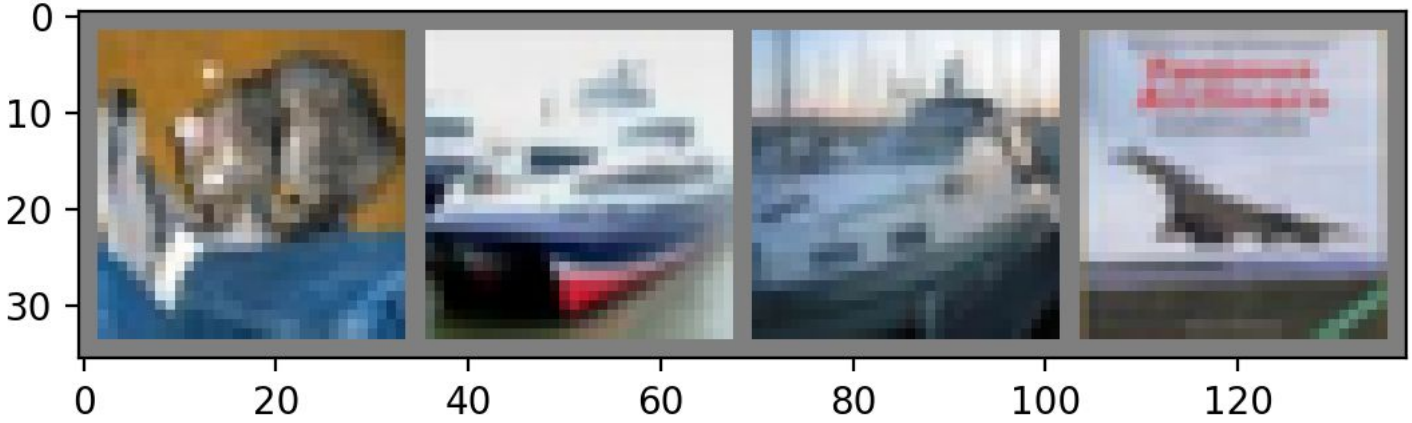
Tomographic Medical Image Reconstruction Using Deep Learning - Milestone 1

Asher Burrell, Christopher Hinton, Ty Mercer

What we did

Task	Completion	Asher	Chris	Ty	Todo
Learn about medical imaging	100%	50%	0%	50%	None
Learn about project tech	70%	30%	10%	30%	Learn Open-GATE simulation
Create machine learning demo	80%	0%	80%	0%	We have a demo, but still need to learn some PyTorch syntax
Set aside validation data	100%	0%	100%	0%	None
Make requirement document	100%	0%	0%	100%	None
Make design document	100%	100%	0%	0%	None
Make test document	100%	0%	100%	0%	None

CNN Demo




```
[2,  6000] loss: 1.349
[2,  8000] loss: 1.337
[2, 10000] loss: 1.282
[2, 12000] loss: 1.296
Finished Training
GroundTruth:  cat   ship  ship  plane
Predicted:  cat   ship  car   ship
```

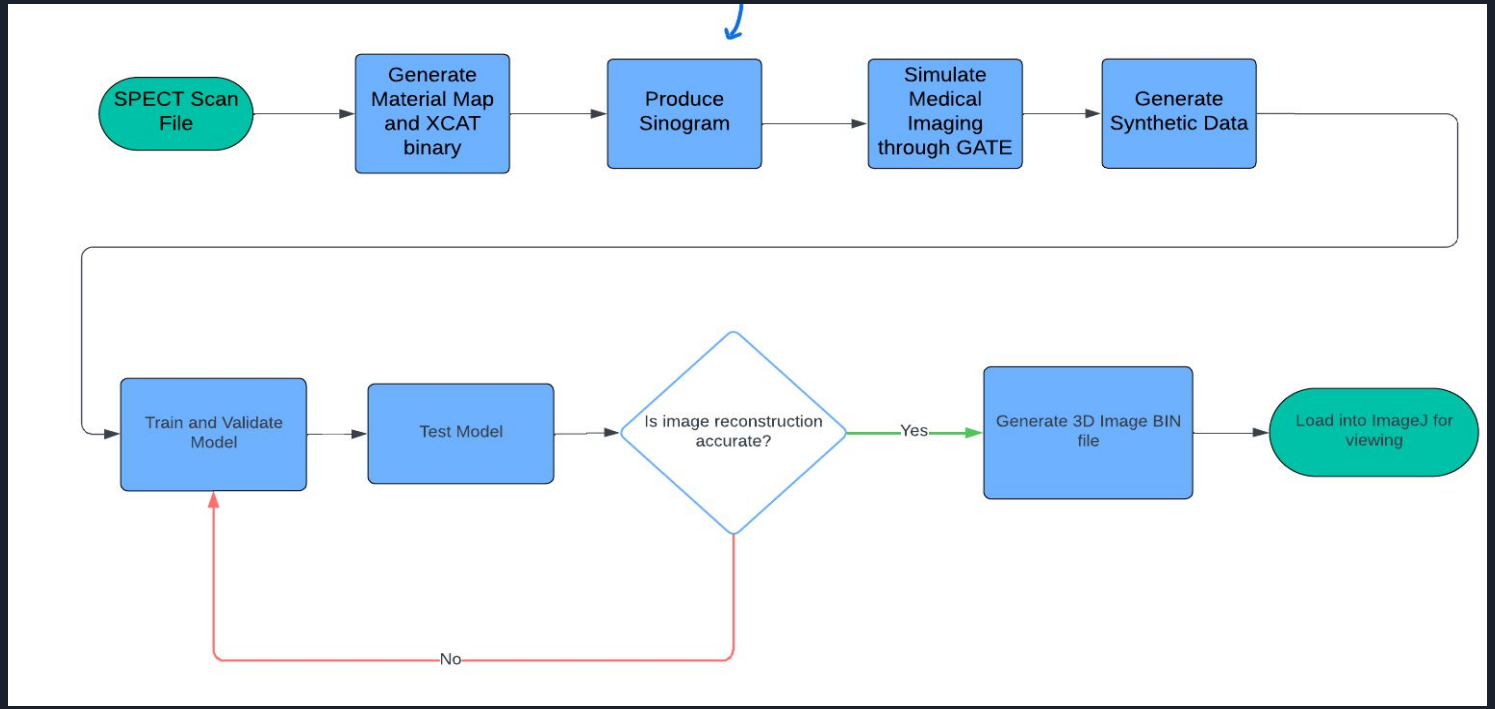
CNN Demo

```
Accuracy of the network on the 10000 test images: 53 %
Accuracy for class: plane is 50.1 %
Accuracy for class: car   is 72.7 %
Accuracy for class: bird  is 55.9 %
Accuracy for class: cat   is 23.5 %
Accuracy for class: deer  is 25.3 %
Accuracy for class: dog   is 46.2 %
Accuracy for class: frog  is 69.6 %
Accuracy for class: horse is 60.7 %
Accuracy for class: ship  is 72.7 %
Accuracy for class: truck is 58.6 %
```

Validation Data

- Reconstructed about 160 medical images
- Can't show them in the slides for privacy reasons
- Rated each image in terms of quality
- Can trace images back to original sinograms

Design: System Pipeline



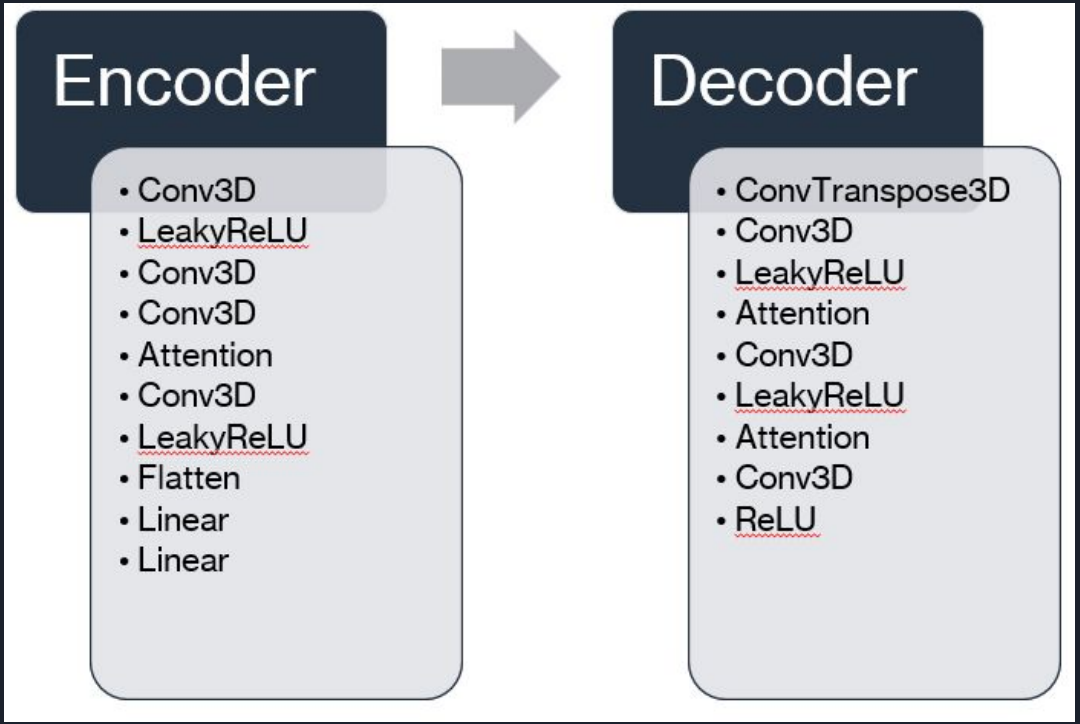
Neural Network Architecture Design

Our model consists of a Convolutional Encoder-Decoder structure. It includes:

- CNNs to process the input image data.
- Attention layers to extract the most import features from the images.
- An encoder-decoder structure to transform the original SPECT image into another form (a 3D model).

Neural Network Architecture Design

Note: This is the baseline model that is in the lab, but since it is not giving the desired results, it will be completely revised. However, the encoder-decoder structure with CNNs and attention layers will remain the same.



Main System Features/Requirements

- Fill in XCAT phantoms with simulated tracer concentrations
- Create synthetic sinogram data by simulating the medical imaging process on these phantoms in Open-GATE
- Train a CNN on this data and use it to reconstruct real SPECT data
    - Accuracy is technically a stretch goal as this is an experimental project, but we expect to see good results

Task Matrix - Milestone 2

Task	Asher	Chris	Ty
Learn how to use Open-GATE simulation	33%	33%	33%
Learn Pytorch syntax (secondary)	33%	33%	33%
Generate synthetic SPECT data	33%	33%	33%

Questions?