Mend It, Don't End It: Improving GRAPPA using Simultaneous Sparsity

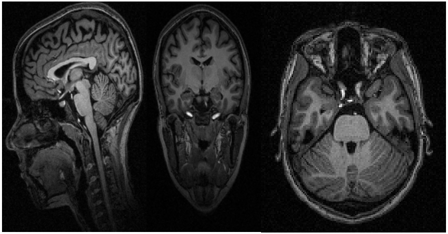
D. S. Weller[a], J. R. Polimeni[b,c], L. Grady[d], L. L. Wald[b,c], E. Adalsteinsson[a], V. K. Goyal[a]

[a] EECS, Massachusetts Institute of Technology
[b] A. A. Martinos Center, Radiology, Massachusetts General Hospital
[c] Radiology, Harvard Medical School
[d] Image Analytics and Informatics, Siemens Corporate Research

7 Sep 2011

Signal Transformation and
Information Representation Group

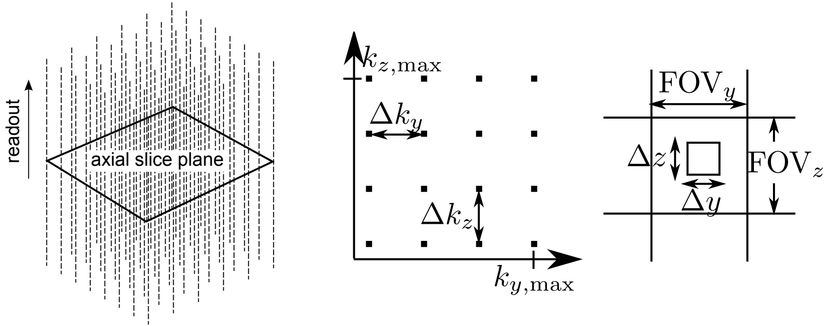
Magnetic Resonance Imaging



Martinos Center for Biomedical Imaging (MGH)

- Acquisition time too long: up to an hour for a session at 3 T
- Faster acquisitions $\Rightarrow \downarrow$ cost, $\uparrow$ comfort, $\uparrow$ quality
- Calibrate accelerated parallel imaging using sparsity
    - Poor calibration $\Rightarrow \uparrow$ noise amplification, $\uparrow$ aliasing
    - We show improved calibration de-noises and mitigates aliasing at high accelerations
- Jointly optimize fidelity to GRAPPA solution and simultaneous sparsity of the transformed coil images

Cartesian Encoding of k-Space



- Sample spatial Fourier transform domain (k-space)
- Raster scan k-space along readout direction k_x
- Transverse (axial) plane: phase-encode directions k_y, k_z
- Time proportional to extent and total # of readout lines:

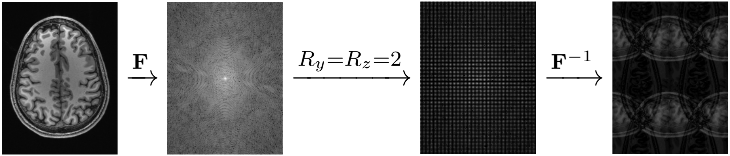
$$\text{time} \propto N_{\text{avg}} \frac{\text{FOV}_y \text{FOV}_z}{\Delta x \Delta y \Delta z} \leftarrow \text{does not depend on FOV}_x$$

Accelerated MRI

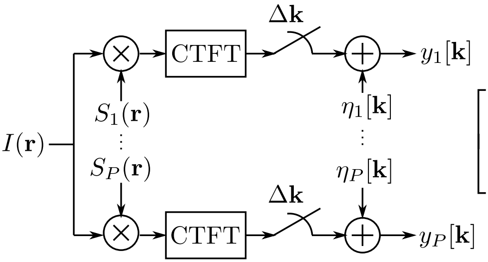
- Typically, acquisitions have SNR and spatial resolution constraints

$$\text{SNR} \propto \sqrt{N_{\text{avg}} \Delta x \Delta y \Delta z}$$

- Strategy to reduce # of readout lines required:
  - Use minimum k-space extent satisfying spatial resolution requirement Δx, Δy, Δz
  - Use fewest # of averages N_{avg} achieving desired SNR
  - Reducing FOV below object size yields aliasing in image domain
  - Use multiple receiver coils in parallel to undo aliasing [Roemer90]



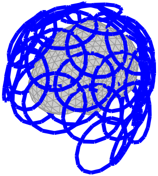
Accelerated Parallel Imaging



$$\begin{bmatrix} \eta_1[\mathbf{k}] \\ \vdots \\ \eta_P[\mathbf{k}] \end{bmatrix} \sim \text{i.i.d. } \mathcal{N}(\mathbf{0}, \mathbf{\Lambda})$$

$\mathbf{r} = [x, y, z]$ $\quad$ $\mathbf{k} = [k_x, k_y, k_z]$ $\quad$ $\Delta\mathbf{k} = [\Delta k_x, R_y \Delta k_y, R_z \Delta k_z]$

- Image $I(\mathbf{r})$ observed by P parallel coils (right)
- Receivers' spatial weightings $S_1(\mathbf{r}), \ldots, S_P(\mathbf{r})$
- k-space undersampled by factor $R_y \times R_z$
- Samples $y_1[\mathbf{k}], \ldots, y_P[\mathbf{k}]$ have complex-valued thermal noise $\eta_1[\mathbf{k}], \ldots, \eta_P[\mathbf{k}]$ with covariance $\mathbf{\Lambda}$



Accelerated Parallel Imaging

Accelerated Parallel Imaging Methods

	SENSE [Pruessmann99]	SMASH [Sodickson97]	GRAPPA [Griswold02]
reconstruction domain	image	k-space	k-space
combines coils	yes		no
uses coil sensitivities	yes		no
auto-calibration (ACS) lines	no		yes
un-aliasing quality	fair[1]	poor[2]	good
amplifies noise	yes		

[1] depends on quality of measured sensitivities

[2] un-aliasing depends on arrangement of coils

Full k-Space Recovery Using GRAPPA

GRAPPA method has two steps:

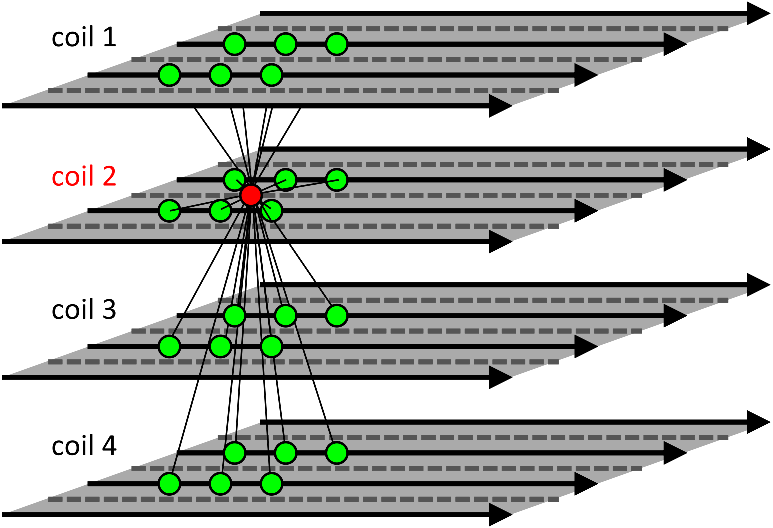
1. **Calibration**: Use least-squares fit of ACS lines to calibrate kernel weights $g_{p,q,r_y,r_z}[b_x, b_y, b_z]$ used to fill in missing k-space in each coil
   - ACS lines must be sufficiently large to fit GRAPPA kernel pattern
   - Should have at least $N_{\text{src}}P$ ACS fit equations

Full k-Space Recovery Using GRAPPA

GRAPPA method has two steps:

1. **Calibration**: Use least-squares fit of ACS lines to calibrate kernel weights $g_{p,q,r_y,r_z}[b_x, b_y, b_z]$ used to fill in missing k-space in each coil
   - ACS lines must be sufficiently large to fit GRAPPA kernel pattern
   - Should have at least $N_{\text{src}}P$ ACS fit equations
2. **Reconstruction**: Use calibrated kernel set $\mathbf{G}$ to fill in missing k-space frequencies in all the coils
   - Can be implemented as a correlation using FFT's (fast) or in the image domain (even faster)
   - Various hybrid 2-D GRAPPA calibration/reconstruction strategies for 3-D datasets discussed in [Brau08]

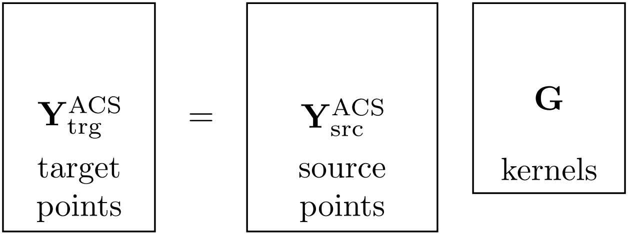
GRAPPA: Coil-by-Coil Reconstruction



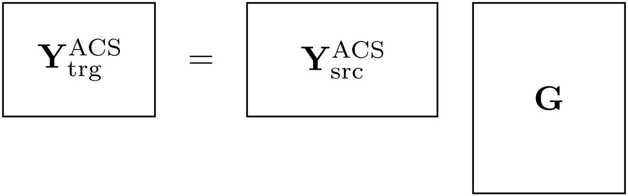
[Image from J. Polimeni, ISMRM, 2011.]

Calibrating GRAPPA Kernels

Standard case: (# of fits $\geq N_{\text{src}}P$)



Underdetermined case: (# of fits $< N_{\text{src}}P$)



Regularization of GRAPPA Kernel Calibration

Existing calibration regularization techniques:

1. Tikhonov/minimum-energy regularization: Use ℓ_2 or least-squares term alongside least-squares ACS fit objective in calibration:

$$\mathbf{G}_{\text{Tikhonov}} = \arg\min_{\mathbf{G}} \frac{1}{N_{\text{Fro}}} \|\mathbf{Y}_{\text{src}}^{\text{ACS}}\mathbf{G} - \mathbf{Y}_{\text{trg}}^{\text{ACS}}\|_F^2 + \|\alpha\mathbf{G}\|_F^2$$

   - Rescale ACS fit term with $N_{\text{Fro}} = P \cdot \min\{R_y R_z - 1, N_{\text{src}}\}$
   - Used with SENSE [Lin04]

2. GRAPPA operator-based regularization [Bydder09]: GRAPPA kernel should behave like frequency-shift operator, so application of the kernel repeated R times should yield the original data (shifted R points)

Regularization of GRAPPA Kernel Calibration

Proposed calibration regularization technique:

- Empirically, MRI images are compressible in domains like finite-differences or DWT, like natural images [Lustig07]
- Since observation noise is uncorrelated across frequencies, amplified noise is expected to remain not sparse in the GRAPPA result
- We observe that the GRAPPA-reconstructed image quality degrades significantly when there are too few ACS lines to get a high-quality kernel fit
- Promote the sparsity of the coil images that would be reconstructed by the calibrated kernel during the calibration step to mitigate noise amplification

Promoting Simultaneous Sparsity

- For a set of full-FOV k-space $\mathbf{Y}^{\text{full}}$ for each coil, each column of the $N \times P$ matrix $\mathbf{W}$ represents the sparse coefficients for the transform $\mathbf{\Psi}$ of that coil image:

$$\mathbf{W} = \mathbf{\Psi}\mathbf{F}^{-1}\mathbf{Y}^{\text{full}}$$

- Use hybrid $\ell_{1,2}$ norm for simultaneous (joint) sparsity:

$$\|\mathbf{W}\|_{1,2} = \sum_{n=1}^{N} \|[W_{n,1}, \ldots, W_{n,P}]\|_2$$

- Promote sparsity of the GRAPPA reconstruction $f(\mathbf{G}, \mathbf{Y}^{\text{acq}})$ convolving GRAPPA kernels $\mathbf{G}$ with acquired data $\mathbf{Y}^{\text{acq}}$

$$\arg\min_{\mathbf{G}} \|\mathbf{\Psi}\mathbf{F}^{-1} f(\mathbf{G}, \mathbf{Y}^{\text{acq}})\|_{1,2}$$

The Sparsity Promoting GRAPPA Kernel Fit

$$\mathbf{G}_{\text{sp}} = \arg\min_{\mathbf{G}} \frac{1}{N_{\text{Fro}}} \|\mathbf{Y}_{\text{src}}^{\text{ACS}}\mathbf{G} - \mathbf{Y}_{\text{trg}}^{\text{ACS}}\|_F^2 + \lambda \|\mathbf{\Psi}\mathbf{F}^{-1} f(\mathbf{G}, \mathbf{Y}^{\text{acq}})\|_{1,2}$$

- Combine terms in objective using tuning parameter $\lambda > 0$
- Approximately solve by iterating re-weighted least-squares problems derived using half-quadratic minimization

$$\mathbf{G}_{\text{sp}}^t = \arg\min_{\mathbf{G}} \frac{1}{N_{\text{Fro}}} \|\mathbf{Y}_{\text{src}}^{\text{ACS}}\mathbf{G} - \mathbf{Y}_{\text{trg}}^{\text{ACS}}\|_F^2 + \frac{\lambda}{2} \|(\mathbf{\Delta}^{t-1})^{0.5} \mathbf{\Psi}\mathbf{F}^{-1} f(\mathbf{G}, \mathbf{Y}^{\text{acq}})\|_F^2,$$

where $\Delta_{n,n}^{t-1} = \frac{1}{\|[W_{n,1}^{t-1}, \ldots, W_{n,P}^{t-1}, \epsilon]\|_2}$ is diagonal, $\mathbf{W}^{t-1} = \mathbf{\Psi}\mathbf{F}^{-1} f(\mathbf{G}_{\text{sp}}^{t-1}, \mathbf{Y}^{\text{acq}})$, and the GRAPPA reconstruction is an affine function of $\mathbf{G}$

See [Weller et al., *SPIE Wavelets and Sparsity XIV*, 2011]

Implementation Notes

- Solutions to these least-squares problems are obtained using LSMR [Fong10]
- Each iteration requires:
  - Computing the residual: $(R_y R_z - 1)P^2$ convolutions, P inverse FFT's, P sparsifying transforms, $2P$ matrix-vector products, 2 scalar-vector multiplications
  - Computing the estimate of the kernel using the (re-scaled) residual has similar complexity
- Updating the diagonal re-weighting matrix $\mathbf{\Delta}$ requires $(R_y R_z - 1)P^2$ convolutions, P inverse FFT's, P sparsifying transforms
- Each of these operations are highly parallelizable

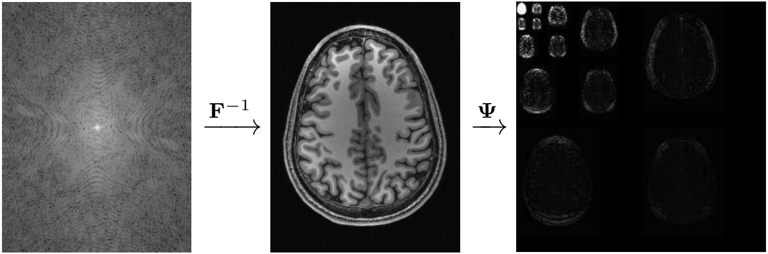
Methods

- Real data: axial slice of 3-D dataset
  - Un-accelerated ground truth (acquisition time: 8 minutes)
  - T_1-weighted MPRAGE sequence
  - $256 \times 256 \times 176$ sagittal slices; 1.0 mm isotropic voxels
  - Siemens Tim Trio 3 T (Siemens Healthcare, Erlangen, Germany) with vendor-supplied 32-channel head-coil receive array
  - Noise-only (no RF excitation) pre-scan to measure $\mathbf{\Lambda}$
  - Axial slice extracted from full dataset, cropped, and undersampled in MATLAB

Methods

- Real data: axial slice of 3-D dataset
  - Un-accelerated ground truth (acquisition time: 8 minutes)
  - T_1-weighted MPRAGE sequence
  - $256 \times 256 \times 176$ sagittal slices; 1.0 mm isotropic voxels
  - Siemens Tim Trio 3 T (Siemens Healthcare, Erlangen, Germany) with vendor-supplied 32-channel head-coil receive array
  - Noise-only (no RF excitation) pre-scan to measure $\mathbf{\Lambda}$
  - Axial slice extracted from full dataset, cropped, and undersampled in MATLAB
- Comparison for kernel calibration with either no regularization or Tikhonov or sparsity-promoting regularization
  - Compared magnitude images against fully-sampled images using difference images
  - Quantitative comparisons using PSNR

Methods

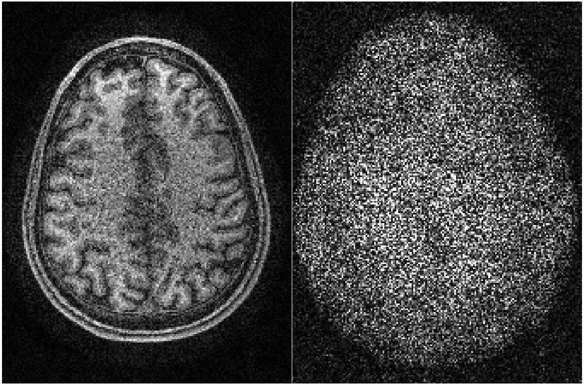


- Ground truth (k-space, image, 4-level ‘9-7’ DWT)
- Uniformly undersample k-space by 4 in each direction ($R_y = R_z = 4$)
- Sample center ACS block (effective acceleration $< R_y R_z$)
- Use ACS lines for kernel calibration and as data during reconstruction

Regularization Compared for Low-Quality ACS Fit

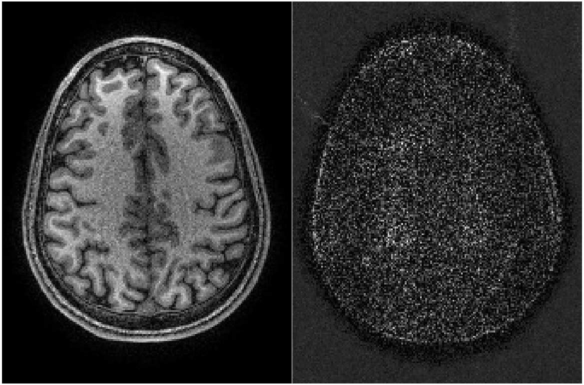
Experiment Parameters	
Uniform undersampling:	$R_y = R_z = 4$
Kernel size:	$B_y = B_z = 4$
# of ACS lines:	$N_{\text{ACS},k_y} = N_{\text{ACS},k_z} = 36$
Size of $\mathbf{Y}_{\text{src}}^{\text{ACS}}$:	576 fits $\times$ 512 source points
$\text{cond}(\mathbf{Y}_{\text{src}}^{\text{ACS}})$:	4192.63
Effective acceleration:	$R_{\text{eff}} = 10.5$

Regularization Compared for Low-Quality ACS Fit



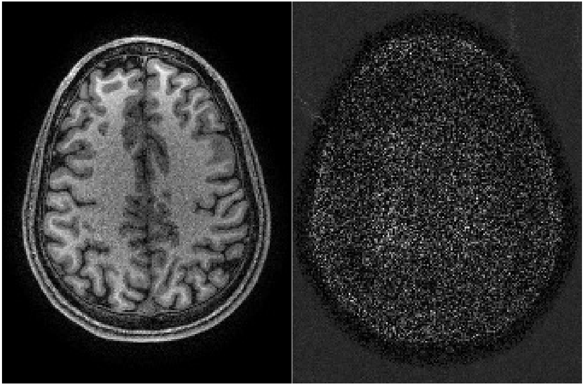
GRAPPA w/ no regularization
PSNR $= 22.2$ dB

Regularization Compared for Low-Quality ACS Fit



GRAPPA w/ Tikhonov regularization ($\alpha^2 = 10^{-1.2}$)
PSNR $= 28.2$ dB

Regularization Compared for Low-Quality ACS Fit



GRAPPA w/ sparsity ($\lambda = 1$)
PSNR $= 28.4$ dB

Regularization Compared for Underdetermined Fit

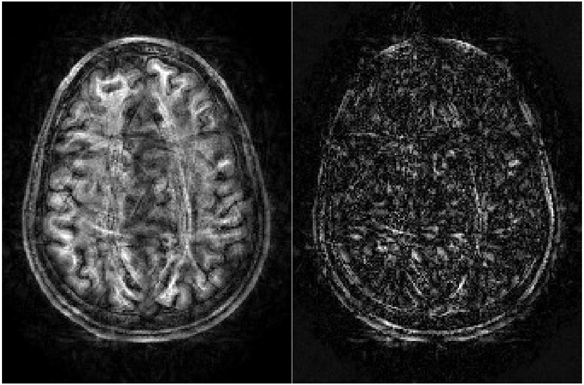
Experiment Parameters	
Uniform undersampling:	$R_y = R_z = 4$
Kernel size:	$B_y = B_z = 4$
# of ACS lines:	$N_{\text{ACS},k_y} = N_{\text{ACS},k_z} = 20$
Size of $\mathbf{Y}_{\text{src}}^{\text{ACS}}$:	64 fits $\times$ 512 source points
$\text{cond}(\mathbf{Y}_{\text{src}}^{\text{ACS}})$:	N/A
Effective acceleration:	$R_{\text{eff}} = 13.7$

Regularization Compared for Underdetermined Fit

Experiment Parameters	
Uniform undersampling:	$R_y = R_z = 4$
Kernel size:	$B_y = B_z = 4$
# of ACS lines:	$N_{\text{ACS},k_y} = N_{\text{ACS},k_z} = 20$
Size of $\mathbf{Y}_{\text{src}}^{\text{ACS}}$:	64 fits $\times$ 512 source points
$\text{cond}(\mathbf{Y}_{\text{src}}^{\text{ACS}})$:	N/A
Effective acceleration:	$R_{\text{eff}} = 13.7$

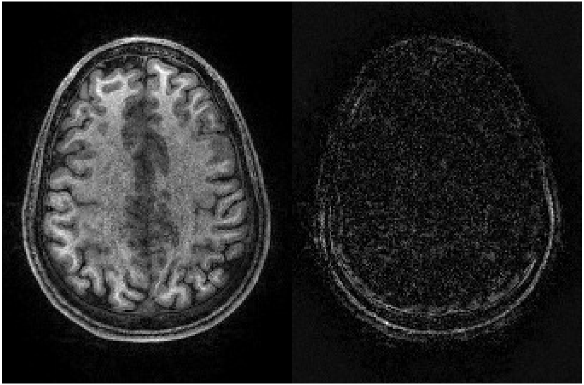
GRAPPA kernel calibration requires regularization in underdetermined case!

Regularization Compared for Underdetermined Fit



GRAPPA w/ Tikhonov regularization ($\alpha^2 = 10^{0.2}$)
PSNR $= 21.1$ dB

Regularization Compared for Underdetermined Fit



GRAPPA w/ sparsity ($\lambda = 10^{-0.8}$)
PSNR $= 25.2$ dB

Discussion

Three operating regimes:

1. # of fits $\gg B_x B_y B_z P$: (not shown)
   - regularization only marginally beneficial
2. # of fits $\sim B_x B_y B_z P$: (first experiment)
   - both types of regularization effectively de-noise the result
   - sparsity-promoting regularization avoids (minor) residual aliasing visible in Tikhonov-regularized result
3. # of fits $< B_x B_y B_z P$: (second experiment)
   - both types of regularization effectively de-noise the result
   - sparsity-promoting regularization removes aliasing whereas Tikhonov-regularization does not

Discussion

- Trade-off: effective acceleration vs. reconstruction error by varying # of ACS lines
  - All methods flat at low effective acceleration (little benefit from high # of ACS lines)
  - Proposed method greatly improves reconstruction at high effective acceleration

- Applications:
  - Acquiring ACS lines is expensive, e.g. MR spectroscopy
  - Maximum acceleration is essential, e.g. echo volumar imaging (EVI)

Future Extensions

- Proposed method leverages uniform Cartesian subsampling
  - Fast non-iterative GRAPPA with one set of kernels does not exist for nonuniform subsampled data
- Extension to nonuniform Cartesian subsampling:
  - Alternate calibration and reconstruction steps
  - Perform calibration with current estimates for uniformly spaced subsampled k-space data
  - Perform reconstruction with current calibrated kernel weights and find full k-space consistent with both kernel and acquired data
- Extend sparsity term from fixed sparsifying transform to learned dictionaries
- Combine with sparsity-promoting post-processing method like SpRING [Weller11]

SpRING: Post-processing GRAPPA with Sparsity

Balance fidelity to the GRAPPA solution ($\mathbf{G}(\mathbf{d})$) and sparsity of the solution while preserving observed data $\mathbf{d}$:

$$\hat{\mathbf{y}} \in \arg\min_{\mathbf{y}} \overbrace{\|\mathbf{C}\mathbf{F}^{-1}(\mathbf{y} - \mathbf{G}(\mathbf{d}))\|_2^2}^{\text{GRAPPA fidelity}} + \lambda \overbrace{\|\mathbf{\Psi}\mathbf{F}^{-1}\mathbf{y}\|_{1,2}}^{\text{joint sparsity}} \text{ s.t. } \overbrace{\mathbf{d} = \mathbf{K}\mathbf{y}}^{\text{preserve data}}$$

- Weight GRAPPA fidelity in image domain using coil combination weights $\mathbf{C}$ determined from the ACS data
- Wavelet domain simultaneous sparsity: $\mathbf{\Psi}$ is the DWT, and

$$\|\mathbf{W}\|_{1,2} = \sum_{n=1}^{N} \|[W_{n,1}, \ldots, W_{n,P}]\|_2$$

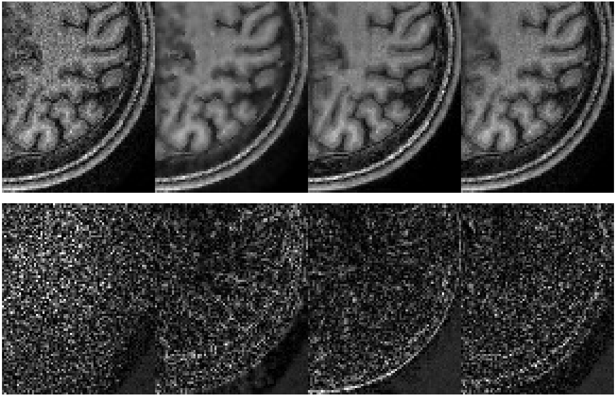
- Tuning parameter λ balances GRAPPA fidelty, sparsity
- Operate in the nullspace of observation matrix $\mathbf{K}$

See [Weller et al., *ISMRM*, 2011; Weller et al., 2011 (preprint)]

PSNR Performance of SpRING

Experiment Parameters	
Uniform undersampling:	$R_y = R_z = 4$
Kernel size:	$B_y = B_z = 3$
# of ACS lines:	$N_{\text{ACS},k_y} = N_{\text{ACS},k_z} = 36$
Size of $\mathbf{Y}_{\text{src}}^{\text{ACS}}$:	784 fits $\times$ 288 source points
$\text{cond}(\mathbf{Y}_{\text{src}}^{\text{ACS}})$:	566.83
Effective acceleration:	$R_{\text{eff}} = 10.5$

PSNR Performance of SpRING



	GRAPPA	GRAPPA + Wiener	L_1 SPIR-iT $\lambda = 10^{-2.6}$	SpRING $\lambda = 10^{0.4}$
PSNR:	25.9 dB	28.1 dB	28.3 dB	28.2 dB

GRAPPA, GRAPPA with Wiener-filter denoising, L_1 SPIR-iT, and SpRING reconstructions, difference images, and PSNRs

Acknowledgments





- Keith Heberlein at Siemens Healthcare for discussion on Tikhonov regularization for GRAPPA
- Funding: NSF CAREER 0643836, NIH R01 EB007942 and EB006847, NIH NCRR P41 RR014075, Siemens Healthcare, and an NSF Graduate Research Fellowship