

# Algorithm for Three Dimensional Reconstruction in Medical Imaging

Peter B. Noël, PhD



i. Reconstruction of 3D Vascular Data from Multiple Angiographic Views



i. Geometric Object Aware Tomography



ii. Hyperfast Computations in Medical Imaging by using Graphics Processing Unit



iii. Radiation Dose Evaluation in CT

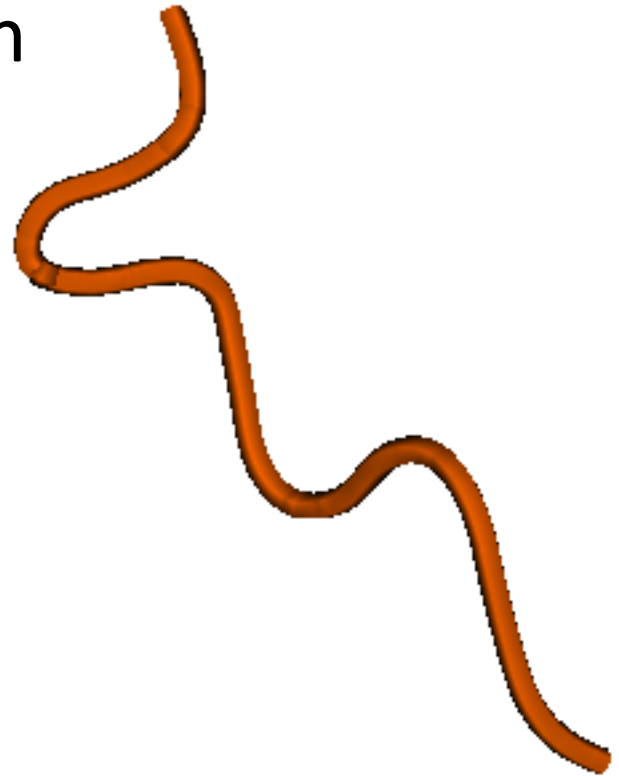


iv. MRT Fat Segmentations as Lifestyle Project

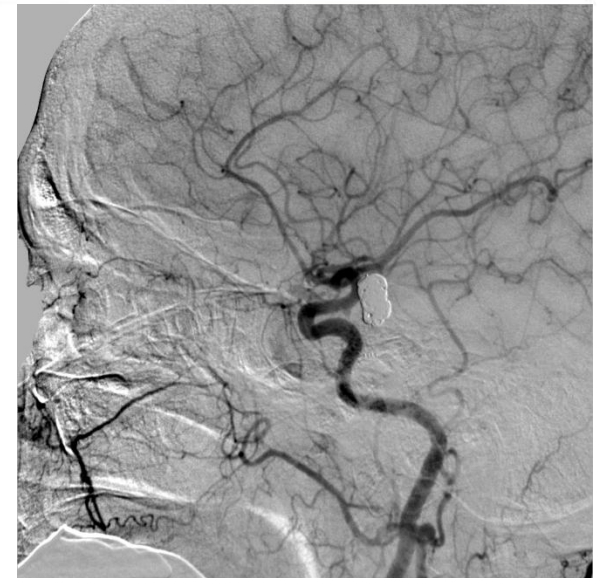
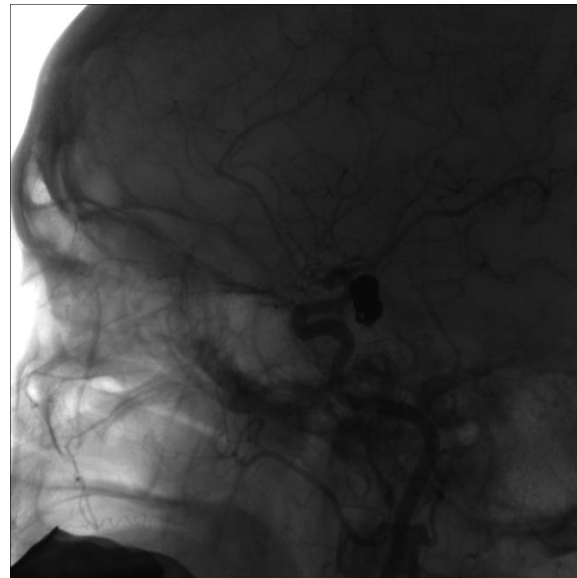
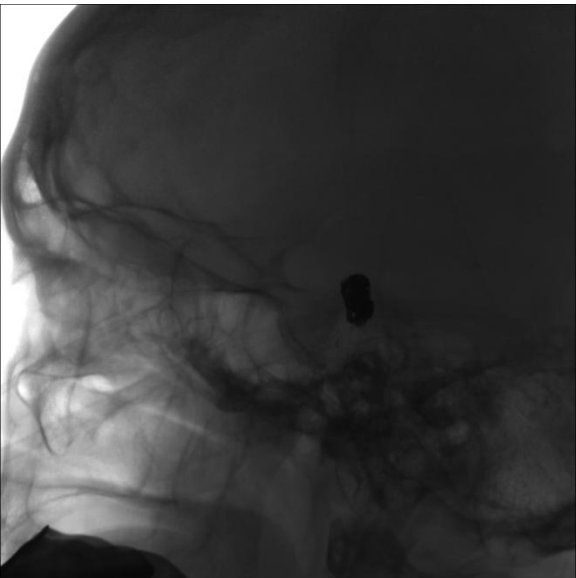


# Reconstruction of 3D Vascular Data from Multiple Angiographic Views

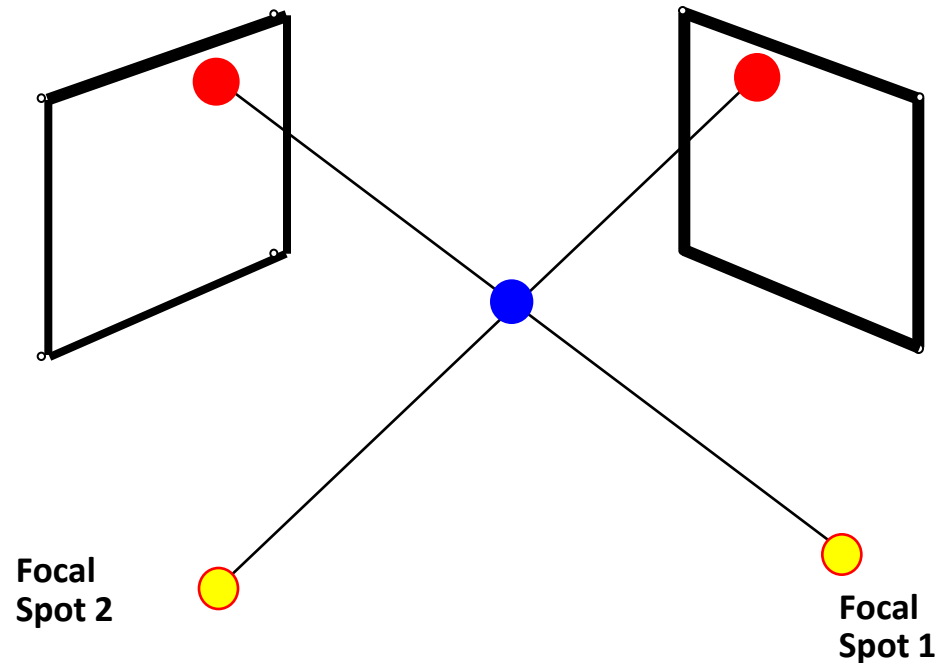
- Vascular diseases, such as stroke, are among the leading causes of death
- Calculation and presentation of 3D vessel information may reduce complications



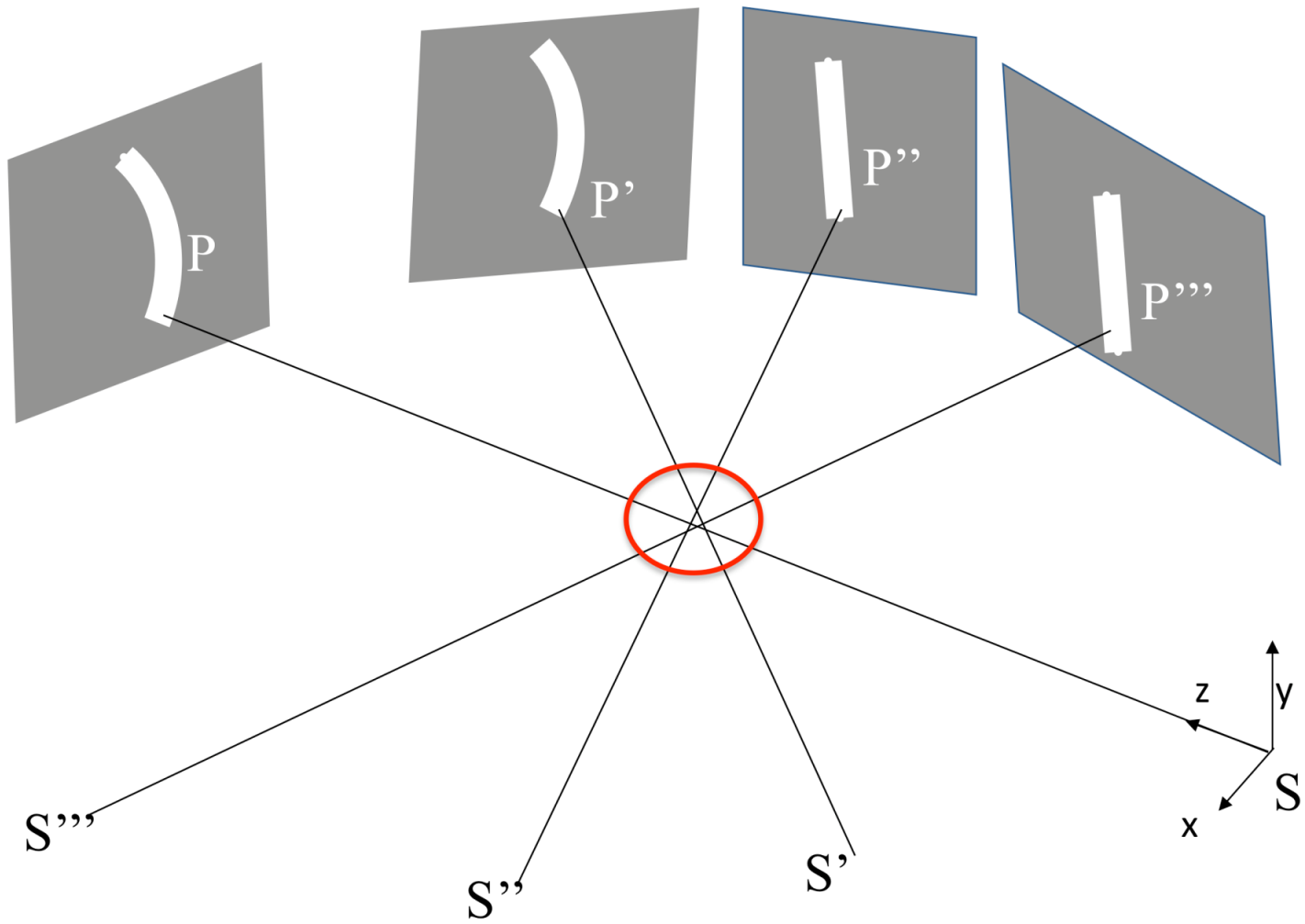
- Visualization of blood vessels
- Done by injecting a contrast agent into the blood vessel and imaging using x-ray systems.



- If you know the biplane imaging geometry and corresponding points in the two images, you can triangulate to get the 3D positions



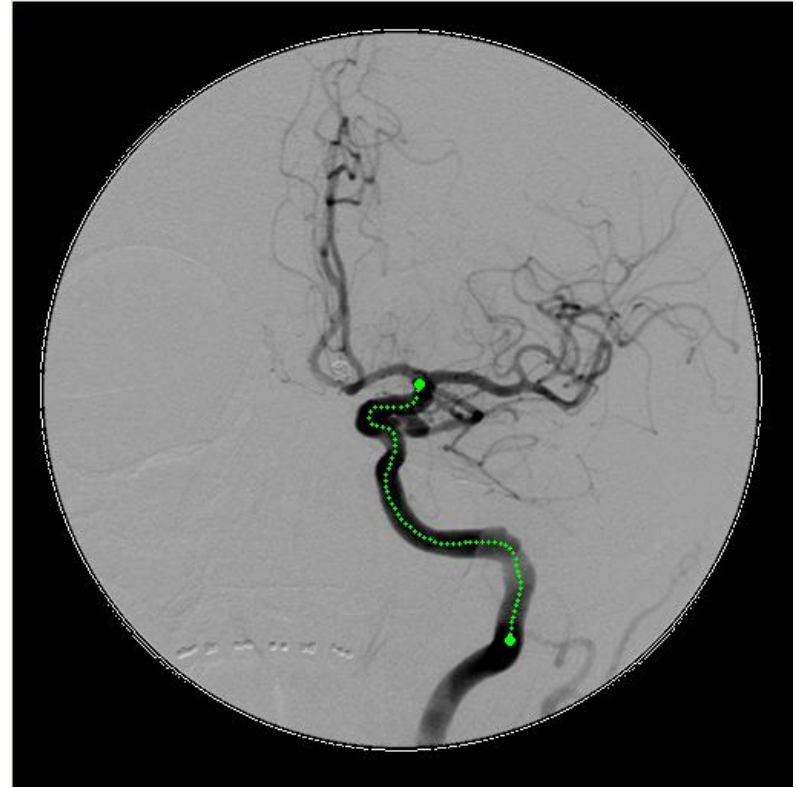
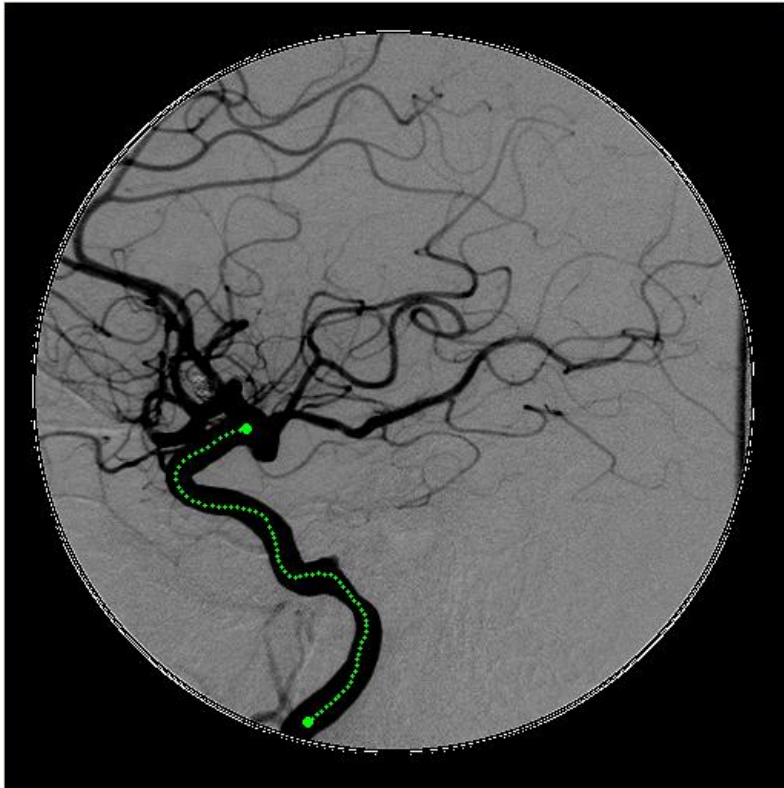
- Using gantry information:
  - The 2D correspondences along the vessel centerline can be incorrect
  - the 3D vessel centerlines vary in shape and absolute location
- With refinement using self-calibrating two-view techniques
  - Shape and location improve, but vessel orientation can be a problem, e.g., end-on





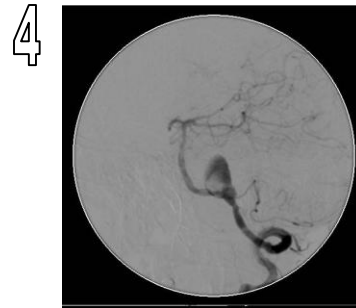
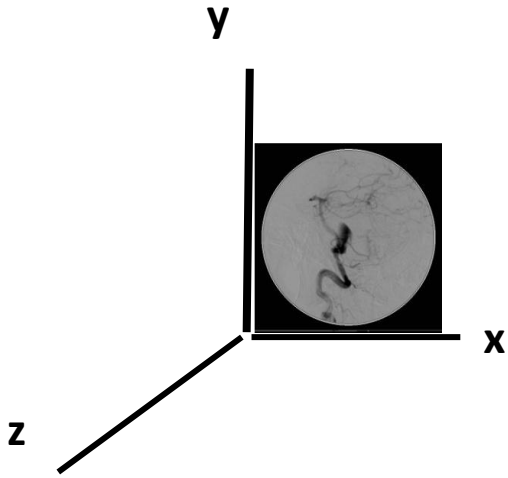
Multiple angiograms are usually acquired during neurovascular interventions.

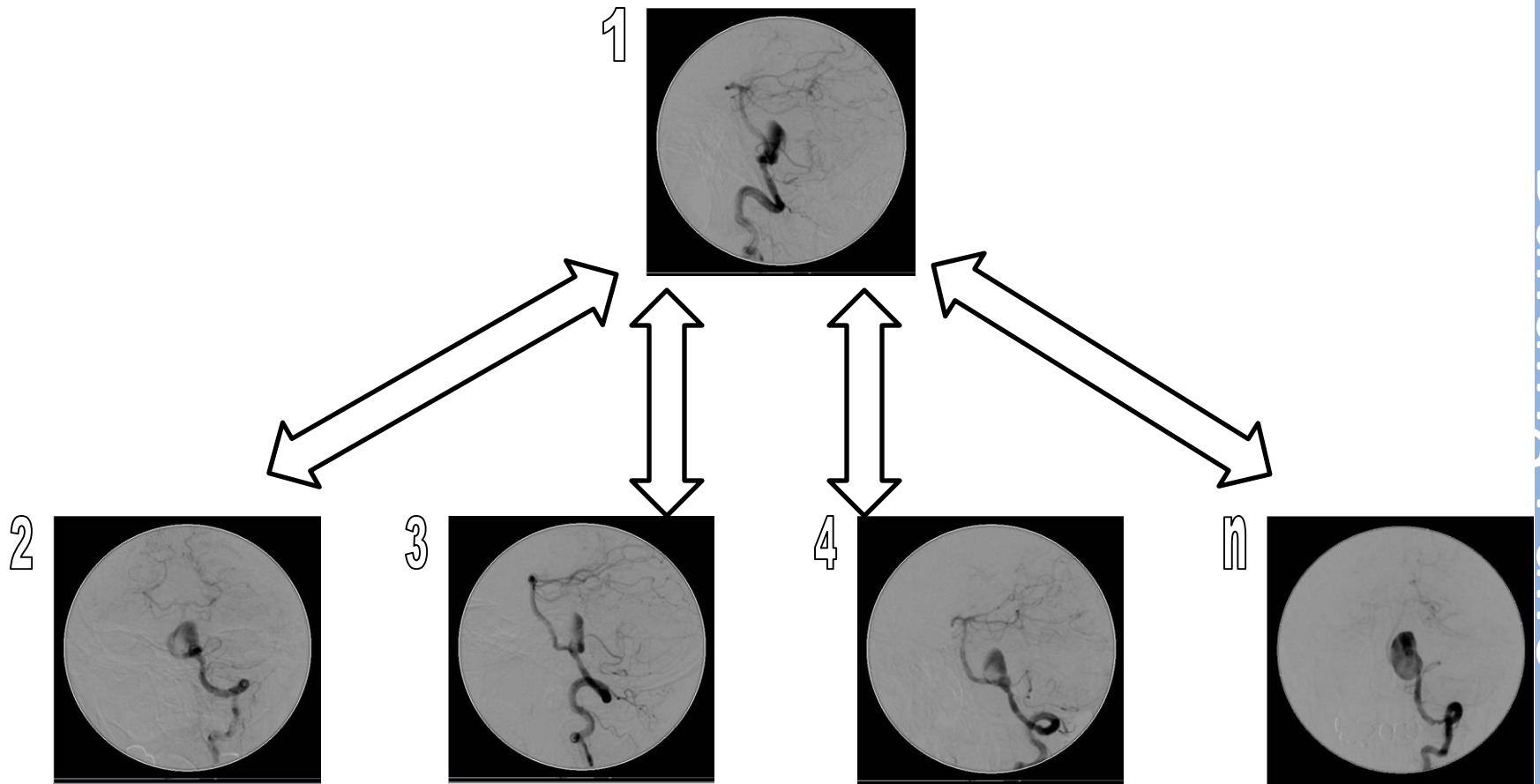
Therefore, we use **all angiographic views** to generate reliable 3D during the procedure.

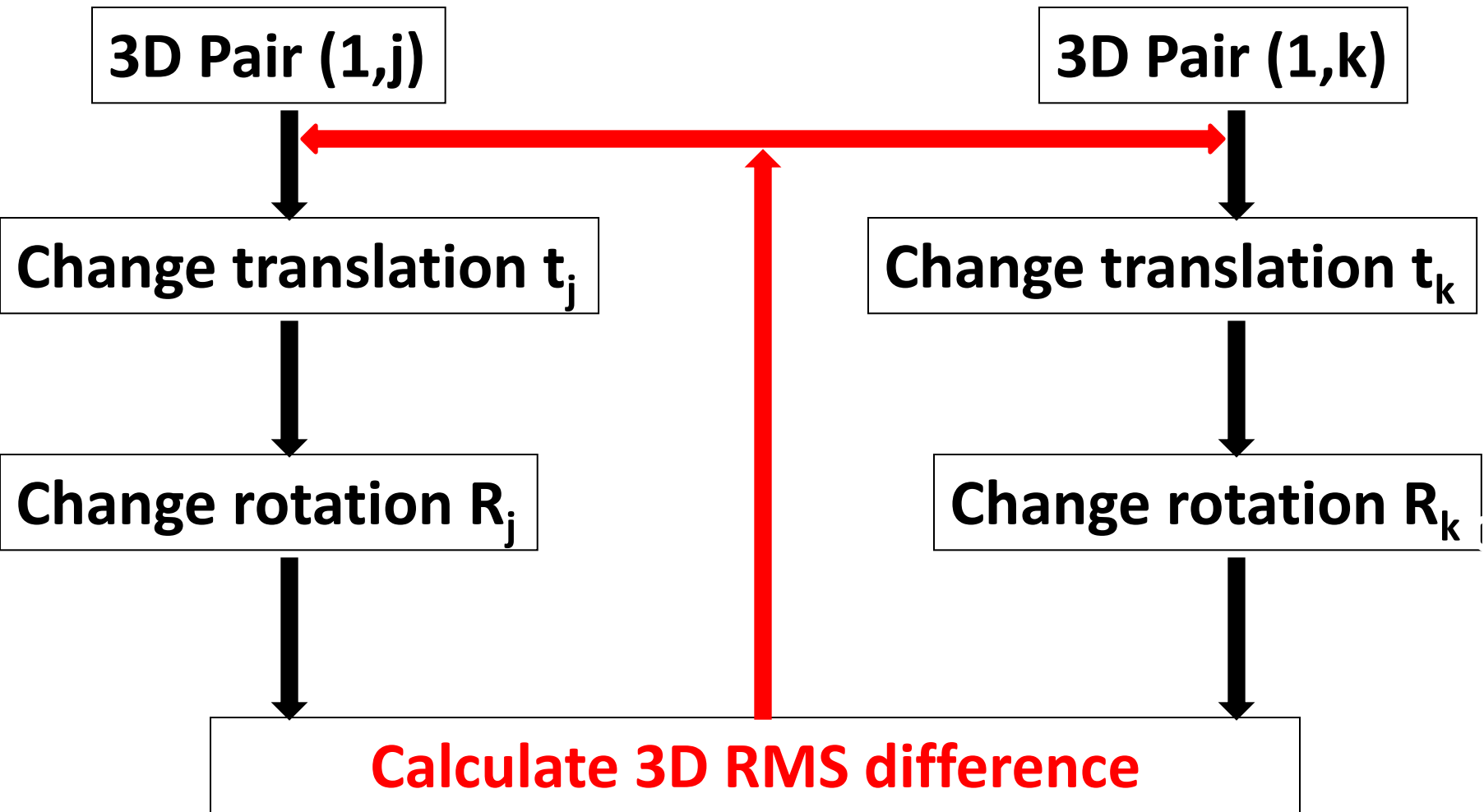


- Angiographic images from each view are selected.
- The centerlines are indicated and fit.

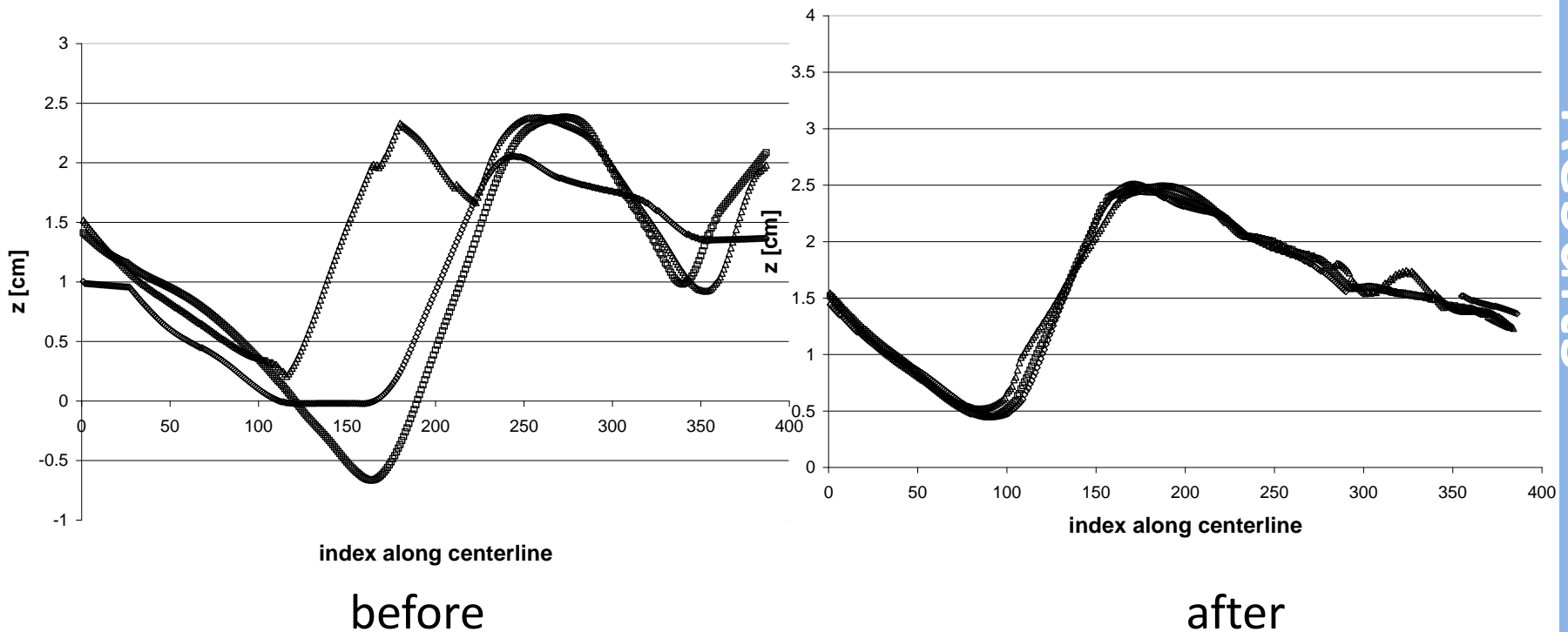
# Common Coordinate system

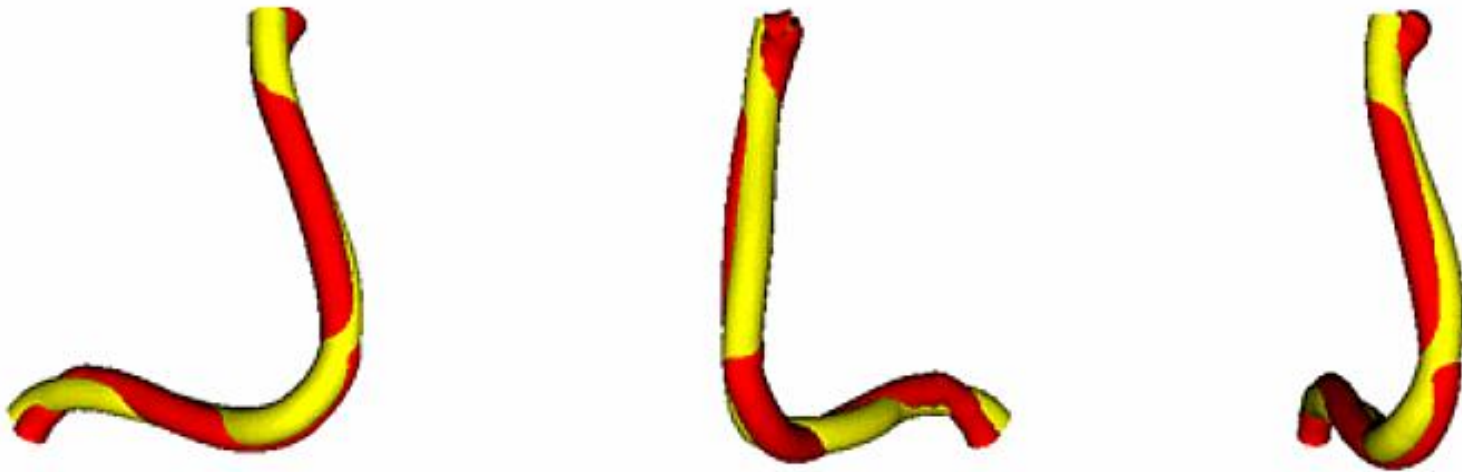




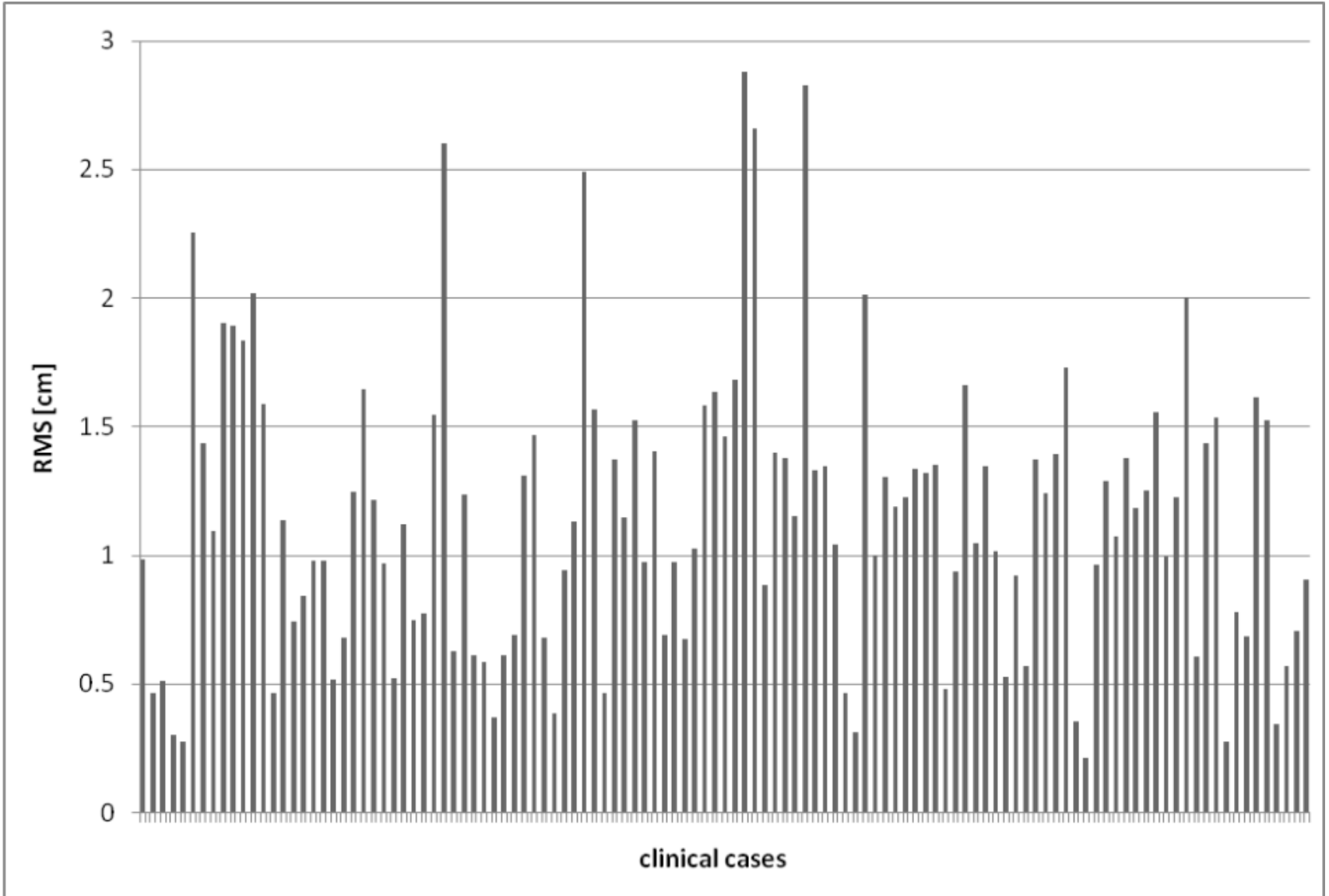


# Z-coordinates before and after optimization

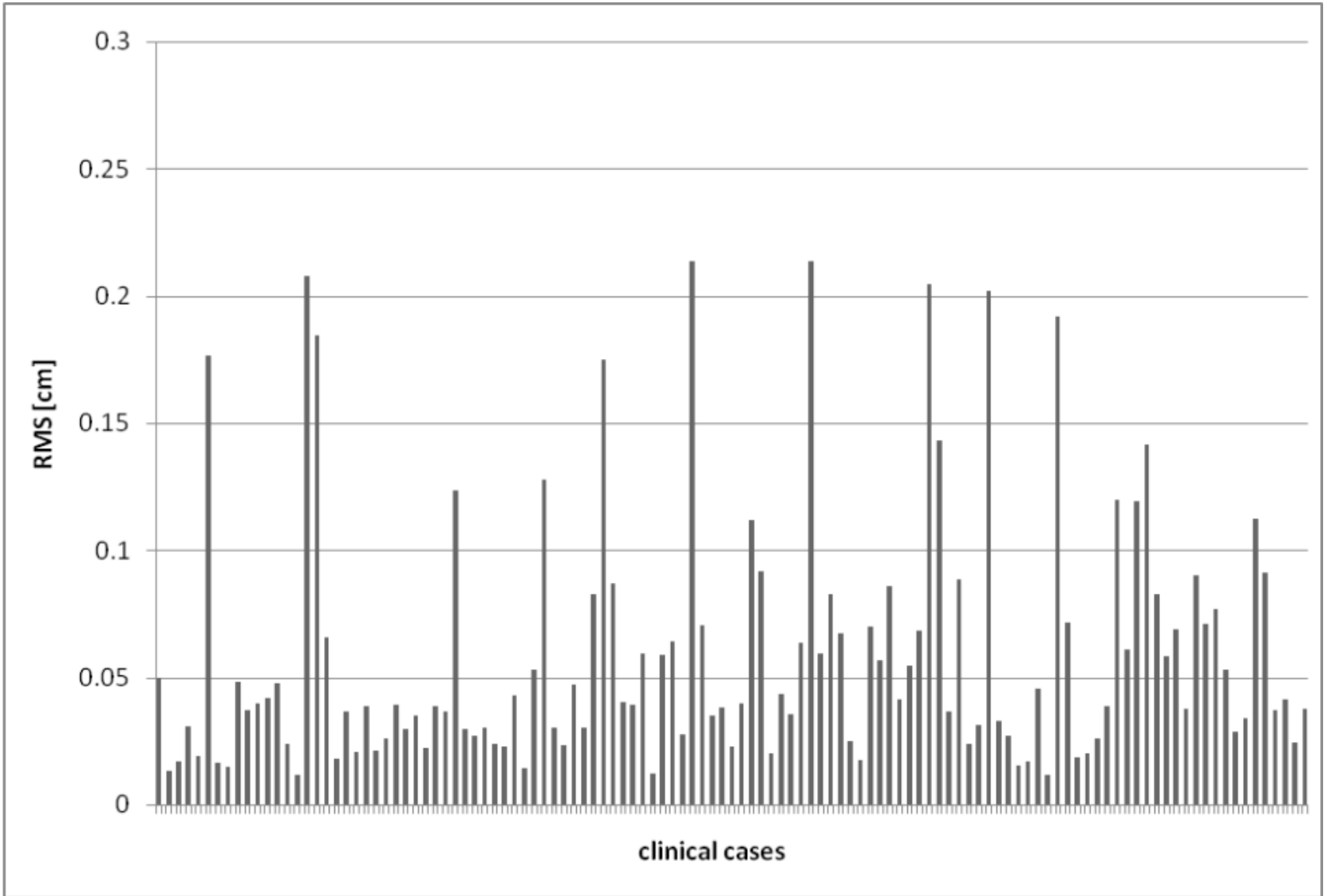




Comparison between a rotational angiographic system (red), which used 220 projection images and our multiple-view techniques (yellow), which used 4 projection images. The two vessels are almost identical, and have an overlap of 99.73%.

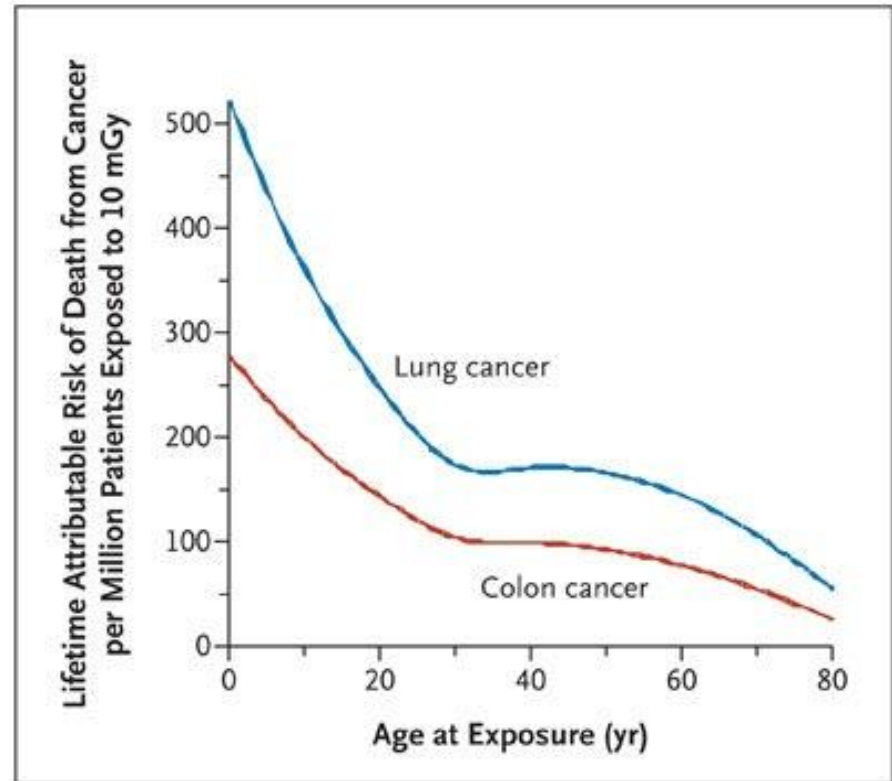
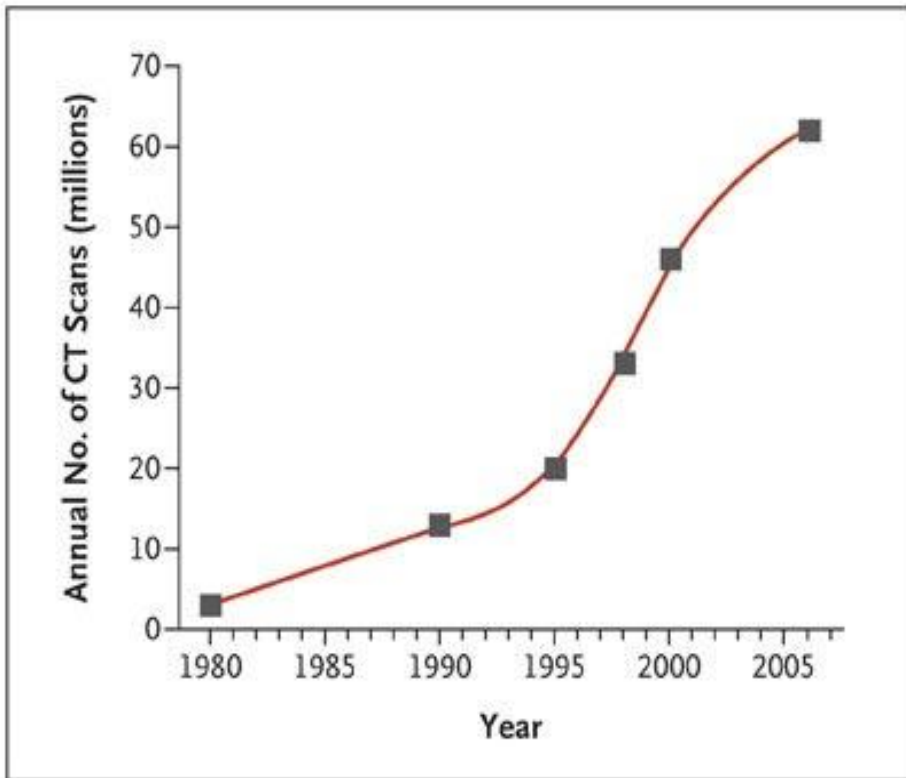




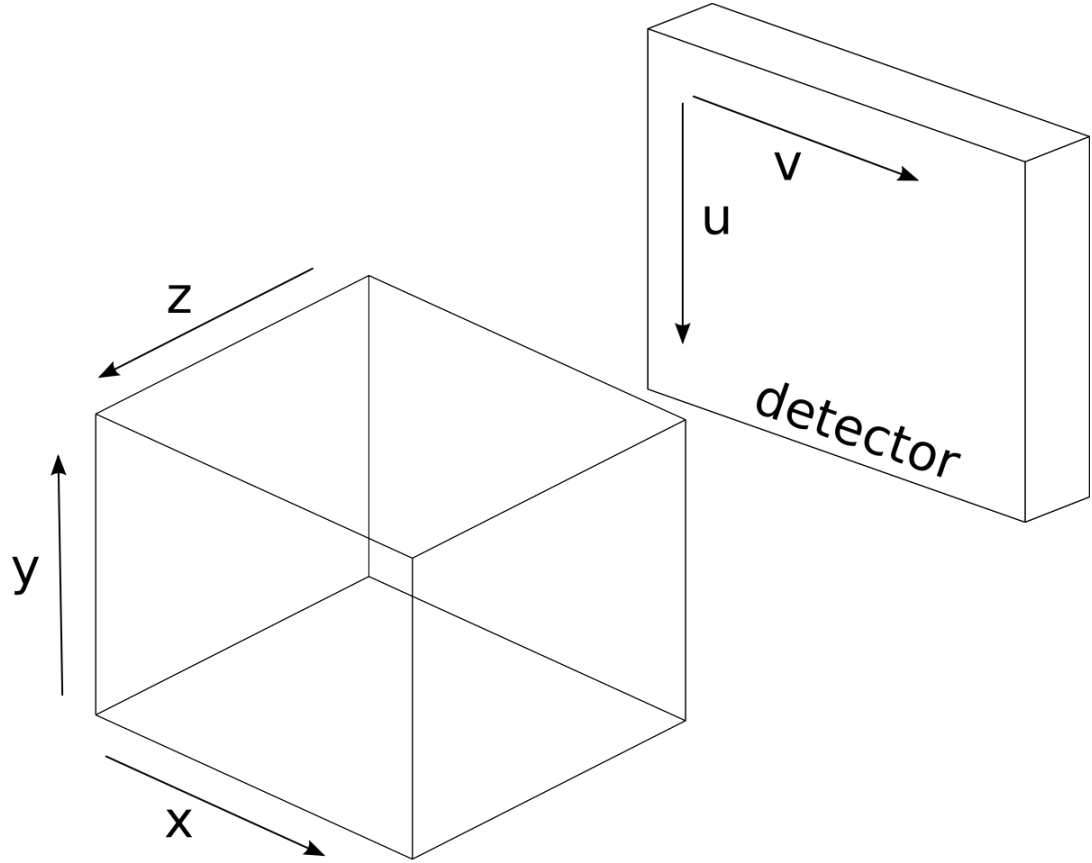


# Limited – View Cone Beam Computed Tomography

- Dose is a Problem in CT
  - abdominal CT scan  $\sim 12.5\text{mGy}$

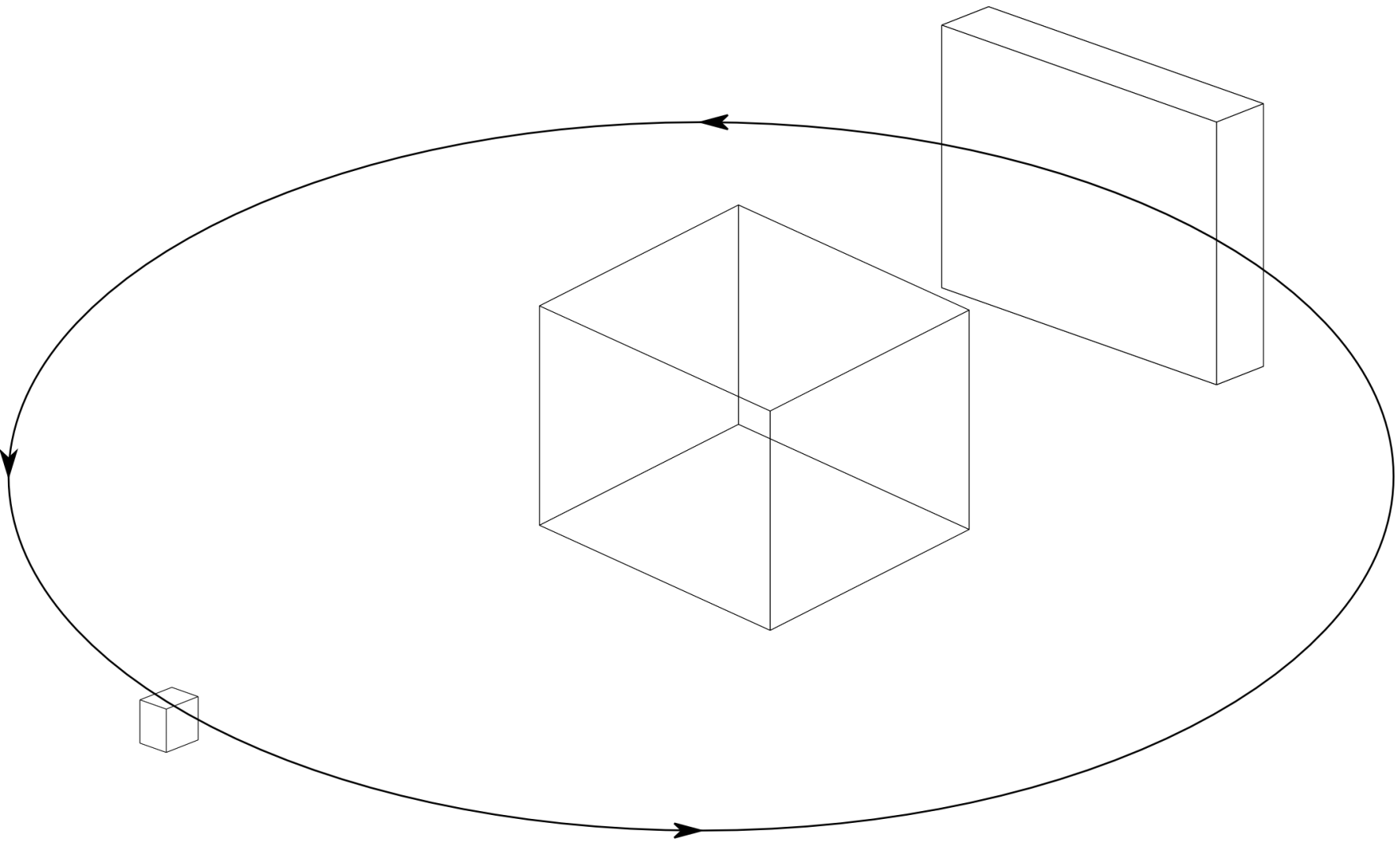


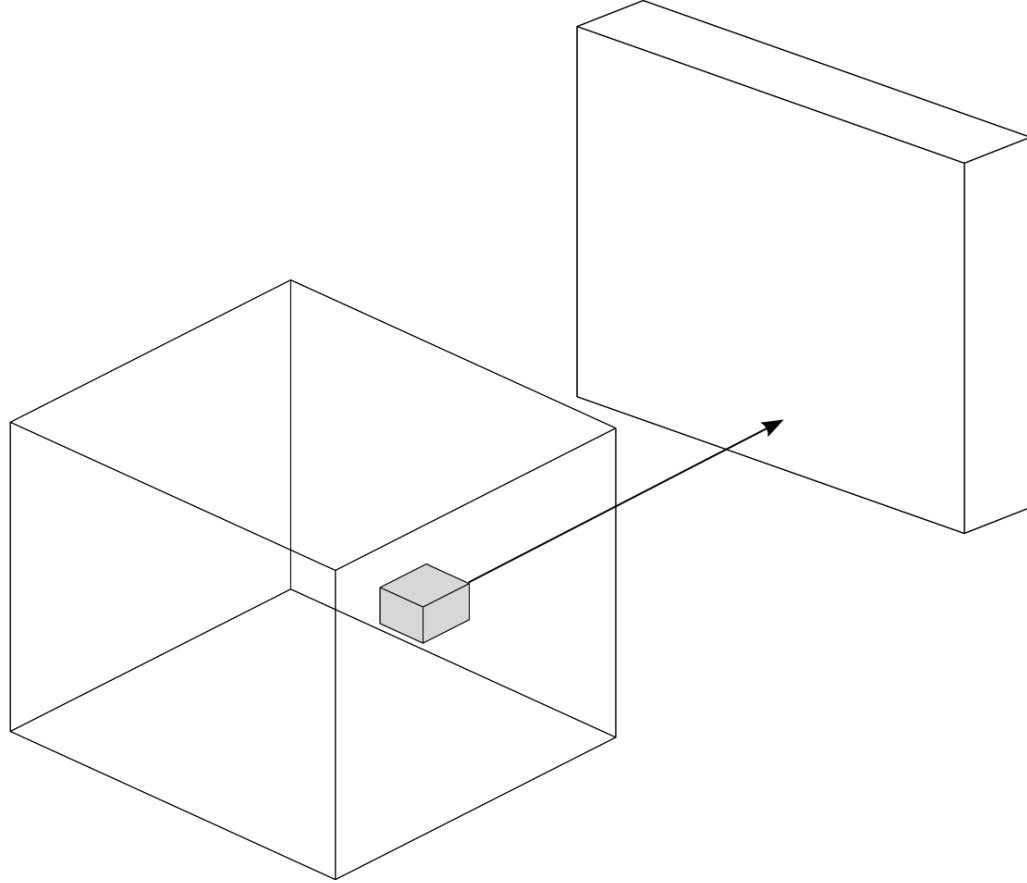
Reference: Brenner et al. 2008



 source

# Cone-Beam CT





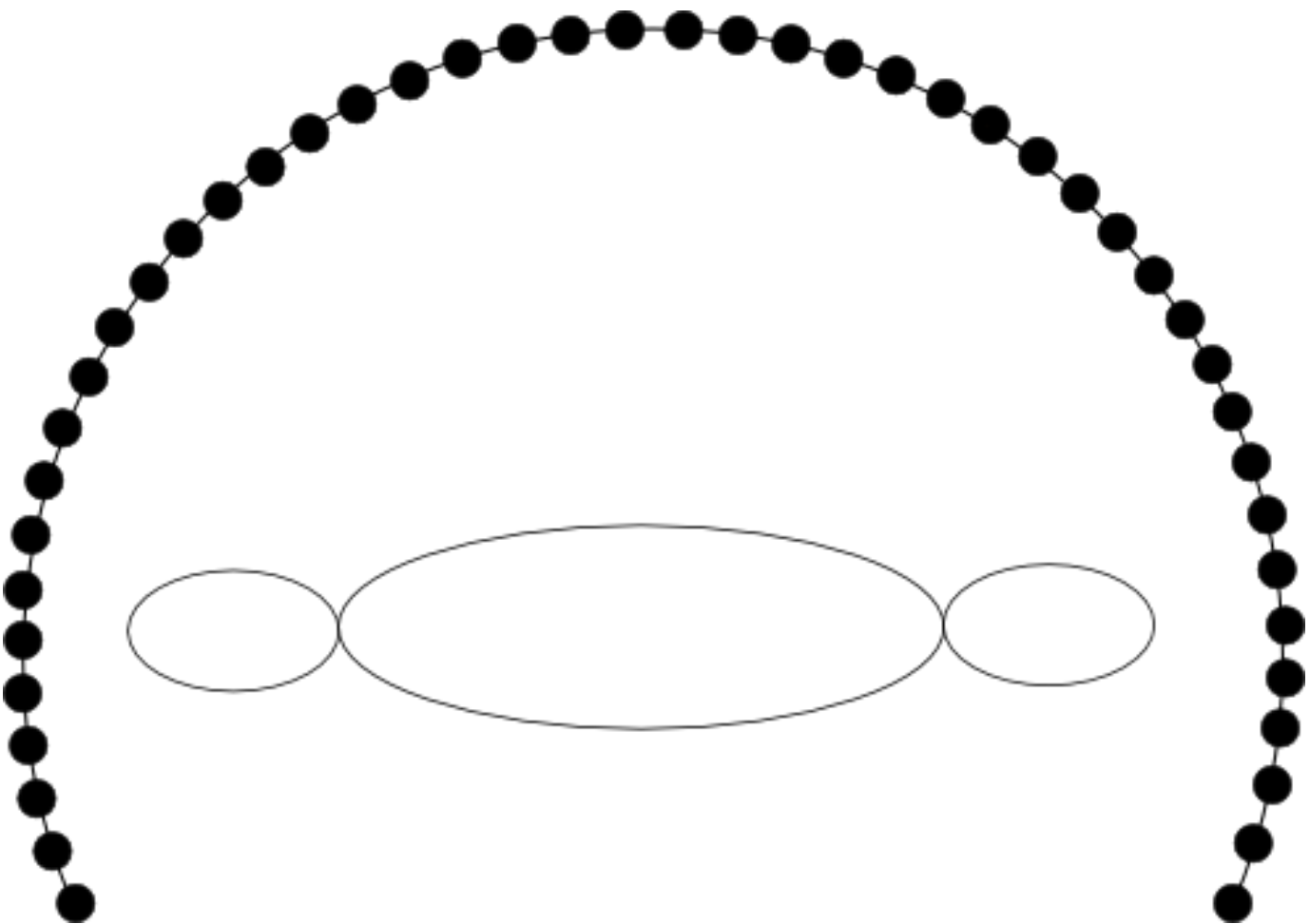
$$f(x, y, z) = \sum_{\phi} p_f [u(x, y, \phi), v(x, y, \phi), \phi]$$

$$u = (SID * x) / (ISO - z)$$

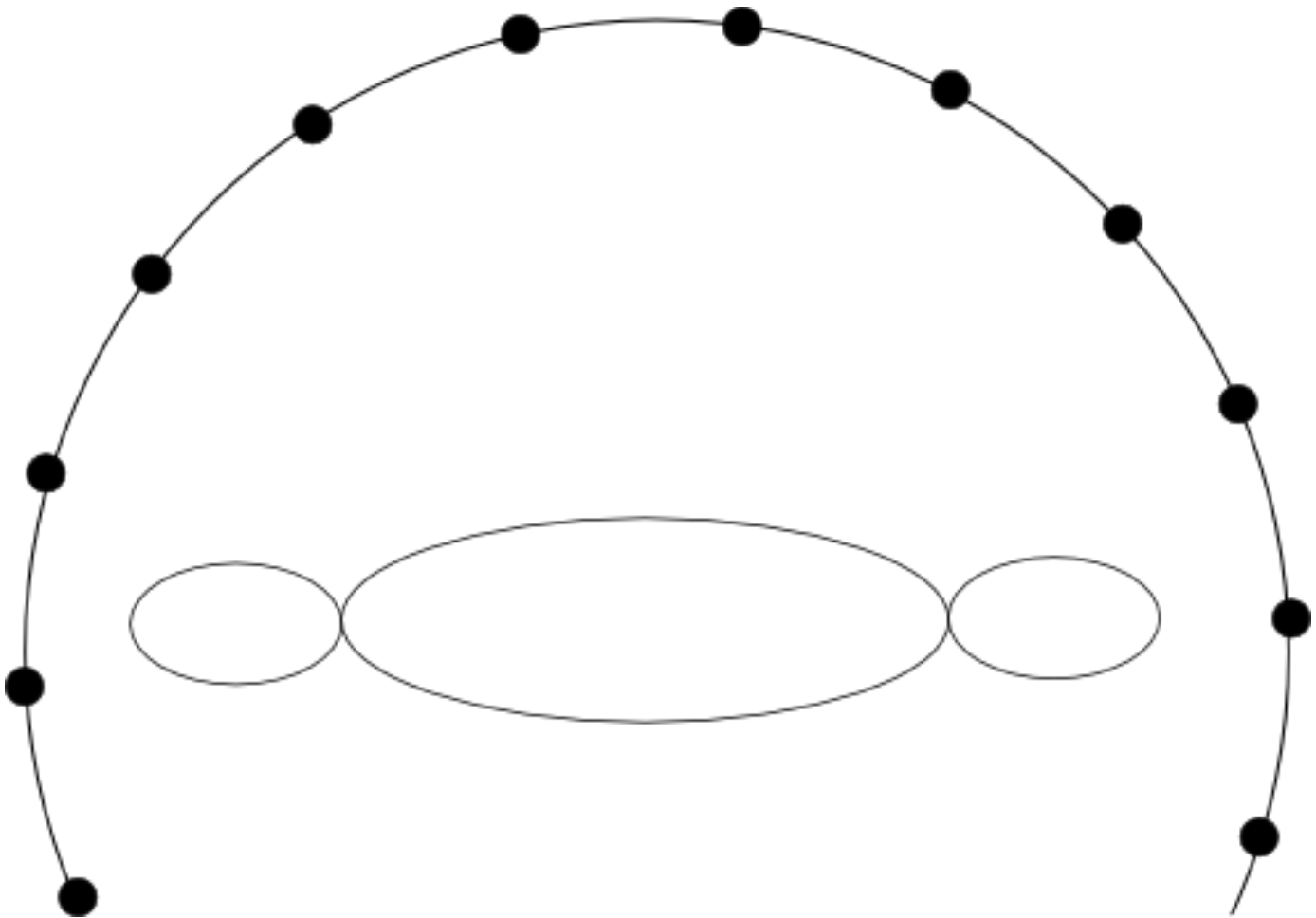
$$v = (SID * y) / (ISO - z)$$

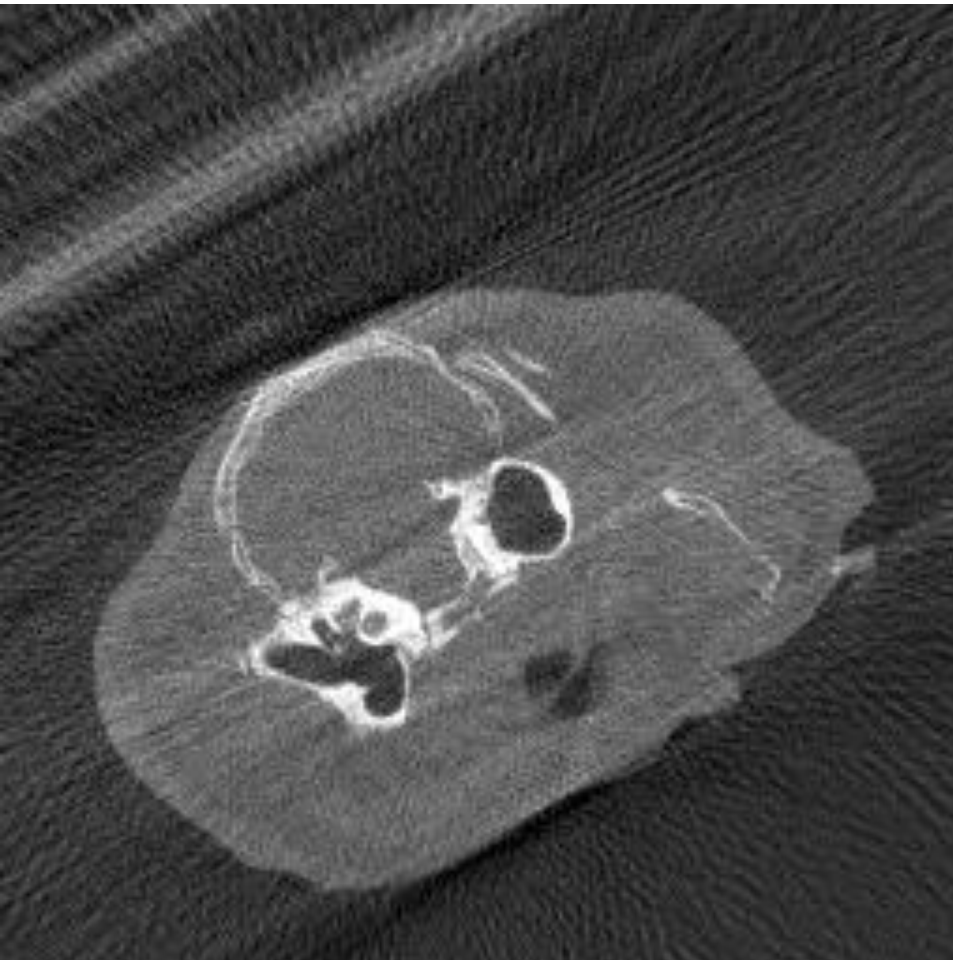
# Filtered Backprojection



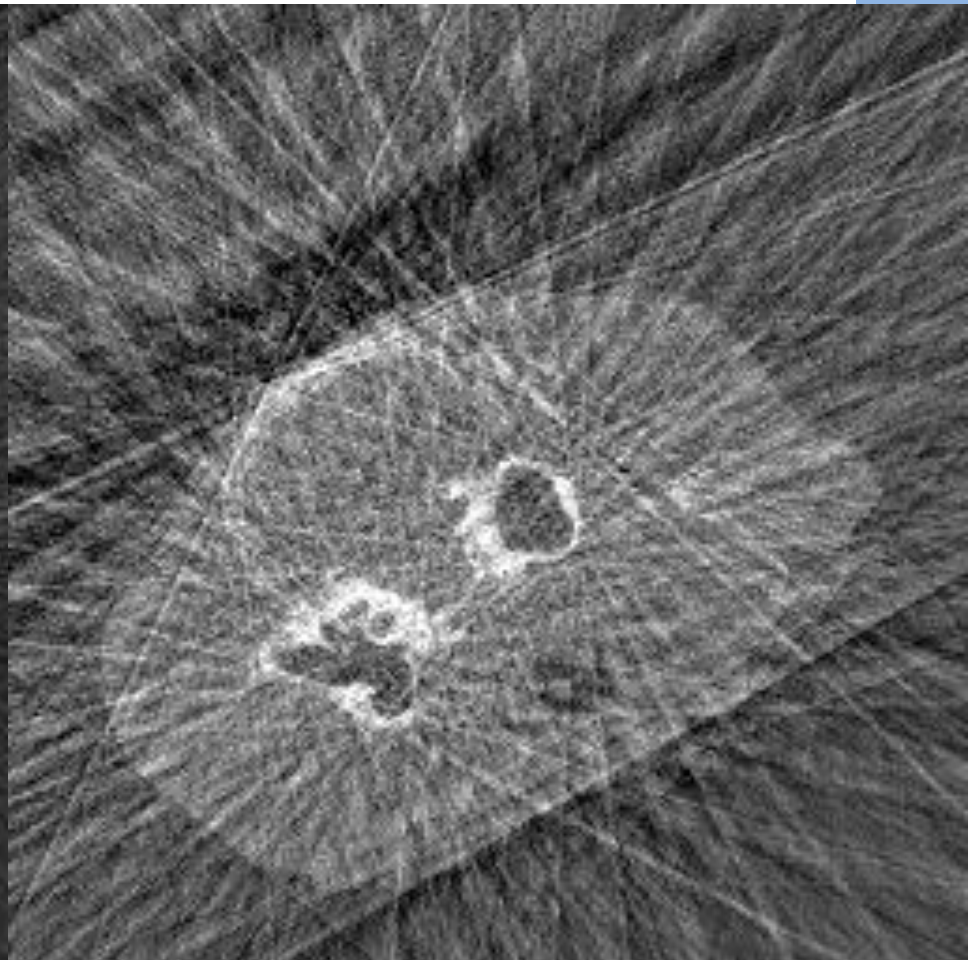






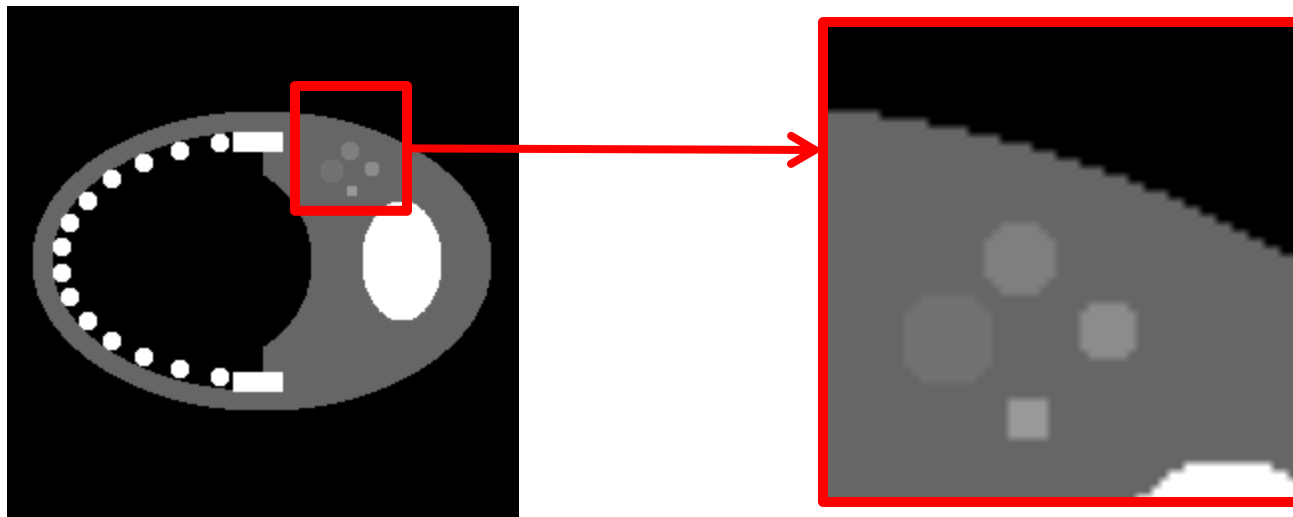


Full reconstruction  
(106 projection images)



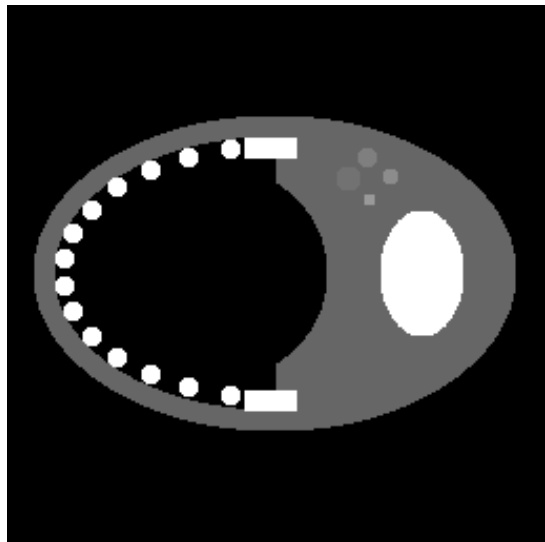
Limited-View reconstruction  
(21 projection images)

- Comparing standard techniques
  - The jaw phantom.

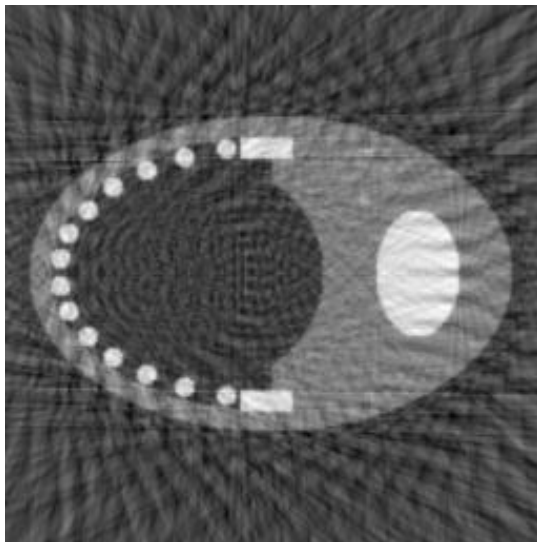


- Reconstruction from 45 equally spaced projections by using standard techniques.

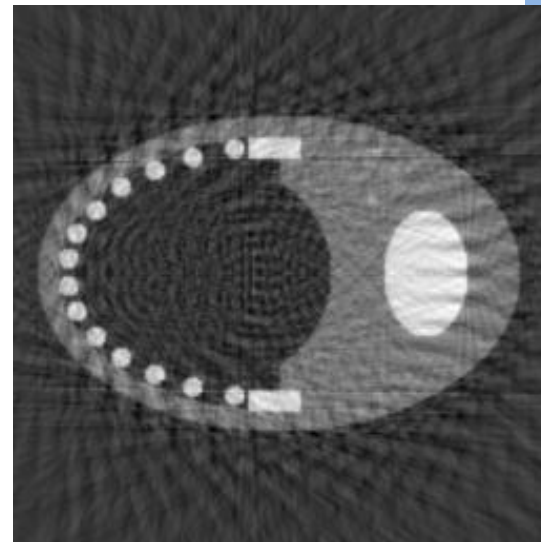
Original



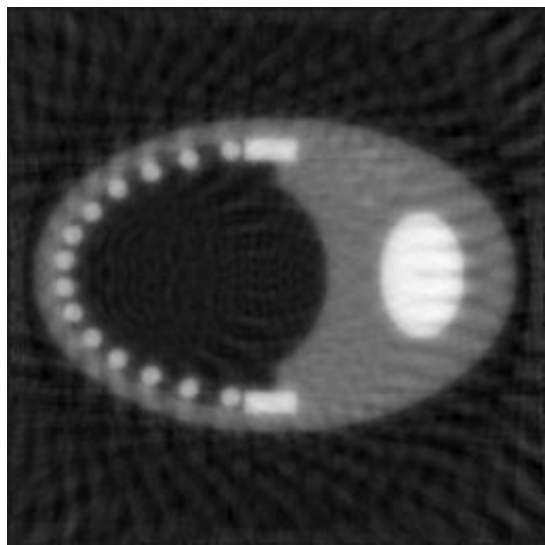
FBP



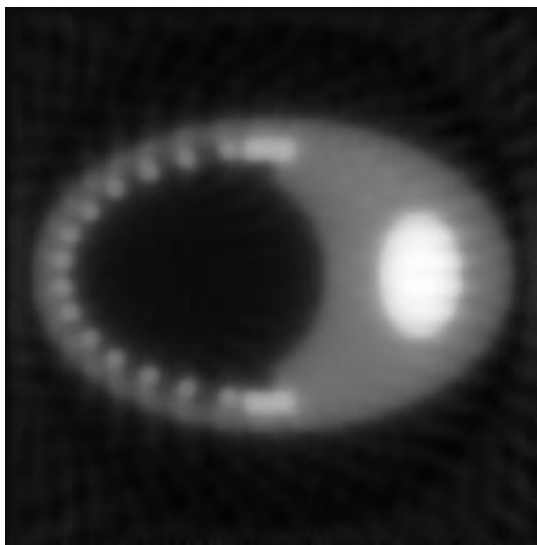
HYPR



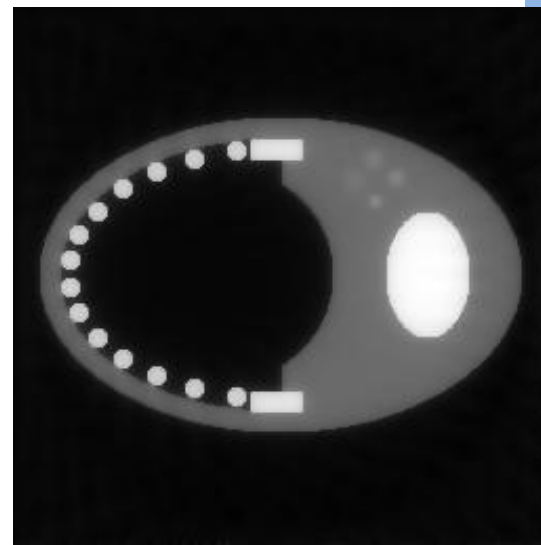
ART



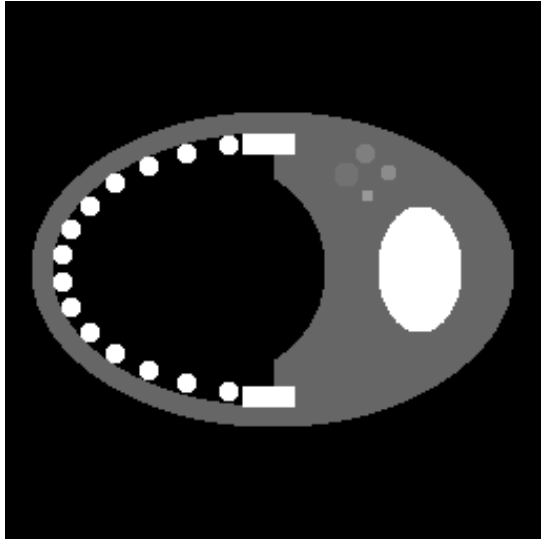
TV



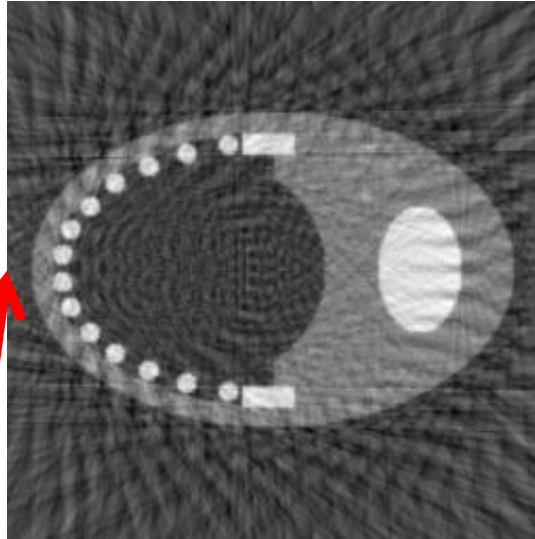
TV with priors



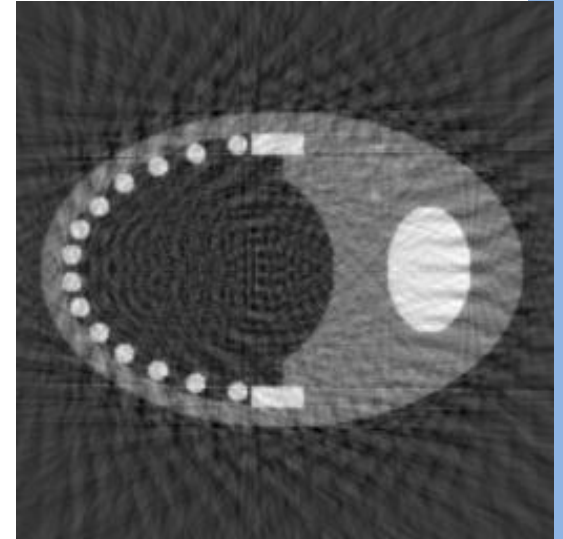
Original



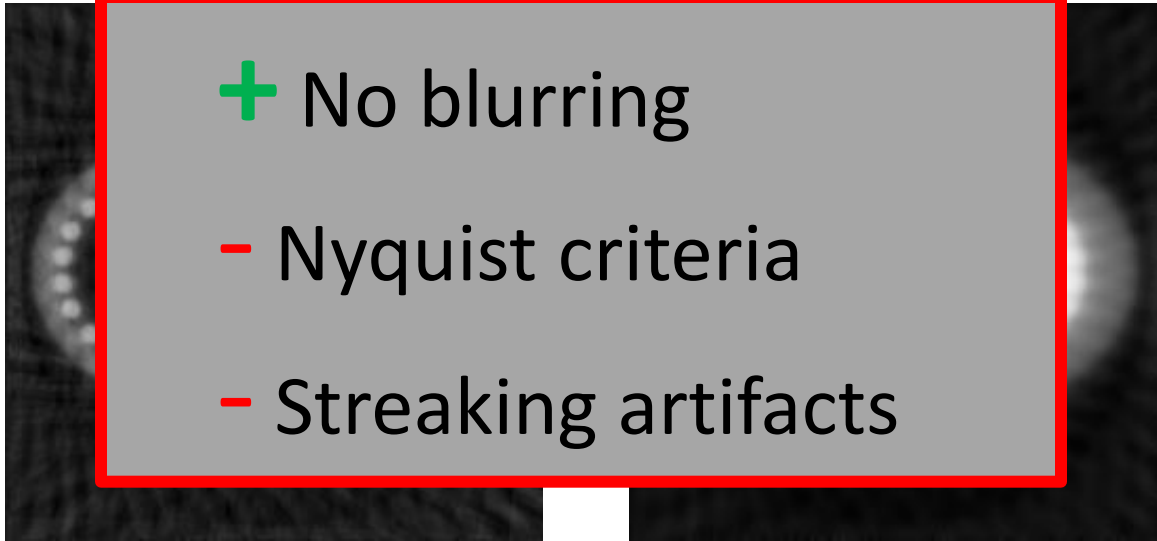
FBP



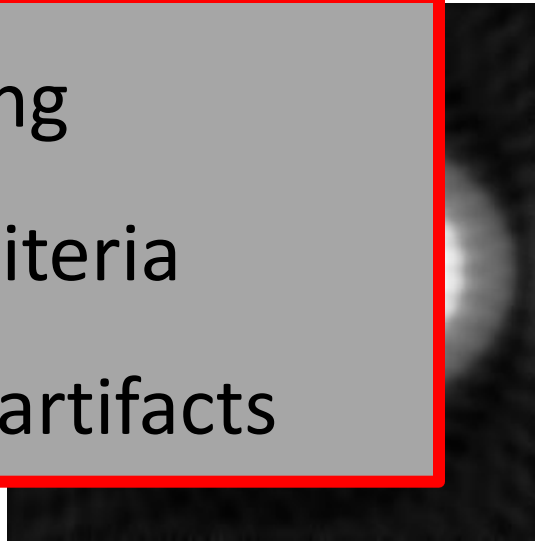
HYPR



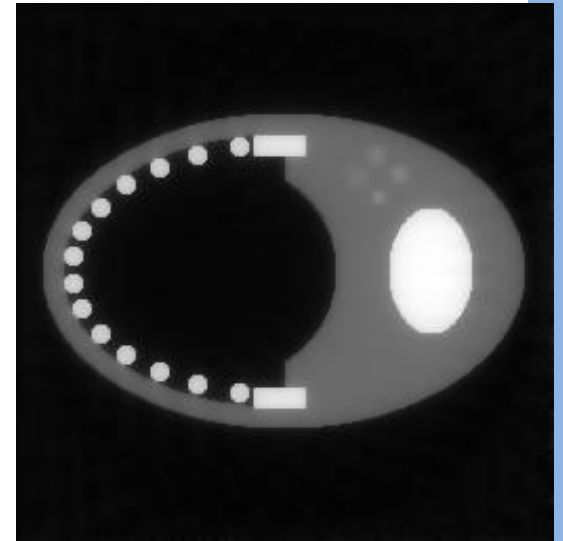
ART



TV

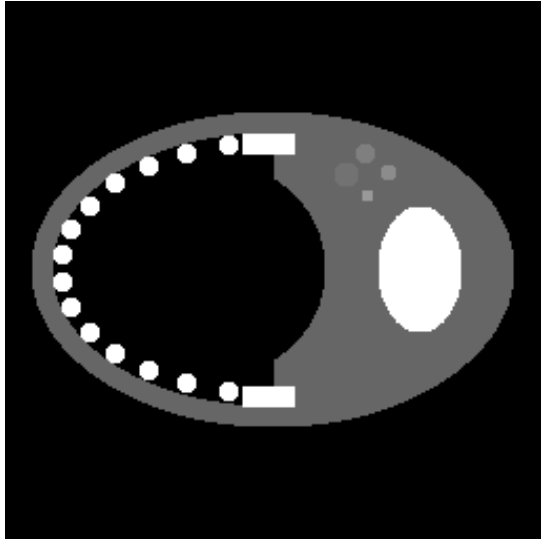


TV with priors

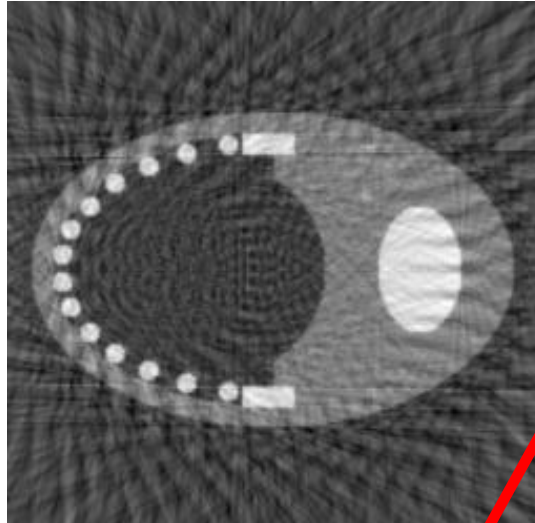


- + No blurring
- Nyquist criteria
- Streaking artifacts

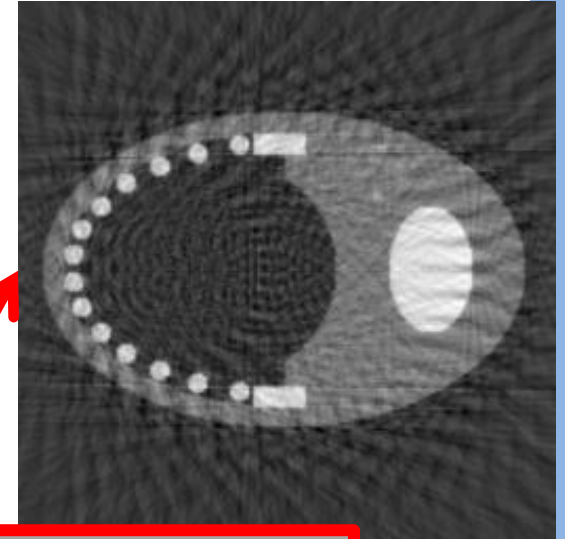
Original



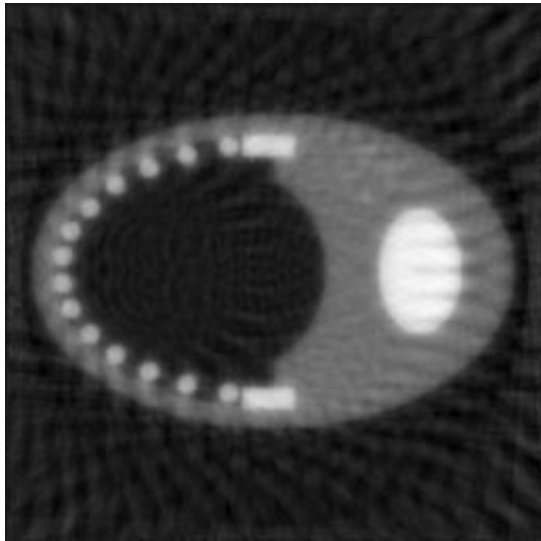
FBP



**HYPR**



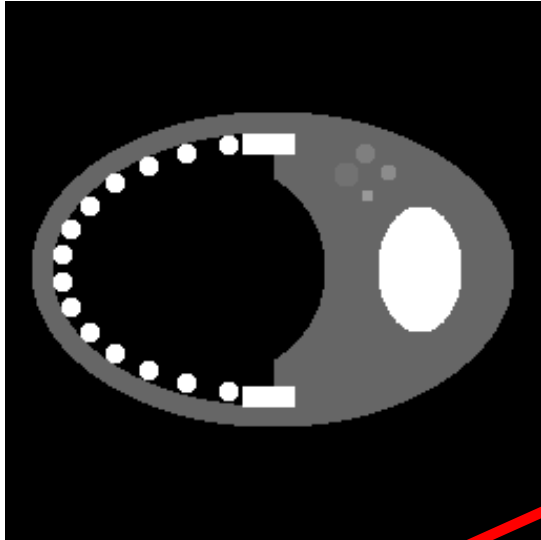
ART



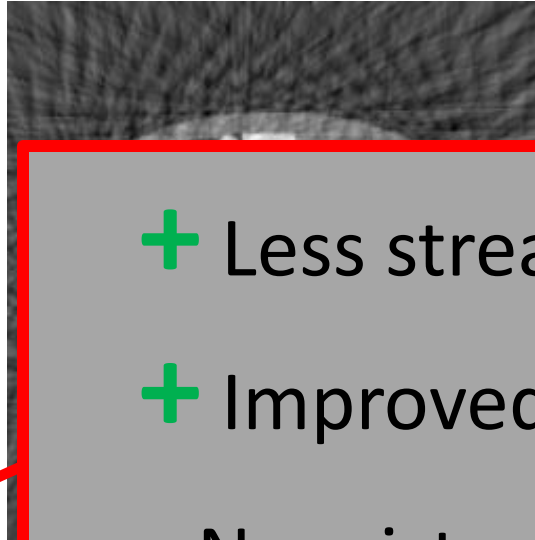
- + No blurring
- + Improved SNR
- Nyquist criteria
- Streaking artifacts



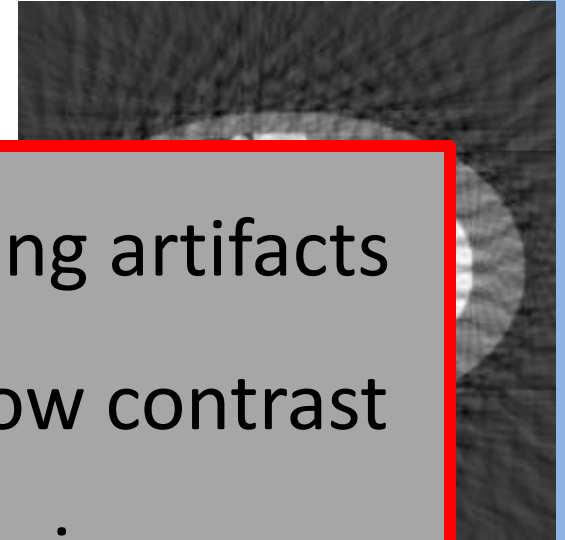
Original



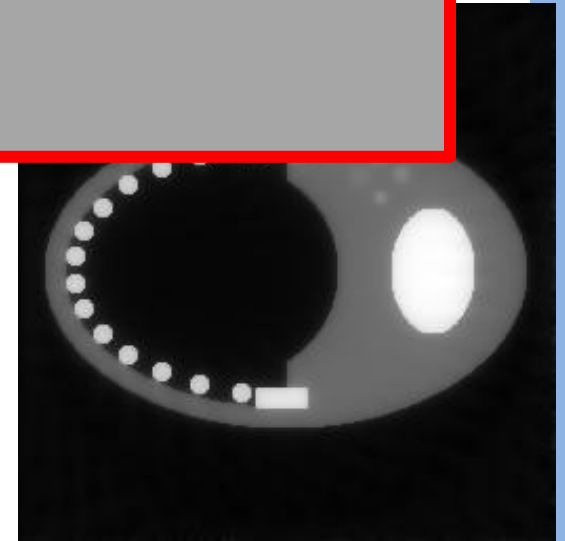
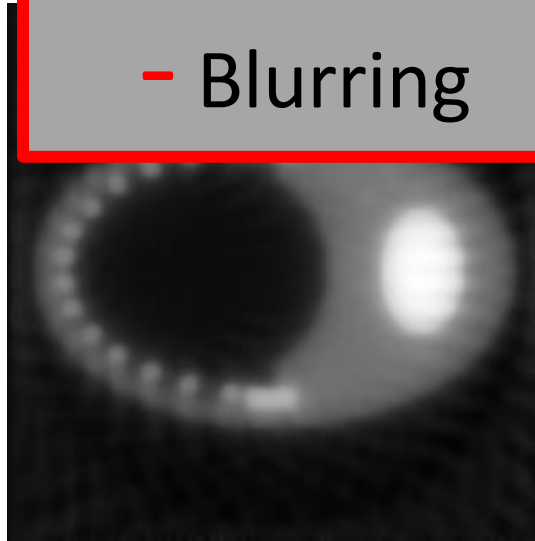
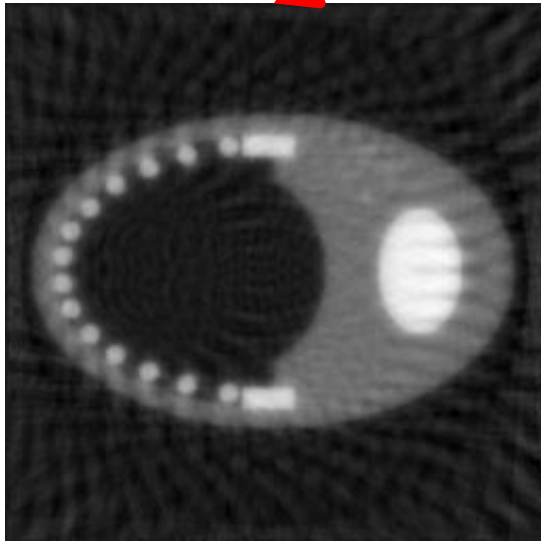
FBP



HYPR

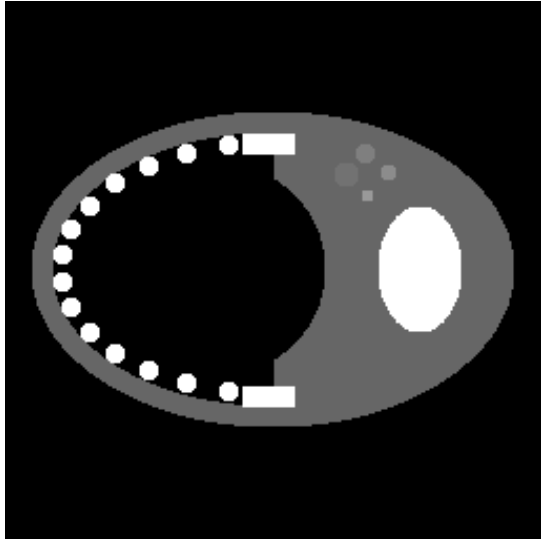


ART

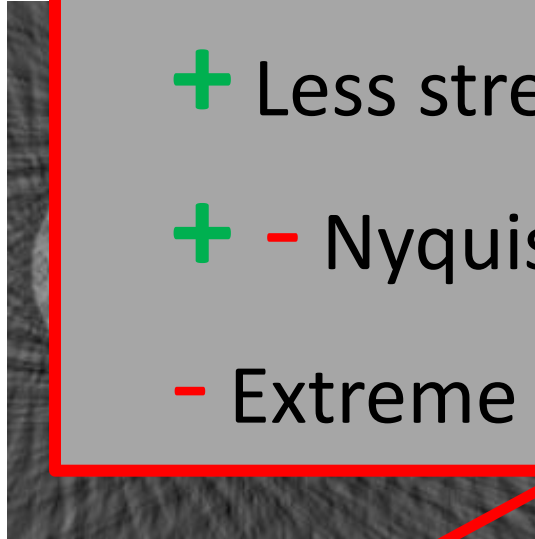


- + Less streaking artifacts
- + Improved low contrast
- Nyquist criteria
- Blurring

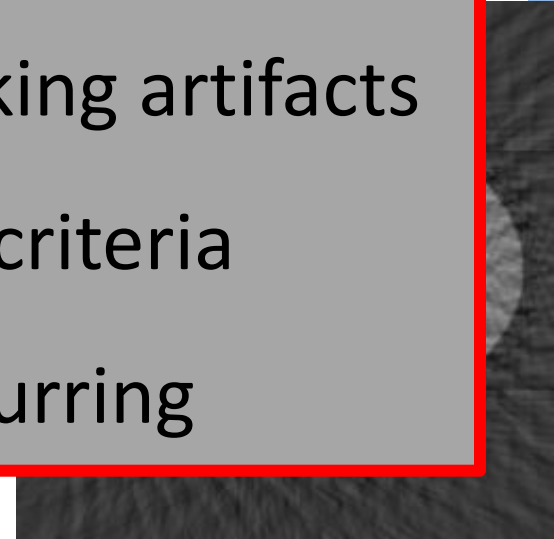
Original



FBP

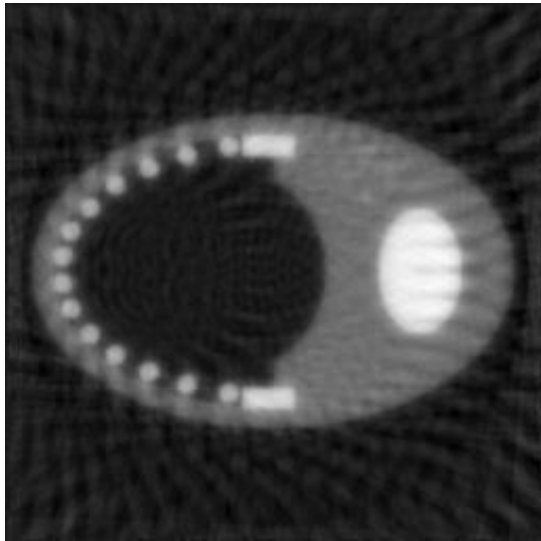


HYPR

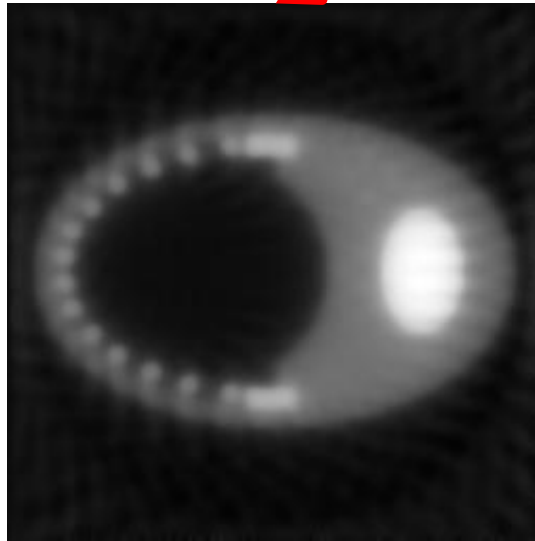


+ Less streaking artifacts  
+ - Nyquist criteria  
- Extreme blurring

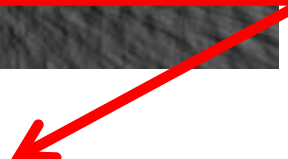
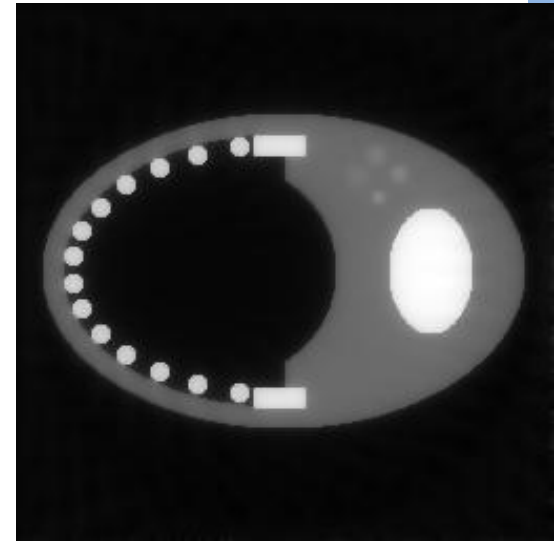
ART



TV

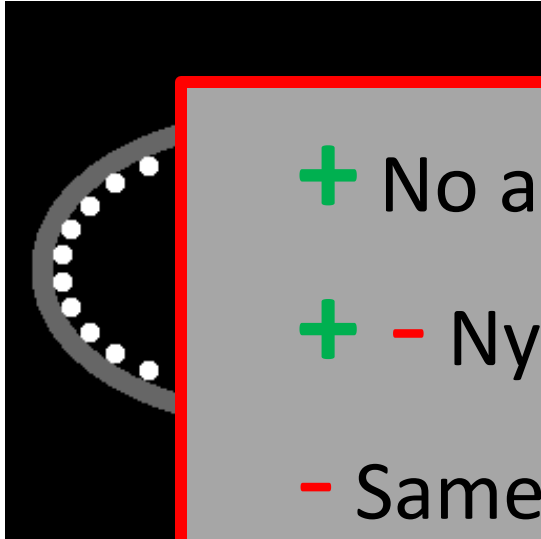


TV with priors





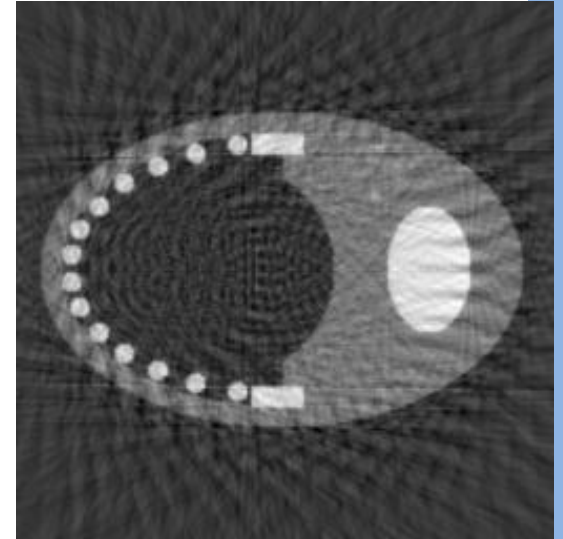
Original



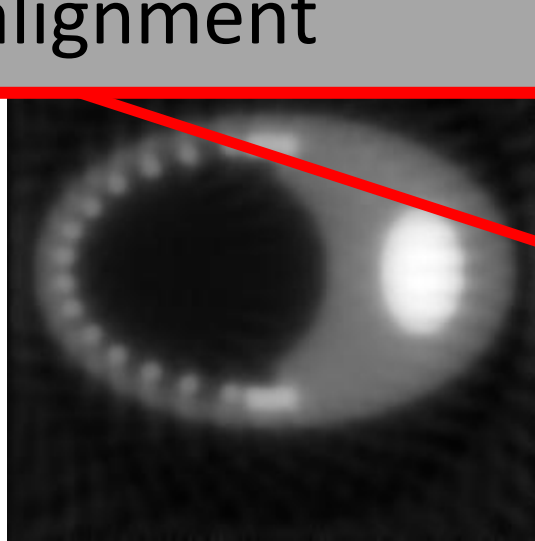
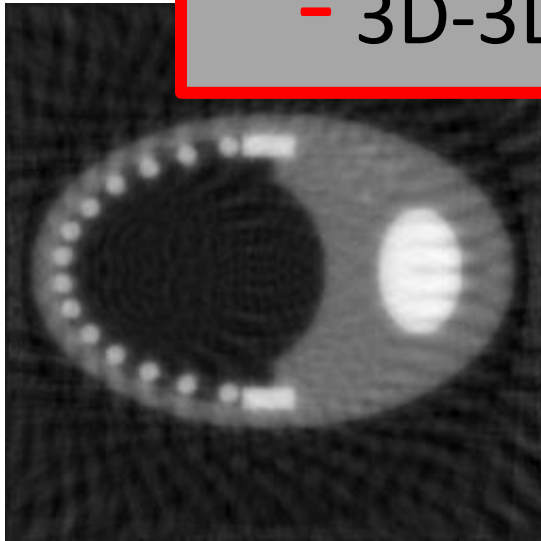
FBP



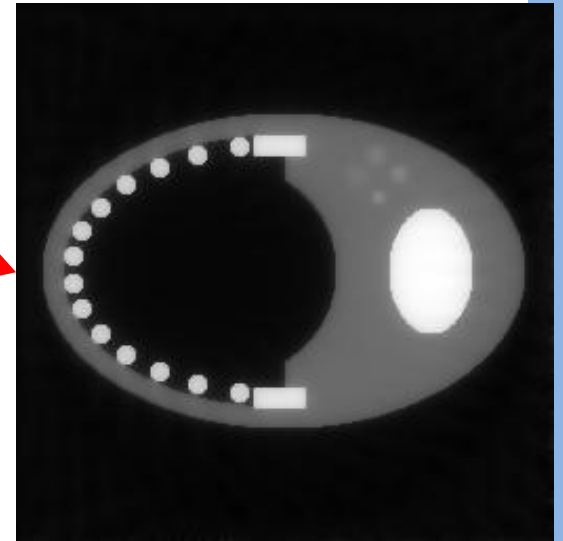
HYPR



ART

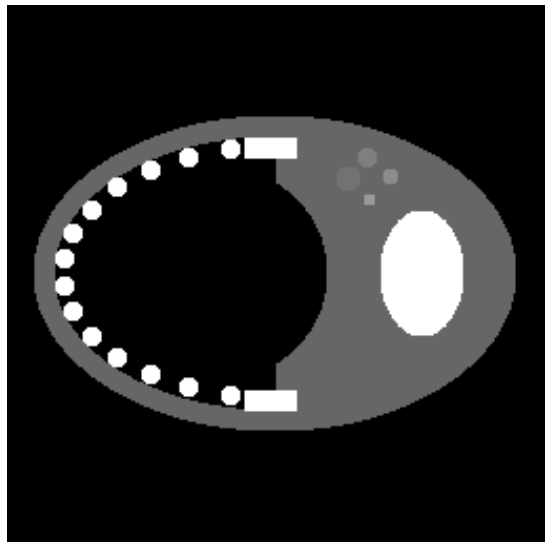


TV with priors

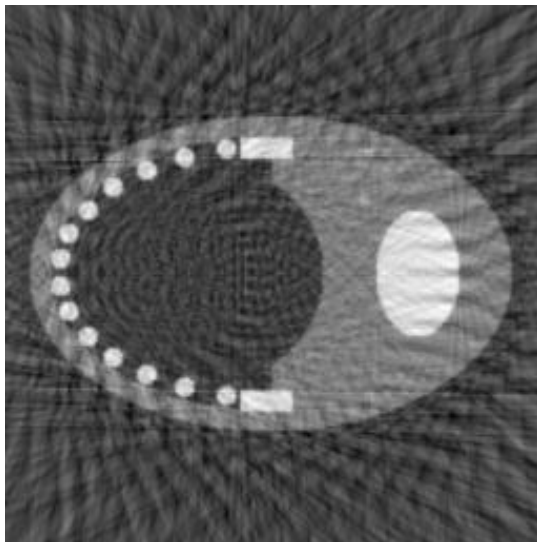


- + No artifacts
- + - Nyquist criteria
- Same patient prior
- 3D-3D alignment

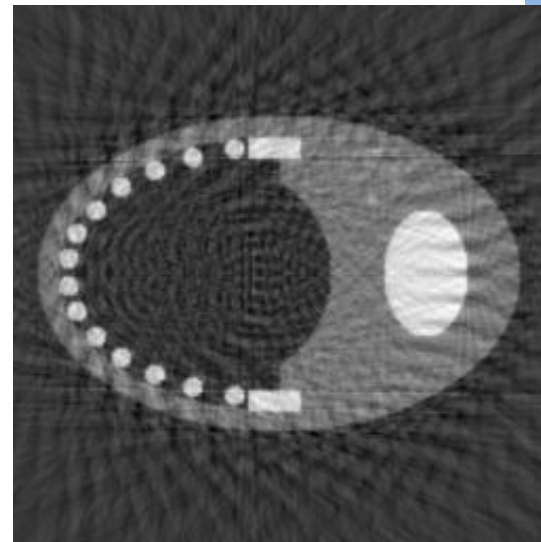
Original



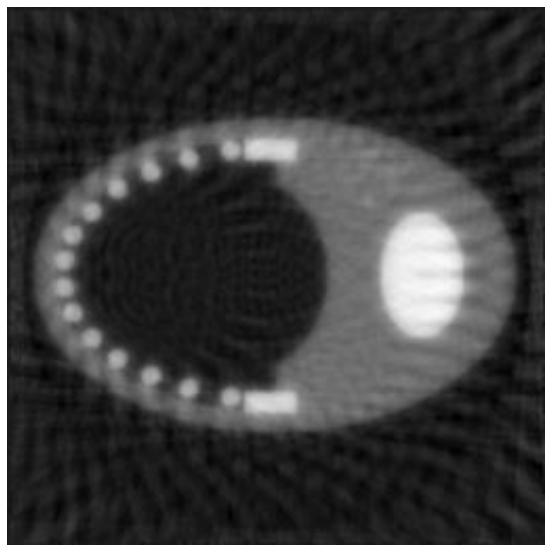
FBP



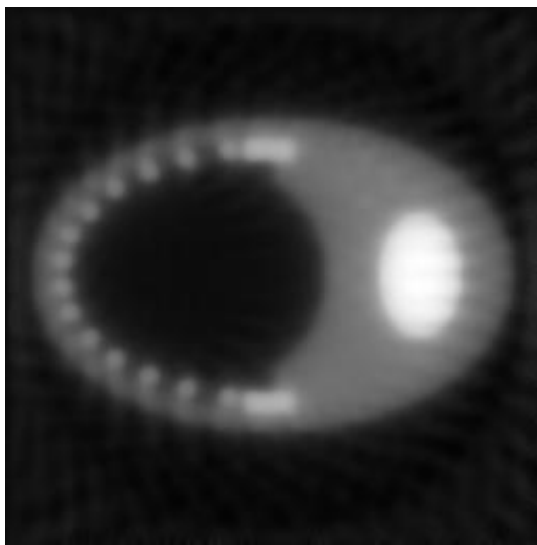
HYPR



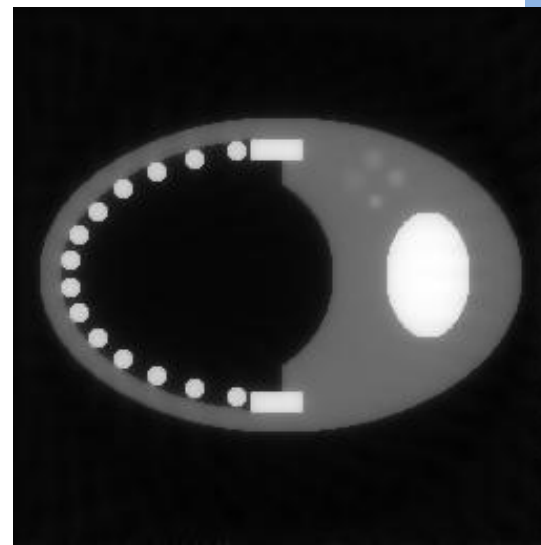
ART



TV



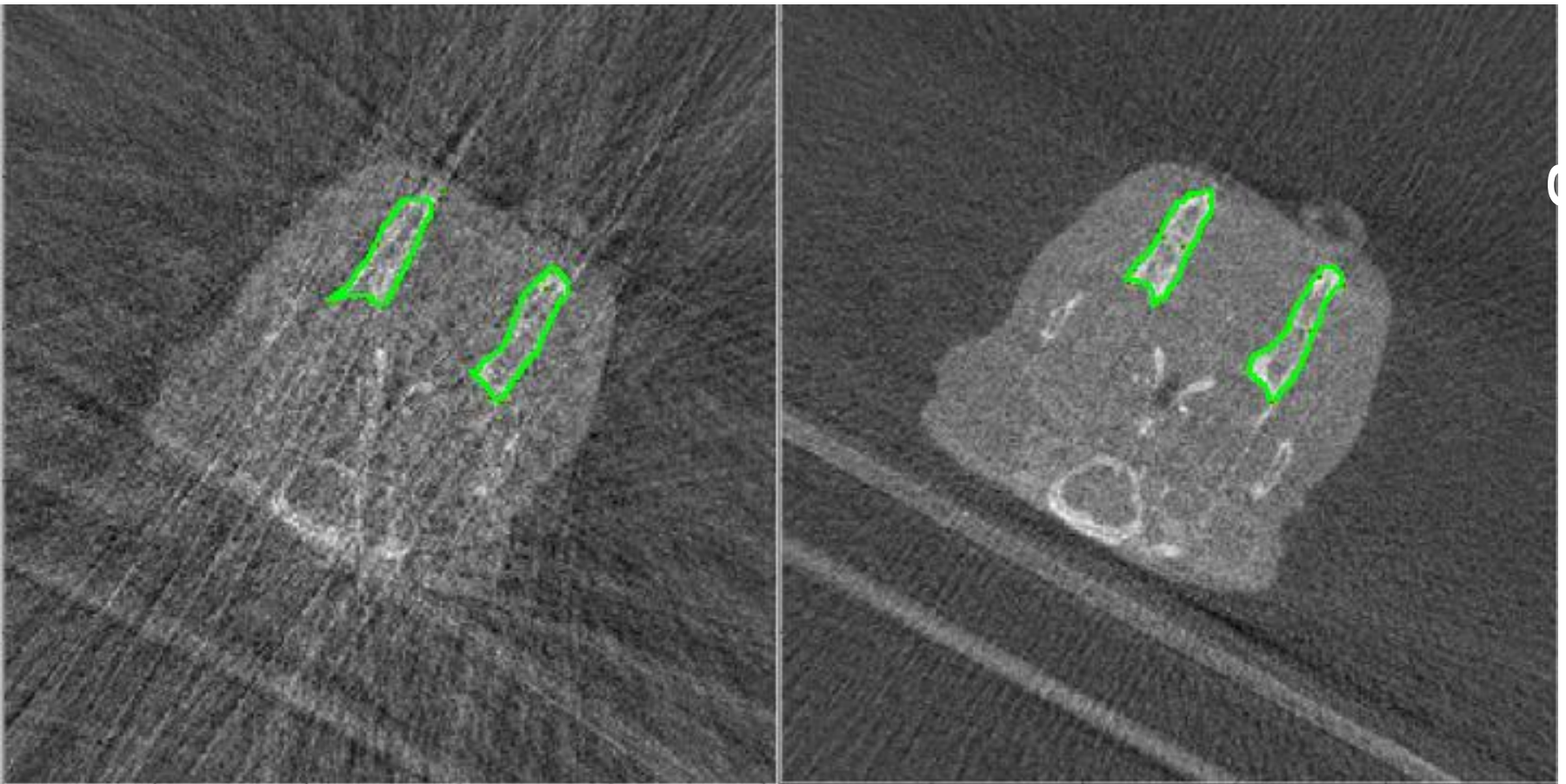
TV with priors



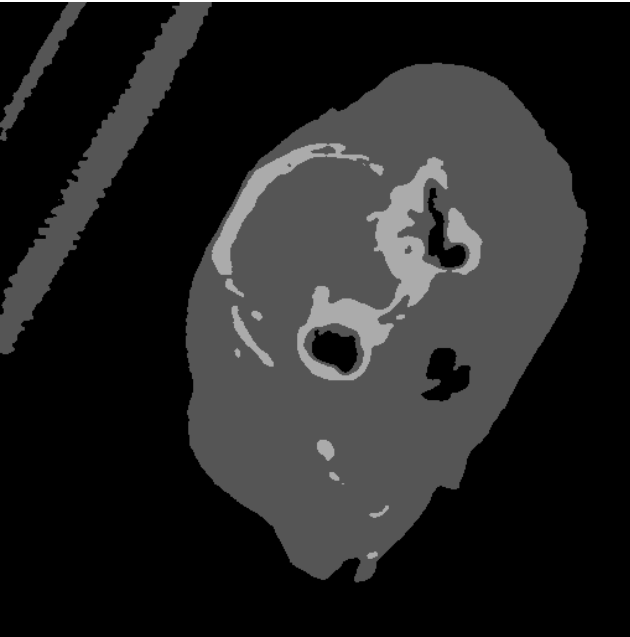
# Geometric Object Aware Tomography

- We want to reconstruction reliable 3D data without prior patient information by:
  - Segmenting anatomic structures from a limited number of projection images
  - Determining from (i) the 3D – 2D relation between segmentation and projection images
  - Reconstruction the different Objects-of-Interest from the outside to the inside
  - Reducing the computation time

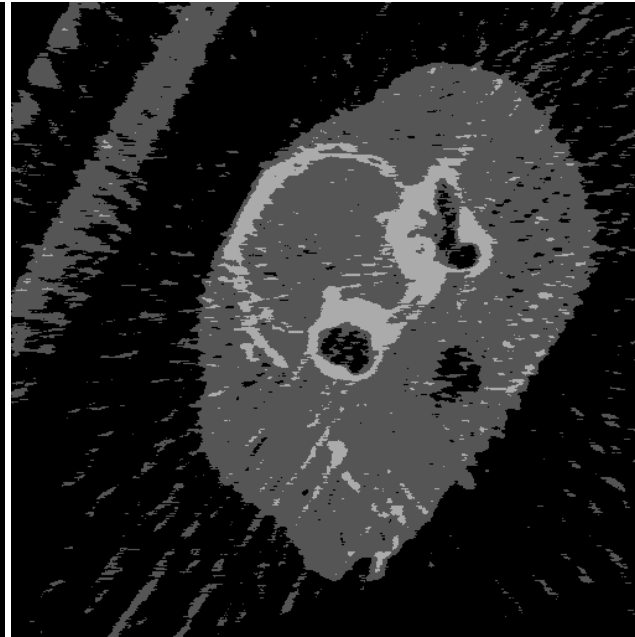
# Supervised learning algorithm using statistical shape models



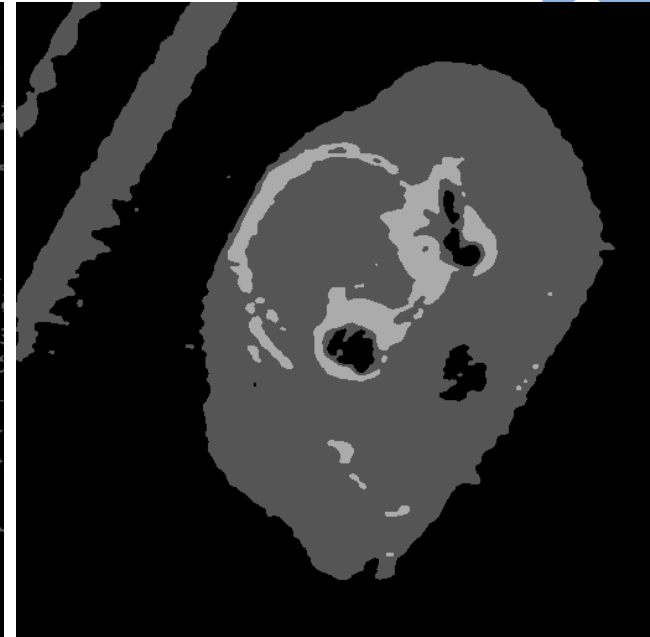
# Unsupervised learning algorithm using clustering techniques



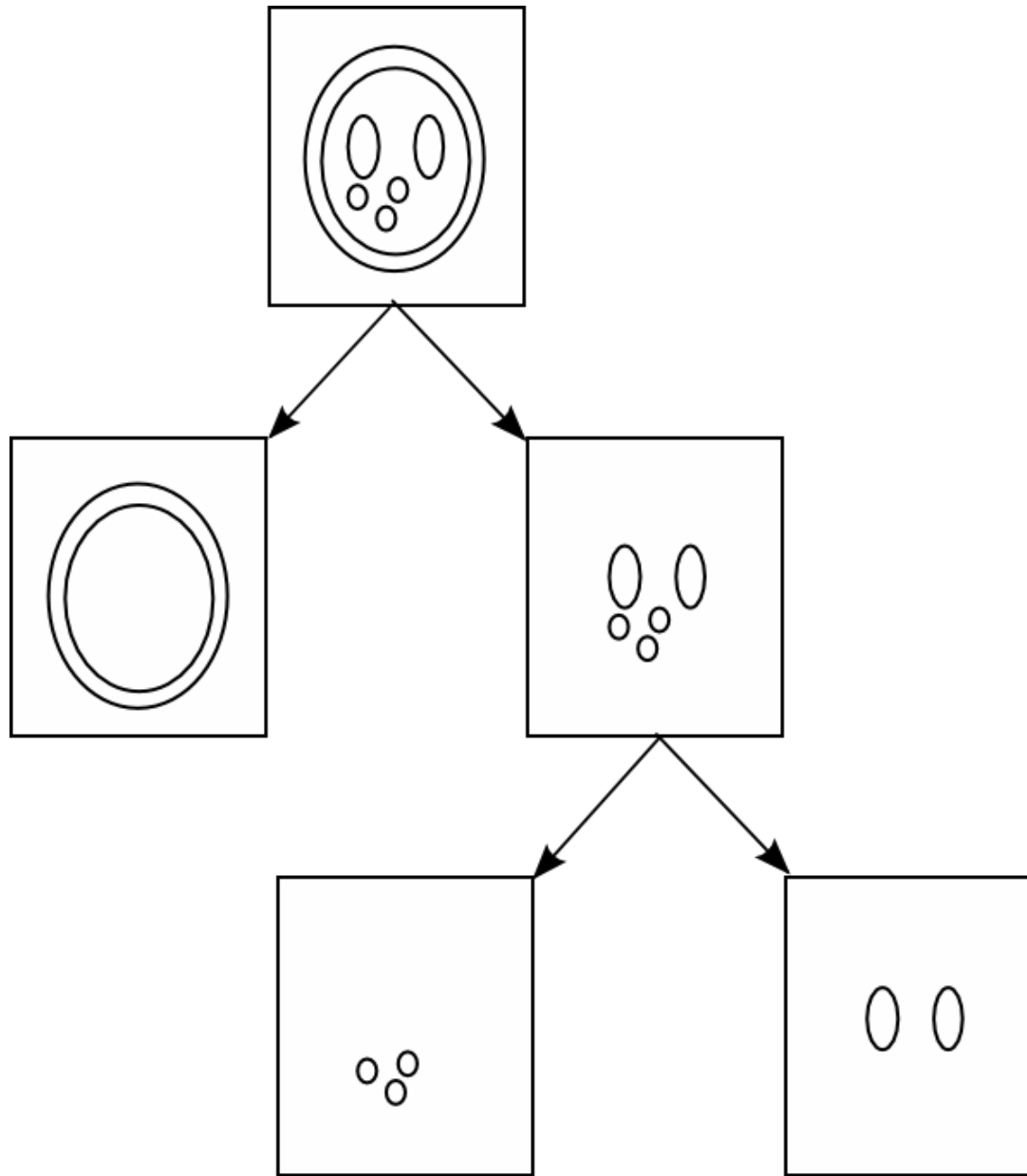
106 projection images

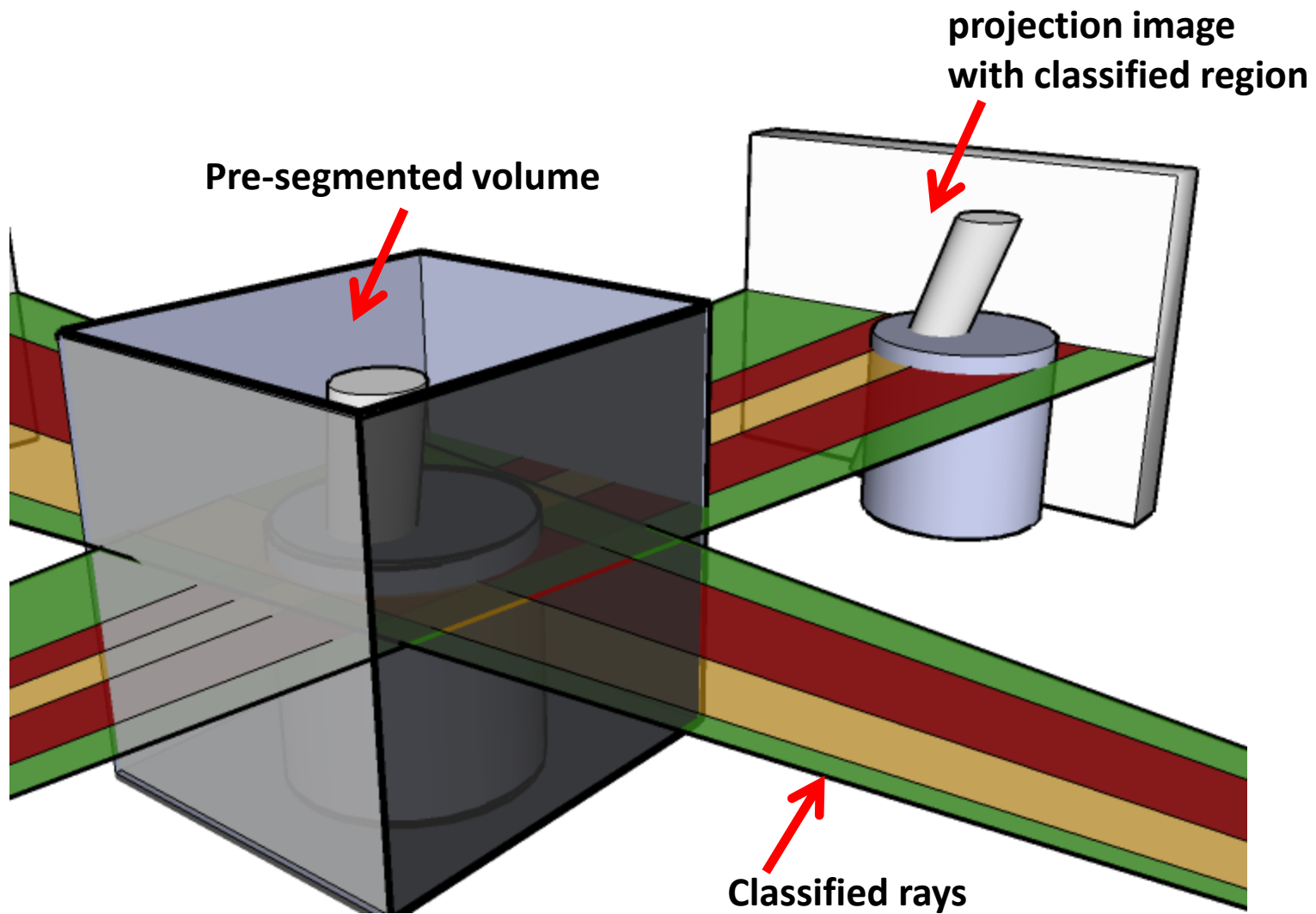


35 projection images



35 projection images





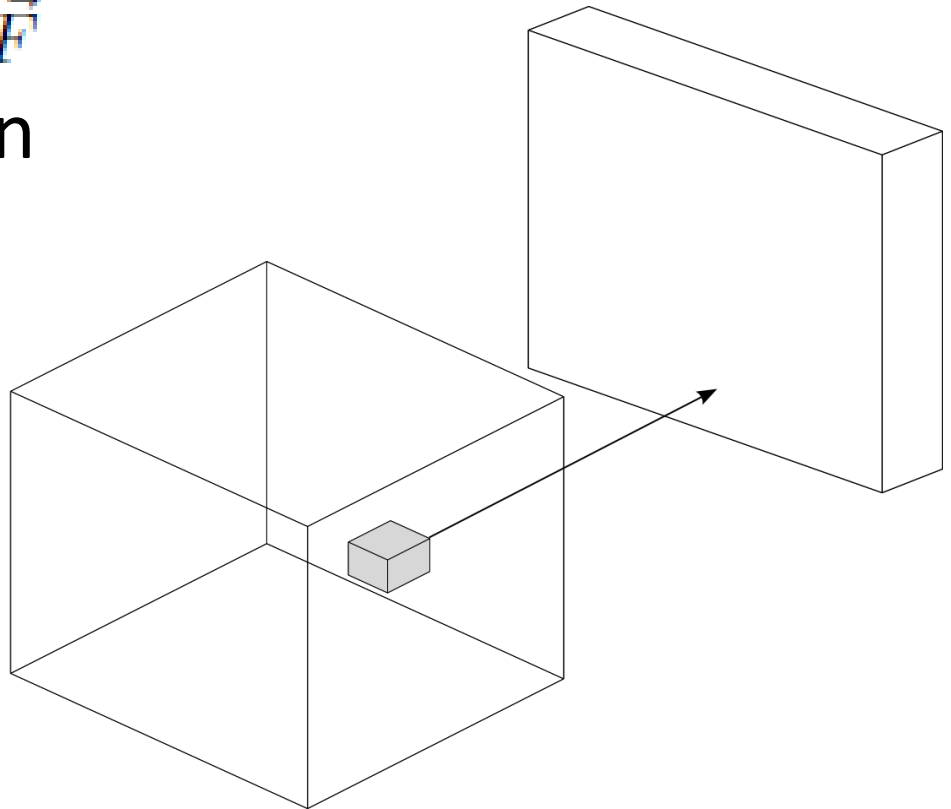
# Reconstruction techniques

- In a linear system, the projections are acquired as:

$$\vec{P} = M\vec{F}$$

Most reconstruction algorithms try to solve:

$$\vec{F} = M^{-1}\vec{P}$$





- We handle each voxel depending on its region, which differs from earlier approaches where all voxels are handled similarly
- From a compressed sensing's point of view, our approach decomposes the complicated 3D volume (matrix) into sparse sub-matrices (or signal) and recovers the whole matrix (signal) iteratively.

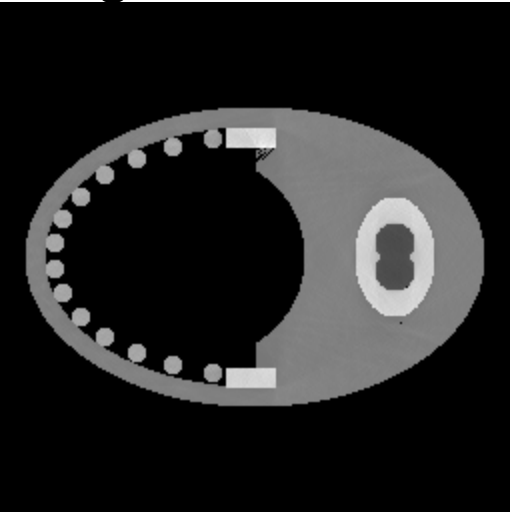
$$P_{OoI_1}^{\vec{}} = M_{OoI_1} F_{OoI_1}^{\vec{}}$$

$$P_{OoI_2}^{\vec{}} = M_{OoI_2} F_{OoI_2}^{\vec{}}$$

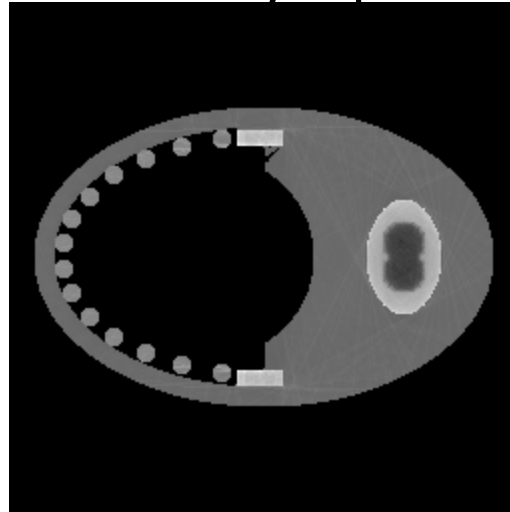
...

$$P_{OoI_n}^{\vec{}} = M_{OoI_n} F_{OoI_n}^{\vec{}}$$

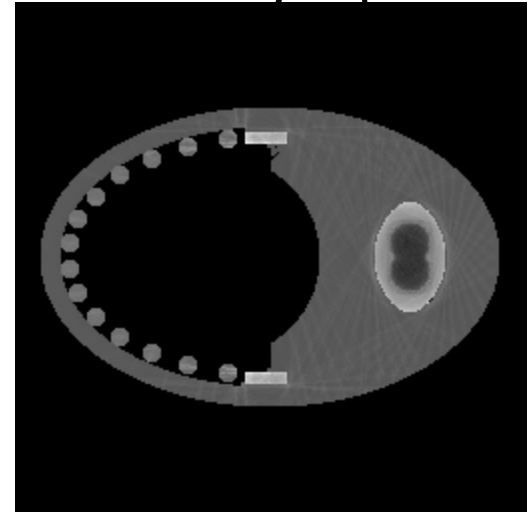
Original



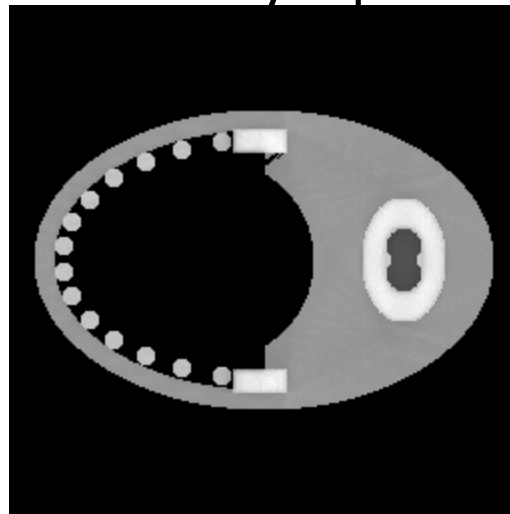
Eroded by 1 pixel



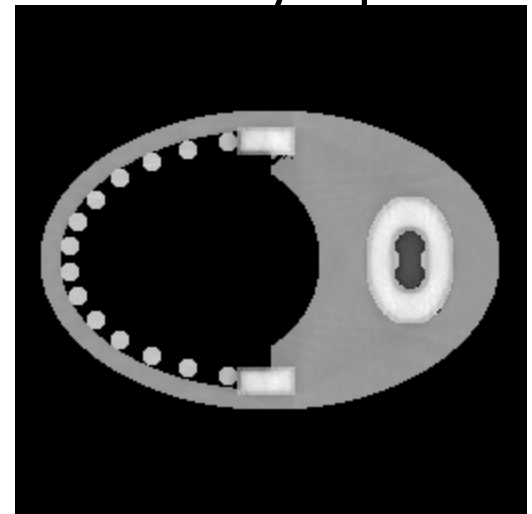
Eroded by 2 pixel



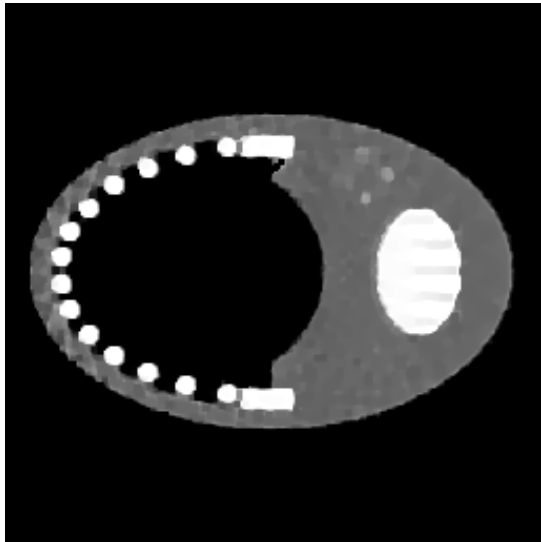
Dilated by 1 pixel



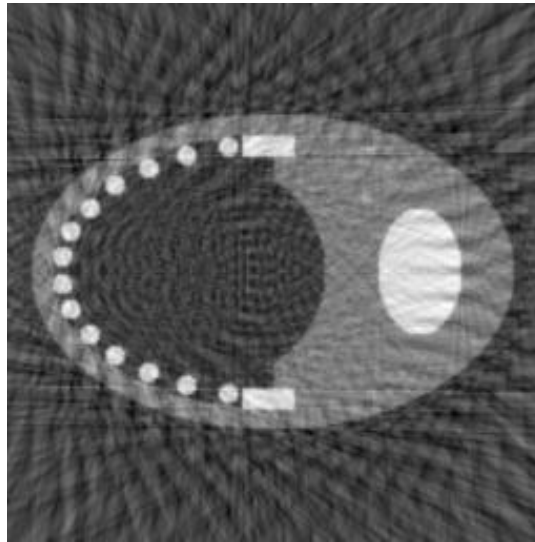
Dilated by 2 pixel



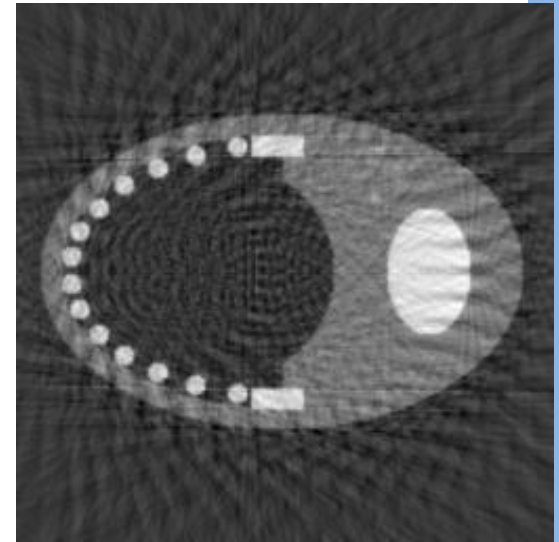
**GOAT**



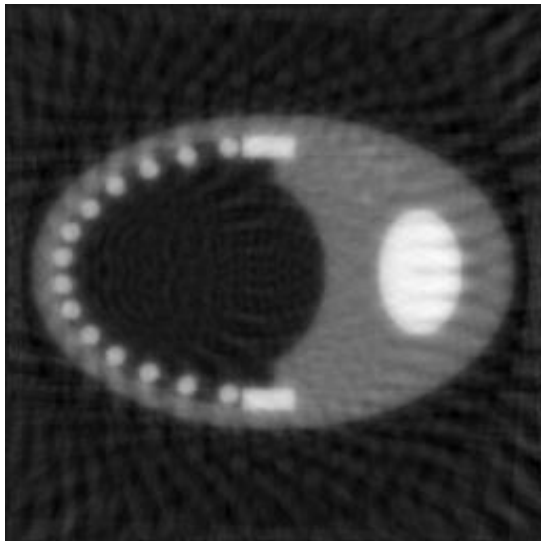
**FBP**



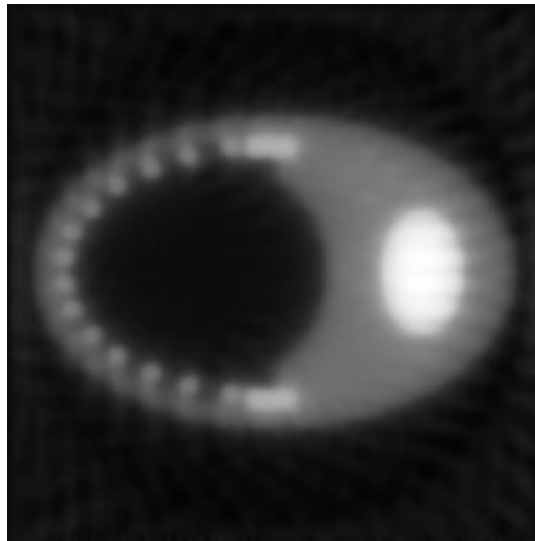
**HYPR**



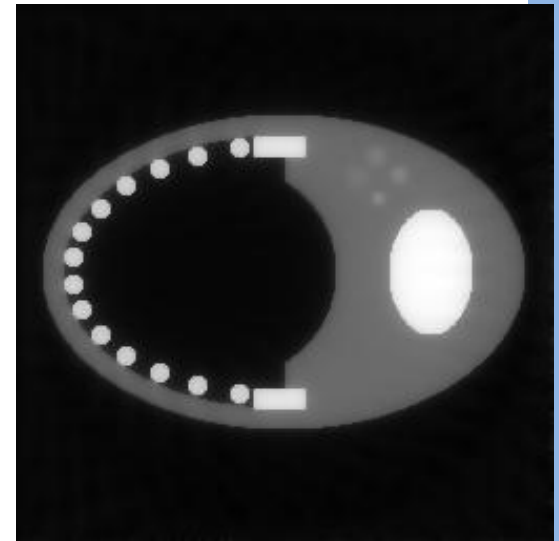
**ART**

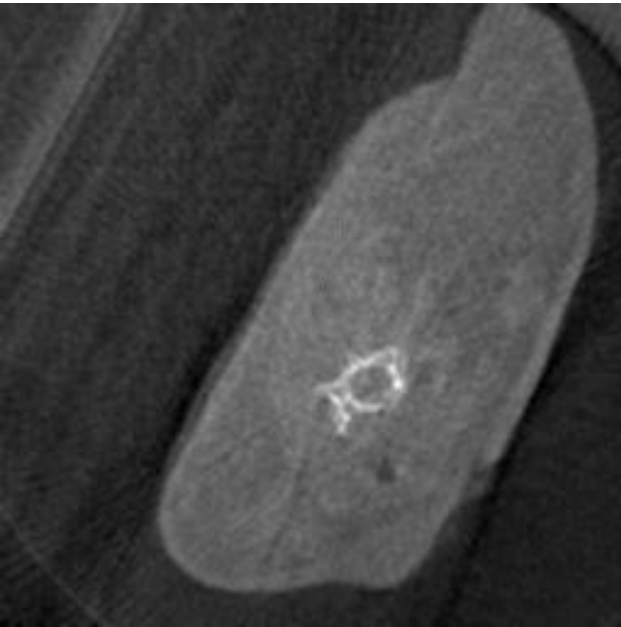


**TV**

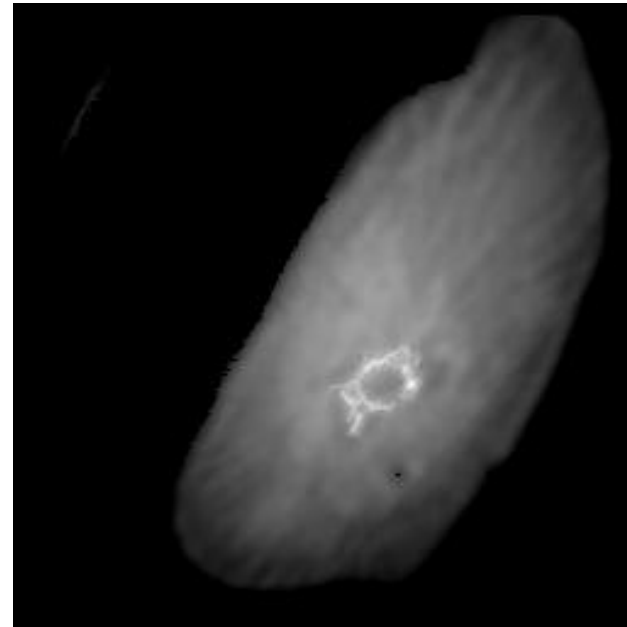


**TV with priors**

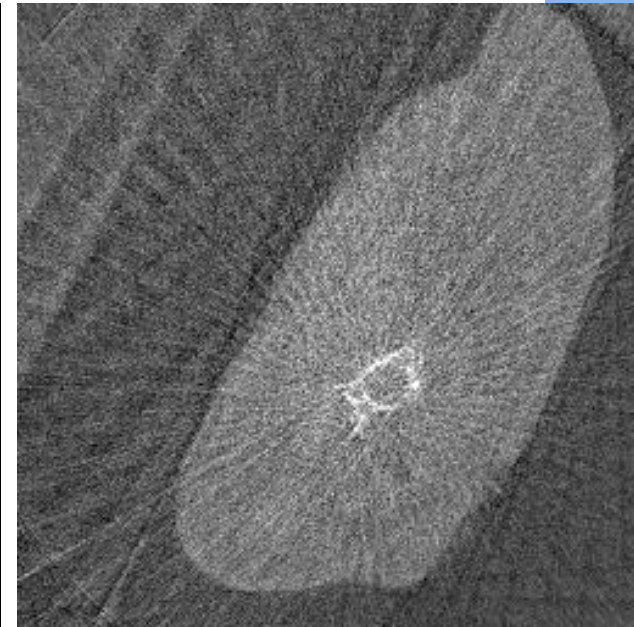




**FBP – 106 projection images**



**GOAT – 35 projection images**



**FBP – 35 projection images**

# Hyperfast Computations in Computed Tomography

**NVIDIA TESLA 2008**  
 960 cores  
 439.1 teraflops/\$1 million \*

**PLAYSTATION 3 CLUSTER 2007**  
 8 PlayStation 3s  
 375 Tflops/\$1 million †

\* Based on the cost of building your own Tesla supercomputer out of four Tesla C1060 units. Complete instructions are available at nvidia.com.

† Seven processors per node at 150 gigaflops per unit for a single processor (11 processors per node for a dual processor); \$400-plus per node.

‡ 6480 Opteron CPUs and 12 960 Cell processors.

## TERAFLOPS PER US \$1 MILLION

**ILLIAC IV 1976**  
 64 processors  
 0.00000048 Tflops/\$1 million

**CRAY-1 1976**  
 1 processor  
 0.00001778 Tflops/\$1 million

**CRAY Y-MP 1988**  
 8 vector processors  
 0.000115 Tflops/\$1 million

**ASCI RED 1997**  
 4510 processors  
 0.01818182 Tflops/\$1 million

**EARTH SIMULATOR 2002**  
 5120 processors  
 0.0175 Tflops/\$1 million

**BLUE GENE/L 2004**  
 65 536 processors  
 2.8 Tflops/\$1 million

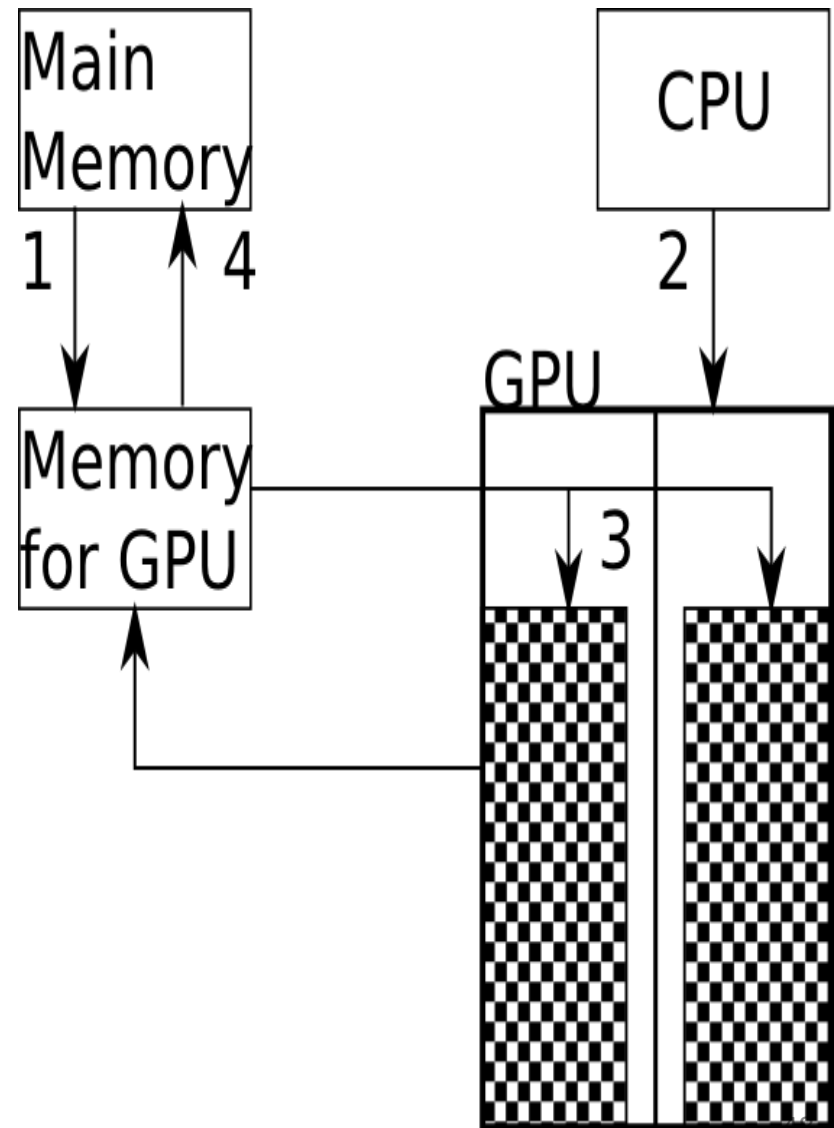
**ROADRUNNER 2008**  
 19 440 processors ‡  
 8.3 Tflops/\$1 million



- The performance profile of a single GPU is:
  - 240 Stream Processors
  - 1296 MHz Shader Clock
  - 1024Mb Memory
  
- 2 x Nvidia Geforce 280 GTX



- We use the Nvidia/CUDA platform
- The main steps of this platform are:
  1. Data copied from CPU to GPU
  2. Instructions are copied to GPU
  3. Instructions are executed on GPU
  4. Data copied back to CPU

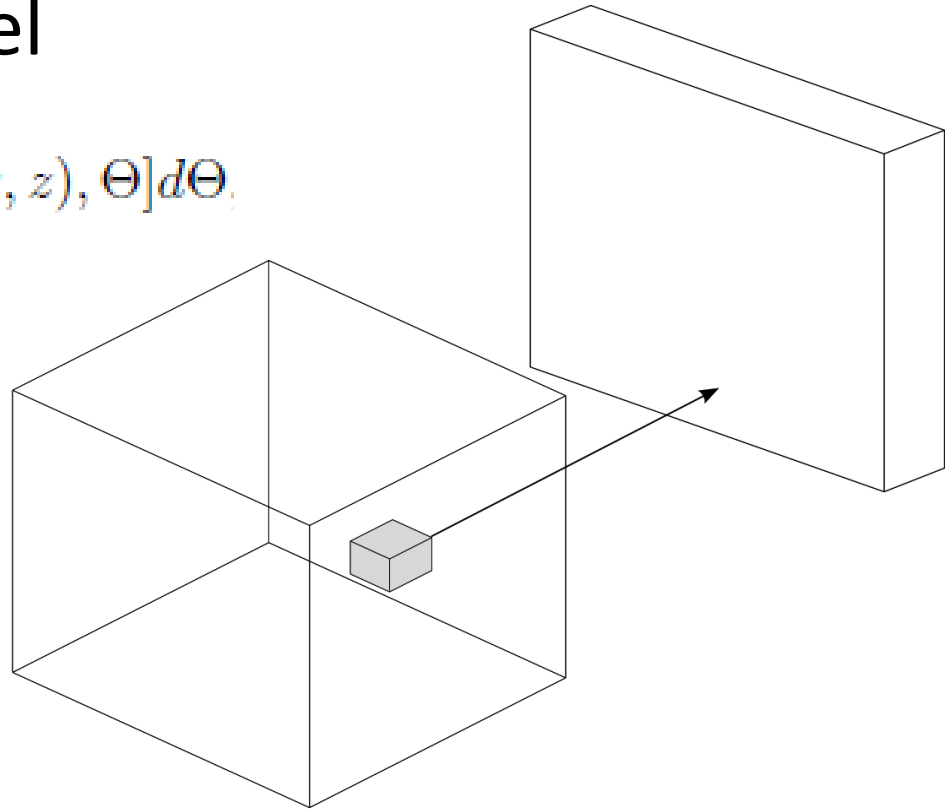




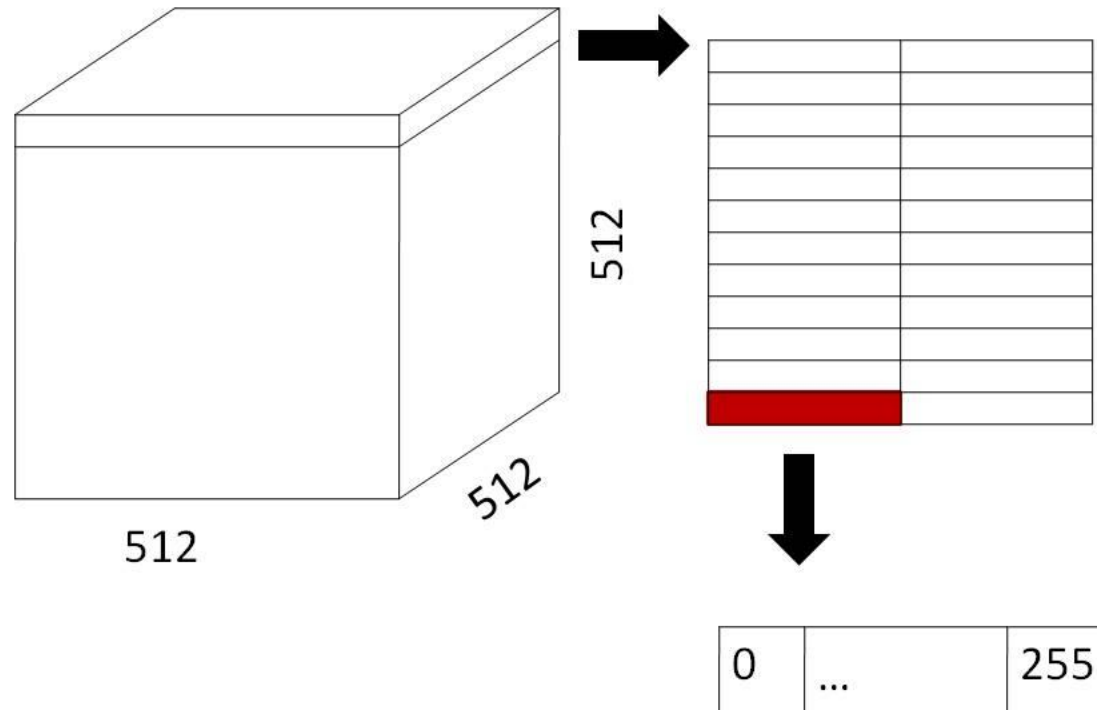
- For our implementation, we used CUDA2.x which is provided by NVIDIA.
  - Advantages
    - uses C-like language
    - user-managed shared memory can be shared among threads
    - faster downloads and readbacks to and from the GPU
  - Disadvantages
    - CUDA does not allow the full use of the shader, so it is slightly slower

- Essential for parallel computations: independent calculations.
- FBP: Each voxel is independent of any neighbor voxel

$$F(\vec{r}) = \int w^2(\Theta, \vec{r}) P_f[u(x, z), v(y, z), \Theta] d\Theta.$$

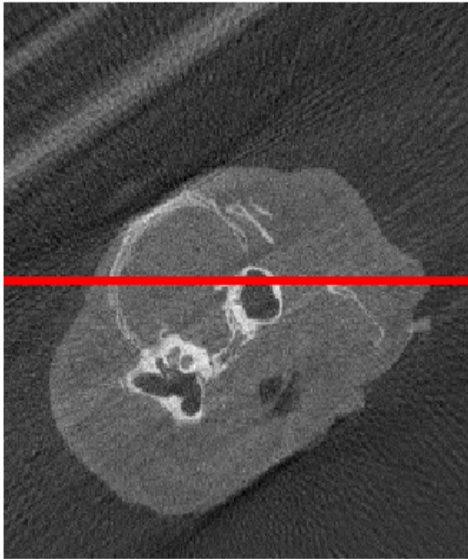


- Parallel approach:
  - Upload all projections into texture memory
  - Upload sub volume into shared memory

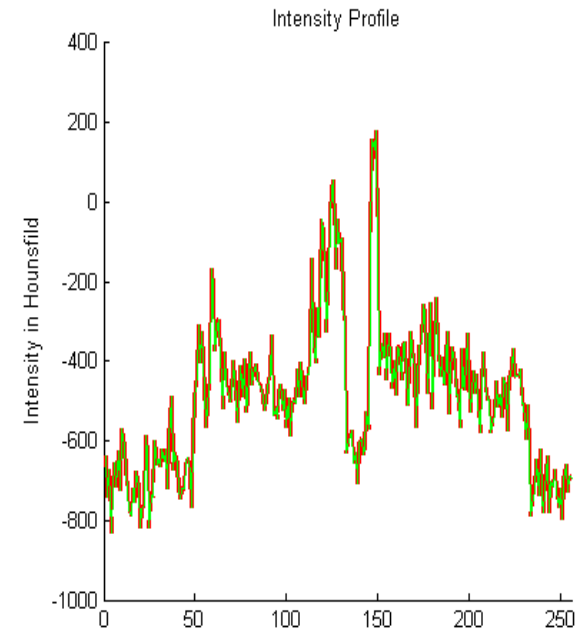
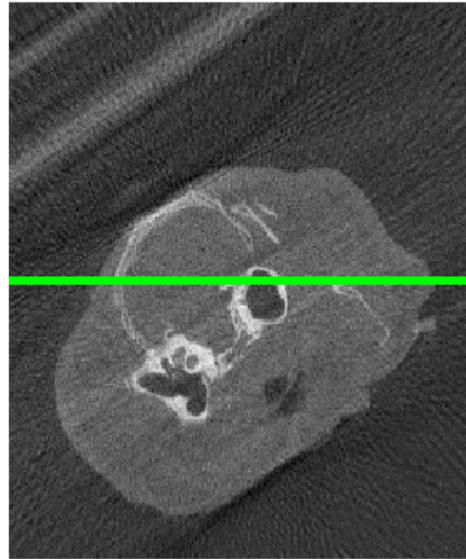


- GPU vs. CPU
  - Floating point calculations / math.h function / NAN are NOT different between GPUs and CPUs

CPU

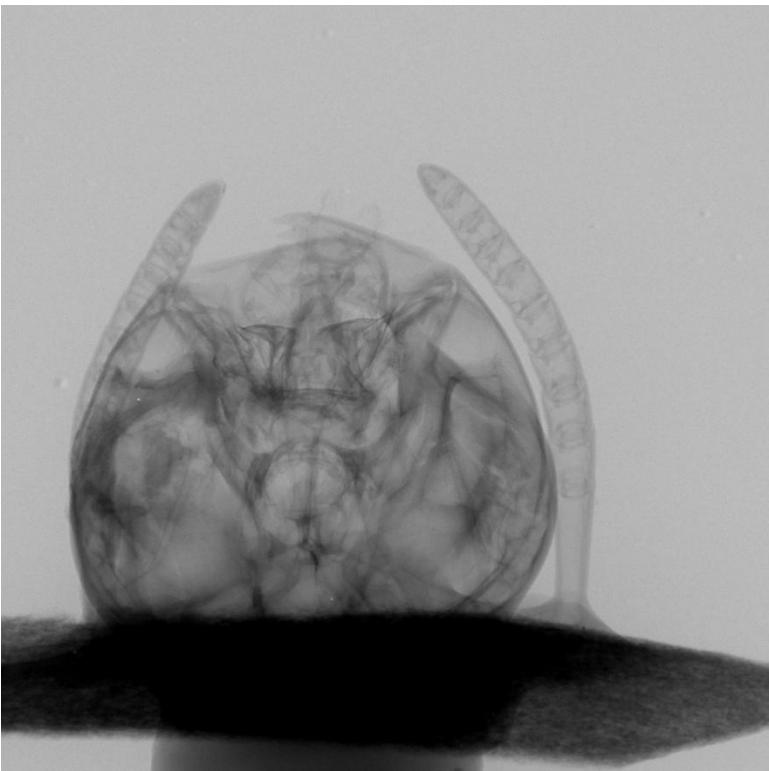
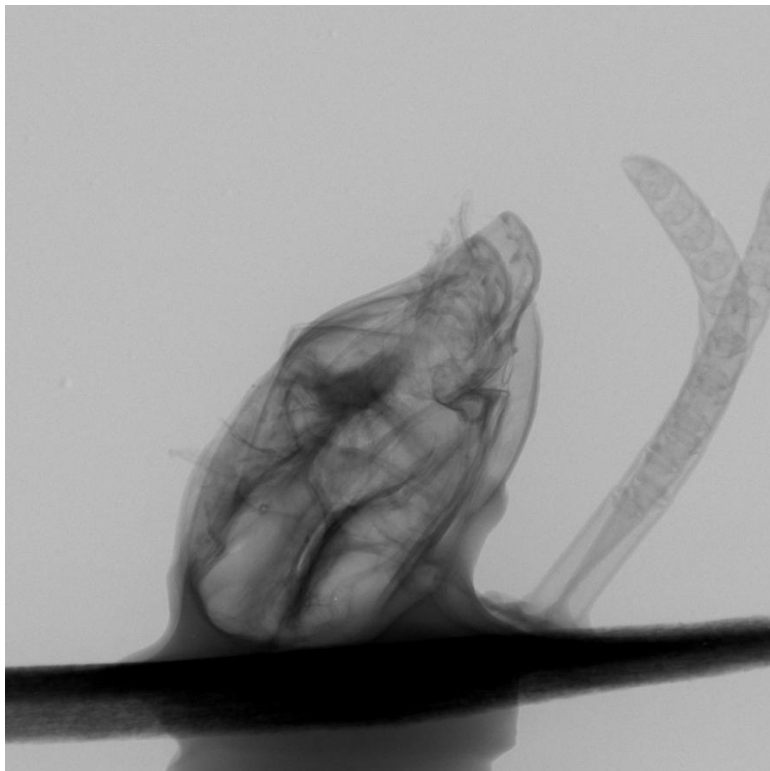


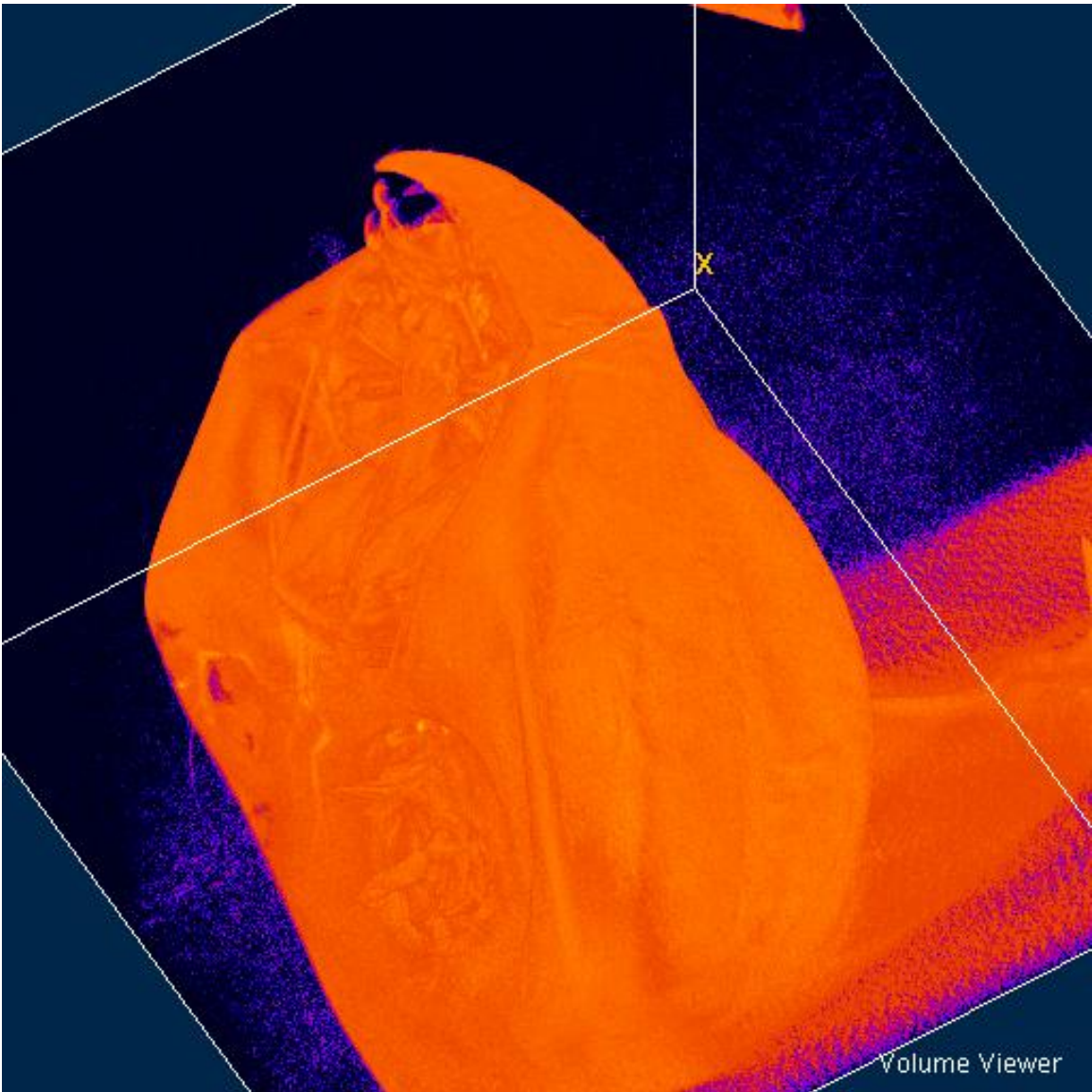
GPU



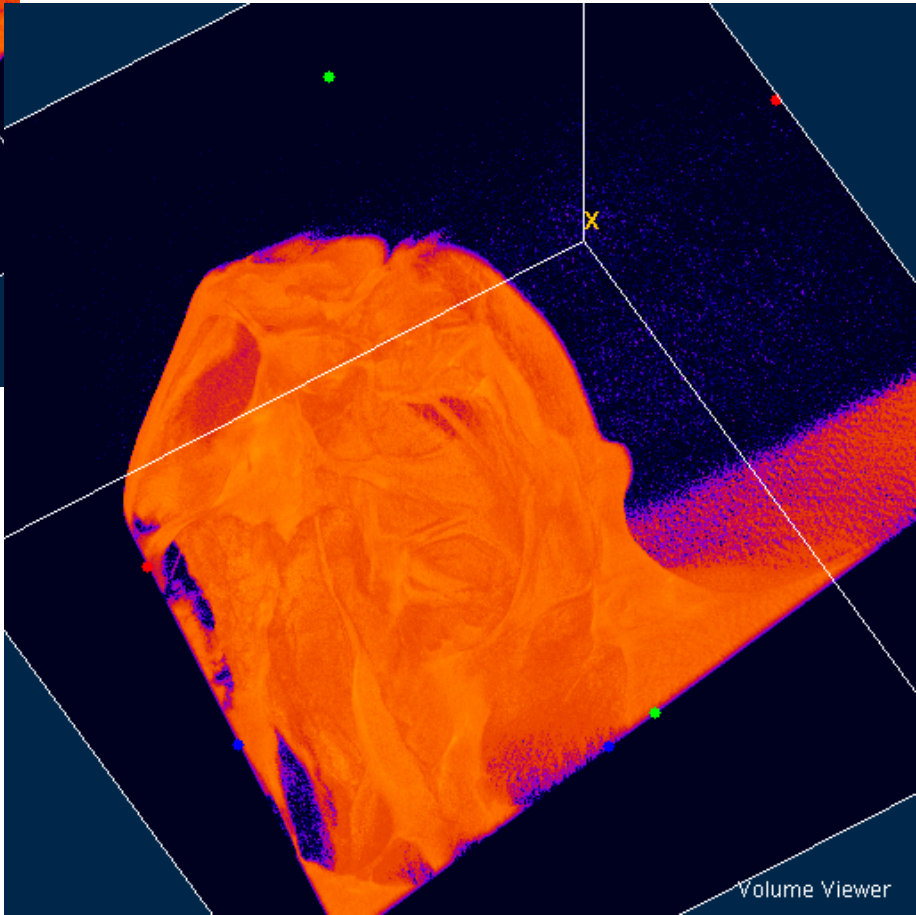
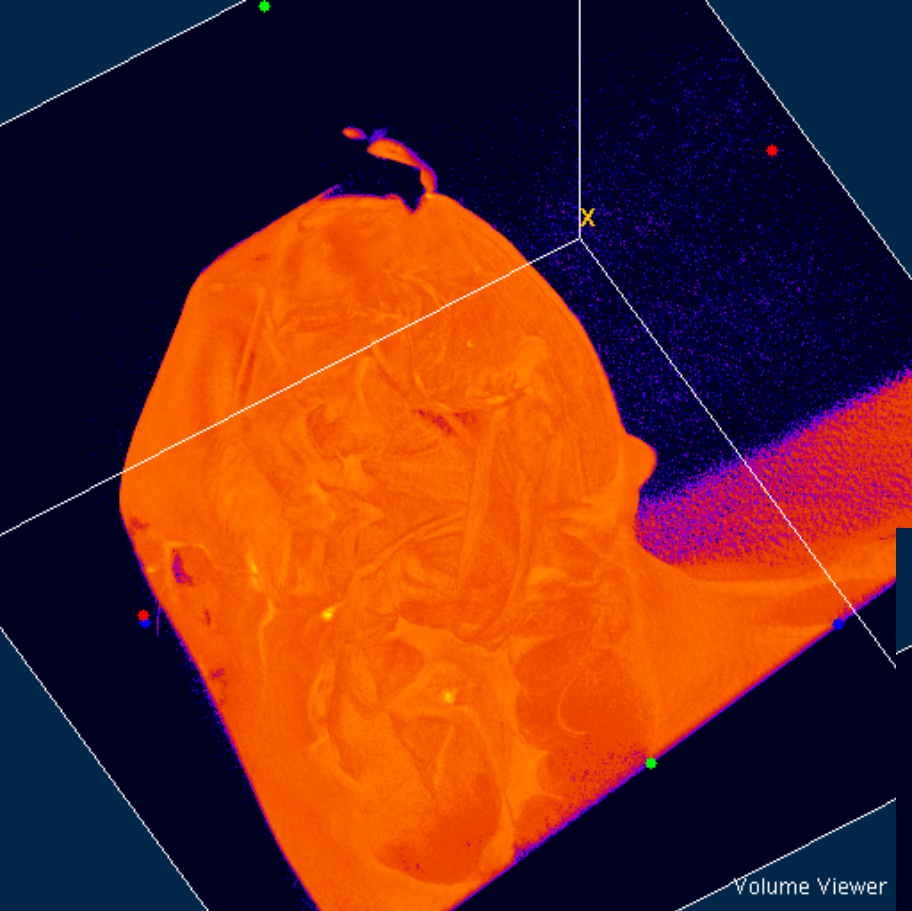
- Results:

Volume Size	Total [s]	Convolution [s]	[fps]	Backprojection [s]
256 <sup>3</sup>	3.3	2.5	42.4	0.8
512 <sup>3</sup>	8.7	2.5	42.4	6.2
1024 <sup>3</sup>	41.2	2.5	42.4	39.7





# GPU Reconstruction





**At University at Buffalo:**

Kenneth R. Hoffmann, Ph.D.

Jinhui Xu, Ph.D.

Jason J. Corso, Ph.D.

Sebastian Schafer, Ph.D.

Alan Walzcack

**At TU Munich:**

Prof. Dr. Ernst J. Rummeny

Bernhard Renger

Questions???

Contact me at: [noel@roe.med.tum.de](mailto:noel@roe.med.tum.de)