Statistical methods for tomographic image reconstruction

Jeffrey A. Fessler

EECS Department, BME Department, and
Nuclear Medicine Division of Dept. of Internal Medicine
The University of Michigan

GE CRD

Jan 7, 2000

Outline

- Group/Lab
- PET Imaging
- Statistical image reconstruction

  Choices / tradeoffs / considerations:
  - 1. Object parameterization
  - 2. System physical modeling
  - 3. Statistical modeling of measurements
  - 4. Objective functions and regularization
  - 5. Iterative algorithms

  Short course lecture notes:

  `http://www.eecs.umich.edu/~fessler/talk`
- Ordered-subsets transmission ML algorithm
- Incomplete data tomography

Students

- El Bakri, Idris — Analysis of tomographic imaging
- Ferrise, Gianni — Signal processing for direct brain interface
- Ghanei, Amir — Model-based MRI brain segmentation
- Kim, Jeongtae — Image registration/reconstruction for radiotherapy
- Stayman, Web — Regularization methods for tomographic reconstruction
- Sotthivirat, Saowapak — Optical image restoration
- Sutton, Brad — MRI image reconstruction
- Yu, Feng (Dan) — Nonlocal regularization for transmission reconstruction

Collaborations with colleagues in Biomedical Engineering, EECS, Nuclear Engineering, Nuclear Medicine, Radiology, Radiation Oncology, Physical Medicine, Anatomy and Cell Biology, Biostatistics

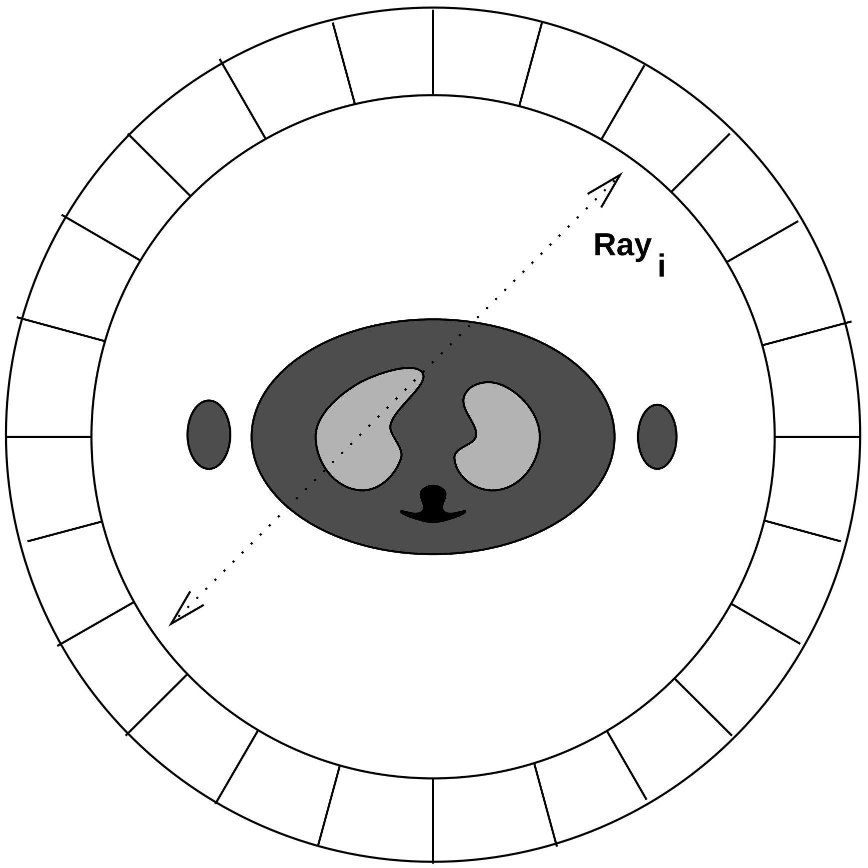
Research Goals

- Develop methods for making "better" images
  (modeling of imaging system physics and measurement statistics)
- Faster algorithms for computing/processing images
- Analysis of the properties of image formation methods
- Design of imaging systems based on performance bounds

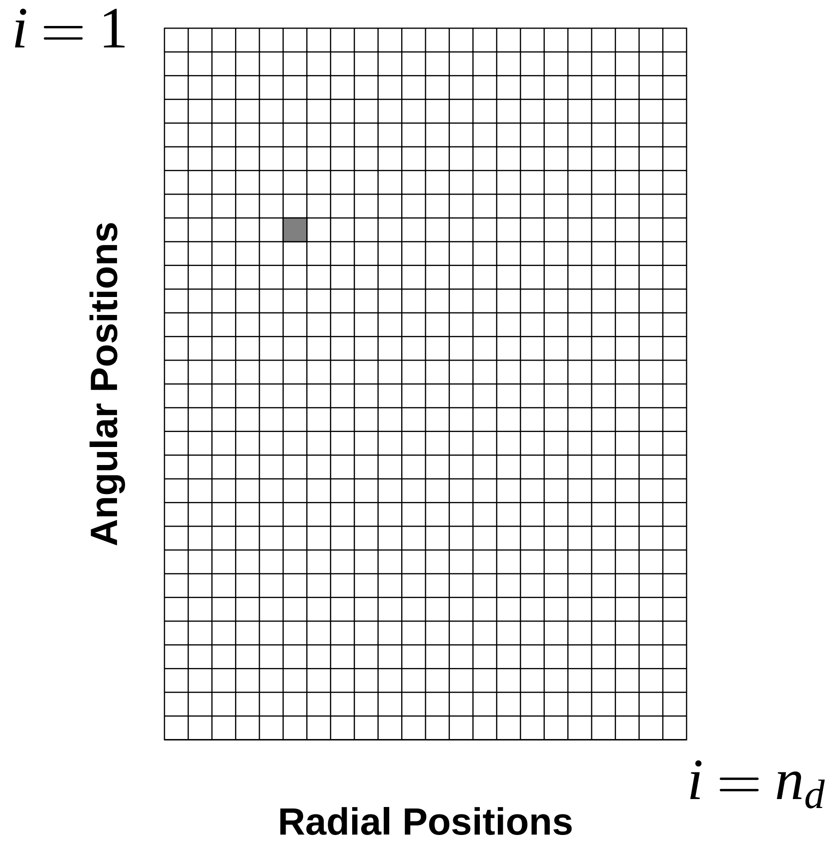
Impact

- ASPIRE (A sparse iterative reconstruction environment) software
  (about 40 registered sites worldwide)
- PWLS reconstruction used routinely for cardiac SPECT at UM,
  following 1996 ROC study. ($>$ 2000 patients scanned)
- Pittsburgh PET/CT "side information" scans reconstructed using ASPIRE

PET Data Collection

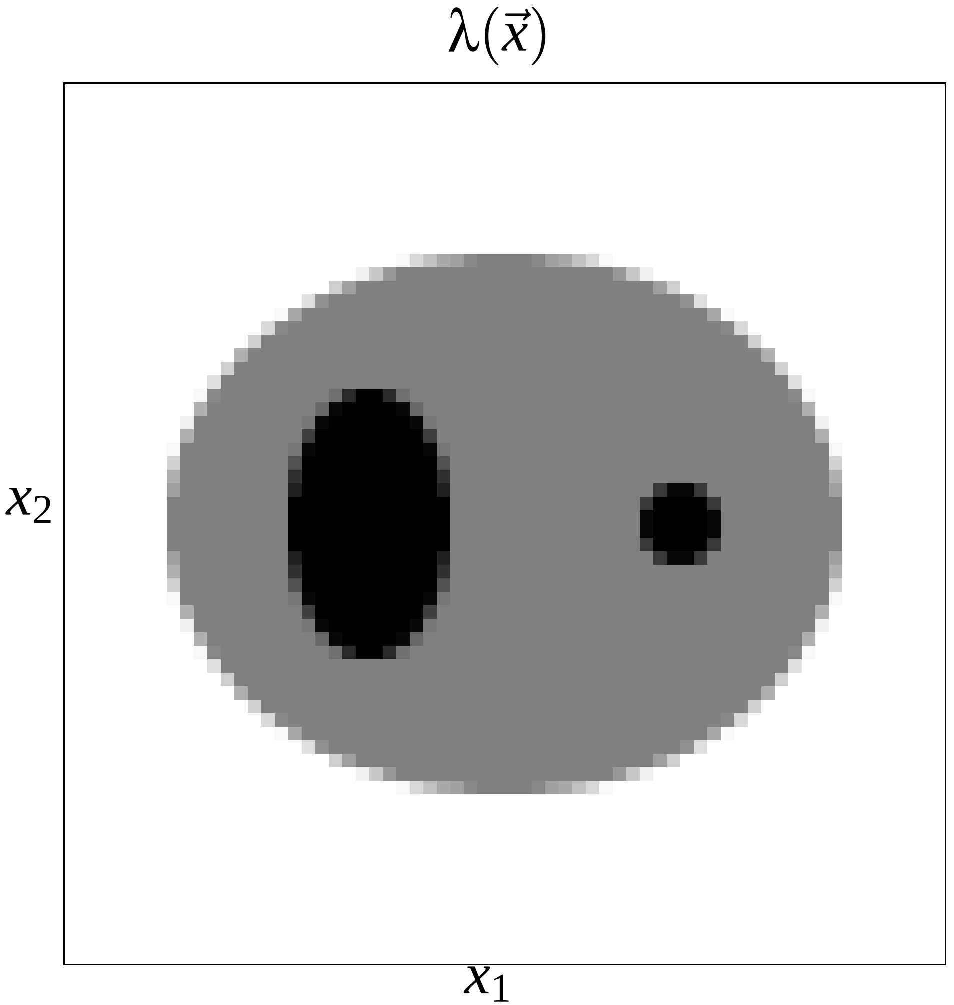


Sinogram

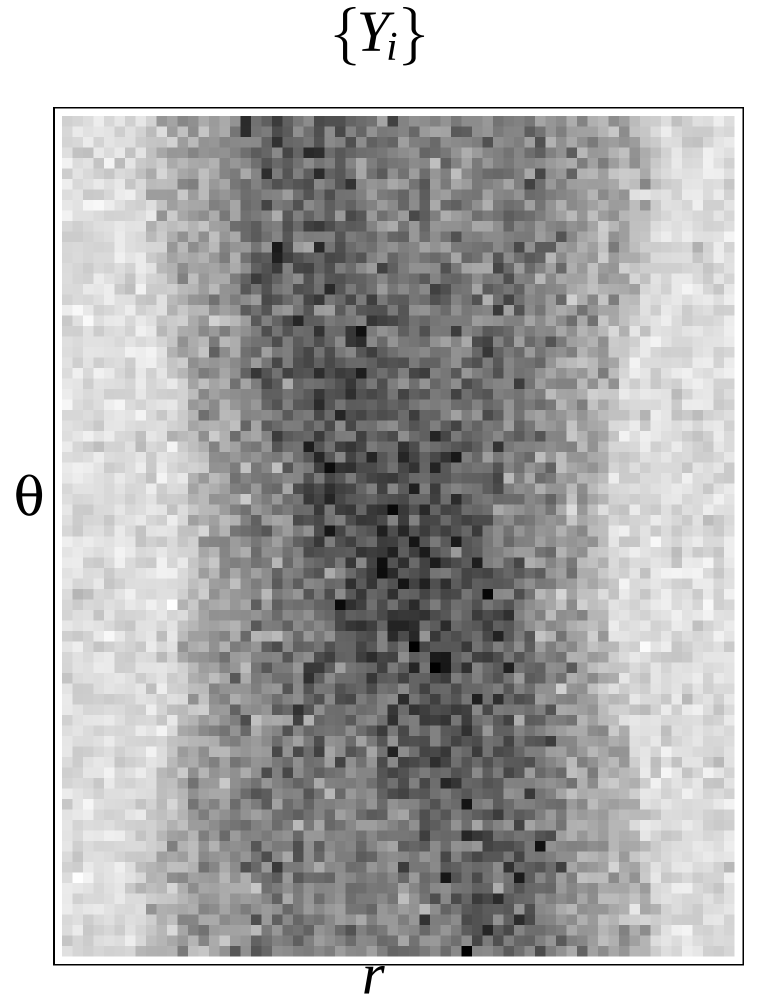


$$n_d \approx (n_{\text{crystals}})^2$$

PET Reconstruction Problem - Illustration



Image



Sinogram

Reconstruction Methods

(Simplified View)

Analytical (FBP)	Iterative (OSEM?)

Reconstruction Methods

- **ANALYTICAL**
  - FBP
  - BPF
  - Gridding
  - ...
- **ITERATIVE**
  - **Algebraic** (y = Ax)
    - ART
    - MART
    - SMART
    - ...
  - **Statistical**
    - **Least Squares**
      - CG
      - CD
      - ISRA
      - ...
    - **Poisson Likelihood**
      - EM (etc.)
      - OSEM
      - SAGE
      - CG
      - Int. Point
      - GCA
      - PSCD
      - FSCD ...

Why Statistical Methods?

- Object constraints (*e.g.* nonnegativity)
- Accurate models of physics (reduced artifacts, quantitative accuracy)
  (*e.g.* nonuniform attenuation in SPECT, scatter, beam hardening, ...)
- System detector response models (*possibly* improved spatial resolution)
- Appropriate statistical models (reduced image noise or dose)
  (FBP treats all rays equally)
- Side information (*e.g.* MRI or CT boundaries)
- Nonstandard geometries ("missing" data, *e.g.* truncation)

Tradeoffs...

- Computation time
- Model complexity
- Software complexity
- Less predictable (due to nonlinearities), especially for some methods
  e.g. Huesman (1984) FBP ROI variance for kinetic fitting

Five Categories of Choices

1. Object parameterization: $\lambda(\vec{x})$ vs $\underline{\lambda}$
2. System physical model: $s_i(\vec{x})$
3. Measurement statistical model $Y_i \sim \boxed{?}$
4. Objective function: data-fit / regularization
5. Algorithm / initialization

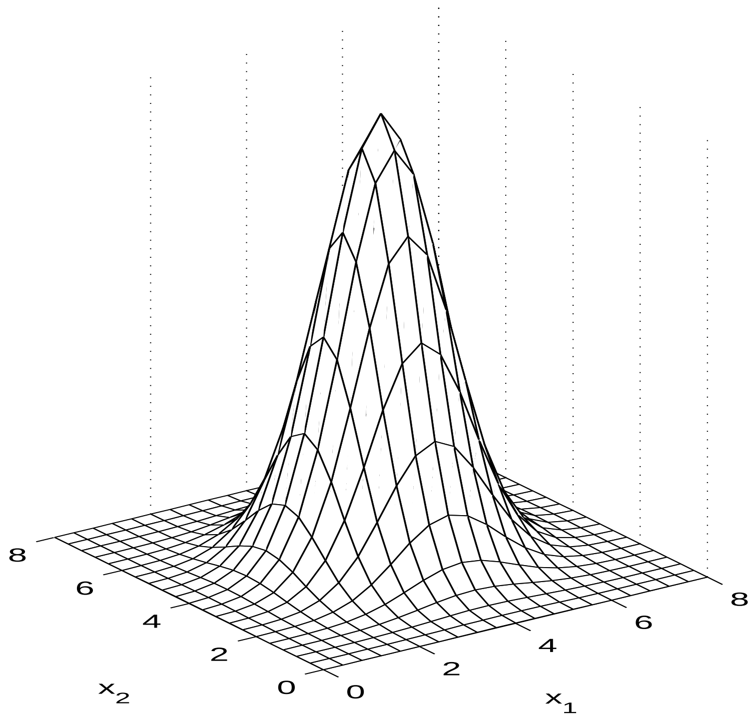
No perfect choices - one can critique all approaches!

Choices impact:

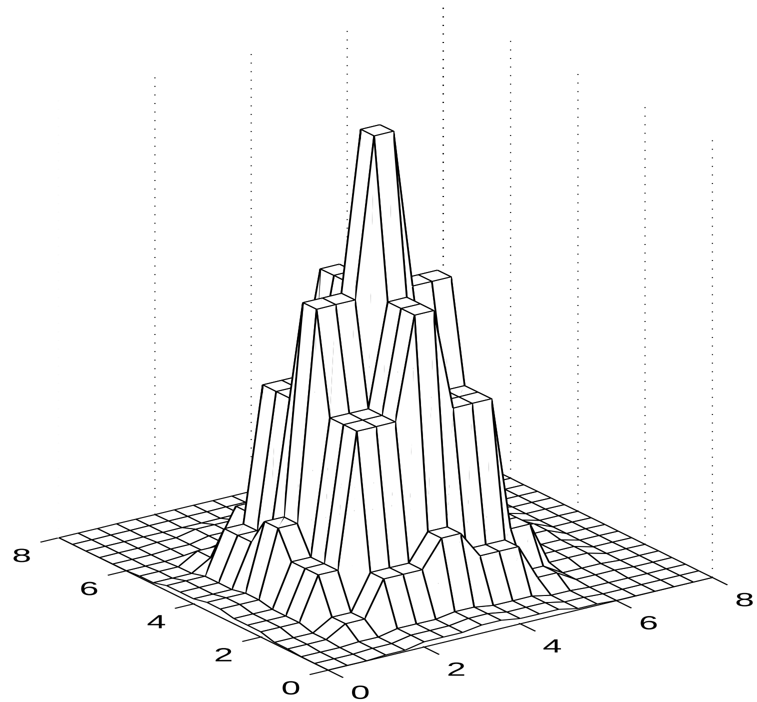
- Image spatial resolution
- Image noise
- Quantitative accuracy
- Computation time
- Memory
- Algorithm complexity

Choice 1. Object Parameterization

Radioisotope spatial distribution $\rightarrow \lambda(\vec{x}) \approx \tilde{\lambda}(\vec{x}) = \sum_{j=1}^{n_p} \lambda_j b_j(\vec{x}) \leftarrow$ Series expansion "basis functions"



Object $\lambda(\vec{x})$



Pixelized approximation $\tilde{\lambda}(\vec{x})$

Basis Functions

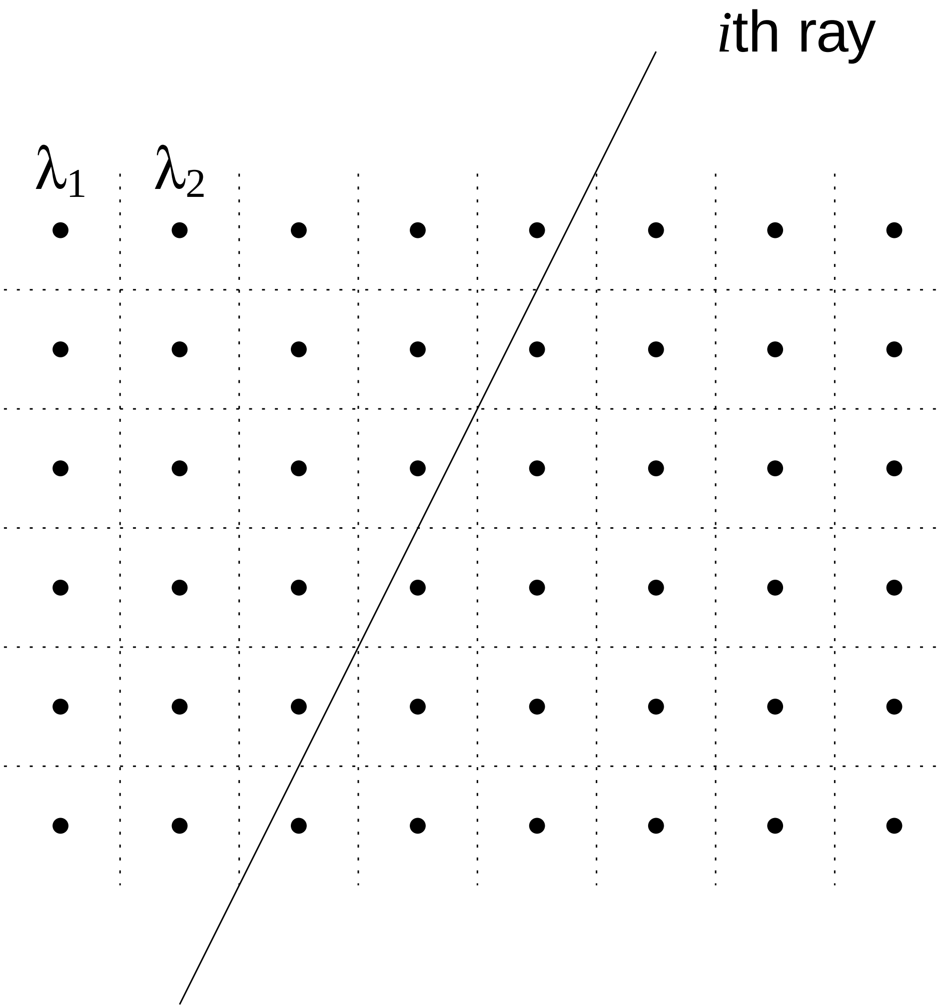
Choices

- Fourier series
- Circular harmonics
- Wavelets
- Kaiser-Bessel windows
- Overlapping disks
- B-splines (pyramids)
- Polar grids
- Logarithmic polar grids
- “Natural pixels”
- Point masses
- pixels / voxels
- ...

Considerations

- Represent object $\lambda(\vec{x})$ “well” with moderate n_p
- system matrix elements $\{a_{ij}\}$ “easy” to compute
- The $n_d \times n_p$ system matrix: $A = \{a_{ij}\}$, should be sparse (mostly zeros).
- Easy to represent nonnegative functions
  e.g., if $\lambda_j \geq 0$, then $\lambda(\vec{x}) \geq 0$, *i.e.* $b_j(\vec{x}) \geq 0$.

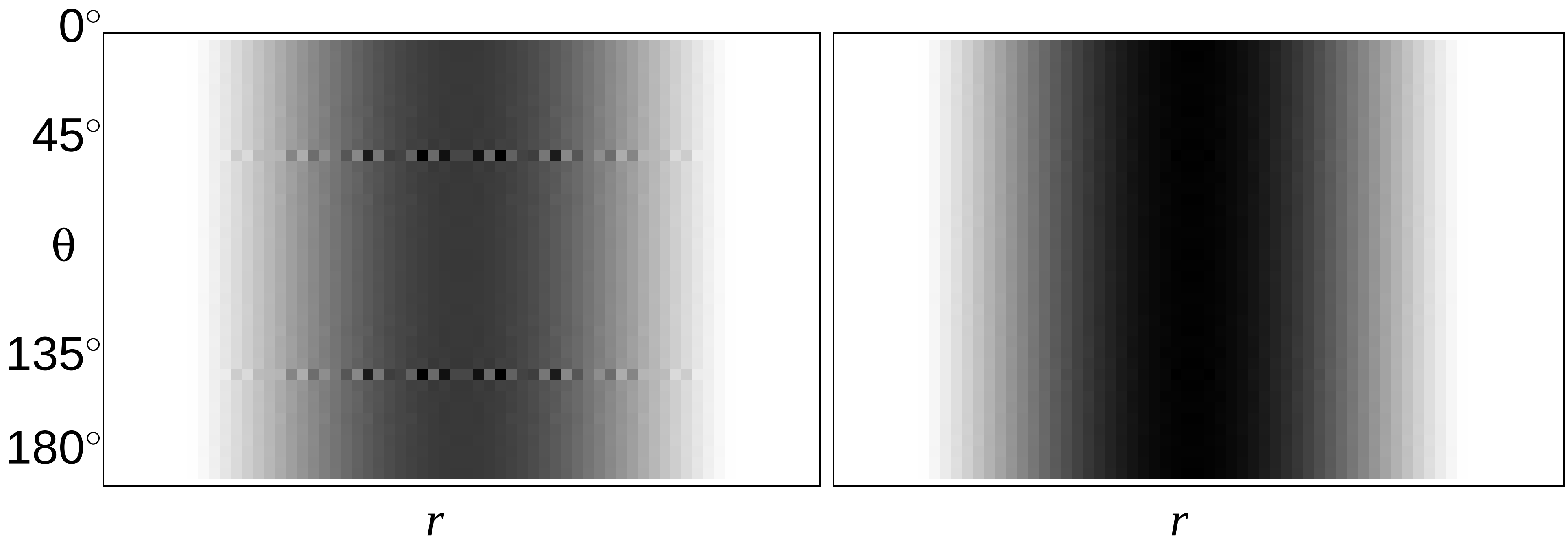
Point-Lattice Projector/Backprojector



a_{ij}'s determined by linear interpolation

Point-Lattice Artifacts

Projections (sinograms) of uniform disk object:



Point Lattice

Strip Area

Choice 2. System Model

System matrix $A = \{a_{ij}\}$ elements:

$$a_{ij} = \text{P[decay in the } j\text{th pixel is recorded by the } i\text{th detector unit]}$$

Physical effects

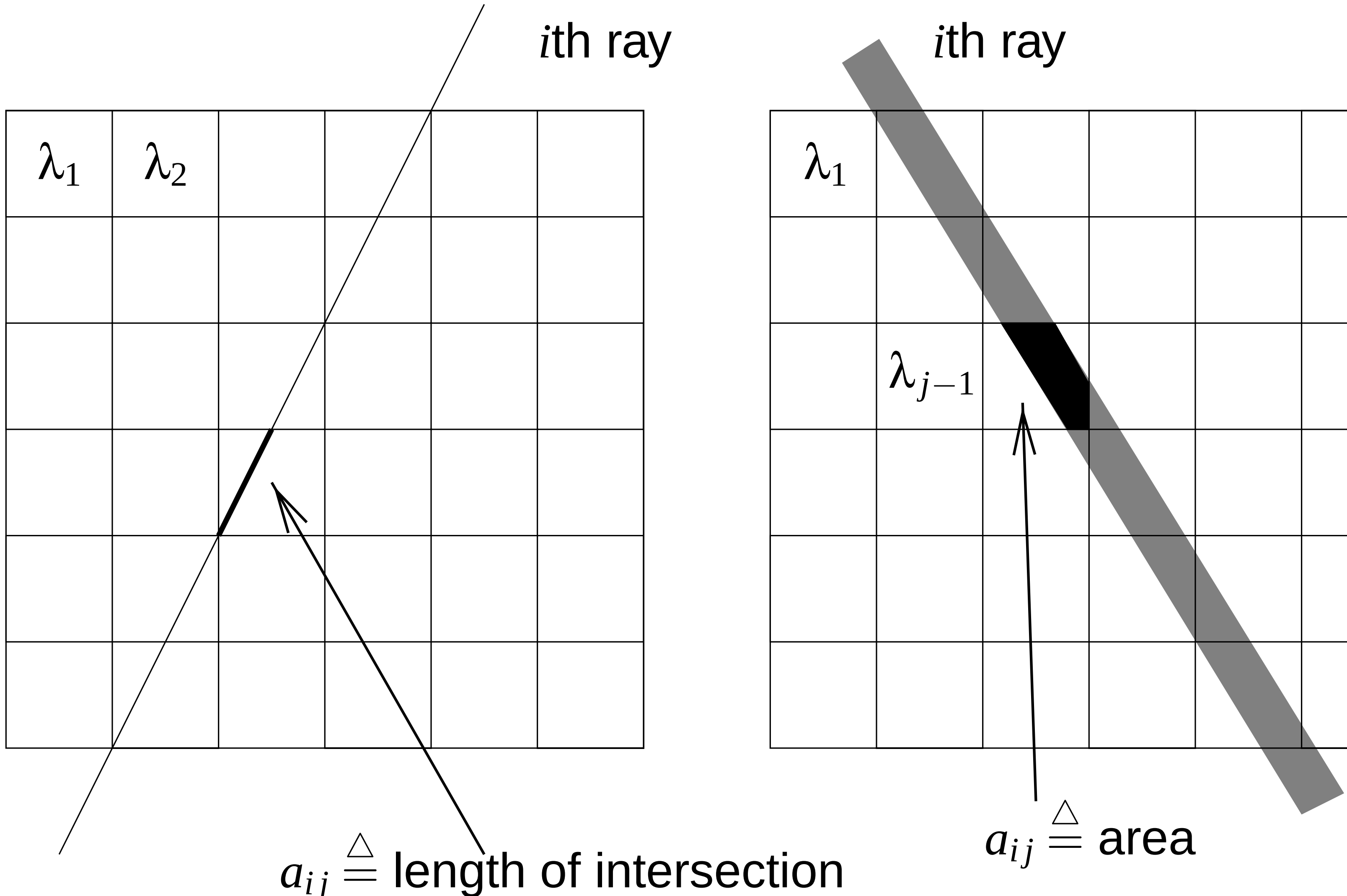
- scanner **geometry**
- solid angles
- detector efficiency
- attenuation
- scatter
- collimation
- detector response
- dwell time at each angle
- dead-time losses
- positron range
- noncolinearity
- ...

Considerations

- Accuracy vs computation and storage vs compute-on-fly
- Model uncertainties
  (*e.g.* calculated scatter probabilities based on noisy attenuation map)
- Artifacts due to over-simplifications

“Line Length” System Model

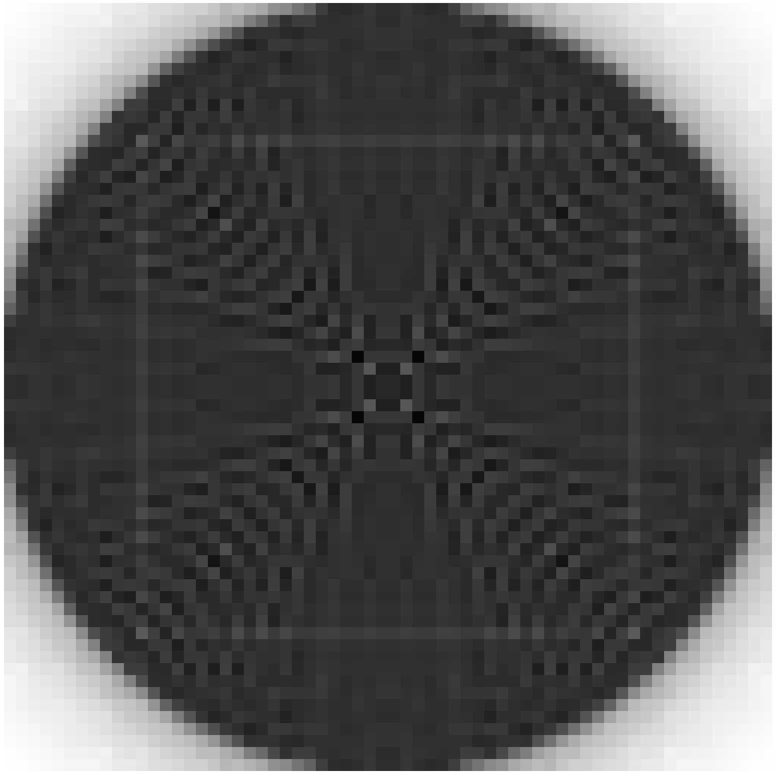
“Strip Area” System Model



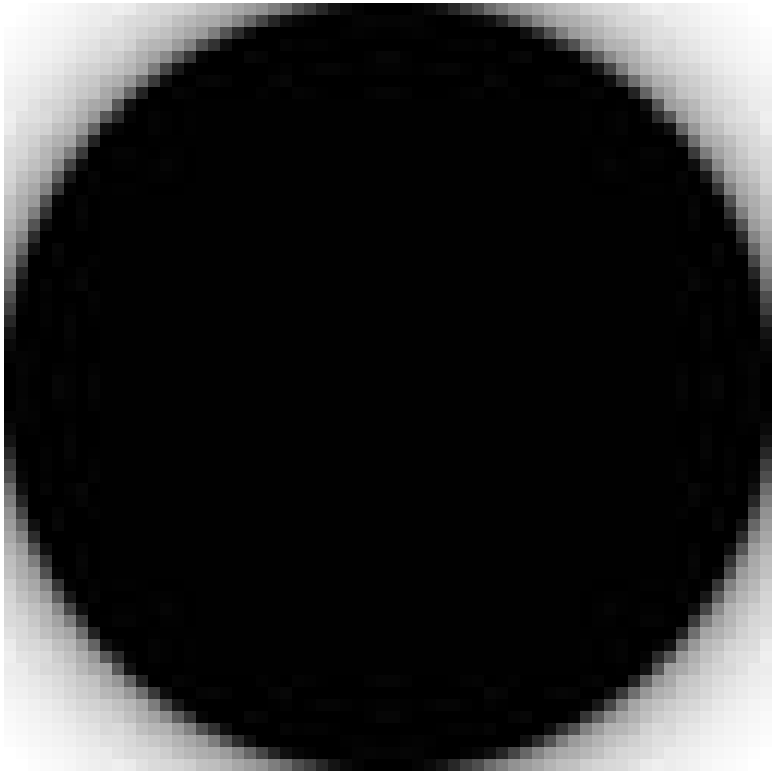
Sensitivity Patterns

$$\sum_{i=1}^{n_d} a_{ij} \approx s(\underline{x}_j) = \sum_{i=1}^{n_d} s_i(\underline{x}_j)$$

Line Length



Strip Area



Forward- / Back-projector “Pairs”

Forward projection (image domain to projection domain):

$$E[Y_i] = \int s_i(\vec{x})\lambda(\vec{x})\,d\vec{x} = \sum_{j=1}^{n_p} a_{ij}\lambda_j = [A\underline{\lambda}]_i, \quad \text{or} \quad E[\underline{Y}] = A\underline{\lambda}$$

Backprojection (projection domain to image domain):

$$A'\underline{y} = \left\{ \sum_{i=1}^{n_d} a_{ij} y_i \right\}_{j=1}^{n_p}$$

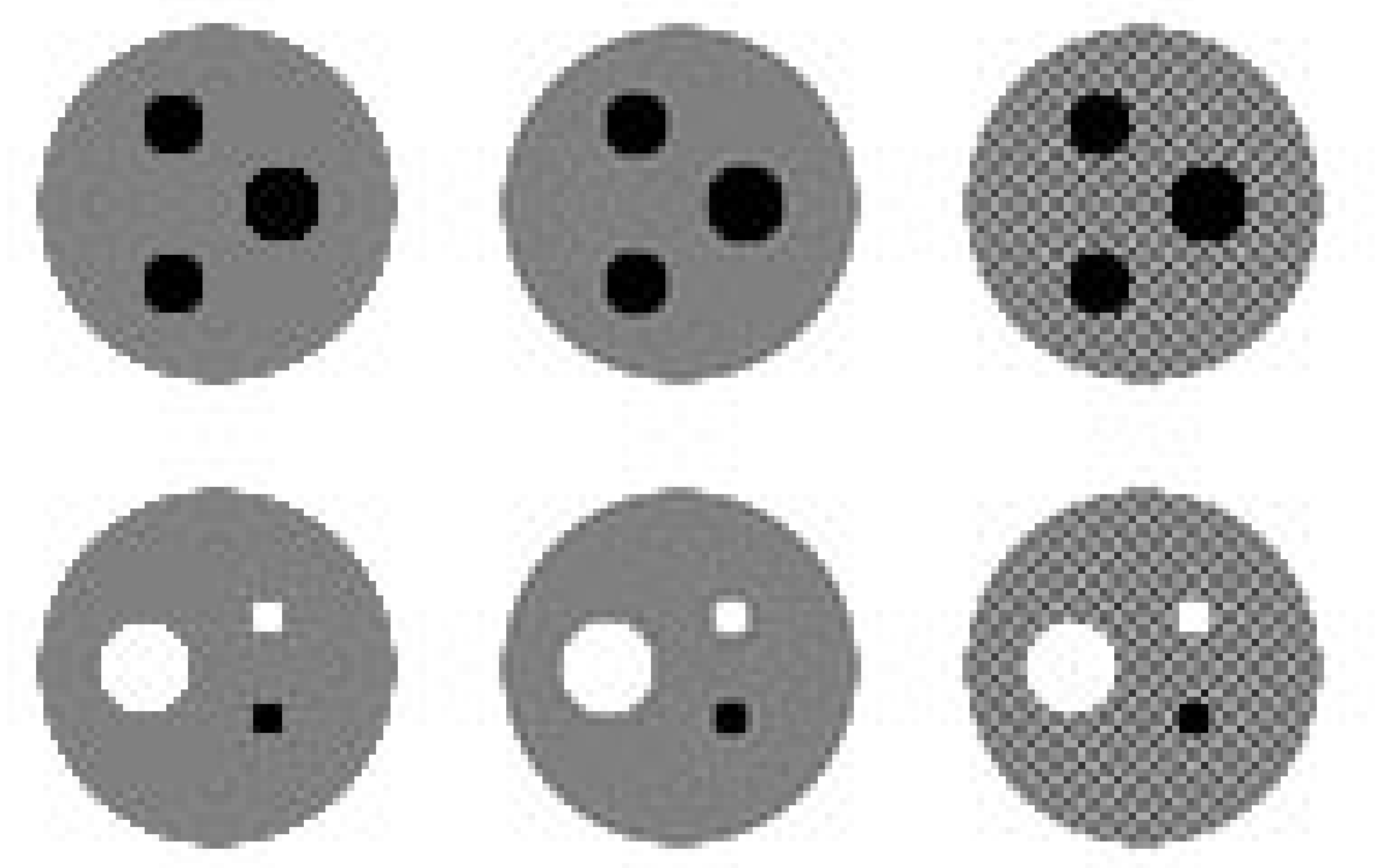
Often A' is implemented as $B\underline{y}$ for some “backprojector” $B \neq A'$

Least-squares solutions (for example):

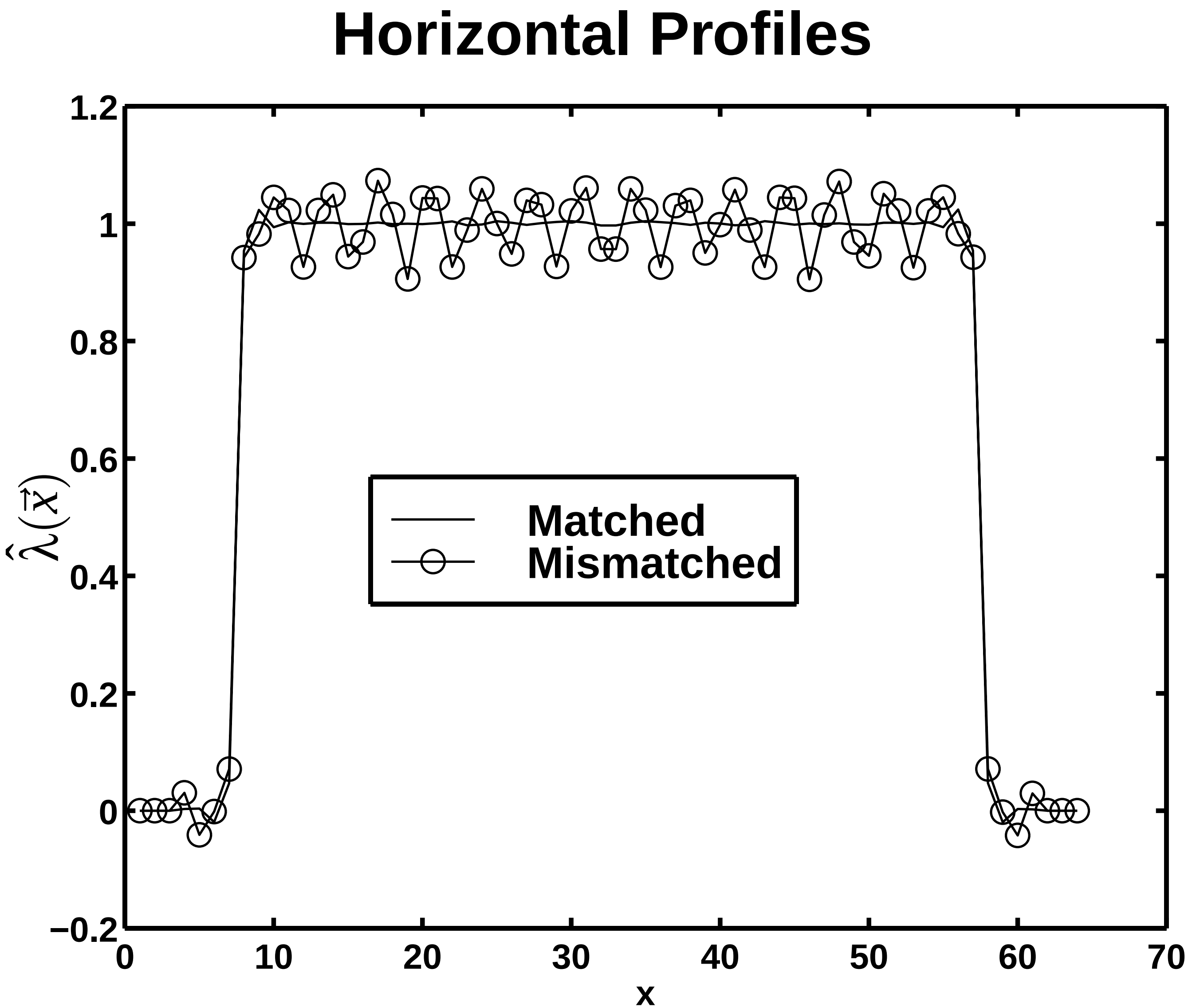
$$\hat{\underline{\lambda}} = [A'A]^{-1}A'\underline{y} \neq [BA]^{-1}B\underline{y}$$

Mismatched Backprojector $B \neq A'$ (3D PET)

$\underline{\lambda}$	$\hat{\underline{\lambda}}$ (PWLS-CG)	$\hat{\underline{\lambda}}$ (PWLS-CG)



$(64 \times 64 \times 4)$	Matched	Mismatched



Horizontal Profiles
1.2
1
0.8
0.6
0.4
0.2
0
−0.2
0
10
20
30
40
50
60
70
x
$\hat{\lambda}(\vec{x})$
Matched
Mismatched

Choice 3. Statistical Models

After modeling the system physics, we have a deterministic "model:"

$$\underline{Y} \approx E[\underline{Y}] = A\underline{\lambda} + \underline{r}.$$

Statistical modeling is concerned with the " $\approx$ " aspect.

Random Phenomena

- Number of tracer atoms injected N
- Spatial locations of tracer atoms $\{\vec{X}_k\}_{k=1}^N$
- Time of decay of tracer atoms $\{T_k\}_{k=1}^N$
- Positron range
- Emission angle
- Photon absorption
- Compton scatter
- Detection $S_k \neq 0$
- Detector unit $\{S_k\}_{i=1}^{n_d}$
- Random coincidences
- Deadtime losses
- ...

Statistical Model Considerations

- More accurate models:
  - can lead to lower variance images,
  - can reduce bias
  - may incur additional computation,
  - may involve additional algorithm complexity
  (*e.g.* proper transmission Poisson model has nonconcave log-likelihood)
- Statistical model errors (*e.g.* deadtime)
- Incorrect models (*e.g.* log-processed transmission data)

Statistical Model Choices

- "None." Assume $\underline{Y} - \underline{r} = A\underline{\lambda}$. "Solve algebraically" to find $\underline{\lambda}$.
- White Gaussian noise. Ordinary least squares: minimize $\|Y - A\underline{\lambda}\|^2$
- Non-White Gaussian noise. Weighted least squares: minimize

$$\|Y - A\underline{\lambda}\|_W^2 = \sum_{i=1}^{n_d} w_i \left(y_i - [A\underline{\lambda}]_i\right)^2, \quad \text{where} \quad [A\underline{\lambda}]_i \triangleq \sum_{j=1}^{n_p} a_{ij}\lambda_j$$

- Ordinary Poisson model (ignoring or precorrecting for background)

$$Y_i \sim \text{Poisson}\{[A\underline{\lambda}]_i\}$$

- Poisson model

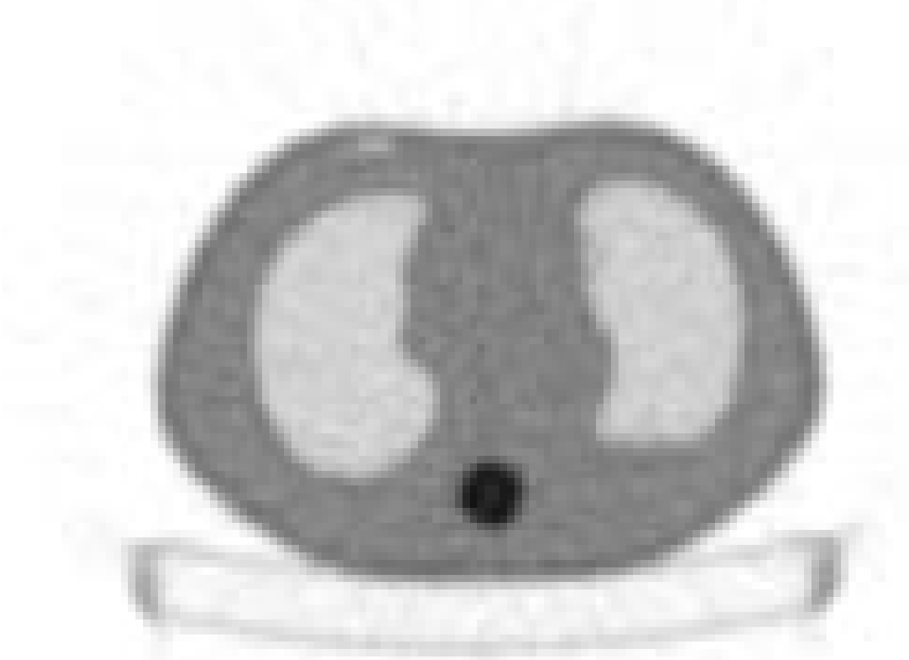
$$Y_i \sim \text{Poisson}\{[A\underline{\lambda}]_i + r_i\}$$

- Shifted Poisson model (for randoms precorrected PET)

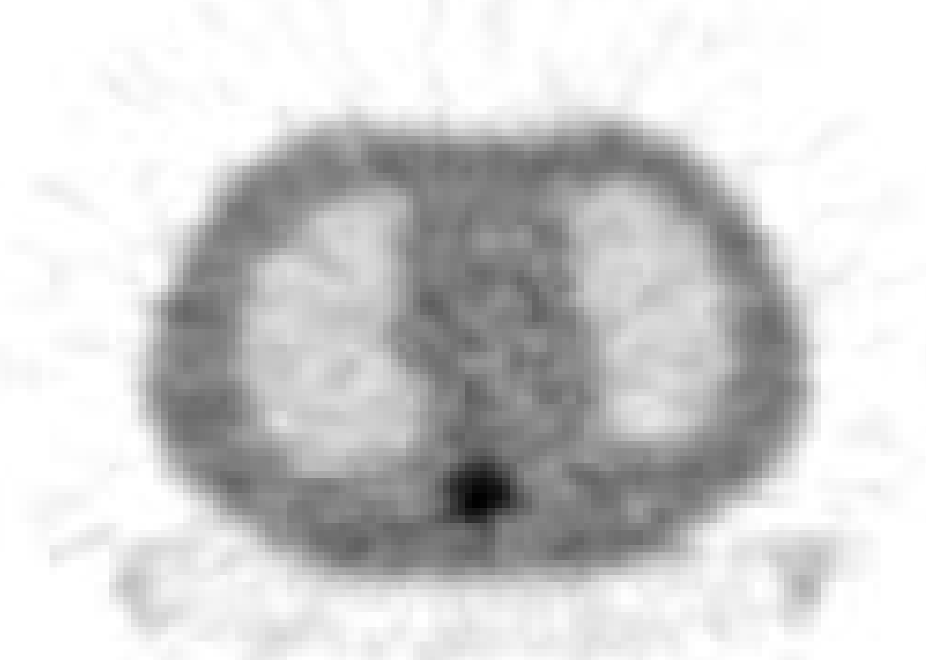
$$Y_i = Y_i^{\text{prompt}} - Y_i^{\text{delay}} \sim \text{Poisson}\{[A\underline{\lambda}]_i + 2r_i\} - 2r_i$$

Transmission Phantom

FBP 7hour



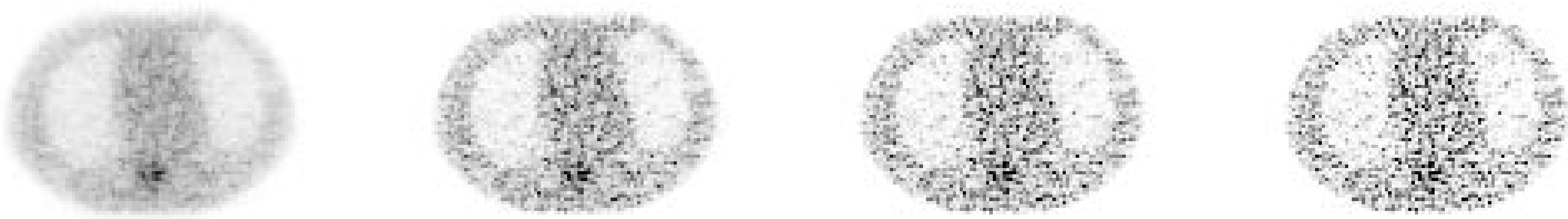
FBP 12min



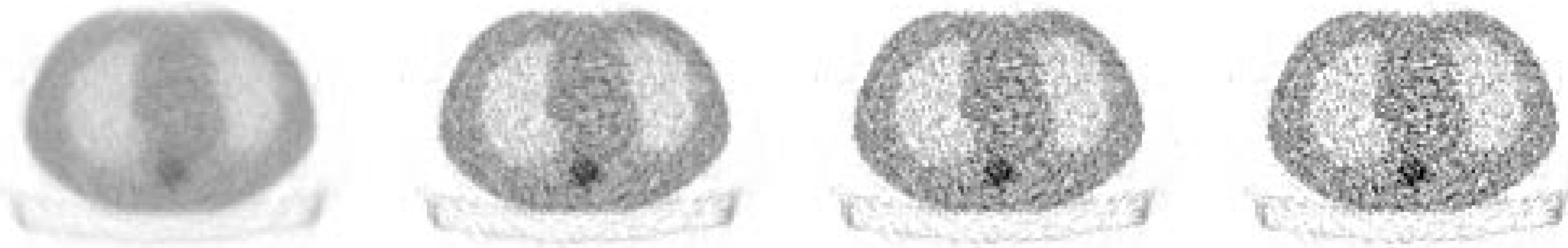
Thorax Phantom
ECAT EXACT

Effect of statistical model

OSEM



OSTR



Iteration: 1 3 5 7

Choice 4. Objective Functions

Components:

- *Data-fit* term
- *Regularization* term (and regularization parameter β)
- Constraints (*e.g.* nonnegativity)

$$\Phi(\underline{\lambda}) = \text{DataFit}(\underline{Y}, A\underline{\lambda} + \underline{r}) - \beta \cdot \text{Roughness}(\underline{\lambda})$$

$$\boxed{\hat{\underline{\lambda}} \stackrel{\triangle}{=} \arg\max_{\underline{\lambda} \geq \underline{0}} \Phi(\underline{\lambda})}$$

“Find the image that ‘best fits’ the sinogram data”

Actually *three* choices to make for Choice 4 ...

Distinguishes “statistical methods” from “algebraic methods” for “$\underline{Y} = A\underline{\lambda}$.”

Why Objective Functions?

(vs "procedure" *e.g.* adaptive neural net with wavelet denoising)

Theoretical reasons

ML is based on maximizing an objective function: the log-likelihood

- ML is asymptotically consistent
- ML is asymptotically unbiased
- ML is asymptotically efficient (under true statistical model...)
- Penalized-likelihood achieves uniform CR bound asymptotically

Practical reasons

- Stability of estimates (if Φ and algorithm chosen properly)
- Predictability of properties (despite nonlinearities)
- Empirical evidence (?)

Choice 4.1: Data-Fit Term

- Least squares, weighted least squares (quadratic data-fit terms)
- Reweighted least-squares
- Model-weighted least-squares
- Norms robust to outliers
- Log-likelihood of statistical model. Poisson case:

$$L(\underline{\lambda};\underline{Y}) = \log P[\underline{Y} = \underline{y};\underline{\lambda}] = \sum_{i=1}^{n_d} y_i \log([A\underline{\lambda}]_i + r_i) - ([A\underline{\lambda}]_i + r_i) - \log y_i!$$

Poisson probability mass function (PMF):

$$P[\underline{Y} = \underline{y};\underline{\lambda}] = \prod_{i=1}^{n_d} e^{-\bar{y}_i} \bar{y}_i^{y_i} / y_i! \quad \text{where} \quad \underline{\bar{y}} \triangleq A\underline{\lambda} + \underline{r}$$

Considerations

- Faithfulness to statistical model vs computation
- Effect of statistical modeling errors

Choice 4.2: Regularization

Forcing too much “data fit” gives noisy images

Ill-conditioned problems: small data noise causes large image noise

Solutions:

- **Noise-reduction methods**
  - Modify the *data* (prefilter or extrapolate sinogram data)
  - Modify an *algorithm* derived for an ill-conditioned problem (stop before converging, post-filter)
- **True regularization methods**

  Redefine the *problem* to eliminate ill-conditioning
  - Use bigger pixels (fewer basis functions)
  - Method of sieves (constrain image roughness)
  - Change objective function by adding a roughness penalty / prior

$$R(\underline{\lambda}) = \sum_{j=1}^{n_p} \sum_{k \in N_j} \psi(\lambda_j - \lambda_k)$$

Noise-Reduction vs True Regularization

Advantages of **“noise-reduction”** methods

- Simplicity (?)
- Familiarity
- Appear less subjective than using penalty functions or priors
- Only fiddle factors are # of iterations, amount of smoothing
- Resolution/noise tradeoff usually varies with iteration
  (stop when image looks good - in principle)

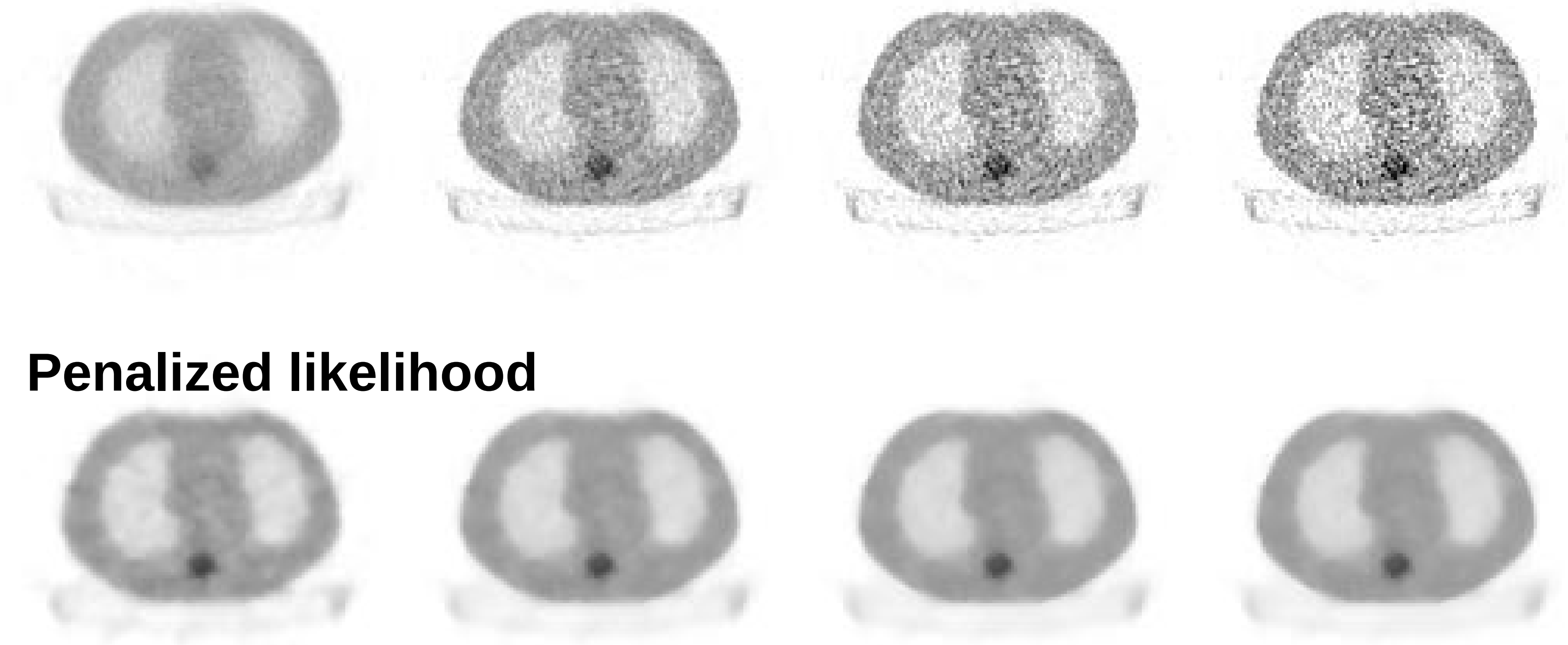
Advantages of **true regularization** methods

- Stability
- Predictability
- Resolution can be made object independent
- Controlled resolution (*e.g.* spatially uniform, edge preserving)
- Start with (*e.g.*) FBP image $\Rightarrow$ reach solution faster.

Unregularized vs Regularized Reconstruction

ML (unregularized) **(OSTR)**

Penalized likelihood



Iteration: 1 3 5 7

Roughness Penalty Function Considerations

$$R(\underline{\lambda}) = \sum_{j=1}^{n_p} \sum_{k \in N_j} \psi(\lambda_j - \lambda_k)$$

- Computation
- Algorithm complexity
- Uniqueness of maximum of Φ
- Resolution properties (edge preserving?)
- # of adjustable parameters
- Predictability of properties (resolution and noise)

Choices

- separable vs nonseparable
- quadratic vs nonquadratic
- convex vs nonconvex

This topic is actively debated!

Nonseparable Penalty Function Example

x_1	x_2	x_3
x_4	x_5	

Example

$$R(\underline{x}) = (x_2 - x_1)^2 + (x_3 - x_2)^2 + (x_5 - x_4)^2 + (x_4 - x_1)^2 + (x_5 - x_2)^2$$

2	2	2
2	1	

$R(\underline{x}) = 1$

3	3	1
2	2	

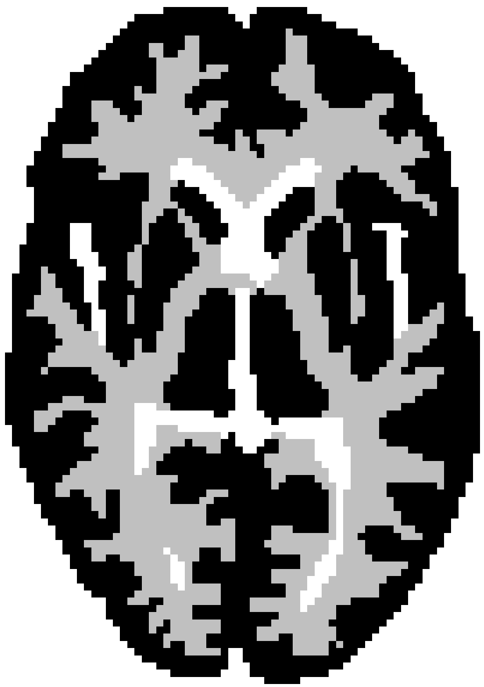
$R(\underline{x}) = 6$

1	3	1
2	2	

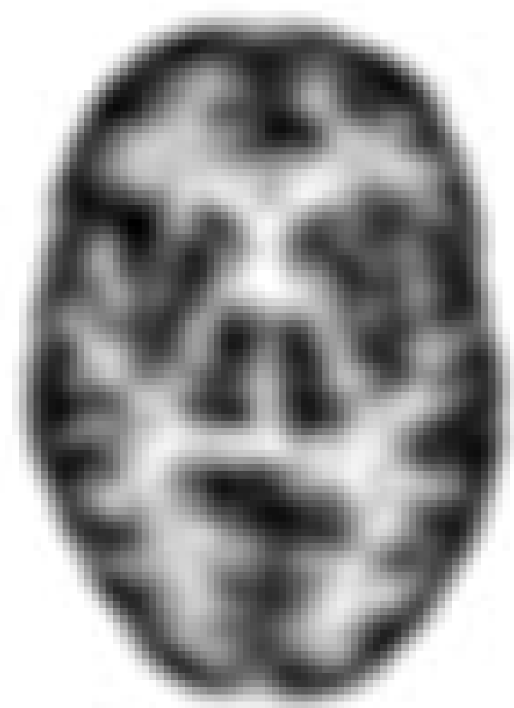
$R(\underline{x}) = 10$

Rougher images $\Rightarrow$ greater $R(\underline{x})$

Penalty Functions: Quadratic vs Nonquadratic



Phantom



Quadratic Penalty

Huber Penalty

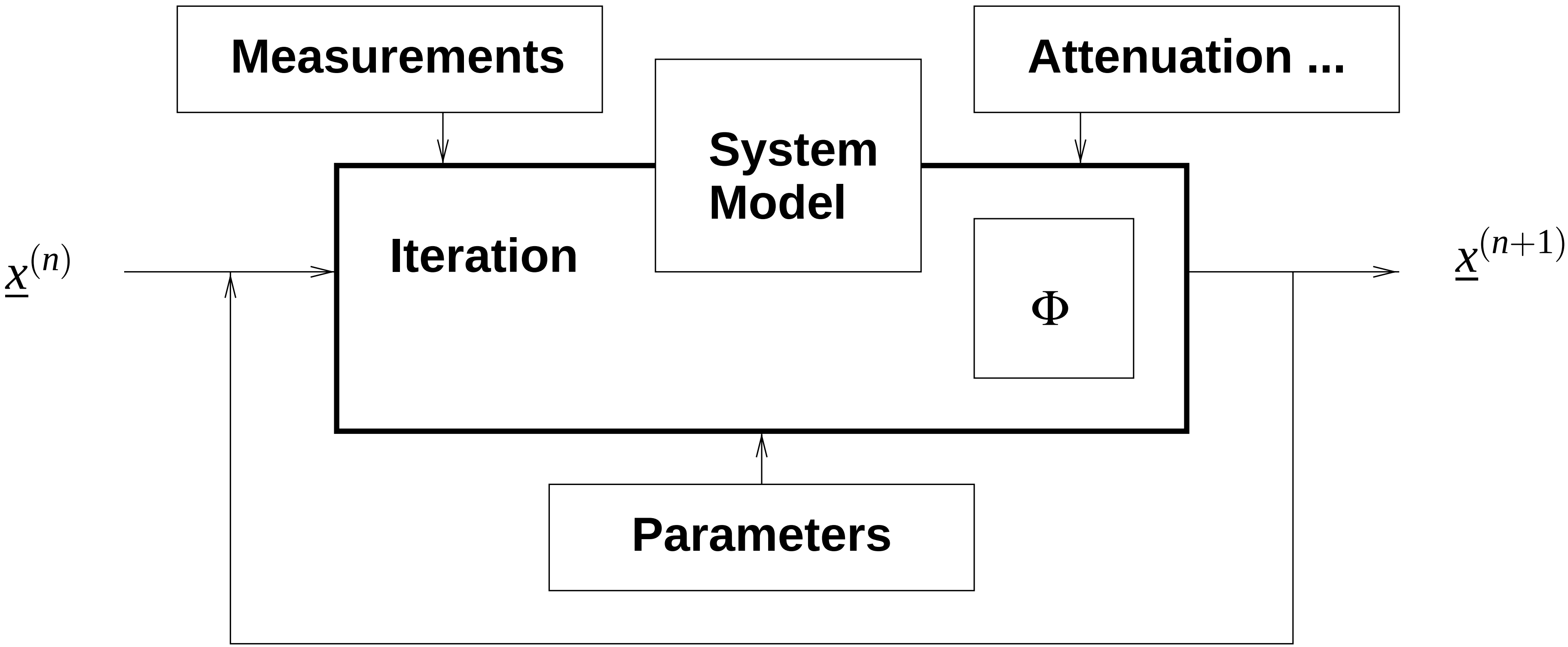
Summary of Modeling Choices

1. Object parameterization: $\lambda(\underline{x})$ vs $\underline{\lambda}$
2. System physical model: $s_i(\underline{x})$
3. Measurement statistical model $Y_i \sim \boxed{?}$
4. Objective function: data-fit / regularization / constraints

Reconstruction Method = Objective Function + Algorithm

5. Iterative algorithm
   ML-EM, MAP-OSL, PL-SAGE, PWLS+SOR, PWLS-CG, . . .

Choice 5. Algorithms



Deterministic iterative mapping: $\underline{x}^{(n+1)} = M\left(\underline{x}^{(n)}\right)$

All algorithms are imperfect. No single best solution.

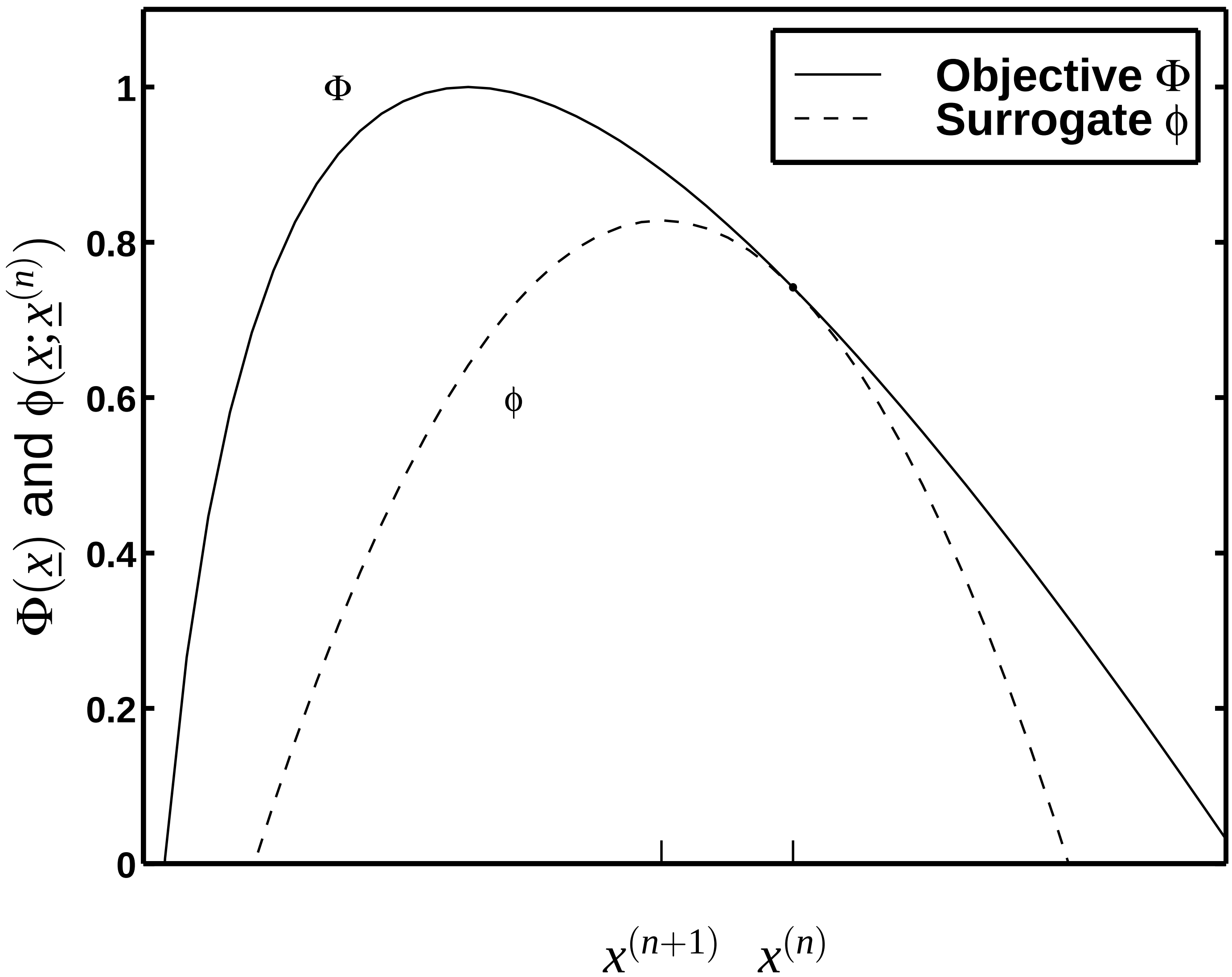
Ideal Algorithm

$$\underline{x}^\star \triangleq \arg\max_{\underline{x} \geq \underline{0}} \Phi(\underline{x}) \quad \text{(global maximum)}$$

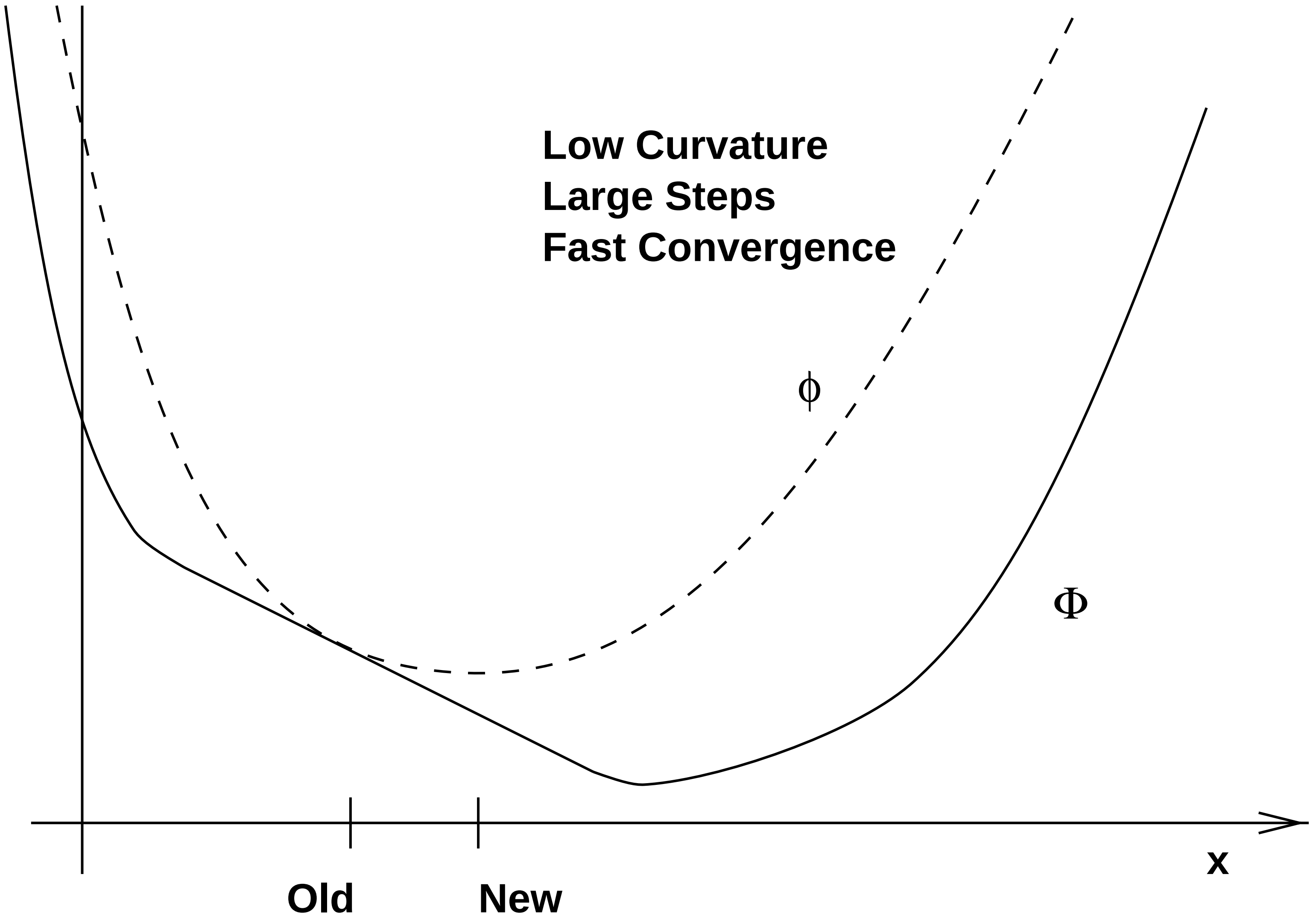
stable and convergent	$\{\underline{x}^{(n)}\}$ converges to $\underline{x}^\star$ if run indefinitely
converges quickly	$\{\underline{x}^{(n)}\}$ gets "close" to $\underline{x}^\star$ in just a few iterations
globally convergent	$\lim_n \underline{x}^{(n)}$ independent of starting image
fast	requires minimal computation per iteration
robust	insensitive to finite numerical precision
user friendly	nothing to adjust (*e.g.* acceleration factors)
monotonic	$\Phi(\underline{x}^{(n)})$ increases every iteration
parallelizable	(when necessary)
simple	easy to program and debug
flexible	accommodates any type of system model (matrix stored by row or column or projector/backprojector)

Choices: forgo one or more of the above

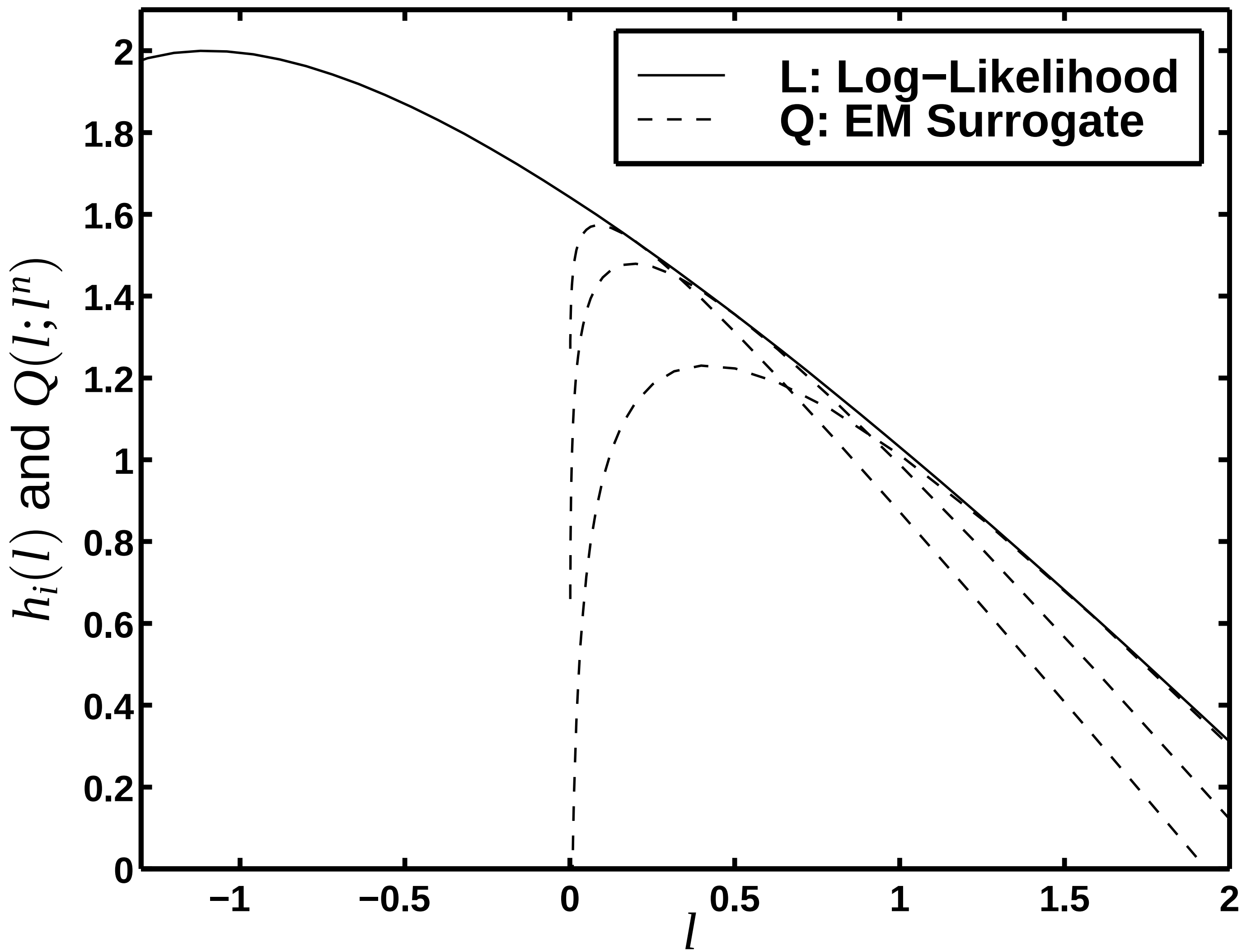
Optimization Transfer Illustrated



Convergence Rate: Fast



Slow Convergence of EM



Paraboloidal Surrogates

- Not separable (unlike EM)
- Not self-similar (unlike EM)
- Poisson log-likelihood replaced by a series of least squares problems.
- Maximize each quadratic problem easily using coordinate ascent.

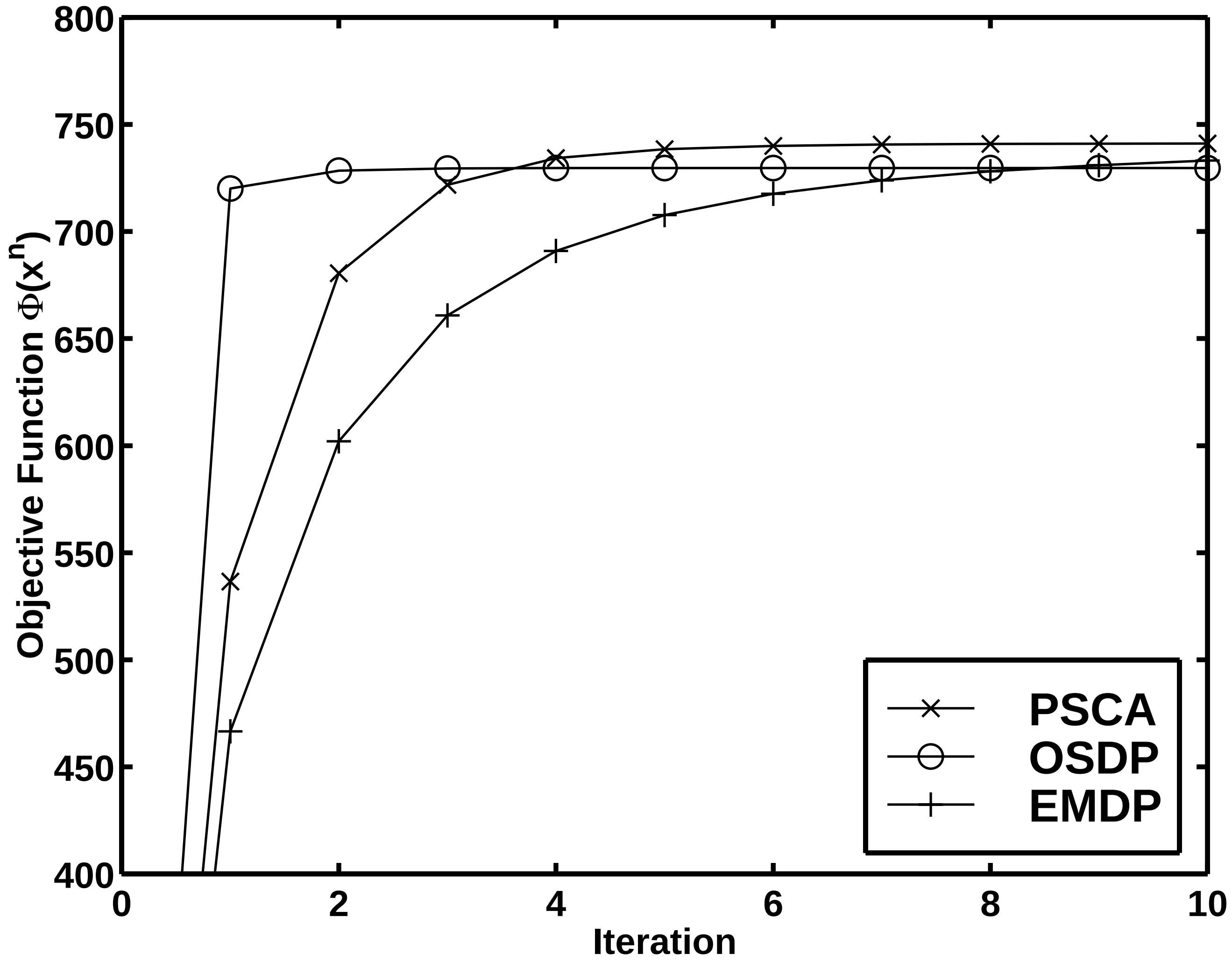
Advantages

- Fast converging
- Instrinsically monotone global convergence
- Fairly simple to derive / implement
- Nonnegativity easy (with coordinate ascent)

Disadvantages

- Coordinate ascent $\therefore$ column-stored system matrix

Convergence rate: PSCA vs EM



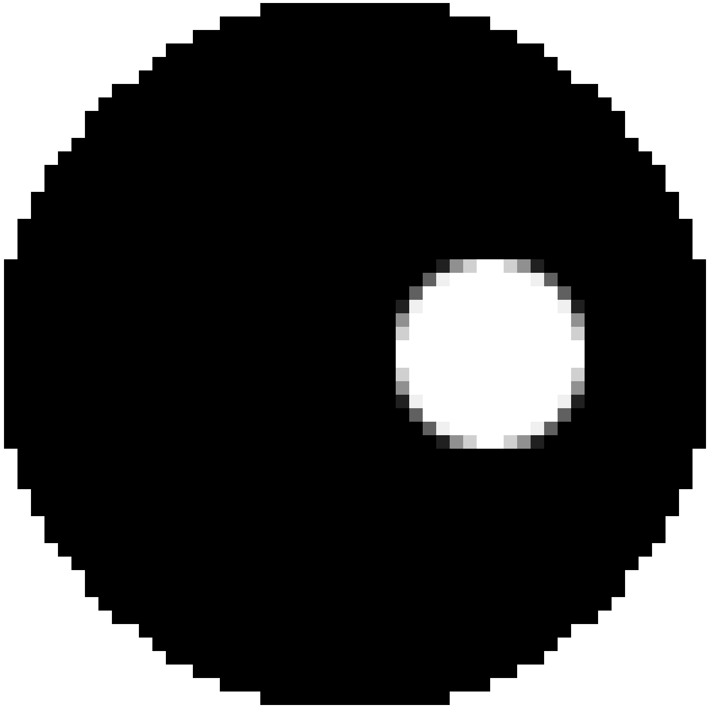
Ordered Subsets Algorithms

- The *backprojection* operation appears in every algorithm.
- Intuition: with half the angular sampling, the backprojection would look fairly similar.
- To "OS-ize" an algorithm, replace all backprojections with partial sums.

Problems with OS-EM

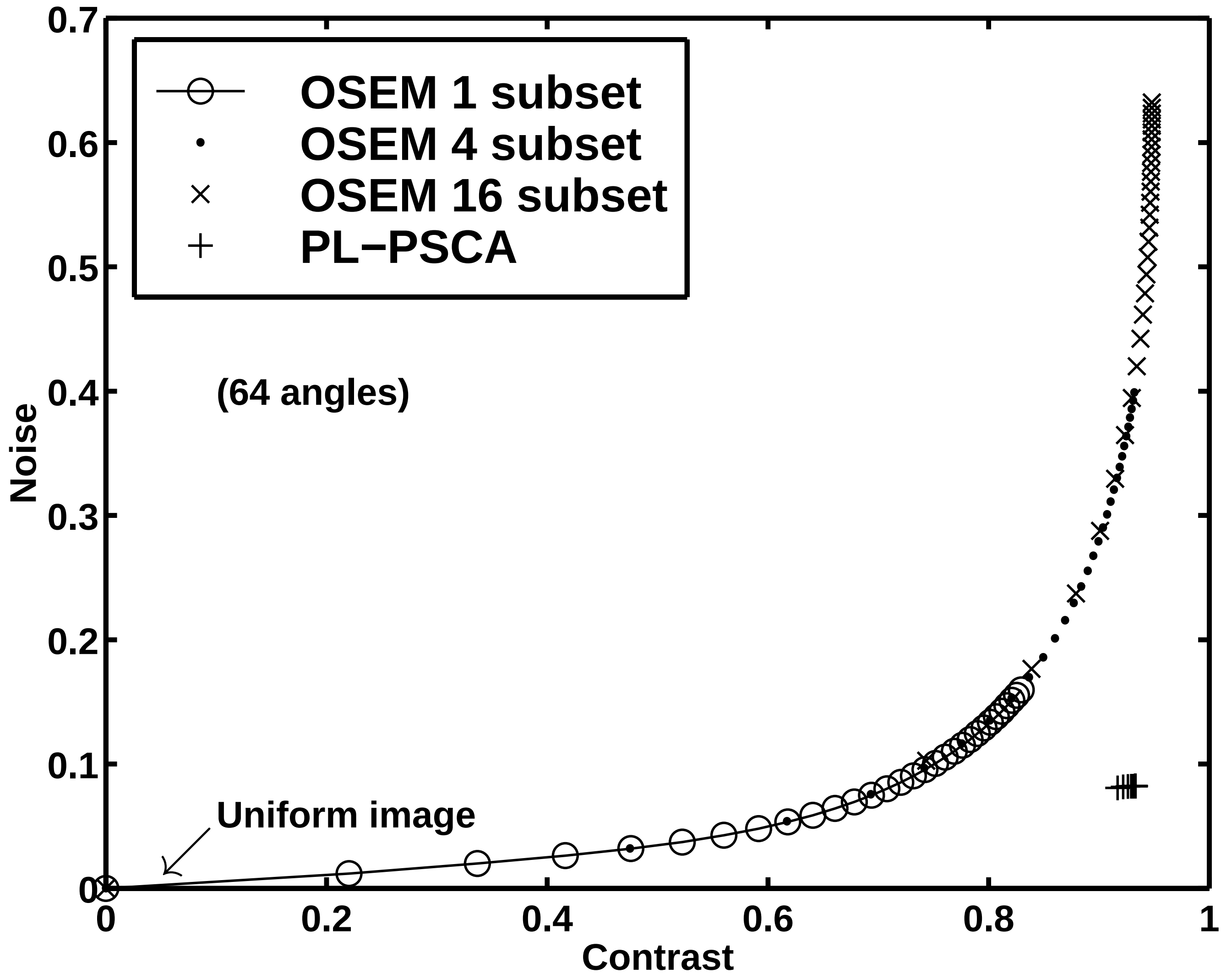
- Non-monotone
- Does not converge (may cycle)
- Byrne's RBBI approach only converges for consistent (noiseless) data
- $\therefore$ unpredictable
  - What resolution after n iterations?
  - Object-dependent, spatially nonuniform
  - What variance after n iterations?
  - ROI variance? (*e.g.* for Huesman's WLS kinetics)

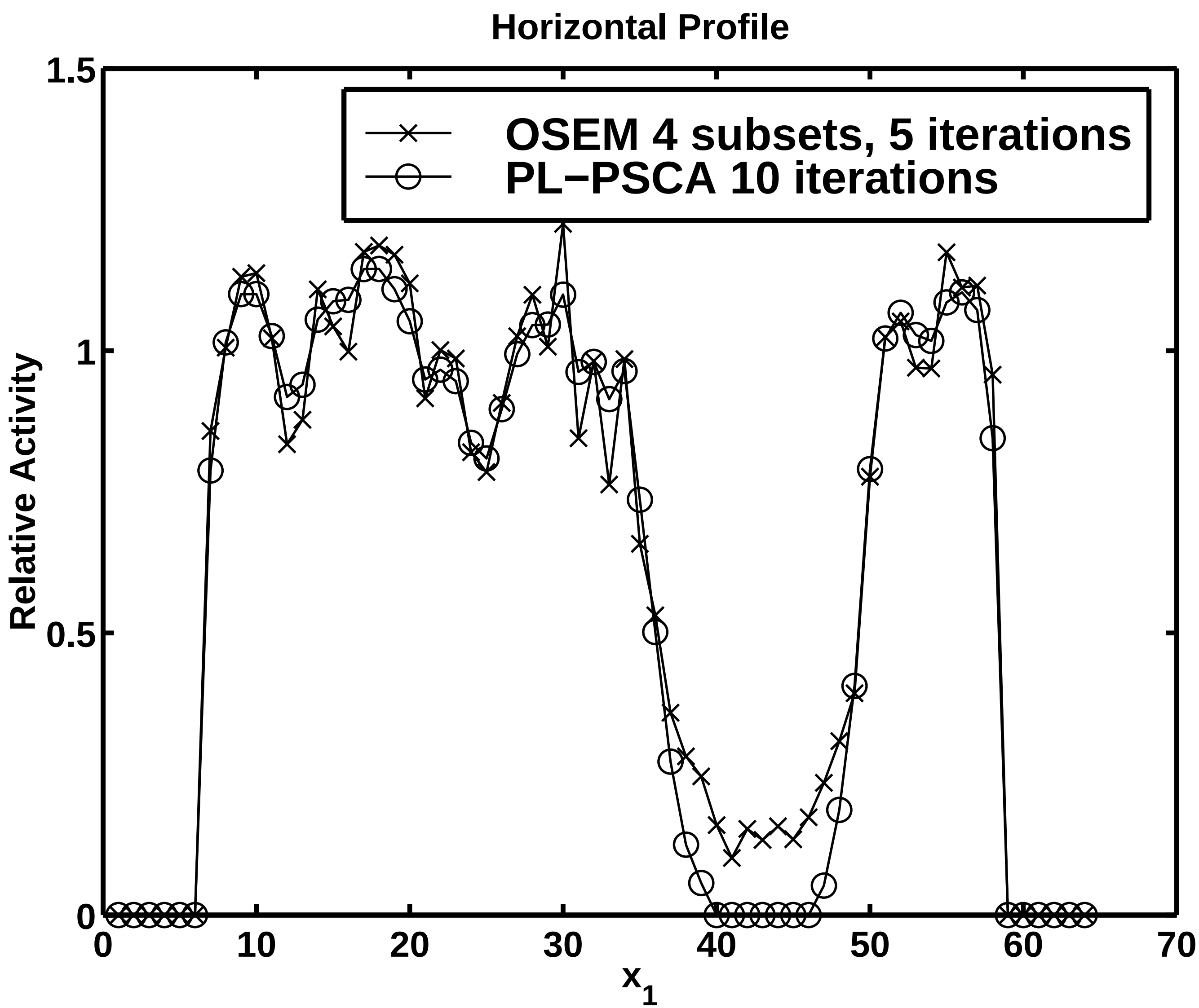
OSEM vs Penalized Likelihood



- 64×62 image
- 66×60 sinogram
- 10^6 counts
- 15% randoms/scatter
- uniform attenuation
- contrast in cold region
- within-region σ opposite side

Contrast-Noise Results




Horizontal Profile
OSEM 4 subsets, 5 iterations
PL−PSCA 10 iterations
Relative Activity
1.5
1
0.5
0
0
10
20
30
40
50
60
70
x_1

Noise Properties

$$\mathrm{Cov}\{\hat{\underline{x}}\} \approx \left[\nabla^{20}\Phi\right]^{-1} \left[\nabla^{11}\Phi\right] \mathrm{Cov}\{\underline{Y}\} \left[\nabla^{11}\Phi\right]^{T} \left[\nabla^{20}\Phi\right]^{-1}$$

- Enables prediction of noise properties
- Useful for computing ROI variance for kinetic fitting

IEEE Tr. Image Processing, 5(3):493 1996

Summary

- General principles of statistical image reconstruction
- Optimization transfer
- Principles apply to transmission reconstruction
- Predictability of resolution / noise and controlling spatial resolution argues for regularized objective-function
- Still work to be done...

An Open Problem

Still no algorithm with all of the following properties:

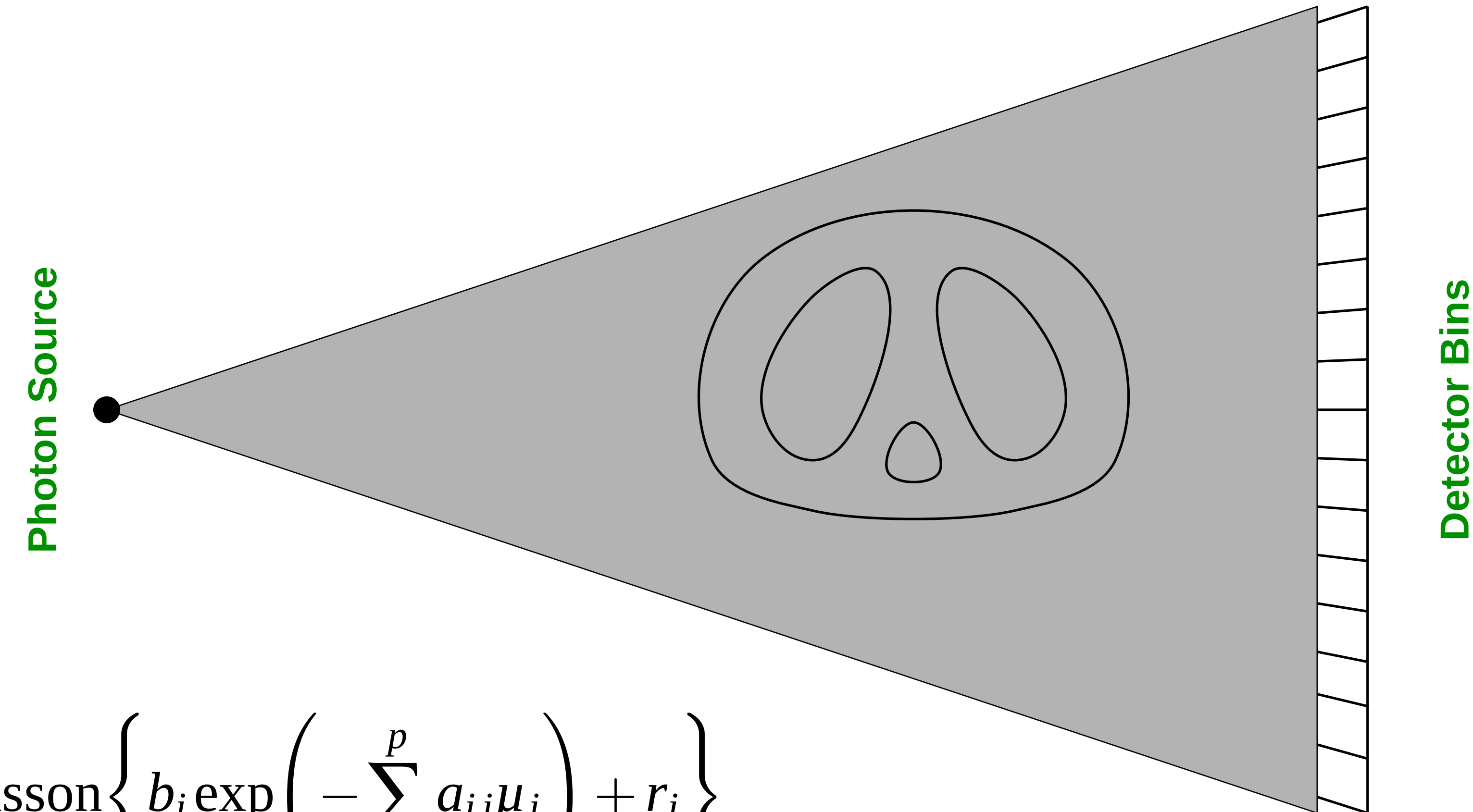
- Nonnegativity easy
- Fast converging
- Intrinsically monotone global convergence
- Accepts any type of system matrix
- Parallelizable

Fast Maximum Likelihood Transmission Reconstruction using Ordered Subsets

Jeffrey A. Fessler, Hakan Erdoğan

EECS Department, BME Department, and
Nuclear Medicine Division of Dept. of Internal Medicine
The University of Michigan

Transmission Scans



$$Y_i \sim \text{Poisson}\left\{ b_i \exp\left(-\sum_{j=1}^{p} a_{ij}\mu_j \right) + r_i \right\}$$

Each measurement Y_i is related to a single “line integral” through the object.

Transmission Scan Statistical Model

$$Y_i \sim \text{Poisson}\left\{ b_i \exp\left(-\sum_{j=1}^{p} a_{ij} \mu_j \right) + r_i \right\}, \; i = 1, \ldots, N$$

- N number of detector elements
- Y_i recorded counts by ith detector element
- b_i blank scan value for ith detector element
- a_{ij} length of intersection of ith ray with jth pixel
- μ_j linear attenuation coefficient of jth pixel
- r_i contribution of room background, scatter, and emission crosstalk

(Monoenergetic case, can be generalized for dual-energy CT)
(Can be generalized for additive Gaussian detector noise)

Maximum-Likelihood Reconstruction

$$\hat{\mu} = \arg\max_{\mu \geq \underline{0}} L(\mu) \quad \text{(Log-likelihood)}$$

$$L(\mu) = \sum_{i=1}^{N} Y_i \log \left[b_i \exp\left(-\sum_{j=1}^{p} a_{ij}\mu_j \right) + r_i \right] - \left[b_i \exp\left(-\sum_{j=1}^{p} a_{ij}\mu_j \right) + r_i \right]$$

Transmission ML Reconstruction Algorithms

- Conjugate gradient

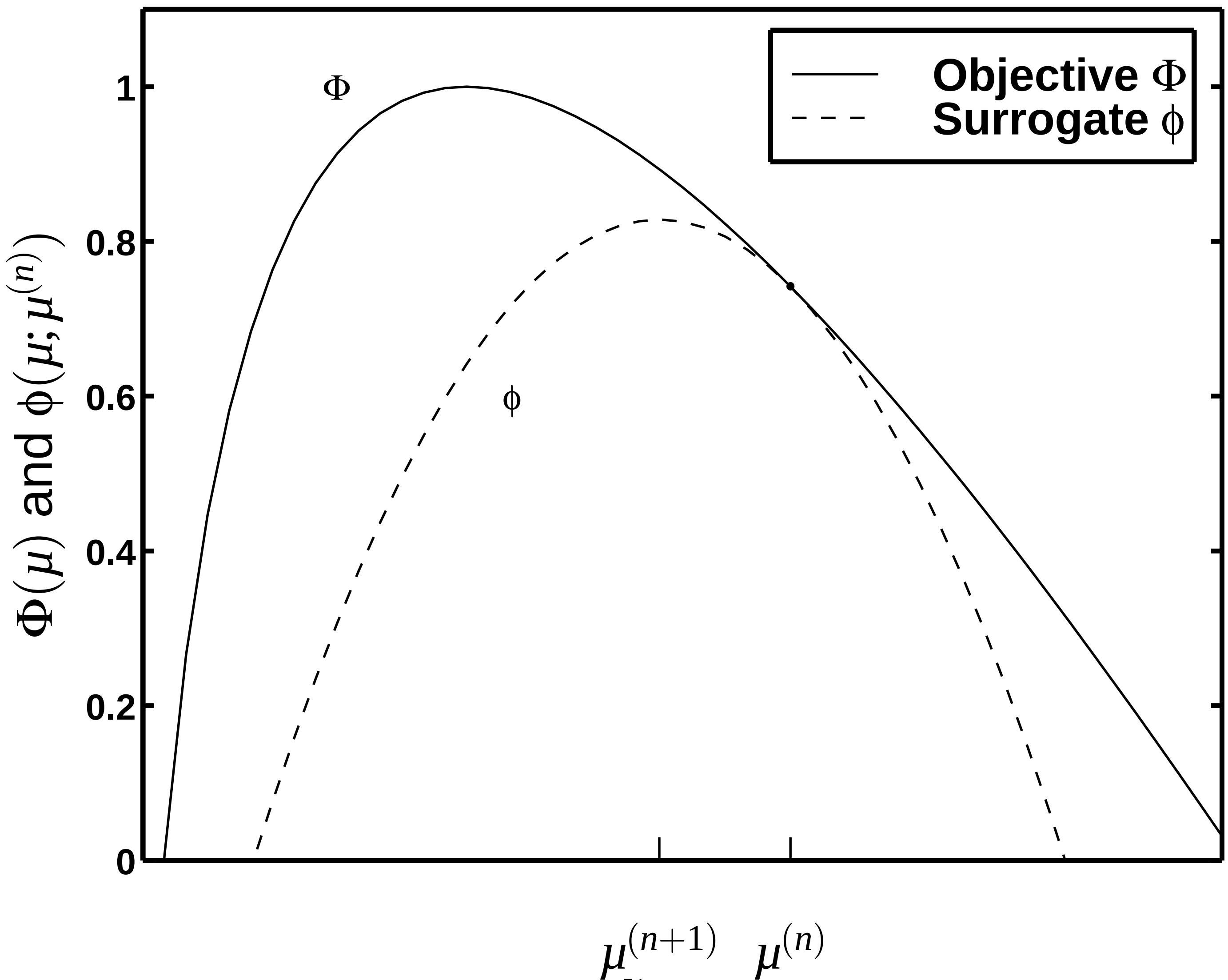
  Mumcuoğlu *et al.*, T-MI, Dec. 1994
- Paraboloidal surrogates coordinate ascent (PSCA)

  Erdoğan and Fessler, T-MI, 1999
- Ordered subsets separable paraboloidal surrogates

  Erdoğan *et al.*, PMB, Nov. 1999
- ~~Transmission expectation maximization (EM) algorithm~~

  Lange and Carson, JCAT, Apr. 1984

Optimization Transfer Illustrated



Parabola Surrogate Function

- $h(l) = y\log(be^{-l}+r) - (be^{-l}+r)$ has a parabola surrogate: $q_{im}^{(n)}$
- Optimum curvature of parabola derived by Erdoğan (T-MI, 1999)
- Replace likelihood with paraboloidal surrogate

$$L(\mu^{(n)}) = \sum_{i=1}^{N} h_i\left(\sum_{j=1}^{p} a_{ij}\mu_j\right) \geq Q_1(\mu;\mu^{(n)}) = \sum_{i=1}^{N} q_{im}^{(n)}\left(\sum_{j=1}^{p} a_{ij}\mu_j\right)$$

- $q_{im}^{(n)}$ is a simple quadratic function
- Iterative algorithm:

$$\boxed{\mu^{(n+1)} = \arg\max_{\mu \geq \underline{0}} Q_1(\mu;\mu^{(n)})}$$

- Maximizing $Q_1(\mu;\mu^{(n)})$ over μ is equivalent to (reweighted) least-squares.
- Natural algorithms
  - Conjugate gradient
  - Coordinate ascent

Separable Paraboloid Surrogate Function

- Parabolas are convex functions
- Apply De Pierro's "additive" convexity trick (T-MI, Mar. 1995)

$$\sum_{j=1}^{p} a_{ij}\mu_j = \sum_{j=1}^{p} \frac{a_{ij}}{a_i}\left[a_i(\mu_j - \mu_j^{(n)})\right] + \left[A\mu^{(n)}\right]_i \quad \text{where} \quad a_i \triangleq \sum_{j=1}^{p} a_{ij}$$

- Move summation over pixels outside quadratic

$$Q_1(\mu;\mu^{(n)}) = \sum_{i=1}^{N} q_{im}^{(n)}\left(\sum_{j=1}^{p} a_{ij}\mu_j\right)$$

$$\geq \; Q_2(\mu;\mu^{(n)}) = \sum_{i=1}^{N}\sum_{j=1}^{p} \frac{a_{ij}}{a_i} q_{im}^{(n)}\left(a_i(\mu_j - \mu_j^{(n)}) + \left[A\mu^{(n)}\right]_i\right)$$

$$= \sum_{j=1}^{p} Q_{2j}^{(n)}(\mu_j), \quad \text{where} \quad Q_{2j}^{(n)}(x) \triangleq \sum_{i=1}^{N} \frac{a_{ij}}{a_i} q_{im}^{(n)}\left(a_i(x - \mu_j^{(n)}) + \left[A\mu^{(n)}\right]_i\right)$$

- Separable paraboloidal surrogate function $\Rightarrow$ trivial to maximize (cf EM)

Iterative algorithm:

$$\mu_j^{(n+1)} = \arg\max_{\mu_j \geq 0} Q_{2j}^{(n)}(\mu_j) = \left[\mu_j^{(n)} + \frac{\frac{\partial}{\partial \mu_j} Q_{2j}^{(n)}(\mu^{(n)})}{-\frac{\partial^2}{\partial \mu_j^2} Q_{2j}^{(n)}(\mu^{(n)})}\right]_+$$

$$= \left[\mu_j^{(n)} + \frac{1}{-\frac{\partial^2}{\partial \mu_j^2} Q_{2j}^{(n)}(\mu^{(n)})} \frac{\partial}{\partial \mu_j} L(\mu^{(n)})\right]_+$$

$$= \left[\mu_j^{(n)} + \frac{\sum_{i=1}^N (y_i / \bar{y}_i^{(n)} - 1) b_i \exp\left(-\left[A\mu^{(n)}\right]_i\right)}{\sum_{i=1}^N a_{ij}^2 a_i c_i^{(n)}}\right]_+, \quad j = 1, \ldots, p$$

- $c_i^{(n)}$'s related to parabola curvatures
- Parallelizable (ideal for multiprocessor workstations)
- Monotonically increases the likelihood each iteration
- Intrinsically enforces the nonnegativity constraint
- Guaranteed to converge if unique maximizer
- Natural starting point for forming ordered-subsets variation

Ordered Subsets Algorithm

- Each $\sum_{i=1}^{N}$ is a backprojection
- Replace “full” backprojections with partial backprojections
- Partial backprojection based on angular subsampling
- Cycle through subsets of projection angles

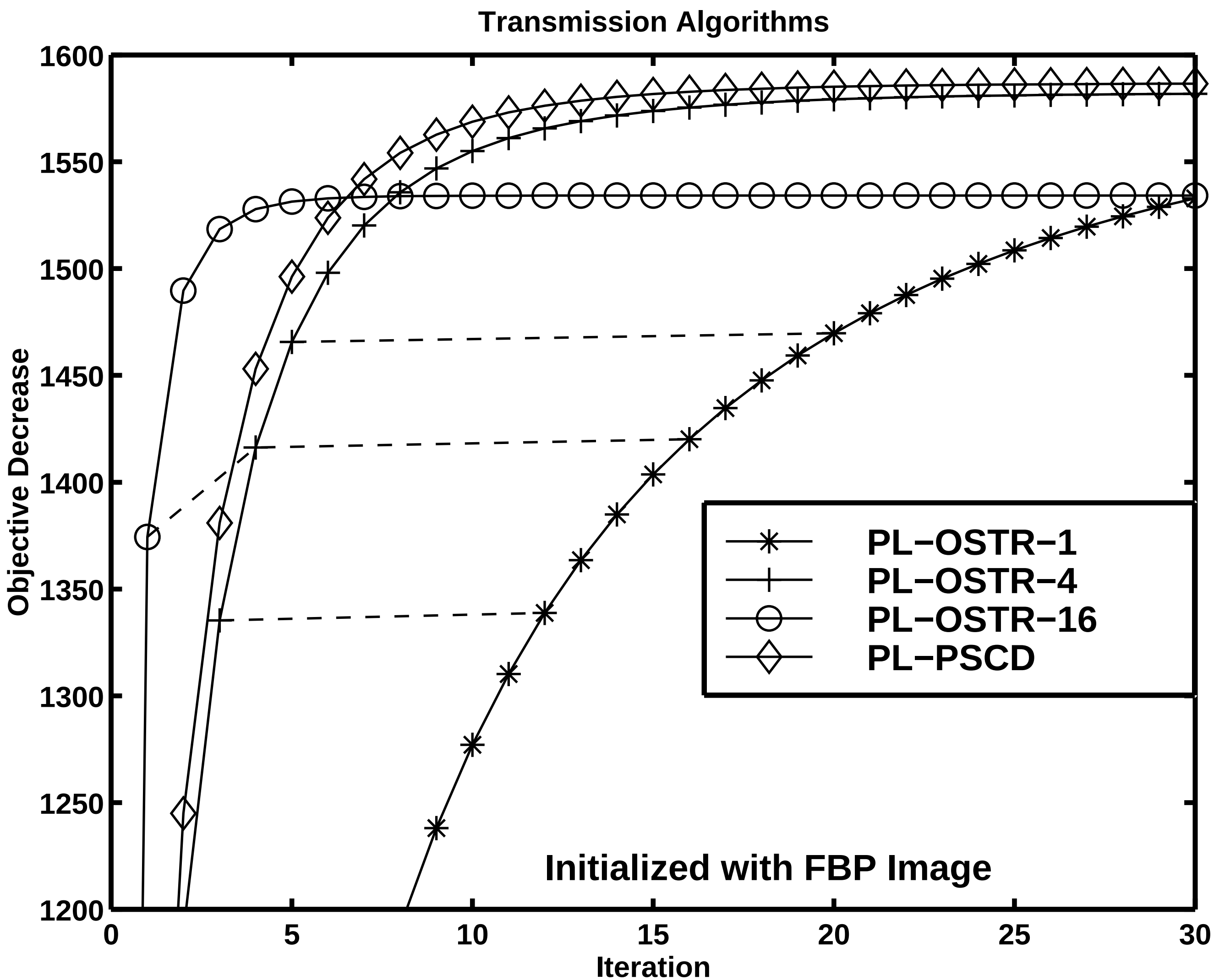
Pros
- Accelerates “convergence”
- Very simple to implement
- Reasonable images in just 1 or 2 iterations
- Regularization easily incorporated

Cons:
- Does not converge to true maximizer
- Makes analysis of properties difficult

Phantom Study

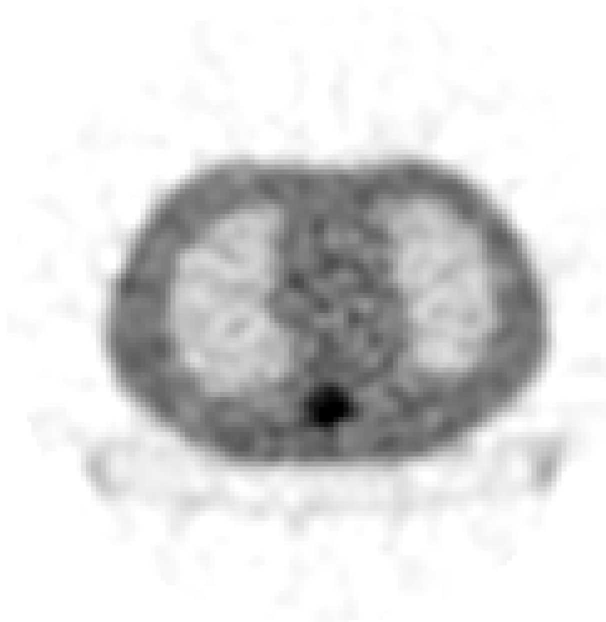
- 12-minute PET transmission scan
- Anthropomorphic thorax phantom (Data Spectrum, Chapel Hill, NC)
- Sinogram: 160 3.375mm bins by 192 angles over 180°
- Image: 128 by 128 4.2mm pixels
- Ground truth determined from 15-hour scan, FBP reconstruction / segmentation

Algorithm Convergence

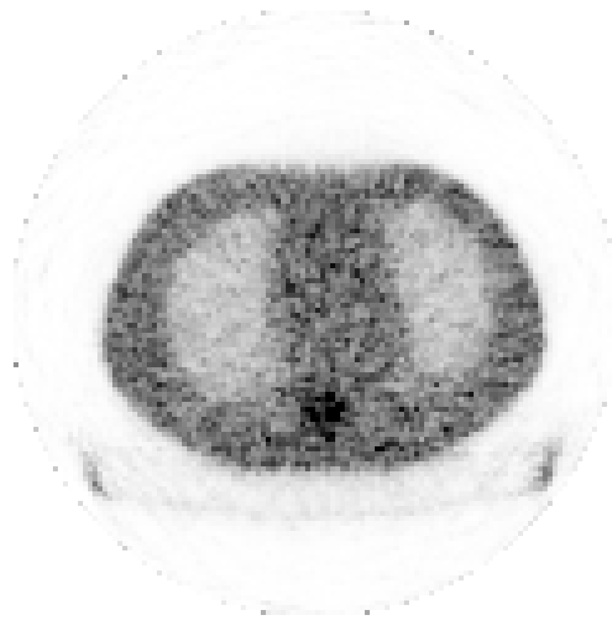


Reconstructed Images

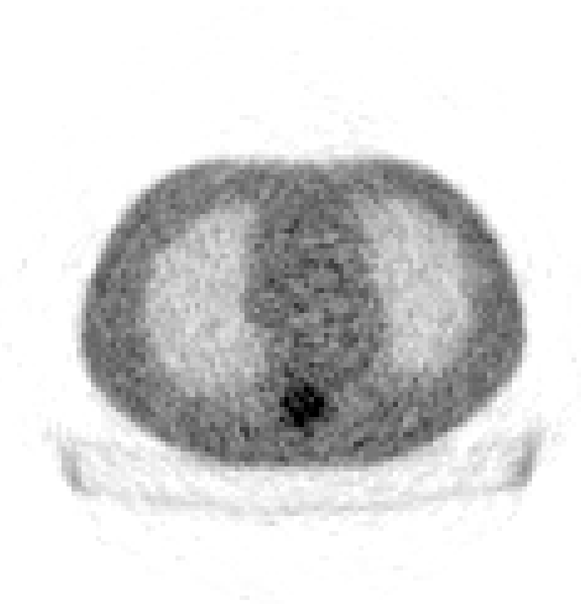
FBP



ML−OSEM−8
2 iterations

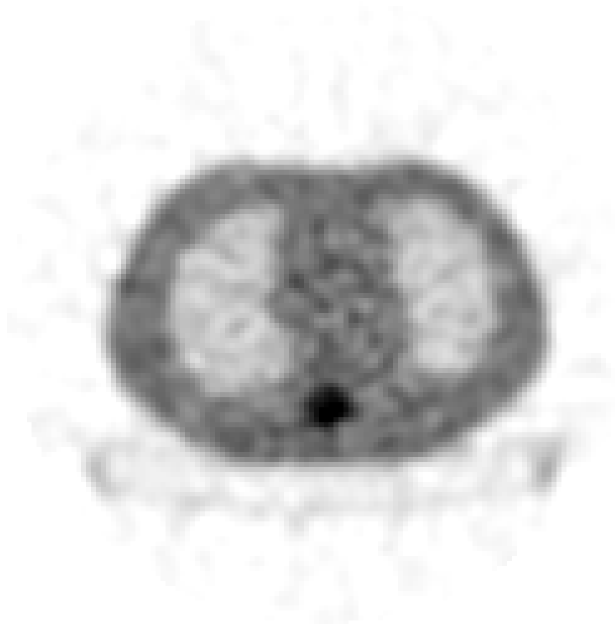


ML−OSTR−8
3 iterations

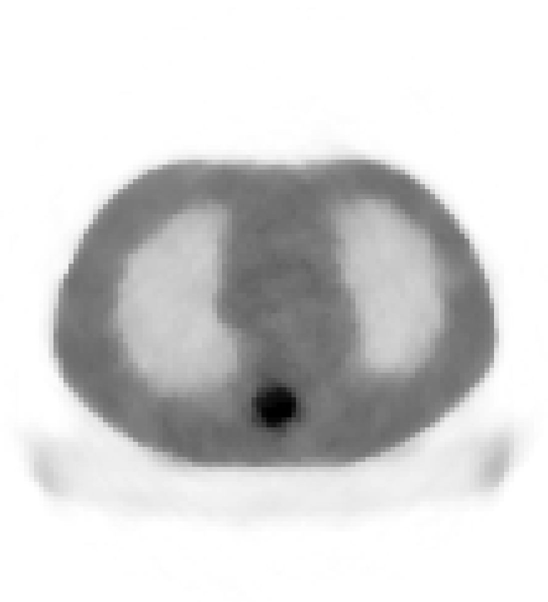


Reconstructed Images

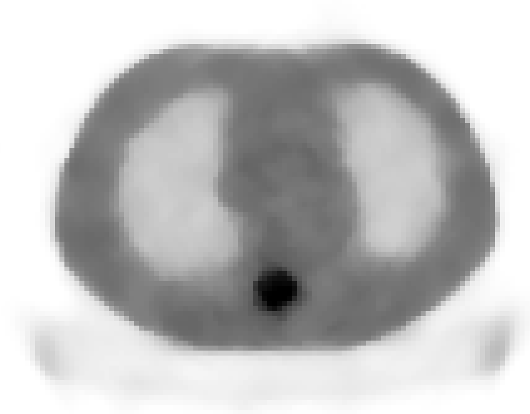
FBP



PL−OSTR−16
4 iterations

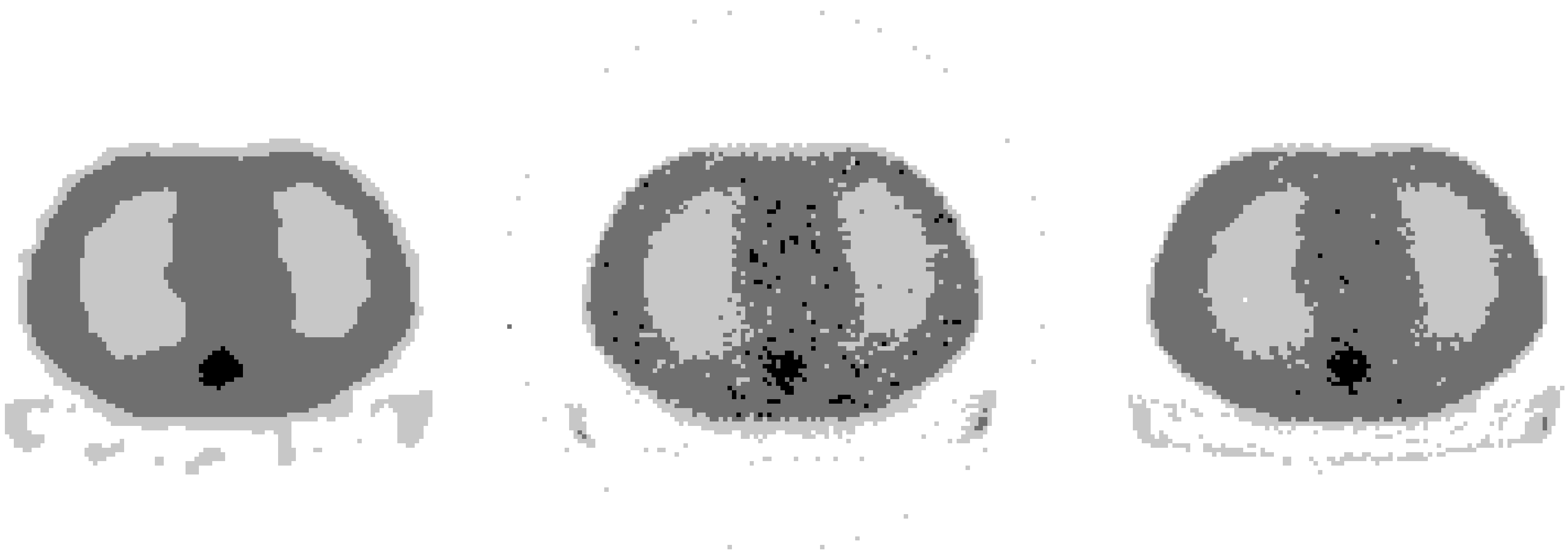


PL−PSCD
10 iterations



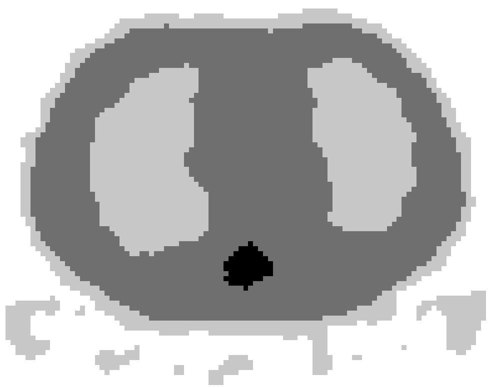
Segmented Images

FBP | **ML−OSEM−8, 2 iterations** | **ML−OSTR−8, 3 iterations**



Segmented Images

FBP

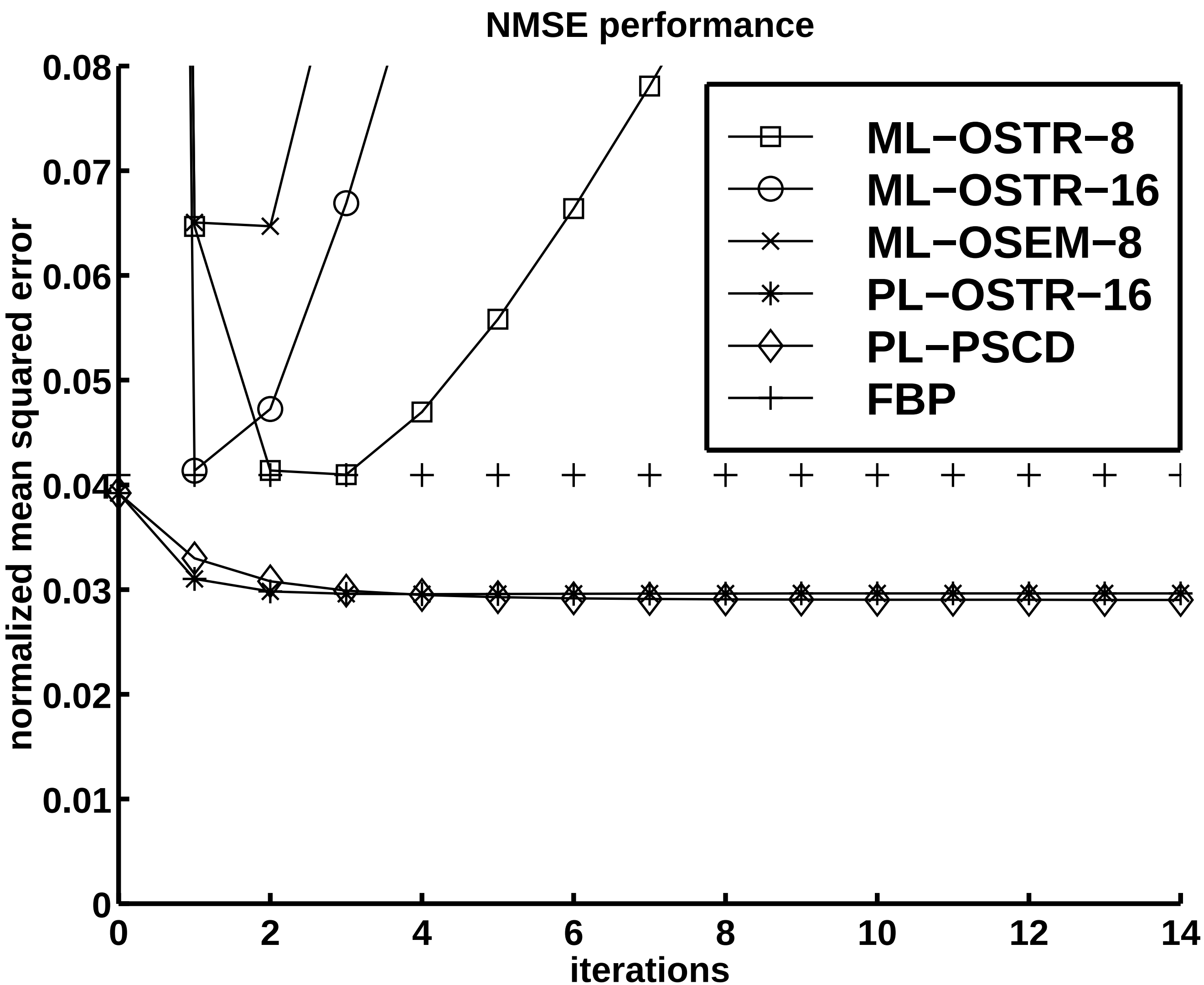


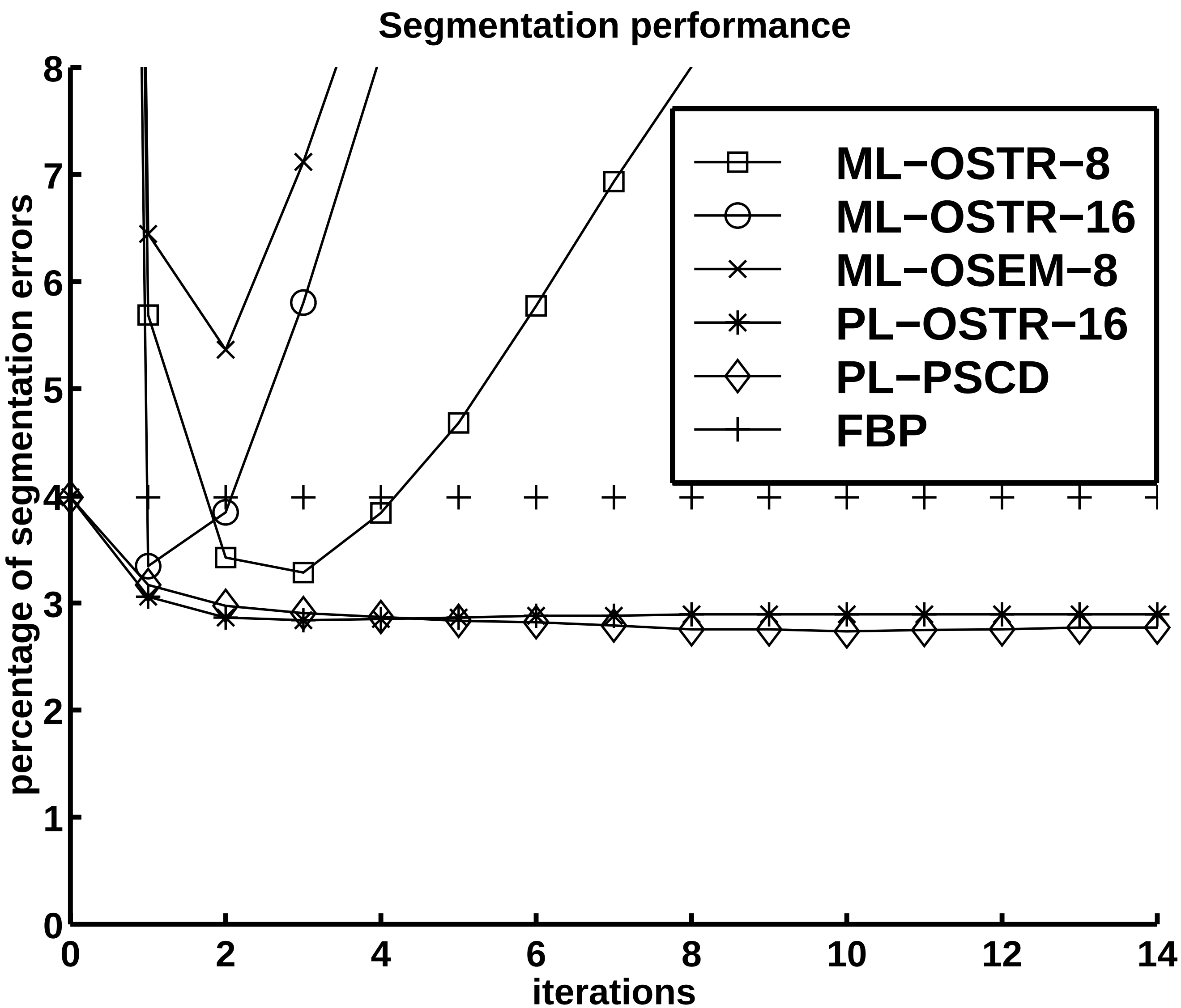
PL−OSTR−16

4 iterations

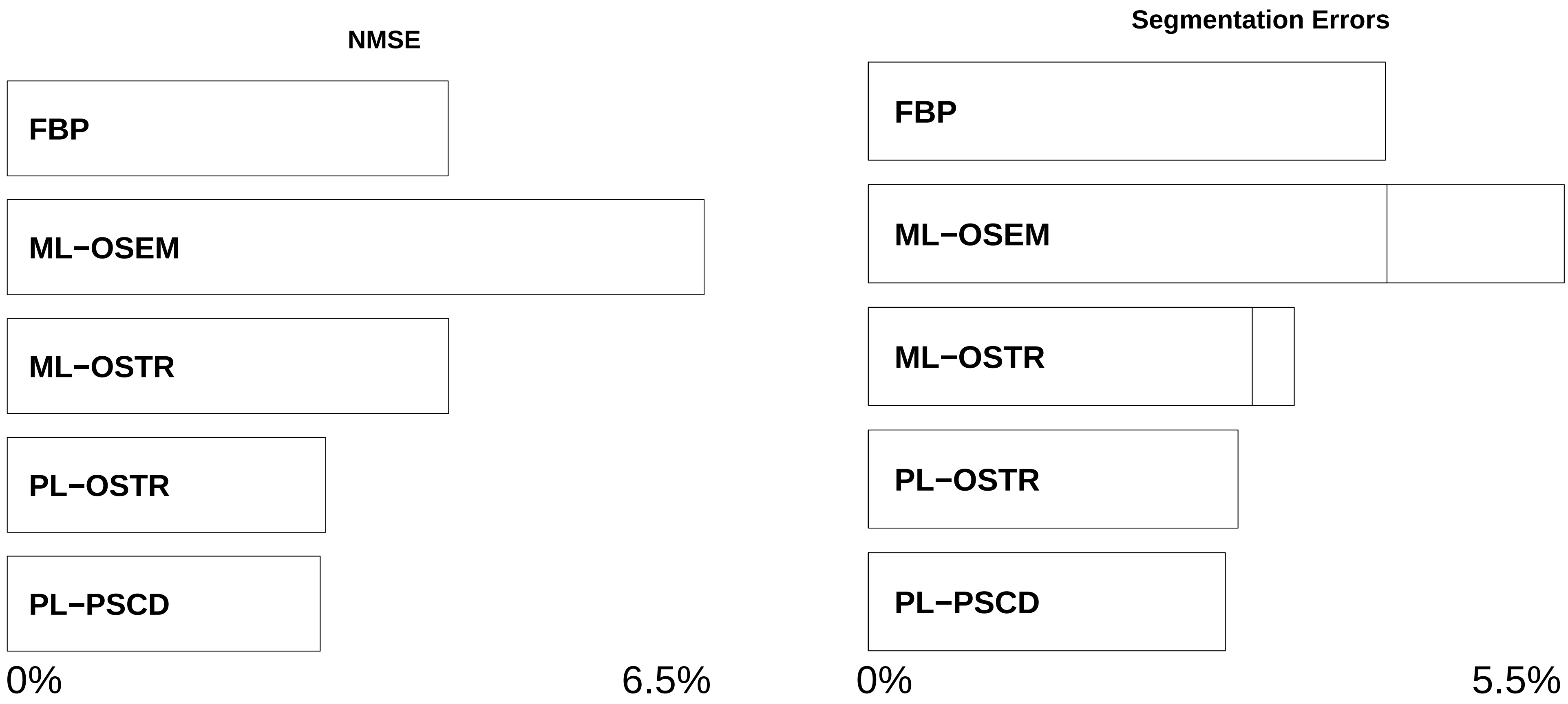
PL−PSCD

10 iterations

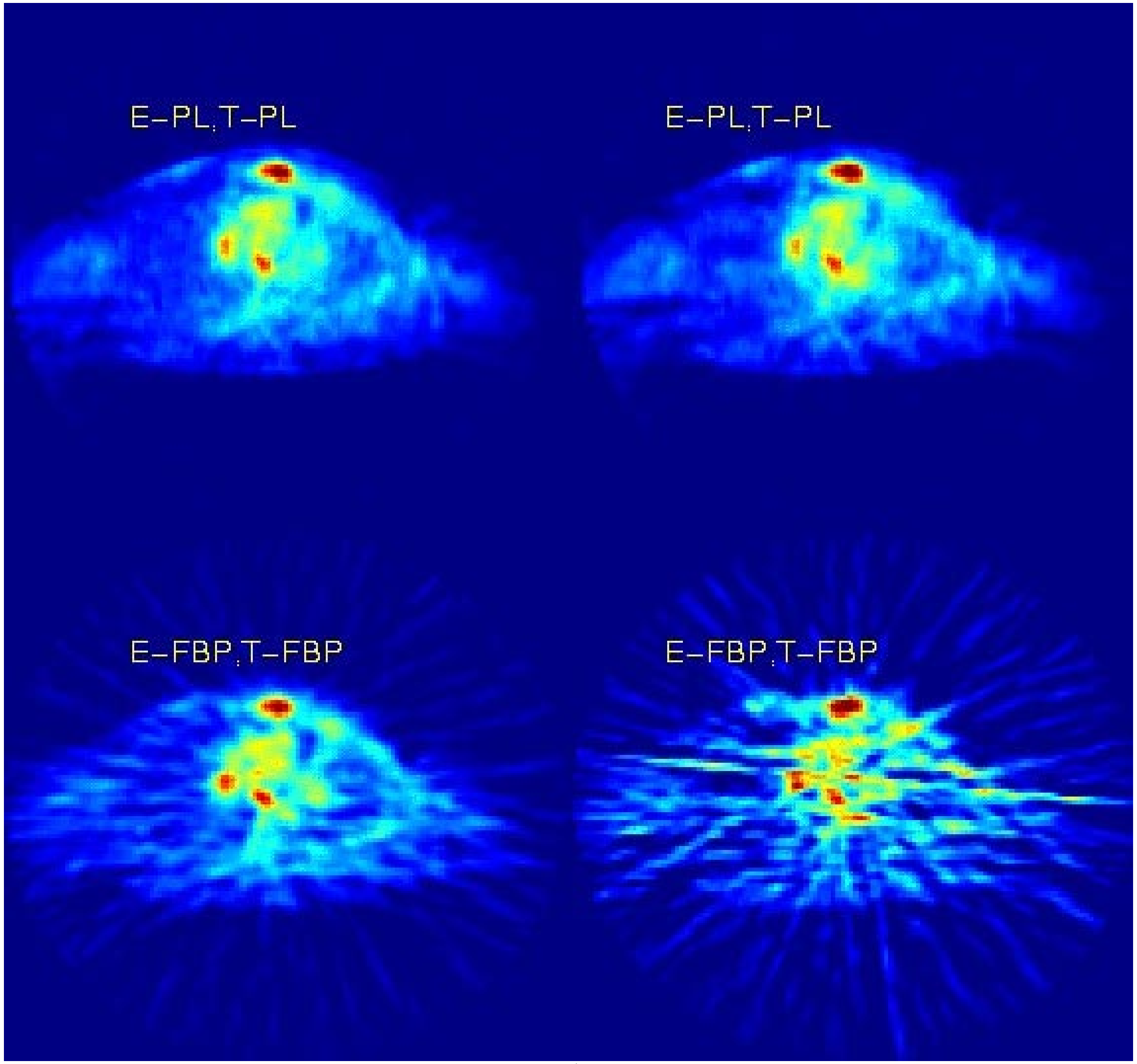

NMSE performance
normalized mean squared error
iterations
0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0
0
2
4
6
8
10
12
14
ML−OSTR−8
ML−OSTR−16
ML−OSEM−8
PL−OSTR−16
PL−PSCD
FBP


Segmentation performance
percentage of segmentation errors
iterations
0
1
2
3
4
5
6
7
8
0
2
4
6
8
10
12
14
ML−OSTR−8
ML−OSTR−16
ML−OSEM−8
PL−OSTR−16
PL−PSCD
FBP

Quantitative Results

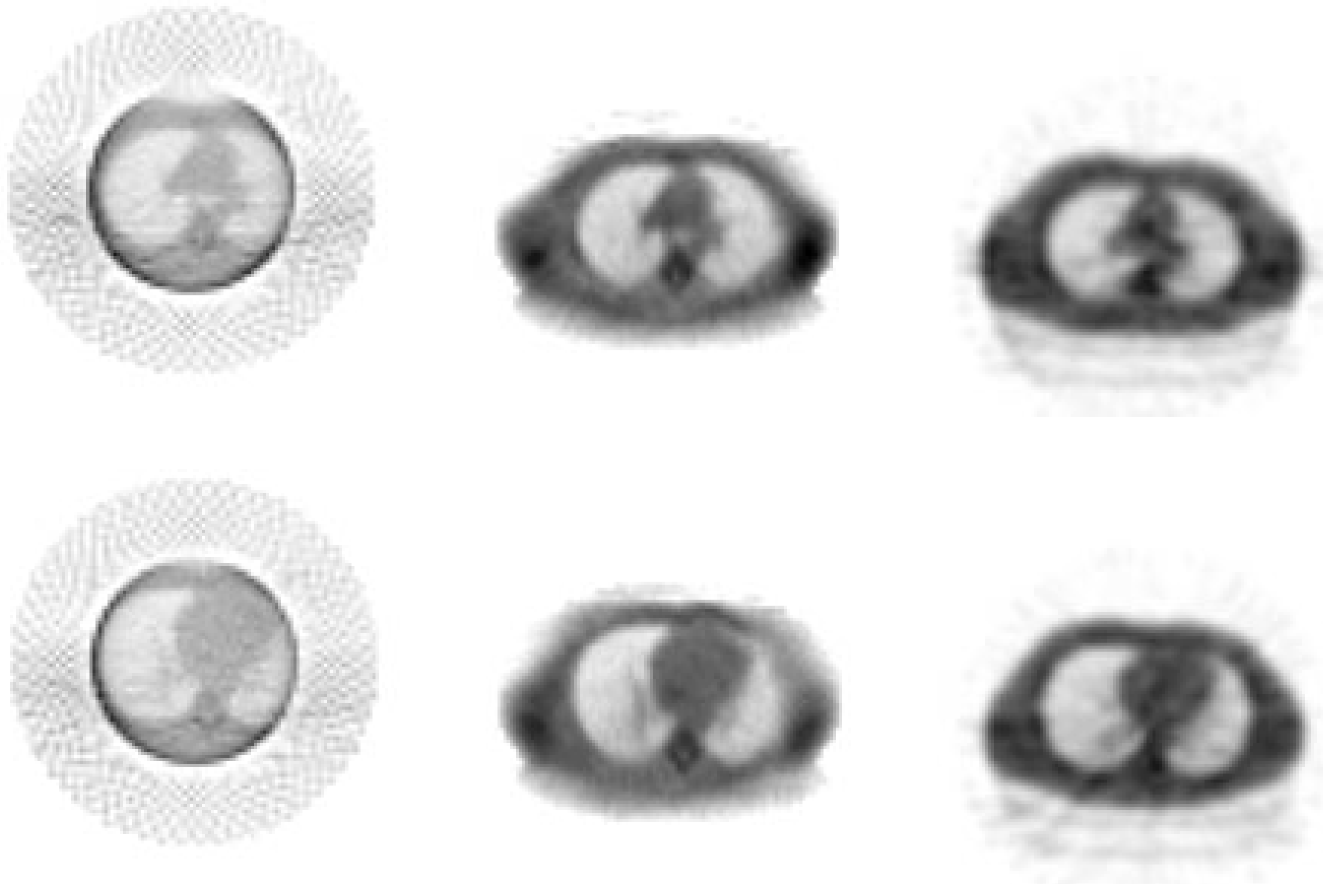


FDG PET Patient Data, PL-OSTR vs FBP



(15-minute transmission scan | 2-minute transmission scan)

Truncated Fan-Beam SPECT Transmission



Truncated FBP | Truncated PWLS | Untruncated FBP