Image Reconstruction

Parallel Imaging

M229 Advanced Topics in MRI

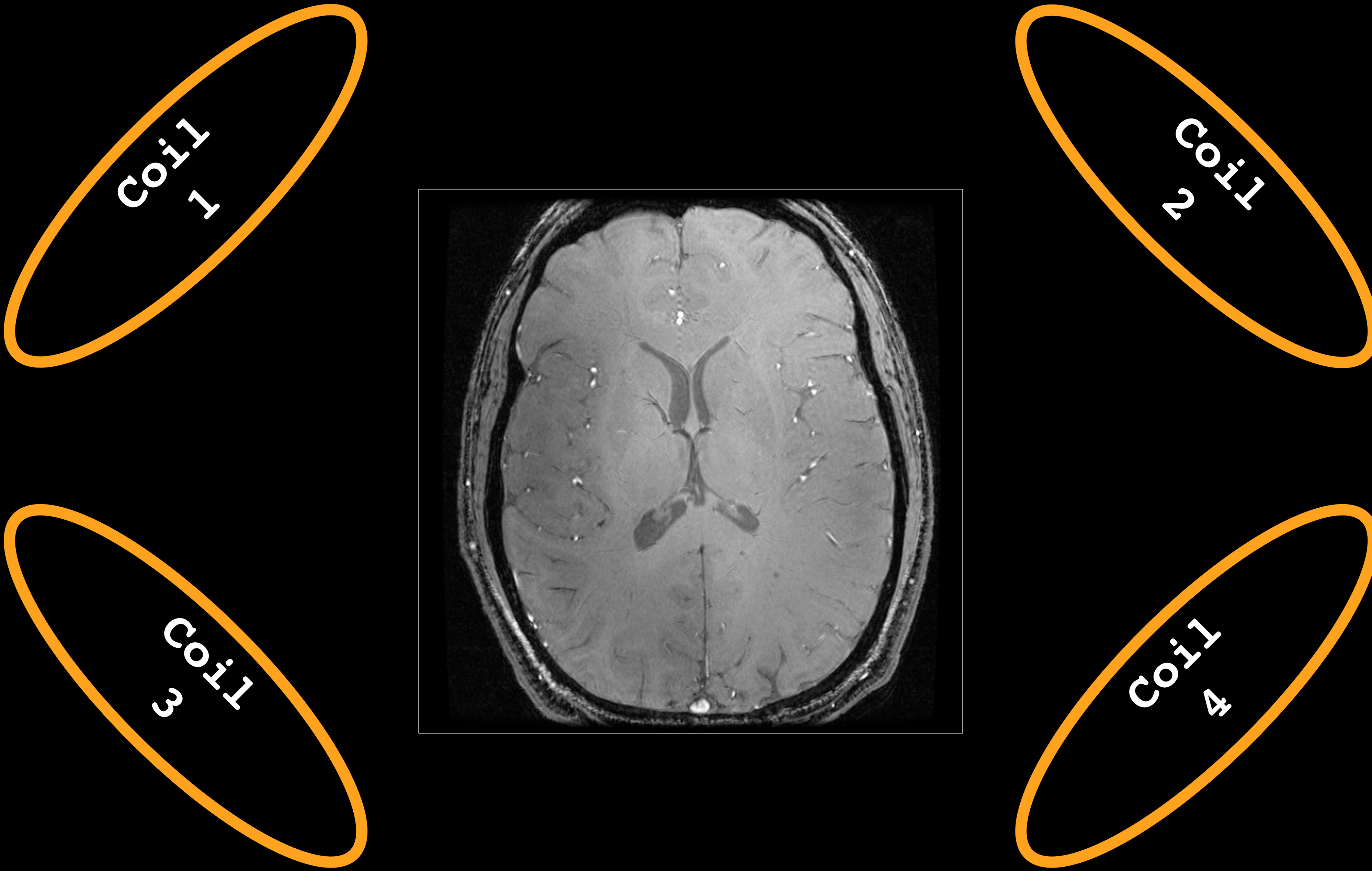
Kyung Sung, Ph.D.

5/17/2022

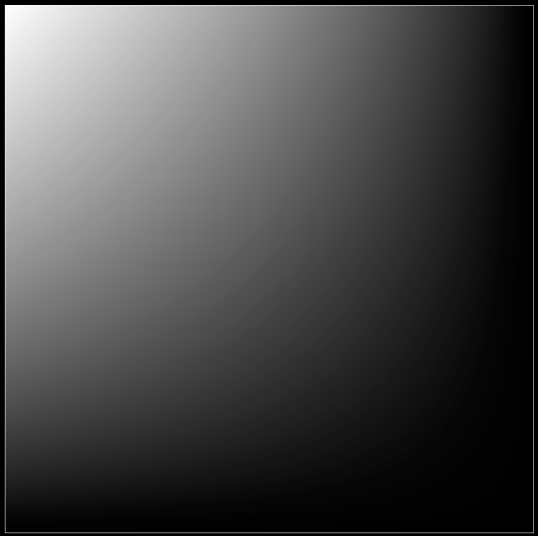
Today’s Topics

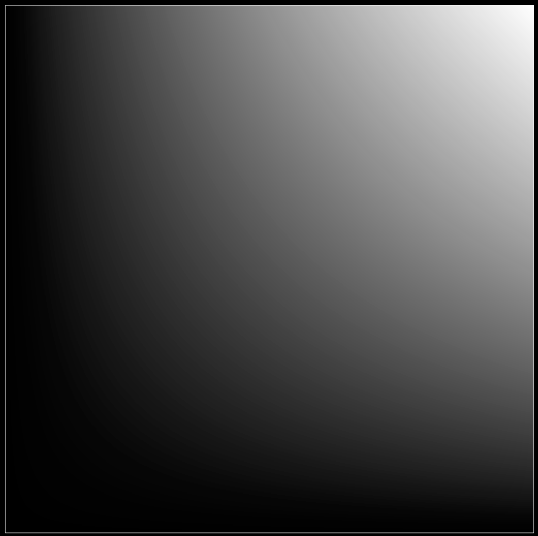
- Multicoil reconstruction
- Parallel imaging
  - Image domain methods:
    - SENSE
  - k-space domain methods:
    - SMASH
    - GRAPPA (next time)

Multi-coil Arrays

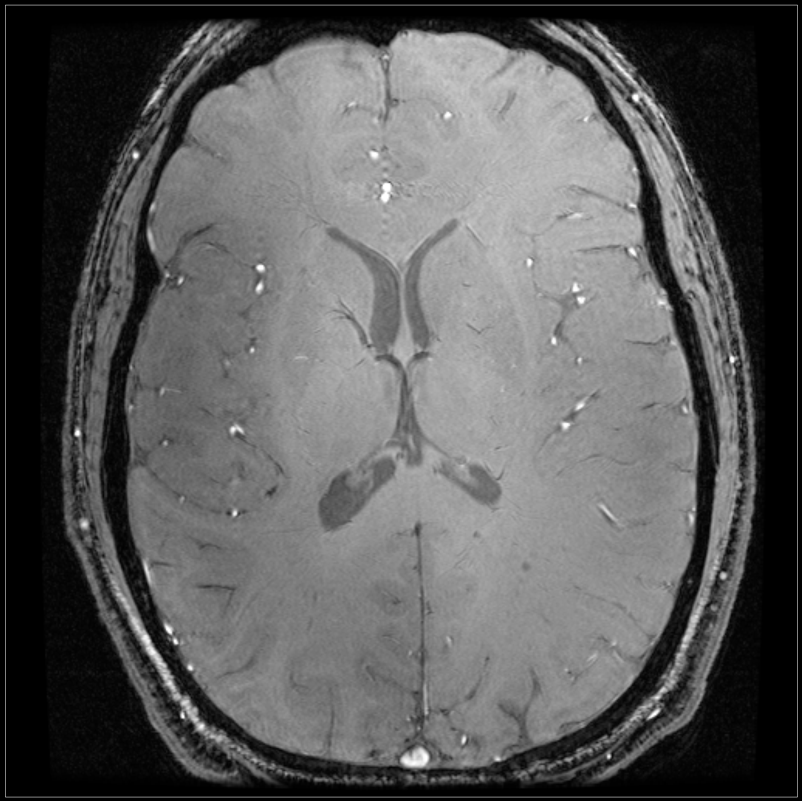


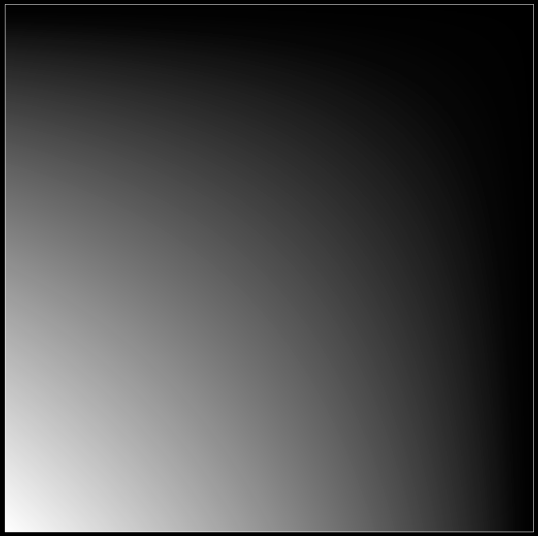
Multi-coil Sensitivity

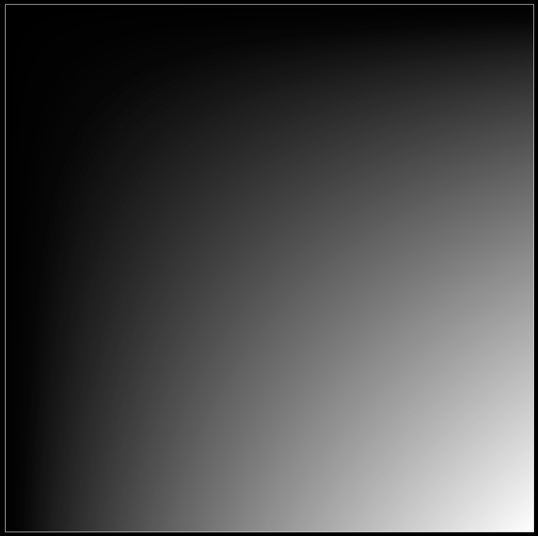




$$\|\vec{B}\,(\vec{r})\,\|$$







Multi-coil Reconstruction

- Each coil has a complete image of whole FOV and an amplitude and phase sensitivity

$$C_l(\vec{x}) \qquad l = 1,2,\ldots L$$

- Coils are coupled, so noise is correlated

$$E[n_i n_j] = \Psi$$

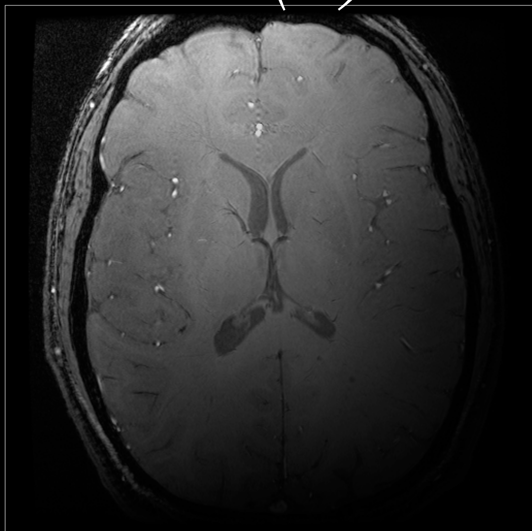
- Received data from coil l:

$$m_l(\vec{x}) = C_l(\vec{x})m(\vec{x}) + n_l(\vec{x})$$

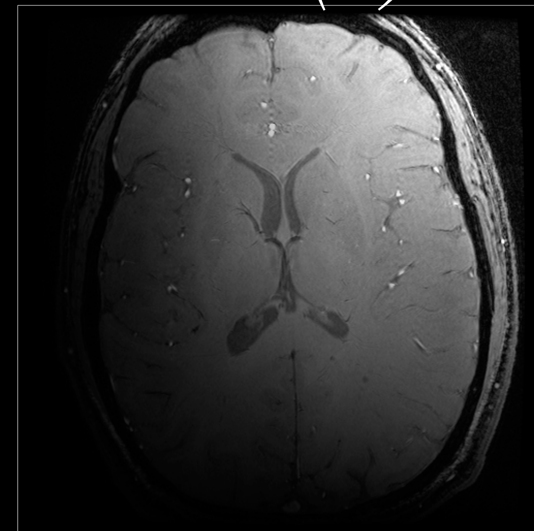
- Given $m_l(x)$, how do we reconstruct $m(x)$?

Multi-coil Images

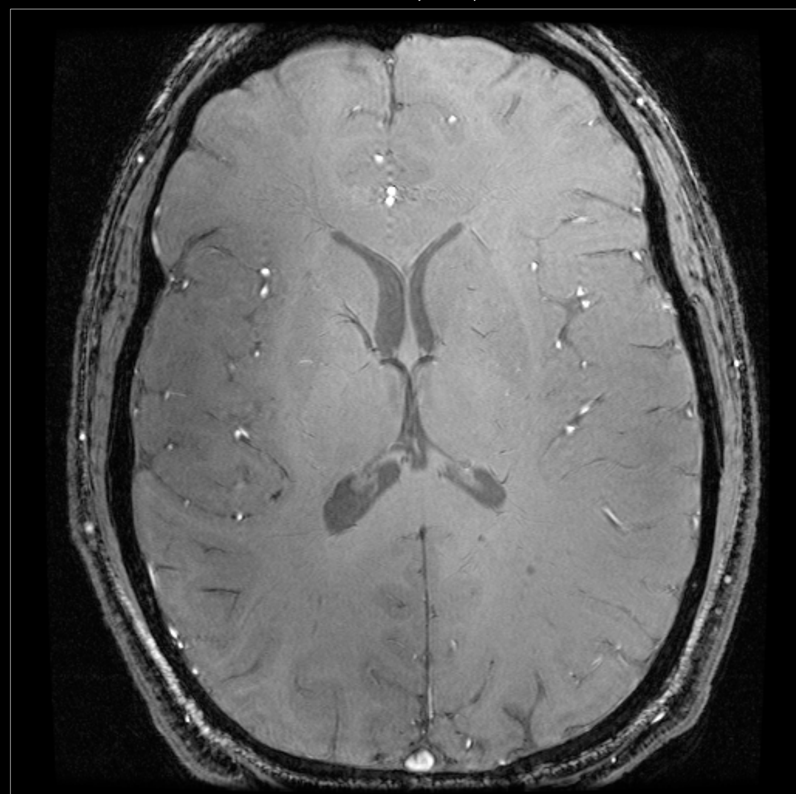
$m_1(x)$



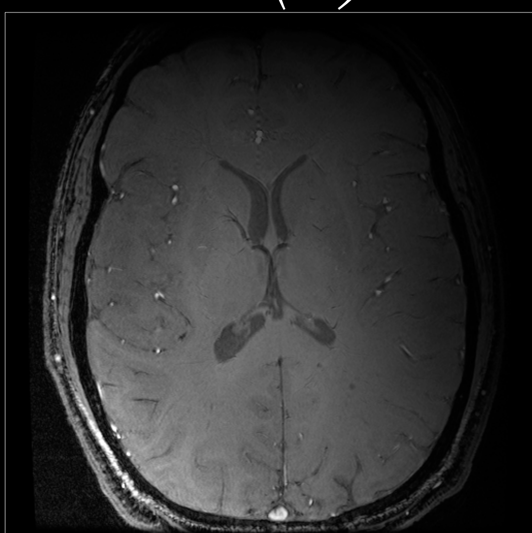
$m_2(x)$



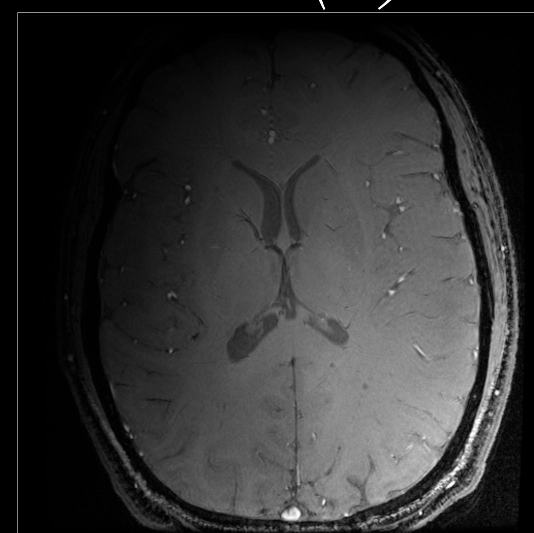
$m_s(x)$



$m_3(x)$



$m_4(x)$



Multi-coil Reconstruction

For a particular voxel x

$$\begin{pmatrix} m_1(\vec{x}) \\ m_2(\vec{x}) \\ \cdot \\ \cdot \\ \cdot \\ m_L(\vec{x}) \end{pmatrix} = \begin{pmatrix} C_1(\vec{x}) \\ C_2(\vec{x}) \\ \cdot \\ \cdot \\ \cdot \\ C_L(\vec{x}) \end{pmatrix} m(\vec{x}) + \begin{pmatrix} n_1(\vec{x}) \\ n_2(\vec{x}) \\ \cdot \\ \cdot \\ \cdot \\ n_L(\vec{x}) \end{pmatrix}$$

OR

$$\underline{m_s(\vec{x})} = \underline{C} m(\vec{x}) + \underline{n}$$

L x 1   L x 1   L x 1

Minimum Variance Estimate

$$\hat{m}(\vec{x}) = \underbrace{(C^*\Psi^{-1}C)^{-1}}_{1 \times 1}\underbrace{C^*\Psi^{-1}}_{1 \times L}\underbrace{m_s(\vec{x})}_{L \times 1}$$

Covariance (variance)

$$COV(\hat{m}(\vec{x})) = C^*\Psi^{-1}C$$

What if Ψ is $\sigma^2 I$?

$$\hat{m}(\vec{x}) = \underbrace{(C^*C)^{-1}}_{\text{Intensity Correction}}\underbrace{C^*}_{\text{Phase Correction}} m_s(\vec{x})$$

Approximate Solution

- Often we don't know $C_l(x)$, but

$$m_l(\vec{x}) = C_l(\vec{x})m(\vec{x})$$

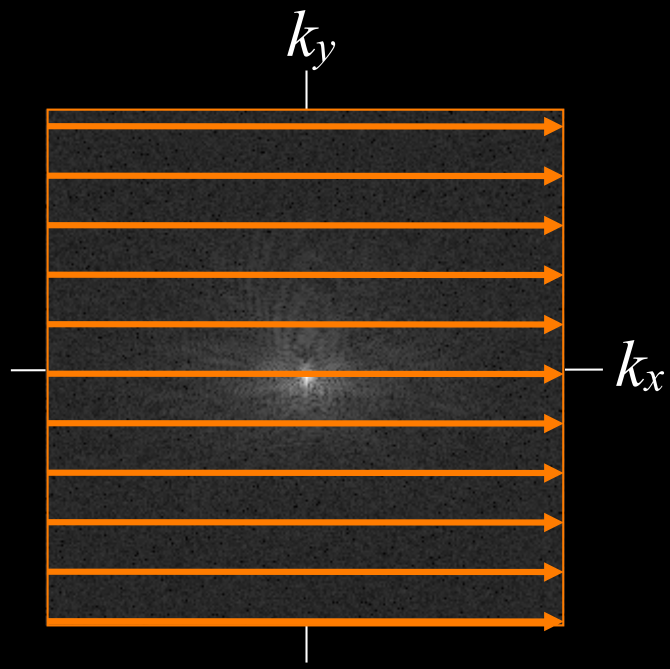
- Approximate solution:

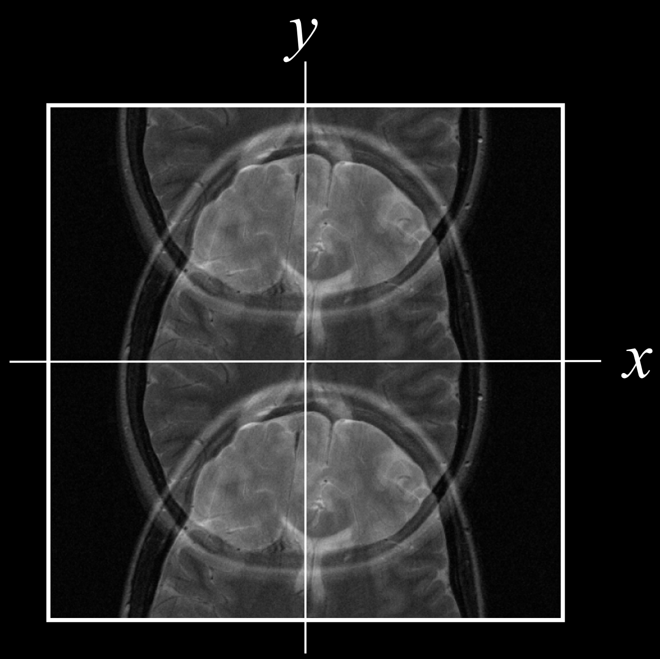
$$\hat{m}_{SS}(\vec{x}) = \sqrt{\sum_l m_l^*(\vec{x})m_l(\vec{x})}$$

- For SNR > 20, within 10% of optimal solution

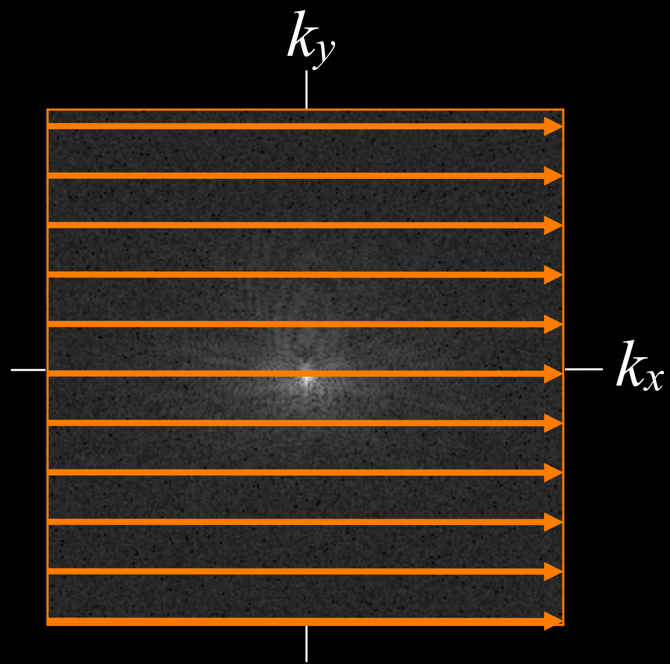
PB Roemer et al. MRM 1990

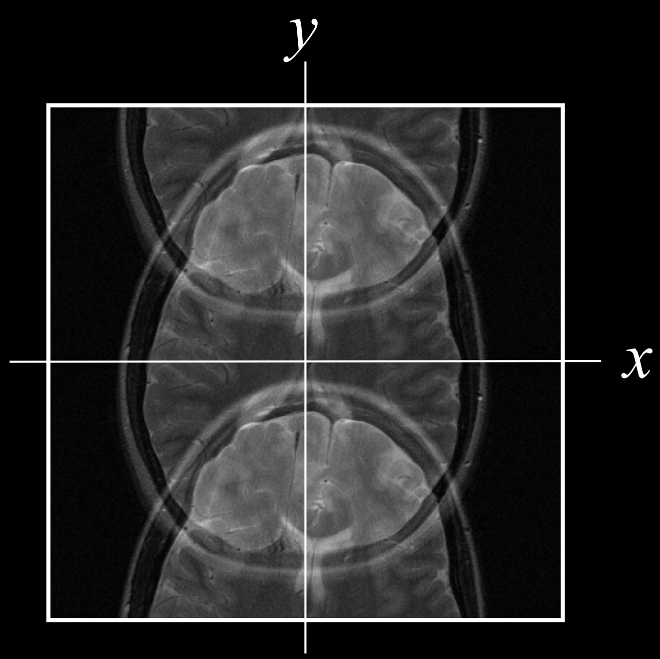
Accelerate Imaging with Array Coils





Accelerate Imaging with Array Coils





- Parallel Imaging
  - Coil elements provide some localization
  - Undersample in k-space, producing aliasing
  - Sort out in reconstruction

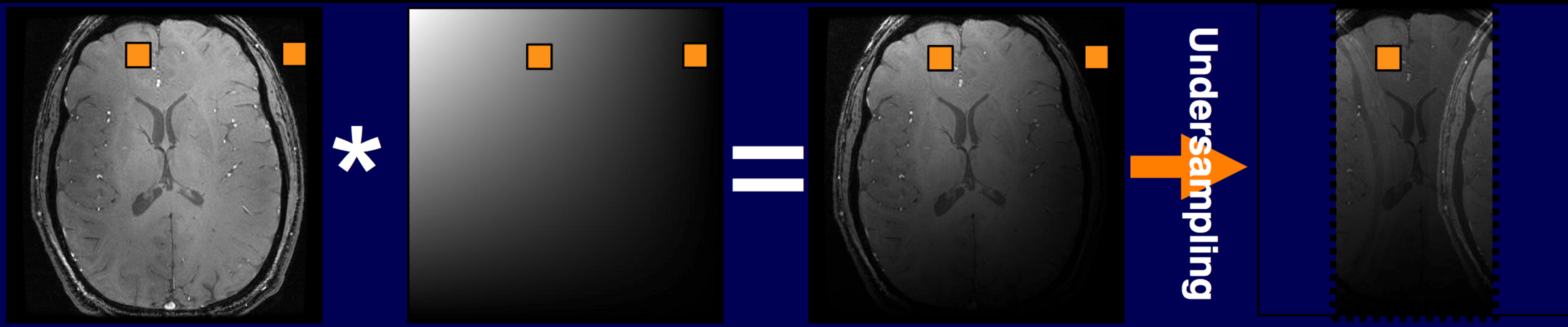
Parallel Imaging

- Many approaches:
  - Image domain - SENSE
  - k-space domain - SMASH, GRAPPA
  - Hybrid - ARC
- We will focus on two:
  - SENSE: optimal if you know coil sensitivities
  - GRAPPA: autocalibrating / robust

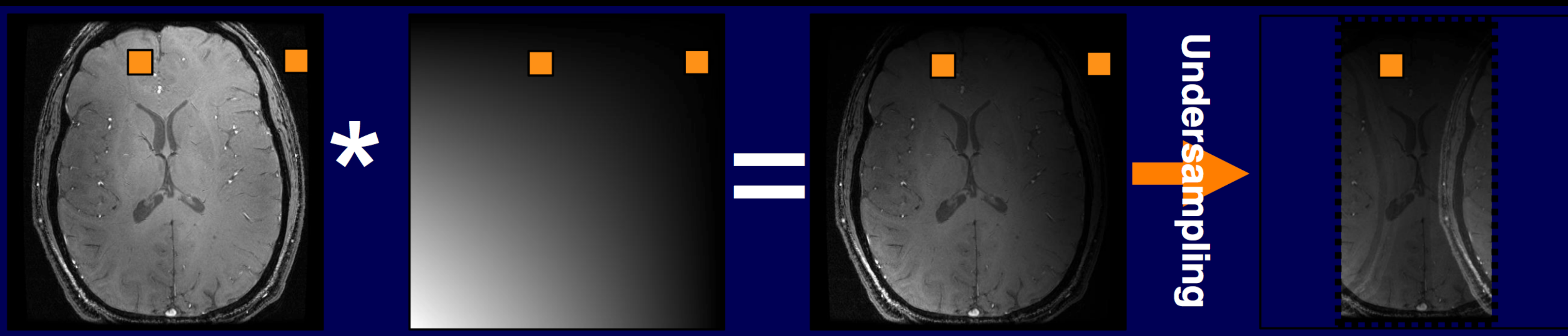
Parallel Imaging
(SENSE)

Cartesian SENSE

$$m_1(\vec{x_1}) = C_1(\vec{x_1})m(\vec{x_1}) + C_1(\vec{x_2})m(\vec{x_2})$$



$$m_2(\vec{x_1}) = C_2(\vec{x_1})m(\vec{x_1}) + C_2(\vec{x_2})m(\vec{x_2})$$



$$\begin{pmatrix} m_1(\vec{x_1}) \\ m_2(\vec{x_1}) \\ . \\ . \\ . \\ m_L(\vec{x_1}) \end{pmatrix} = \begin{pmatrix} C_1(\vec{x_1}) & C_1(\vec{x_2}) \\ C_2(\vec{x_1}) & C_2(\vec{x_2}) \\ . & \\ . & \\ . & \\ C_L(\vec{x_1}) & C_L(\vec{x_2}) \end{pmatrix} \begin{pmatrix} m(\vec{x_1}) \\ m(\vec{x_2}) \end{pmatrix} + \begin{pmatrix} n_1(\vec{x_1}) \\ n_2(\vec{x_1}) \\ . \\ . \\ . \\ n_L(\vec{x_1}) \end{pmatrix}$$

Aliased Images

Sensitivity at Source Voxels

Source Voxels

OR

2 x 1

$$m_s = Cm + n$$

L x 1 L x 2 L x 1

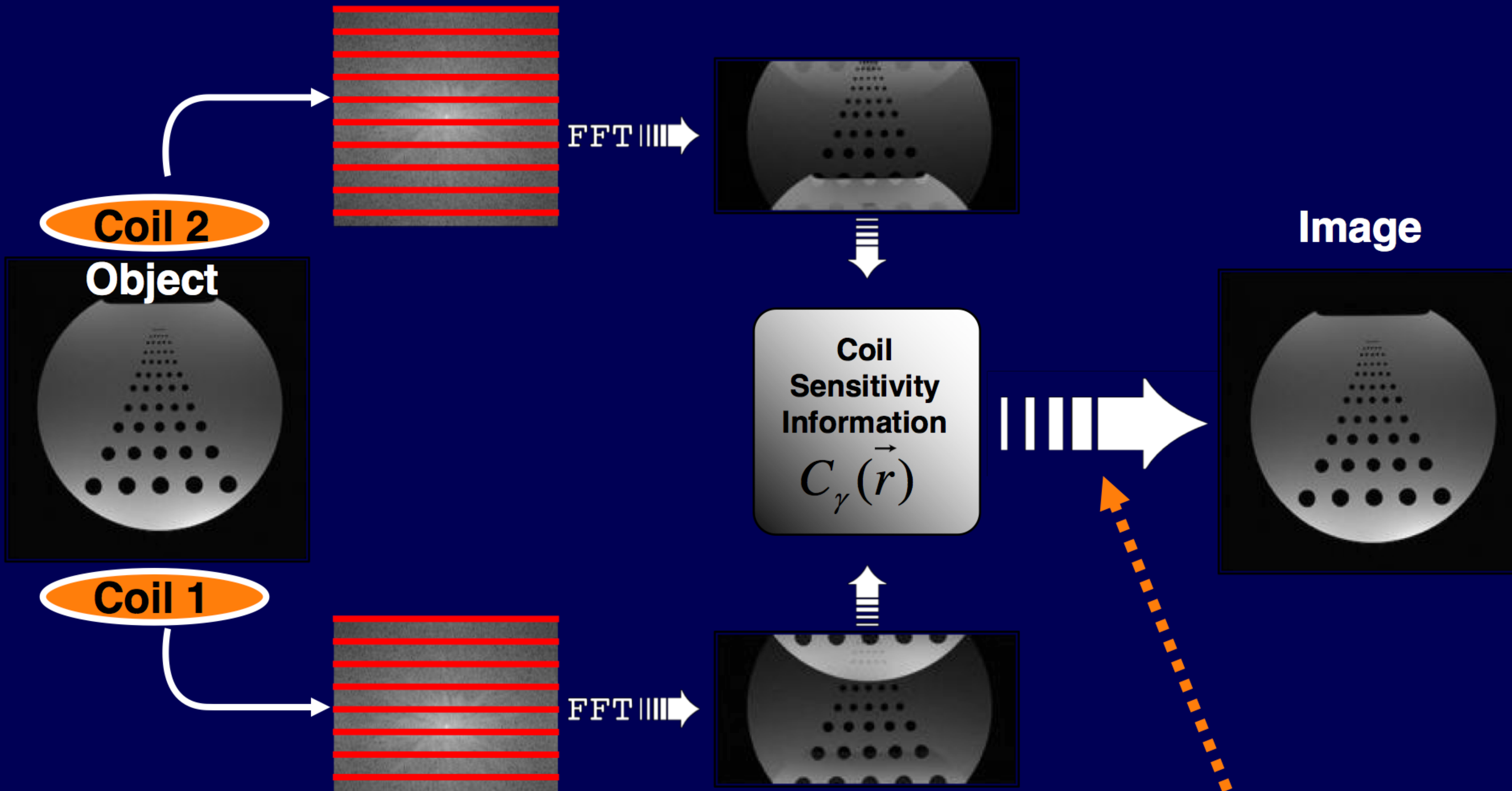
$$\hat{m}(\vec{x}) = \underbrace{(C^* \Psi^{-1} C)^{-1}}_{2 \times 2} \underbrace{C^* \Psi^{-1}}_{2 \times L} \underbrace{m_s(\vec{x})}_{L \times 1}$$

L aliased reconstruction resolves 2 image pixels

For an N x N image, we solve (N/2 x N) 2 x 2 inverse systems

For an acceleration factor R, we solve (N/R x N) R x R inverse systems

SENSE Reconstruction



$$\hat{m}(\vec{x}) = (C^*\Psi^{-1}C)^{-1}C^*\Psi^{-1}m_s(\vec{x})$$

Unwrap fold over in image space

SNR Cost

- How large can R be?
- Two SNR loss mechanisms
  - Reduced scan time
  - Condition of the SENSE decomposition
- SNR Loss

$$SNR_{SENSE} = \frac{SNR}{\underbrace{g}_{\text{Geometry Factor}}\underbrace{\sqrt{R}}_{\text{Reduced Scan Time}}}$$

Geometry Factor

- Covariance for a fully sampled image (variance of one voxel):

$$\chi_F = \frac{1}{n_F}(C_F^* \Psi^{-1} C_F)^{-1}$$

- Covariance for a reduced encoded image:

$$\chi_R = \frac{1}{n_R}(C_R^* \Psi^{-1} C_R)^{-1}$$

[redacted]

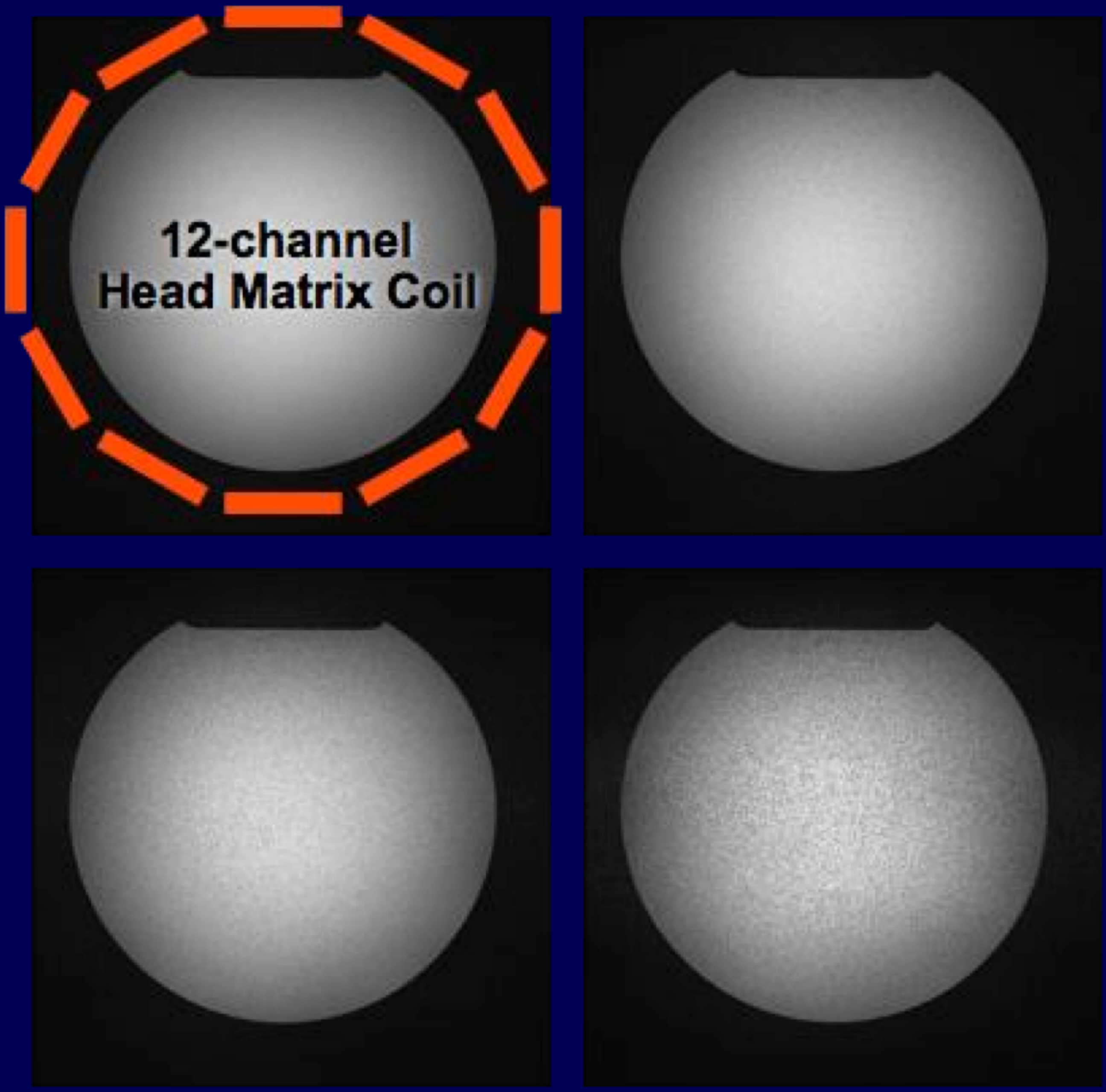
Geometry Factor

- g-factor is critical since it depends on:
  - Acceleration
  - Spatial position
  - Aliasing direction
  - Coil geometry
- Minimizing g-factor drives system design
- Sense coils are different from traditional array coils

[redacted]

Parallel Imaging Tradeoffs

ref



PAT x 2

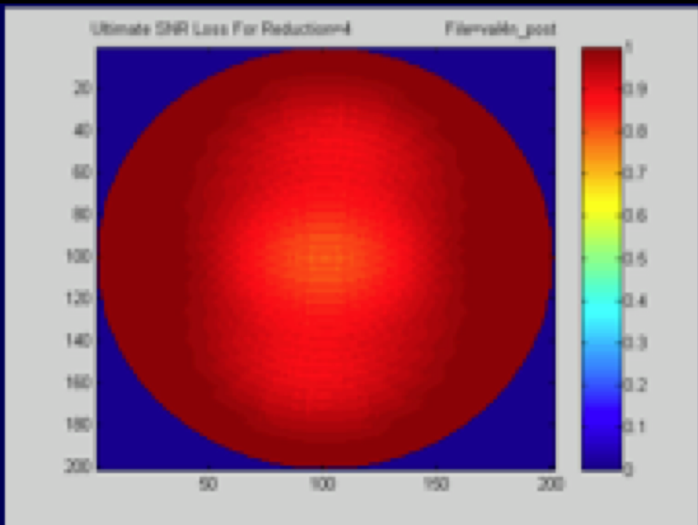
f_p = acceleration factor

g = coil geometry factor

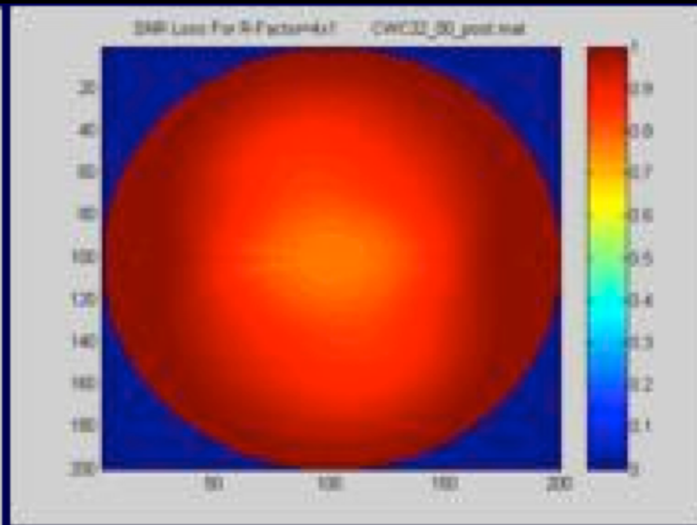
PAT x 3

PAT x 4

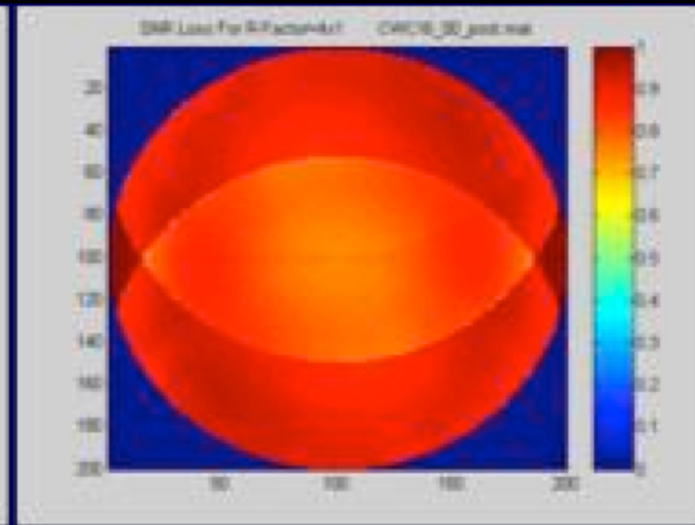
1/g-factor Map for R=4



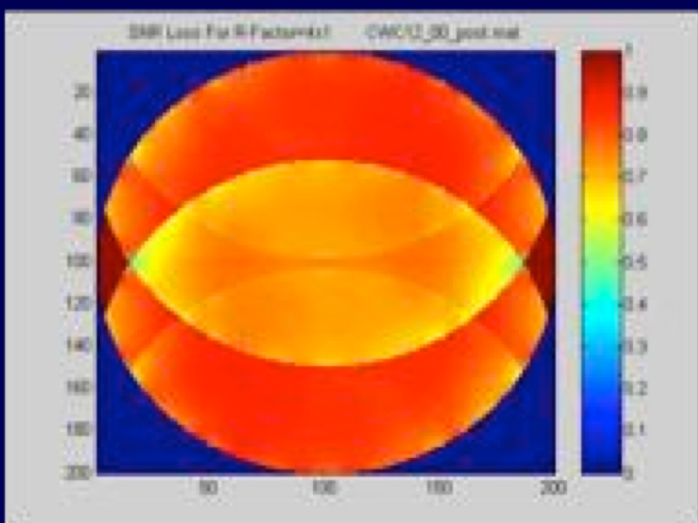
∞ elements



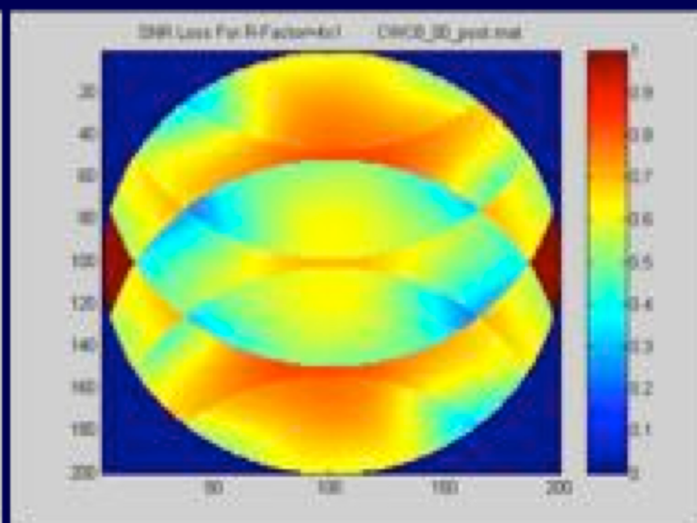
32 elements



16 elements



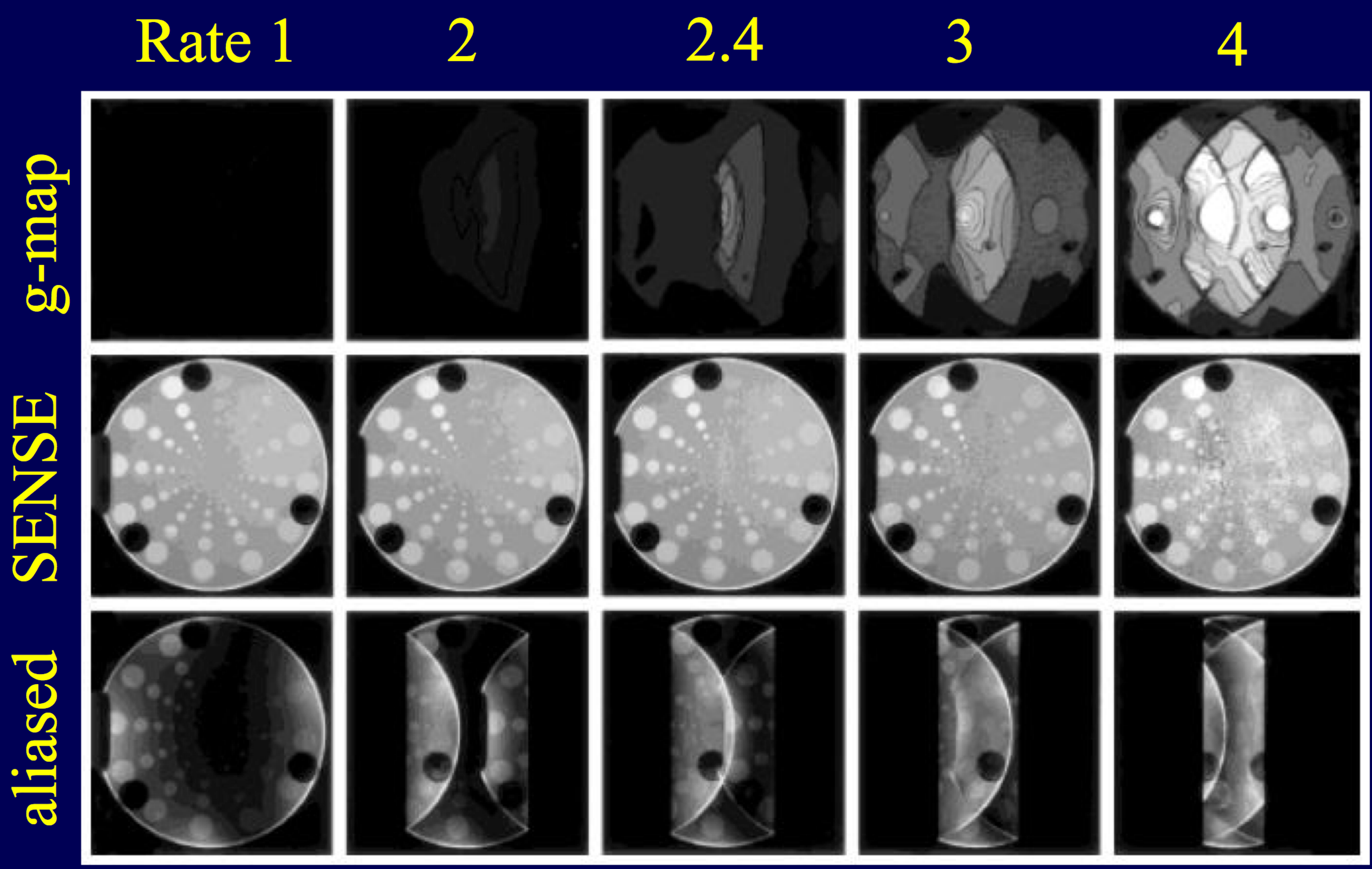
12 elements



8 elements

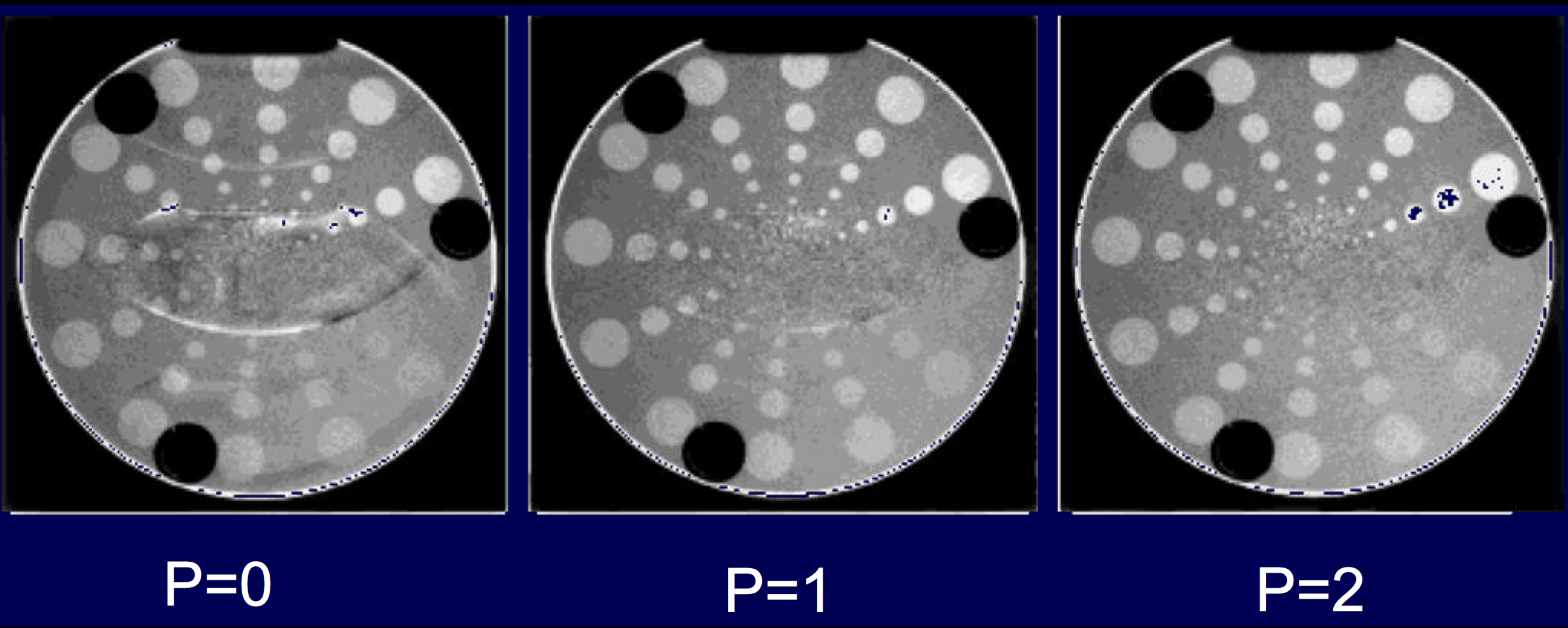
Relative SNR Scale

g-factor and its impact on images



Dependence on Coil Sensitivity

- Images reconstructed using coil sensitivity maps with different order P of polynomial fitting



Pruessmann et al. MRM 1999

Parallel Imaging
(SMASH)

SMASH

- Simultaneous Acquisition of Spatial Harmonics (SMASH) uses linear combinations of acquired k-space data from multiple coils to generate multiple data sets with offsets in k-space

Phase Encoding by Amplitude Modulation

- Signal Equation:

$$\hat{m}_j(k_x, k_y) = \int_y \int_x C_j(x,y) m(x,y) \exp^{-i2\pi(k_x \cdot x + k_y \cdot y)} \, dx dy$$

m(x,y) = image

$C_j(x,y)$ = j^{th} coil sensitivity

Phase Encoding by Amplitude Modulation

$$\hat{m}_j(k_y) = \int_y C_j(y) m(y) \exp^{-i2\pi(k_y \cdot y)} \, dy$$

- Use the arrangement of coils to construct sinusoidal sensitivity profiles
  - Sensitivity profiles can be a combination of multiple coils $$\sum_{j=0}^{L-1} a_{j,m} C_j(y)$$

Phase Encoding by Amplitude Modulation

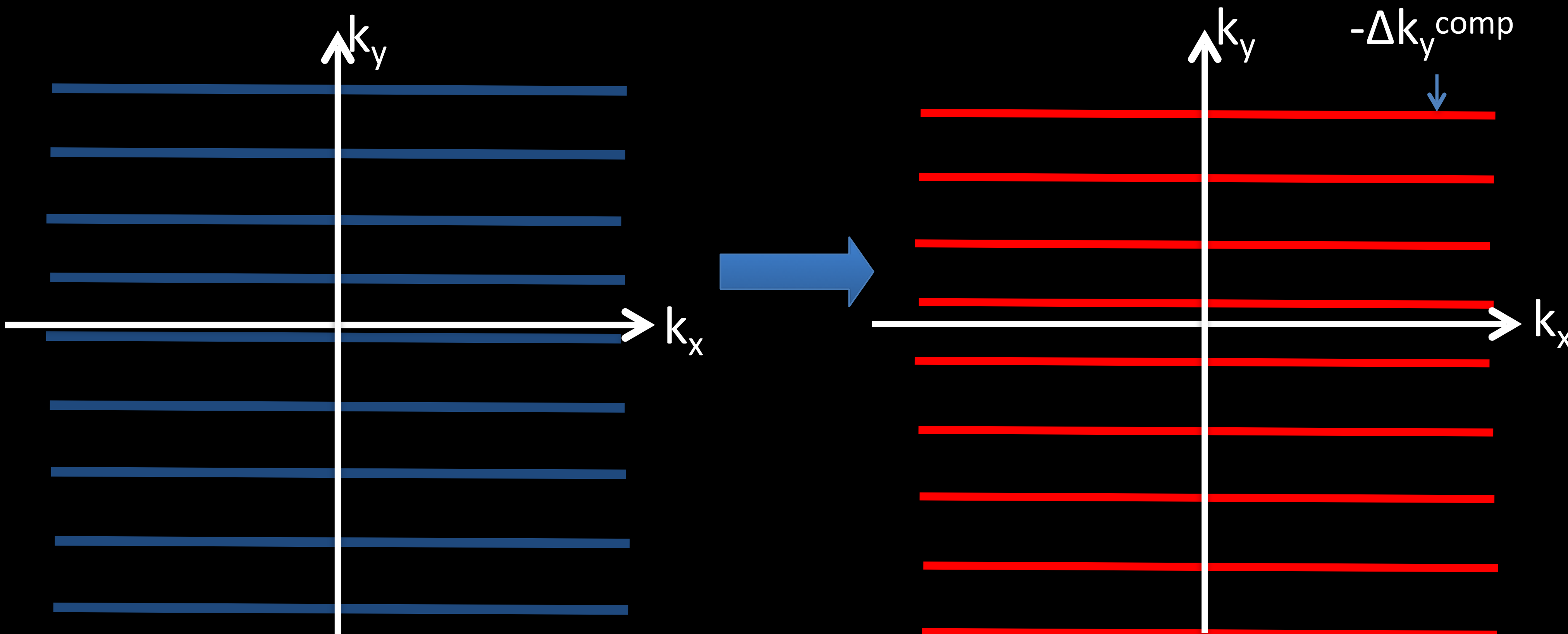
- Sensitivity profiles are combination of multiple coils, whose signals are combined to produce the desired sinusoidal sensitivity

$$C^{comp}(y) = \cos(\Delta k_y^{comp} y) + i \sin(\Delta k_y^{comp} y)$$
$$= e^{i\Delta k_y^{comp} y}$$

The wavelength could be $\lambda = 2\pi/\Delta k_y = FOV$

$C(x,y) \approx 1$

$C^{comp}(x,y) = \exp(i\Delta k_y^{comp} y)$



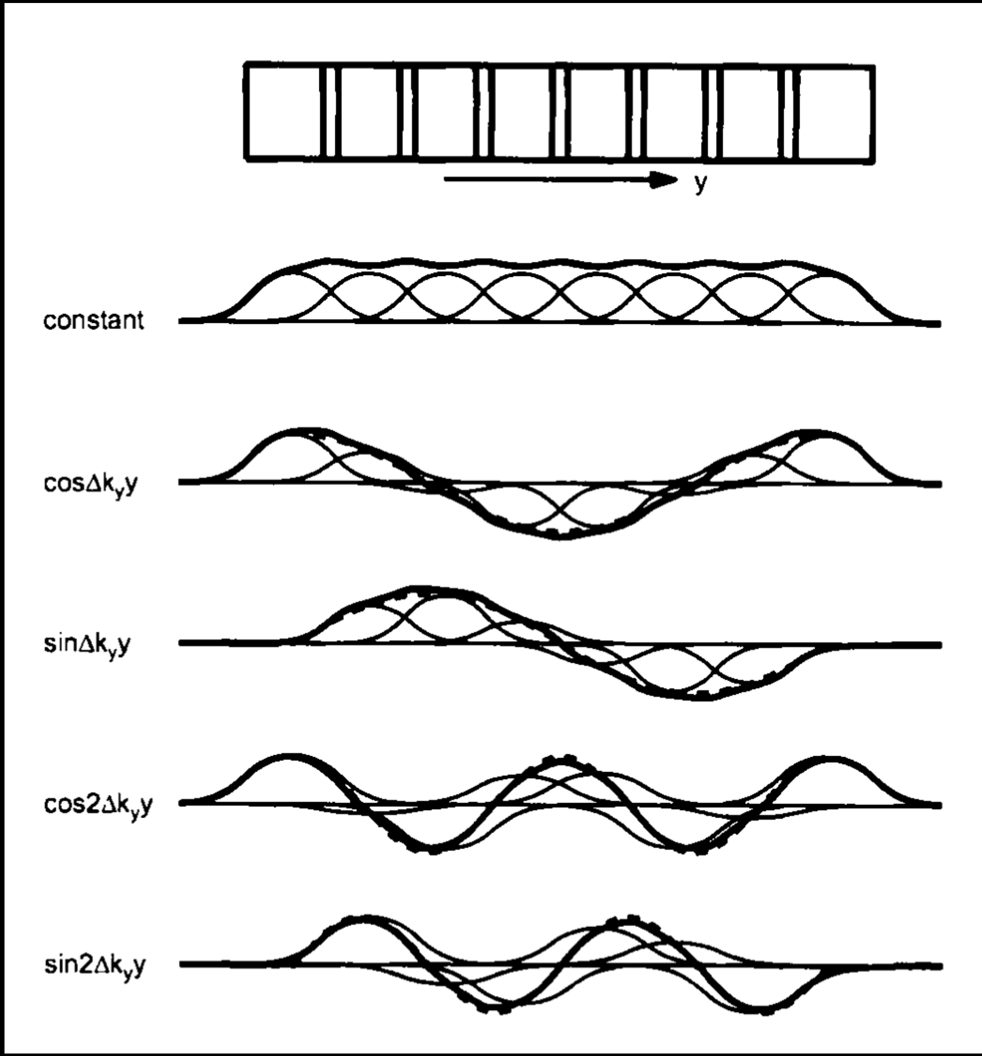
Spatial Harmonic Generation Using Coil Arrays

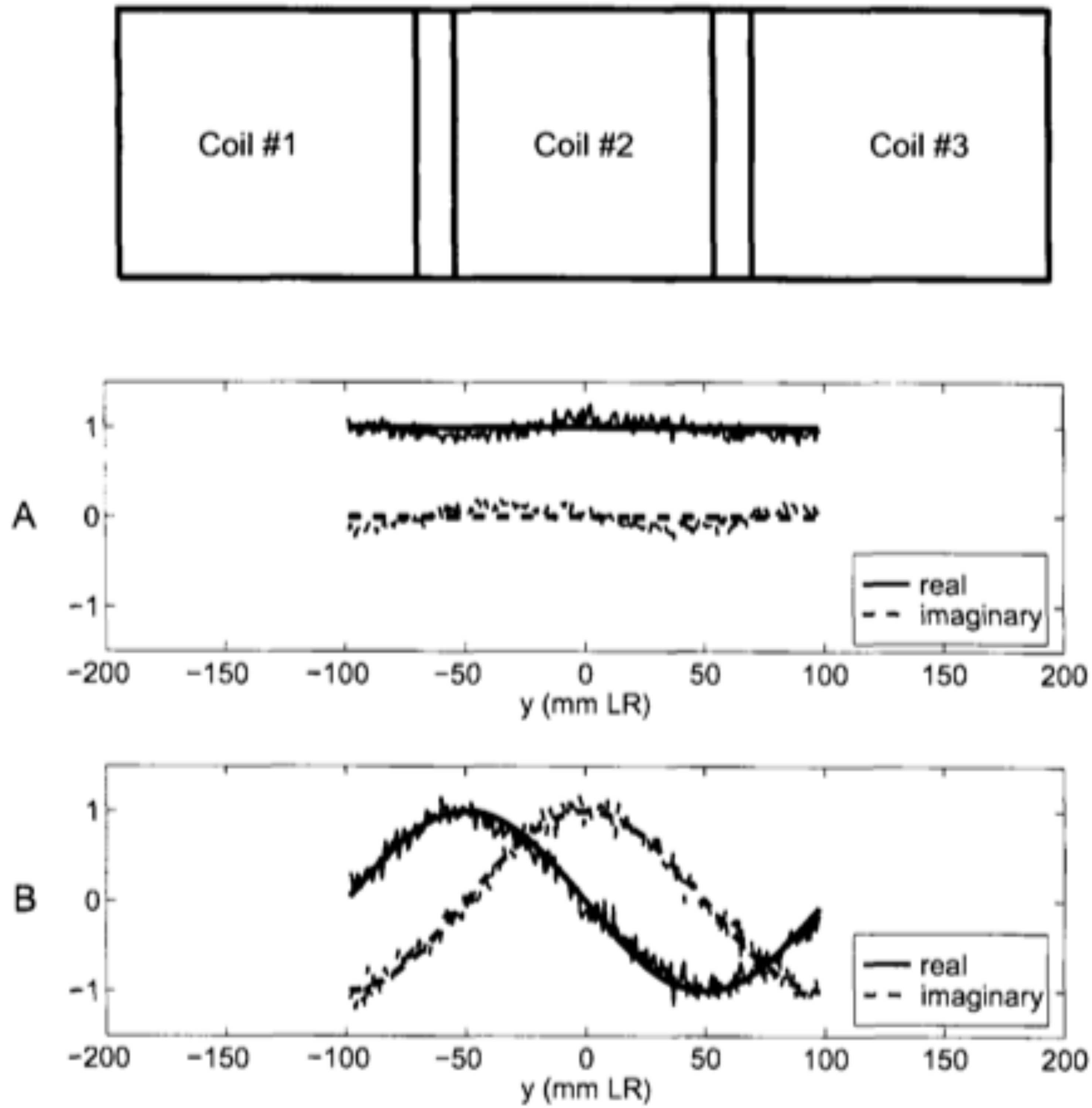
$$C_m^{comp}(y) = \sum_j a_{j,m} C_j(y) = e^{-i2\pi m \Delta k_y y}$$

- Linear surface coil array sensitivities C_j are combined with linear weights, $a_{j,m}$, to produce composite sinusoidal sensitivity
- Composite sensitivities are arranged to be spatial harmonics
- m is an integer, chosen to be a desired harmonic

Theory: Spatial Harmonics

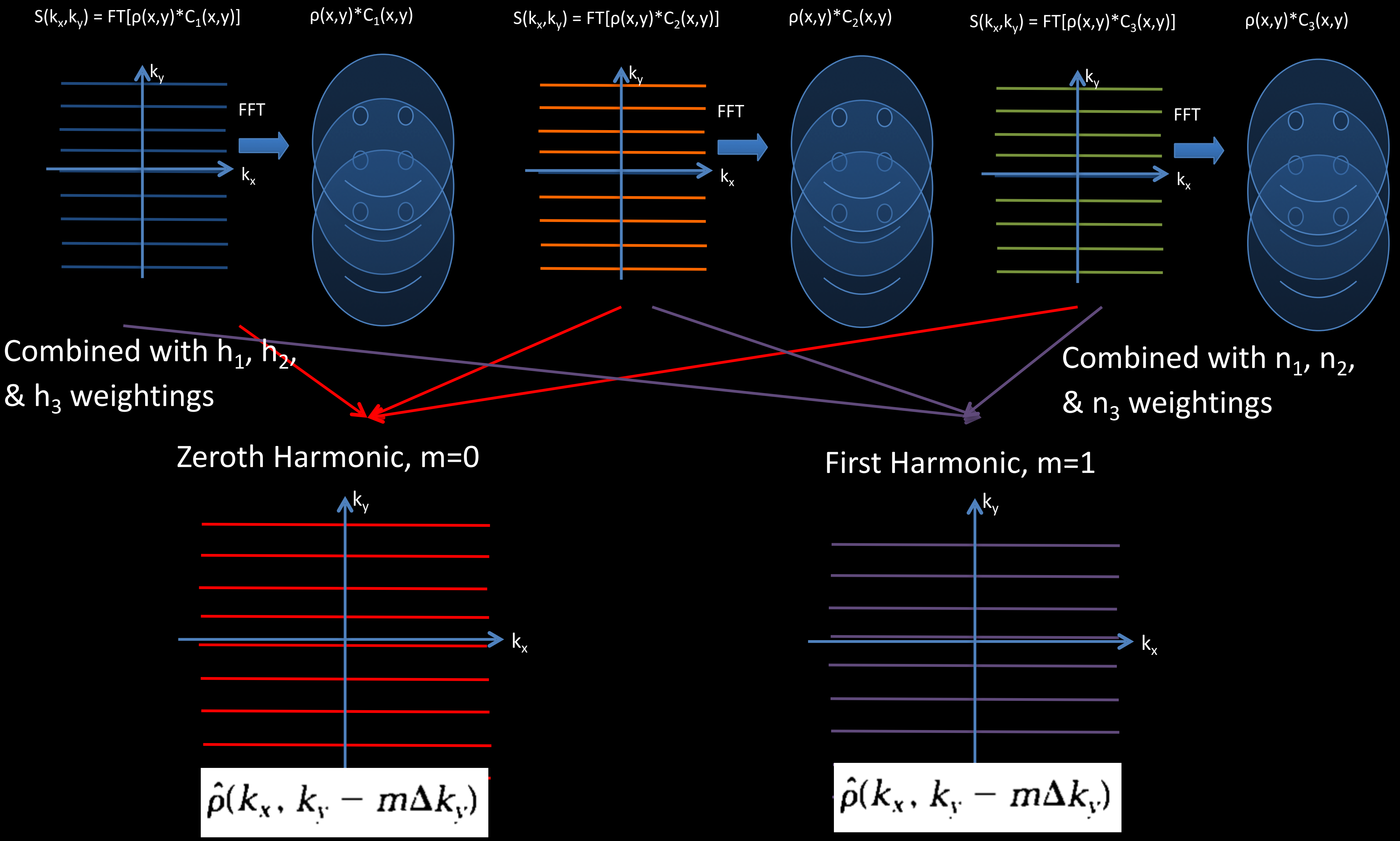
- 8 coil array
- Gaussian coil sensitivity distribution used
- m = 0, 1, -1, 2, -2
- Each spatial harmonic generated is shifted by $-m\Delta k_y$





Coil #1
Coil #2
Coil #3
A
1
0
-1
real
imaginary
-200
-150
-100
-50
0
50
100
150
200
y (mm LR)
B
1
0
-1
real
imaginary
-200
-150
-100
-50
0
50
100
150
200
y (mm LR)

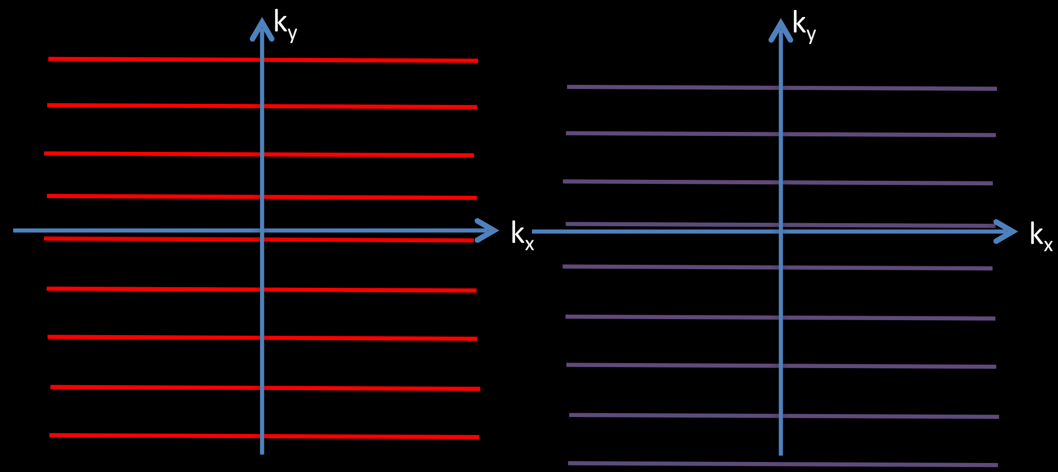
SMASH Reconstruction

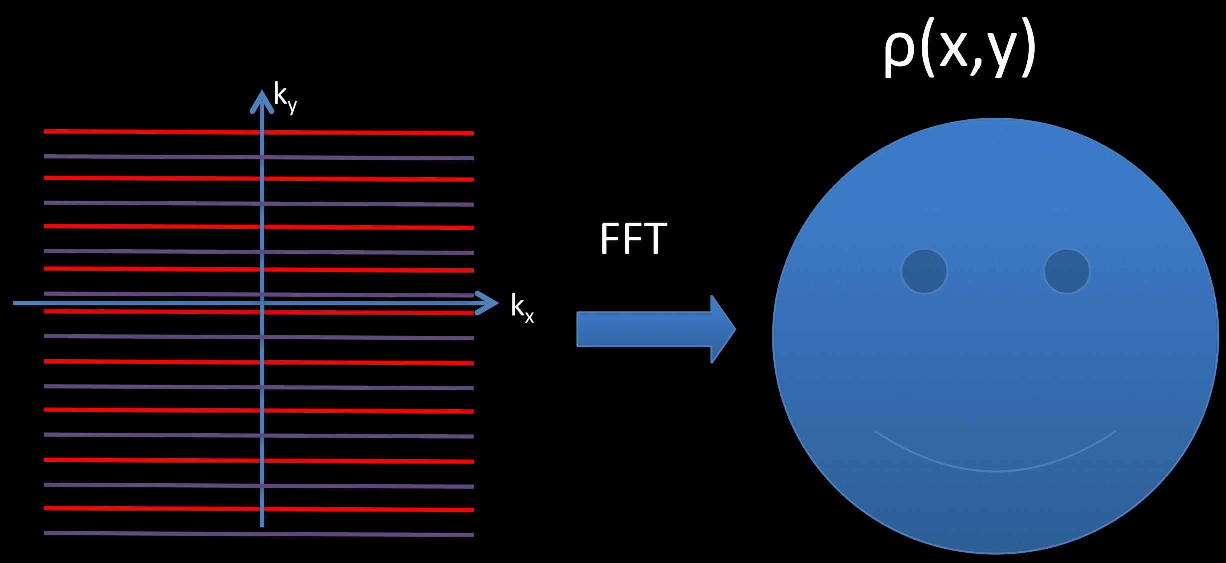


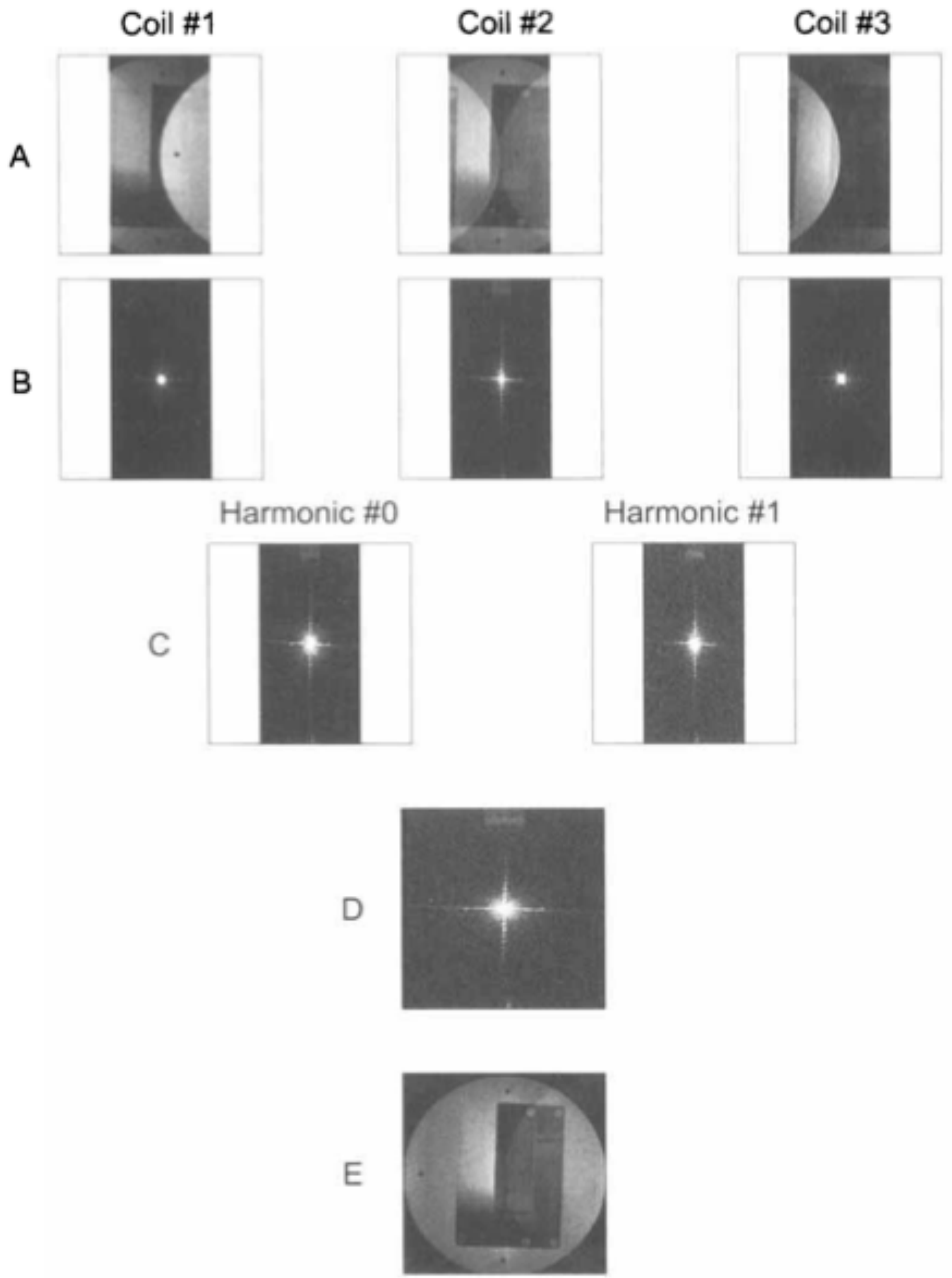
SMASH Reconstruction

Zeroth Harmonic, m=0 First Harmonic, m=1

$$\hat{\rho}(k_x, k_y - m\Delta k_y)$$





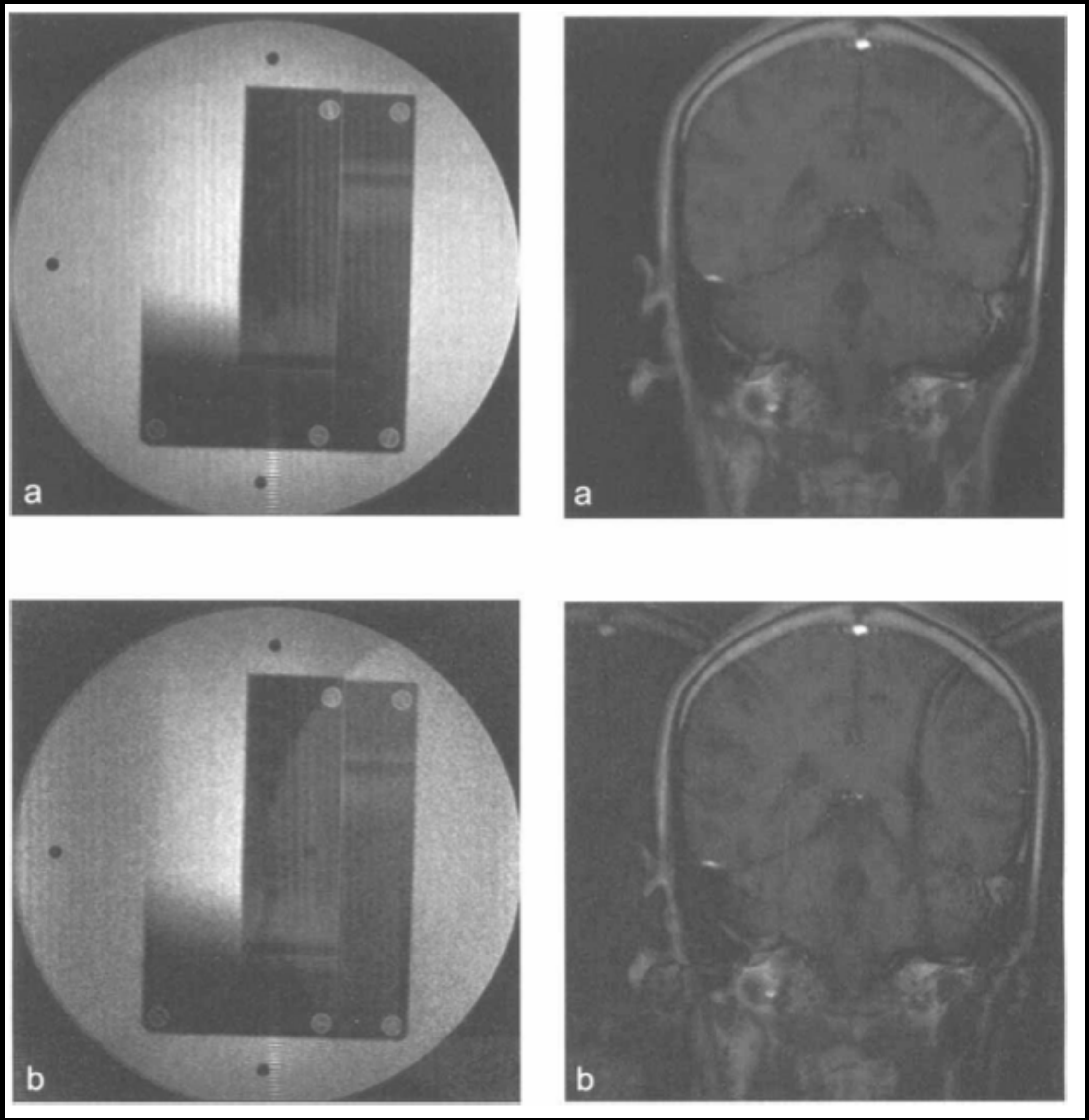


Coil #1
Coil #2
Coil #3
A
B
Harmonic #0
Harmonic #1
C
D
E

Three-Element Array

Reference images

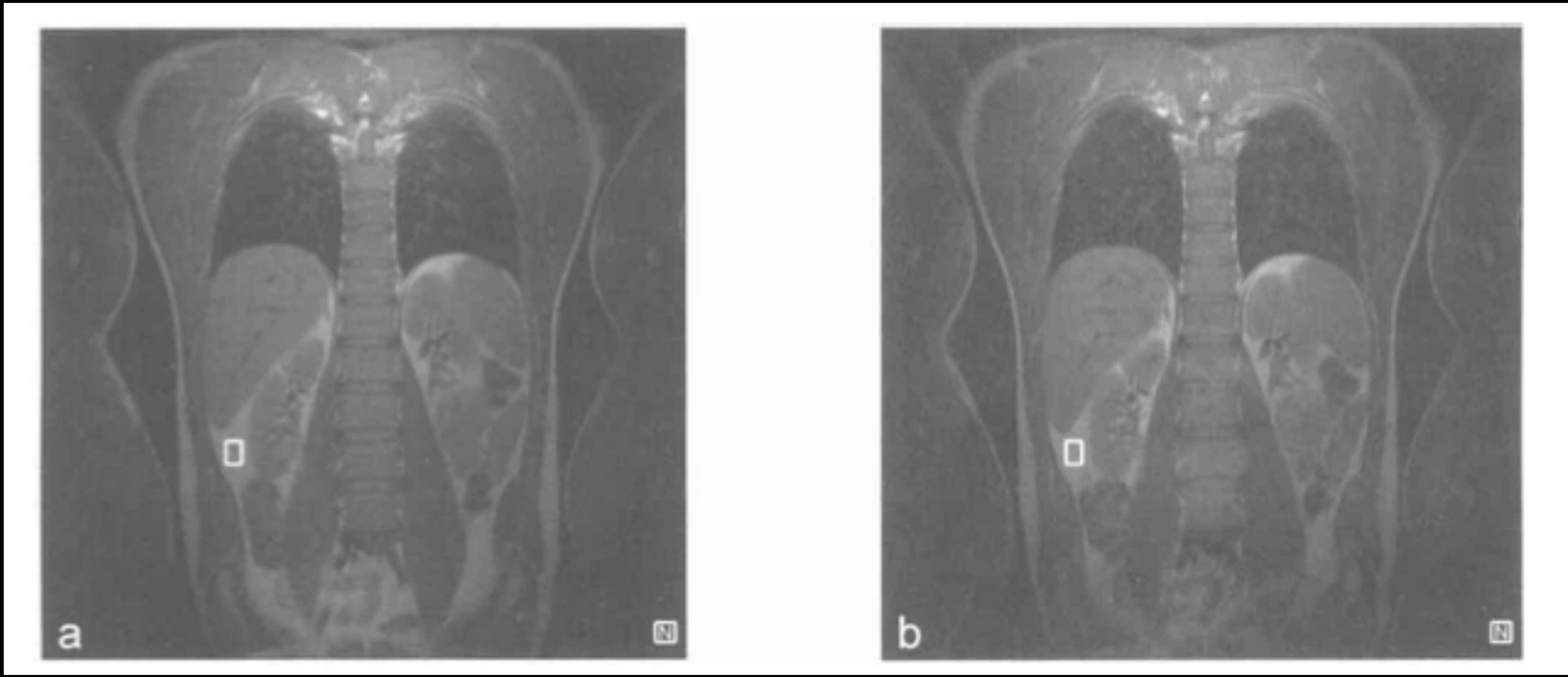
SMASH images



Four-Element Array

Reference images

SMASH images



Key Points of SMASH

- k-space lines are synthesized by combining signals from multiple coils such that it creates a partial replacement for a phase encoding gradient
- Decreases acquisition time by 1/N
  - N is the number of generated spatial Harmonics

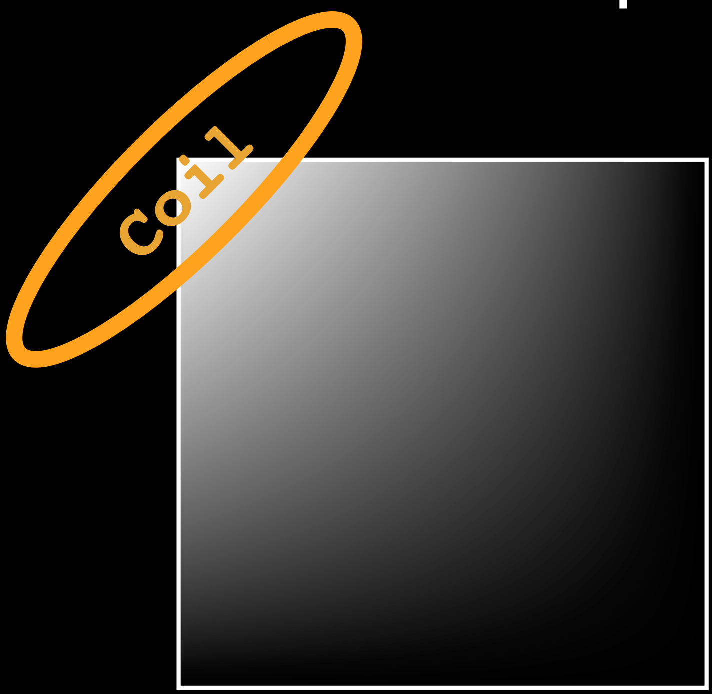
$$\sum_j a_{j,m} C_j(y) = e^{-i2\pi \Delta k_y y}$$

Sodickson et al. MRM 1997

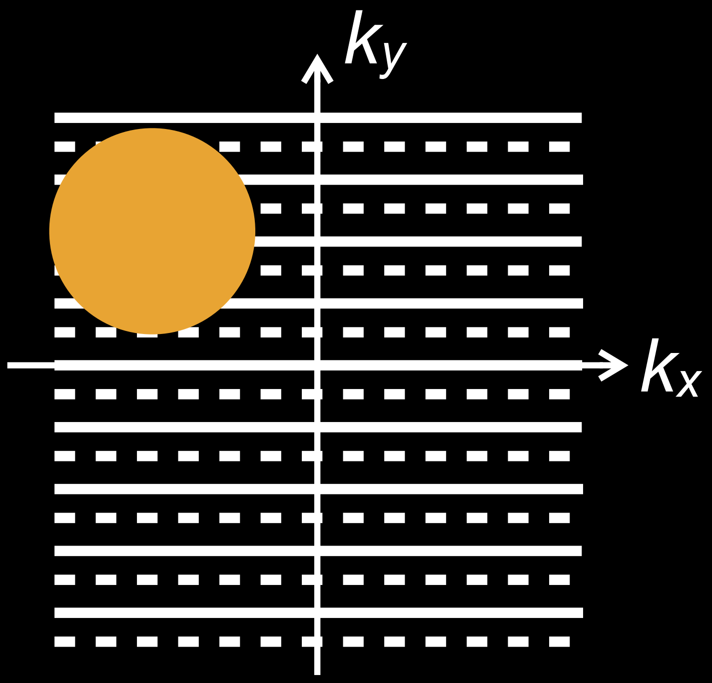
Parallel Imaging
(GRAPPA)

GRAPPA

- Coil sensitivities are
  - Smooth in image space
  - Local in k-space



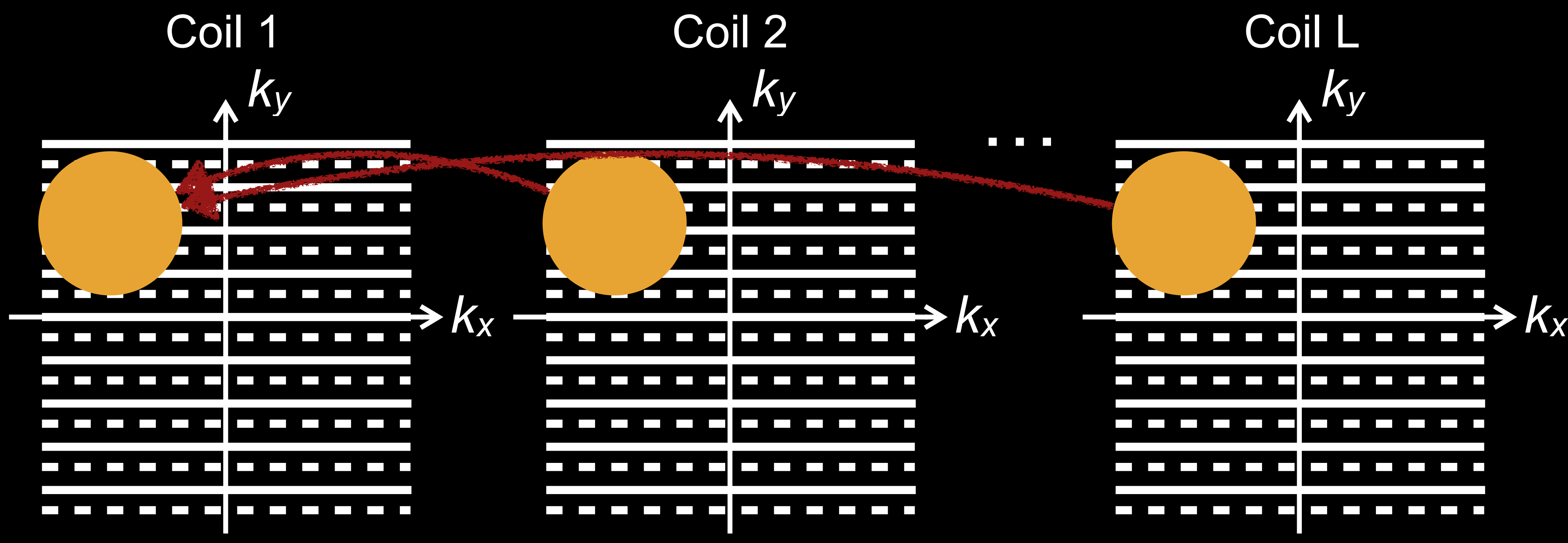
$m(\vec{x})C_j(\vec{x})$



$M(\vec{k}) * C_j(\vec{k})$

GRAPPA

- Missing information is implicitly contained by adjacent data



$M(\vec{k}) * C_1(\vec{k})$ $M(\vec{k}) * C_2(\vec{k})$ $M(\vec{k}) * C_L(\vec{k})$

GRAPPA Reconstruction

- How do we find missing data from these samples?

$$\underbrace{\hat{m}_k(k_x, k_y)}_{\text{missing data for each coil}} = \sum_{i,j,k} \underbrace{a_{i,j,k}}_{\text{weights}} \cdot \underbrace{m_k(k_x + i\Delta k_x, k_y + j\Delta k_y)}_{\text{neighborhood data for each coil}}$$

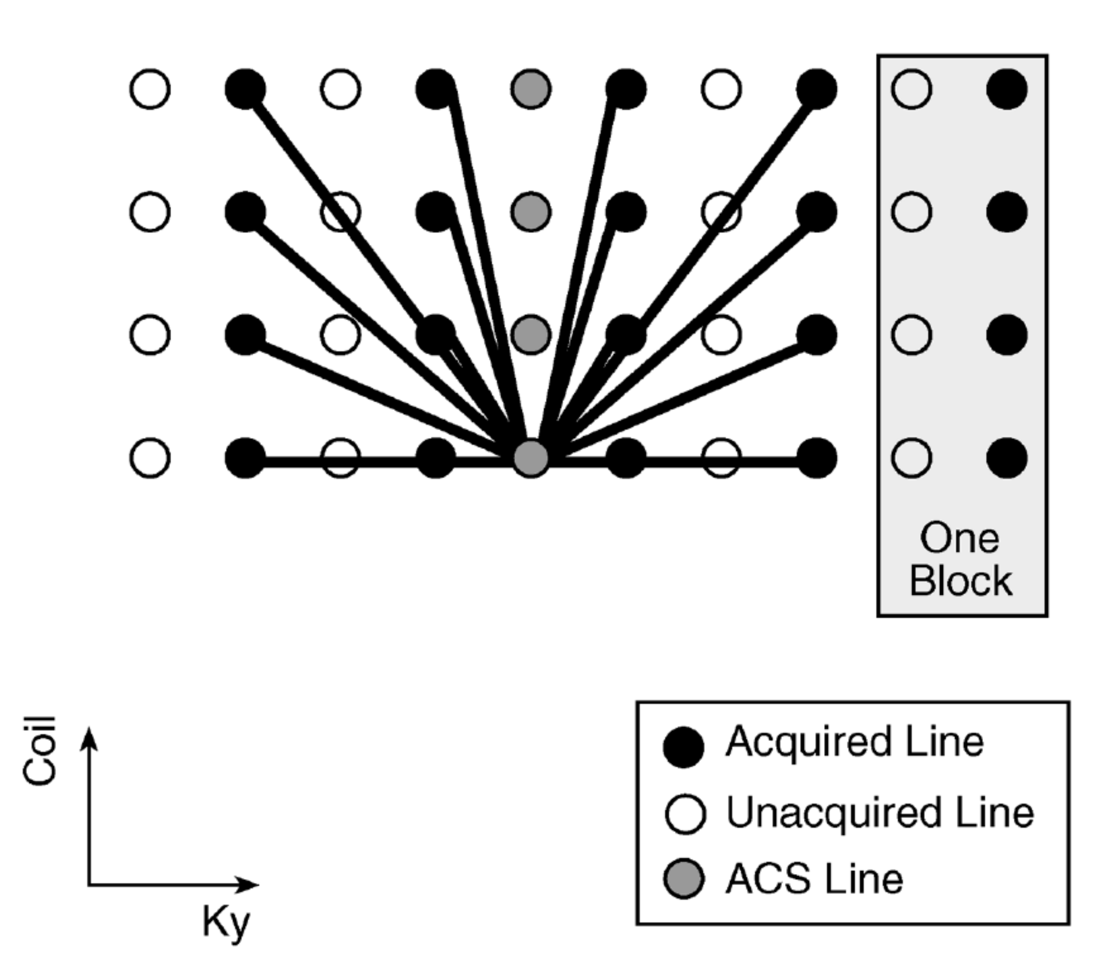
missing data
for each coil

weights

neighborhood data
for each coil

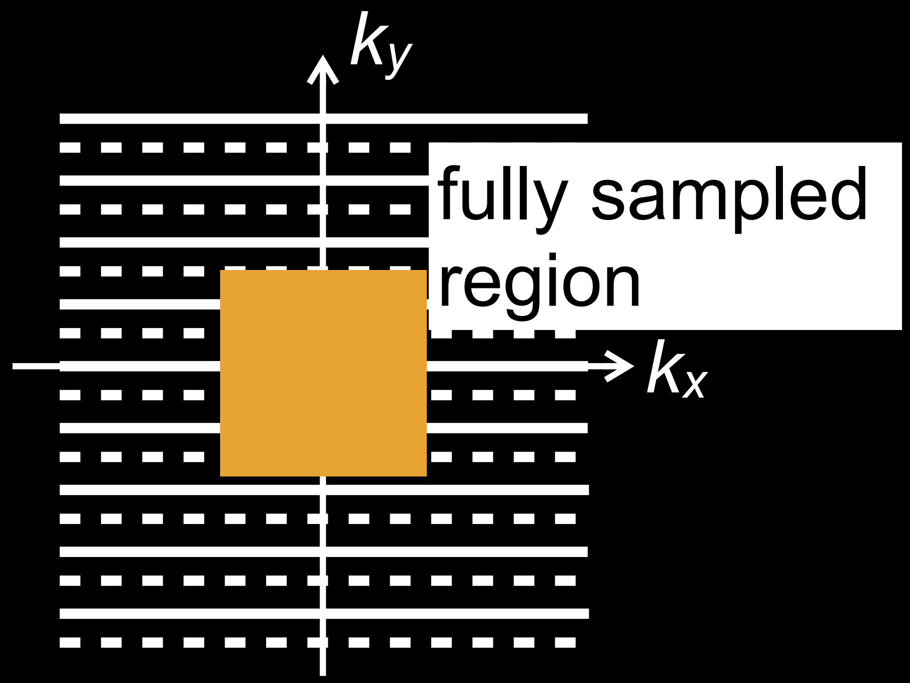
Auto-Calibration

$$\hat{m}_k(k_x, k_y) = \sum_{i,j,k} a_{i,j,k} \cdot m_k(k_x + i\Delta k_x, k_y + j\Delta k_y)$$



Auto-Calibration

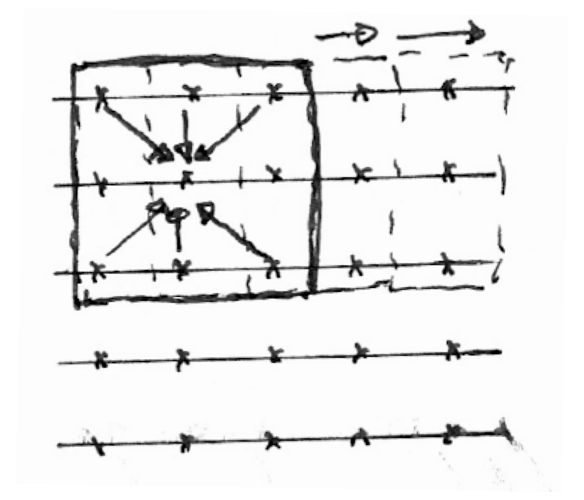
- Assume there is a fully sampled region
- We have samples of what the GRAPPA synthesis equations should produce



- Invert this to solve for GRAPPA weights

Auto-Calibration

- Calibration area has to be larger than the GRAPPA kernel
- Each shift of kernel gives another equation



- Here, 3x3 kernel, 5x5 calibration area gives 9 equations

Auto-Calibration

$$\hat{m}_k(k_x, k_y) = \sum_{i,j,k} a_{i,j,k} \cdot m_k(k_x + i\Delta k_x, k_y + j\Delta k_y)$$

- Write as a matrix equation

GRAPPA Coefficients

$$M_{k,c} = M_A \cdot a_k$$

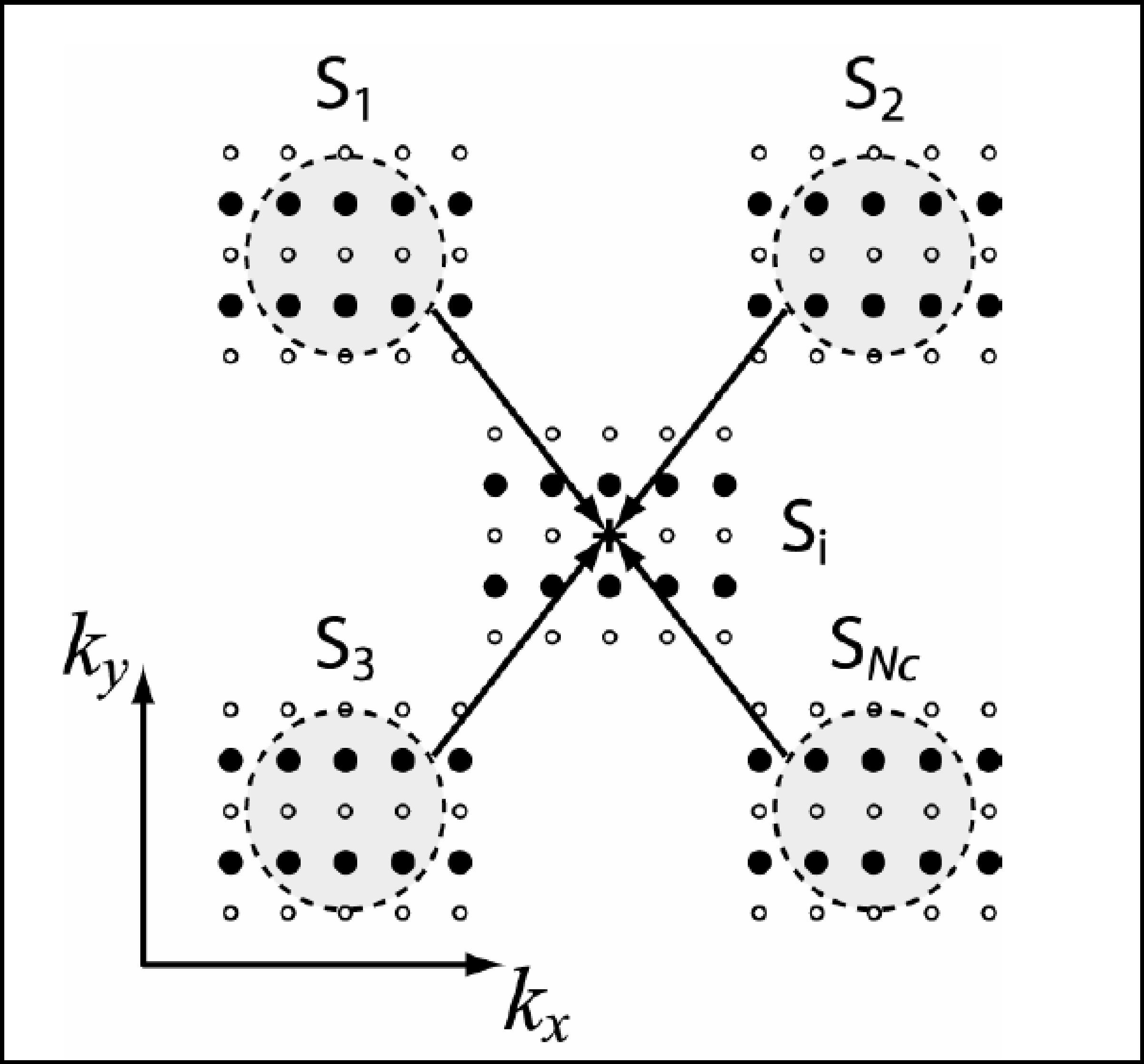
Calibration Data

Neighborhood Data

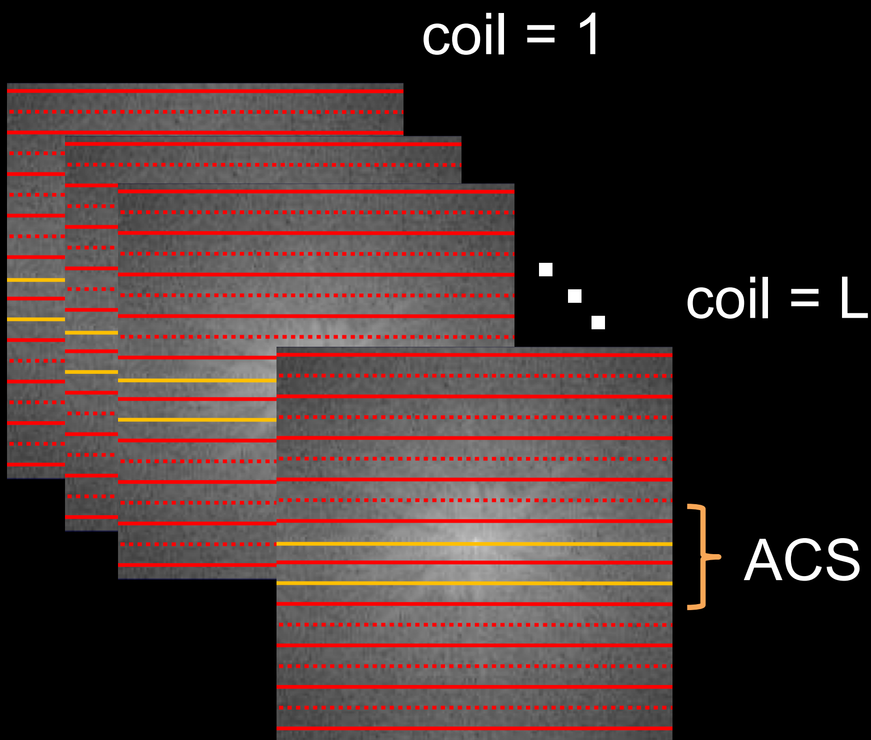
- GRAPPA weights are:

$$a_k = (M_A^* M_A + \lambda I)^{-1} M_A^* M_{k,c}$$

GRAPPA - Synthesis



Auto-Calibration Parallel Imaging



ACS (Auto-Calibration Signal) lines

$$\sum_{l=1}^{L} S_l^{ACS}(k_y - m\Delta k_y) = \sum_{l=1}^{L} n(l, m) S_l(k_y)$$

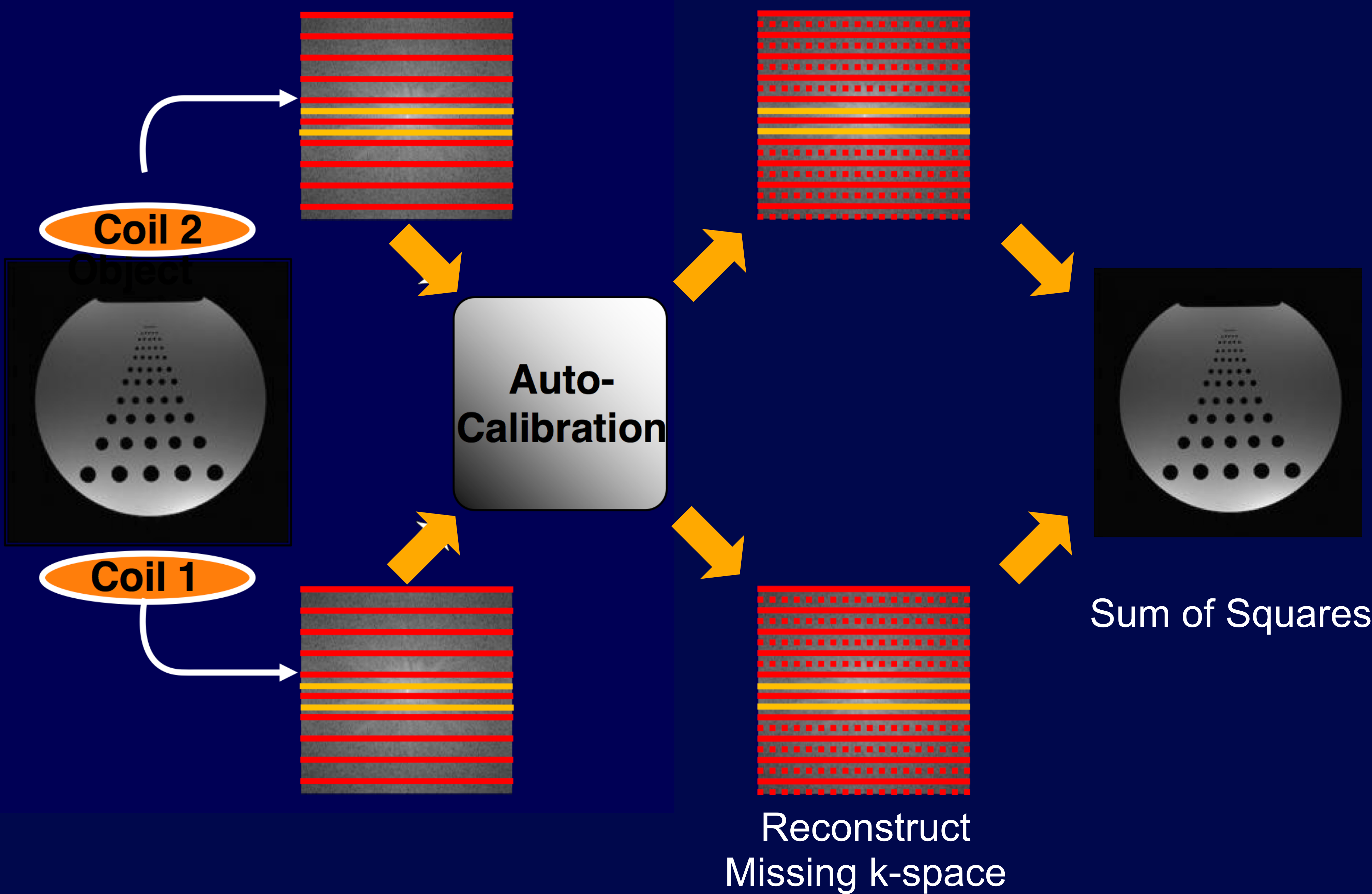
GRAPPA formula to reconstruct signal in one channel

$$S_j(k_y - m\Delta k_y) = \sum_{l=1}^{L} \sum_{b=0}^{N_b - 1} n(j, b, l, m) S_l(k_y - bA\Delta k_y)$$

A: Acceleration factor
n(j,b,l,m): GRAPPA weights

Griswold et al. MRM, 47(6):1202-1210 (2002)

GRAPPA Reconstruction

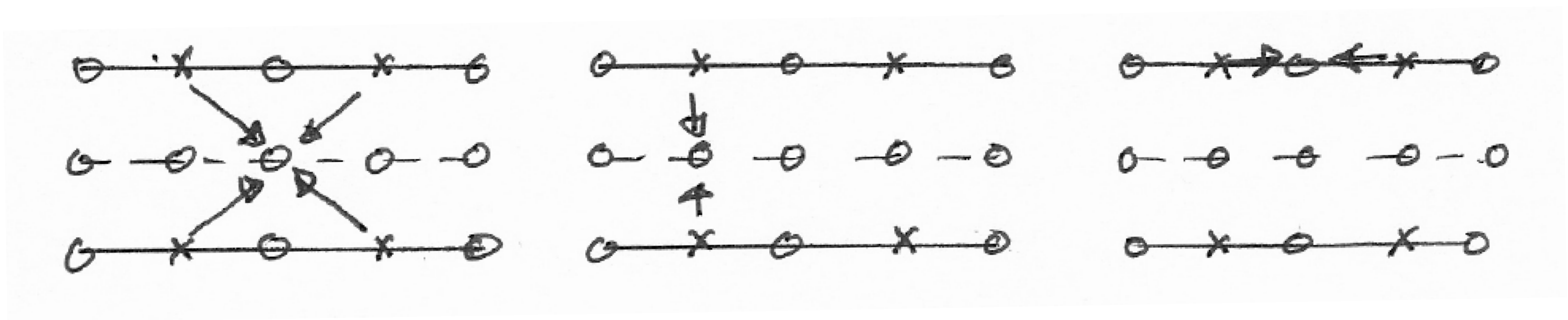


GRAPPA

- Compute GRAPPA weights from calibration region
- Compute missing k-space data using the GRAPPA weights
- Reconstruct individual coil images
- Combine coil images

Considerations of GRAPPA

- Calibration region size
- GRAPPA kernel size
- Sample geometry dependence



GRAPPA

- Compute GRAPPA weights from calibration region
- Compute missing k-space data using the GRAPPA weights
- Reconstruct individual coil images
- Combine coil images

Summary

- Parallel imaging utilizes coil sensitivities to increase the speed of MRI
- Cases for parallel imaging
  - Higher patient throughput,
  - Real-time imaging/Interventional imaging
  - Motion suppression
- Cases against parallel imaging
  - SNR starving applications

Summary

- Many approaches:
  - Image domain - SENSE
  - k-space domain - SMASH, GRAPPA
  - Hybrid - ARC

- We will focus on two:
  - SENSE: optimal if you know coil sensitivities
  - GRAPPA: autocalibrating / robust

Further Reading

- Multi-coil Reconstruction
  - http://onlinelibrary.wiley.com/doi/10.1002/mrm.1910160203/abstract
- SENSE
  - http://www.ncbi.nlm.nih.gov/pubmed/10542355
- SMASH
  - http://www.ncbi.nlm.nih.gov/pubmed/9324327
- Parallel Imaging Overview
  - http://www.ncbi.nlm.nih.gov/pubmed/17374908

Thanks!

- Next time
  - Compressed Sensing

Kyung Sung, PhD

ksung@mednet.ucla.edu

https://mrrl.ucla.edu/sunglab/