Tutorial session: winter school 2017

Implementation of iterative CT reconstruction

Kim, Kyungsang

Gordon Center for Medical Imaging

Massachusetts General Hospital and Harvard Medical School

Harvard Medical School

Massachusetts General Hospital

Gordon Center for Medical Imaging

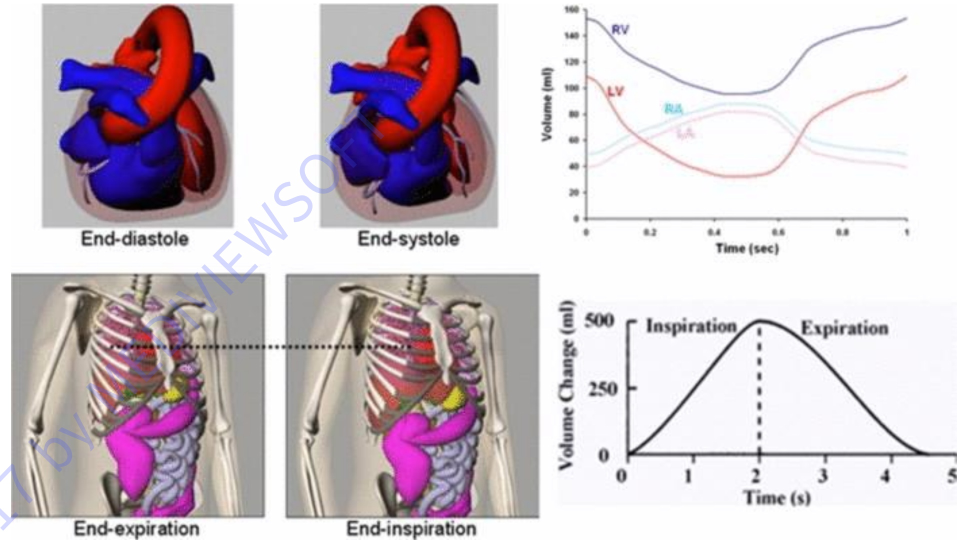
Contents

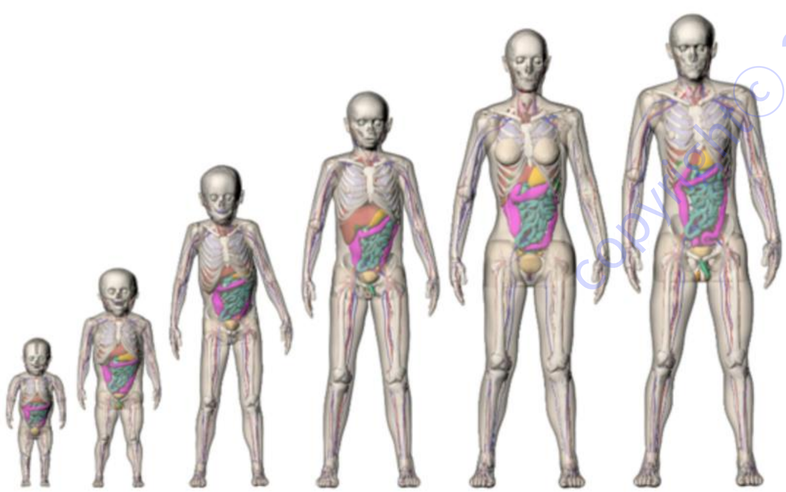
- **XCAT Phantom simulation**
  - *Geometry parameter setting*
  - *Sinogram generation with Poisson noise*
  - *Iterative reconstruction (SART, SQS)*
  - *Quadratic penalty for noise reduction*
- **Metal artifact correction (MAR)**
  - *Sinogram generation with metal artifact*
  - *Sinogram inpainting-based MAR*
- **Real patient data experiment**

Gordon Center for Medical Imaging

XCAT phantom

- Vector-based image generation
- Respiratory motion
- Cardiac motion
- Support various body conditions such as size, gender, etc.
- Used for CT and PET (3D and 4D)





Segars, William Paul, et al. "Realistic CT simulation using the 4D XCAT phantom." Medical physics 35.8 (2008): 3800-3808.

Gordon Center for Medical Imaging

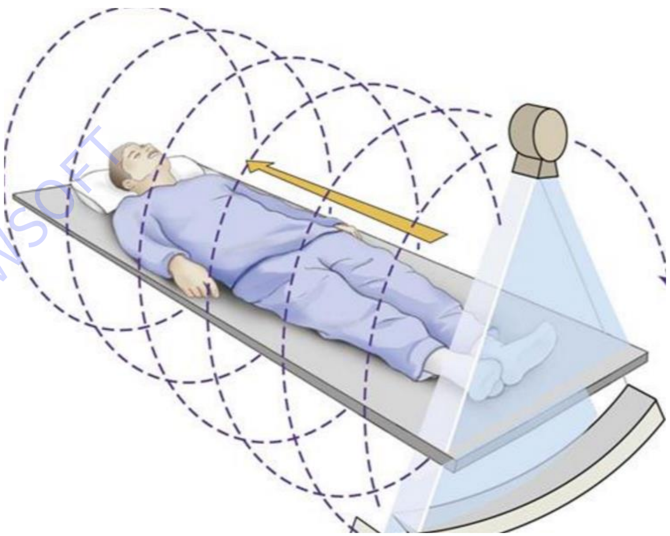
Geometry of scanner

- **Helical CT:** whole-body

↓

- **Slice rebinning for Helical CT**
  - **2D slice reconstruction**

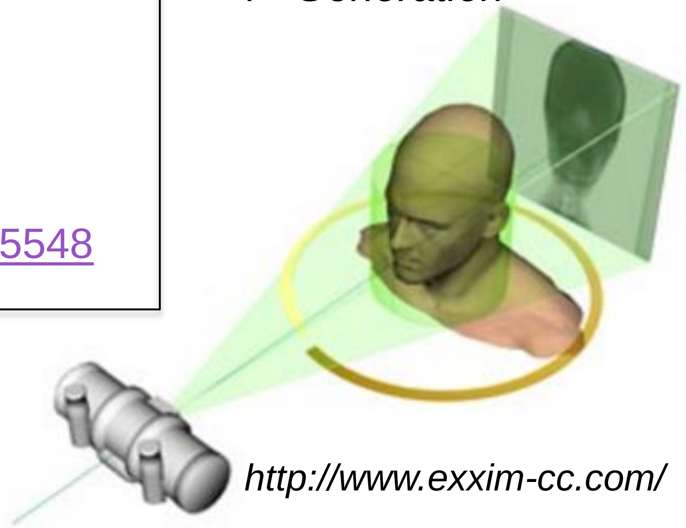
3rd Generation



- **Cone-beam CT with flat detector:**
  - Dental, small animal, chest, brain…

Free MATLAB Cone beam CT reconstruction code:
http://www.mathworks.com/matlabcentral/fileexchange/35548

4th Generation



http://www.exxim-cc.com/

Initialization

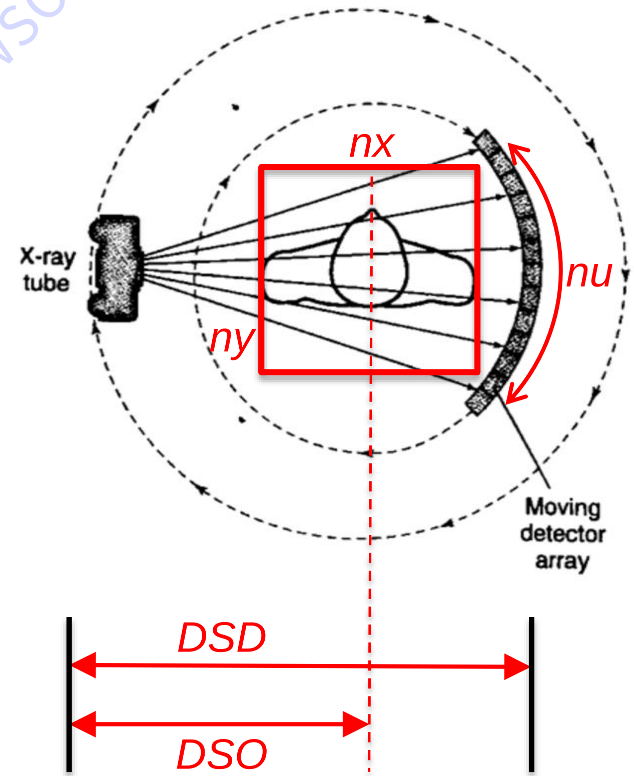
- Requirements: Matlab + C compiler (OpenMP)
- Two zip files are given
  - PhantomExample_WinterSchool2017.zip
  - PatientExample_WinterSchool2017.zip
- Unzip “PhantomExample_WinterSchool2017.zip”
- Open “MeasurementGen.m” and run (F5)
  - If you see error message, please run CompileWindows/Linux/Mac
  - C compiler is required for MEX compilation
- If success, now you are ready
- Provide “projector” and “backprojector”

Geometry parameter setting

- **Open “ParamSetting.m”**

```
%% Parameter setting %%

param.nx = 256; % number of pixels
param.ny = 256;

param.dx = 2; % mm
param.dy = 2;

param.sx = param.nx*param.dx; % mm (whole size)
param.sy = param.ny*param.dy; % mm

% X-ray source and detector setting
param.DSD = 1085.6;   %  Distance source to detector
param.DSO = 595;   %  X-ray source to object axis distance

%The detector panel pixels  (number of pixels)
param.nu = 736;

% Detector setting, cylindrical detector
param.du = 1.2858;
param.fan_angle = param.du/param.DSD*180/3.141592*param.nu; % mm

% angle setting
param.dir = 1;   % gantry rotating direction
param.nview = 540 ; % number of views in one rotation (360 deg)
param.dang = 360/param.nview;
param.deg = [0:param.nview-1]*param.dang;
param.deg = param.deg*param.dir;

% filter='ram-lak' % high pass for sintetic images
param.filter='hamming'; % high pass for sintetic images

param.da = param.fan_angle/param.nu/180*3.141592; % rad
param.off_a = 0; % rad
```

nview: # of projections



Gordon Center for Medical Imaging

Measurement generation

- **Open "MeasurementGen.m"**

```
load img_ref.mat
Slice = 1;
i0 = 8000;                                    ← "i0" is a blank scan value

img_groundtruth = img_ref(:,:,Slice);

sino = CTproj_fan(img_groundtruth,param);     ← projection to make a sinogram

sino_n = max(i0*exp(-sino),1);
sino_n = poissrnd(sino_n);                    ← "poissrnd" function applies Poisson noise
sino_n = -log(max(min(sino_n,i0),1)/i0);
```


$$y_e = i_0 e^{-\mu L}$$

$$y = -\log\left(\frac{y_e}{i_0}\right) = \mu L$$

Gordon Center for Medical Imaging

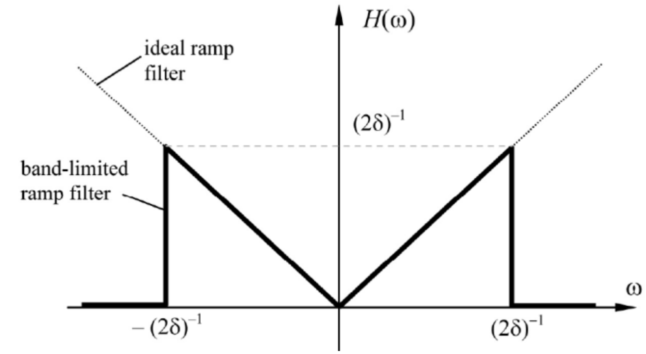
Filtered back-projection (FBP)

- **Two files: “filtering.m”, “CTbackproj_fan.m”**

```
param.filter='hann';

sino_filt = filtering(sino_n,param);

img = max(CTbackproj_fan(sino_filt,param),0);
```

$$f(x,y)=\boxed{\int_0^\pi d\theta\int_{-\infty}^{\infty}\boxed{P(\omega,\theta)\left|\omega\right|e^{j2\pi\omega t}}\,d\omega}$$

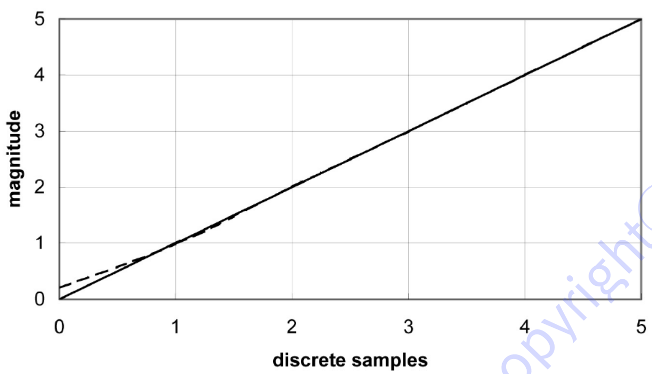
(2) (1)

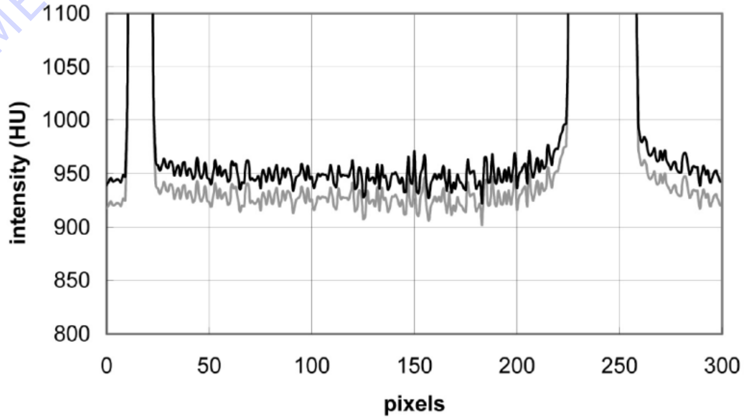


Filtered back-projection (FBP)

- **Two files: “filtering.m”, “CTbackproj_fan.m”**

```
param.filter='hann';

sino_filt = filtering(sino_n,param);

img = max(CTbackproj_fan(sino_filt,param),0);
```





Inaccurate ramp filter makes DC offsets

→ Digitalized ramp kernel is required

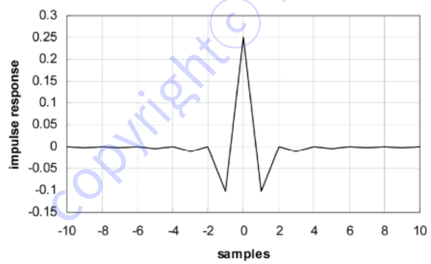
Filtered back-projection (FBP)

- **Two files: “filtering.m”, “CTbackproj_fan.m”**

```
param.filter='hann';

sino_filt = filtering(sino_n,param);

img = max(CTbackproj_fan(sino_filt,param),0);
```

❖ Solution

$$h(n\delta)=\begin{cases}\dfrac{1}{4\delta^2}, & n=0,\\ 0, & n=\text{even},\\ -\dfrac{1}{(n\pi\delta)^2}, & n=\text{odd}.\end{cases}$$



FFT ➡ Resolution-based Ramp kernel

Additional low-pass filtering can be applied to reduce noise

Gordon Center for Medical Imaging

Iterative reconstruction

- **We define …**
  - **Projection: A**
  - **Back-projection: A^T**

Ax : y = CTproj_fan(x,param);

Input: image, Output: sinogram

A^Ty : x = CTbackproj_fan(y,param);

Input: sinogram, Output: image

SART

- **Simultaneous algebraic reconstruction technique**

```
%% SART

Norimg = CTbackproj_fan(CTproj_fan(ones(param.nx,param.ny,'single'),param),param);

for iter = 1:50

    tic;
    % diff back projection
    sino_diff = CTproj_fan(img,param) - sino_n;
    Diffimg = CTbackproj_fan(sino_diff,param);

    % update
    img = max(img-Diffimg./Norimg,0);
    img(isnan(img))=0;

    figure(11); imagesc(max(img,0),[0 0.03]); axis off; axis equal; colormap gray; colorbar;   title(['iter - ',num2str(iter)]);
    pause(0.1);
    exetime = toc;

    disp([num2str(iter),' - iteration done.. ','/ Exe. time (sec) : ', num2str(exetime)]);
end
```

SART

- **Simultaneous algebraic reconstruction technique**

$$x^{n+1} = x^n - \frac{A^T(Ax^n - y)}{\boxed{A^T A}}$$

(0)

$$A^T A = A^T A I,\ \ I = \text{image of ones}$$

We can pre-calculate the normalization term

```
Norimg = CTbackproj_fan(CTproj_fan(ones(param.nx,param.ny,'single'),param),param);
```

$$y = -\log(y_e / i_0)$$

Gordon Center for Medical Imaging

SART

- **Simultaneous algebraic reconstruction technique**

$$x^{n+1} = x^n - \frac{\overset{(2)}{\boxed{A^T(\overset{(1)}{\boxed{Ax^n - y}})}}}{A^T A}$$

(1) sino_diff = CTproj_fan(img,param) - sino_n;

(2) Diffimg = CTbackproj_fan(sino_diff,param);

SART

- **Simultaneous algebraic reconstruction technique**

(4)

$$x^{n+1} = \boxed{x^n - \boxed{\frac{A^T(Ax^n - y)}{A^T A}}}$$

(3)

```
% update
img = max(img-Diffimg./Norimg,0);
```

(4) (3)

SQS *(separable quadratic surrogate)*

Precompute and store: $d_j^* = \sum_{i=1}^{N} a_{ij}\gamma_i (y_i - r_i)^2 / y_i$, where $\gamma_i = \sum_j a_{ij}$

for each subset Compute: $\hat{l}_i, \dot{h}_i$ as in (19) in table 1

Update:

$$\mu_j := \left[\mu_j - \frac{M \sum_{i \in S_m} a_{ij}\dot{h}_i}{d_j^*}\right]_+ .$$

$$\boxed{\hat{l}_i = \sum_{j=1}^{p} a_{ij}\hat{\mu}_j \qquad \dot{h}_i = \left(\frac{y_i}{b_i\,\mathrm{e}^{-\hat{l}_i} + r_i} - 1\right) b_i\,\mathrm{e}^{-\hat{l}_i}}$$

end

Scatter and noise

$$x^{n+1} = x^n - \frac{A^T\left(\frac{y_e}{i_0 e^{-Ax^n} + \boxed{r}} - 1\right) i_0 e^{-Ax^n}}{A^T \frac{y_e - r}{y_e} A}$$

Erdogan, Hakan, and Jeffrey A. Fessler. "Ordered subsets algorithms for transmission tomography." *Physics in medicine and biology* 44.11 (1999): 2835.

Gordon Center for Medical Imaging

SQS *(separable quadratic surrogate)*

$$x^{n+1} = x^n - \frac{A^T\left(\dfrac{y_e}{i_0 e^{-Ax^n} + r} - 1\right) i_0 e^{-Ax^n}}{A^T \dfrac{y_e - r}{y_e} A}$$

Pre-corrected
Scatter and noise

$$\approx x^n - \frac{A^T\left(\dfrac{y_e}{i_0 e^{-Ax^n}} - 1\right) i_0 e^{-Ax^n}}{A^T y_e A}$$

MLTR: *maximum likelihood transmission*

Erdogan, Hakan, and Jeffrey A. Fessler. "Ordered subsets algorithms for transmission tomography." *Physics in medicine and biology* 44.11 (1999): 2835.

MGH 1811
Gordon Center for Medical Imaging

SQS (separable quadratic surrogate)

$$x^{n+1} = x^n - \frac{A^T \boxed{\left(\frac{y_e}{i_0 e^{-\boxed{Ax^n}}} - 1\right) i_0 e^{-\boxed{Ax^n}}}}{\boxed{A^T y_e A}}$$

$\boxed{A^T y_e A}$: *Pre-calculation*

```
Norimg = CTbackproj_fan(sino_exp.*CTproj_fan(ones(param.nx,param.ny,'single'),param),param);
```

$\boxed{Ax}$: *Projection*

$\boxed{A^T(\cdot)}$: *Backprojection of residual*

Erdogan, Hakan, and Jeffrey A. Fessler. "Ordered subsets algorithms for transmission tomography." *Physics in medicine and biology* 44.11 (1999): 2835.

SQS *(separable quadratic surrogate)*

```
%% SQS
sino_exp = i0*exp(-sino_n);
Norimg = CTbackproj_fan(sino_exp.*CTproj_fan(ones(param.nx,param.ny,'single'),param),param);

for iter = 1:50

    tic;
    % diff back projection
    sino_tmp = i0*exp(-CTproj_fan(img,param) );
    sino_diff = (sino_exp./sino_tmp - 1) .* sino_tmp;
    sino_diff(isnan(sino_diff)) = 0;
    sino_diff(isinf(sino_diff)) = 0;

    Diffimg = CTbackproj_fan(sino_diff,param);

    % update
    img = max(img-Diffimg./Norimg,0);
    img(isnan(img))=0;

    figure(11); imagesc(max(img,0),[0 0.03]); axis off; axis equal; colormap gray; colorbar;   title(['iter - ',num2str(iter)]);
    pause(0.1);
    exetime = toc;

    disp([num2str(iter),' - iteration done.. ','/ Exe. time (sec) : ', num2str(exetime)]);
end
```


Gordon Center for Medical Imaging

Quadratic penalty

$$R(x) = \frac{1}{2}\sum_{j}^{N_v}\sum_{j'\in\Omega}\rho_{jj'}\left(x_j - x_{j'}\right)^2$$

$$\dot{R}(x_j) = \sum_{j'\in\Omega}\rho_{jj'}\left(x_j - x_{j'}\right)$$

$$\ddot{R}(x_j) = \sum_{j'\in\Omega}\rho_{jj'} = 1$$

- quadpenalty(img,1) = $\dot{R}(x_j)$

Iterative recon + quadratic penalty

- SART + quadratic penalty

$$x^{n+1} = x^n - \frac{A^T(Ax^n - y) + \beta \dot{R}(x^n)}{A^T A + \beta \ddot{R}(x^n)}$$

- SQS + quadratic penalty

$$x^{n+1} = x^n - \frac{A^T \left\{ \left(\frac{y_e}{i_0 e^{-Ax^n}} - 1 \right) i_0 e^{-Ax^n} \right\} + \beta \dot{R}(x^n)}{A^T y_e A + \beta \ddot{R}(x^n)}$$

Gordon Center for Medical Imaging

SART with Quadratic penalty

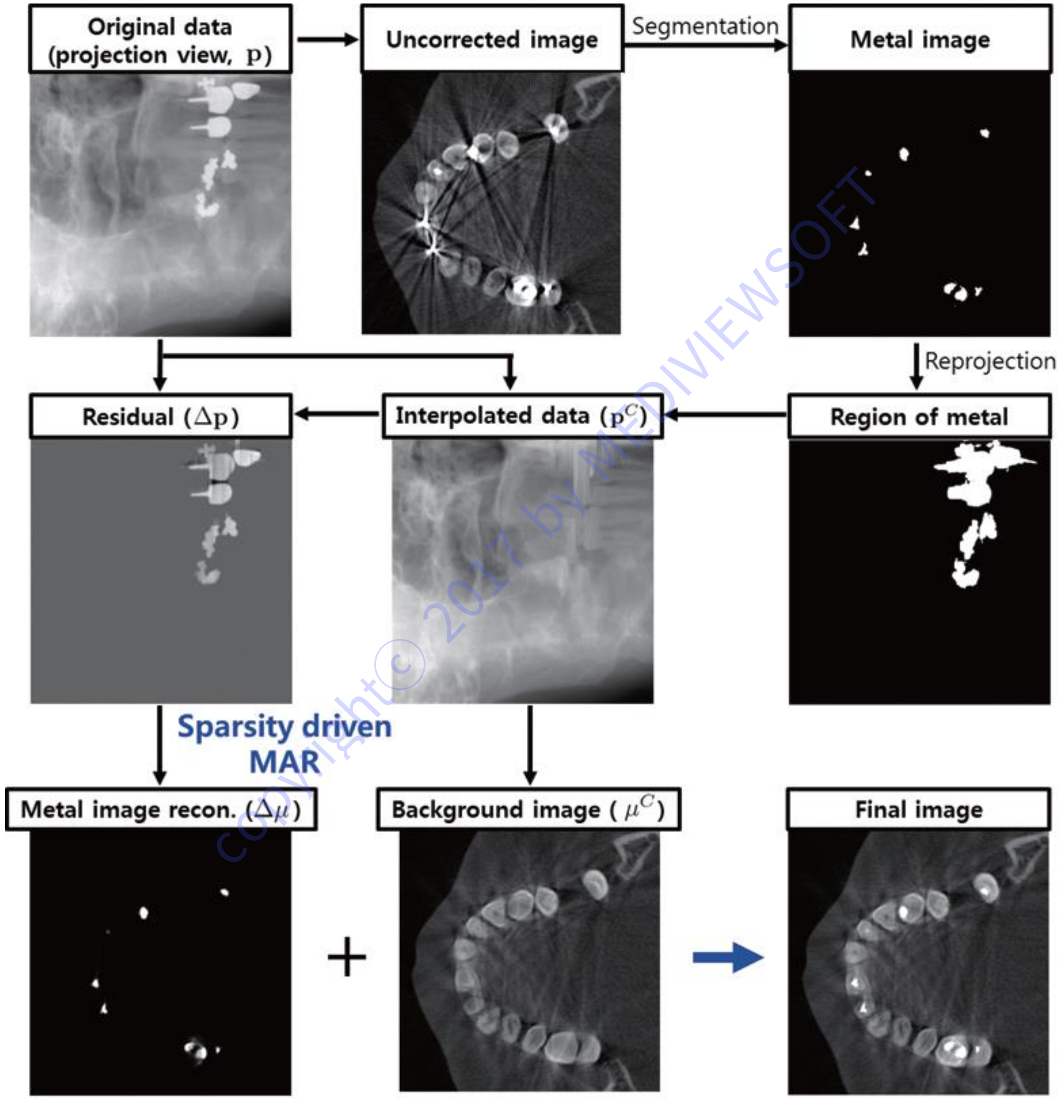
```
Norimg = CTbackproj_fan(CTproj_fan(ones(param.nx,param.ny,'single'),param),param);
Beta = 0.03.*Norimg;

for iter = 1:50

    tic;
    % diff back projection
    sino_diff = CTproj_fan(img,param) - sino_n;
    Diffimg = CTbackproj_fan(sino_diff,param);

    % Quadratic penalty
    img_quad = quadpenalty(img,1);

    % update
    img = max(img-(Diffimg+Beta.*img_quad)./(Norimg+Beta),0);
    img(isnan(img))=0;

    figure(11); imagesc(max(img,0),[0 0.03]); axis off; axis equal; colormap gray; colorbar;   title(['iter - ',num2str(iter)]);
    pause(0.1);
    exetime = toc;

    disp([num2str(iter),' - iteration done.. ','/ Exe. time (sec) : ', num2str(exetime)]);
end
```

SQS with Quadratic penalty

```
%% SQS
sino_exp = i0*exp(-sino_n);
Norimg = CTbackproj_fan(sino_exp.*CTproj_fan(ones(param.nx,param.ny,'single'),param),param);
Beta = 0.03.*Norimg;

for iter = 1:50

    tic;
    % diff back projection
    sino_tmp = i0*exp(-CTproj_fan(img,param) );
    sino_diff = (sino_exp./sino_tmp - 1) .* sino_tmp;
    sino_diff(isnan(sino_diff)) = 0;
    sino_diff(isinf(sino_diff)) = 0;

    Diffimg = CTbackproj_fan(sino_diff,param);

    % Quadratic penalty
    img_quad = quadpenalty(img,1);

    % update
    img = max(img-(Diffimg+Beta.*img_quad)./(Norimg+Beta),0);
    img(isnan(img))=0;

    figure(11); imagesc(max(img,0),[0 0.03]); axis off; axis equal; colormap gray; colorbar;   title(['iter - ',num2str(iter)]);
    pause(0.1);
    exetime = toc;

    disp([num2str(iter),' - iteration done.. ','/ Exe. time (sec) : ', num2str(exetime)]);
end
```


don
ter for
Medical
Imaging

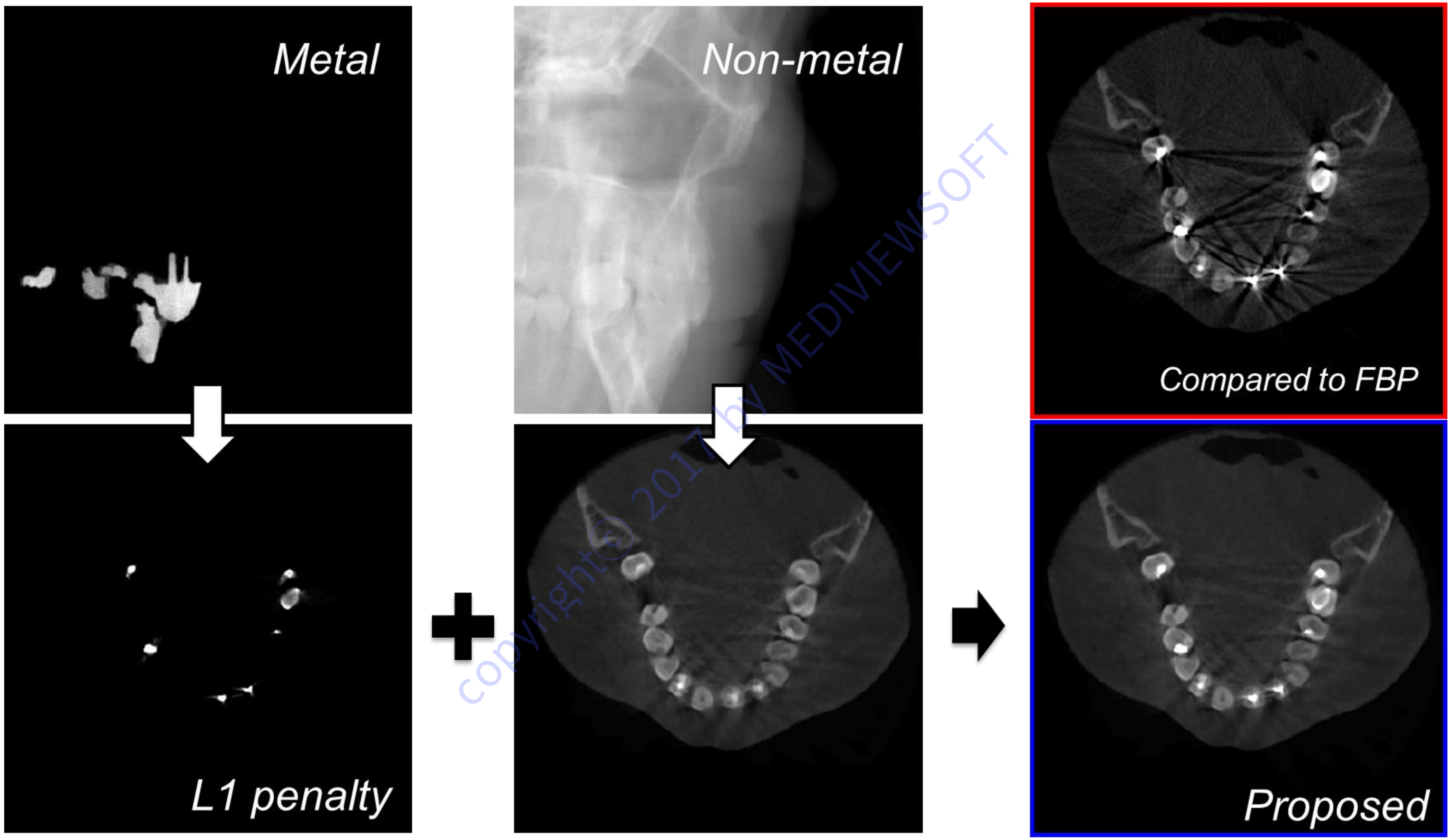
Metal artifact reduction flow chart



J. Choi, K. Kim, M. W. Kim, W. Seong and J. C. Ye, Sparsity Driven Metal Part Reconstruction for Artifact Removal in Dental CT, Journal of X-ray Science and Technology, vol. 19, pp. 457-475, October 2011.

Gordon Center for Medical Imaging

Metal artifact reduction flow chart



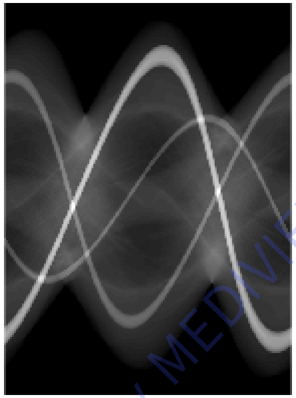
K. Kim, J. C. Ye, G. E. Fakhri and Q. Li, Metal artifact reduction using l1 and non-local penalties with iterative sinogram correction, The Third International Conference on Image Formation in X-Ray Computed Tomography, Salt Lake City, Utah, USA, June, 2014

Gordon Center for Medical Imaging

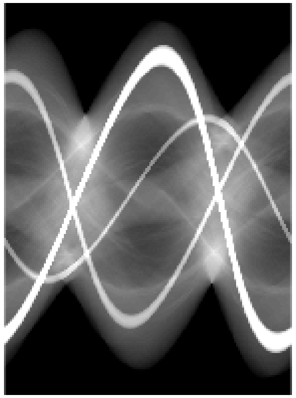
Metal artifact sinogram generation

```
nMetal = 3;
cx = [35, 105, 208];
cy = [129, 81, 119];
intensity = [ 0.3, 0.3, 0.2];
r= [7, 3, 5];
for id = 1:nMetal
    for ix=cx(id)-r(id):cx(id)+r(id)
        for iy = cy(id)-r(id):cy(id)+r(id)
            if sqrt((ix-cx(id))^2+(iy-cy(id))^2)<=r(id)
                img_groundtruth(ix,iy) = intensity(id)
            end
        end
    end
end


sino = CTproj_fan(img_groundtruth,param);

sino_n = max(i0*exp(-sino),1);
% sino_n = poissrnd(sino_n);
sino_n = -log(max(min(sino_n,i0),1)/i0);
```

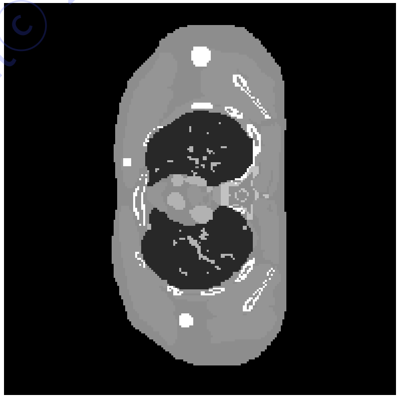
Noiseless



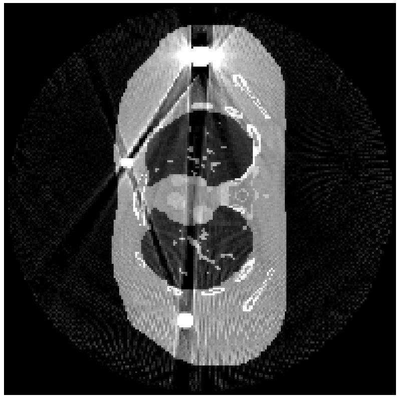
Metal Artifact



Noiseless

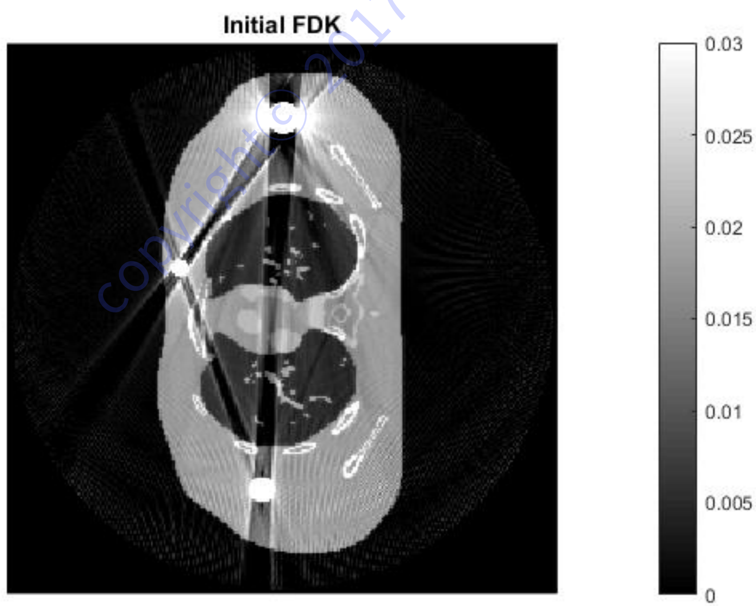


Metal Artifact



Initial FDK

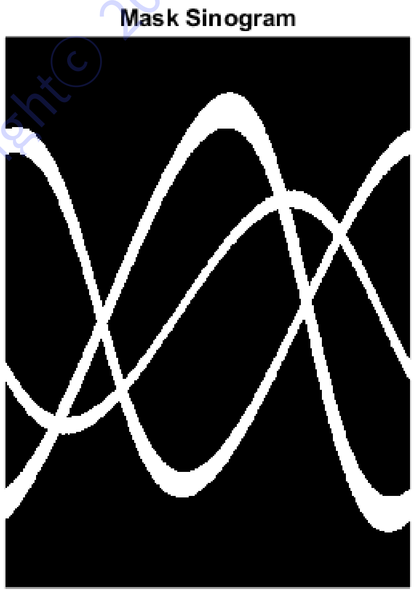
```
%% 1. filtered backprojection (initialization)

param.filter='hann';

tic;
sino_filt = filtering(sino_n,param);
imgFDK = max(CTbackproj_fan(sino_filt,param),0);
toc;

figure(1); imagesc(imgFDK,[ 0 0.03]); axis off; axis equal; colormap gray; colorbar; title('Initial FDK');
pause(0.1);
```



Gordon Center for Medical Imaging

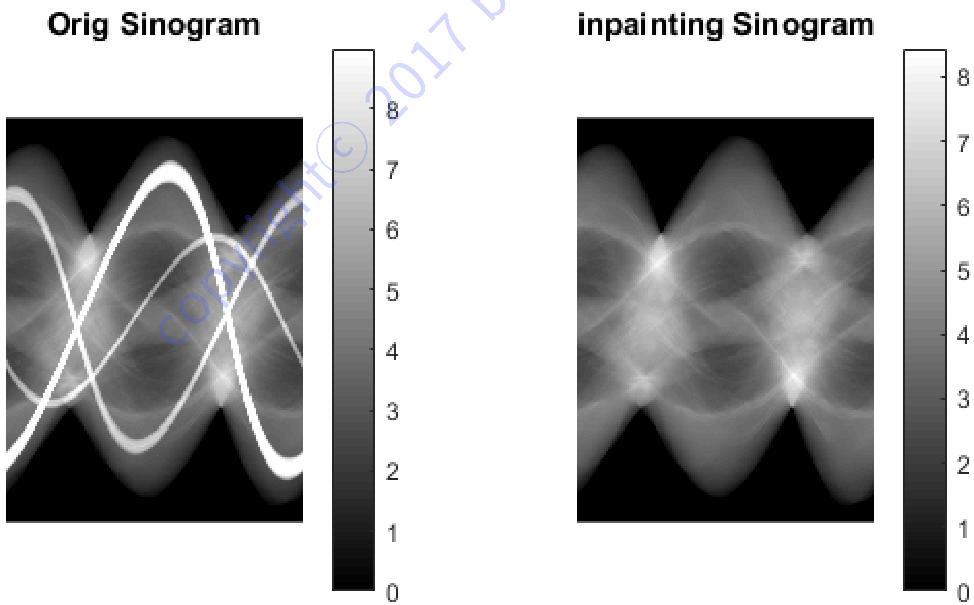
Mask of metallic part

```
%% 2. Metal object reprojection to make a mask

Metal_Threshold = 0.1;
img_metal = zeros(param.nx, param.ny, 'single');
img_metal(imgFDK>Metal_Threshold) = 1;

sino_mask = CTproj_fan(img_metal, param);
sino_mask(sino_mask>0) = 1;

figure(2); imagesc(sino_mask); axis off; axis equal; colormap gray; colorbar; title('Mask Sinogram');
pause(0.1);
```



Sinogram inpainting

```
%% 3. inpainting

sino_nonMetal = sino_n;
sino_nonMetal(sino_mask == 1) = NaN;

sino_nonMetal = inpaintn(sino_nonMetal,300);

figure(3);
subplot(1,2,1); imagesc(sino_n); axis off; axis equal; colormap gray; colorbar; title('Orig Sinogram');
subplot(1,2,2); imagesc(sino_nonMetal); axis off; axis equal; colormap gray; colorbar; title('inpainting Sinogram');
pause(0.1);
```

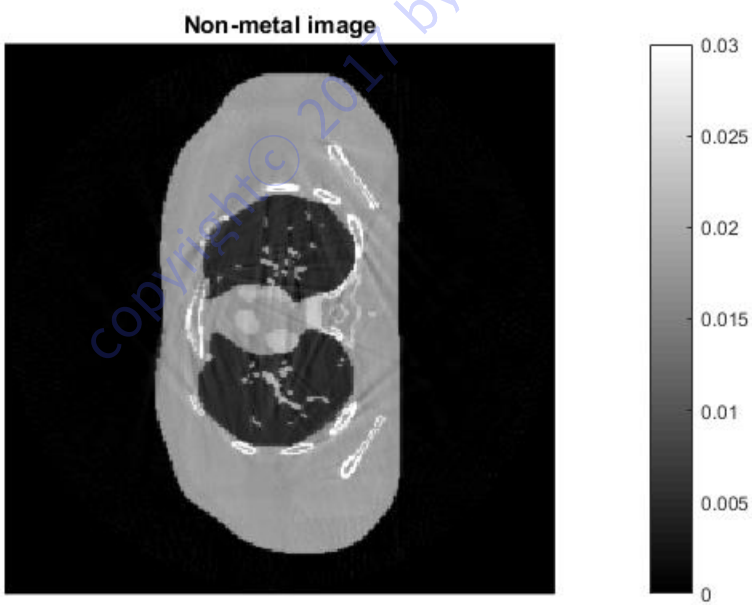


D. Garcia, “Robust smoothing of gridded data in one and higher dimensions with missing values,” Computational Statistics & Data Analysis, vol. 54, no. 4, pp. 1167–1178, 2010.

Gordon Center for Medical Imaging

Reconstruction of non-metal object

```
%% 4. reconstruction of non-metal part

param.filter='hann';

sino_filt = filtering(sino_nonMetal,param);
img_nonMetal = max(CTbackproj_fan(sino_filt,param),0);

figure(4); imagesc(img_nonMetal,[ 0 0.03]); axis off; axis equal; colormap gray; colorbar; title('Non-metal image');
pause(0.1);
```



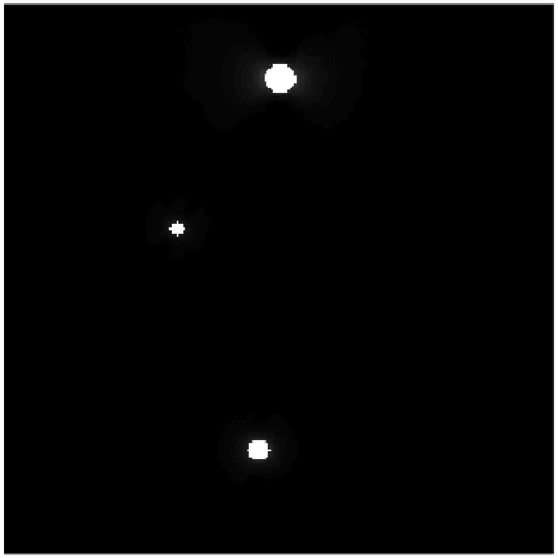
Gordon Center for Medical Imaging

L1-based metal object recon.

```
%% Metal reconstruction (FDK -> SART+L1)
sino_Metal = sino_n - CTproj_fan(img_nonMetal,param);
sino_filt = filtering(sino_Metal,param);
img_Metal = max(CTbackproj_fan(sino_filt,param),0);

Norimg = CTbackproj_fan(CTproj_fan(ones(param.nx,param.ny,'single'),param),param);
Beta = 0.5.*Norimg;
gamma = 0.01;

for iter = 1:50

    tic;
    % diff back projection
    sino_diff = CTproj_fan(img_Metal,param) - sino_Metal;
    Diffimg = CTbackproj_fan(sino_diff,param);

    % L1 penalty: soft thresholding
    img_L1 = img_Metal - max(img_Metal-gamma,0);

    % update
    img_Metal = max(img_Metal-(Diffimg+Beta.*img_L1)./(Norimg+Beta),0);
    img_Metal(isnan(img_Metal))=0;

    figure(5); imagesc(max(img_Metal,0),[0 0.03]); axis off; axis equal; colormap gray; colorbar;   title(['Metal image iter - ',num2str(iter)]);
    pause(0.01);
    exetime = toc;

    disp([num2str(iter),' - iteration done.. ','/ Exe. time (sec) : ', num2str(exetime)]);
end
```

$$\min_{x_m} \frac{1}{2}||y_m - \mathbf{A}x_m||_2^2 + \lambda||x_m||_1$$

Metal image iter - 50

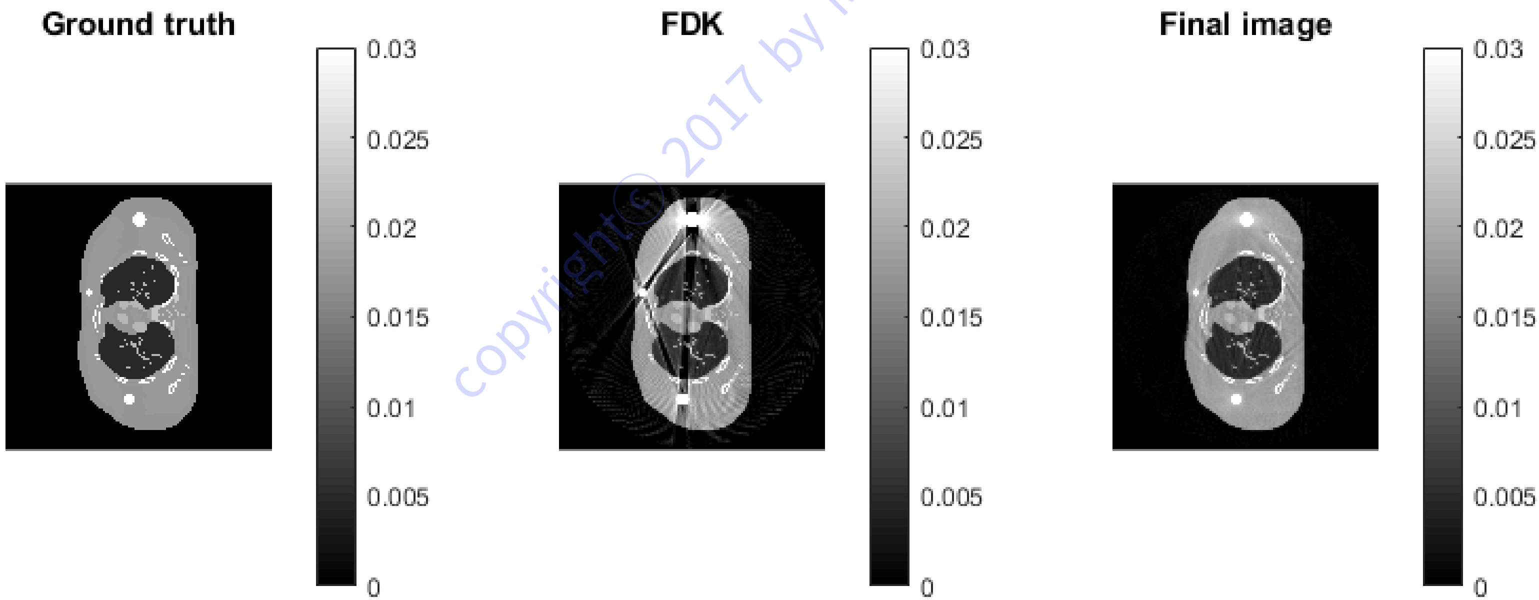


K. Kim, J. C. Ye, G. E. Fakhri and Q. Li, Metal artifact reduction using l1 and non-local penalties with iterative sinogram correction, The Third International Conference on Image Formation in X-Ray Computed Tomography, Salt Lake City, Utah, USA, June, 2014

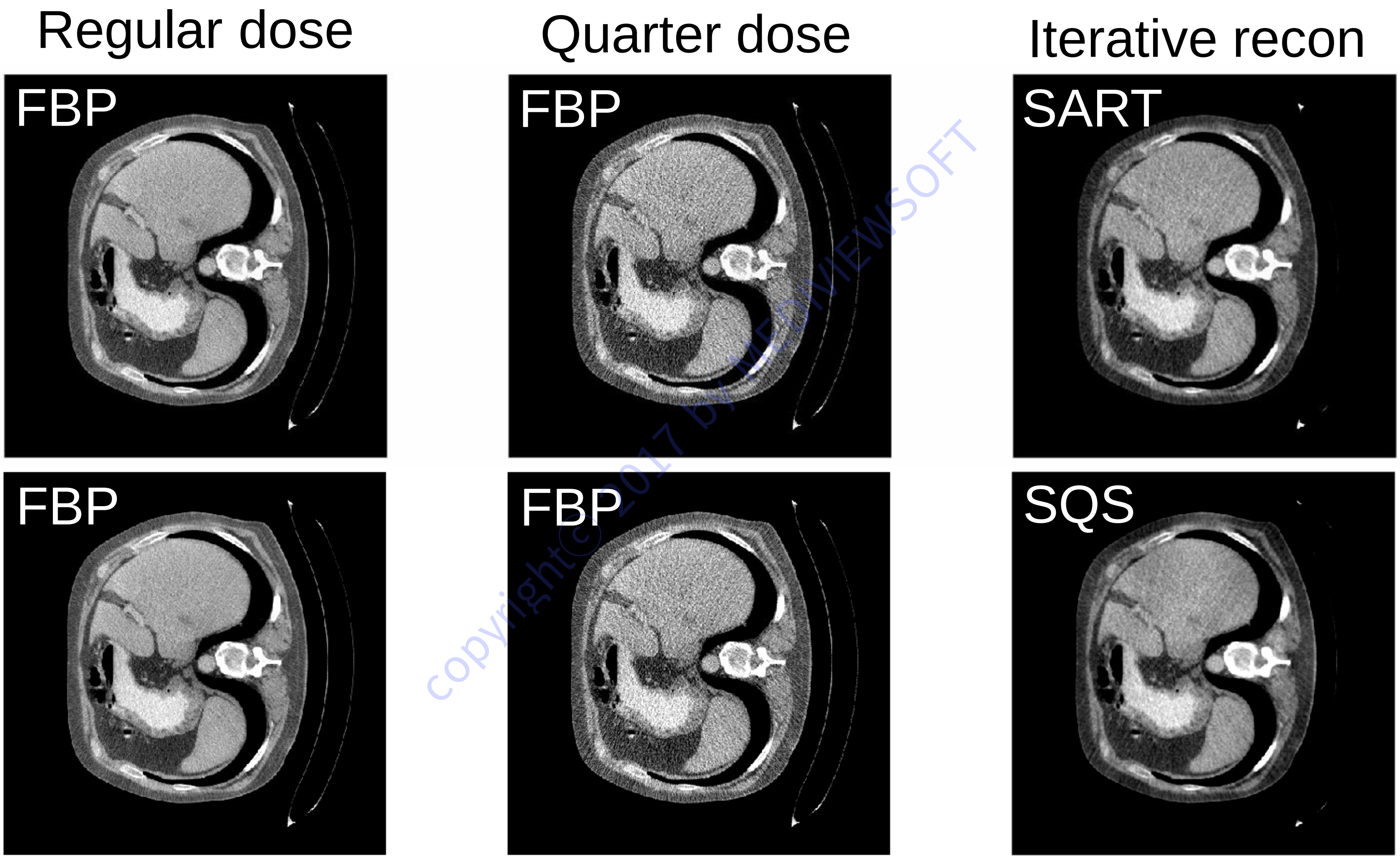
Gordon Center for Medical Imaging

Sum of non-metal and metal images

```
%% Final image

img = img_nonMetal+img_Metal;


figure(6);
subplot(1,3,1); imagesc(max(img_groundtruth,0),[0 0.03]); axis off; axis equal; colormap gray; colorbar;   title(['Ground truth']);
subplot(1,3,2); imagesc(max(imgFDK,0),[0 0.03]); axis off; axis equal; colormap gray; colorbar;   title(['FDK']);
subplot(1,3,3); imagesc(max(img,0),[0 0.03]); axis off; axis equal; colormap gray; colorbar;   title(['Final image']);
pause(0.1);
```

Patient data



Gordon Center for Medical Imaging

Thank you for your attention!