Homework 4 Report

Ayhan Okuyan
ayhan.okuyan[at]bilkent.edu.tr

December 16, 2020

Contents

Question 1

This question is about backprojection in CT imaging. Question states that a backprojection image $b_\theta(x, y)$ can be generated from a projection $g(l, \theta)$. We are given the projection $g(l, 15°) = e^{-l}$.

a. We write the backprojection image for any θ as follows.

$$b_\theta(x, y) = g(l, \theta)|_{l=x\cos\theta+y\sin\theta} = e^{-(x\cos\theta+y\sin\theta)} \tag{1}$$

Hence, we can write the backprojection for the specified degree as follows.

$$b_{15°}(x, y) = e^{-(x\cos(15°)+y\sin(15°))} \tag{2}$$

b. This part asks if we can infer $b_{165°}(x, y)$ from $g(l, 15°)$. No, we cannot since the projection $g(l, 165°)$ is irrelevant with $g(l, 15°)$.

c. This part asks if we can infer $b_{195°}(x, y)$ from $b_{15°}(x, y)$. On the contrary with the previous part, we can, since the 195° projection is the same with the 15° projection, on the opposite direction. Hence, we can write the projection as follows.

$$g(l, 195°) = g(-l, 15°) \tag{3}$$

Then we can write the backprojection as,

$$\begin{aligned} b_{195°}(x, y) &= g(l, 195°)|_{l=x\cos(195°)+y\sin(195°)} && (4)\\ &= g(-l, 15°)|_{l=x\cos(15°)+y\sin(15°)} && (5)\\ &= e^{x\cos(15°)+y\sin(15°)} && (6) \end{aligned}$$

Question 2

In this question, we are asked to find and sketch $g(l, \theta)$ for $\theta = \{0, \pi/2\}$. The object is provided in graphical form shown below and the linear attenuation coefficients $\mu_1 = 0.25cm^{-1}$, $\mu_2 = 0.05cm^{-1}$, $\mu_3 = 0.35cm^{-1}$ and $r = 15cm$. The object is presented below.

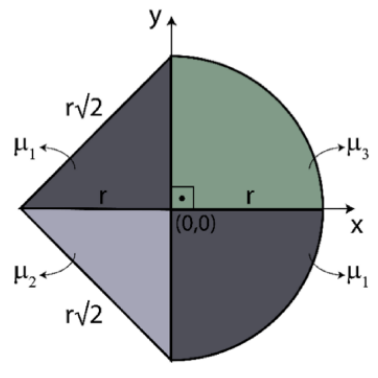

Figure 1: Object that will be CT Scanned

a. The first part asks about finding the projection of the object mentioned for $\theta = 45°$. The plot below, shows that there are three pieces that should be examined.

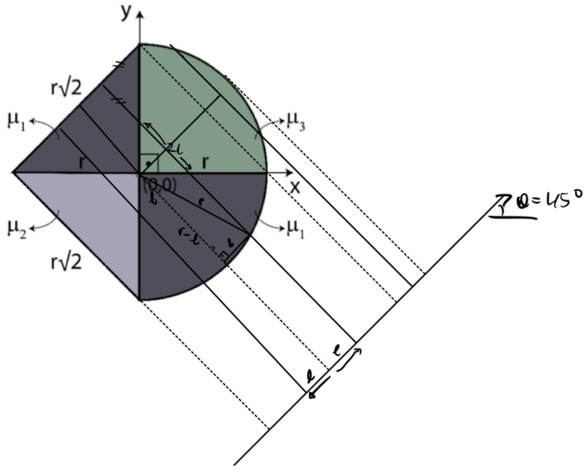

Figure 2: The projection plane, $g(l, 45°)$

Once we investigate each section, we can define the projection as a piecewise function in terms of r, μ_1, μ_2 and μ_3 as follows.

$$g_{r,\mu_1,\mu_2,\mu_3}(l, 45°) = \begin{cases} \mu_1(\frac{r\sqrt{2}}{2} + \sqrt{r^2+l^2} + 2l) - \mu_2(2l) & -\frac{r\sqrt{2}}{2} < l < 0 \\ \mu_1(\frac{r\sqrt{2}}{2} + \sqrt{r^2-l^2} - 2l) + \mu_3(2l) & 0 < l < \frac{r\sqrt{2}}{2} \\ \mu_3(2\sqrt{r^2-l^2}) & \frac{r\sqrt{2}}{2} < l < r \\ 0 & else \end{cases}$$

Replacing the parameters with the real values, we get,

$$g(l, 45°) = \begin{cases} 0.25(\frac{15\sqrt{2}}{2} + \sqrt{15^2+l^2} + 2l) - 0.05(2l) & -\frac{15\sqrt{2}}{2} < l < 0 \\ 0.25(\frac{15\sqrt{2}}{2} + \sqrt{15^2-l^2} - 2l) + 0.35(2l) & 0 < l < \frac{15\sqrt{2}}{2} \\ 0.35(2\sqrt{15^2-l^2}) & \frac{15\sqrt{2}}{2} < l < 15 \\ 0 & else \end{cases}$$

The sketch is as follows.

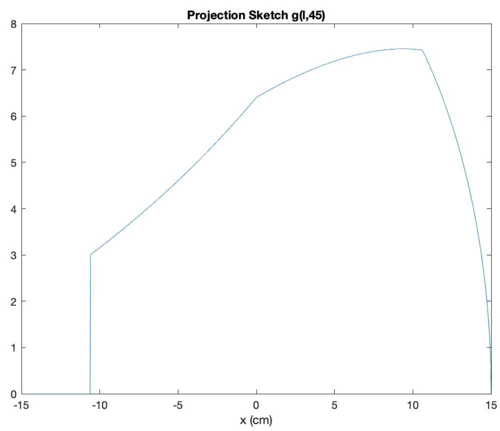

Figure 3: The sketch of $g(l, 45°)$

b. In this part, we are asked to display the backprojected image with $\theta = 45°$. for this, we have used the **iradon** function of MATLAB. The resulting image is provided below.

Figure 4: Backprojection $b_{45^\circ}(x, y)$

c. This part asks about the smallest possible circular FOV to image the entire image. FOV is the distance of the object to be measured from the centre of the CT-System, which corresponds to a distance of $R = 15cm$ from the center. We are also asked the shortest length of the detector array to cover the FOV, which is the maximum of the minimum range of the projections for all possible angles, which is $2R$ in this case, making it $30cm$. We see no relevance if the information that is given to us, which is the distance from the source to the detector being $1m$. The parallel-ray projections should be taken such that the centre of the CT system is accepted as the FOV distance from the detector, regardless of the distance to the source.

d. For this part, we are asked to record the angles that should be get, which is the number of projections. The minimum number of projections should be equal to $\frac{\pi}{2}N_{proj}$, which is equal to $\frac{\pi}{2}256 = 402.12$. Hence, we set the number of projections to 403. Together, this makes a $403x256$ projection plane. By backprojection, the reconstructed image becomes, $256x256$ in size.

Question 3

In this question, we are asked to find the 2D Radon transforms $g(l, \theta)$ of the following functions using the Projection-slice Theorem. Projection-slice theorem is the theorem in which we can use the Fourier transform to find the radon transform of the functions. We can define it as follows.

$$g(l, \theta) = \mathcal{F}_{1D}^{-1}\{G(\rho\cos\theta, \rho\sin\theta)\} \tag{7}$$

$$where \quad G(\rho\cos\theta, \rho\sin\theta) = F(\rho\cos\theta, \rho\sin\theta) \tag{8}$$

$$where \quad F(\rho\cos\theta, \rho\sin\theta) = \mathcal{F}_{2D}\{f(x, y)\}|_{u=\rho\cos\theta, v=\rho\sin\theta} \tag{9}$$

a. $f(x, y) = e^{-x^2} = e^{-\pi(x/\sqrt{\pi})^2}$
First, we find the 2D Fourier transform,

$$F(u, v) = \pi e^{-\pi^2 u^2} \tag{10}$$

Then, we change the coordinate system to ρ and θ.

$$F(\rho\cos\theta, \rho\sin\theta) = \pi e^{-\pi^2(\rho\cos\theta)^2} \tag{11}$$

Then, we use 1D inverse Fourier transform to get the Radon transform.

$$g(l,\theta) = \mathcal{F}_{1D}^{-1}\{\pi e^{-\pi\left(\sqrt{\pi}\rho\cos\theta\right)^2}\} \tag{12}$$
$$= \frac{1}{cos^2\theta} e^{-l^2/cos^2\theta} \tag{13}$$

b. $f(x,y) = rect(x,y) - \delta(x,y)$
We follow the same approach as in the previous part.

$$F(u,v) = sinc(u,v) - 1 \tag{14}$$
$$F(\rho\cos\theta, \rho\sin\theta) = sinc(\rho\cos\theta, \rho\sin\theta) - 1 \tag{15}$$
$$= sinc(\rho\cos\theta) sinc(\rho\sin\theta) - 1 \tag{16}$$

$$g(l,\theta) = \mathcal{F}_{1D}^{-1}\{sinc(\rho\cos\theta)sinc(\rho\sin\theta) - 1\} \tag{17}$$
$$= \frac{1}{\sin\theta\cos\theta}\left[rect\left(\frac{l}{\cos\theta}\right) * rect\left(\frac{l}{\sin\theta}\right)\right] - \delta(l) \tag{18}$$
$$= \frac{2}{\sin 2\theta}\left[rect\left(\frac{l}{\cos\theta}\right) * rect\left(\frac{l}{\sin\theta}\right)\right] - \delta(l) \tag{19}$$

Question 4

this question asks us to find $f(x,y)$, given their Radon transforms $g(l,\theta)$. Here, we approach this question using the Projection-Slice, theorem as explained in the previous part. The operations are provided as follows. We first take the 1-D Fourier transform of g.

$$G(\rho,\theta) = \mathcal{F}_{1D}\{g(l,\theta)\} \tag{20}$$

After that, we can express $G(\rho,\theta)$ in terms of spatial frequencies.

$$G(\rho,\theta) = G(\rho\cos\theta, \rho\sin\theta) = F(u,v) \tag{21}$$

Then, we can say that,

$$f(x,y) = \mathcal{F}_{2D}^{-1}\{F(u,v)\} \tag{22}$$

For some parts, where $G(\rho,\theta)$ are circularly symmetric, we use the inverse Hankel transform instead of the inverse Fourier transform as follows.

$$G(\rho,\theta) = \mathcal{F}_{1D}\{g(l,\theta)\} = G(\rho) \tag{23}$$
$$f(r) = \mathcal{H}^{-1}\{G(\rho)\} \tag{24}$$
$$f(x,y) = f(\sqrt{x^2+y^2}) \tag{25}$$

a. $g(l,\theta) = \delta(l)$

$$G(\rho,\theta) = \mathcal{F}_{1D}\{g(l,\theta)\} = 1 = F(u,v) \tag{26}$$
$$f(x,y) = \mathcal{F}_{2D}^{-1}\{F(u,v)\} = \delta(x,y) \tag{27}$$

This is intuitive since the overall projection can be an impulse in $l = 0$ only if the object is also an impulse in the origin of the system.

b. $g(l, \theta) = \delta(l - a\sin\theta)$

$$G(\rho, \theta) = \mathcal{F}_{1D}\{g(l, \theta)\} \tag{28}$$
$$= e^{-2\pi j(\rho a \sin\theta)}.1 \tag{29}$$
$$= e^{-2\pi j(a\rho \sin\theta)} \tag{30}$$

Then, we can write $F(u, v)$ as given.

$$F(u, v) = G(\rho\cos\theta, \rho\sin\theta) \tag{31}$$
$$= e^{-2\pi j(av)} \tag{32}$$

Then, we can find $f(x, y)$ using inverse Fourier transform,

$$f(x, y) = \delta(x, y - a) \tag{33}$$

c. $g(l, \theta) = rect(l)$

$$G(\rho, \theta) = \mathcal{F}_{1D}\{g(l, \theta)\} = sinc(\rho) \tag{34}$$
$$\tag{35}$$

using the Hankel transform table and knowing the forward and inverse Hankel transforms yield the same result (Medical Imaging Signals and Systems, Prince and Links), we can find $f(r)$.

$$f(r) = sinc(r) \xrightarrow{\mathcal{H}} F(q) = \frac{2rect(q)}{\pi\sqrt{1 - 4q^2}} \tag{36}$$

then,

$$f(r) = \mathcal{H}^{-1}\{sinc(\rho)\} = \frac{2rect(r)}{\pi\sqrt{1 - 4r^2}} \tag{37}$$
$$f(x, y) = \frac{2rect(\sqrt{x^2 + y^2})}{\pi\sqrt{1 - 4(x^2 + y^2)}} \tag{38}$$

d. $g(l, \theta) = rect(l - a\sin\theta)$

$$G(\rho, \theta) = \mathcal{F}_{1D}\{g(l, \theta)\} = e^{-2\pi j(a\rho\sin\theta)}sinc(\rho) \tag{39}$$

Here, we can separate the functions and evaluate them separately.

$$G_1(\rho, \theta) = e^{-2\pi j(a\rho\sin\theta)} \tag{40}$$
$$G_2(\rho) = sinc(\rho) \tag{41}$$

Hence, these become the parts, we answered in parts b and c. We can directly write their corresponding $f_i(x, y)$ $i \in 1, 2$ as given.

$$f_1(x, y) = \delta(x, y - a) \tag{42}$$
$$f_2(x, y) = \frac{2rect(\sqrt{x^2 + y^2})}{\pi\sqrt{1 - 4(x^2 + y^2)}} \tag{43}$$

Hence, we can define $f(x, y)$ using the Convolution property of Fourier transform and write the solution.

$$f(x, y) = f_1(x, y) * f_2(x, y) \tag{44}$$
$$= \delta(x, y - a) * \frac{2rect(\sqrt{x^2 + y^2})}{\pi\sqrt{1 - 4(x^2 + y^2)}} \tag{45}$$
$$= \frac{2rect(\sqrt{x^2 + (y - a)^2})}{\pi\sqrt{1 - 4(x^2 + (y - a)^2)}} \tag{46}$$

Question 5

In this question, we are required to construct an ideal image, then use the MATLAB functions **radon** and **iradon** functions to emulate, CT projections and then use backprojection with changing parameters, such as filter type and number of projections. The source code is presented in the Appendix.

a. This part asks us to create and plot the ideal image P. The image is given below.

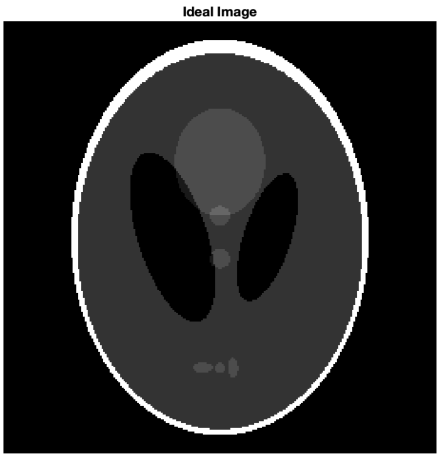

Figure 5: The Ideal Image

b. This part asks us to take a sufficient number of projections and then reconstruct using the inverse Radon transform. For the `sufficient` number, we have chosen 120 angles uniformly distributed between $[0, 179]^{\circ}$, so that there are no visible artifacts in the backprojected image.

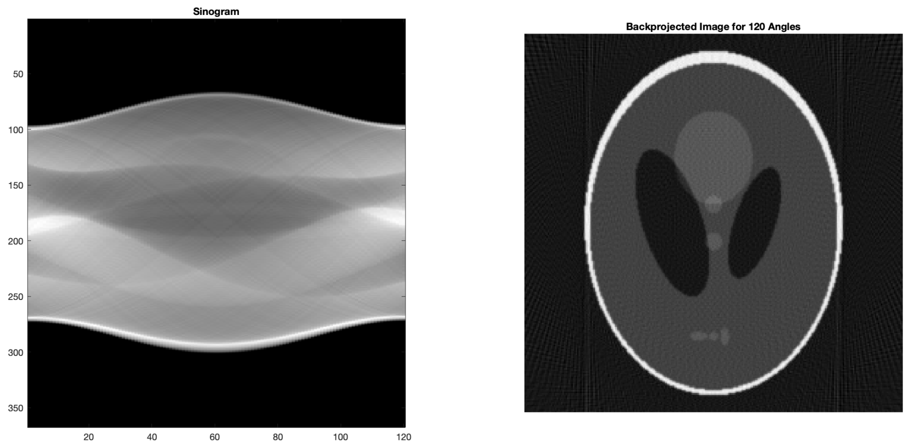

Figure 6: The Sinogram and the Backprojected Image for 120 Angles

c. This part asks us to plot the projections of P for the following angles $\theta = \{0, 45, 90, 135\}^{\circ}$. The projections are as follows.

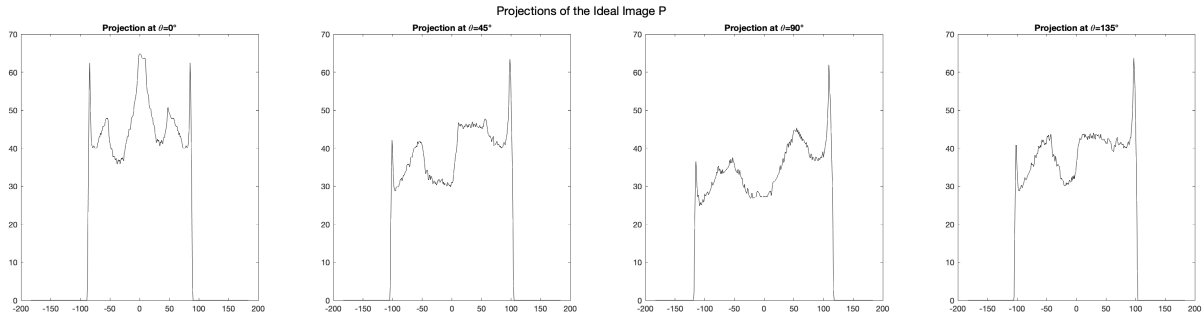

Figure 7: Projections for $\theta = \{0, 45, 90, 135\}^{\circ}$

d. This time, we are asked to repeat Part B, with fewer number of projection angles, hence for this part we have used 40, so that there are visible artifacts. The question also asks the kind of artifact we are observing on the image. The artifact we observe on the backprojection is called the aliasing artifact, as we take linear projections that pass through the center of the image plane, we are losing information between the projections as the distance from the origin increases, and as a result of insufficient number of projections, we see this artifact. The sinogram and the backprojected image is as follows.

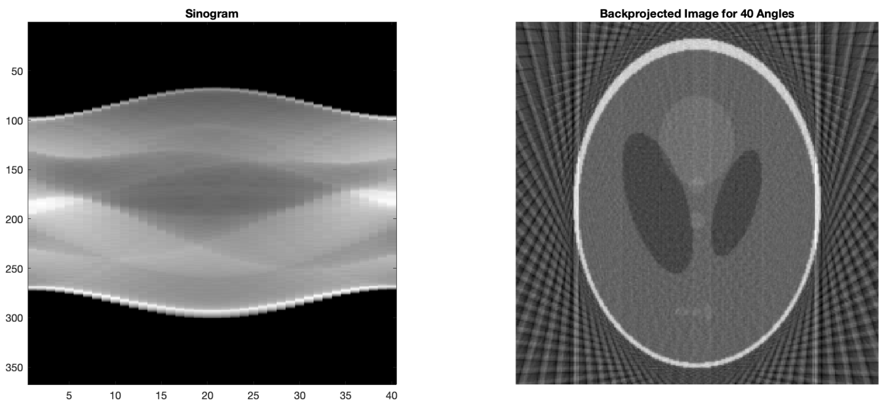

Figure 8: The Sinogram and the Backprojected Image for 40 Angles

e. In this part, we are required to use two different filters for the filtered backprojection and then use the unfiltered backprojection and compare the results. The first filter is the cropped ramp filter, which is the multiplication of a ramp with a rect function, which is given by the **Ram-Lak** filter option in MATLAB. The second filter is the Hamming filter (**Hamming** in MATLAB). The different backprojections are as follows.

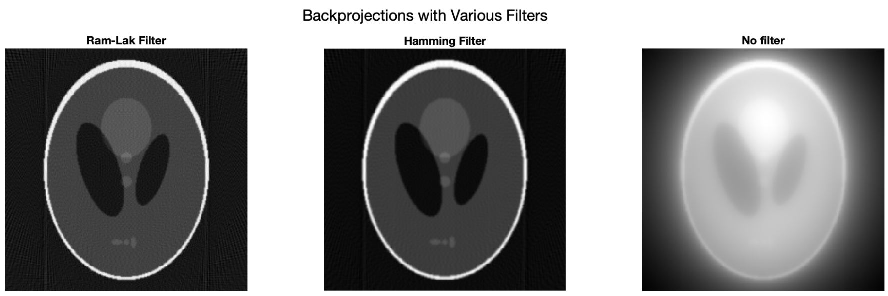

Figure 9: Backprojected Images with Various Filters - 120 Angles

Here, we can easily see that the un-filtered backprojection has a visible hazing effect, which was expected from the sum of the projections, which is not a good image in general. when we compare the Ram-Lak filter and the Hamming filter, we see that the Hamming filter has less visible artifacts and has better contrast in general.

f. In this part, we are asked to repeat the previous part for the fewer number of projection angles. The backprojected images are as follows.

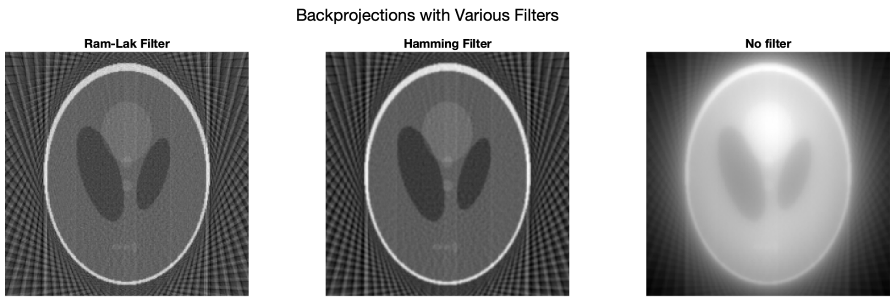

Figure 10: Backprojected Images with Various Filters - 40 Angles

Here, we see that, the unfiltered option is better in terms of artifacts since the image content is too bright in the area object covers. However, the image is hazy again with this number of angles. The first two filters presented a considerably noisy image with aliasing artifacts.

MATLAB Code

Code that is used in Questions 2 and 5 are presented below.

hw4q2.m

```
close all; clear;
x = linspace(-15,15, 1000);
R = 15; mu1 = 0.25; mu2 = 0.05; mu3 = 0.35;
fx = [];
for xi = x
    fx = [fx f(xi,R,mu1,mu2,mu3)];
end
plot(x,fx)
disp(size(fx'))
title('Projection Sketch g(l,45)')
xlabel('x (cm)')
figure()
backproj = iradon(fx', 45, 'none');
imshow(backproj, [])

function res=f(l,r,mu1,mu2,mu3)
    if(l>(-r*sqrt(2)/2) && l<=0)
        res = mu1*(r*sqrt(2)/2 + sqrt(r^2+l^2) + 2*l) - mu2*(2*l);
    elseif(l>0 && l<=(r*sqrt(2)/2))
        res = mu1*(r*sqrt(2)/2 + sqrt(r^2-l^2) - 2*l) + mu3*(2*l);
    elseif(l>(r*sqrt(2)/2) && l<r)
        res = mu3 * (2*sqrt(r^2-l^2));
    else
        res = 0;
    end
end
```

hw4.m

```
close all; clear;
% generate phantom image
p = phantom('Modified Shepp-Logan',256);

%% a) display p
figure()
imshow(p, [])
title('Ideal Image')

%% b) take radon transform and backproject
NUM_ANGLES_SUFF = 120;
angles = linspace(0,179, NUM_ANGLES_SUFF);
projections = radon(p, angles);
figure()
subplot(1,2,1);
colormap(gray);
imagesc(projections);

title('Sinogram')
backproj = iradon(projections, angles);
subplot(1,2,2)
imshow(backproj, [])
title(['Backprojected Image for ', num2str(NUM_ANGLES_SUFF) ' Angles'])

%% c) plot projections for 0,45,90,135
% we ahve used linear spacing for angles, so we again use radon transform
% the get the desired projections.
angles = [0,45,90,135];
[projections,xp] = radon(p, angles);
figure()
for i=1:4
    subplot(1,4,i)
    plot(xp,projections(:,i),'color', 'black');
```

```
    title(['Projection at \theta=', num2str(angles(i)), '  ']);
end
sgtitle('Projections of the Ideal Image P')

%% d) repeat part b for fewer number of projections
NUM_ANGLES_INSUF = 40;
angles = linspace(0,179,NUM_ANGLES_INSUF);
projections = radon(p, angles);
figure()
subplot(1,2,1);
colormap(gray);
imagesc(projections);
title('Sinogram')
backproj = iradon(projections, angles);
subplot(1,2,2)
imshow(backproj, [])
title(['Backprojected Image for ', num2str(NUM_ANGLES_INSUF), ' Angles'])
% This is the aliasing artifact in CT resulting from the insufficient
% number of projections.

%% e) repeat b with three types of filters.
angles = linspace(0,179, NUM_ANGLES_SUFF);
projections = radon(p, angles);
figure()
filters = ["Ram-Lak", "Hamming", "none"];
num_filters = size(filters);
for i=1:num_filters(2)
    backproj = iradon(projections, angles, filters(i));
    subplot(1,num_filters(2),i)
    imshow(backproj, [])
    if i==3
        title('No filter')
    else
        title(filters(i) + ' Filter')
    end
end
sgtitle('Backprojections with Various Filters')

%% f) repeat part e for d
angles = linspace(0,179, NUM_ANGLES_INSUF);
projections = radon(p, angles);
figure()
filters = ["Ram-Lak", "Hamming", "none"];
num_filters = size(filters);
for i=1:num_filters(2)
    backproj = iradon(projections, angles, filters(i));
    subplot(1,num_filters(2),i)
    imshow(backproj, [])
    if i==3
        title('No filter')
    else
        title(filters(i) + ' Filter')
    end
end
sgtitle('Backprojections with Various Filters')
```