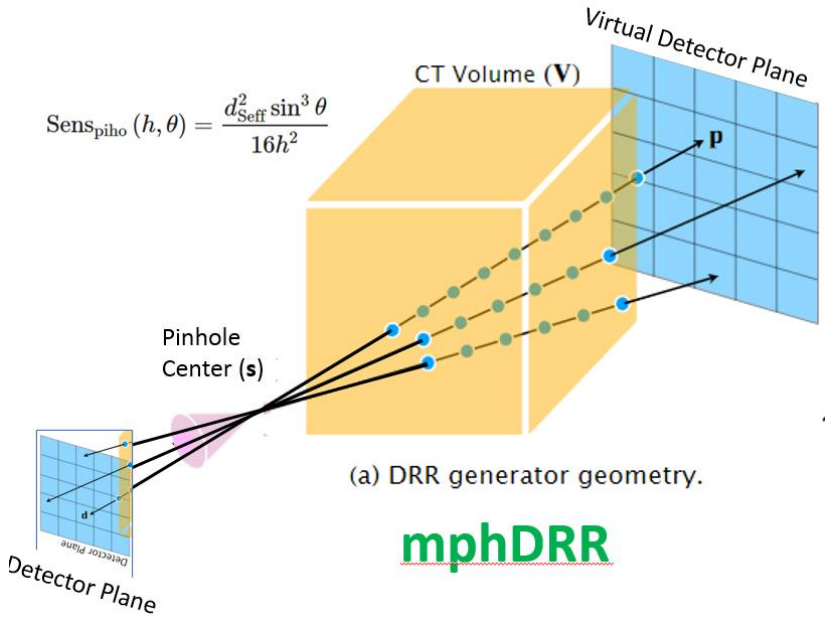
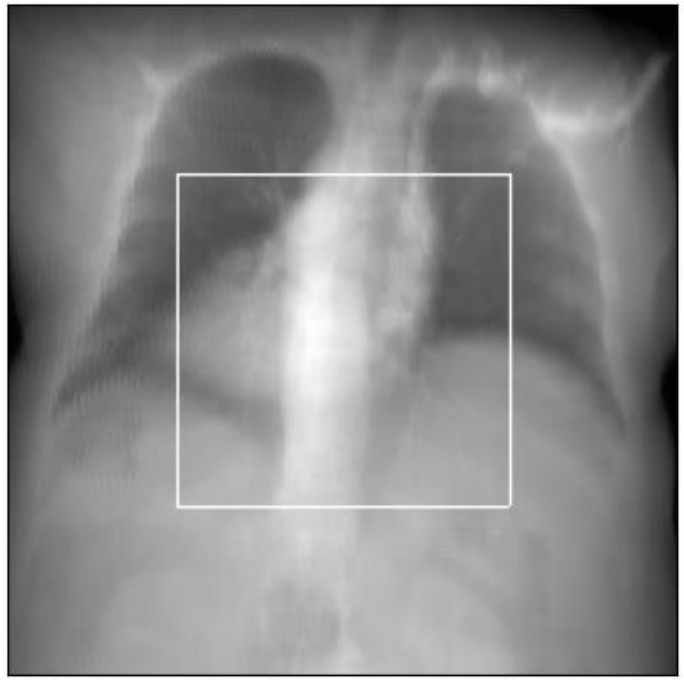
mphDRR in PyTorch: Multiple Pinhole SPECT/'CT' Digitally Reconstructed Radiographs

Bill Worstell

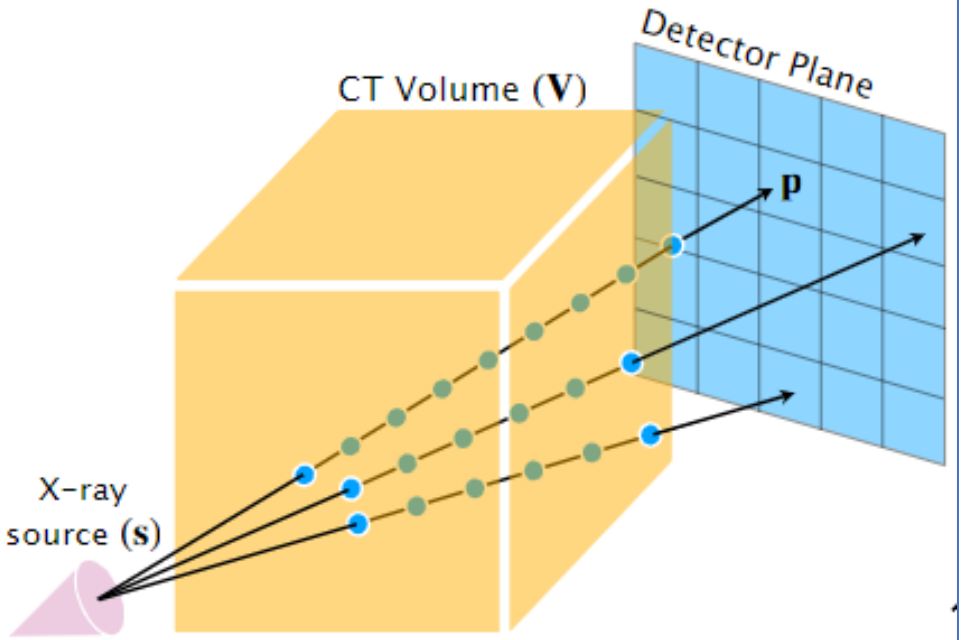
PicoRad->MGH

1/2/2024





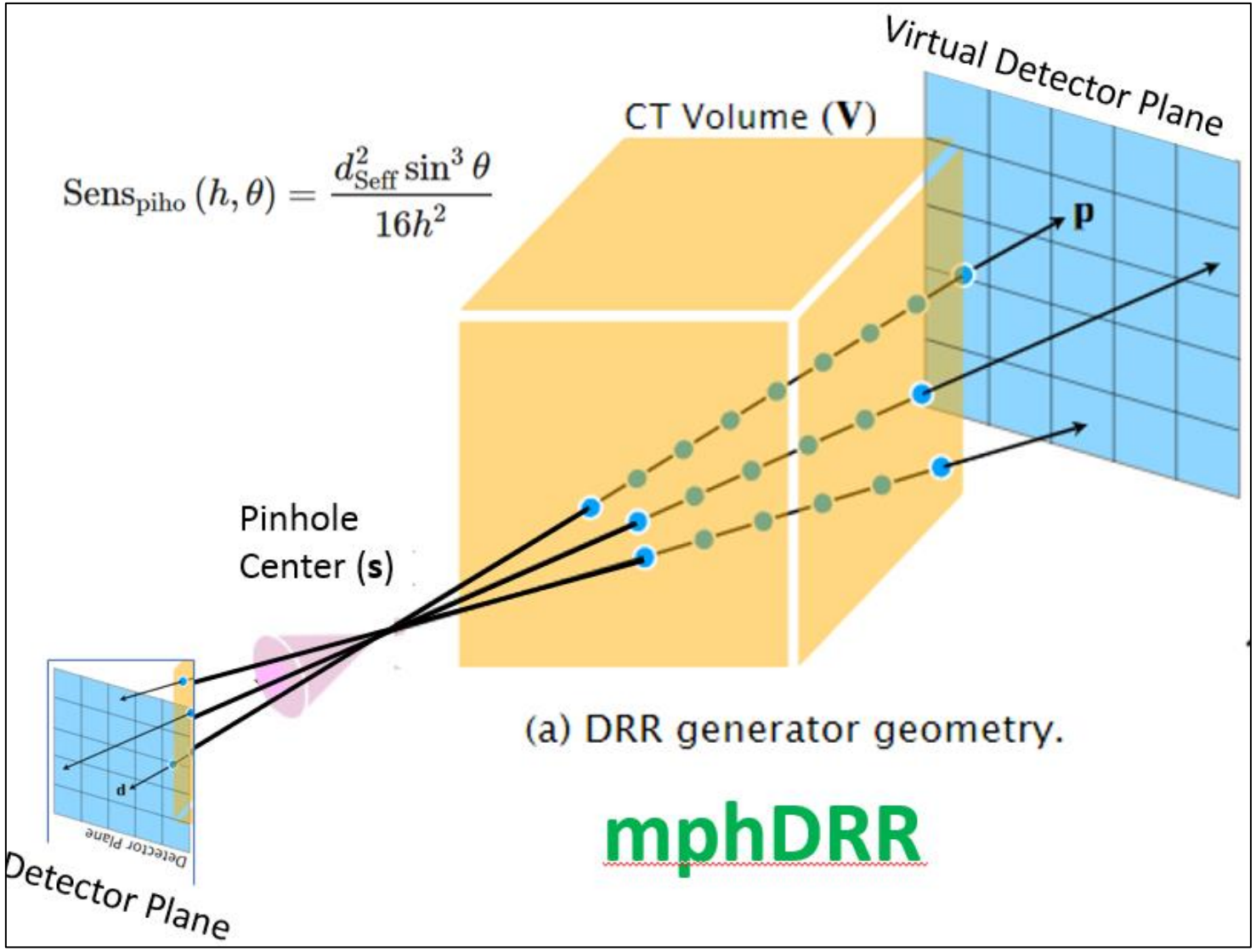
Xray DRR



(a) DRR generator geometry.

Gopalakrishnan, V. and Golland, P., 2022, September. Fast auto-differentiable digitally reconstructed radiographs for solving inverse problems in intraoperative imaging. In *Workshop on Clinical Image-Based Procedures* (pp. 1-11). Cham: Springer Nature Switzerland.

Fig 1a: X-ray DRR system geometry



(a) DRR generator geometry.

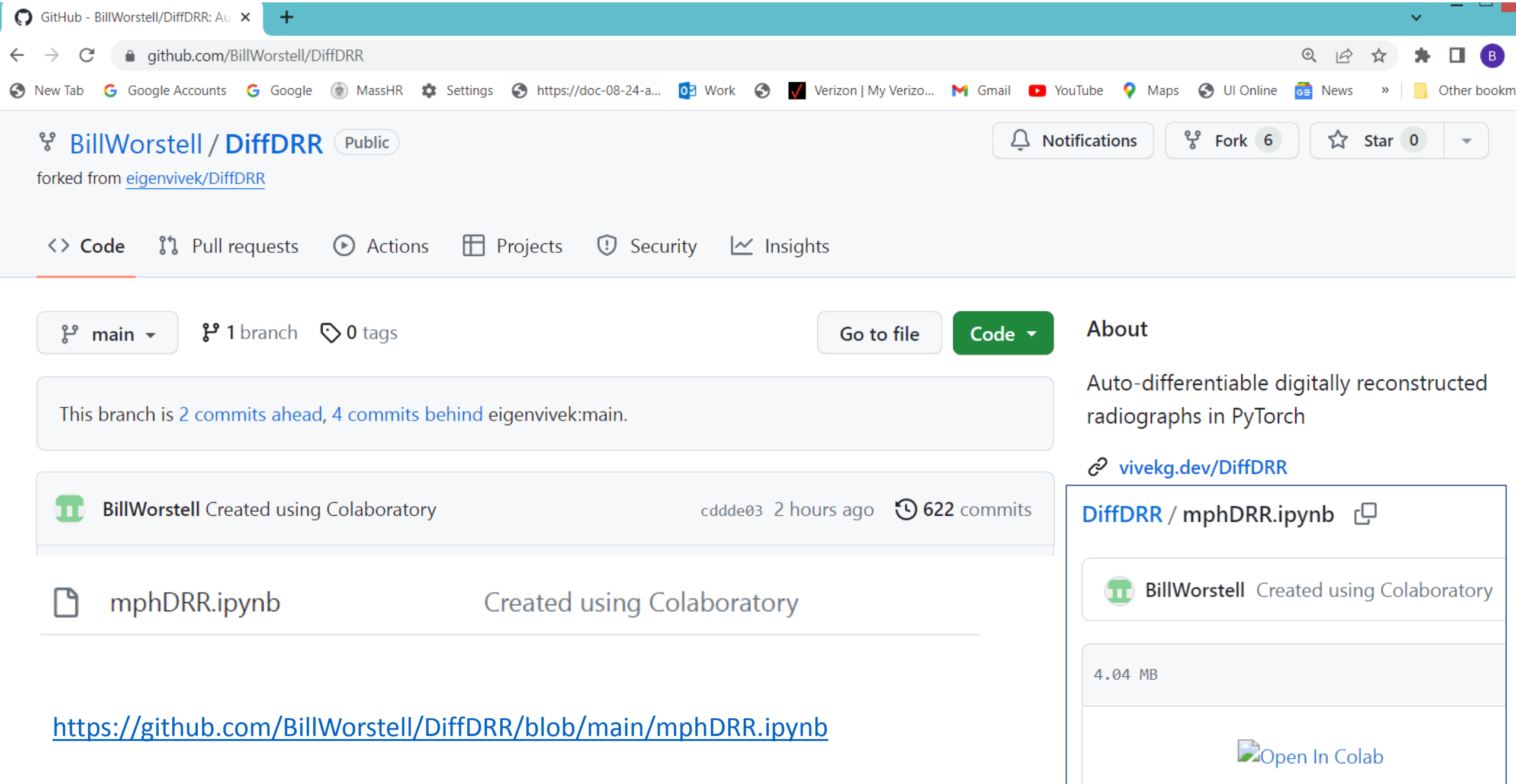
mphDRR

Fig 1b: Multiple Pinhole SPECT DRR system geometry

```
# Initialize the DRR module for generating synthetic X-rays
device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
drr = DRR(
    volume,  # The CT volume as a numpy array
    spacing,  # Voxel dimensions of the CT
    sdr=300.0,  # Source-to-detector radius (half of the source-to-detector distance)
    height=200,  # Height of the DRR (if width is not seperately provided, the generated image is square)
    delx=4.0,  # Pixel spacing (in mm)
).to(device)
# Set the camera pose with rotations (yaw, pitch, roll) and translations (x, y, z)
rotations = torch.tensor([[torch.pi, 0.0, torch.pi / 2]], device=device)
translations = torch.tensor([[bx, by, bz]], device=device)
img = drr(rotations, translations, parameterization="euler_angles", convention="ZYX")
```

```
# Initialize the DRR module for generating synthetic X-rays
device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
drr = DRR(
    volume,  # The CT volume as a numpy array
    spacing,  # Voxel dimensions of the CT
    sdr=vsdr[imod],  # Source-to-virtual-detector radius (half of the source-to-virtual-detector distance)
    height=500,  # Height of the DRR (if width is not seperately provided, the generated image is square)
    delx=2.5,  # Pixel spacing (in mm)
).to(device)
# Set the camera pose with rotations (yaw, pitch, roll) and translations (x, y, z)
rotations = torch.tensor([alpha[imod],beta[imod],0.], device=device)
translations = torch.tensor([[bx, by, bz]], device=device)
img = drr(rotations, translations, parameterization="euler_angles", convention="ZYX")
```

https://github.com/BillWorstell/DiffDRR



https://github.com/BillWorstell/DiffDRR/blob/main/mphDRR.ipynb

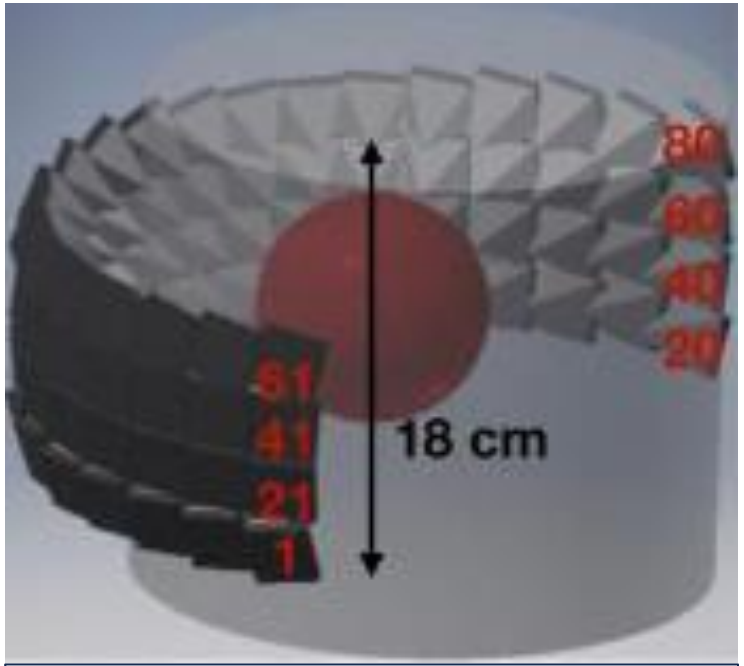


Fig 2a: DC-SPECT system geometry

Fig 2b: DC-SPECT system collimator assembly parts

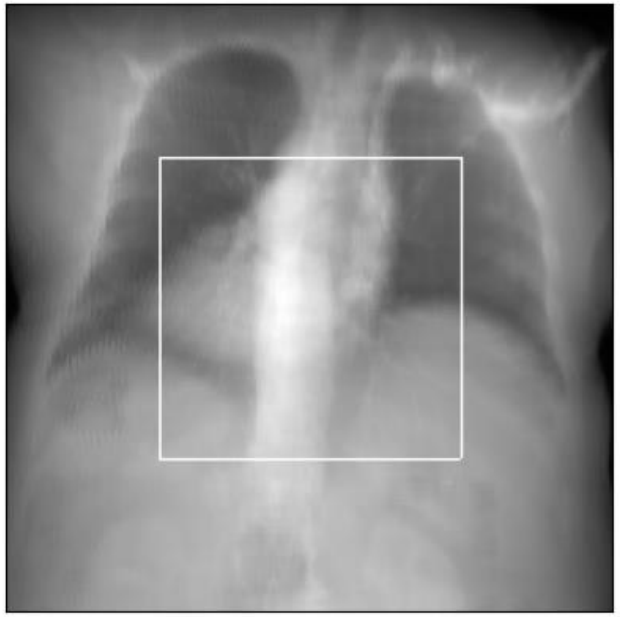


Fig 2c: DRR from example CT

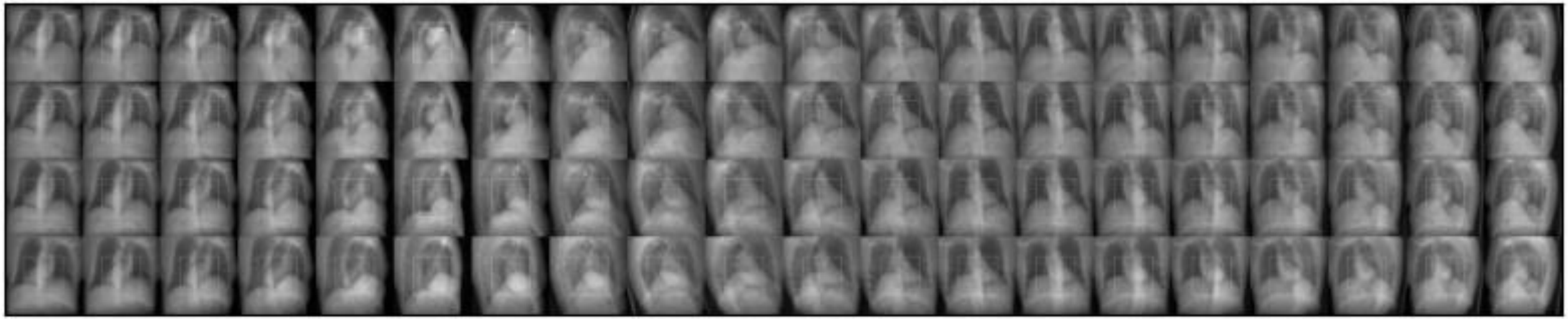
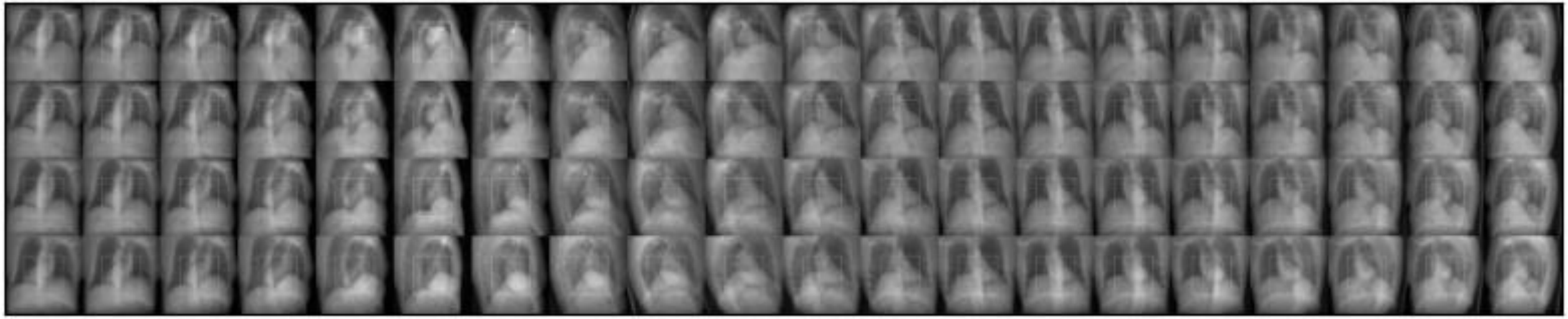


Fig 2d: Synthesized DC-SPECT system mpcDRRs corresponding to X-ray line integrals as in DRR

Wide FOV (2x) Synthesized X-ray DRRs for DC Cardiac SPECT Camera



Full FOV (1x) Synthesized X-ray DRRs for DC Cardiac SPECT Camera