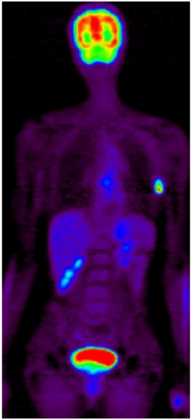
Tomographic Image Reconstruction.
An Introduction.

Milan Zvolský

Lecture on Medical Physics
by Erika Garutti & Florian Grüner

28.11.2014

Outline.



Introduction.

Tomography

Imaging by sectioning: The word tomography is derived from the greek *tome* (cut) and *graphein* (to write).

Introduction.

Tomography

Imaging by sectioning: The word tomography is derived from the greek *tome* (cut) and *graphein* (to write).

Cutting into slices is a bad idea to perform on humans

Introduction.

Figure: PET-CT

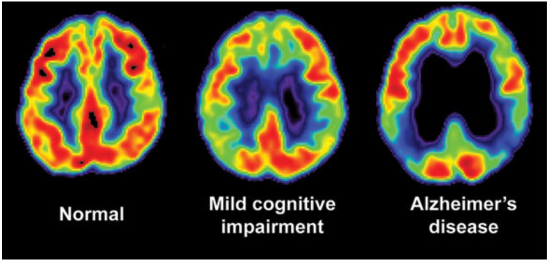


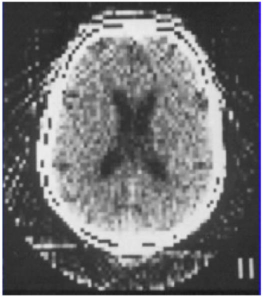
Figure: PET brain images

Figure: CT: 2D $\rightarrow$ 3D

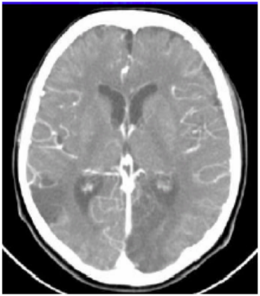
Introduction.

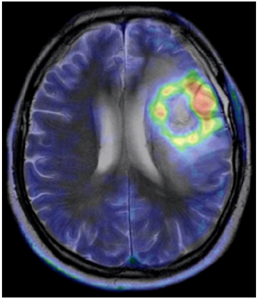
- Reconstruction quality is a crucial point in medical imaging
- No use in building accurate detector when image quality is poor
- Very active field of research

1974



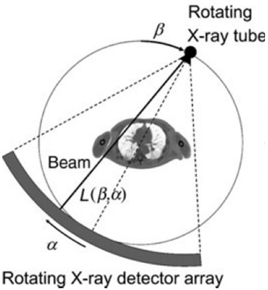
1994

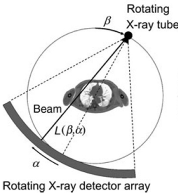


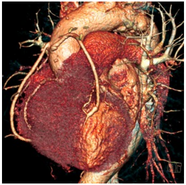


Computer Tomography (CT).

- Rotating X-ray tube + detector
- X-rays propagate through a x-section of patient







Computer Tomography (CT).

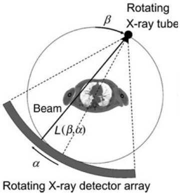
- Rotating X-ray tube + detector
- X-rays propagate through a x-section of patient

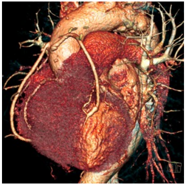
- Measure the exit beam intensity *integrated* along a line between X-ray source and detector

$$I_d = I_0 \exp\left[-\int_0^d \mu(s;\overline{E})\mathrm{d}s\right]$$

with μ linear attenuation coefficient as a function of the location s and the effective energy $\overline{E}$

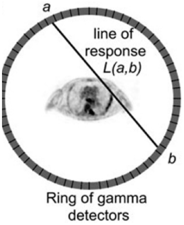
- Basic measurement of CT: Line integral of the linear attenuation coefficient

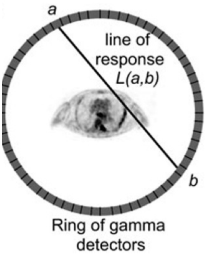


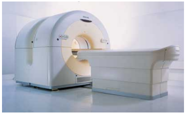


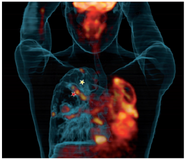
Positron Emission Tomography (PET).

- Glucose-like ^{18}FDG (β^+ emitter) concentrates in the metabolical active areas
- We are interested in the radioactivity distribution within the body
- $e^+e^- \rightarrow 2\gamma$ (back-to-back, 511 keV each)





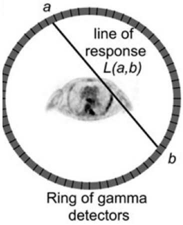




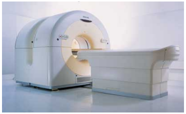
Nature Reviews | Microbiology

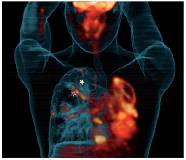
Positron Emission Tomography (PET).

- Glucose-like ^{18}FDG (β^+ emitter) concentrates in the metabolical active areas
- We are interested in the radioactivity distribution within the body
- $e^+e^- \rightarrow 2\gamma$ (back-to-back, 511 keV each)



- Detect the two γs in *coincidence*
- 2 detectors fire at the same time $\rightarrow$ Draw a line
- These are called *Lines of Response (LOR)*
- Measure the *integrated* activity of the LOR

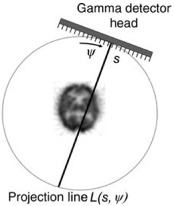


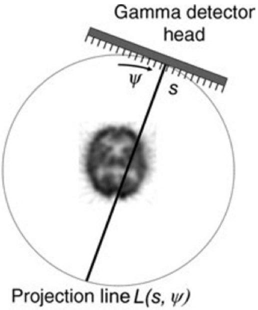


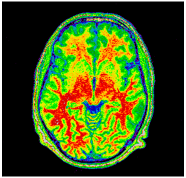
Nature Reviews | Microbiology

SPECT.

- Single Photon Emission Computed Tomography
- γ-emitting radionuclide $\rightarrow$ 1 γ $\rightarrow$ no LOR
- Gamma camera with collimators $\rightarrow$ reconstruct lines perpendicular to the detector

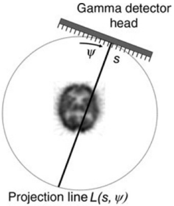




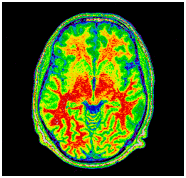


SPECT.

- Single Photon Emission Computed Tomography
- γ-emitting radionuclide $\rightarrow$ 1 γ $\rightarrow$ no LOR
- Gamma camera with collimators $\rightarrow$ reconstruct lines perpendicular to the detector



- Similar to PET (morphologic images)
- Cheaper/simpler than PET
- Worse spacial resolution, lower sensitivity



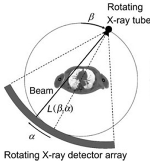
Emission & Transmission Tomography

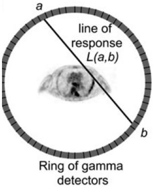
Transmission Tomography (CT)

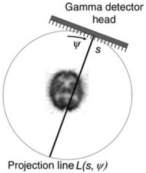
- Radiation source outside the patient: X-ray tube or long-lived radionuclide rotates around the body
- Quantity to be reconstructed is the photon linear attenuation coefficient of the body

Emission Tomography (PET & SPECT)

- Radiation source inside the patient: γ- or e^+-emitting radionuclide
- Quantity to be reconstructed is the activity concentration of the radiopharmacon inside the body



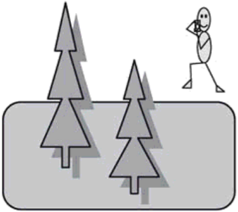


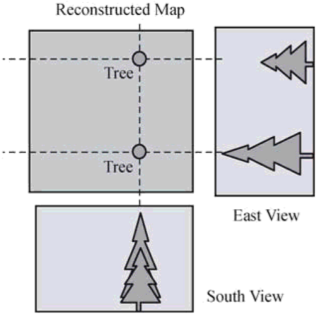


Basic Idea.

Example: Photography

Two trees in a park, make 2 pictures from east and south, try to create a map of the park.





A photo is a <u>projection</u> of an object onto a plane

Basic Idea.

Example: Another Photography

Other configuration: If you see two separate trees on both views, can you uniquely reconstruct the map of trees?
Here you *cannot* reconstruct the position and height of both trees.

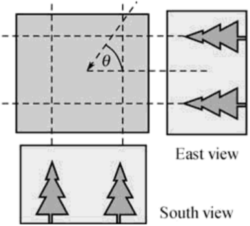


Figure: Two trees seen on *both* views

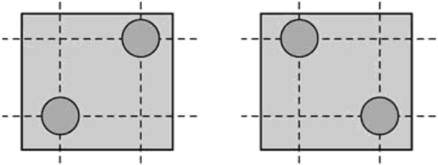


Figure: There are two solutions

Basic Idea.

Example: Another Photography

Other configuration: If you see two separate trees on both views, can you uniquely reconstruct the map of trees?
Here you *cannot* reconstruct the position and height of both trees.

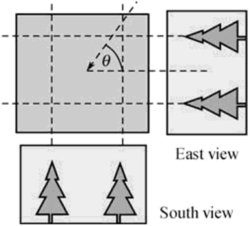


Figure: Two trees seen on *both* views

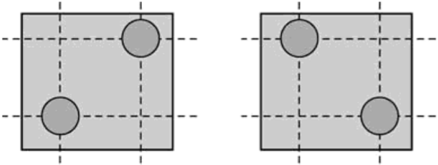
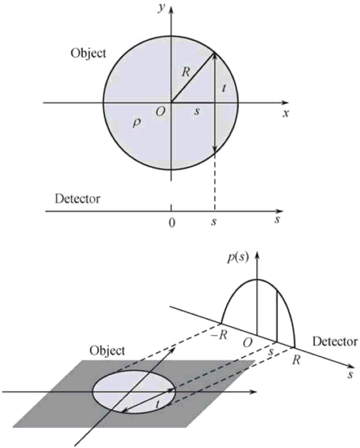


Figure: There are two solutions

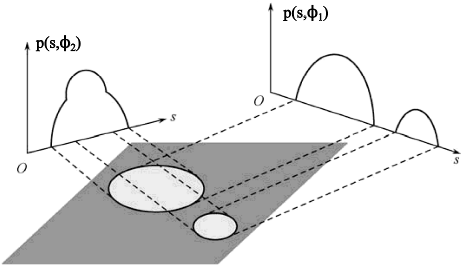
If we take another picture at 45°, we are able to solve the ambiguity.

Basic Idea: Projections.

- Before: photo, now: Projection is a line integral
- Projection $p(s, \phi)$ at angle ϕ, s is coordinate on detector



Projection $p(s)$ the same for any ϕ

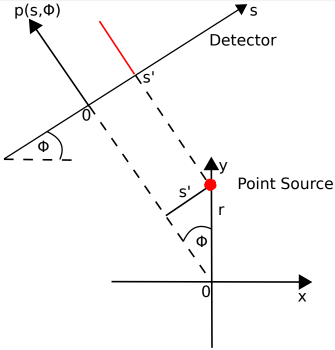


Projection $p(s, \phi)$ depends on orientation

Projections: Angle dependency.

Example: Point source on the y axis

Location s of the spike on the 1D detector: $s = r \sin \phi$.
The projection $p(s, \phi)$ in the s-ϕ-coordinate system is a sine function.



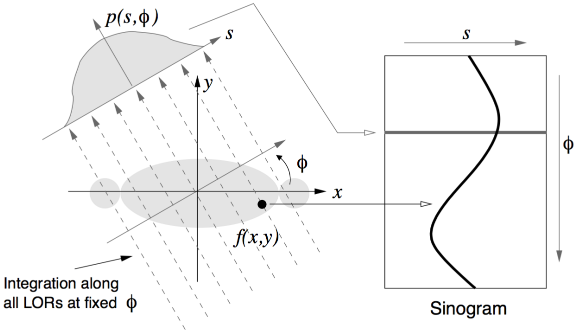
Sinogram

A sinogram is a representation of the projections on the s-ϕ plane.

Projections: Angle dependency.

Example: Point source on the y axis

Location s of the spike on the 1D detector: $s = r \sin \phi$.
The projection $p(s, \phi)$ in the s-ϕ-coordinate system is a sine function.



Sinogram

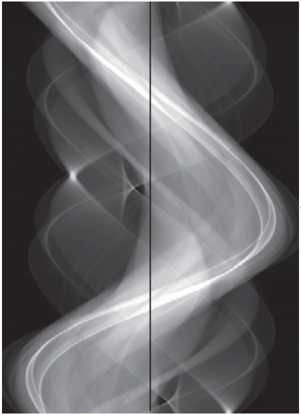
A sinogram is a representation of the projections on the s-ϕ plane.

Projections: Angle dependency.

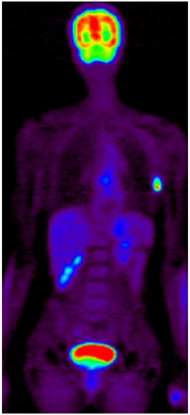
Tomography

Find the image $f(x, y)$ from the measured projections $p(s, \phi)$

We measure a sinogram:



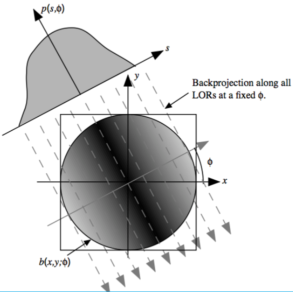
We want an image:



Backprojection Procedure.

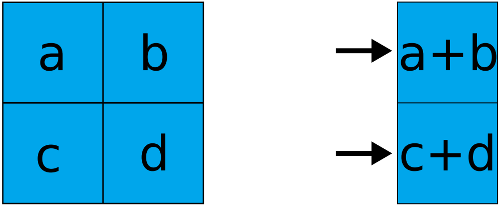
Backprojection

- Placing a value of $p(s, \phi)$ back into the position of the appropriate LOR
- But the knowledge of where the values came from was lost in the projection step
- The best we can do is place a constant value into all elements along the line



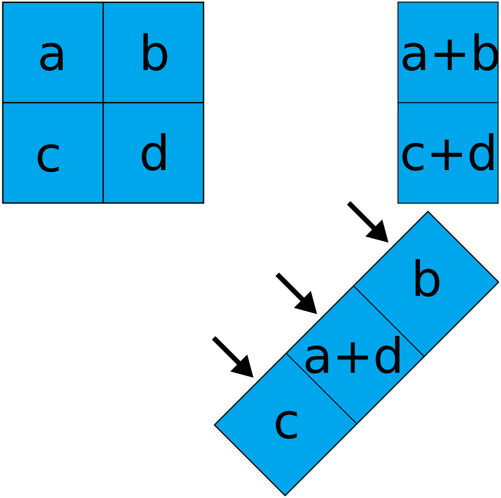
Backprojection Example.

1st projection



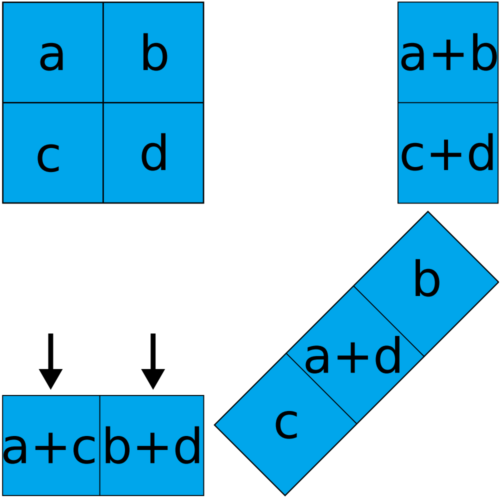
Backprojection Example.

2nd projection



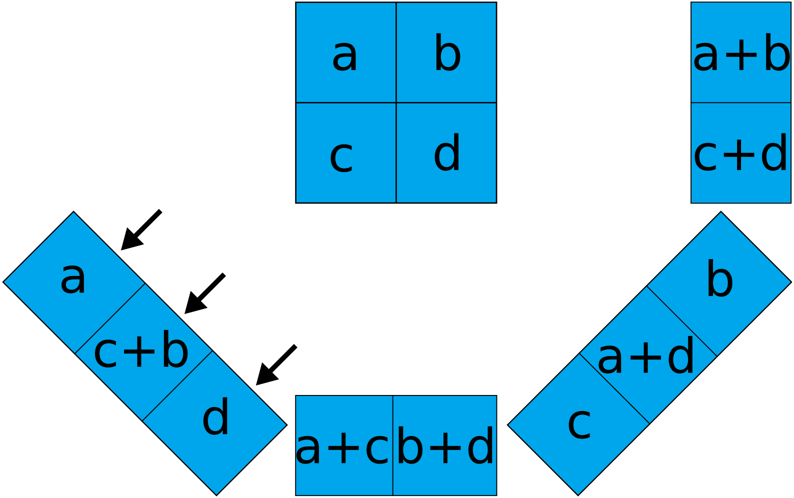
Backprojection Example.

3rd projection



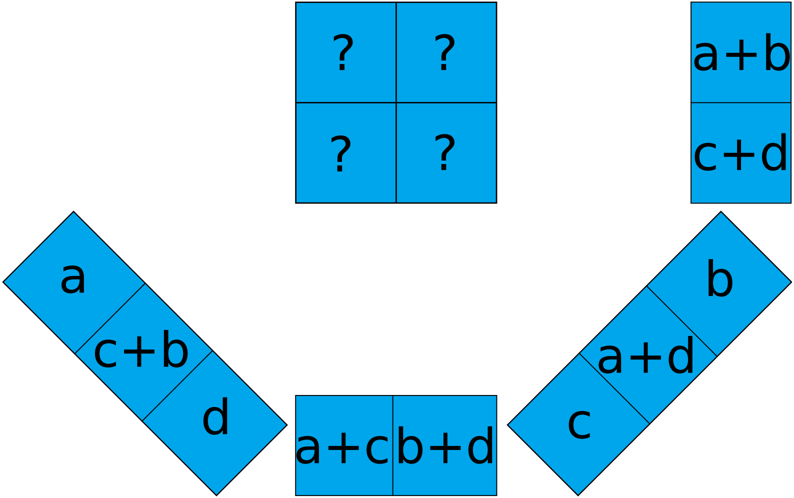
Backprojection Example.

4th projection



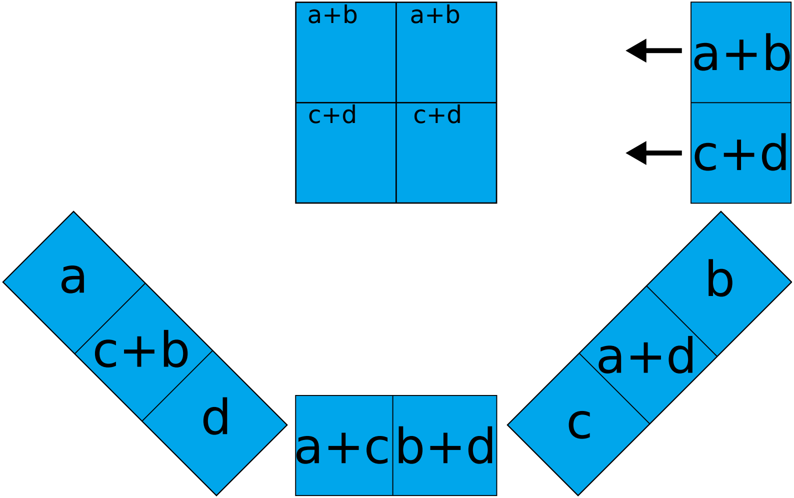
Backprojection Example.

Backproject



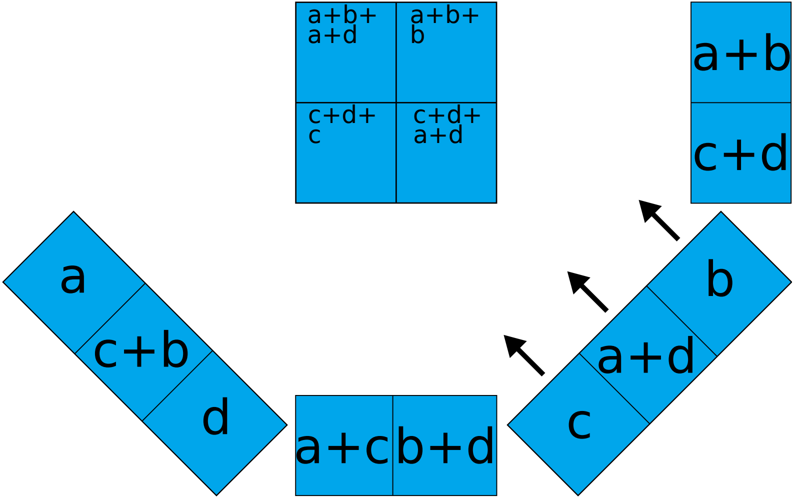
Backprojection Example.

1st backprojection



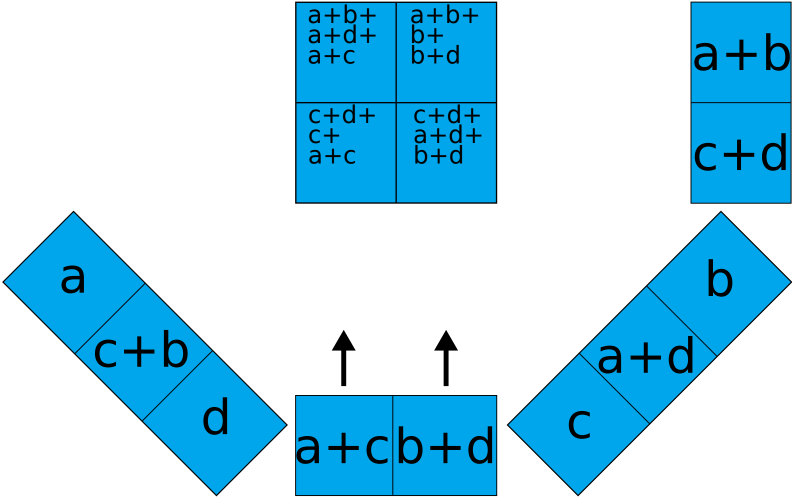
Backprojection Example.

2nd backprojection



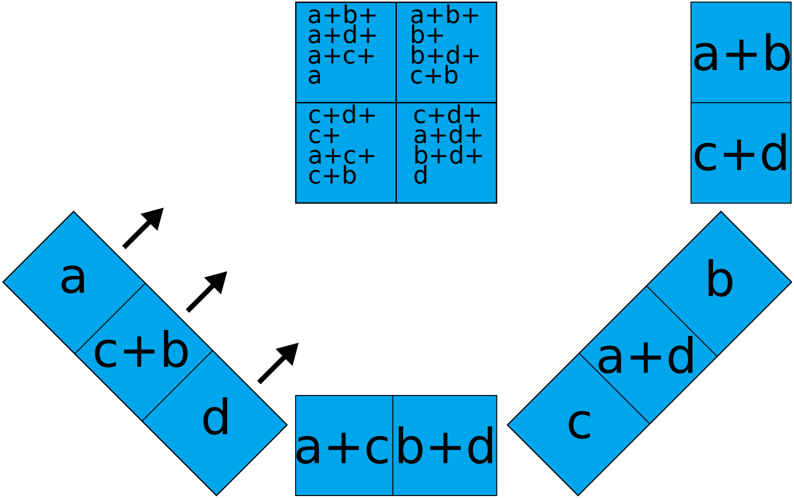
Backprojection Example.

3rd backprojection



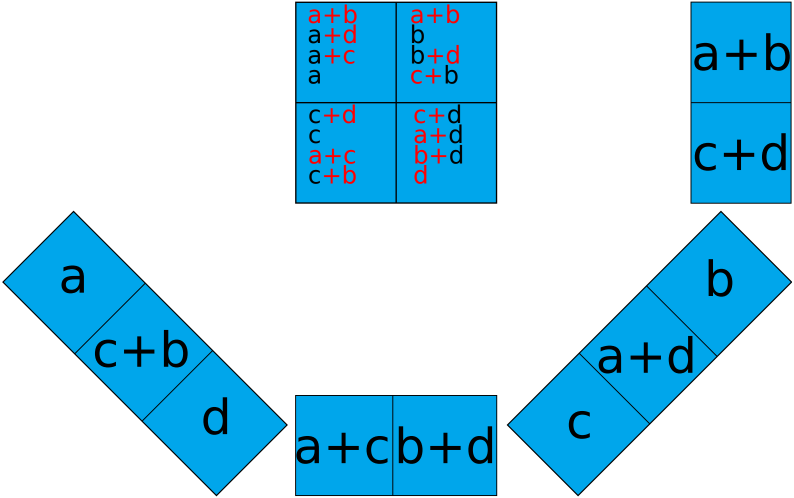
Backprojection Example.

4th backprojection



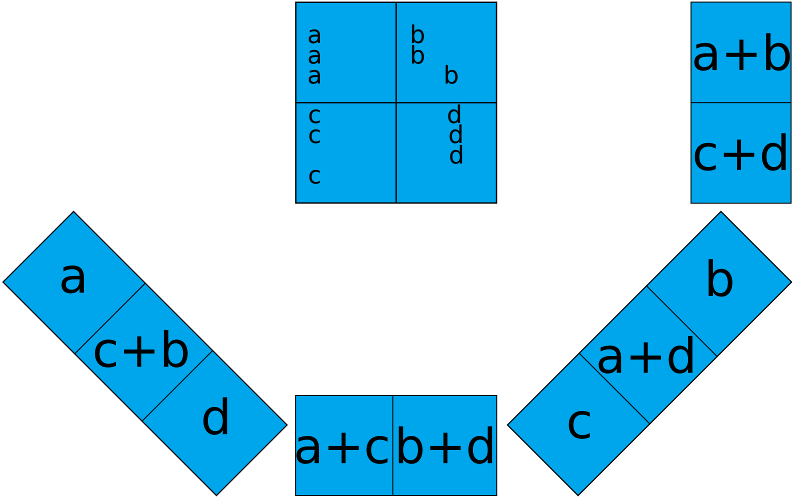
Backprojection Example.

Subtract projection sum from each entry



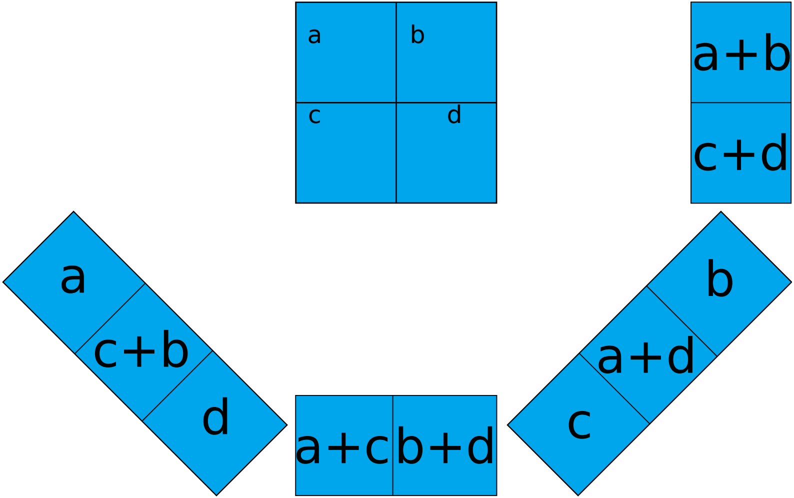
Backprojection Example.

Subtract projection sum from each entry

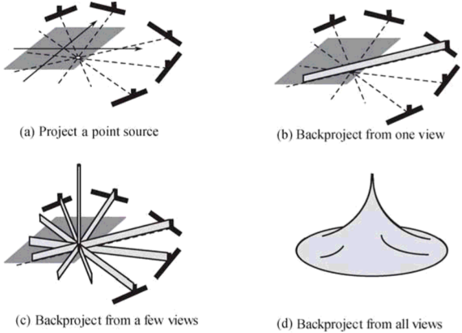


Backprojection Example.

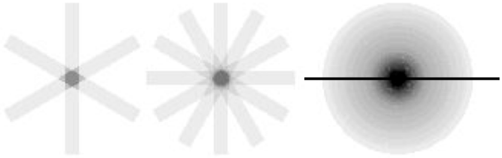
Divide by number of projections $-1 = 3$



Backprojection Procedure.



- 1 view: spike of intensity 1. This is sum of activity along projection path
- Re-distribute activity back to its original path
- Give equal activity everywhere along the line
- Many angles → Tall spike at the location of the point source
- (d) Ups...



Backprojection: Blurring of the image.

Figure: Original *Shepp-Logan phantom*

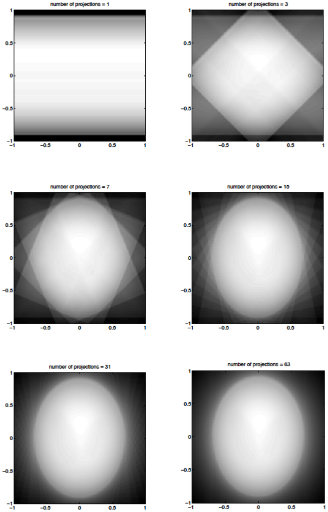


Figure: Backprojected image for (1,3,7,15,31,63) projections

Backprojection.

The Radon transform

The Radon transform of a distribution $f(x, y)$ is given by

$$p(s, \phi) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) \cdot \delta(x \cos\phi + y \sin\phi - s) \mathrm{d}x \mathrm{d}y$$

- Delta function: Integrand is zero everywhere except on the line $L(s, \phi)$

Backprojected image:

$$b(x, y) = \int_{0}^{\pi} p(s, \phi)|_{s = x\cos\phi + y\sin\phi} \mathrm{d}\phi$$

- Integrate over 180°, the other half doesn't give extra information

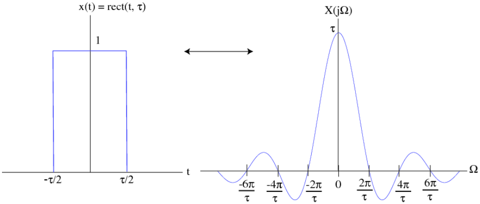
Reconstructed image is blurred:

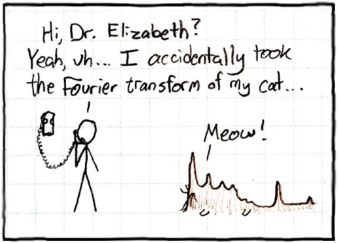
$$b(x, y) = f(x, y) \times \frac{1}{\sqrt{x^2 + y^2}}$$

There is a close relationship between Radon and the Fourier trafo!

Fourier transform (FT)

$$\mathcal{F}\{p(s)\} = P(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} p(s) e^{-i\omega s} \mathrm{d}s$$





- Summation of lines causes duplication in the center
- Oversampling in the center of the Fourier space

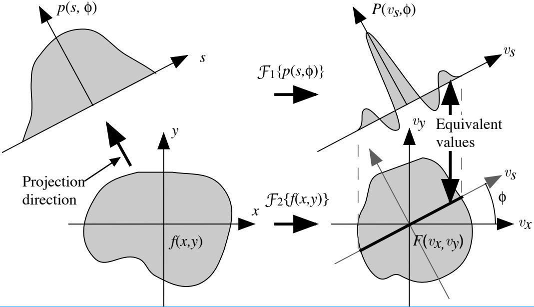
Central Slice Theorem

Central Slice Theorem (CST)

$$\mathcal{F}_1\{p(s,\phi')\} = \mathcal{F}_2\{f(x,y)\}_{|\phi=\phi'}$$

The following operations are the equivalent:

- Take a 2D function $f(x,y)$, project it onto a line, and do a FT of that projection.
- Do a 2D FT of $f(x,y)$ first, and then take a slice through the origin, parallel to the projection line.



Filtered Backprojection: “Proof”.

- Re-write the image $f(x, y)$ via the inverse FT:

$$f(x, y) = \mathcal{F}_2^{-1}\{F(v_x, v_y)\}$$

Filtered Backprojection: "Proof".

- Re-write the image $f(x, y)$ via the inverse FT:

$$f(x, y) = \mathcal{F}_2^{-1}\{F(v_x, v_y)\} = \int_{-\infty}^{\infty} \mathrm{d}v_x \int_{-\infty}^{\infty} \mathrm{d}v_y F(v_x, v_y) e^{2\pi i (v_x x + v_y y)}$$

Filtered Backprojection: “Proof”.

- Re-write the image $f(x, y)$ via the inverse FT:

$$f(x,y) = \mathcal{F}_2^{-1}\{F(v_x, v_y)\} = \int_{-\infty}^{\infty} \mathrm{d}v_x \int_{-\infty}^{\infty} \mathrm{d}v_y F(v_x, v_y) e^{2\pi i(v_x x + v_y y)}$$

- Polar coordinates: $v_x = \omega \cos\phi$, $v_y = \omega \sin\phi$, $\mathrm{d}v_x \mathrm{d}v_y = \omega \mathrm{d}\omega \mathrm{d}\phi$:

Filtered Backprojection: “Proof”.

- Re-write the image $f(x, y)$ via the inverse FT:

$$f(x,y) = \mathcal{F}_2^{-1}\{F(v_x, v_y)\} = \int_{-\infty}^{\infty} \mathrm{d}v_x \int_{-\infty}^{\infty} \mathrm{d}v_y F(v_x, v_y) e^{2\pi i(v_x x + v_y y)}$$

- Polar coordinates: $v_x = \omega \cos\phi$, $v_y = \omega \sin\phi$, $\mathrm{d}v_x \mathrm{d}v_y = \omega \mathrm{d}\omega \mathrm{d}\phi$:

$$f(x,y) = \int_0^{2\pi} \mathrm{d}\phi \int_0^{\infty} \mathrm{d}\omega\omega \underbrace{F(\omega\cos\phi, \omega\sin\phi)}_{=P(\omega)\ \text{(CST)}} e^{2\pi i\omega(\overbrace{x\cos\phi + y\sin\phi}^{=s})}$$

$$= \int_0^{\pi} \mathrm{d}\phi \int_{-\infty}^{\infty} \mathrm{d}\omega|\omega| P(\omega) e^{2\pi i\omega s} \quad \text{(changing the integration limits)}$$

Filtered Backprojection: “Proof”.

- Re-write the image $f(x, y)$ via the inverse FT:

$$f(x,y) = \mathcal{F}_2^{-1}\{F(v_x, v_y)\} = \int_{-\infty}^{\infty} \mathrm{d}v_x \int_{-\infty}^{\infty} \mathrm{d}v_y F(v_x, v_y) e^{2\pi i (v_x x + v_y y)}$$

- Polar coordinates: $v_x = \omega \cos\phi$, $v_y = \omega \sin\phi$, $\mathrm{d}v_x \mathrm{d}v_y = \omega \mathrm{d}\omega \mathrm{d}\phi$:

$$f(x,y) = \int_0^{2\pi} \mathrm{d}\phi \int_0^{\infty} \mathrm{d}\omega\omega \underbrace{F(\omega\cos\phi, \omega\sin\phi)}_{=P(\omega)\ \text{(CST)}} e^{2\pi i \omega \overbrace{(x\cos\phi + y\sin\phi)}^{=s}}$$

$$= \int_0^{\pi} \mathrm{d}\phi \int_{-\infty}^{\infty} \mathrm{d}\omega |\omega| P(\omega) e^{2\pi i \omega s} \quad \text{(changing the integration limits)}$$

$$f(x,y) = \int_0^{\pi} \mathrm{d}\phi \underbrace{\left[\int_{-\infty}^{\infty} \mathrm{d}\omega |\omega| P(\omega) e^{2\pi i \omega s} \right]}_{=\mathcal{F}^{-1}\{|\omega| P(\omega)\} \equiv p'(s,\phi)} = \int_0^{\pi} \mathrm{d}\phi p'(s,\phi)$$

Filtered Backprojection.

$$f(x,y) = \int_0^\pi \mathrm{d}\phi \Big[\int_{-\infty}^{\infty} \mathrm{d}\omega |\omega| P(\omega) e^{2\pi i \omega s} \Big]_{s = x\cos\phi + y\sin\phi}$$

Filtered Backprojection.

$$f(x,y) = \int_0^{\pi} \mathrm{d}\phi \Big[\int_{-\infty}^{\infty} \mathrm{d}\omega |\omega| P(\omega) e^{2\pi i \omega s} \Big]_{s = x\cos\phi + y\sin\phi}$$

- FT of projection $p(s,\phi)$

Filtered Backprojection.

$$f(x,y) = \int_0^\pi \mathrm{d}\phi \Big[\int_{-\infty}^{\infty} \mathrm{d}\omega |\omega| P(\omega) e^{2\pi i \omega s} \Big]_{s = x\cos\phi + y\sin\phi}$$

- FT of projection $p(s,\phi)$
- Multiply by frequency filter $|\omega|$

Filtered Backprojection

$$f(x,y) = \int_0^\pi \mathrm{d}\phi \Big[\int_{-\infty}^{\infty} \mathrm{d}\omega |\omega| P(\omega) e^{2\pi i \omega s} \Big]_{s = x\cos\phi + y\sin\phi}$$

- FT of projection $p(s, \phi)$
- Multiply by frequency filter $|\omega|$
- Inverse-transform this product

Filtered Backprojection.

$$f(x,y) = \int_0^{\pi} \mathrm{d}\phi \Big[\int_{-\infty}^{\infty} \mathrm{d}\omega |\omega| P(\omega) e^{2\pi i \omega s} \Big]_{s=x\cos\phi + y\sin\phi}$$

- FT of projection $p(s,\phi)$
- Multiply by frequency filter $|\omega|$
- Inverse-transform this product
- This filtered projection is backprojected

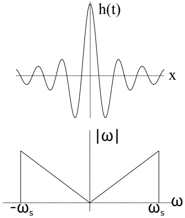
Filtered Backprojection.

$$f(x,y) = \int_0^{\pi} \mathrm{d}\phi \Big[\int_{-\infty}^{\infty} \mathrm{d}\omega |\omega| P(\omega) e^{2\pi i \omega s} \Big]_{s = x\cos\phi + y\sin\phi}$$

- FT of projection $p(s,\phi)$
- Multiply by frequency filter $|\omega|$
- Inverse-transform this product
- This filtered projection is backprojected
- Then sum over all filtered projections

$$f(x,y) = \int_0^\pi \mathrm{d}\phi \Big[\int_{-\infty}^{\infty} \mathrm{d}\omega |\omega| P(\omega) e^{2\pi i \omega s} \Big]_{s=x\cos\phi + y\sin\phi}$$

- FT of projection $p(s,\phi)$
- Multiply by frequency filter $|\omega|$
- Inverse-transform this product
- This filtered projection is backprojected
- Then sum over all filtered projections



- Ramp filter $|\omega|$
- Blurring (low ω) minimised
- Contrasts (high ω) accentuated

A *complete* set of 1D projections allows the reconstruction of the *original* 2D distribution without loss of information

BP vs. FBP.

One single projection for both methods:

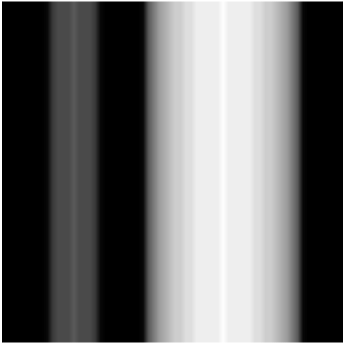


Figure: Backprojection

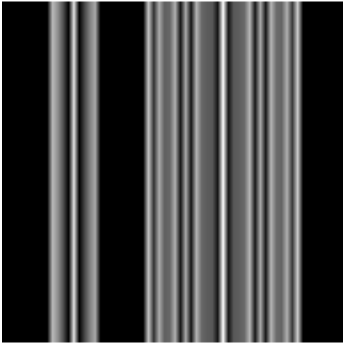
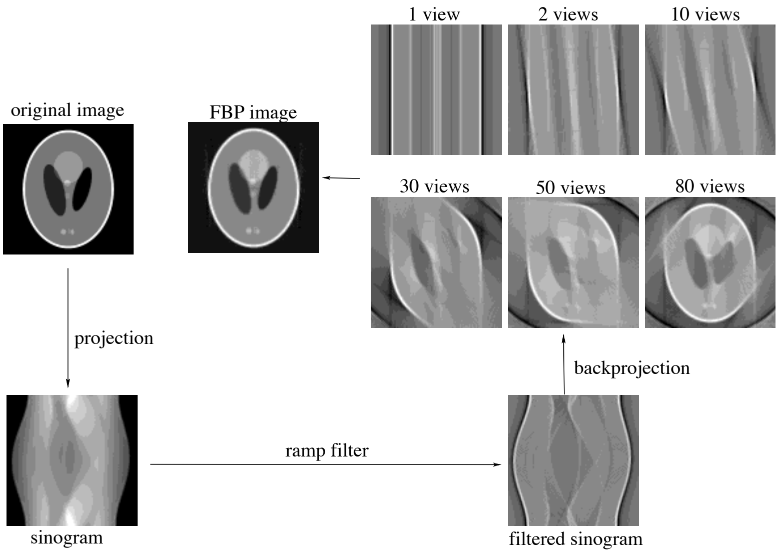


Figure: Filtered Backprojection

Filtered Backprojection.



BP vs. FBP.

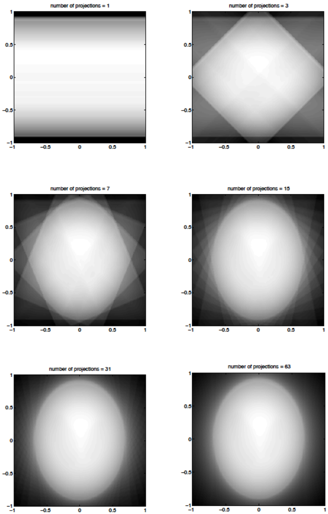


Figure: Backprojection

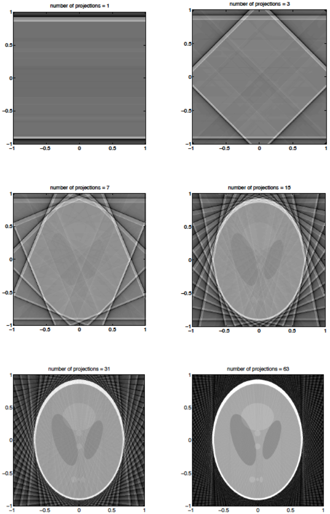
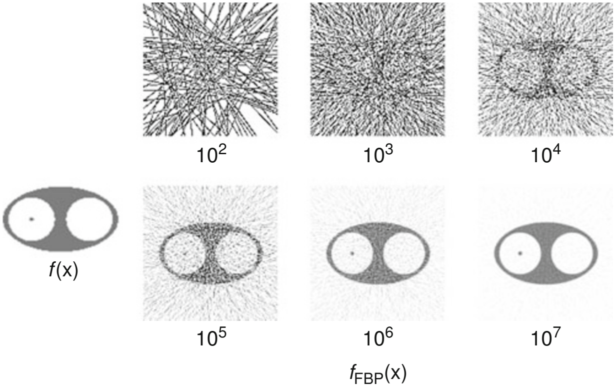


Figure: Filtered Backprojection

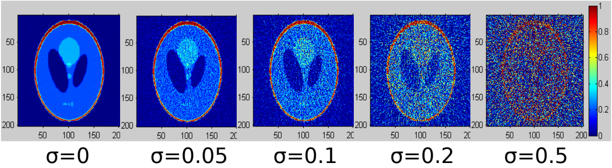
Image Quality vs. Number of Events



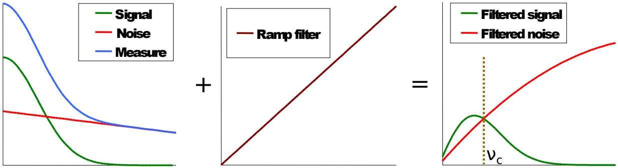
Number of events follows Poisson distribution
$\rightarrow$ Signal-to-noise ratio $\approx \sqrt{N}$

Analytical Reconstruction works exactly for $N = \infty$

Effect of Poissonian Noise.



- Typically low noise for CT, high noise for PET
- High signal variability but FBP assumes exact distribution
- Amplification of high-frequency noise (→ use cut-off)

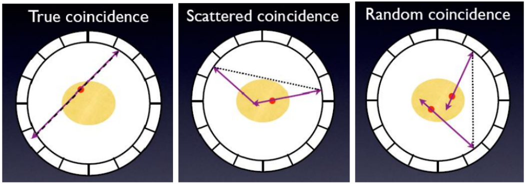


FBP bad if data is noisy → Iterative Image Reconstruction

FBP vs. Reality

Reality is more complex:

- Data is discrete: FBP only precise if all angles are available
- Data is noisy: Measurements follow a probability distribution
- Detectors are unprecise: mis-positioning of photons
- Detector geometry may not provide complete data
- Not all photons travel along straight lines: scatter, absorption

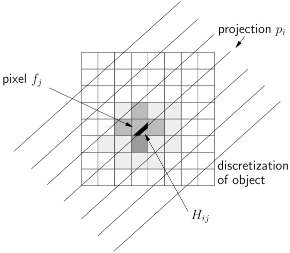


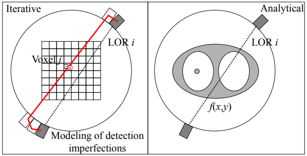
Input prior knowledge of the detector & the physics process

Iterative Methods - Basic Idea.

- Emission tomography described as
  $\mathbf{y} = \mathbf{S}(\mathbf{x}) + \text{noise}$

  $\mathbf{y}$: measurement, $\mathbf{S}$: System operator, $\mathbf{x}$: activity
- Model system operator $\mathbf{S}$ with system matrix $\mathbf{A}$ $\rightarrow$ linear approximation
- A_{ij}: Probability that emission in voxel i (x_i) results in a detection in detector j (y_j)
- $\mathbf{y} = \mathbf{A}\mathbf{x}$ or $y_j = \sum_i A_{ij} x_i$
- Inverse problem: Know y_i, want x_i
- Could do $\mathbf{x} = \mathbf{A}^{-1}\mathbf{y}$
- But: $\mathbf{A}$ is huge & cannot be inverted





Iterative image reconstruction

- Solve inverse problem iteratively
- Assume Poissonian noise on y_i

Iterative Methods - Basic Idea.

- Measurements y_i: independent random variables (**Poisson**)
- Expectation value: $\mu_i = \mathbf{A}_i \cdot \mathbf{x} = \sum_j A_{ij} x_j$
- Probability to measure k given λ: $p(k|\lambda) = \frac{e^{-\lambda}\lambda^k}{k!}$

Likelihood:

$$L(\mathbf{x}) = p(\mathbf{y}|\mathbf{x}) = \prod_i p(y_i|\mathbf{x}) = \prod_i \frac{e^{-\mathbf{A}_i \cdot \mathbf{x}} (\mathbf{A}_i \cdot \mathbf{x})^{y_i}}{y_i!}$$

- Among all possible images $\mathbf{x}$ we choose the one that maximises the probability of producing the data (find most likely image)
- Maximise the log-likelihood, i.e. maximise $\log(p(\mathbf{y}|\mathbf{x}))$

Maximum Likelihood - Expectation Maximisation (ML-EM)

- Objective function to maximise: log-Likelihood (ML)
- Maximisation algorithm: Expectation Maximisation (EM)

ML-EM Algorithm.

- Initial guess for the image (uniform)
- Simulate measurements from estimate (forward proj.)
- Compare this with actual measurements
- Improve image estimate (backward projection)
- Repeat until convergence

ML-EM

$$x_i^{(n+1)} = x_i^{(n)} \cdot \frac{1}{\sum_j A_{ij}} \cdot \sum_j A_{ij} \frac{y_j}{\sum_k A_{kj} x_k^{(n)}}$$

ML-EM Algorithm.

- Initial guess for the image (uniform)
  $x_i^{(0)}$
- Simulate measurements from estimate (forward proj.)
- Compare this with actual measurements
- Improve image estimate (backward projection)
- Repeat until convergence

ML-EM

$$x_i^{(n+1)} = x_i^{(n)} \cdot \frac{1}{\sum_j A_{ij}} \cdot \sum_j A_{ij} \frac{y_j}{\sum_k A_{kj} x_k^{(n)}}$$

ML-EM Algorithm.

- Initial guess for the image (uniform)
  $x_i^{(0)}$
- Simulate measurements from estimate (forward proj.)
  $y_j^{\text{simu}} = \sum_k A_{kj} x_k^{(0)}$
- Compare this with actual measurements
- Improve image estimate (backward projection)
- Repeat until convergence

ML-EM

$$x_i^{(n+1)} = x_i^{(n)} \cdot \frac{1}{\sum_j A_{ij}} \cdot \sum_j A_{ij} \frac{y_j}{\sum_k A_{kj} x_k^{(n)}}$$

ML-EM Algorithm.

- Initial guess for the image (uniform)
  $x_i^{(0)}$
- Simulate measurements from estimate (forward proj.)
  $y_j^{\text{simu}} = \sum_k A_{kj} x_k^{(0)}$
- Compare this with actual measurements
  Ratio $R_j = \frac{y_j}{y_j^{\text{simu}}}$
- Improve image estimate (backward projection)
- Repeat until convergence

ML-EM

$$x_i^{(n+1)} = x_i^{(n)} \cdot \frac{1}{\sum_j A_{ij}} \cdot \sum_j A_{ij} \frac{y_j}{\sum_k A_{kj} x_k^{(n)}}$$

ML-EM Algorithm.

- Initial guess for the image (uniform)
  $x_i^{(0)}$
- Simulate measurements from estimate (forward proj.)
  $y_j^{\text{simu}} = \sum_k A_{kj} x_k^{(0)}$
- Compare this with actual measurements
  Ratio $R_j = \frac{y_j}{y_j^{\text{simu}}}$
- Improve image estimate (backward projection)
  $x_i^{(1)} = x_i^{(0)} \cdot \frac{1}{\sum_j A_{ij}} \cdot \sum_j A_{ij} R_j$
- Repeat until convergence

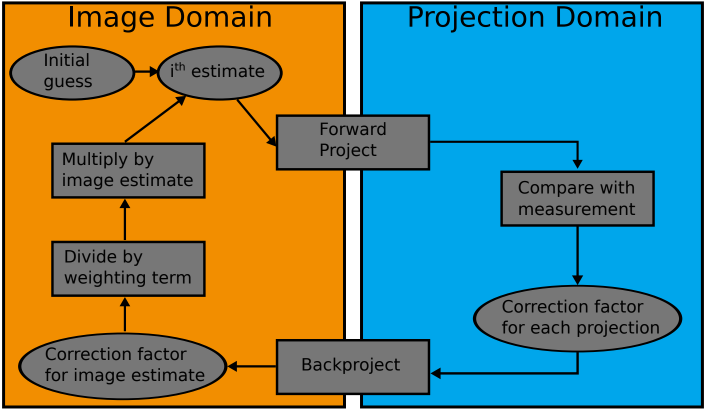
ML-EM

$$x_i^{(n+1)} = x_i^{(n)} \cdot \frac{1}{\sum_j A_{ij}} \cdot \sum_j A_{ij} \frac{y_j}{\sum_k A_{kj} x_k^{(n)}}$$

ML-EM Algorithm.

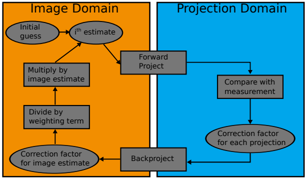
ML-EM

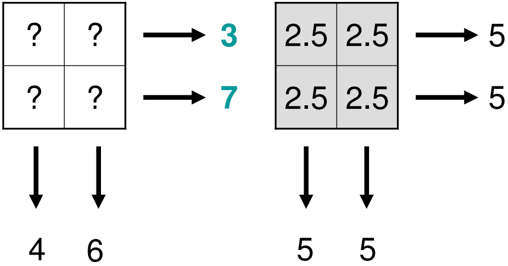
$$x_i^{(n+1)} = x_i^{(n)} \cdot \frac{1}{\sum_j A_{ij}} \cdot \sum_j A_{ij} \frac{y_j}{\sum_k A_{kj} x_k^{(n)}}$$



ML-EM

$$x_i^{(n+1)} = x_i^{(n)} \cdot \frac{1}{\sum_j A_{ij}} \cdot \sum_j A_{ij} \frac{y_j}{\sum_k A_{kj} x_k^{(n)}}$$





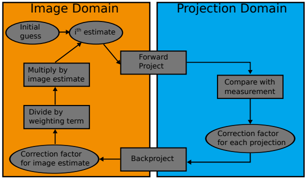
- Assume $A_{ij} = 1 \; \forall i, j$
- $\sum_j A_{ij} = 2 \; \forall i$

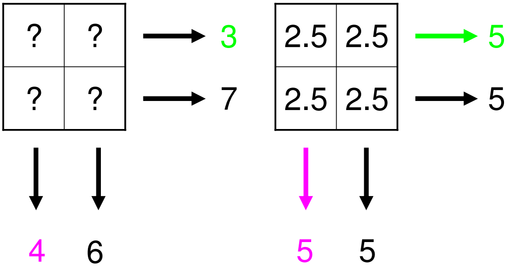
x=(3+7)/4=10/4=2.5

ML-EM Algorithm.

ML-EM

$$x_i^{(n+1)} = x_i^{(n)} \cdot \frac{1}{\sum_j A_{ij}} \cdot \sum_j A_{ij} \frac{y_j}{\sum_k A_{kj} x_k^{(n)}}$$





- Assume $A_{ij} = 1 \; \forall i, j$
- $\sum_j A_{ij} = 2 \; \forall i$

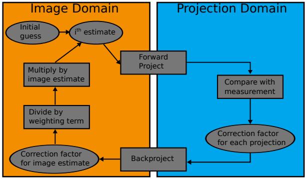
c11 = (3/5 + 4/5)/2 = 0.7

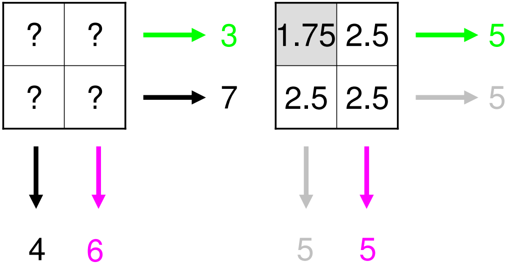
x11 = 1.75

ML-EM Algorithm.

ML-EM

$$x_i^{(n+1)} = x_i^{(n)} \cdot \frac{1}{\sum_j A_{ij}} \cdot \sum_j A_{ij} \frac{y_j}{\sum_k A_{kj} x_k^{(n)}}$$





- Assume $A_{ij} = 1 \; \forall i, j$
- $\sum_j A_{ij} = 2 \; \forall i$

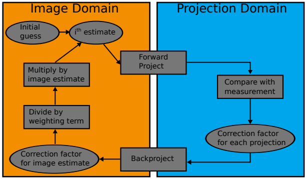
c12 = (3/5 + 6/5)/2 = 0.9

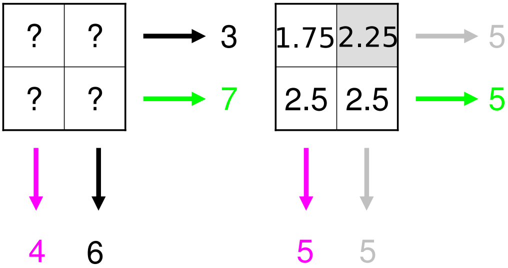
x12 = 2.25

ML-EM Algorithm.

ML-EM

$$x_i^{(n+1)} = x_i^{(n)} \cdot \frac{1}{\sum_j A_{ij}} \cdot \sum_j A_{ij} \frac{y_j}{\sum_k A_{kj} x_k^{(n)}}$$





- Assume $A_{ij} = 1 \; \forall i, j$
- $\sum_j A_{ij} = 2 \; \forall i$

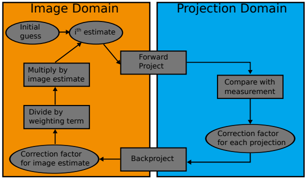
c13 = (7/5 + 4/5)/2 = 1.1

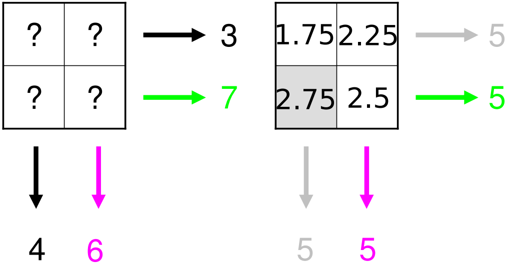
x13 = 2.75

ML-EM Algorithm.

ML-EM

$$x_i^{(n+1)} = x_i^{(n)} \cdot \frac{1}{\sum_j A_{ij}} \cdot \sum_j A_{ij} \frac{y_j}{\sum_k A_{kj} x_k^{(n)}}$$





- Assume $A_{ij} = 1 \ \forall i, j$
- $\sum_j A_{ij} = 2 \ \forall i$

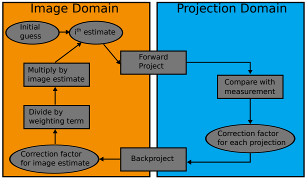
c14 = (7/5 + 6/5)/2 = 1.3

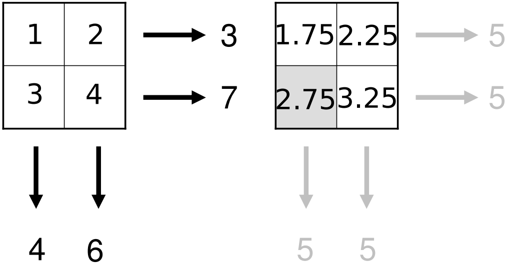
x14 = 3.25

ML-EM Algorithm.

ML-EM

$$x_i^{(n+1)} = x_i^{(n)} \cdot \frac{1}{\sum_j A_{ij}} \cdot \sum_j A_{ij} \frac{y_j}{\sum_k A_{kj} x_k^{(n)}}$$



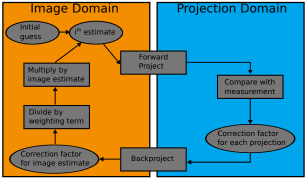


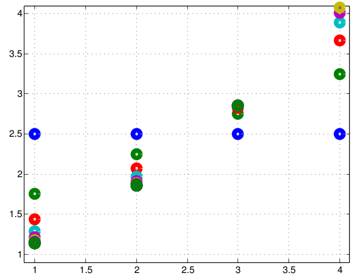
- Assume $A_{ij} = 1 \; \forall i, j$
- $\sum_j A_{ij} = 2 \; \forall i$

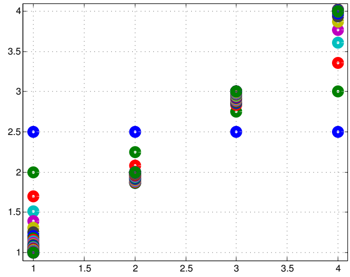
ML-EM Algorithm.

ML-EM

$$x_i^{(n+1)} = x_i^{(n)} \cdot \frac{1}{\sum_j A_{ij}} \cdot \sum_j A_{ij} \frac{y_j}{\sum_k A_{kj} x_k^{(n)}}$$







4 projections

No convergence to true image

5 projections

Deviation $< 1\%$ after 50 it.

ML-EM: Number of Iterations.

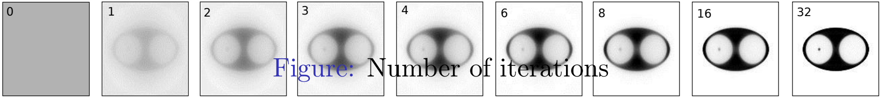
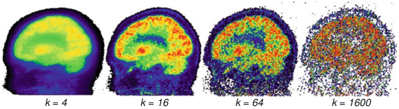
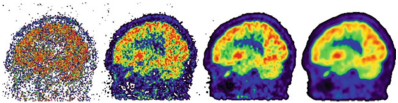


Figure: Number of iterations

- With noise: > 50 iterations needed
- ≈ 100 times slower than FBP
- Low frequencies are reconstructed first. For higher iterations, the alg. will attempt to recover noise $\rightarrow$ high spacial variance
  - Early termination (fast, but non-uniform convergence)
  - Post-smoothing with Gaussian kernel

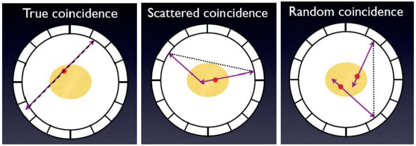


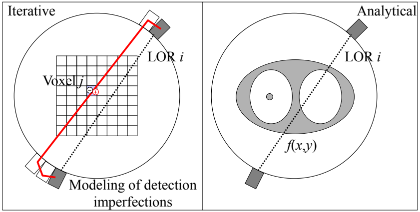


System Model.

Reconstruction can only be as good as our model agrees with reality
→ Consider all physical effects governing our system:

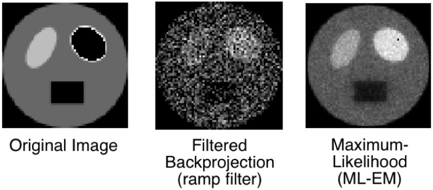
- Positron range
- Photon acollinearity
- Scattering in the body
- Attenuation in the body
- Scattering in the detector
- Absorption in the detector
- Deadtime of detector
- Random coincidences coming from the background

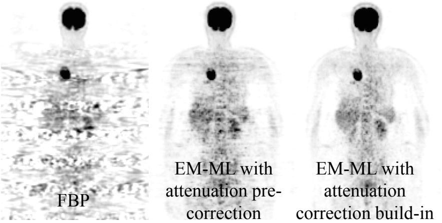




ML-EM vs. FBP.

Iterative methods perform much better than analytical methods if data is noisy:

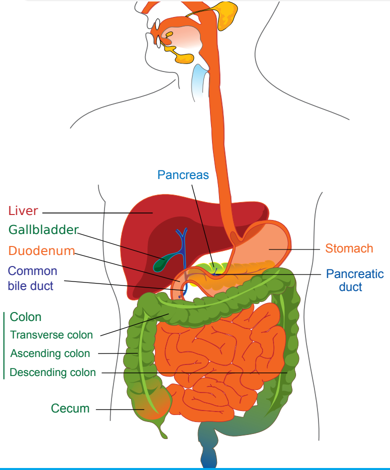




The EndoTOFPET-US Project

Positron Emission Tomography (PET)

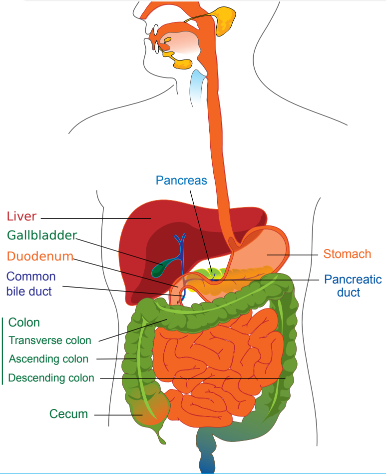
- Radiotracer (β^+ emitter) concentrates in the metabolical active areas
- $e^+e^- \to 2\gamma$ (back-to-back, 511 keV each)
- Detect the two γs in *coincidence*



The EndoTOFPET-US Project

Objectives

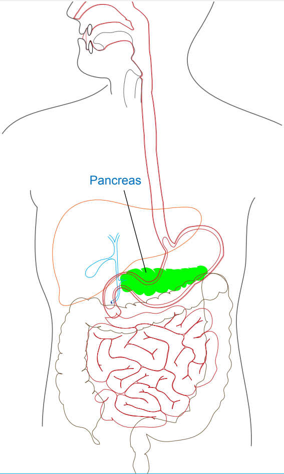
- Development of new biomarkers
- Intra-operative Time-of-Flight (TOF) PET Detector
- Prototype for prostate & pancreas cancer



The EndoTOFPET-US Project

Objectives

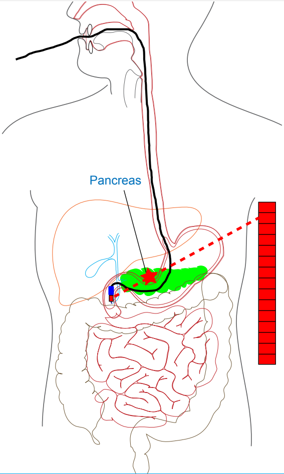
- Development of new biomarkers
- Intra-operative Time-of-Flight (TOF) PET Detector
- Prototype for prostate & pancreas cancer



The EndoTOFPET-US Project

Objectives

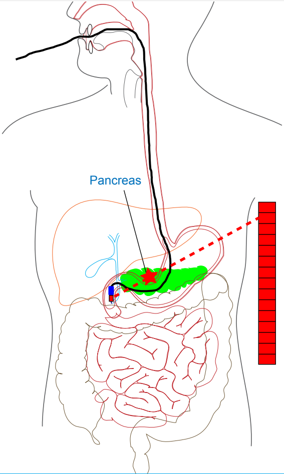
- Development of new biomarkers
- Intra-operative Time-of-Flight (TOF) PET Detector
- Prototype for prostate & pancreas cancer

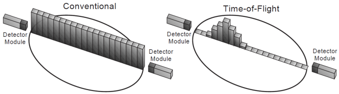


The EndoTOFPET-US Project

Objectives

- Development of new biomarkers
- Intra-operative Time-of-Flight (TOF) PET Detector
- Prototype for prostate & pancreas cancer





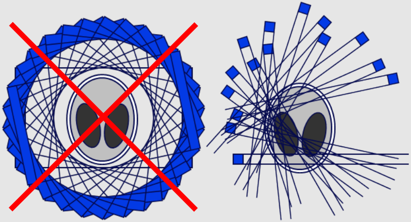
Challenges

- Extreme miniaturisation
- Fast crystals & ultra-fast photodetection
- Aim for our project: Coincidence time resolution 200 ps FWHM (3 cm)
- This reduces background noise from other organs → better image quality
- Image reconstruction for free-hand imaging, image resolution of 1 mm

EndoTOFPET Image Reconstruction

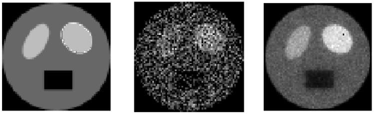
Challenges

- Time of flight (TOF)
- Limited angle problem
- Freehand
  $\rightarrow$ undefined volume of interest
- Low sensitivity ($\approx 1\%$), high noise
- Reconstruct the image on-line to provide guidance for the physician



Solution

- ML-EM iterative reconstruction: Good performance in case of Poissonian noise
- GPU Computation: Solving a massively parallel problem



Original Image

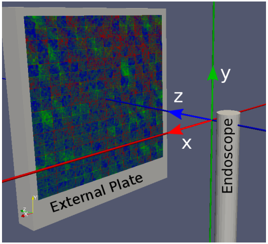
Filtered Backprojection (ramp filter)

Maximum-Likelihood (ML-EM)

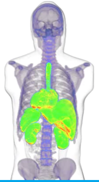
- GPU speedup by factor $\mathcal{O}(10)$
- Image reconstruction in $\mathcal{O}(\text{min.})$

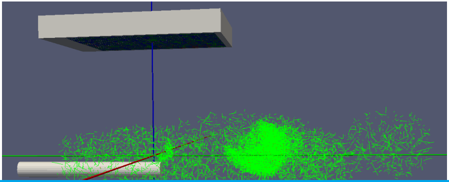
Full System Simulation.

- Simulation of whole detector system → optimise design, test reconstruction
- Geant4-based toolkit GAMOS with custom extensions (e.g. TOF)
- Study sensitivity, image resolution etc. on simple phantoms
- Study medical procedures on full-body phantoms



- Expected resolution in X & Y: ≈ 1 mm
  (limited by crystal size and tracking precision)
- Expected resolution in Z: ≈ 5 mm
  (limited by low angular coverage)

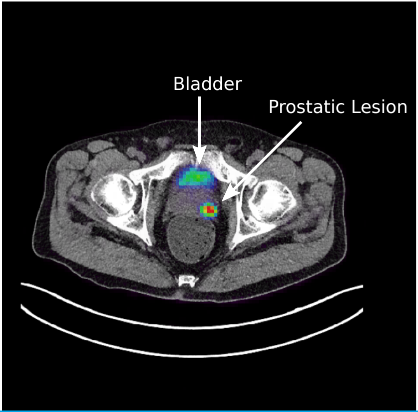


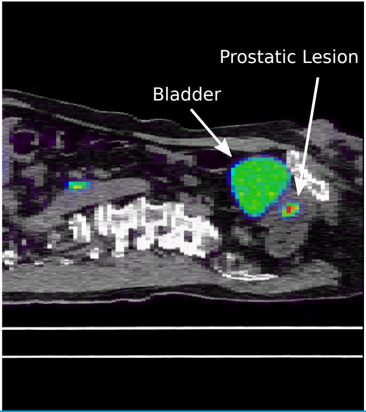


- Quantify image quality
- How good can we suppress background from other organs?
- How much data do we need to acquire?

Patient Data: PSMA.

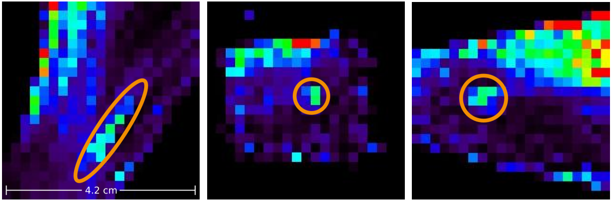
- PET/CT scans performed at TU Munich (Klinikum rechts der Isar)
- Prostate-specific membrane antigen (PSMA)
- One specific patient with injected dose of 140 MBq
- Bladder uptake ≈ 20 MBq, prostatic lesion ≈ 0.5 MBq
- Full-body PET resolution: ≈ 6 mm
- Downsize datasets: Merge voxels to speed up simulation
- Load PET (activity) and CT (scatter) DICOM images into the simulation



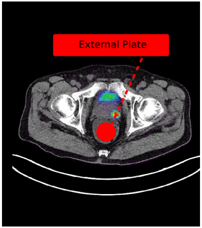


Patient Data: PSMA.

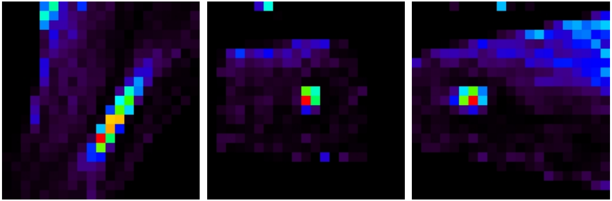
First reconstructed image of full-body PSMA PET data with the EndoTOFPET system



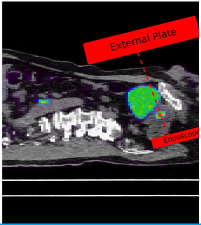
(a) Transverse (b) Coronal (c) Sagittal



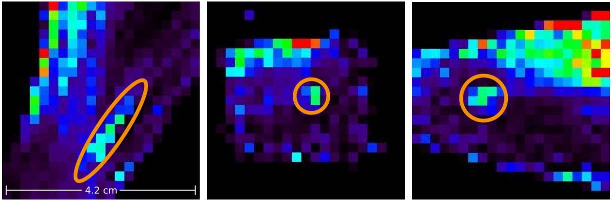
Increase lesion uptake $\times 10$:



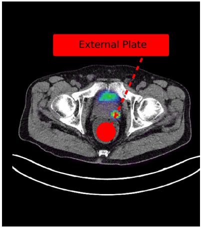
(a) Transverse (b) Coronal (c) Sagittal



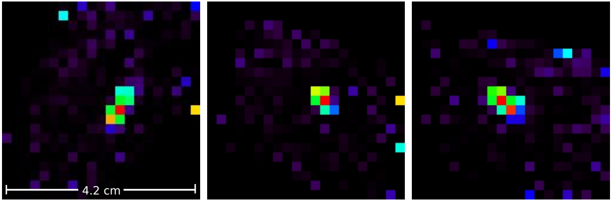
First reconstructed image of full-body PSMA PET data with the EndoTOFPET system



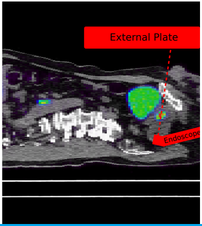
(a) Transverse (b) Coronal (c) Sagittal



Increase lesion uptake $\times 10$ and tilt the PET head:



(a) Transverse (b) Coronal (c) Sagittal



Summary.

Tomography

- Take projection data,convert it into cross-section images
- A projection is a line integral of an object
- Backprojection: sums data from all projection views

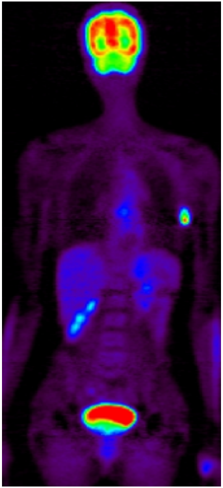
Analytic Reconstruction

- Simple backprojection → Blurring
- Use the Central Slice Theorem to solve the oversampling (blurring) in the Fourier space → **Filtered Backprojection**

Iterative Reconstruction

- Start with guess, compare with measurement, update the guess, iterate until best solution is reached
- Model scatter, attenuation and limited detector resolution to better reflect reality (e.g. **ML-EM**)

Thank you for your attention!



Literatur: [1, 2, 3, 4, 2, 4, 5, 6, 7, 8, 9, 10, 11, 12]

Literature I

J.L. Prince and J.M. Links.
Medical imaging signals and systems.
Pearson Prentice Hall, 2006.

G. Zeng.
Medical image reconstruction: A conceptual Tutorial.
Springer, 2010.

R. Brinks and T.M. Buzug.
Image reconstruction in positron emission tomography (pet): the 90th anniversary of radon's solution.
Biomedizinische Technik, 52(6):361–364, 2007.

A. Alessio and P. Kinahan.
Pet image reconstruction.
Nuclear Medicine, 1, 2006.

C. Grupen and I. Buvat.
Handbook of particle detection and imaging, volume 1.
Springer, 2011.

M. Slaney and A. Kak.
Principles of computerized tomographic imaging.
SIAM, Philadelphia, 1988.

Literature II

R.M. Leahy and J. Qi.
Statistical approaches in quantitative positron emission tomography.
Statistics and Computing, 10(2):147–165, 2000.

J.M. Ollinger and J.A. Fessler.
Positron-emission tomography.
Signal Processing Magazine, IEEE, 14(1):43–55, 1997.

L.A. Shepp and Y. Vardi.
Maximum likelihood reconstruction for emission tomography.
Medical Imaging, IEEE Transactions on, 1(2):113–122, 1982.

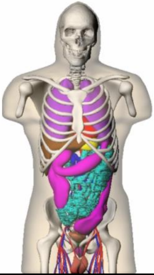
J. Beyerer and F. Puente León.
Die Radontransformation in der digitalen Bildverarbeitung.
at-Automatisierungstechnik, 50(10/2002):472, 2002.

J. Cui, G. Pratx, S. Prevrhal, and C.S. Levin.
Fully 3D list-mode time-of-flight PET image reconstruction on GPUs using CUDA.
Medical Physics, 38(12):6775, 2011.

A.J. Reader and H. Zaidi.
Advances in PET image reconstruction.
PET Clinics, 2(2):173–190, 2007.

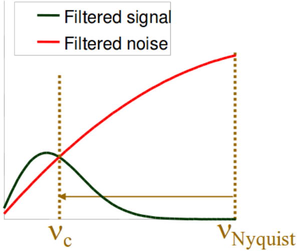
Spare Slides

4D NURBS XCAT* Cardiac Torso Phantom.

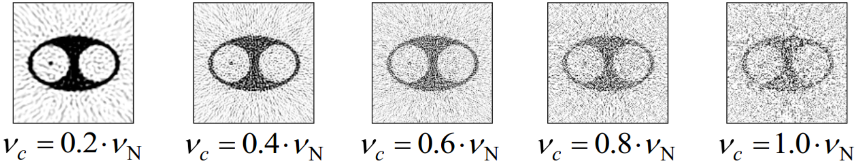


- Realistic model of human anatomy incl. cardiac and respiratory patient motion
- Place Tumor in Prostate, inject Radiopharmacon

Frequency cut-off.



Do not reconstruct high spatial frequencies



Interlude: Fast Fourier Transform (FFT).

- Discrete FTs are computationally very expensive. Solution: FFT
- FT: $\mathcal{O}(N^2)$, FFT: $\mathcal{O}(N \log_2 N)$ (2 weeks vs. 30 s CPU time)
- FT of length N can be written as sum of 2 FTs of length $N/2$, one for even (e) and one for odd (o) parts:

$$
\begin{aligned}
F_k = \sum_{j=0}^{N-1} f_j e^{-\frac{2\pi i}{N} jk} &= \sum_{j=0}^{N/2-1} f_{2j} e^{-\frac{2\pi i}{N}(2j)k} + \sum_{j=0}^{N/2-1} f_{2j+1} e^{-\frac{2\pi i}{N}(2j+1)k} \\
&= F_k^{(e)} + W^k F_k^{(o)} \quad , \quad W^k = \exp(2\pi i k/N) \ \text{(root of unity)} \\
&= F_k^{(e\ldots e)} + \ldots + W^{\cdots} F_k^{(eoo\ldots o)} + \ldots + W^{pk} F_k^{(o\ldots o)}
\end{aligned}
$$

- Divide data all the way down to *FTs of length 1*.
- FT of length 1 is function value itself
- $F_k^{(eoee\ldots o)} = f_n$ for some $n \rightarrow$ Each combination (eoeoo...o) corresponds to an element of the input vector f.

Interlude: Fast Fourier Transform (FFT).

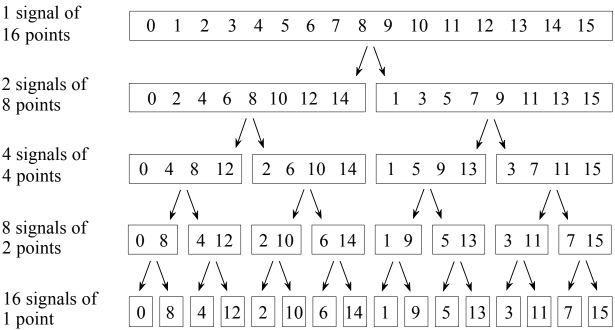


Figure: An N point signal is decomposed into N signals à 1 point

- Reverse the pattern ($eoee$...), assign binary number ($e = 0$, $o = 1$)
- E.g.: $(eoeo) \rightarrow (oeoe) = (1010) = 10$
- Combine adjacent elements (with the W^{nk}'s) to get 2-point trafo, etc

Interlude: Fast Fourier Transform (FFT).

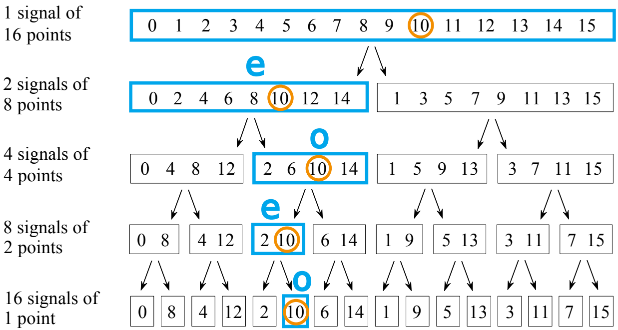
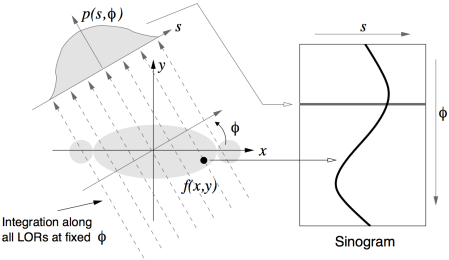
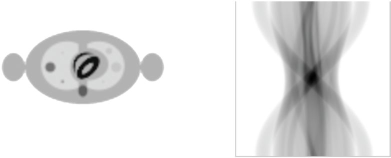


Figure: An N point signal is decomposed into N signals à 1 point

- Reverse the pattern ($eoee$...), assign binary number ($e = 0$, $o = 1$)
- E.g.: ($eoeo$) $\rightarrow$ ($oeoe$) $=$ (1010) $=$ 10
- Combine adjacent elements (with the W^{nk}'s) to get 2-point trafo, etc





- Sinogram represents the projection of the radiopharmacon distribution onto the detector
- Fixed point in the object traces a sinusoidal path in projection space
- Sinogram: Superposition of all sinusoids corresponding to each point of activity in the object
- LORs described by $p(s, \phi)$
- $f(x, y) \rightarrow p(s, \phi)$: Radon transform
- The task of tomography is to find $f(x, y)$ from $p(s, \phi)$

Some examples.

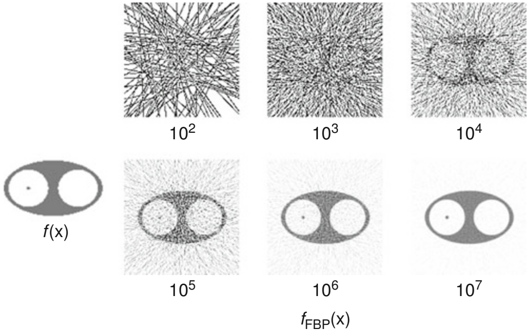
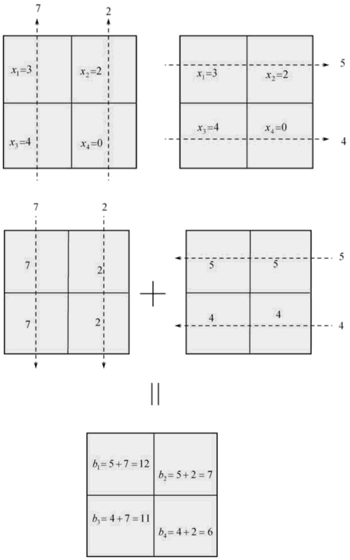


Figure: The effect of the number of detected photons on the quality of the reconstructed image. A synthetic image was projected and Poisson noise was added to the projections. Reconstructed with the FBP algorithm

Noise has a high impact on the image quality

Backprojection: Blurring of the image.



- Assign equal intensity along each LOR
- The projections are formed one view at a time
- Sum up the LORs
- The final backprojected image is the summation of the backprojections from all views
- Backprojected image $\neq$ original image
- Solution? For this we need some math...

Backprojection.

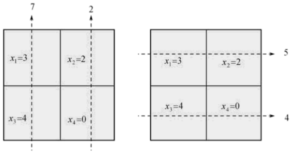
Discrete case

Projection: $P = AX$

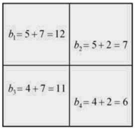
Backprojected image: $B = A^T P$

$$X = (x_1, x_2, x_3, x_4)^T$$

$$P = \big(p(1,0°), p(2,0°), p(1,90°), p(2,90°)\big)^T$$
$$= (7,2,5,4)^T$$



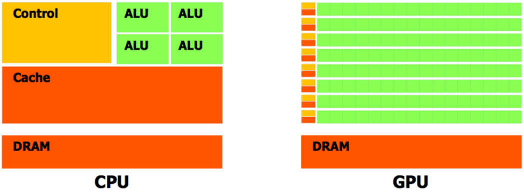
$$B = A^T P = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}^T \begin{pmatrix} 7 \\ 2 \\ 5 \\ 4 \end{pmatrix} = \begin{pmatrix} 12 \\ 7 \\ 11 \\ 6 \end{pmatrix}$$



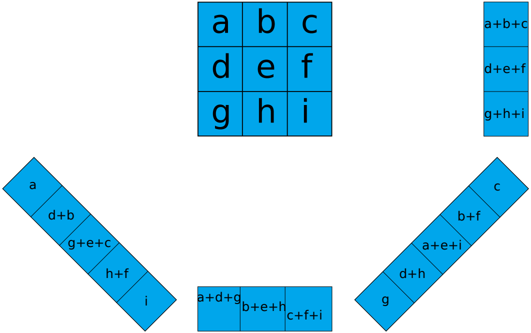
Interlude: GPU Programming.

ML-EM is much slower than FBP. But this is not a real problem nowadays...

- Parallelise your work
- 384 cores, 32768 registers/block
- Each one can perform simple tasks very quickly



Backprojection Numerical Example.



$$e \to 3e + \sum \text{proj} \Rightarrow e \to e$$

$$a \to a + b + c + a + e + i + a + d + g + a = 3a + \sum \text{proj} - f - h$$

$$\Rightarrow a \to a - 1/3(f + h)$$

Blurring of the image!

Filtered Backprojection.

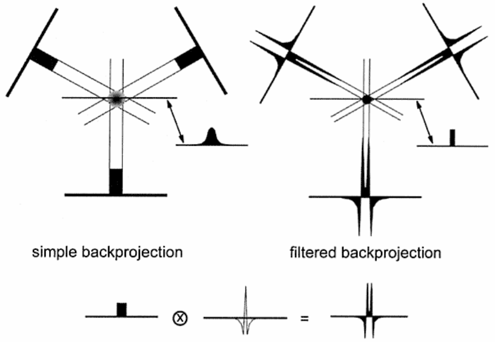
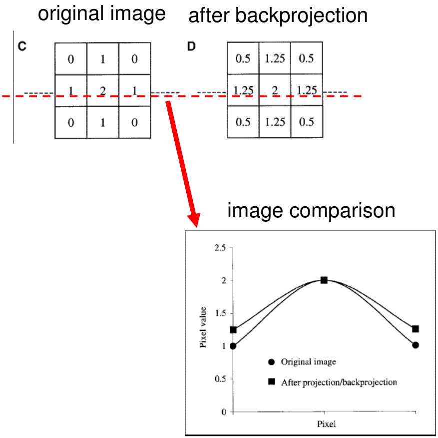


FIGURE 13-28. Simple backprojection is shown on the left; only three views are illustrated, but many views are actually used in computed tomography. A profile through the circular object, derived from simple backprojection, shows a characteristic 1/r blurring. With filtered backprojection, the raw projection data are convolved with a convolution kernel and the resulting projection data are used in the backprojection process. When this approach is used, the profile through the circular object demonstrates the crisp edges of the cylinder, which accurately reflects the object being scanned.

Backprojection: Blurring.



$1/r$ blurring of the image!

Central Slice Theorem.

Central Slice Theorem (CST)

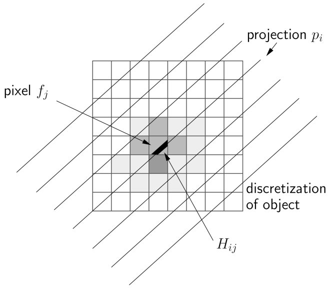
$$\mathcal{F}_1\{p(s,\phi')\} = \mathcal{F}_2\{f(x,y)\}_{|\phi=\phi'}$$

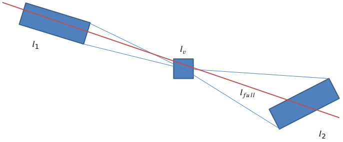
- The CST relates the Fourier transform (FT) of $f(x,y)$ with the FT of $p(s;\phi)$
- Summation of lines causes duplication in the center
- High density of central slices in Fourier space
- Oversampling in the center of the Fourier space needs to be filtered in order to have equal sampling throughout the Fourier space

A *complete* set of 1D projections allows the reconstruction of the *original* 2D distribution without loss of information

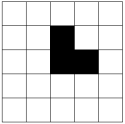
Iterative Methods - Basic Idea.

1. Image Model:
   - Discretisation of image into N distinct voxels f_j
2. System Model:
   - System matrix H_{ij}: probability that an emission from voxel j is detected in projection i: $\overline{p}_i = \sum_{j=1}^{N} H_{ij} f_j$
3. Data Model:
   - Variation of projection measurements around expected mean values (Poisson). Derived from our basic understanding of the acquisition process
4. Governing Principle:
   - Defines the 'best' image.
   - e.g. Maximum Likelihood cost function
5. Algorithm:
   - Optimises the cost function: Maximum Likelihood - Expectation Maximisation





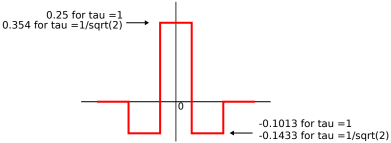
FBP Numerical Example.

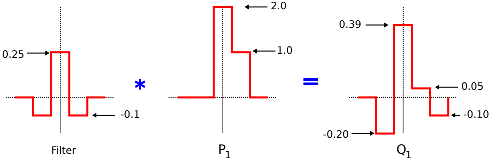


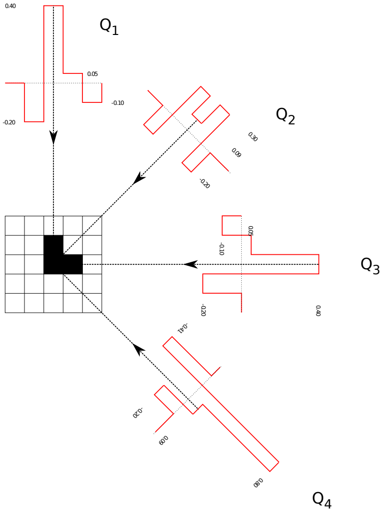
- black = 1, white = 0
- For projections at 0° and 90° the spacial resolution is one pixel (i.e. $\tau = 1$), for the other projection it is $\sqrt{2}$ pixels.

- Convolute filter with projection

  (Product in Fourier space = convolution in position space)
- Discrete convolution: $(f * g)(t) = \sum_k f(k)g(t-k)$







Push filtered projections back into image

	-0.20	0.40	0.05	-0.10
	-0.20	0.40	0.05	-0.10
	-0.20	0.40	0.05	-0.10
	-0.20	0.40	0.05	-0.10
	-0.20	0.40	0.05	-0.10

Q_1 back projected

		-0.20	0.30	0.09
	-0.20	0.30	0.09	0.30
-0.20	0.30	0.09	0.30	-0.20
0.30	0.09	0.30	-0.20	
0.09	0.30	-0.20		

Q_2 back projected

-0.01	0.50	-0.31	0.25	-0.11
-0.15	-0.26	1.55	-0.22	0.25
0.20	0.30	0.98	1.55	-0.31
0.10	-0.31	0.30	-0.26	0.50
0.09	0.10	0.20	-0.15	-0.01

$Q_1 + Q_2 + Q_3 + Q_4$

-0.10	-0.10	-0.10	-0.10	-0.10
0.05	0.05	0.05	0.05	0.05
0.40	0.40	0.40	0.40	0.40
-0.20	-0.20	-0.20	-0.20	-0.20

Q_3 back projected

0.09	0.80	-0.41		
-0.20	0.09	0.80	-0.41	
	-0.20	0.09	0.80	-0.41
		-0.20	0.09	0.80
			-0.20	0.09

Q_4 back projected

-0.01	0.39	-0.24	0.20	-0.09
-0.12	-0.20	1.22	-0.17	0.20
0.16	0.24	0.77	1.22	-0.24
0.08	-0.24	0.24	-0.20	0.39
0.07	0.08	0.16	-0.12	-0.01

$\frac{\pi(Q_1 + Q_2 + Q_3 + Q_4)}{4}$

0	0	0	0	0
0	0	1	0	0
0	0	1	1	0
0	0	0	0	0
0	0	0	0	0

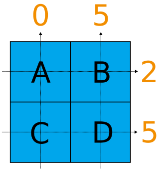
round

- Usually do filtering in Fourier space.
- FT computationally expensive → use FFT (2 weeks vs. 1 min. CPU time)

Basic Idea.

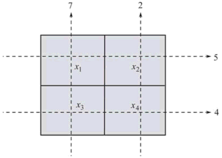
Example: 2×2 Matrix

Sum of 1st row is 2, sum of 2nd row is 3, sum of 1st column is 0, sum of 2nd column is 5.
$\Rightarrow x_1 = 0,\ x_2 = 2,\ x_3 = 0,\ x_4 = 3$



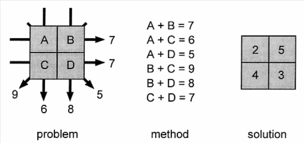
Example: Another 2×2 Matrix

Sum of 1st row is 5, sum of 2nd row is 4, sum of 1st column is 7, sum of 2nd column is 2.
$\Rightarrow$ No unique solution.



Way out: Take more views?

Hypothesis: With many views from many angles, we are able to perfectly reconstruct the image

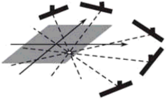


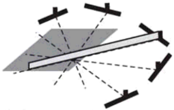
Backprojection.

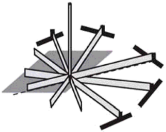
Summary

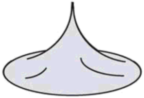
- Tomography is a process of taking projection data and converting the data into cross-section images. Projection data from multiple views are required
- A projection is a line integral of an object
- Backprojection is a superposition procedure and it sums the data from all projection views. Backprojection evenly distributes the projection domain data back along the same lines from which the line-integrals were formed.

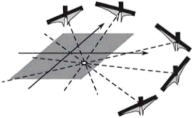
Eliminate blurring by introducing negative “wings” around the spike in the projections before backprojecting

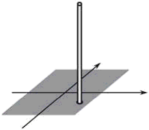

(a) Project a point source


(b) Backproject from one view


(c) Backproject from a few views


(d) Backproject from all views


(e) Add negative wings


(f) Backproject modified data

BP vs. FBP.

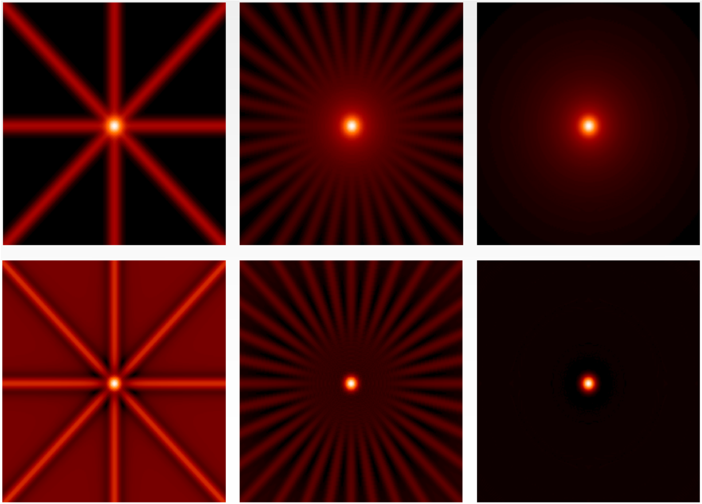


Figure: Reconstruction of a pointsource with simple backprojection (top) and filtered Backprojection (bottom)