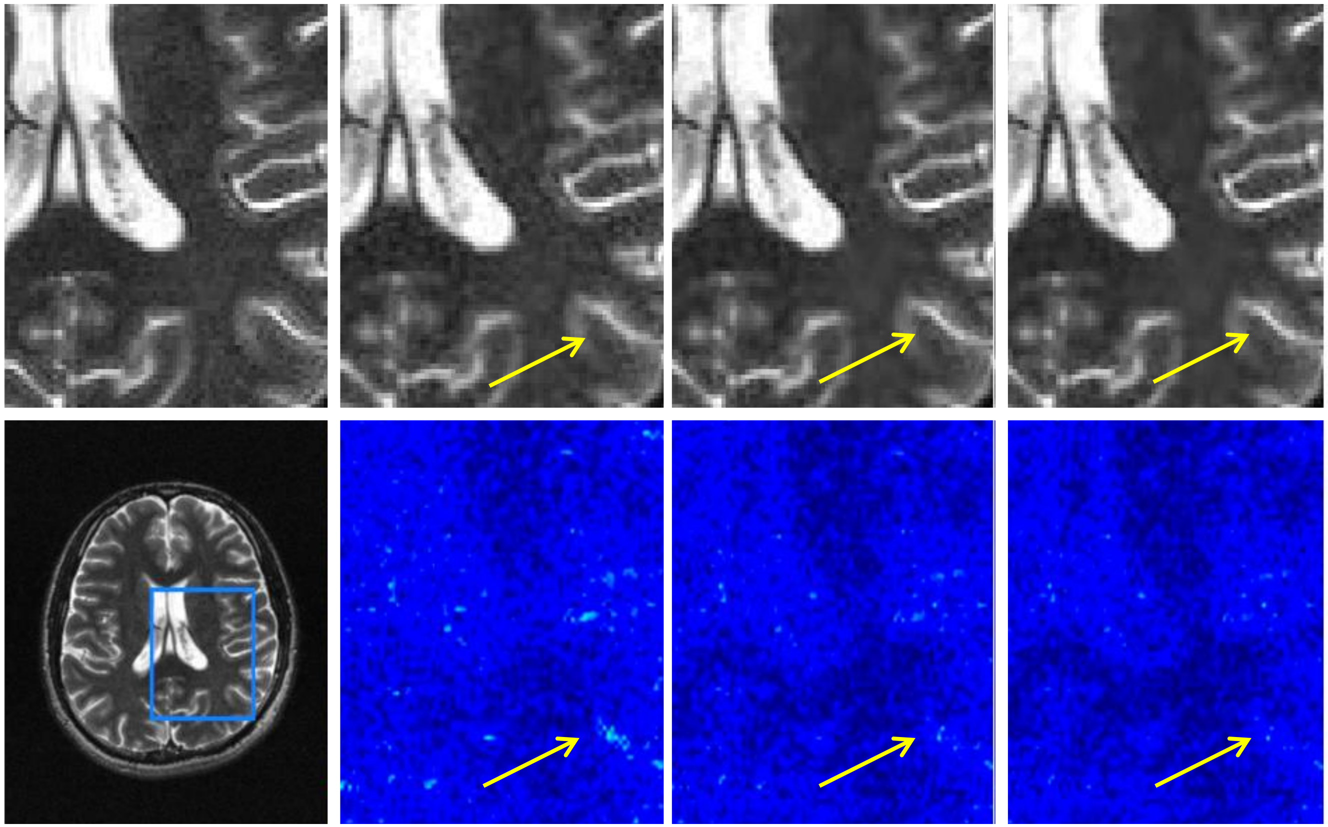
CS-recovery from
50% random k-space samples
Image size: 256x256

(gradient sparsity penalty)

Outline

1. Prior Art
2. Proposed Algorithm
3. Applications

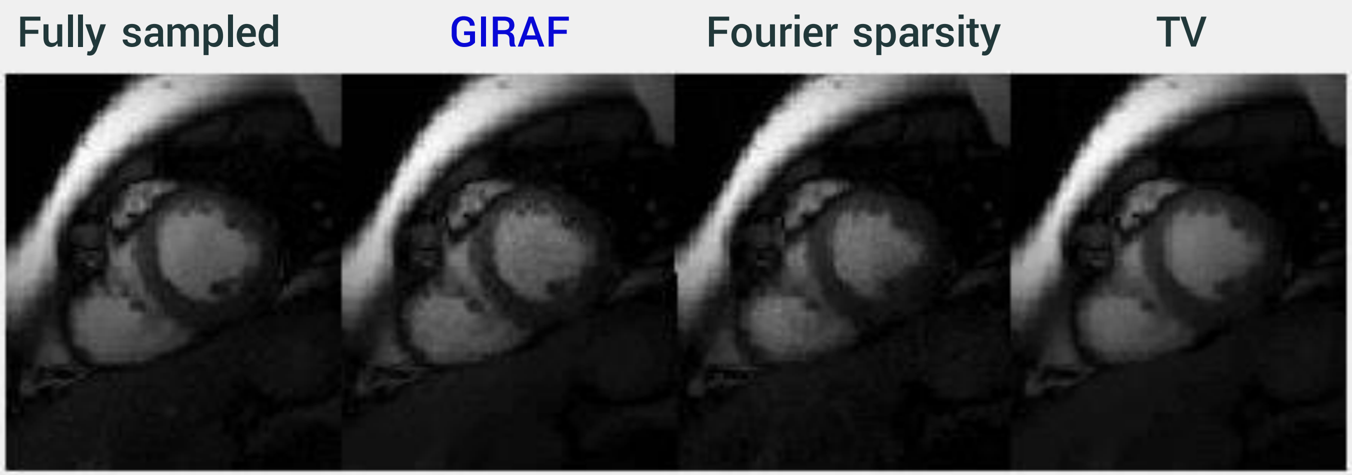
GIRAF enables larger filter sizes → improved compressed sensing recovery

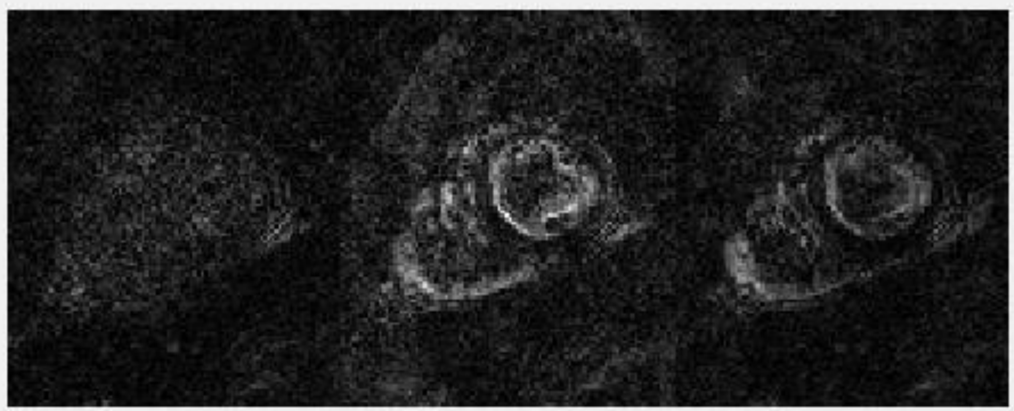


Fully sampled	15×15 filter	31×31 filter	45×45 filter
	SNR=18.5dB	SNR=19.4dB	SNR=19.8dB
	Runtime: 29s	Runtime: 65s	Runtime:117s

+1 dB improvement →

GIRAF enables extensions to multi-dimensional imaging: Dynamic MRI



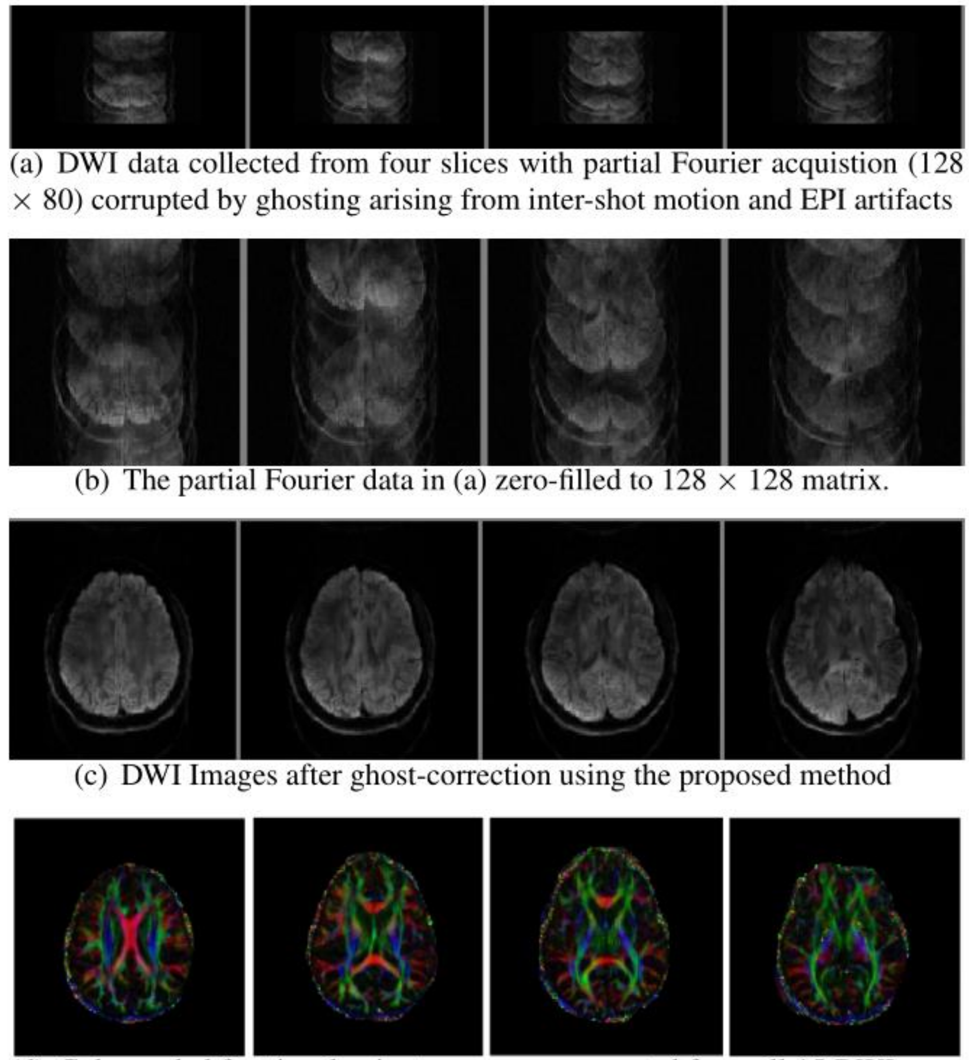


error images

[Balachandrasekaran, O., & Jacob, Submitted to ICIP 2016]

GIRAF enables extensions to multi-dimensional imaging: Diffusion Weighted Imaging

Correction of ghosting artifacts In DWI using annihilating filter framework and GIRAF



(a) DWI data collected from four slices with partial Fourier acquistion (128 $\times$ 80) corrupted by ghosting arising from inter-shot motion and EPI artifacts

(b) The partial Fourier data in (a) zero-filled to 128 $\times$ 128 matrix.

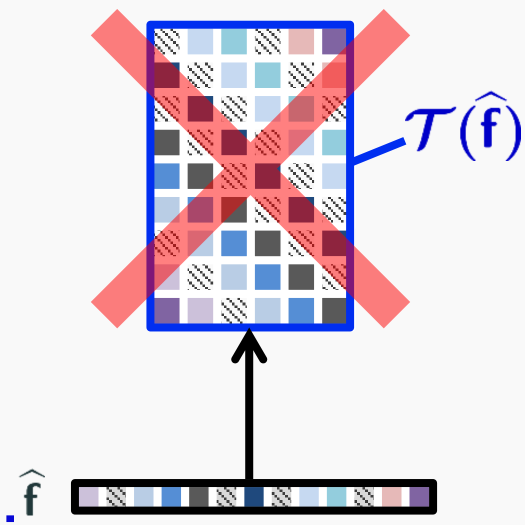
(c) DWI Images after ghost-correction using the proposed method

(d) Color-coded fractional anisotropy maps computed from all 15 DWIs corrected for ghosting artifacts

[Mani et al., ISMRM 2016]

Summary

- *Emerging trend*: Powerful Fourier domain low-rank penalties for MRI reconstruction
  - State-of-the-art, but computational challenging
  - Current algs. work directly with big "lifted" matrices
- New **GIRAF** algorithm for structured low-rank matrix formulations in MRI
  - Solves "lifted" problem in "unlifted" domain
  - No need to create and store large matrices



- Improves recovery & enables new applications
  - Larger filter sizes → improved CS recovery
  - Multi-dimensional imaging (DMRI, DWI, MRSI)

References

- Krahmer, F., & Ward, R. (2014) Stable and robust sampling strategies for compressive imaging. *Image Processing, IEEE Transactions on*, 23(2): 612-622.
- Pan, H., Blu, T., & Dragotti, P. L. (2014). Sampling curves with finite rate of innovation. *Signal Processing, IEEE Transactions on*, *62*(2), 458-471.
- Shin, P. J., Larson, P. E., Ohliger, M. A., Elad, M., Pauly, J. M., Vigneron, D. B., & Lustig, M. (2014). Calibrationless parallel imaging reconstruction based on structured low-rank matrix completion. *Magnetic resonance in medicine*, 72(4), 959-970.
- Haldar, J. P. (2014). Low-Rank Modeling of Local-Space Neighborhoods (LORAKS) for Constrained MRI. *Medical Imaging, IEEE Transactions on*, 33(3), 668-681
- Jin, K. H., Lee, D., & Ye, J. C. (2015, April). A novel k-space annihilating filter method for unification between compressed sensing and parallel MRI. *In Biomedical Imaging (ISBI), 2015 IEEE 12th International Symposium on* (pp. 327-330). IEEE.
- Ongie, G., & Jacob, M. (2015). Super-resolution MRI Using Finite Rate of Innovation Curves. *Proceedings of ISBI 2015, New York, NY*.
- Ongie, G. & Jacob, M. (2015). Recovery of Piecewise Smooth Images from Few Fourier Samples. *Proceedings of SampTA 2015, Washington D.C.*
- Ongie, G. & Jacob, M. (2015). Off-the-grid Recovery of Piecewise Constant Images from Few Fourier Samples. *Arxiv.org preprint.*
- Fornasier, M., Rauhut, H., & Ward, R. (2011). Low-rank matrix recovery via iteratively reweighted least squares minimization. *SIAM Journal on Optimization,* 21(4), 1614-1640.
- Mohan, K, and Maryam F. (2012). Iterative reweighted algorithms for matrix rank minimization." *The Journal of Machine Learning Research* 13.1 3441-3473.

Acknowledgements

- Supported by grants: NSF CCF-0844812, NSF CCF-1116067, NIH 1R21HL109710-01A1, ACS RSG-11-267-01-CCE, and ONR-N000141310202.

Thank you!
Questions?

GIRAF algorithm for structured low-rank matrix recovery formulations in MRI

- Exploit convolution structure to simplify IRLS algorithm
- Do not need to explicitly form large lifted matrix
- Solves problem in *original domain*