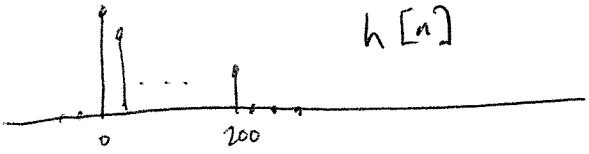
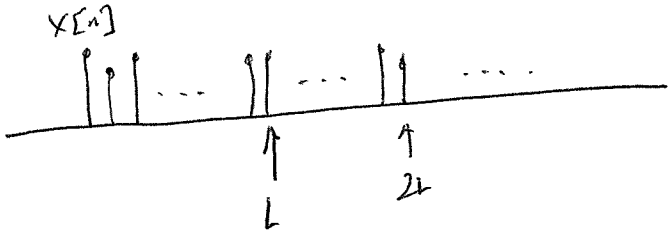
Homework 7 solutions

7.1

a/ $x[n]$ $N = 10^6$

$h[n]$ $P = 200$





DFT Length $1024 = L + P - 1$

$L = 825$

$10^6 / 825 = 1212.12$

round up, since need 1 DFT for last segment

1213 DFTs

But, also need 1 DFT for $h[n]$

(over)

Then, back to time Domain w/ 1213 IDFTs

So,

$x[n] \longleftrightarrow X[k]$ 1213 DFTs

$h[n] \longleftrightarrow H[k]$ 1 DFT

$Y[k] \longleftrightarrow y[n]$ 1213 IDFTs

So

1214 DFTs
1213 IDFTs

2427 total

(Also OK if you assume $h[k]$ is already calculated, then 2426 total).

b/

Direct Convolution

$x[n] * h[n] = y[n]$

$N \cdot 200$ multiplies

$N \cdot 200$ Adds
$\approx$ or 199 adds

$= 2 \cdot 10^8$ Multiplies
and
$= 2 \cdot 10^8$ Adds

DFT Based

$x[n] * h[n] = y[n]$

1 DFT $\sim N+P-1 \approx N$

1 DFT

$X[k] \cdot H[k] = Y[k]$

1 - IDFT $\sim N$ point

$y[n]$

$2\left(\frac{N}{2}\log_2 N\right)$ multiplies

$2(N \log_2 N)$ adds

N multiplies
0 adds

$\frac{N}{2}\log_2 N$ multiplies

$N \log_2 N$ adds

$= 3.1 \times 10^7$ multiplies

$= 6.0 \times 10^7$ Adds

Overlap and adds based

$x[n] * h[n]$

↓

$x_L[n] * h[n]$

↓ 1213 1024 pt DFTs ↑ 1 1024 point DFT

$X[k]\ H[k] = Y_L[k]$

↓ 1213 1024 point IDFTs

$y_L[n]$

↓ $199 \cdot 1212$ adds

$y[n]$

$1214 \cdot \frac{N}{2} \log_2 N$ mult

$1214 \cdot \frac{N}{2} \log_2 N$ adds

$1213 \cdot 1024$ multipliers

$1213\ \frac{N}{2} \log_2 N$ mult

$1213\ N \log_2 N$ adds

$199 \cdot 1212$ adds

$\approx 1.4 \times 10^7$ multipliers

$\approx 2.5 \times 10^7$ adds

Contents

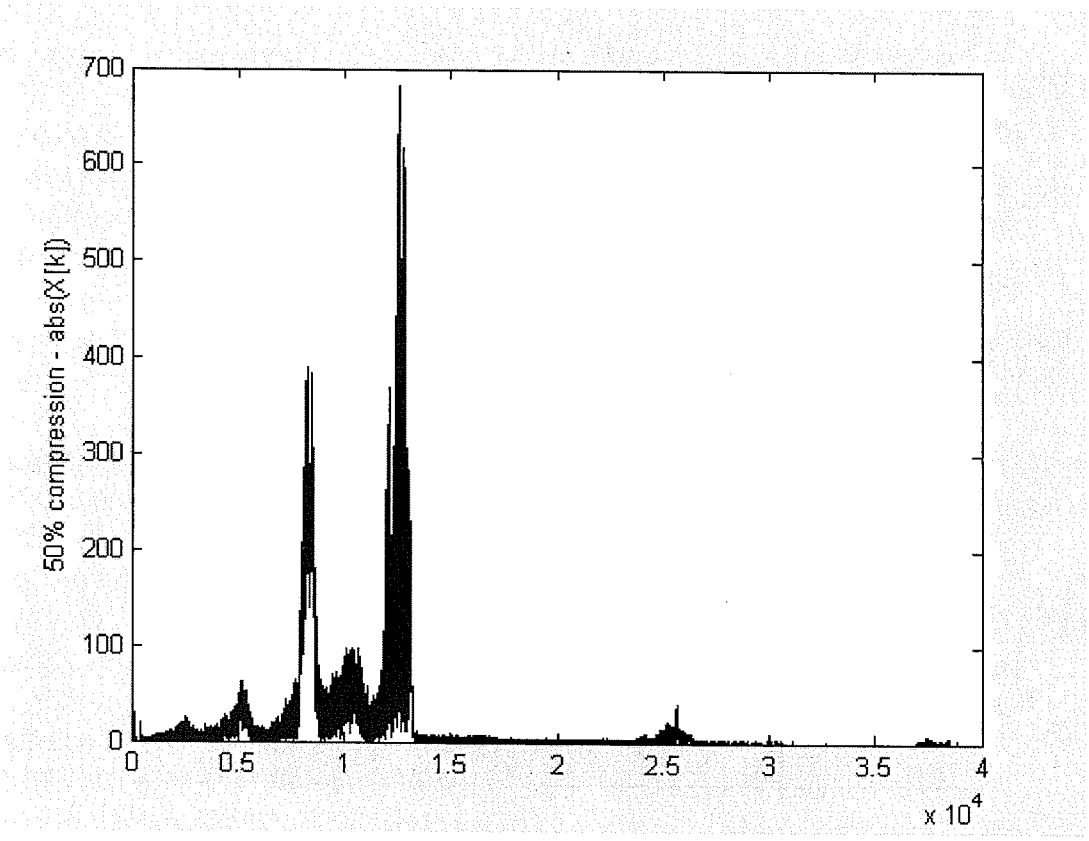
Homework 7 Problem 2

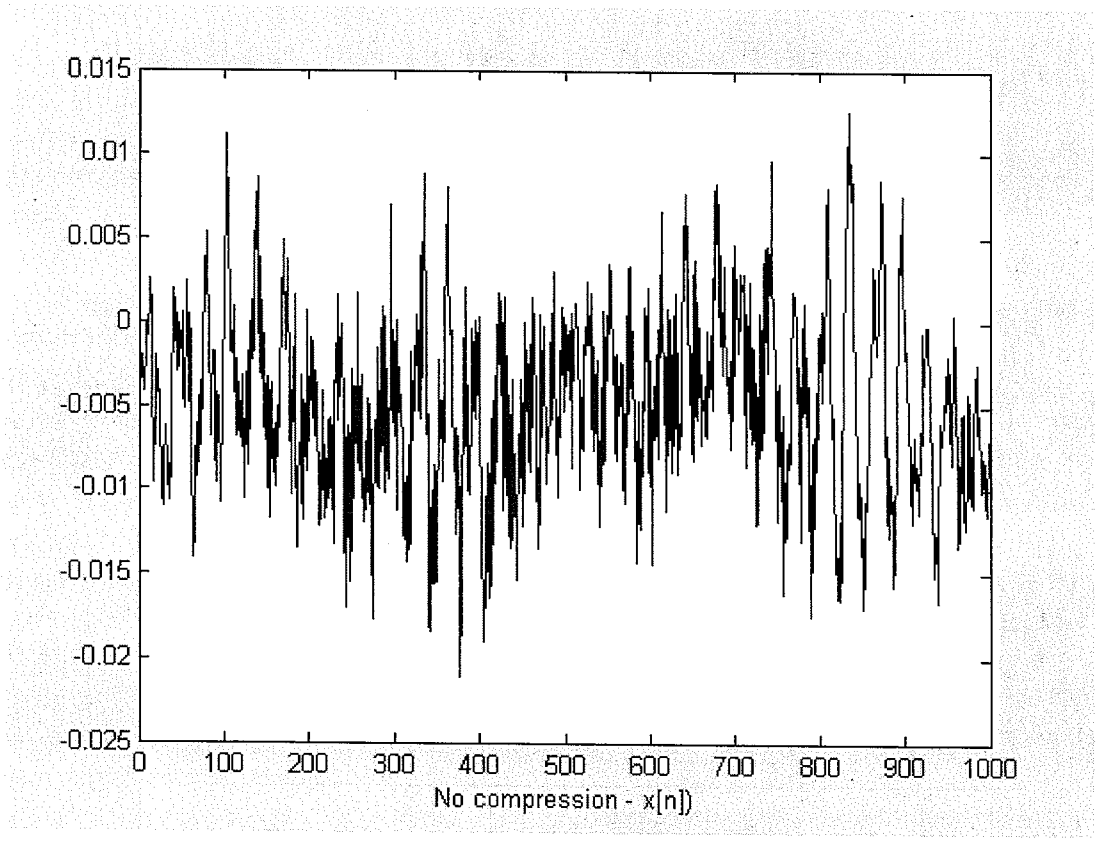
```
clear
close all
clc
```

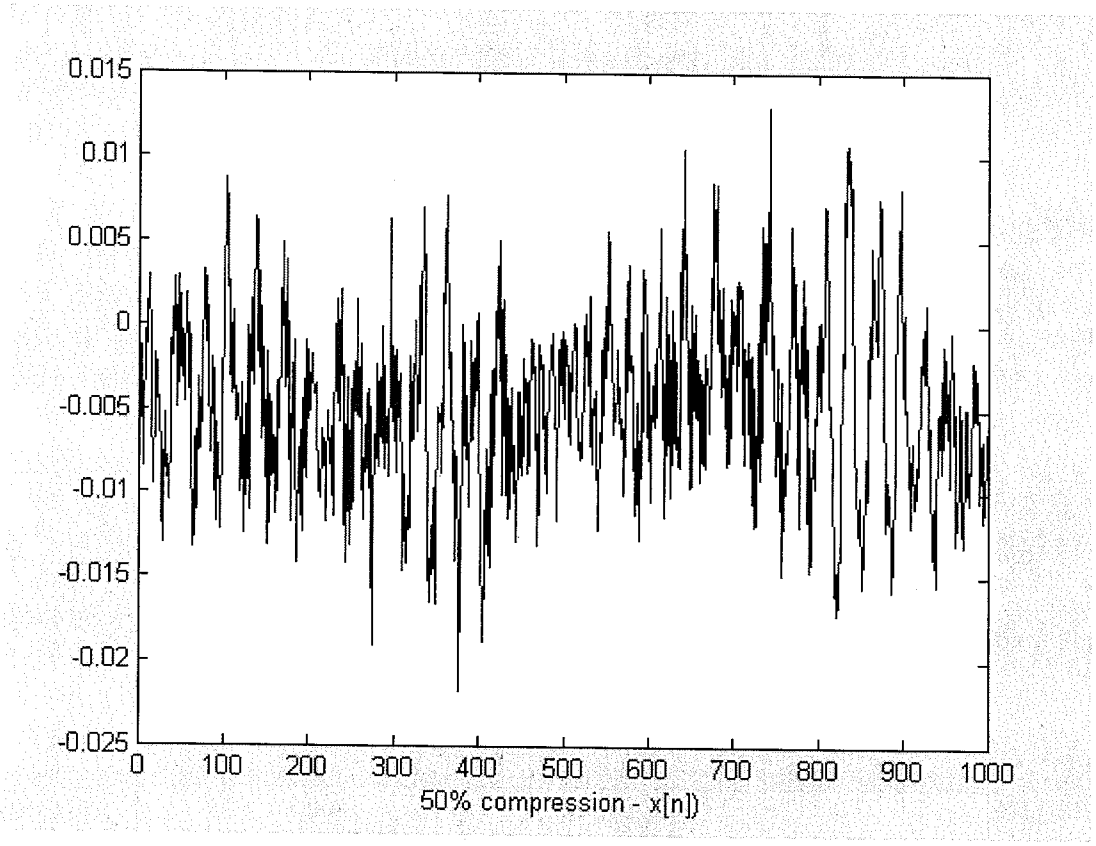
Problem 2a) - 50% compression and original

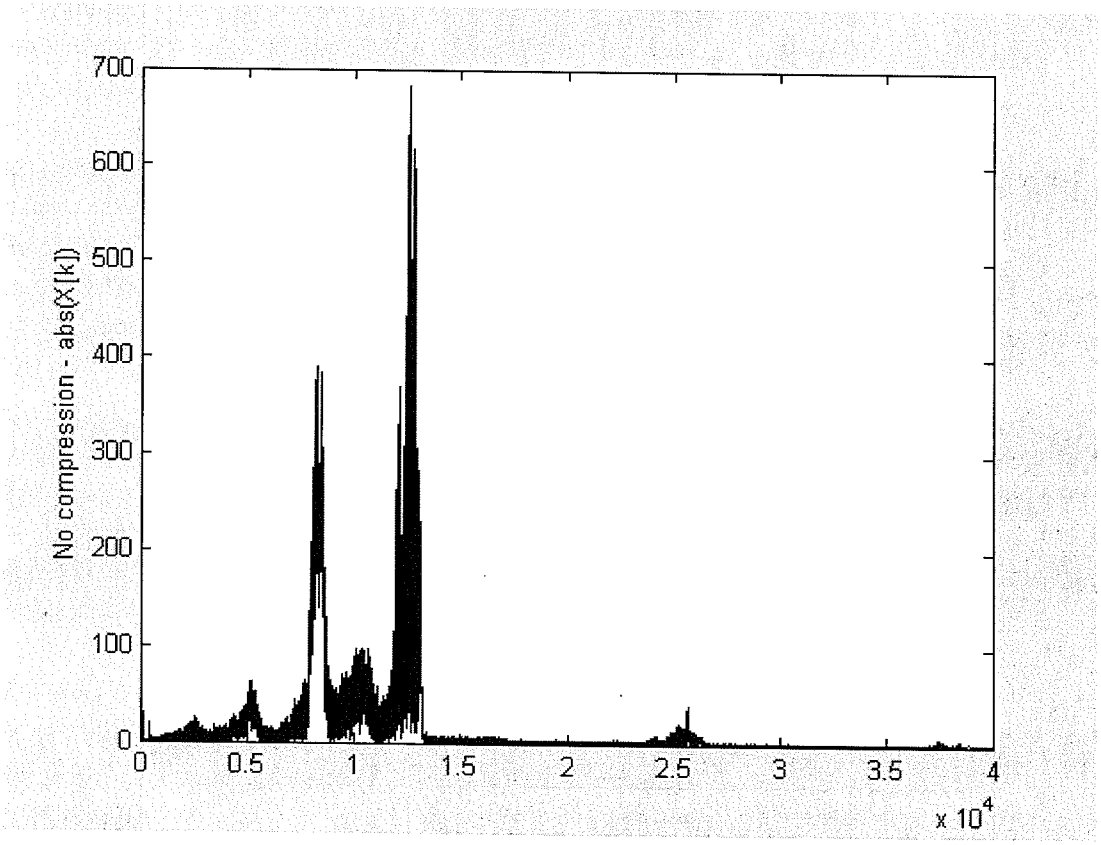
```
load loonHW.mat

M = ceil(length(loon)/2*.50)
L = fft(loon);
Lposfreqs = L(1:ceil(length(loon)/2));
[y,ind] = sort(abs(Lposfreqs));
Lposfreqs(ind(1:M)) = 0;
Lcomp = [Lposfreqs; flipud(conj(Lposfreqs(2:ceil(length(loon)/2))))];

looncomp = ifft(Lcomp);

%Plot original
figure
plot(abs(L(1:length(Lcomp)/2)))
ylabel('No compression - abs(X[k])')
figure
plot(loon(1:1000))
xlabel('No compression - x[n])')
sound(loon,fs)

%Plot compressed
figure
plot(abs(Lcomp(1:length(Lcomp)/2)))
ylabel('50% compression - abs(X[k])')
figure
plot(looncomp(1:1000))
xlabel('50% compression - x[n])')
sound(looncomp,fs)
```

```
M =

       20001

Warning: Integer operands are required for colon operator when used as index
Warning: Integer operands are required for colon operator when used as index
```


50% compression - abs(X[k])
700
600
500
400
300
200
100
0
0
0.5
1
1.5
2
2.5
3
3.5
4
x 10^4


0.015
0.01
0.005
0
-0.005
-0.01
-0.015
-0.02
-0.025
0
100
200
300
400
500
600
700
800
900
1000
No compression - x[n]



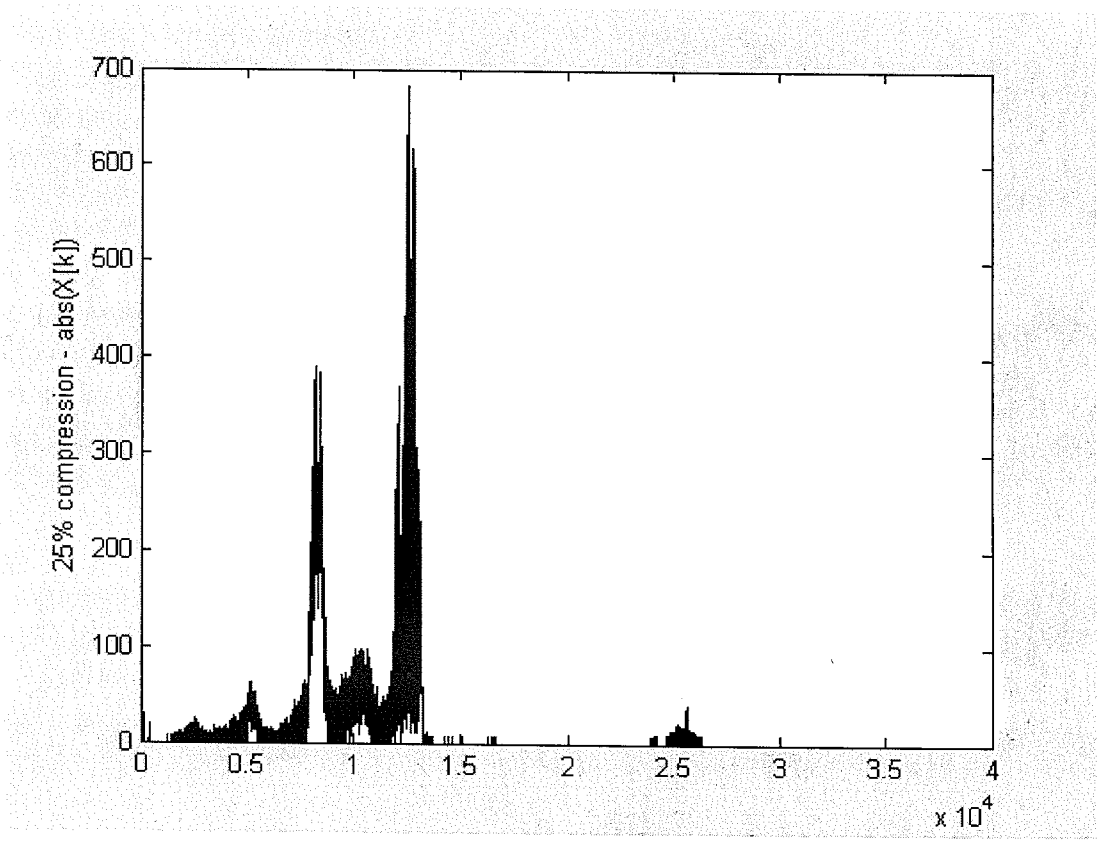


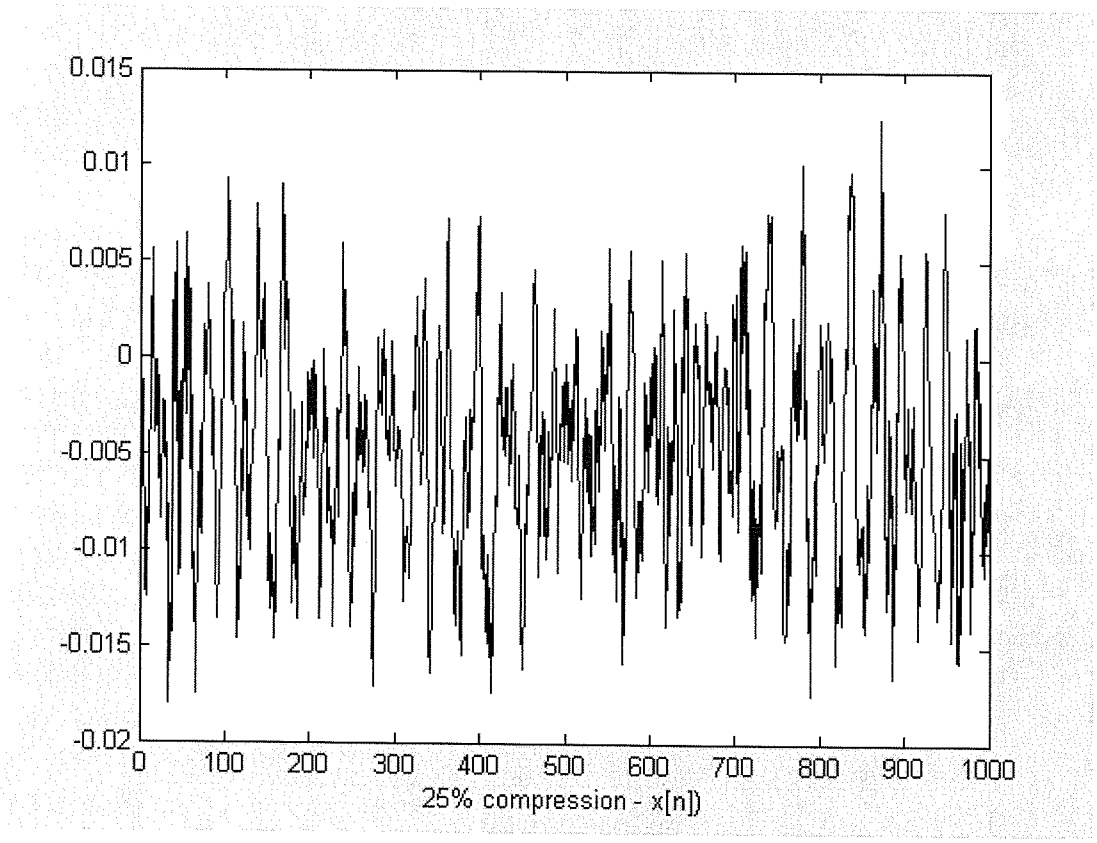
Problem 2b) - 25% compression

```
M = ceil(length(loon)/2*.75)
```

```
L = fft(loon);
Lposfreqs = L(1:ceil(length(loon)/2));
[y,ind] = sort(abs(Lposfreqs));
Lposfreqs(ind(1:M)) = 0;
Lcomp = [Lposfreqs; flipud(conj(Lposfreqs(2:ceil(length(loon)/2))))];

looncomp = ifft(Lcomp);

figure
plot(abs(Lcomp(1:length(Lcomp)/2)))
ylabel('25% compression - abs(X[k])')
figure
plot(looncomp(1:1000))
xlabel('25% compression - x[n])')
sound(looncomp,fs)
```

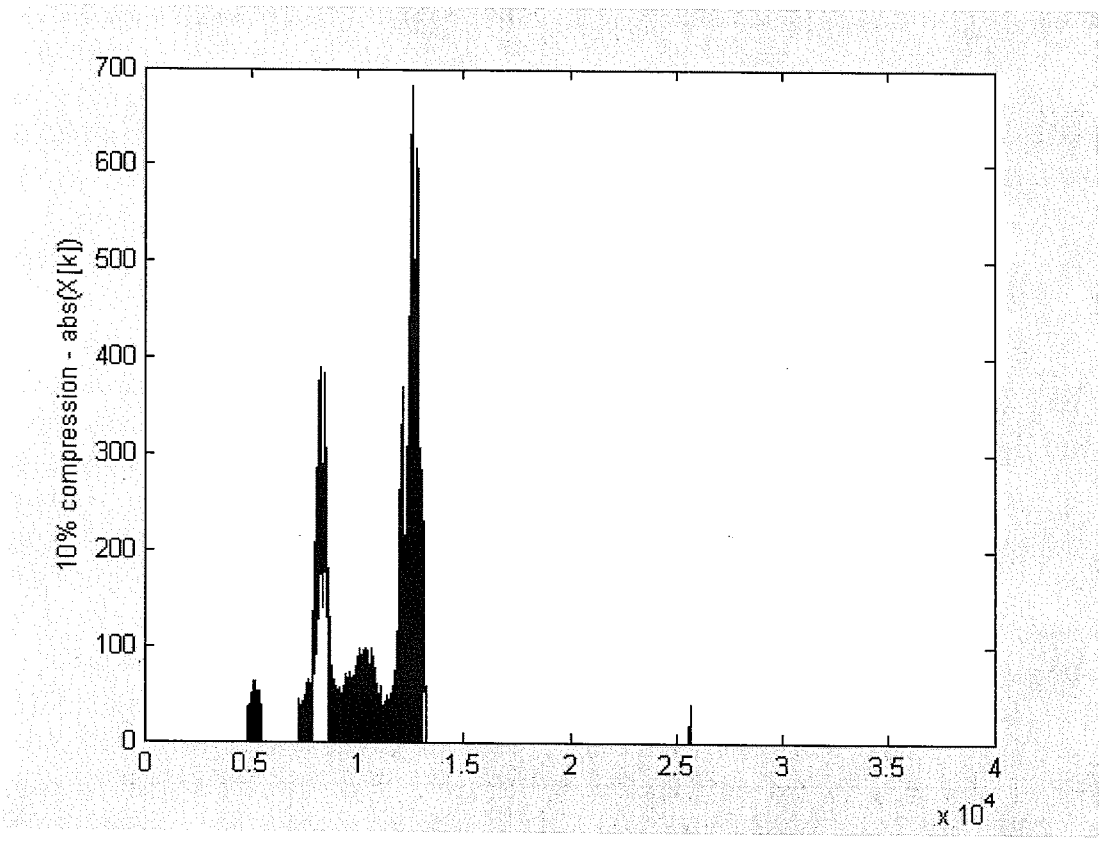
```
M =

       30001

Warning: Integer operands are required for colon operator when used as index
```

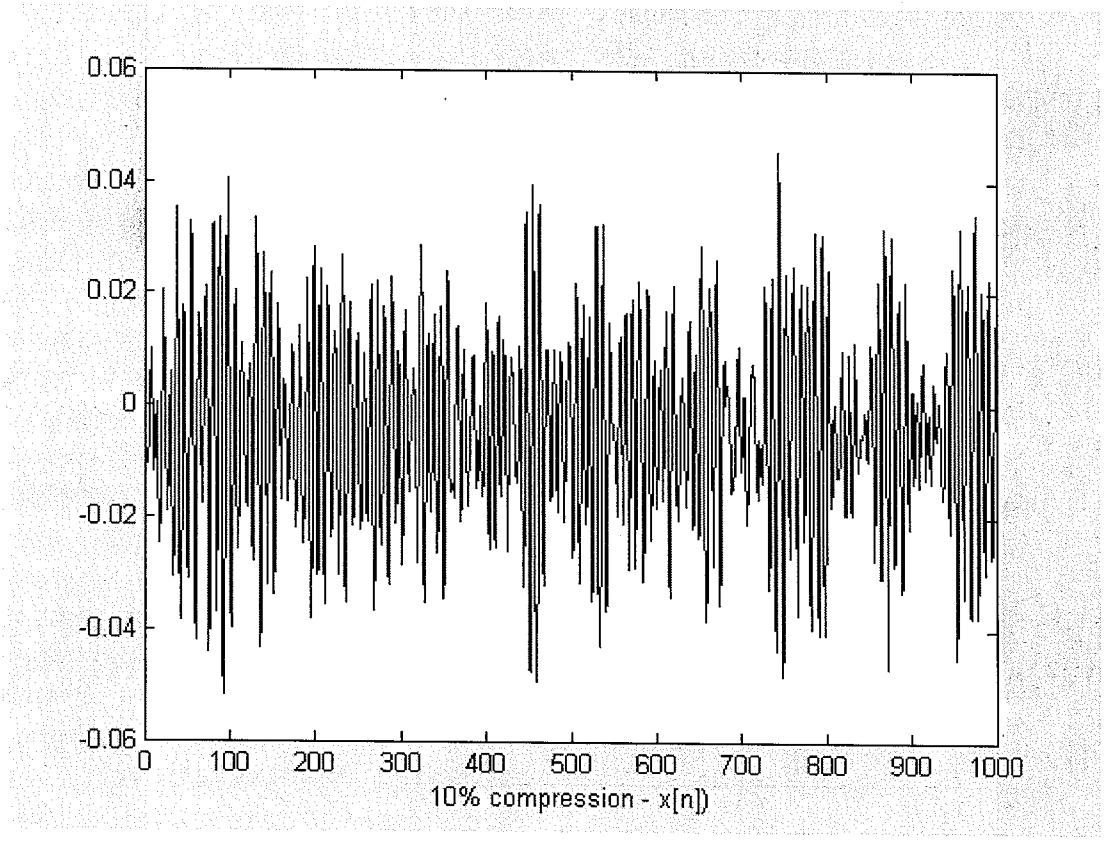




Problem 2c) - 10% compression

```
M = ceil(length(loon)/2*.90)
L = fft(loon);
Lposfreqs = L(1:ceil(length(loon)/2));
[y,ind] = sort(abs(Lposfreqs));
Lposfreqs(ind(1:M)) = 0;
Lcomp = [Lposfreqs; flipud(conj(Lposfreqs(2:ceil(length(loon)/2))))];

looncomp = ifft(Lcomp);

figure
plot(abs(Lcomp(1:length(Lcomp)/2)))
ylabel('10% compression - abs(X[k])')
figure
plot(looncomp(1:1000))
xlabel('10% compression - x[n])')
sound(looncomp,fs)


M =

       36001

Warning: Integer operands are required for colon operator when used as index
```

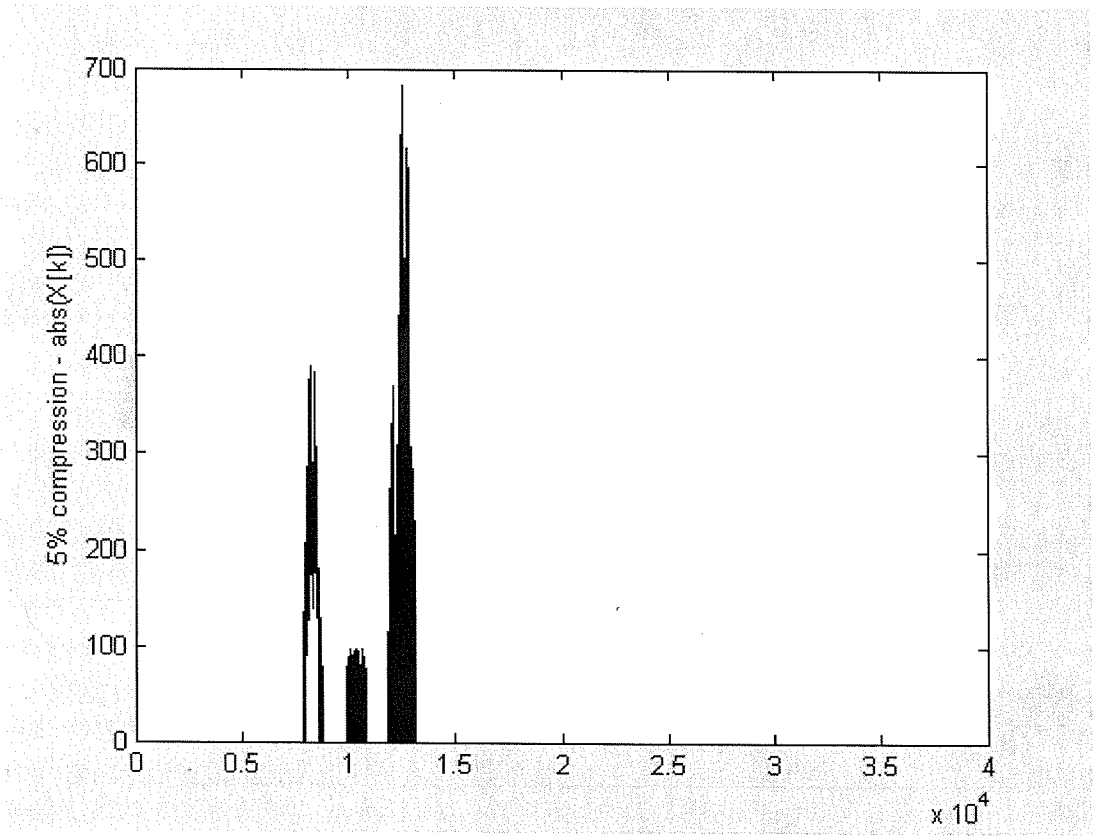


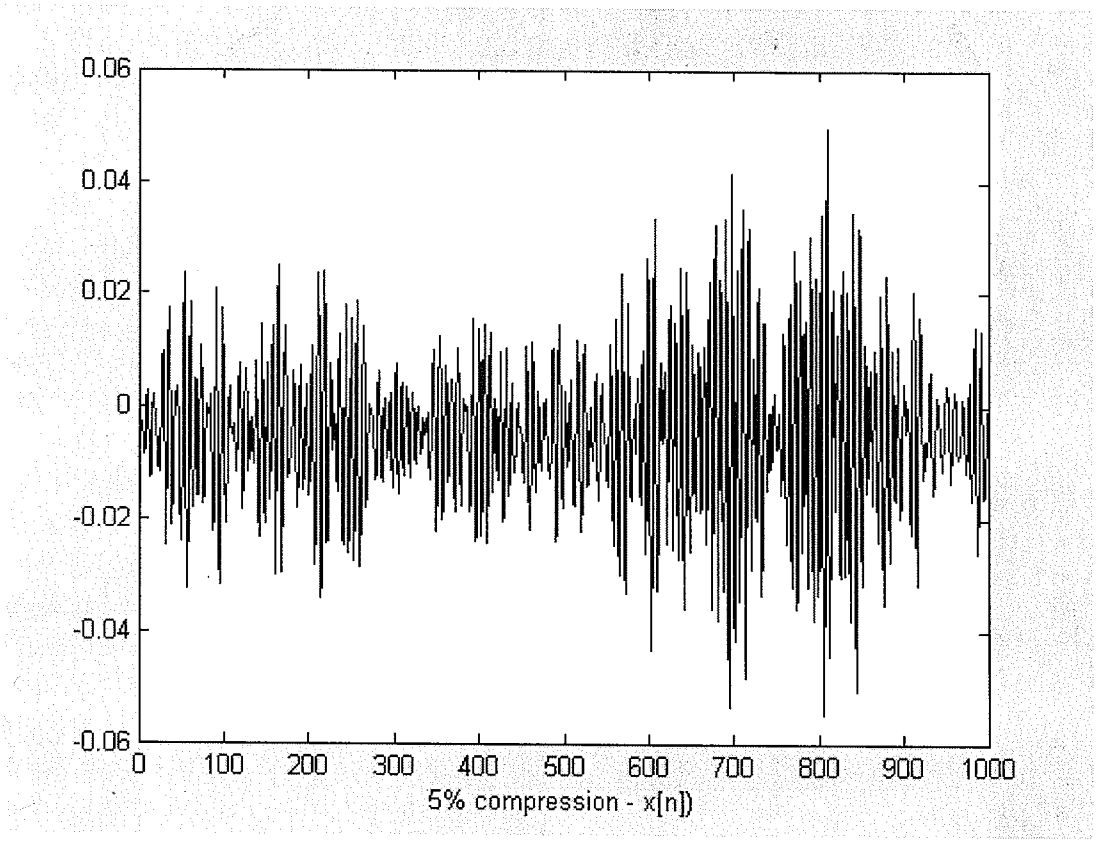


Problem 2d) - 5% compression

```
M = ceil(length(loon)/2*.95)
```

```
L = fft(loon);
Lposfreqs = L(1:ceil(length(loon)/2));
[y,ind] = sort(abs(Lposfreqs));
Lposfreqs(ind(1:M)) = 0;
Lcomp = [Lposfreqs; flipud(conj(Lposfreqs(2:ceil(length(loon)/2))))];

looncomp = ifft(Lcomp);

figure
plot(abs(Lcomp(1:length(Lcomp)/2)))
ylabel('5% compression - abs(X[k])')
figure
plot(looncomp(1:1000))
xlabel('5% compression - x[n]')
sound(looncomp,fs)


M =

       38001

Warning: Integer operands are required for colon operator when used as index
```





Published with MATLAB® 7.6

7.3

$$f * h = \sum_{k_1} \sum_{k_2} f(k_1, k_2)\, b(n_1 - k_1,\ n_2 - k_2)$$

$$h(n_1 - k_1, n_2 - k_2) = \begin{bmatrix} 1 & 1 \\ -1 & -1 \end{bmatrix}$$

$f * h$

$$\begin{bmatrix} -1 & -2 & -2 & -2 & -2 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 2 & 2 & 2 & 2 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

This filter highlights horizontal changes.

Contents

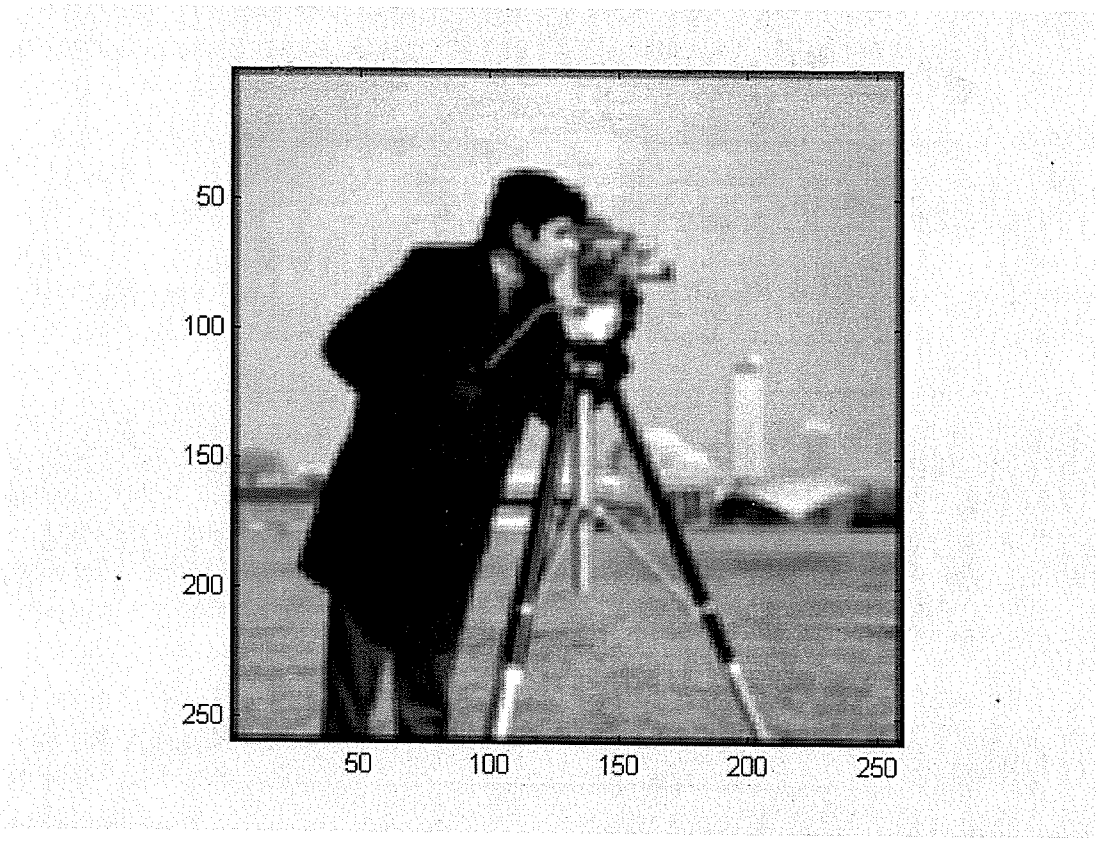
Homework 7 Problem 4

```
clc
clear
close all

load camera.mat

h1 = ones(4,4)/4;
h2 = ones(5,5)/25;
h3 = [1 2 1; 2 4 2; 1 2 1]/16;
h4 = [ 1 1; -1 -1];
h5 = [1 -1;1 -1];
h6 = [1 -2 1; -2 4 -2; 1 -2 1];
```

Problem 4a)

This filter averages surrounding pixels, which blurrs or dulls the image

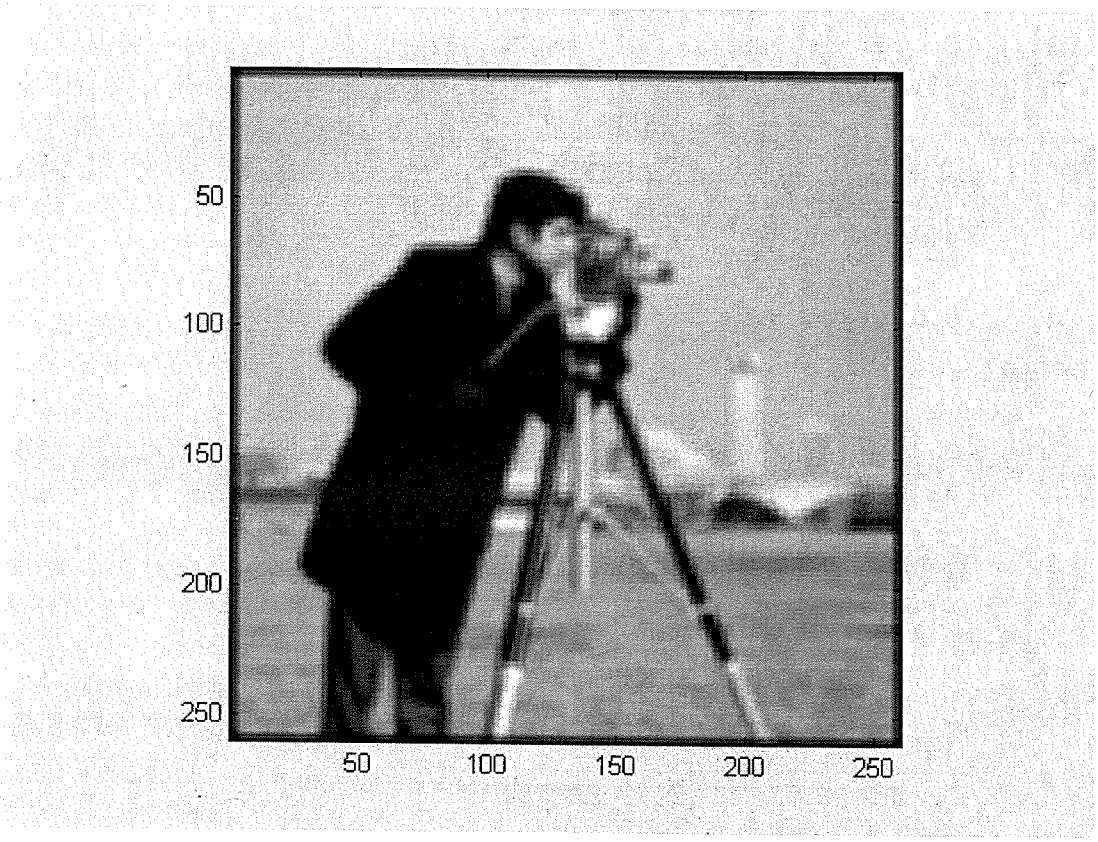
```
figure
imagesc(conv2(h1,im))
colormap(gray)
axis('square')
```



Problem 4b)

This filter also averages surrounding pixels, which blurrs or dulls the image

```
figure
imagesc(conv2(h2,im))
colormap(gray)
axis('square')
```

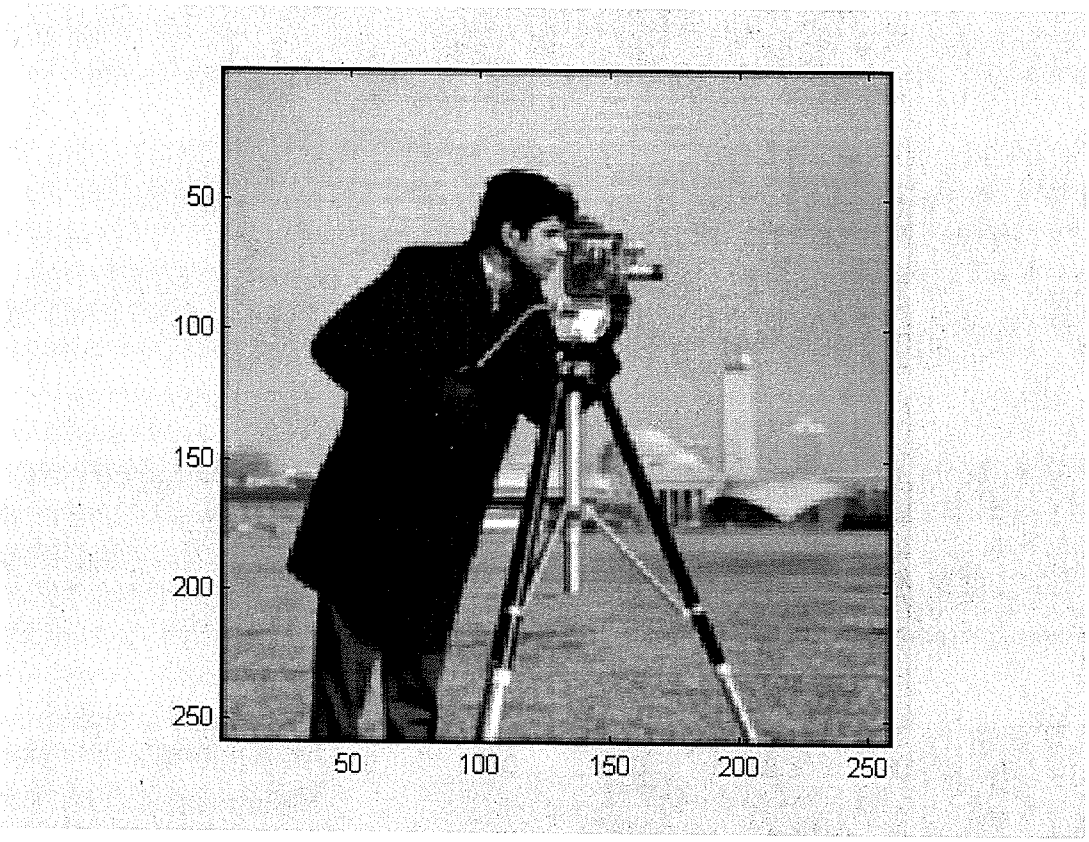


Problem 4c)

This filter is the same as applying h1 two times

$$h_3 = h_1 * h_1$$

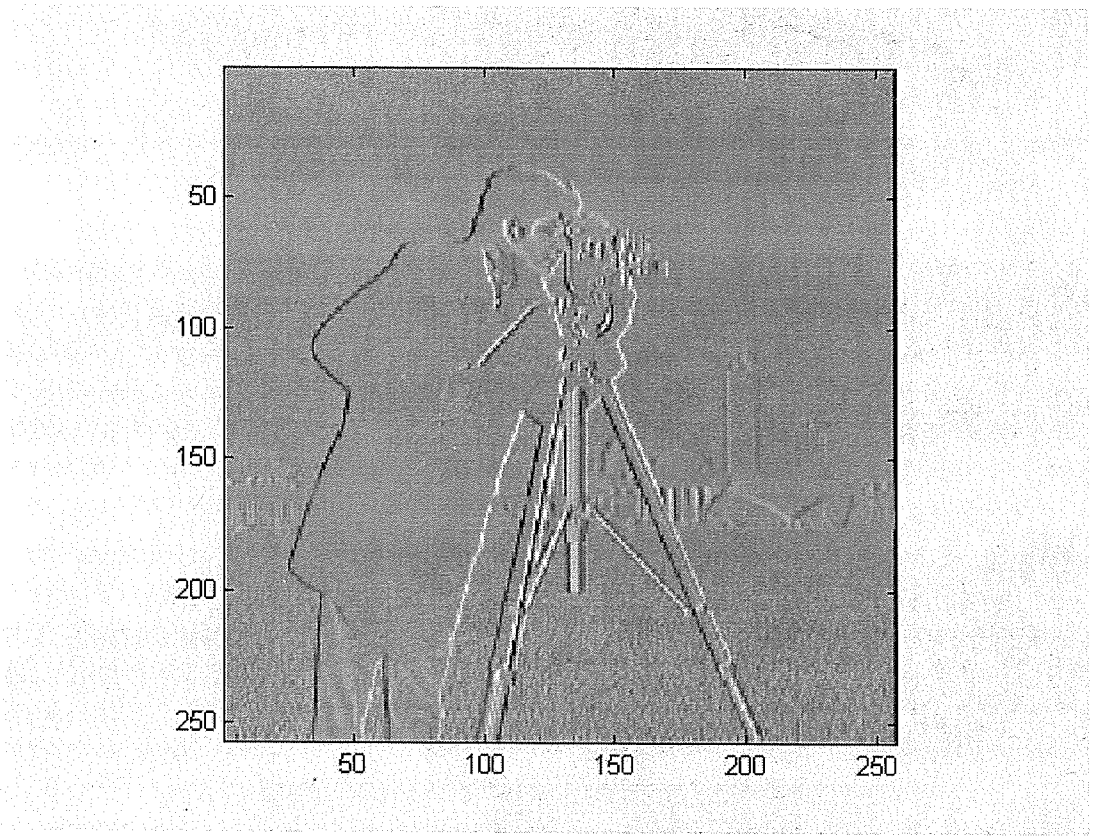
```
figure
imagesc(conv2(h3,im))
colormap(gray)
axis('square')
```



Problem 4d)

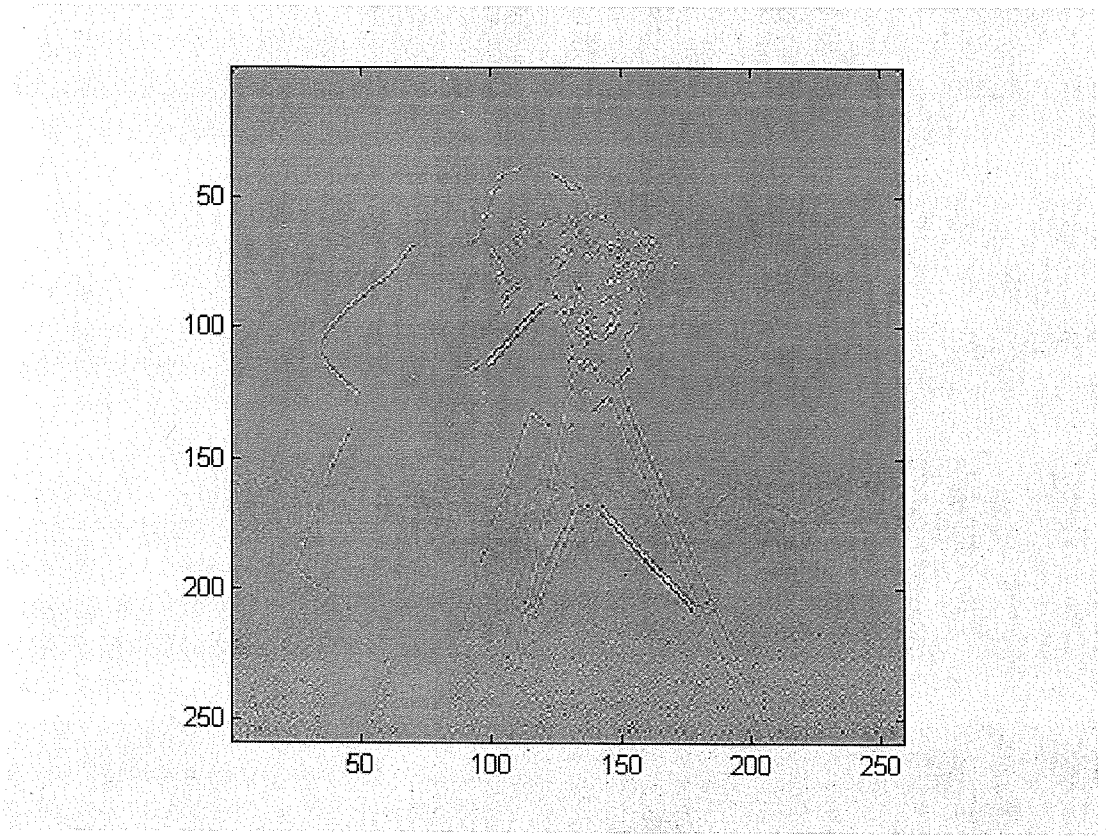
This filter highlights horizontal lines or changes

```
figure
imagesc(conv2(h4,im))
colormap(gray)
axis('square')
```



Problem 4f)

```
figure
imagesc(conv2(h6,im))
colormap(gray)
axis('square')
```



Published with MATLAB® 7.6