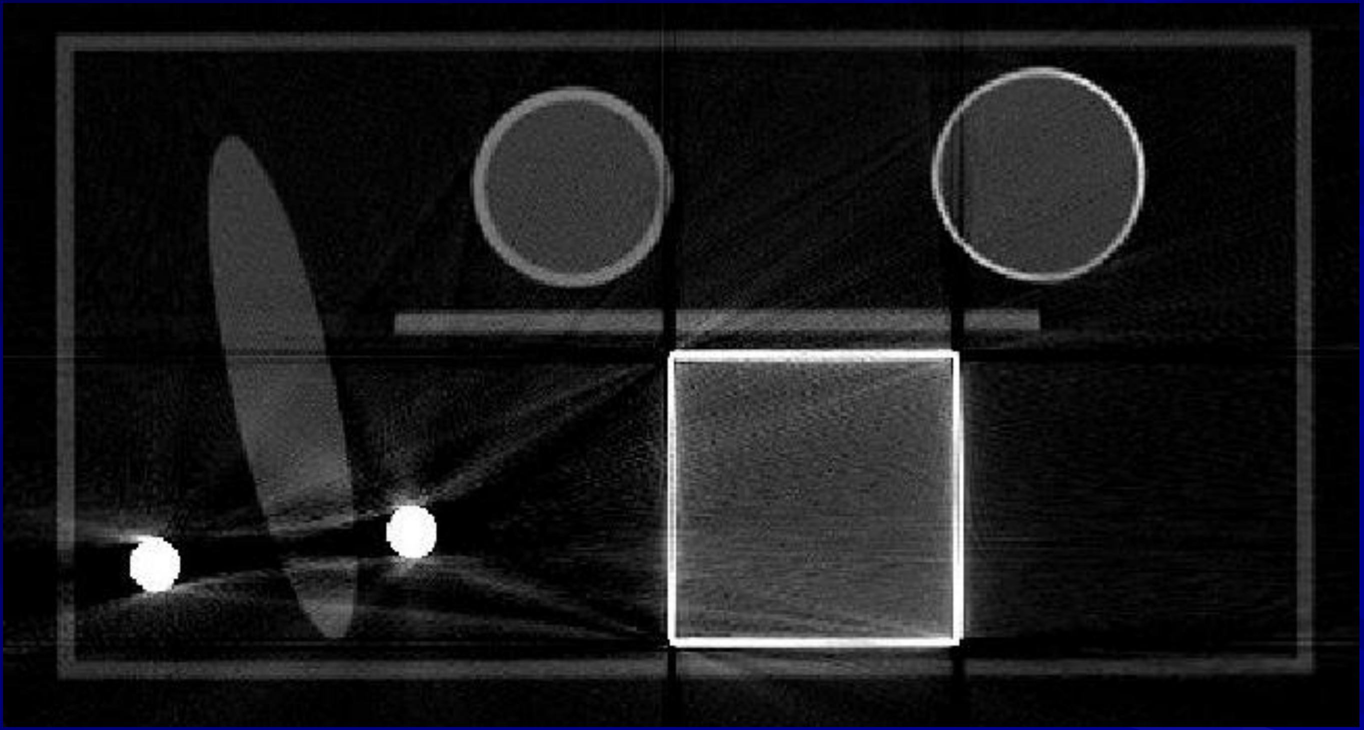
Simulation Task

Taly Gilat Schmidt, PhD
Department of Biomedical Engineering
Marquette University

Conclusions

- Created simulated projection data from Imatron scanner and created standardized mathematical phantoms
- Simulated data matched the values, noise, scatter, artifacts of experimental data
- Simulated data was useful for recon development
- Physics is validated, next step is object complexity
- Simulations can generate library of data for algorithm development, system testing, performance predictions

Suitcase Phantom



- Deliverable is simulated data and phantom definition
- Simulated data reconstructed by filtered backprojection

Potential Impact of Simulations

- Performance of future scanners may be simulated to reduce time to market and cost
- Investigate range of system parameters
- Overcome the issue of limited data for system testing and training
- Facilitate algorithm development
  - Known ground truth
- May be possible to predict detection performance of new scanners

Task Objectives

- Simulation tools for security scanners not known to exist in the public domain
- Develop common set of numerical phantom definitions and simulated data
- Leverage concepts and tools in the medical imaging field to develop simulation tools for future projects
- Validate that simulated data replicates experimental data

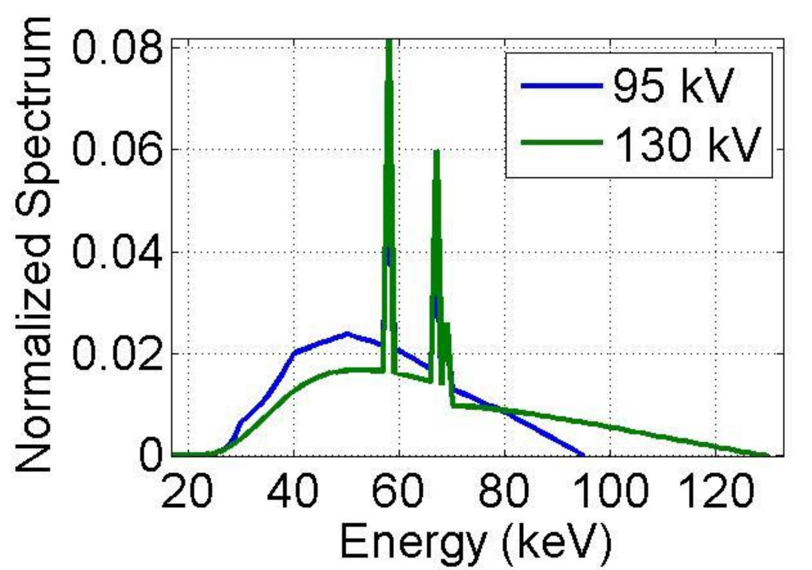
Simulation Methods

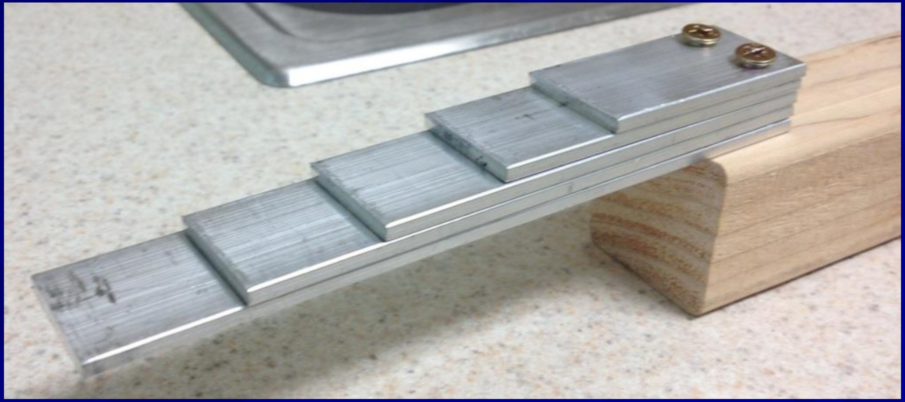
- Raytracing software analytically calculates intersection of rays with primitive shapes
  - Cylinders, ellipses, boxes, cones
  - Models focal spot and detector aperture
- Monte Carlo simulations estimate scatter signal
- Matlab scripts combine ray-tracing, scatter, photon noise, and electronic noise.

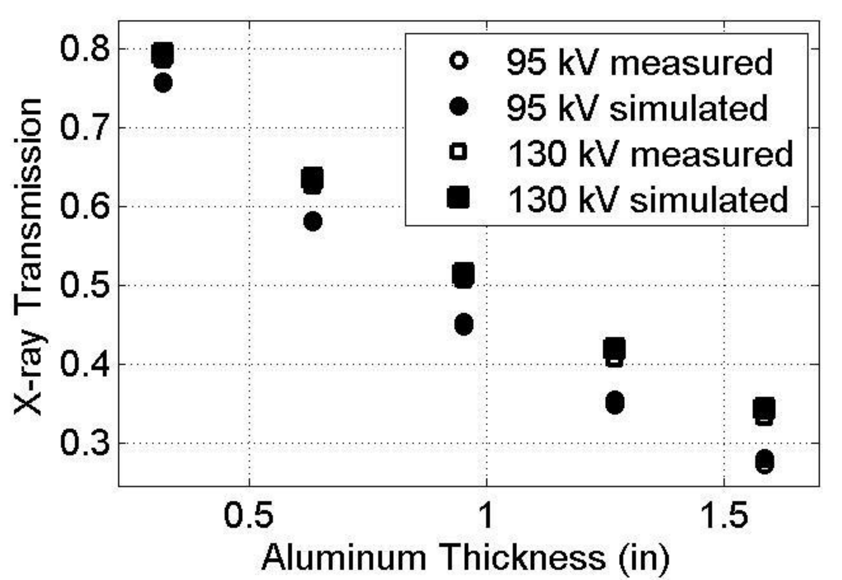
Validation

- Match the Imatron spectra
- Match the Imatron fluence
- Match the Imatron geometry
- Match the reconstructed HU mean and standard deviation
- Match the scatter level and artifacts

X-ray Spectra

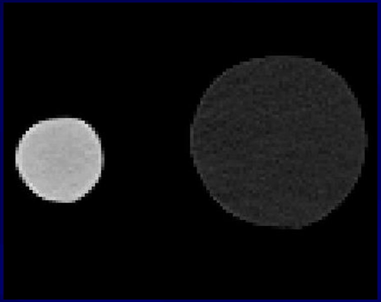


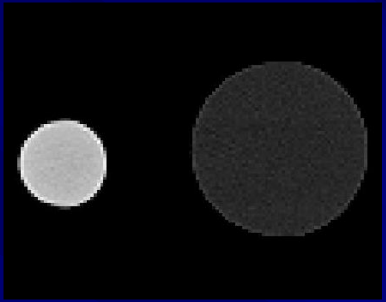


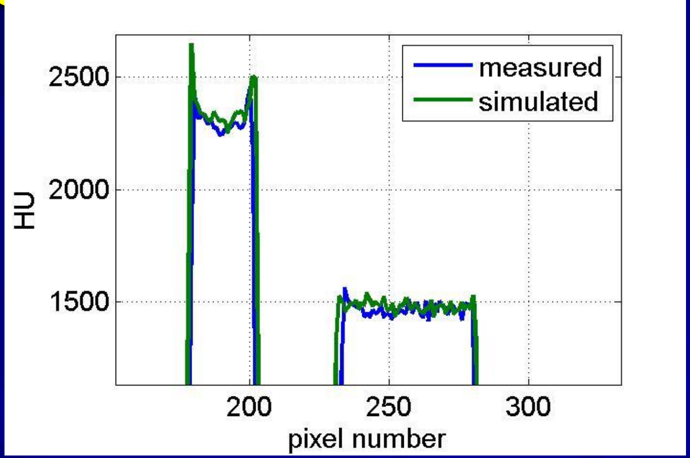


Graphite and Magnesium

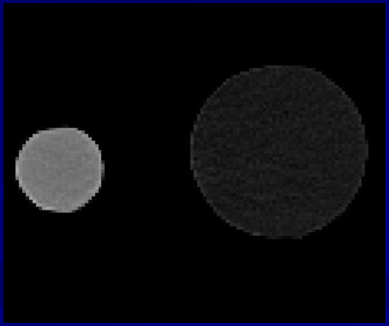
95 kV

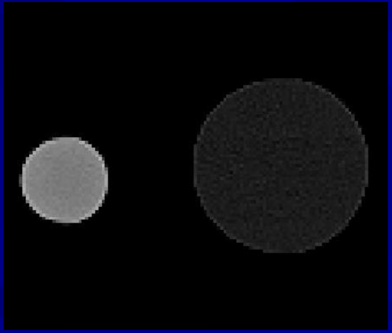


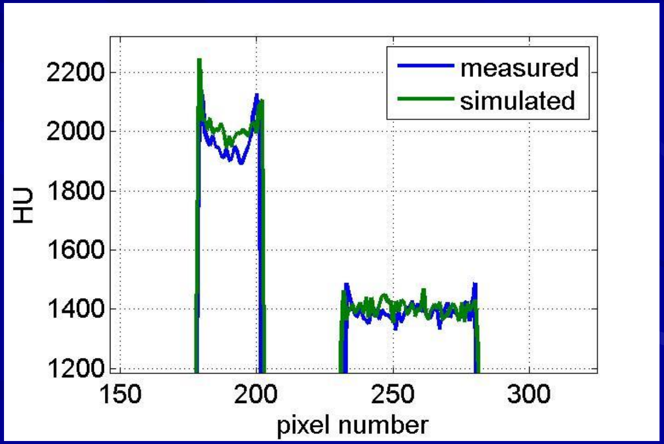




130 kV



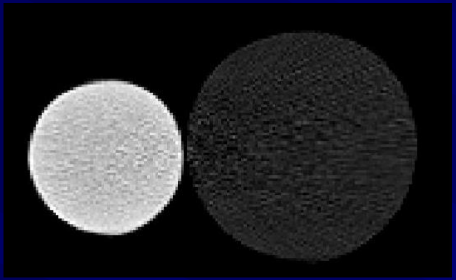


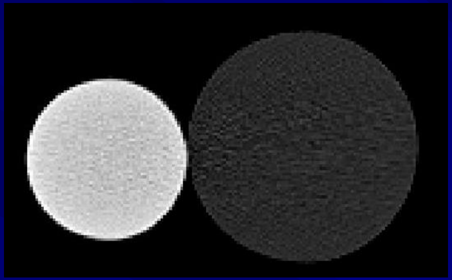


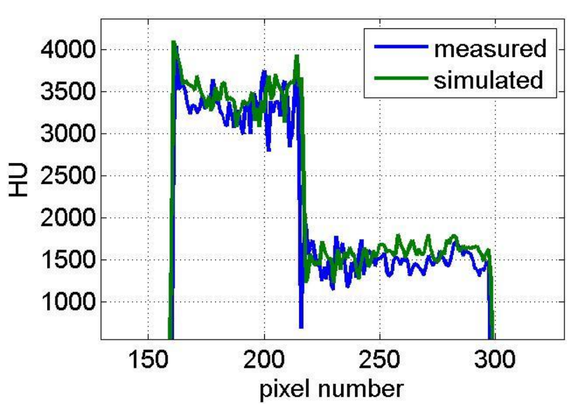
Measured **Simulated** **Horizontal Profile**

Graphite and Aluminum

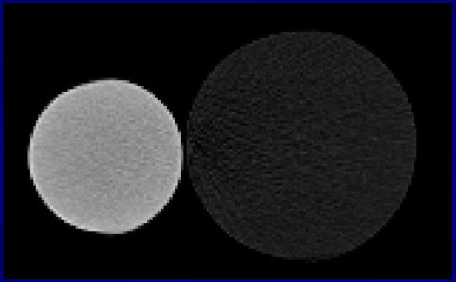
95 kV

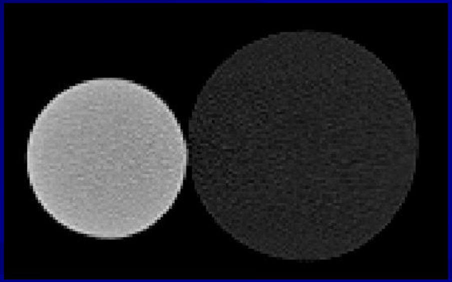


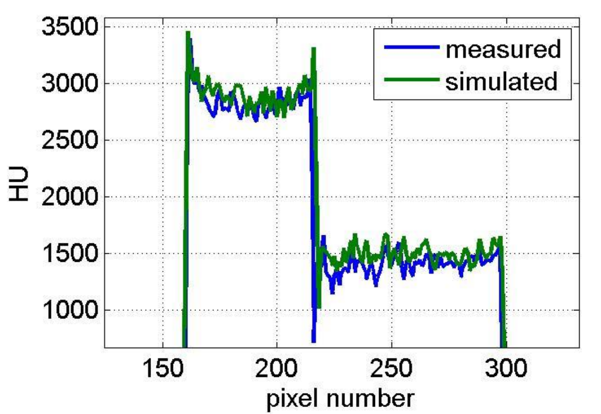




130 kV

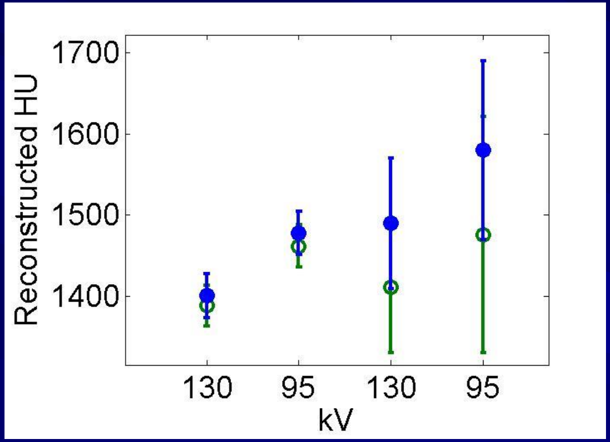




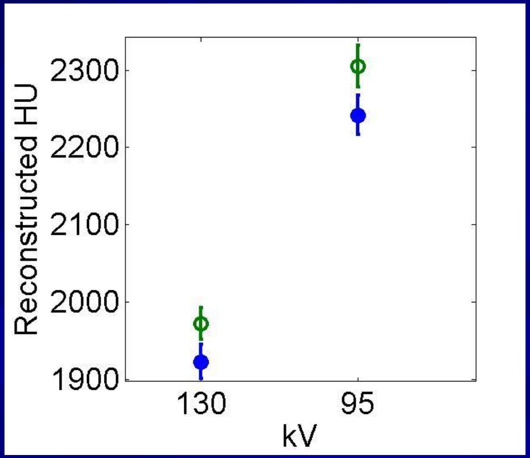


Measured **Simulated** **Horizontal Profile**

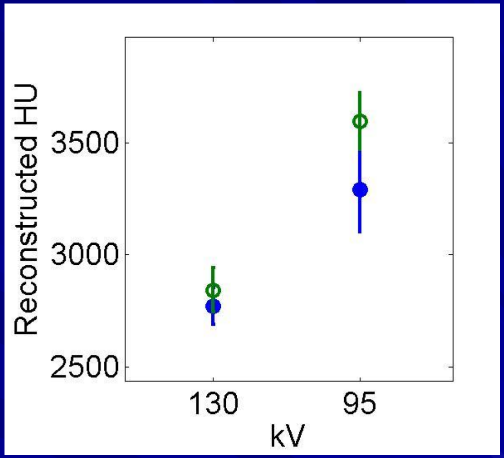
Reconstructed HU Values



Graphite



Magnesium



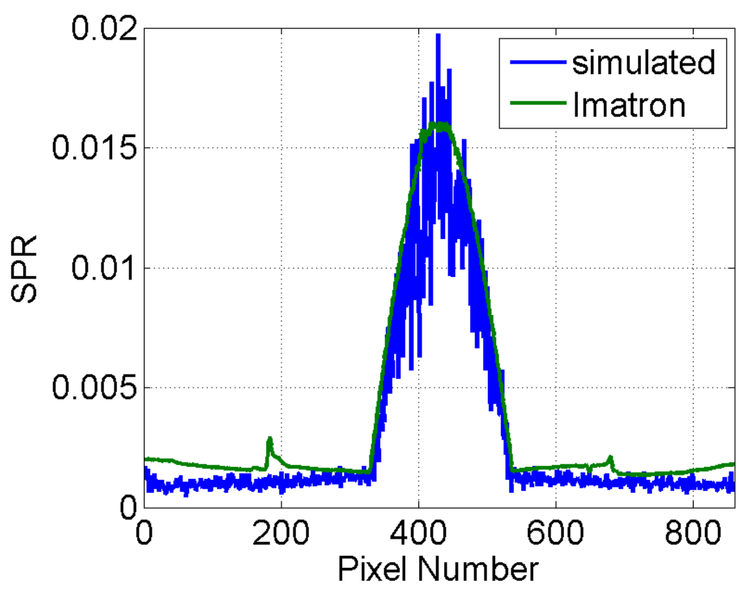
Aluminum

● Experiments

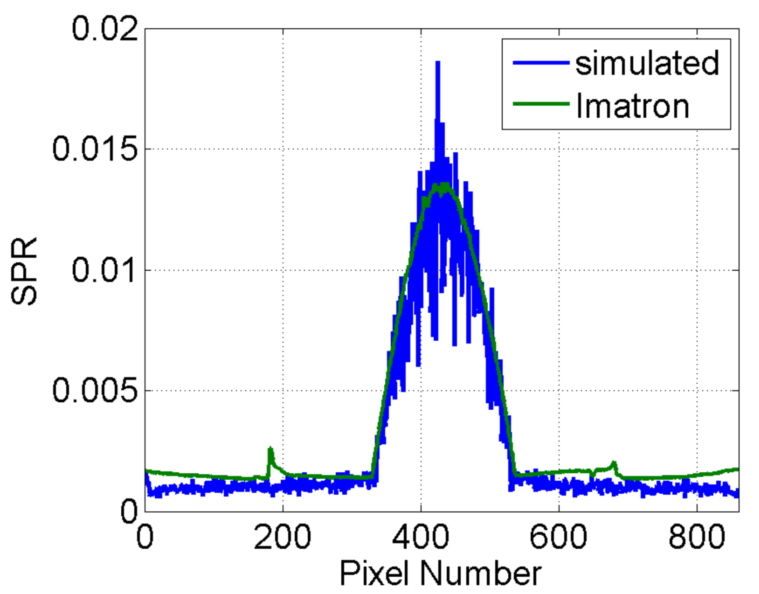
○ Simulations

Good agreement between mean and std values

Scatter-to-primary ratio



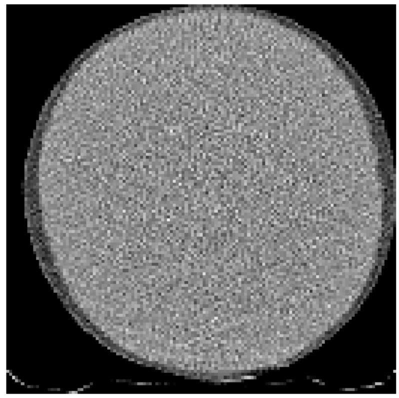
95 kV

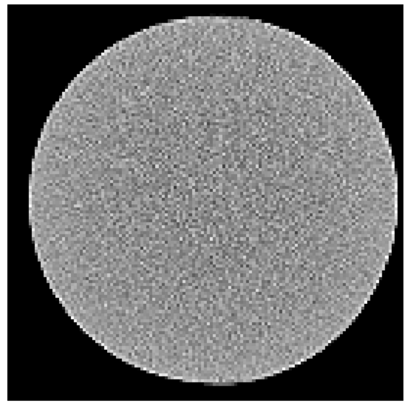


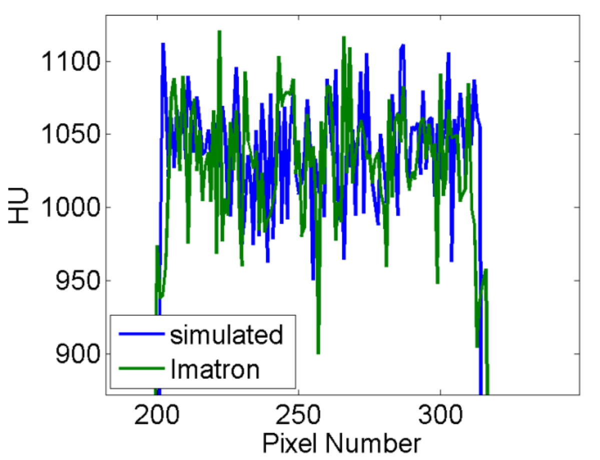
130 kV

Images Reconstructed With Scatter

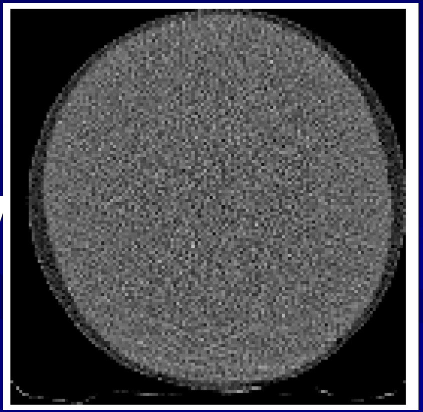
95 kV

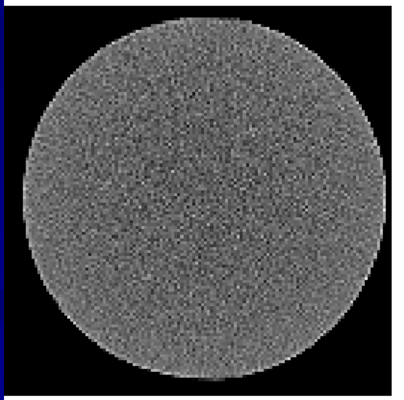


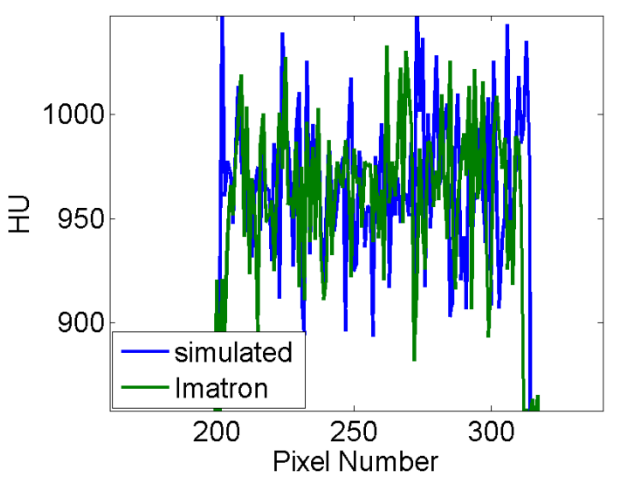




130 kV



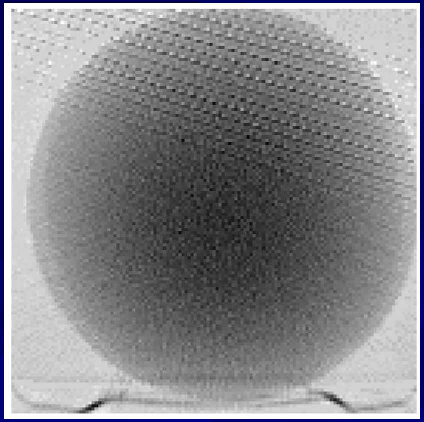


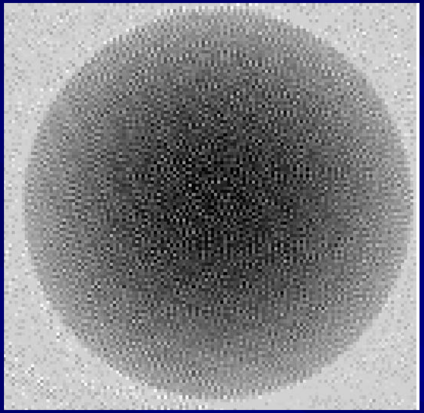


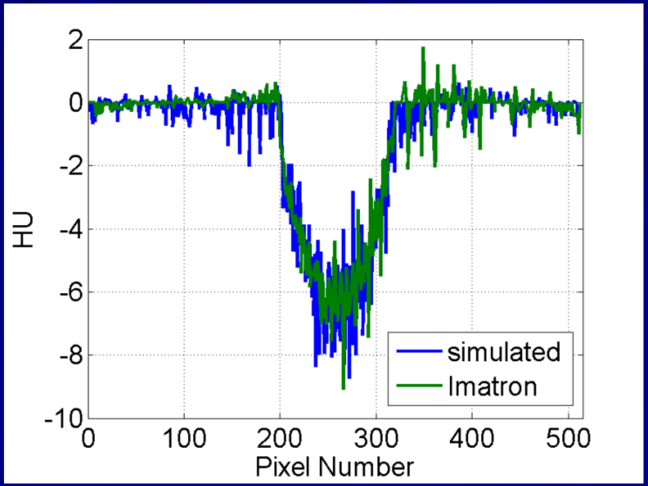
Imatron **Simulated** **Horizontal Profile**

Scatter Artifact

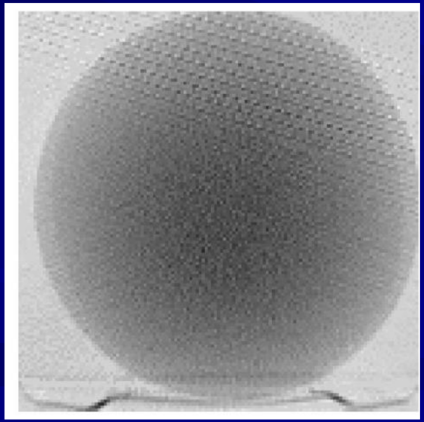
95 kV

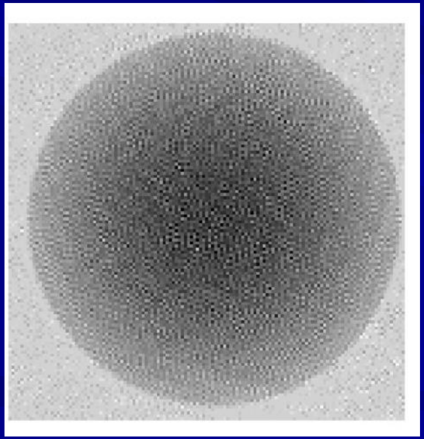


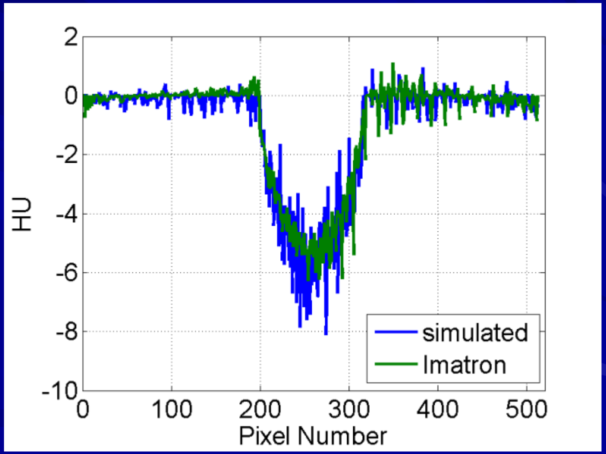




130 kV

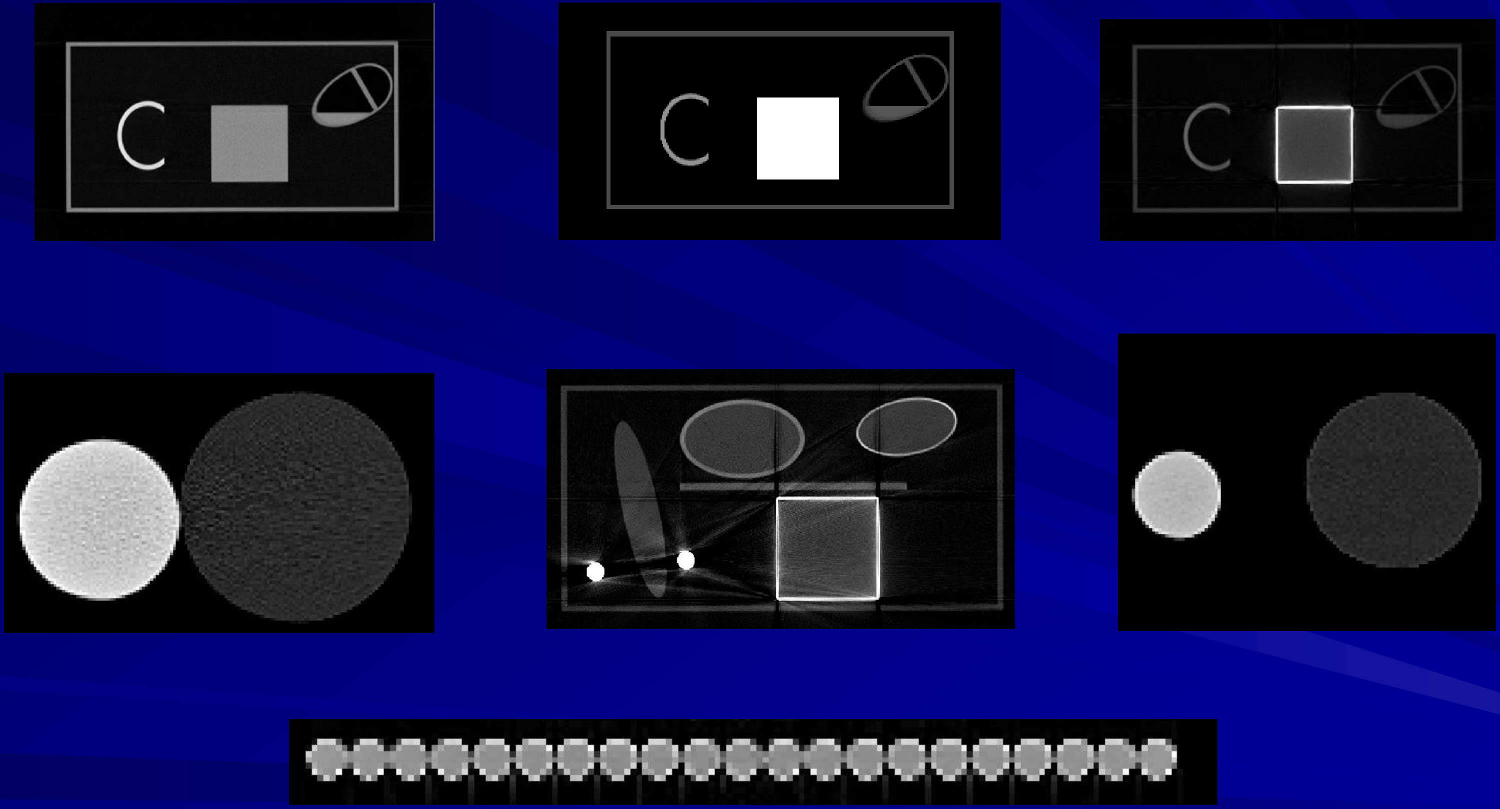






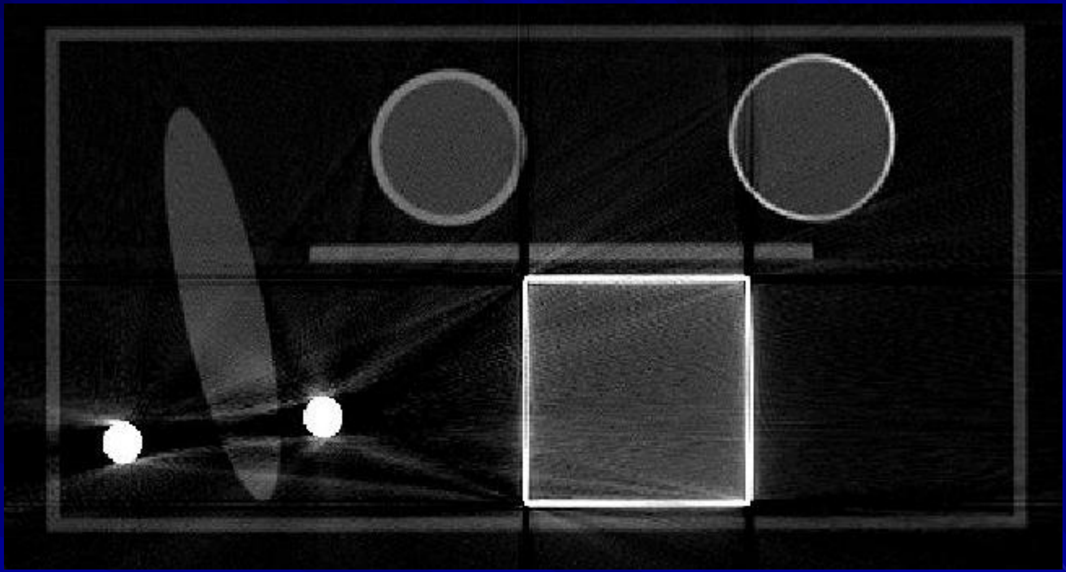
Imatron **Simulated** **Horizontal Profile**

Suitcase Phantoms and Data



Phantom ‘Standard’

- Water objects in four configurations / containers
- Rubber sheet object
- Metal artifacts



```
Phantom
//Text Case
{ [ Box: x=0 y=0 z=0  dx= 39 dy= 20 dz=28 ] formula=C2H4 rho=0.95 }
{ [ Box: x=0 y=0 z=0  dx= 38 dy= 19 dz=27 ] formula=C8H8 rho=0.1 }


// Text Block
{ [ Box: x=2 y=-2 z=5.5 dx=9 dy=9 dz=12 a_x(0.707,0,0.707)
a_y(0,1,0) ] formula=Al rho=2.699 }
```

Feedback from Recon Teams

- Simulations helpful for algorithm development and testing
- Helpful to have a true gold standard
- Validated spectra and fluence models helpful for reconstruction development

Lessons Learned

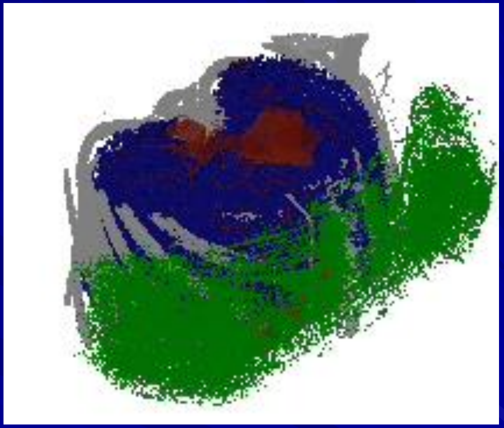
- Defining complex objects with primitive shapes is difficult, limited, and time consuming
- Primitive shape definition varies across software packages
  - Forbild, g3d, GEANT all use different definitions
- Scatter must be modeled to have realistic streak artifacts
- Good simulations require detailed information from scanner vendor

Successfully validated the physics simulation models

More development required to fully realize impact on security community

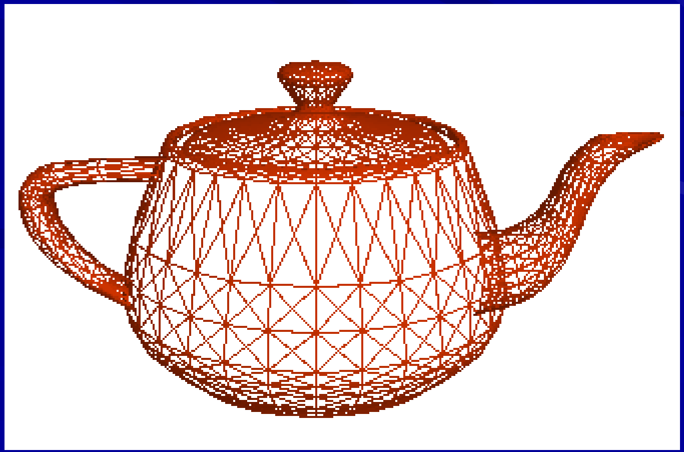
Voxelized Models

- Represent complex objects by cartesian grid of voxels. Each voxel has one μ value
- Ray-tracing algorithms available
- Model heterogenous texture
- Convert an experimental image set into a software phantom
- Require large memory
- Partial volume limitations



Polygonal Mesh Objects

- Defines the object surfaces through mesh points
- Standard CAD output (e.g. .stl)
- Numerous object models available in public domain (e.g., Google sketch up 3D warehouse)
- Ray tracing algorithms available



Next Step: Complex Objects

- Design simulation software that can handle multiple object types
  - Polygonal meshes (CAD output)
  - Voxelized models
  - Take advantage of public domain libraries
- Some Monte Carlo packages already do this
  - GEANT, pen mesh
- Need software architecture that makes it easy to handle multiple data types
- Develop packing software

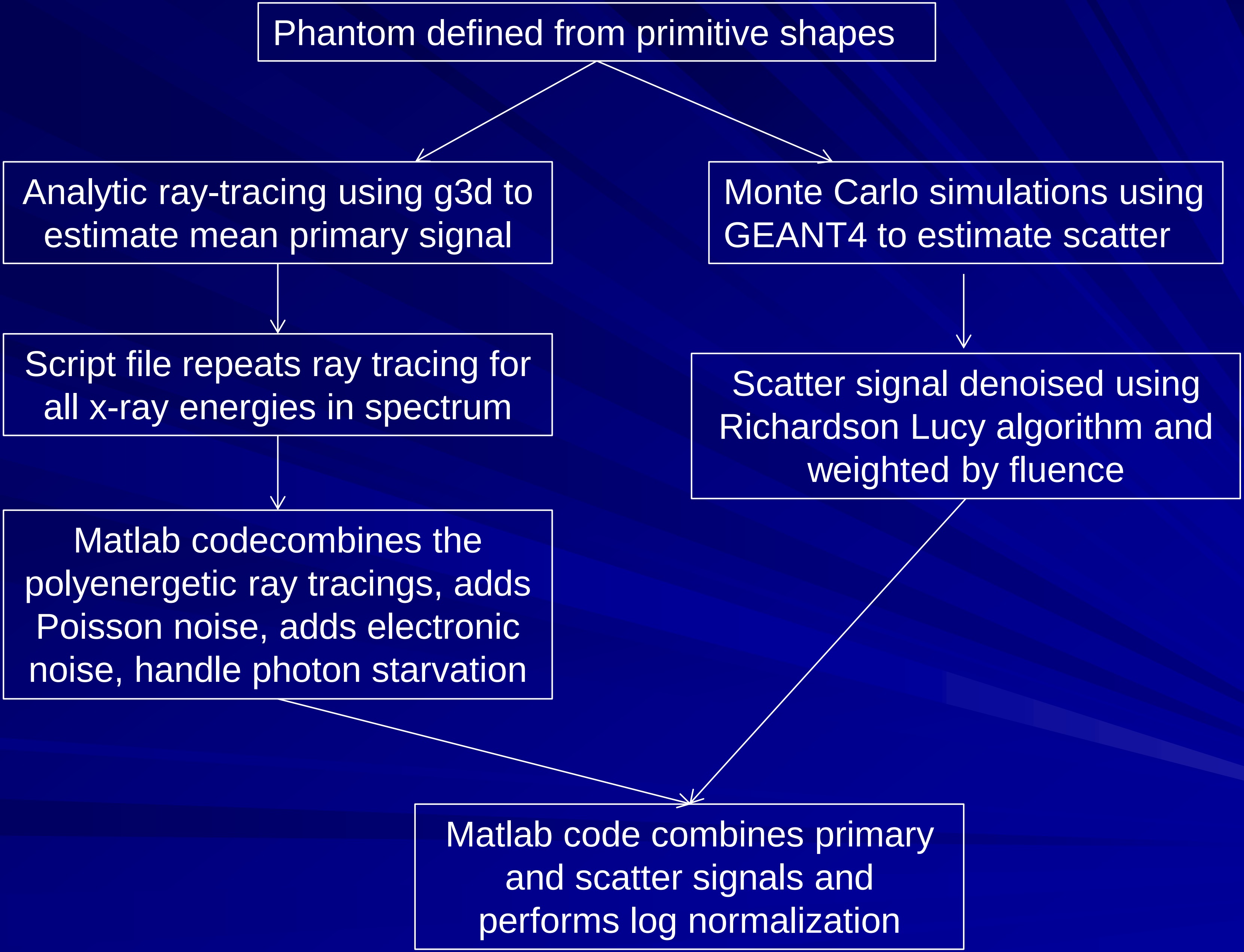
Next Step: Object Oriented Simulation Software

- Can add new scanner geometries without changing existing code
- Can add new object types without changing existing
- Efficient parallel, open-source development
- Easier new user adoption – low level details hidden from users

Conclusions

- Created simulated projection data from Imatron scanner and created standardized mathematical phantoms
- Simulated data matched the values, noise, scatter, artifacts of experimental data
- Simulated data was useful for recon development
- Physics is validated, next step is object complexity
- Simulations can generate library of data for algorithm development, system testing, performance predictions

EXTRA SLIDES



Phantom defined from primitive shapes

Analytic ray-tracing using g3d to estimate mean primary signal

Script file repeats ray tracing for all x-ray energies in spectrum

Matlab codecombines the polyenergetic ray tracings, adds Poisson noise, adds electronic noise, handle photon starvation

Monte Carlo simulations using GEANT4 to estimate scatter

Scatter signal denoised using Richardson Lucy algorithm and weighted by fluence

Matlab code combines primary and scatter signals and performs log normalization

Spectral Shape Validation Methods

- Calculated the transmission through the aluminum wedge phantom from the Imatron sinogram data
- Calculated the transmission through a simulated aluminum wedge phantom using the modeled spectra
- Compared aluminum transmission plots
- Compared images reconstructed from the KACH_EDEC_1 and KACH_EDEC_2 datasets
- The *.nrm Imatron sinograms were used (after normalization, before beam hardening correction and interpolation)
- The simulated geometry was the physical Imatron geometry
- Both the measured and simulated sinograms were interpolated to equiangular fan beam data and reconstructed using identical algorithims
- Beam hardening correction was not applied to either the measured or simulated data
- The mean in the test objects was compared for simulated and measured data

Photon Fluence Validation Methods

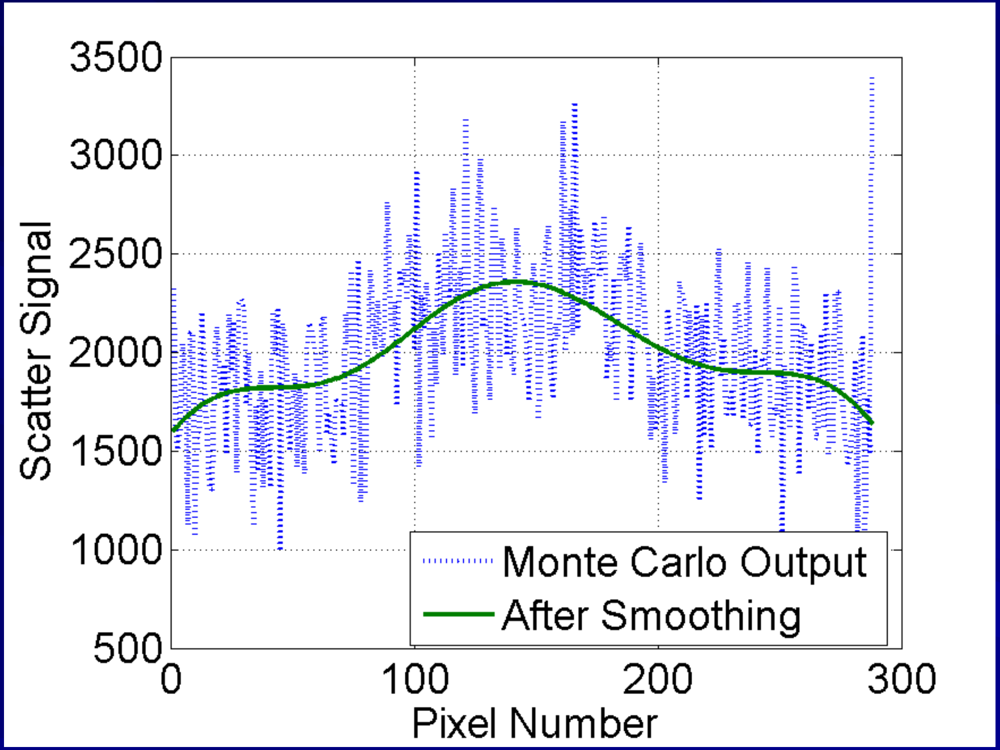
- The photon fluence estimation method detailed by Seemeen was performed using air scans from 130 kV and 95 kV acquisitions
  - ~120,000 photons per ray for the 95 kV spectrum
  - ~100,000 photons per ray for 130 kV spectrum
- The noise standard deviation was calculated in images simulated assuming this photon fluence and compared to images reconstructed from Imatron sinograms

  -same recon routines for both cases
- The photon fluence estimates were adjusted so that the simulated standard deviation matched the measured data
  - ~180,000 photons per ray for the 95 kV spectrum
  - ~170,000 photons per ray for 130 kV spectrum

Scatter Validation

- Combined simulated ray tracing and Monte Carlo images of the water_2000ml phantom
- Compared simulated reconstructed images (including scatter) to Imatron images reconstructed without scatter correction
- Compared the scatter artifact (image_with_scatter – scatter_corrected_image) for both simulated and Imatron data

Smoothing of Monte Carlo Output

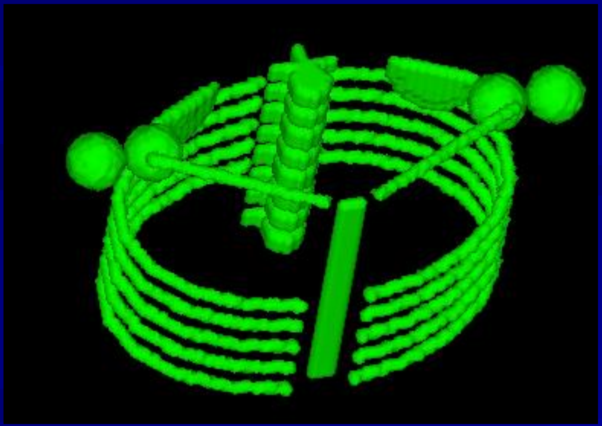


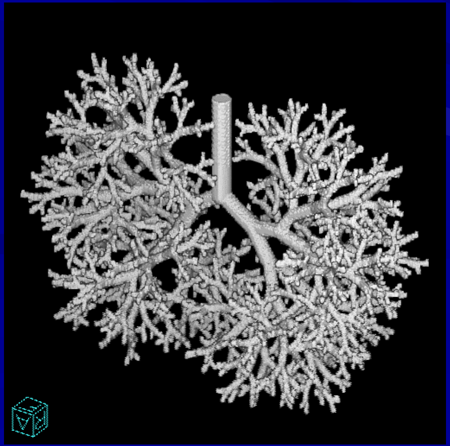
*The amount of smoothing is adjustable

After smoothing, the scatter signal is scaled to adjust for differences in fluence in the MC simulations compared to ray-tracing. Poisson noise is added to the scaled scatter signal , which is then added to the ray tracing generated primary signal

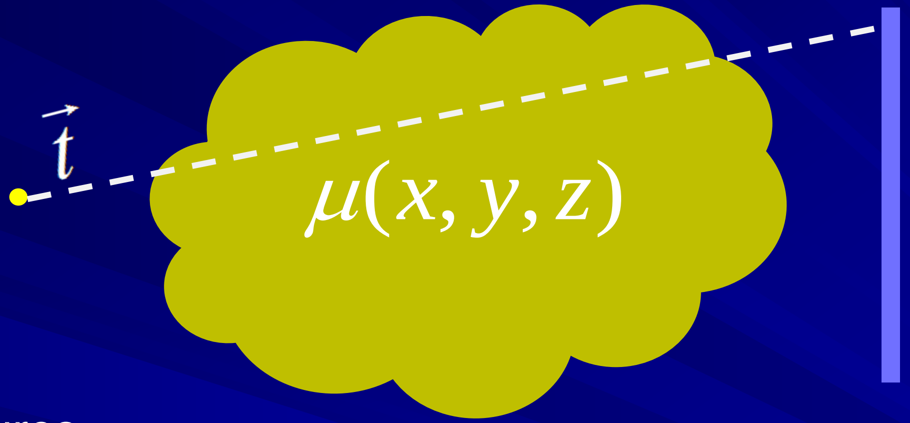
Analytical Ray Tracing

- Analytical expressions for the intersection of a line and 3D objects
  - -spheres, ellipses, cylinders, boxes, cones...
- g3d software (Carl Crawford) performs these calculations





More realistic, complex objects



source

detector

$$\bar{N} = \int N_o(E) e^{\left(-\int \mu(x,y,z) d\vec{t}\right)} dE$$

How to calculate line integrals through μ(x,y,z)?

Voxelized Ray Tracing