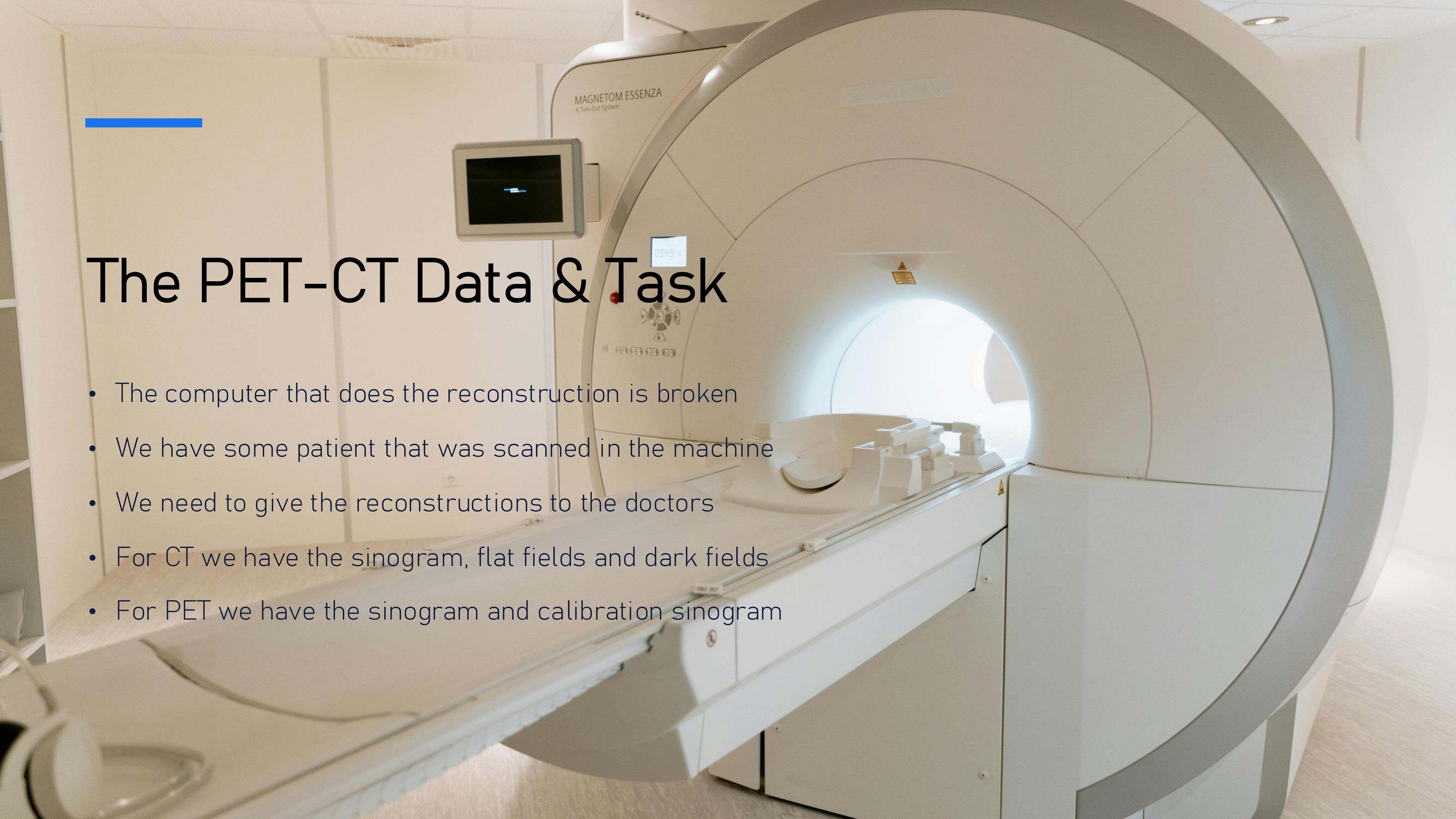
LEWIS MCCONKEY

Multimodal Imaging for Cancer Detection & AI Segmentation

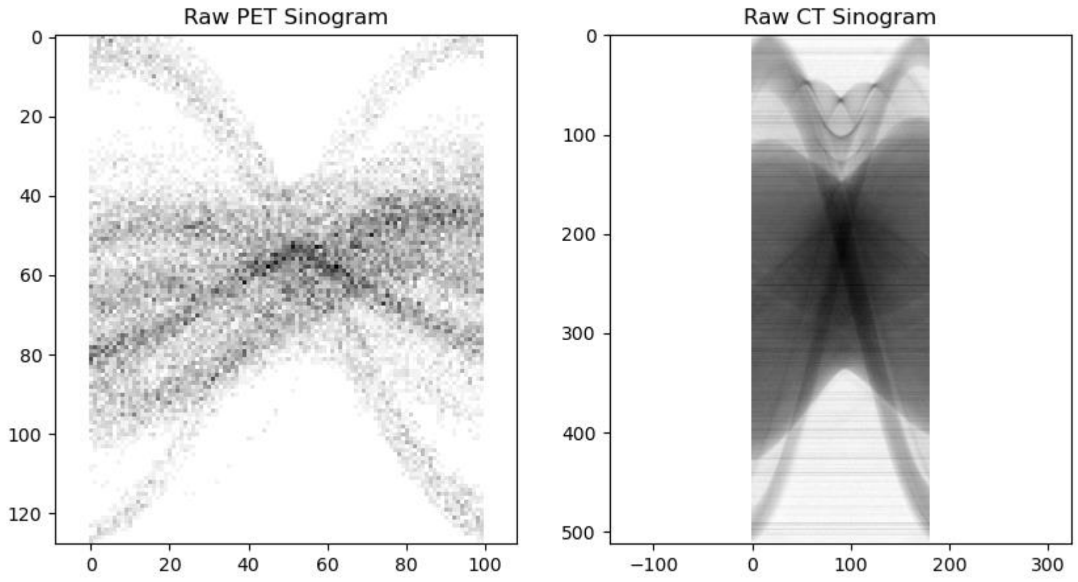
3 main focuses

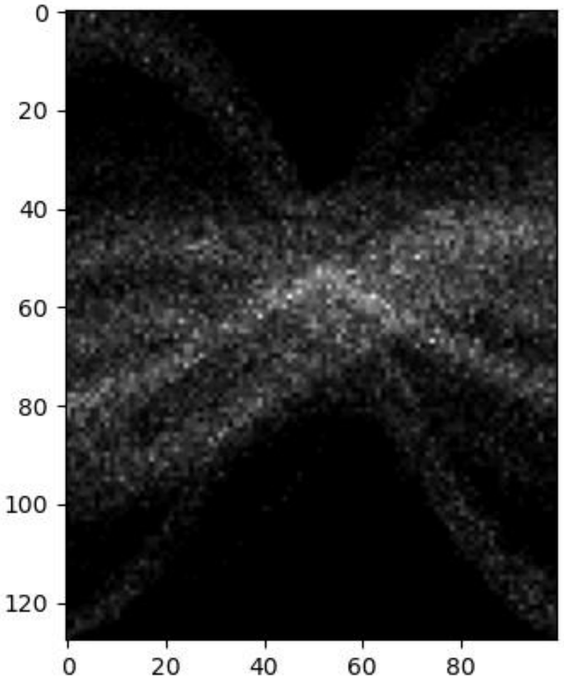
- PET-CT Image Reconstruction
- MRI Image Denoising
- CT Image Segmentation & Classification

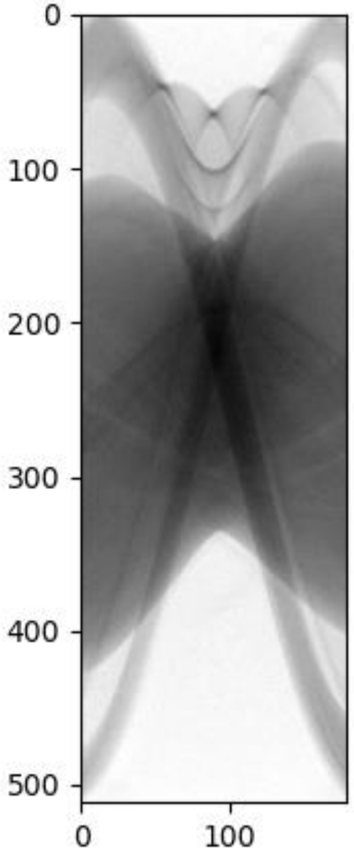


The PET-CT Data & Task

- The computer that does the reconstruction is broken
- We have some patient that was scanned in the machine
- We need to give the reconstructions to the doctors
- For CT we have the sinogram, flat fields and dark fields
- For PET we have the sinogram and calibration sinogram


Raw PET Sinogram
0
20
40
60
80
100
120
0
20
40
60
80
100
Raw CT Sinogram
0
100
200
300
400
500
−100
0
100
200
300


0
20
40
60
80
100
120
0
20
40
60
80

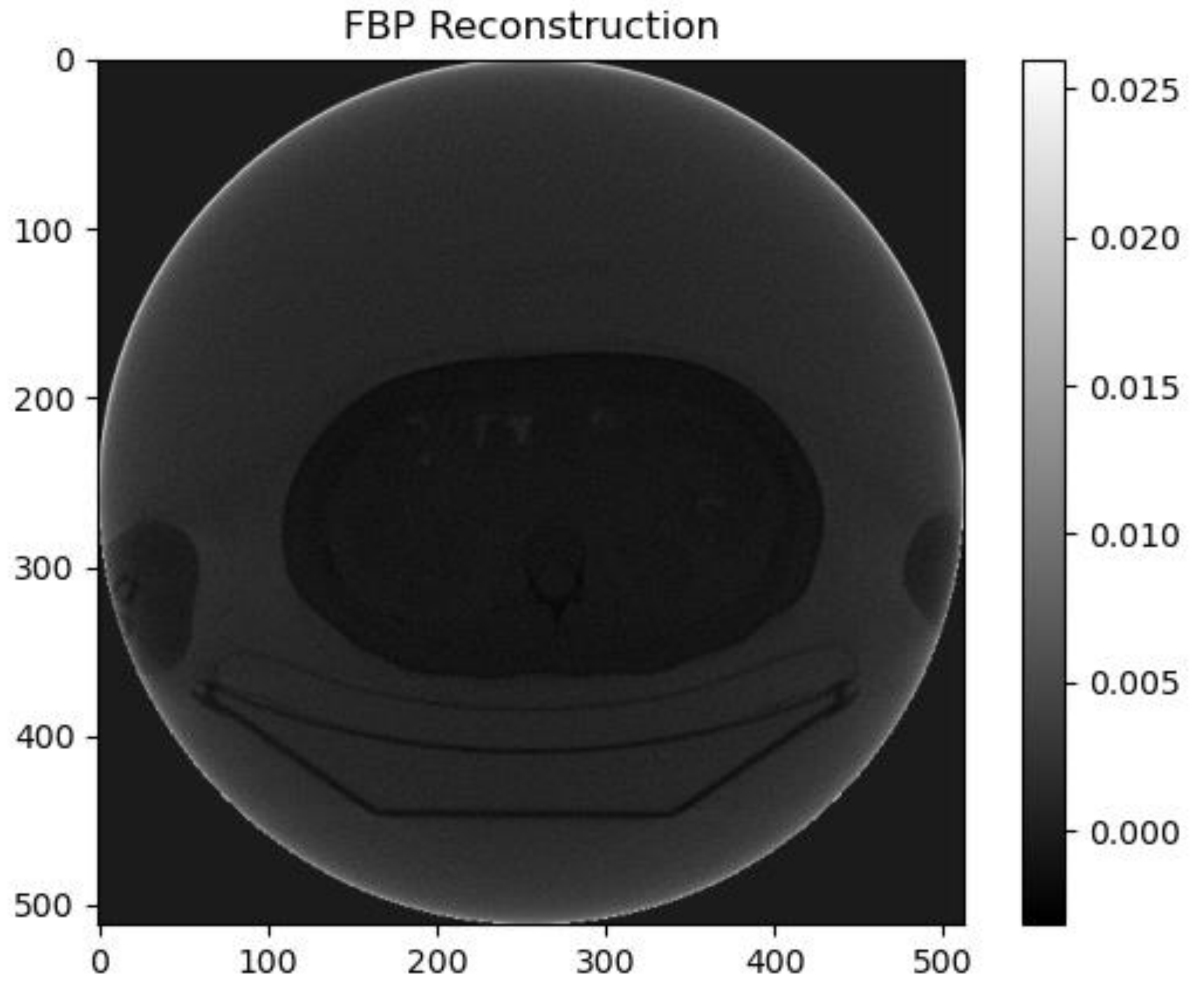

0
100
200
300
400
500
0
100

CT Reconstruction

- Reconstruct with Filtered backprojection
- Direct reconstruction method
- Uses a filter and backprojects

$$f_{fbp}(x,y) = \int q_\theta(x cos(\theta) + y sin(\theta)) d\theta$$

$$\text{where } q_\theta(t) = \int P_\theta(\omega)|\omega| e^{2\pi i \omega t} d\omega$$



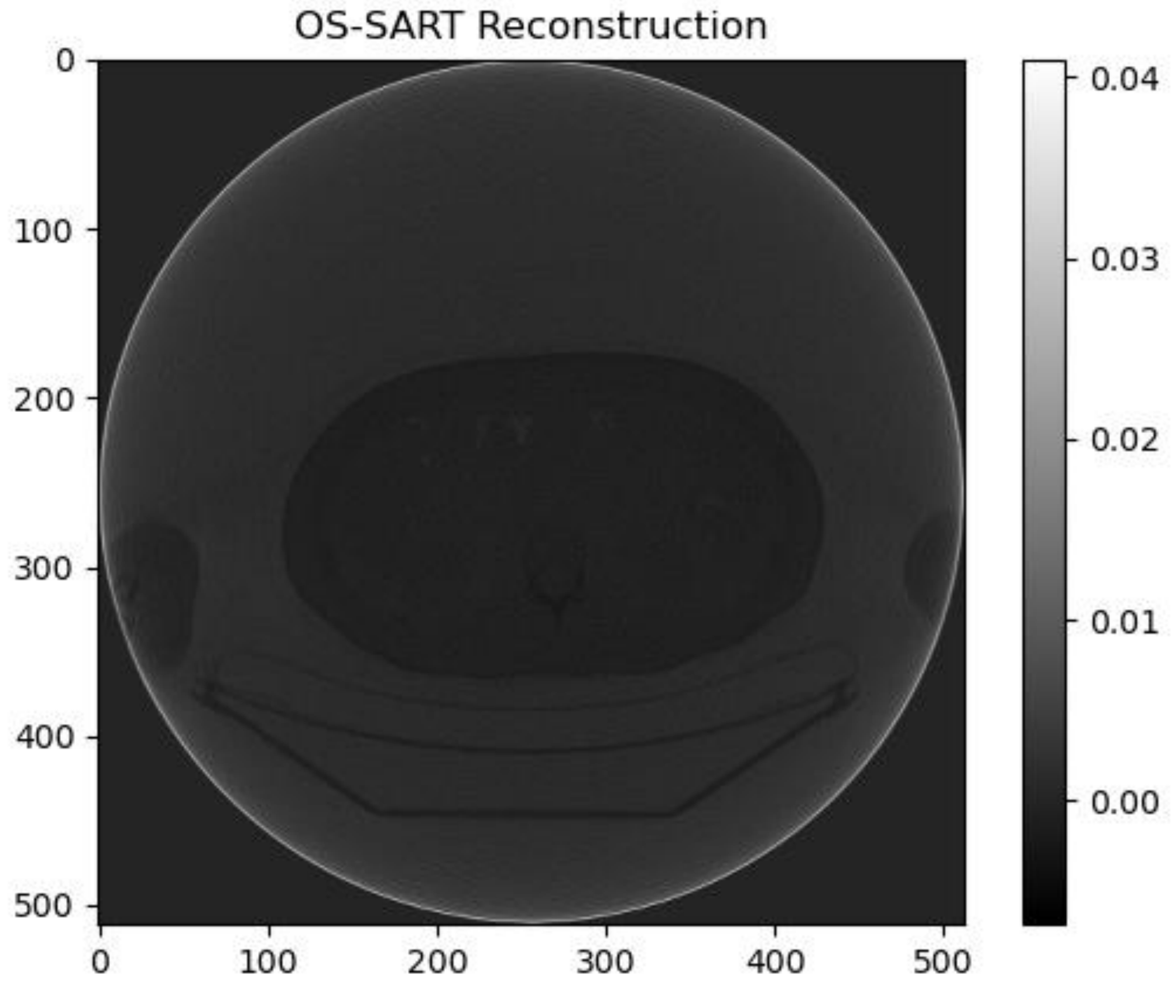
CT Reconstruction

- Reconstruct with OS-SART

$$x^{K+1} = x^k + \gamma A_i^T(A_i x^k - b)$$

- Iterative algorithm
- Tuning parameters
- Observations

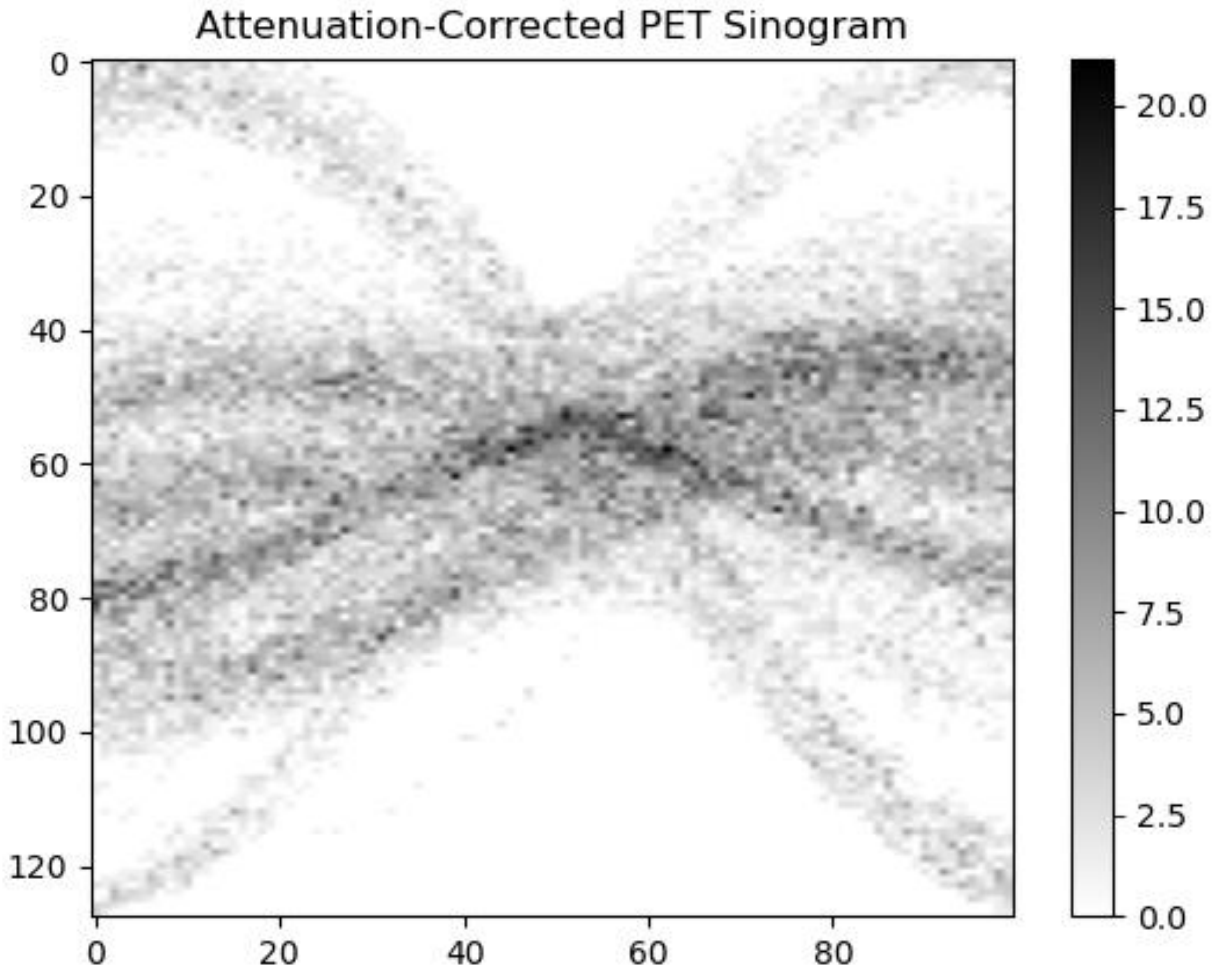
Algorithm	Iterations	time (s)
OS-SART	10	8-14
SIRT	50	132
FBP	N/A	0.25-0.5



Attenuation correction

- Photon Attenuation
- Observations
- AC vs NAC

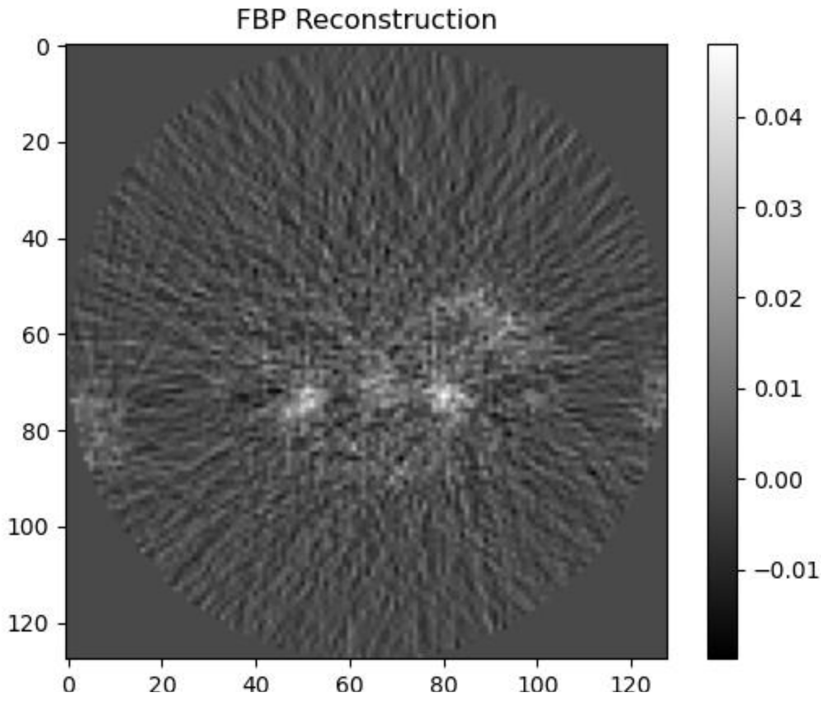
$$\boxed{I(l) = I_0 e^{-\int \mu(l)dl}}$$

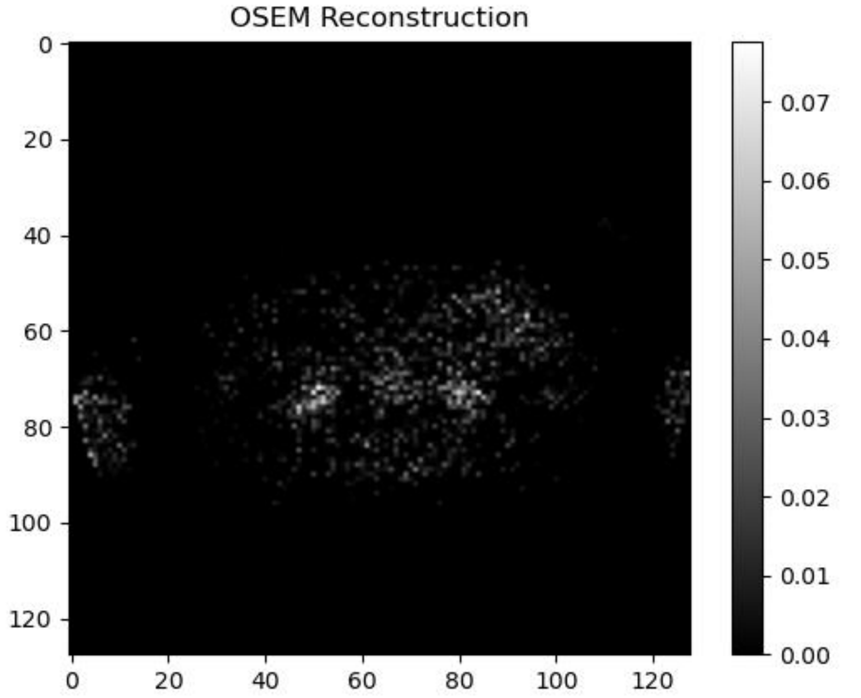


PET Reconstruction

- Reconstruct with fbp, OSEM and MLEM

$$x^{k+1} = x^k A_i^T \left(\frac{b}{A_i x^k} \right)$$



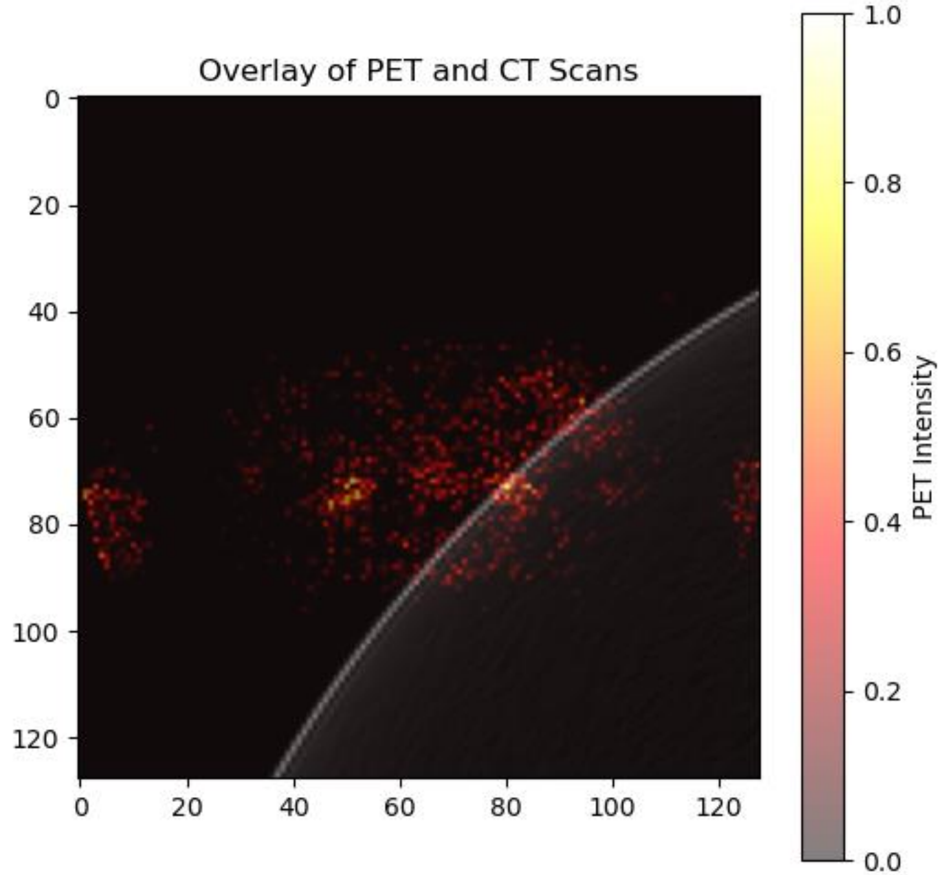


Extra

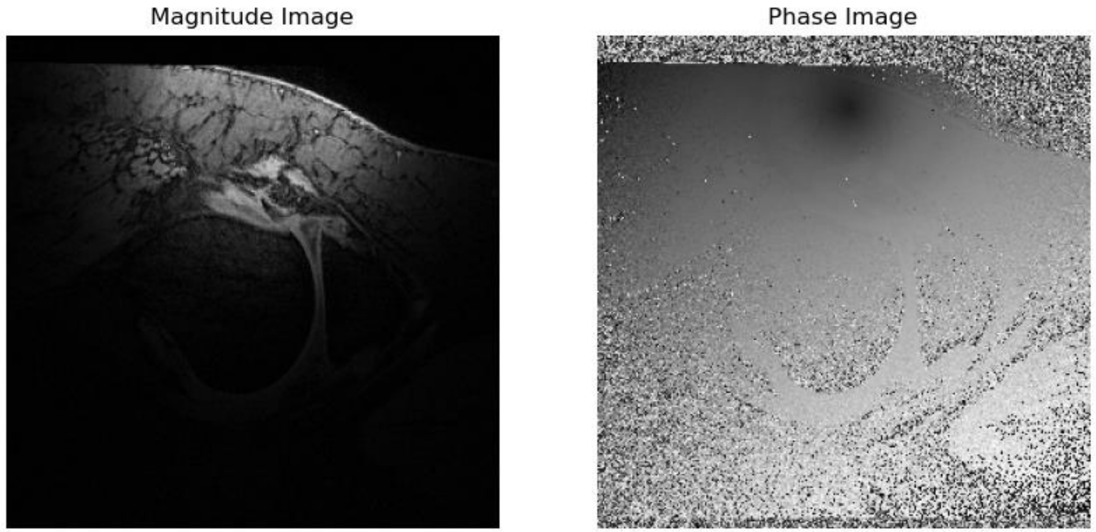
- Overlay of PET-CT scans
- TOF-PET scanner

$$\Delta x = \frac{c \cdot \Delta t}{2}$$

- OSEM is the most used algorithm in PET
- PET-MR scanners exist



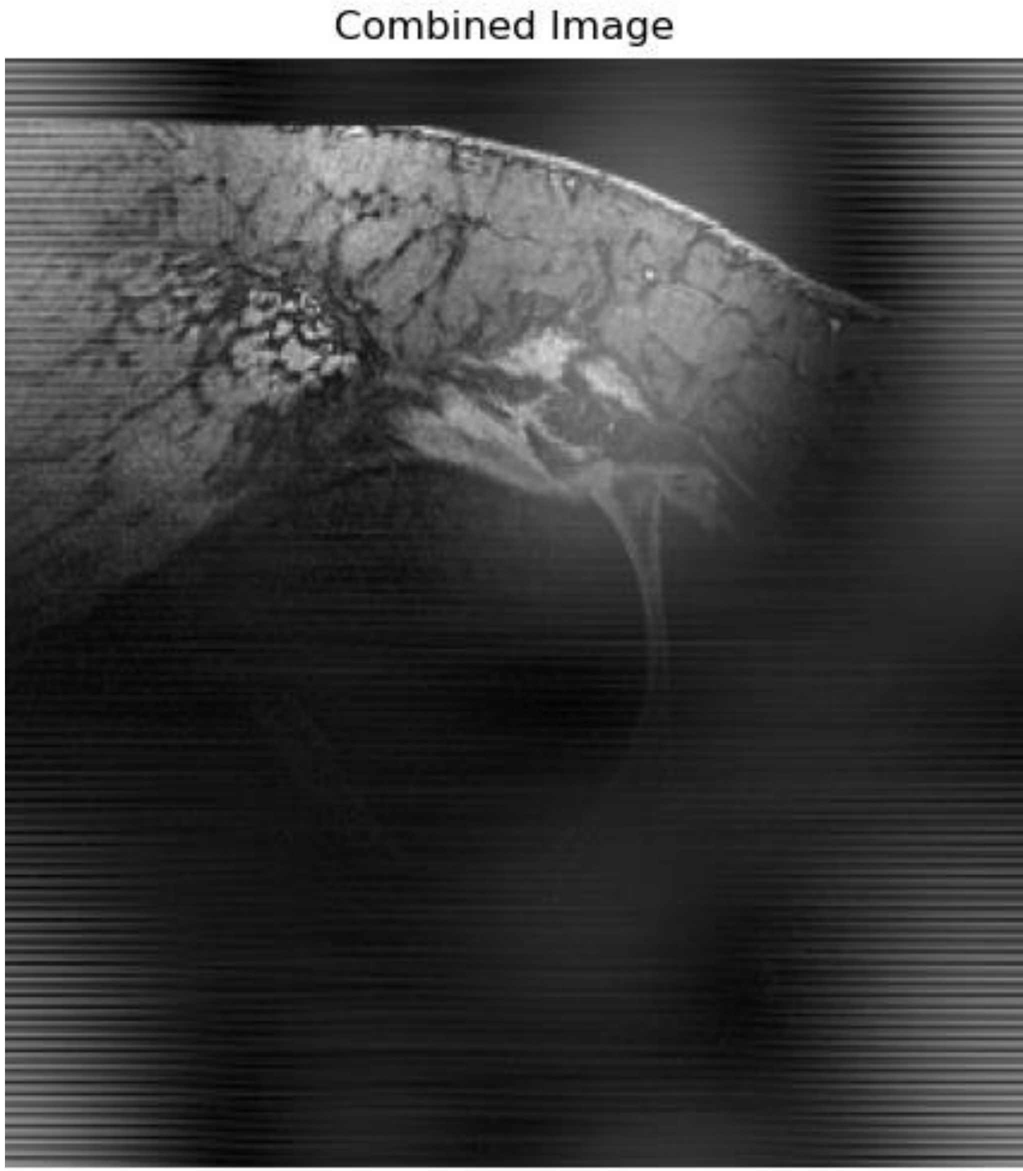
MRI Image Denoising



- 3D knee image with 6 coil recievers
- Contains artifacts
- Magnitude & phase image of 1st coil

Combined Image

- Combine all coils into one image
- Observations
- SENSE

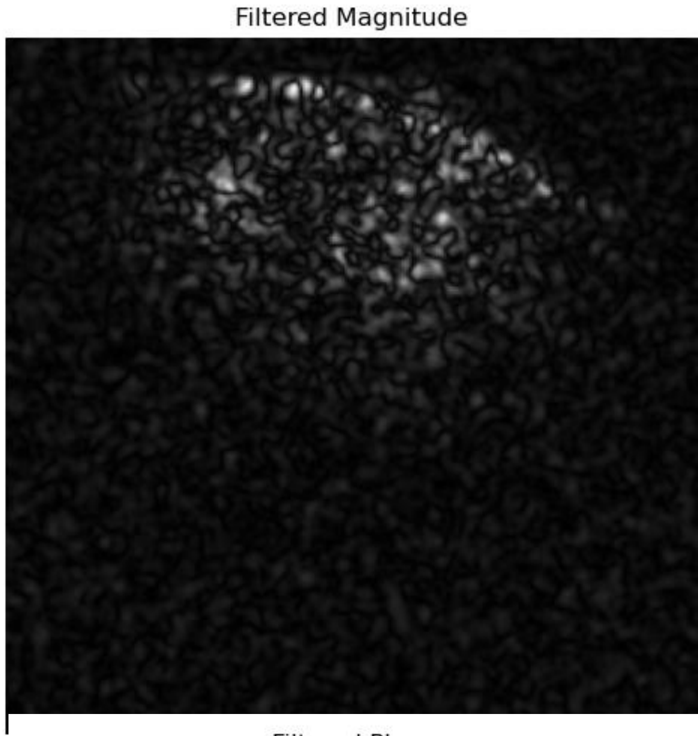


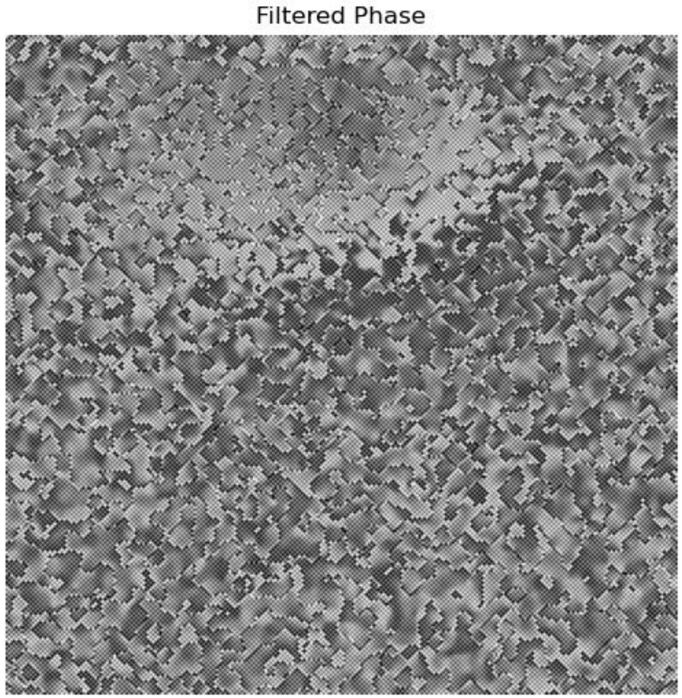
Denoising Methods

- 3 denoising methods: Gaussian, Bilateral and Wavelet

Low-Pass Butterworth Filter

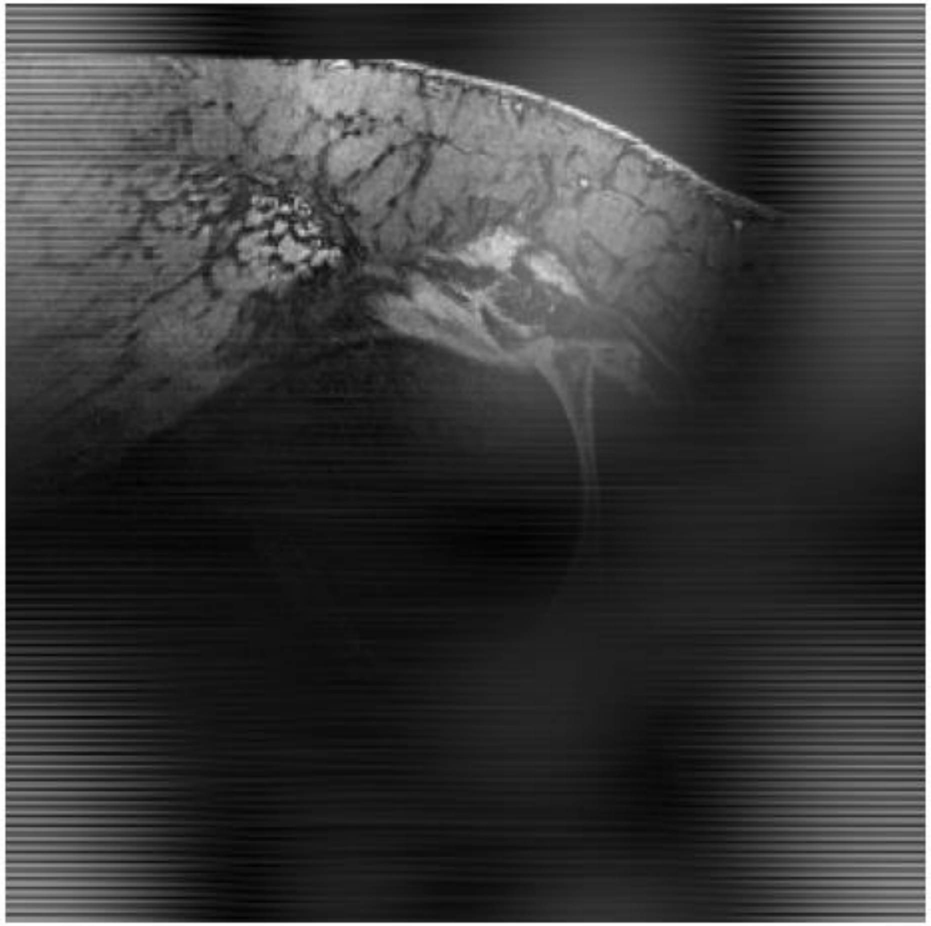
- def butterworth_lowpass_filter(shape, D0=30, n=2): P, Q = shape[0], shape[1]
  u = np.arange(P) - P // 2
  v = np.arange(Q) - Q // 2
- U, V = np.meshgrid(u, v, indexing='ij') D = np.sqrt(U**2 + V**2)
  H = 1 / (1 + (D / D0) ** (2 * n)) return H

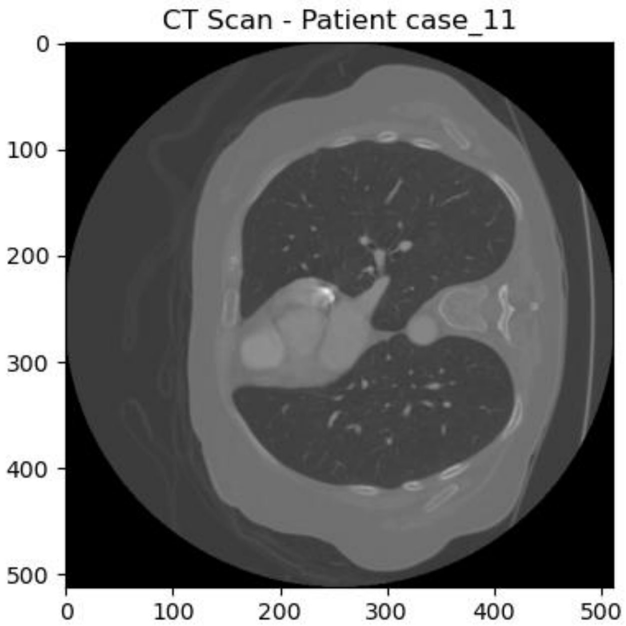


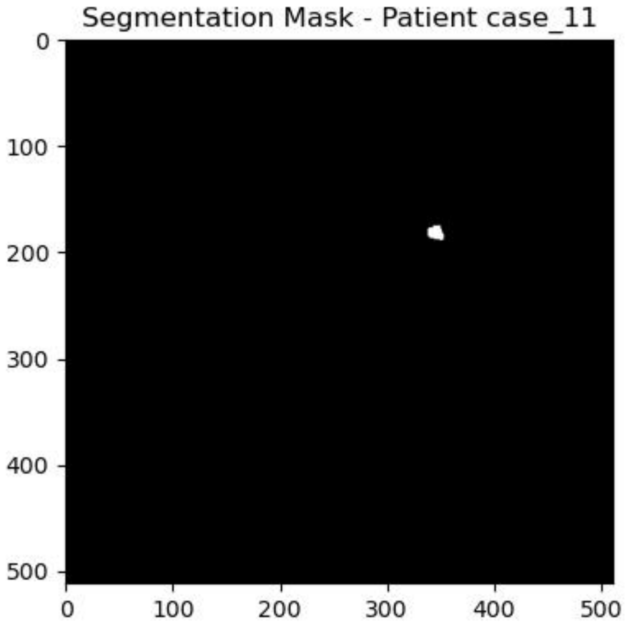


Recreate Combined Image

- Using wavelet to recreate a new combined image
- Methods to improve this:
- Increase K-space sampling
- SENSE, GRAPPA…
- Other denoising methods







CT Image Segmentation & Classification

- CT scans for a selection of 40 patients of lung cancer
- Create 1 Numpy array per patient scan and 1 per segmentation mask
- Find range of voxels in which the segmentation exists for each patient
- Create a numpy array with a subvolume of the images

Processing-Based Segmentation Function

- Function segmentations vs ground truth ones
- How can we improve it?

Segmented Tumor - Patient case_0

Image Feature Extraction & Classification

- 3 Histogram-based radiomic features
- MAD is a useful feature to classify between benign and malignant lesions

$$E = \sum_{i=1}^{N} (V_i)^2 \quad MAD = \frac{1}{N} \sum_{i=1}^{N} |V_i - \bar{V}|$$

Patient	MAD	Intensity	Uniformity
0	189.50	27673732	0.0001
1	106.04	57513796	0.0001
10	274.03	6673689	0

$$U = \sum_{i=1}^{M} p_i^2$$

where p is the normalised histogram, i.e.

$$p_i = P(i)/N$$

being $P(i)$ the number of entries in the ith bin of the histogram.

Thank You for Listening :)

ARE THERE ANY QUESTIONS?