MATH473 HW3

Jiasen Zhang

1 Problem 1

Sufficiency: If $x \in \mathbb{C}^N$ is real valued, then $\bar{x} = x$ and $\bar{x}_m = x_m$ for $0 \leq m \leq N-1$.

$$X_k = \sum_{m=0}^{N-1} x_m e^{-2\pi ikm/N} \qquad X_{-k} = \sum_{m=0}^{N-1} x_m e^{2\pi ikm/N}$$

$$\begin{aligned}\bar{X}_k &= \sum_{m=0}^{N-1} \bar{x}_m e^{2\pi ikm/N} \\ &= \sum_{m=0}^{N-1} x_m e^{2\pi ikm/N} \\ &= X_{-k}\end{aligned}$$

Necessity: If $\bar{X}_k = X_{-k}$,

$$\sum_{m=0}^{N-1} \bar{x}_m e^{2\pi ikm/N} = \sum_{m=0}^{N-1} x_m e^{2\pi ikm/N}$$

That means $\bar{x}_m = x_m$ for $0 \leq m \leq N-1$, and therefore x is real valued.

So we have: $x \in \mathbb{C}^N$ is real valued $\iff \bar{X}_k = X_{-k}$.

2 Problem 2

2.1 (a)

The result is in Figure 1.

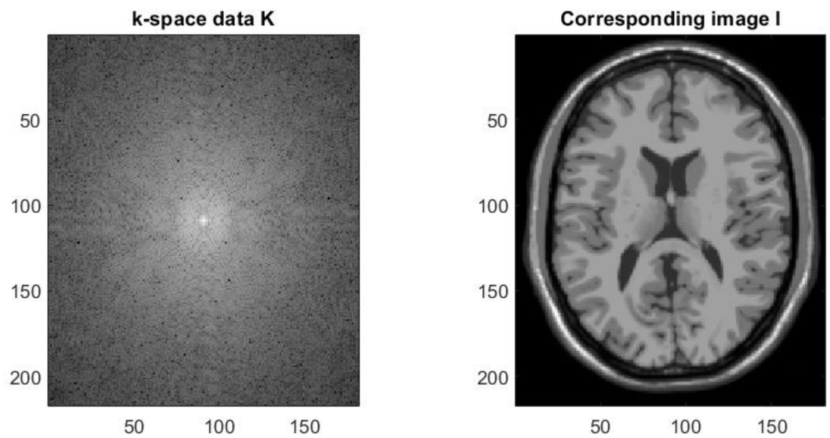


Figure 1: Original image

2.2 (b)

The result is in Figure 2. Compared with the image in (a), the filtered image is blurred, or to say, smoothed.

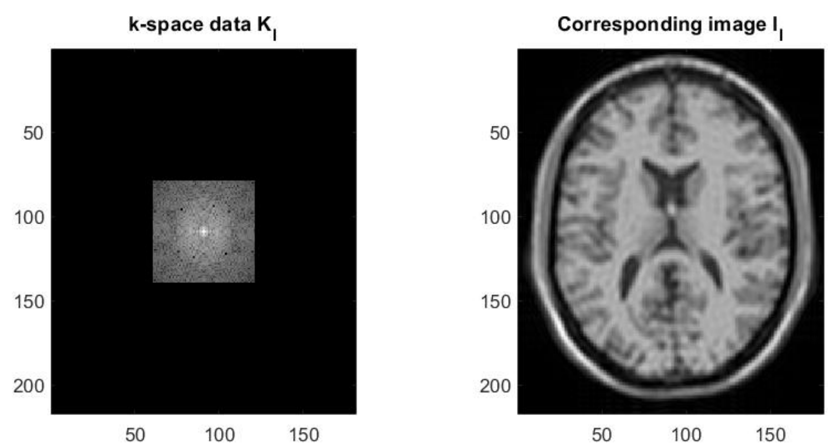


Figure 2: Low pass filtering

2.3 (c)

The result is in Figure 3. Compared with the image in (a), the edge is sharpened by removing where the image is constant(white). We can see more outlines of the brain.

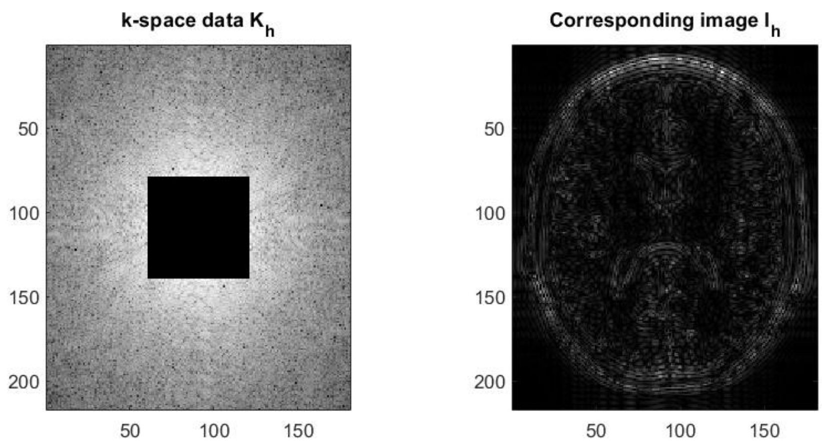


Figure 3: High pass filtering

2.4 (d)

The result is in Figure 4. Some constant parts are removed and the edge is slightly smoothed. As the result more useful details are preserved compared with (c).

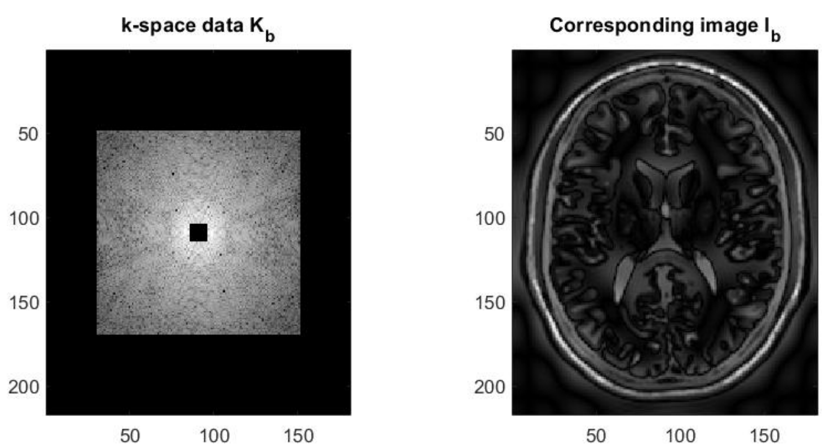


Figure 4: Band pass filtering

2.5 (e)

According to problem 1, when the image I is real valued and its DFT has 0 frequency at the center, the DFT satisfies $\bar{X}_k = X_{-k}$. In 2D case, we have $\bar{X}_{k,j} = X_{-k,-j}$. So the k-space data K is conjugate symmetry.

In this problem K is 217×181 with center at $(109, 91)$. Suppose $K(r, s) = a + bi$ where $1 \leq r \leq 217$ and $1 \leq s \leq 181$, then we know $K(218 - r, 182 - s) = a - bi$. That means we can sample half of the k-space data, and get the other half using such conjugate symmetry. Therefore the scan time is reduced by half.

2.6 (f)

The result is in Figure 5.

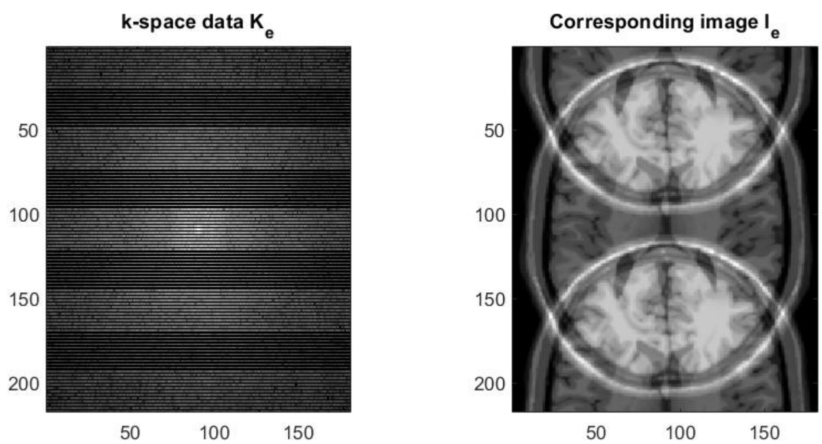


Figure 5: Equidistance under-sampling

3 Problem 3

3.1 (a)

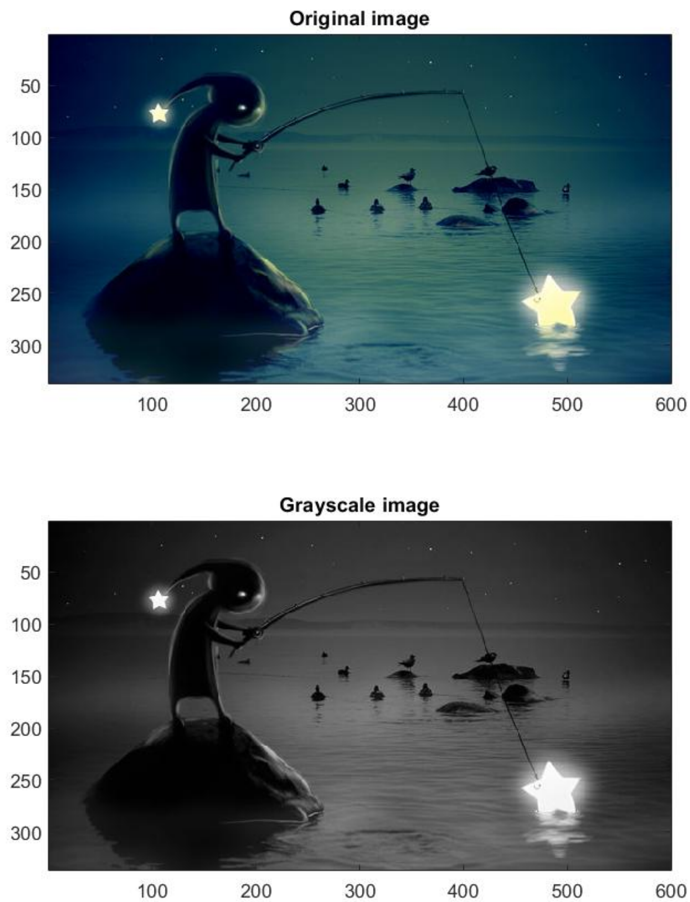
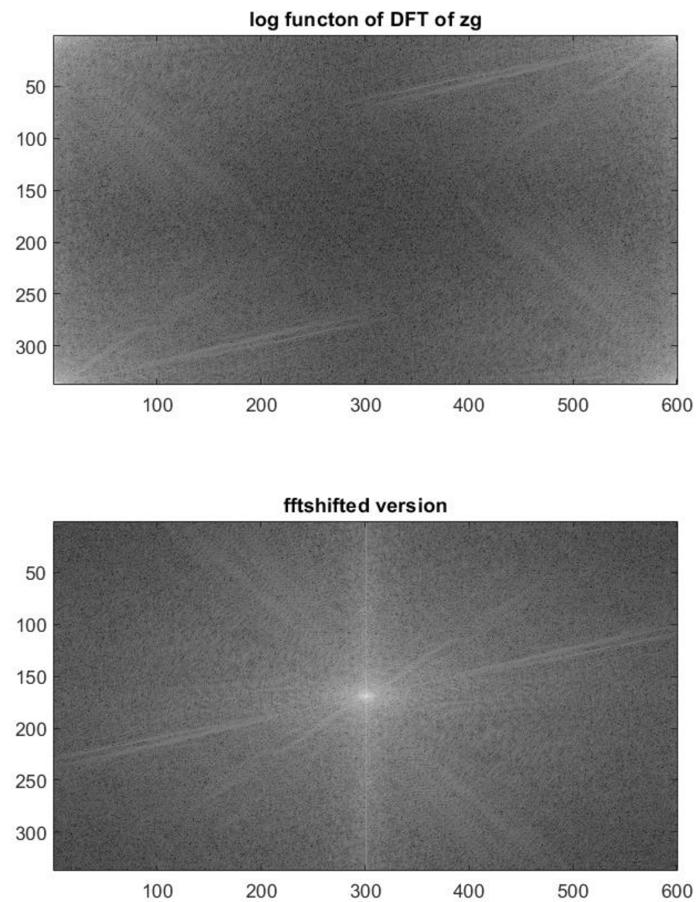


Figure 6: Original image and grayscale image.

3.2 (b)



log functon of DFT of zg
50
100
150
200
250
300
100
200
300
400
500
600
fftshifted version
50
100
150
200
250
300
100
200
300
400
500
600

Figure 7

3.3 (c)

Figure 8 shows the compressed images when *thresh*=0.0001, 0.0005, 0.001, 0.005, 0.01, 0.05.

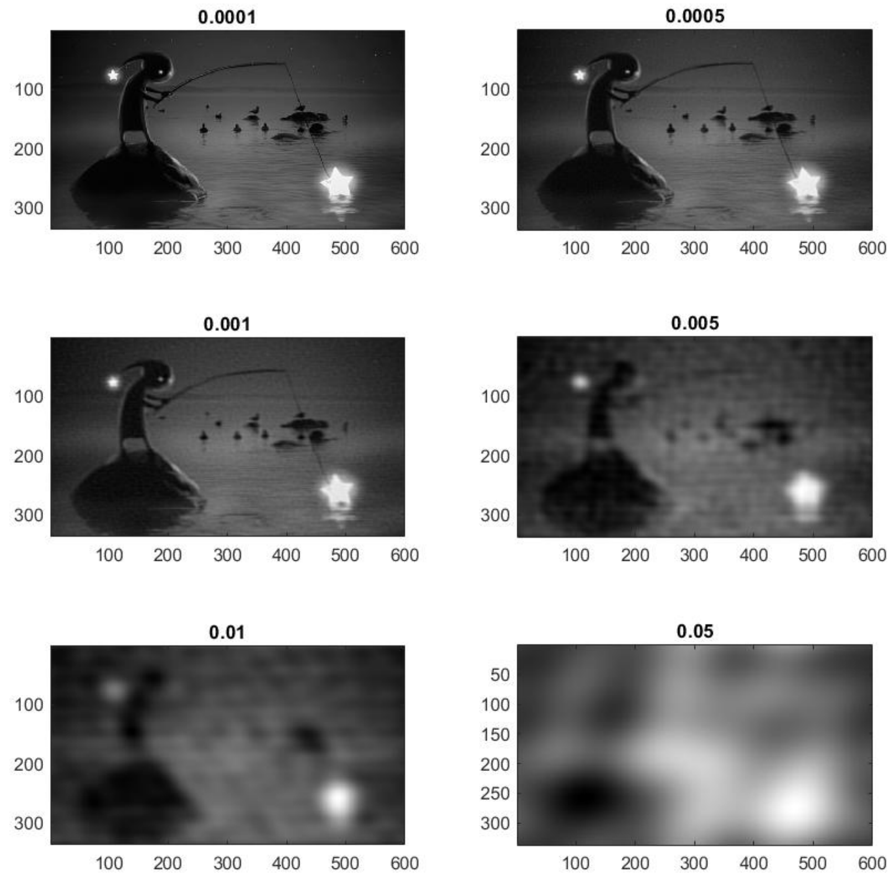


Figure 8

Table 1 gives the compression ratio, distortion and my rate of quality for each *thresh*. The first three images are almost the same as the original image, so I rate A and A- for them. The images with *thresh*=0.005 and 0.01 are blurred but we can still recoginize some edges, so I rate B and C for them. Figure 9 shows the plot of log(distortion) vs. log(compression ratio)

thresh	compression ratio	distortion	quality
0.0001	0.473096	0.024391	A
0.0005	0.073576	0.328921	A
0.001	0.027997	0.646262	A-
0.005	0.002646	2.140116	B
0.01	0.000915	3.342565	C
0.05	0.000114	9.161264	F

Table 1

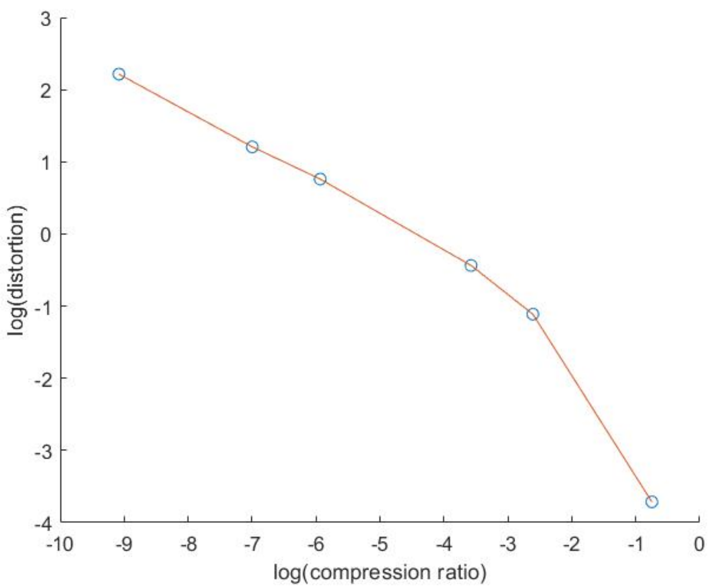


Figure 9

4 Problem 4

Here $N = 4$, $x = \frac{F_N^*}{N} X = \frac{F_4^*}{4} X$

$$F_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & z & z^2 & z^3 \\ 1 & z^2 & z^4 & z^6 \\ 1 & z^3 & z^6 & z^9 \end{bmatrix}$$

$$\text{where} \quad z = e^{-2\pi i/4} = \cos(-\frac{\pi}{2}) + i\sin(-\frac{\pi}{2}) = 0 + i(-1) = -i$$

So we have

$$F_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -i & -1 & i \\ 1 & -1 & 1 & -1 \\ 1 & i & -1 & -i \end{bmatrix}$$

The inverse DFT of X is:

$$x = \frac{1}{4}F_4^* X = \frac{1}{4}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & i & -1 & -i \\ 1 & -1 & 1 & -1 \\ 1 & -i & -1 & i \end{bmatrix}\begin{bmatrix} 3 \\ 1+i \\ 1 \\ 1-i \end{bmatrix} = \frac{1}{4}\begin{bmatrix} 6 \\ 0 \\ 2 \\ 4 \end{bmatrix} = \begin{bmatrix} 1.5 \\ 0 \\ 0.5 \\ 1 \end{bmatrix}$$

5 Problem 5

The 1D DFT of $\mathbf{x}$ and $\mathbf{y}$ are $\mathbf{X} = F_m\mathbf{x}$, $\mathbf{Y} = F_n\mathbf{y}$.

$$\mathbf{X}\mathbf{Y}^T = F_m\mathbf{x}\mathbf{y}^T F_n^T = F_m\mathbf{z}F_n^T$$

The outer product $\mathbf{z} = \mathbf{x}\mathbf{y}^T$ is a $m \times n$ matrix, its 2D DFT is $\mathbf{Z} = F_m\mathbf{z}F_n^T = \mathbf{X}\mathbf{Y}^T$

So $\mathbf{Z} = \mathbf{X}\mathbf{Y}^T$.

6 Problem 6

Define two vectors:

$\mathbf{e_j} \in \mathbb{C}^m$ having a 1 in the row j and zeros elsewhere.
$\mathbf{e_k} \in \mathbb{C}^n$ having a 1 in the row k and zeros elsewhere.

According to the result of problem 5, because $\mathbf{e_{j,k}} = \mathbf{e_j}\mathbf{e_k}^T$, we have:

$$\text{DFT}(\mathbf{e_{j,k}}) = \text{DFT}(\mathbf{e_j})\text{DFT}(\mathbf{e_k})^T$$

$$\begin{aligned}\text{DFT}(\mathbf{e_j}) &= F_m\mathbf{e_j} = \text{jth column of } F_m \\ &= [1, z_m^{j-1}, z_m^{2(j-1)}, \cdots, z_m^{(m-1)(j-1)}]^T \quad \text{where } z_m = e^{-2\pi i/m}\end{aligned}$$

$$\begin{aligned}\text{DFT}(\mathbf{e_k}) &= F_n\mathbf{e_k} = \text{kth column of } F_n \\ &= [1, z_n^{k-1}, z_n^{2(k-1)}, \cdots, z_n^{(n-1)(k-1)}]^T \quad \text{where } z_n = e^{-2\pi i/n}\end{aligned}$$

So we have:

$$\begin{aligned}\text{DFT}(\mathbf{e_{j,k}}) &= \begin{bmatrix} 1 \\ z_m^{j-1} \\ z_n^{2(j-1)} \\ \vdots \\ z_m^{(m-1)(j-1)} \end{bmatrix}\begin{bmatrix} 1 \\ z_n^{k-1} \\ z_n^{2(k-1)} \\ \vdots \\ z_n^{(n-1)(k-1)} \end{bmatrix}^T \\ &= \begin{bmatrix} 1 & z_n^{k-1} & \cdots & z_n^{(n-1)(k-1)} \\ z_m^{j-1} & z_m^{j-1}z_n^{k-1} & & \\ \vdots & & \ddots & \\ z_m^{(m-1)(j-1)} & & & z_m^{(m-1)(j-1)}z_n^{(n-1)(k-1)} \end{bmatrix}\end{aligned}$$

The entry (r,s) of $\text{DFT}(\mathbf{e_{j,k}})$ is $(1 \le r \le m,\ 1 \le s \le n)$:

$$\begin{aligned}\text{DFT}(\mathbf{e_{j,k}})_{r,s} &= z_m^{(r-1)(j-1)}z_n^{(s-1)(k-1)} \\ &= \exp\left[-2\pi i\Big(\frac{(r-1)(j-1)}{m} + \frac{(s-1)(k-1)}{n}\Big)\right]\end{aligned}$$

7 Matlab Code

7.1 Problem 2

```
clear all;clc;
load T1Brain;
%(a)
K = fftshift(fft2(I));
[m,n]=size(K);
figure();
subplot(1,2,1);
imagesc(log(abs(K)));axis image;colormap(gray);
title('k-space data K');
subplot(1,2,2);
imagesc(abs(I));axis image;colormap(gray);
title('Corresponding image I');
aK=log(abs(K));

%(b) low pass
Kl=zeros(m,n);
side=61;
ci = round(m/2); % center of K
cj = round(n/2);
Kl(ci-(side-1)/2:ci+(side-1)/2,cj-(side-1)/2:cj+(side-1)/2)...
= K(ci-(side-1)/2:ci+(side-1)/2,cj-(side-1)/2:cj+(side-1)/2);
Il=ifft2(Kl);
figure();
subplot(1,2,1);
imagesc(log(abs(Kl)));axis image;colormap(gray);
title('k-space data K_l');
subplot(1,2,2);
imagesc(abs(Il));axis image;colormap(gray);
title('Corresponding image I_l');

%(c) high pass
Kh=K;
Kh(ci-(side-1)/2:ci+(side-1)/2,cj-(side-1)/2:cj+(side-1)/2)= 0;
Ih=ifft2(Kh);
figure();
subplot(1,2,1);
imagesc(log(abs(Kh)));axis image;colormap(gray);
title('k-space data K_h');
subplot(1,2,2);
imagesc(abs(Ih));axis image;colormap(gray);
title('Corresponding image I_h');

%(d)
Kb=zeros(m,n);
side1=11;
side2=121;
Kb(ci-(side2-1)/2:ci+(side2-1)/2,cj-(side2-1)/2:cj+(side2-1)/2)...
= K(ci-(side2-1)/2:ci+(side2-1)/2,cj-(side2-1)/2:cj+(side2-1)/2);
Kb(ci-(side1-1)/2:ci+(side1-1)/2,cj-(side1-1)/2:cj+(side1-1)/2)=0;
Ib=ifft2(Kb);
figure();
subplot(1,2,1);
imagesc(log(abs(Kb)));axis image;colormap(gray);
title('k-space data K_b');
subplot(1,2,2);
imagesc(abs(Ib));axis image;colormap(gray);
title('Corresponding image I_b');

%(f)
```

```
Ke = K;
for i=2:2:m-1
    Ke(i,:)=0;
end
Ie=ifft2(Ke);
figure();
subplot(1,2,1);
imagesc(log(abs(Ke)));axis image;colormap(gray);
title('k-space data K_e');
subplot(1,2,2);
imagesc(abs(Ie));axis image;colormap(gray);
title('Corresponding image I_e');
```

7.2 Problem 3

```
clear all;clc;
%(a)
z = imread('myimage.jpg');
zg = 0.2989*double(z(:,:,1)) + 0.5870*double(z(:,:,2)) + 0.1140*double(z(:,:,3));
figure();
subplot(2,1,1);
imagesc(z);axis image;colormap(gray);
title('Original image');
subplot(2,1,2);
imagesc(zg);axis image;colormap(gray);
title('Grayscale image');

%(b)
Z = fft2(zg);
Zlog = log(1 + abs(Z));
M = max(max(Zlog));
figure();
subplot(2,1,1);
imagesc(Zlog);axis image;colormap(gray);
%image(255*Zlog/M);axis image;colormap(gray);
title('log functon of DFT of zg');
Zs=fftshift(fft2(zg));
subplot(2,1,2);
imagesc(log(1+abs(Zs)));axis image;colormap(gray);
title('fftshifted version');

%(c)
[m,n]=size(zg);
t=[0.0001;0.0005;0.001;0.005;0.01;0.05]; % 6 values of thresh
M = max(max(Z));
figure();
for i=1:6
    thresh = t(i);
    Zthresh = (abs(Z)>thresh*M).*Z;
    compressionRatio(i) = sum(sum(abs(Zthresh) >0))/(m*n);
    zthresh=real(ifft2(Zthresh));
    distortion(i) = 100*(norm(zg-zthresh,'fro')^2)/(norm(zg,'fro')^2);
    subplot(3,2,i);
    imagesc(zthresh);axis image;colormap(gray);
    title(strcat(num2str(thresh)));
end
figure();
scatter(log(compressionRatio),log(distortion));hold on;
plot(log(compressionRatio),log(distortion));
xlabel('log(compression ratio)');
ylabel('log(distortion)');
```