Hanfei Geng

hgeng4

Nov 4th, 2016

Report Item 1:

code:

x = [1,4,-4,-3,2,5,-6];

h = [1,2,3,4,5];

H = convmtx(h,length(x));

y = x\*H;

from\_conv = conv(x,h);

y == from\_conv

figure(1)

imagesc(H)

axis('xy')

title('convolution matrix')

figure(2)

stem(y,'r');

title('y')

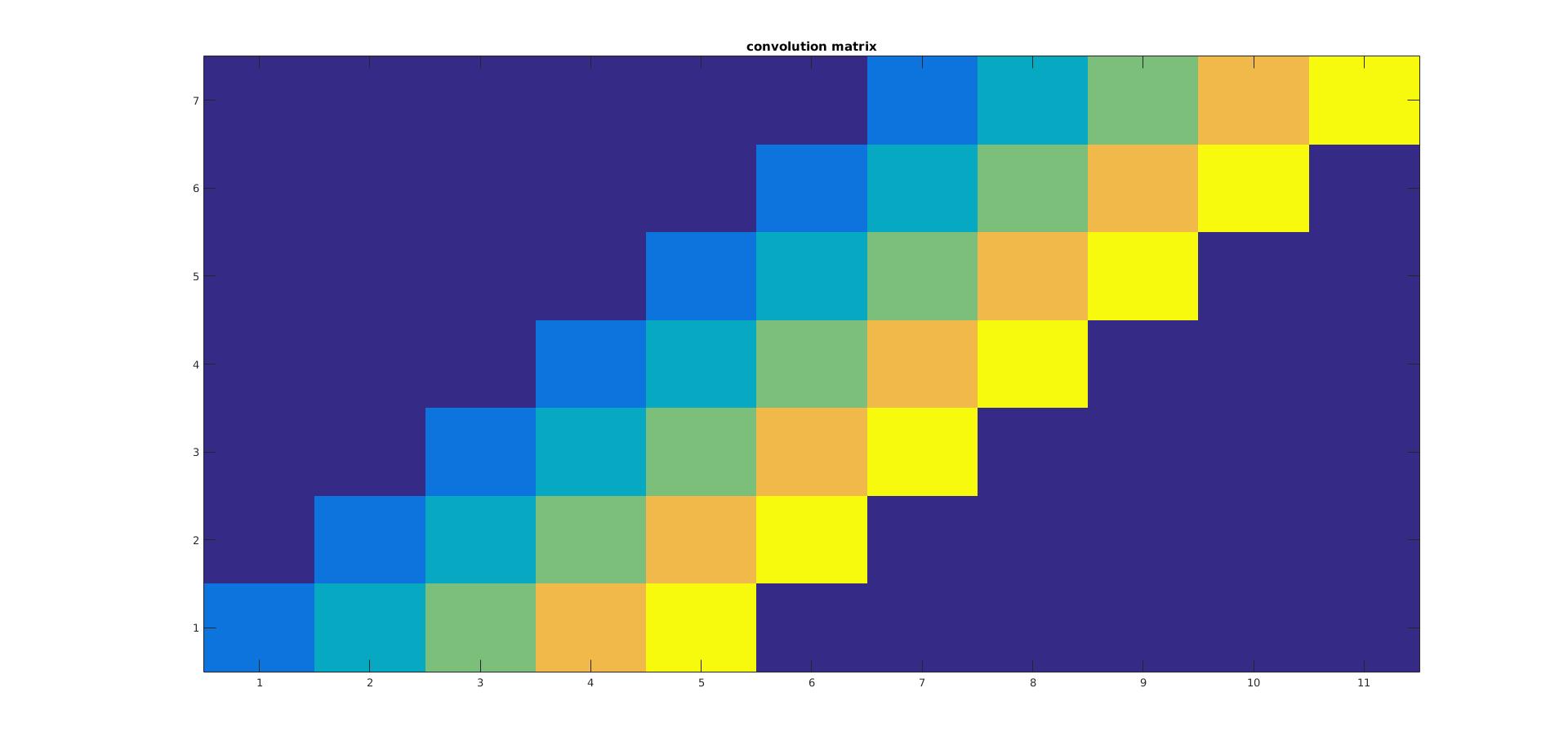
xlabel('n')

ylabel('y')

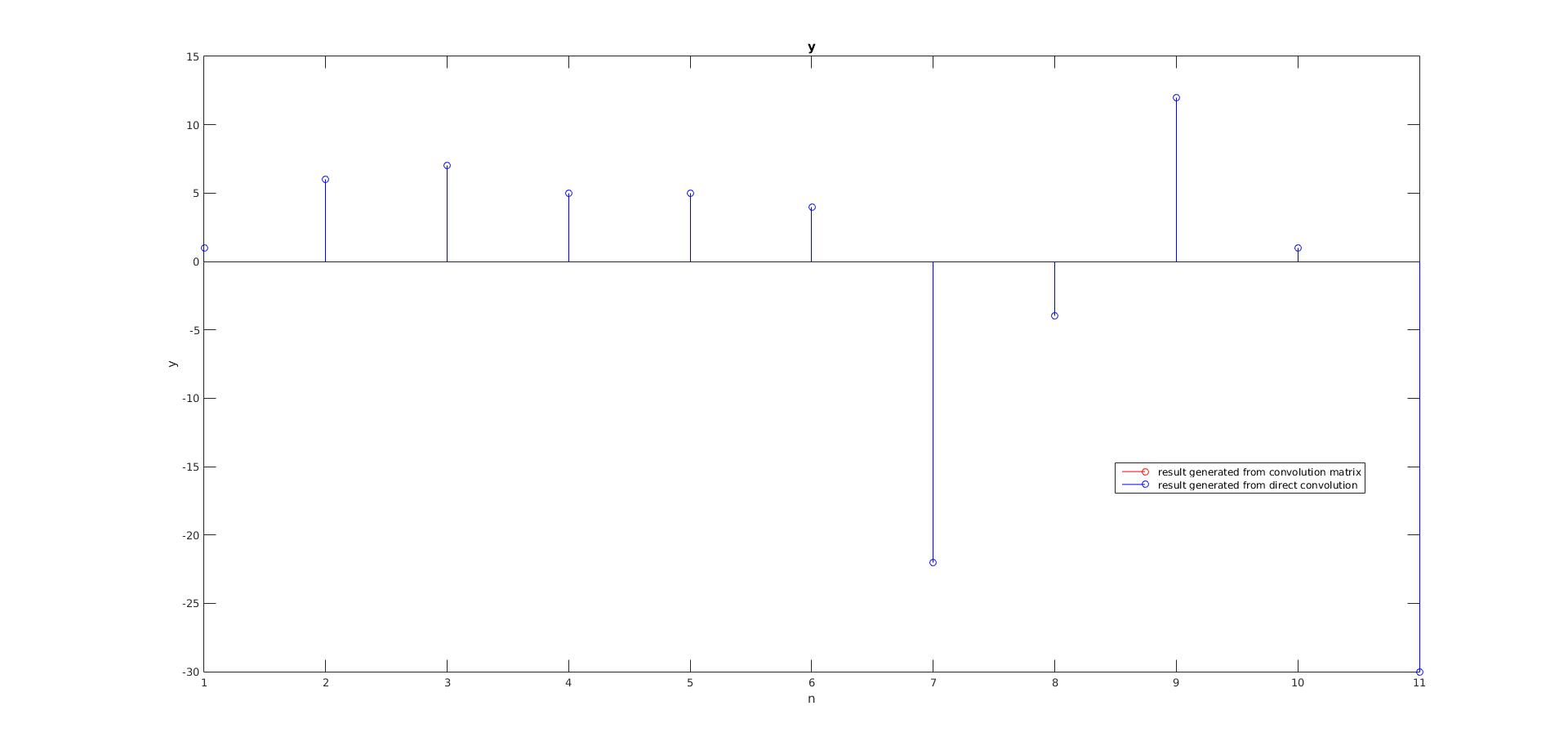
hold on

stem(from\_conv,'b')

legend('result generated from convolution matrix','result generated from direct convolution')



The structure of the convolution matrix is diagonal.



Report Item 2:

AHA = VΣHUHUΣVH

AHA = VΣHΣVH

AHAv = VΣHΣ

Thus AHA is the eigenvector of V.

Report Item 3:

code:

A = [1,4,-2;3,11,5;7,7,7];

[V1,D1] = eig(A\*(A'));

[V2,D2] = eig((A')\*A);

[U,S,V] = svd(A);

It is verified that verified that U = V1 and V = V2

Report Item 4:

code:

x = [1,1,4,-4,-3,2,5,-6,3,2,4,-2,5,9,-8,4];

F = dftmtx(length(x))

subplot(121)

imagesc(real(F))

axis('xy')

title('real part of F')

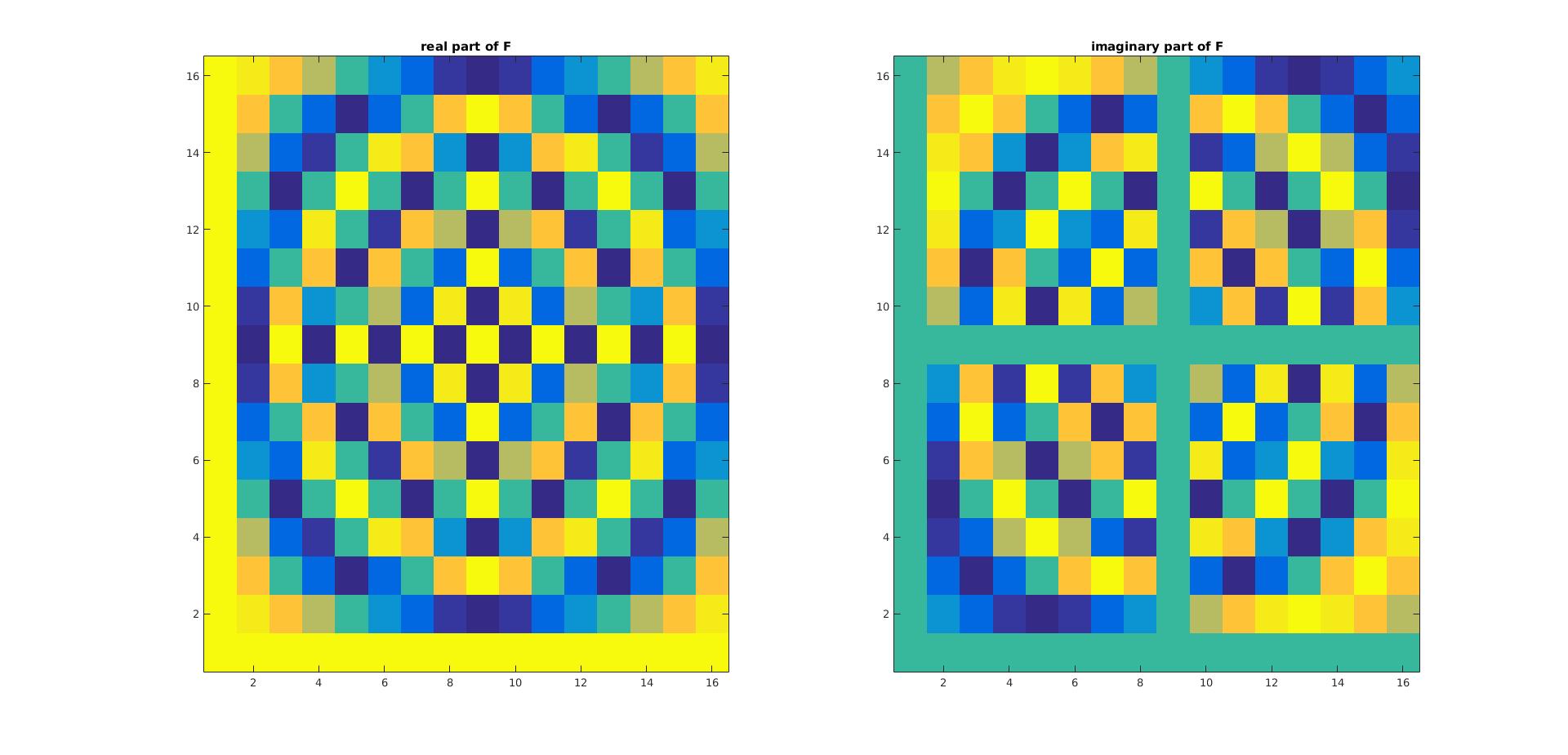
subplot(122)

imagesc(imag(F))

axis('xy')

title('imaginary part of F')

Figure:



The dot product of DFT coefficient and x results in the DFT of x.

In each row, the color seems to alternate with different frequency. At higher row number, The frequency of the color changing seems to increase and decrease after certain row.

Report Item 5:

x = [1,1,4,-4,-3,2,5,-6,3,2,4,-2,5,9,-8,4];

F = dftmtx(length(x))

inverse\_F = 1/length(x) \* F'

A = inverse\_F' \* inverse\_F

The columns of this matrix are orthogonal to each other.

Report Item 6:

Code:

X = loadImages('yalefaces');

mean = computeMeanVec(X)

reshape\_mean = reshape(mean,[60,80])

colormap gray

imagesc(reshape\_mean)

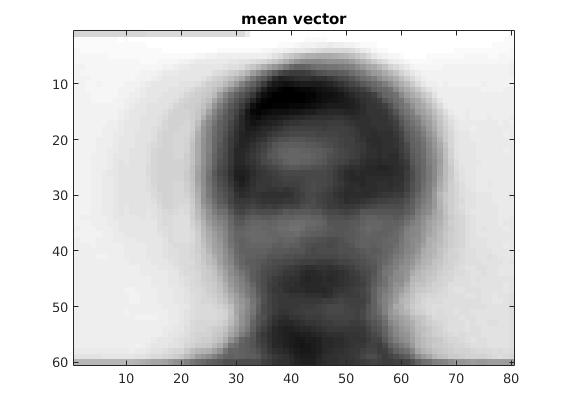
title('mean vector')

function mean = computeMeanVec(x)

[M,N] = size(x);

mean = 1/M \* sum(x)

end



We see a blurry figure.

Report Item 7:

[row,col] = size(X);

X\_delta = zeros(row,col);

for i = 1:row

X\_delta(i,:) = X(i,:) - mean; %mean vector from last report item

end

R = X\_delta' \* X\_delta;

[U,S,V] = svd(R);

%eigen values

eigen = diag(S);

eigen\_values = eigen(1:100);

%eigen\_vector

eigen\_vectors = U(:,1:100);

figure(1)

plot(eigen\_values)

title('eigenvalues')

xlabel('index')

ylabel('eigenvalues')

figure(2)

first\_4 = U(:,1:4);

reshape\_first\_4 = reshape(first\_4,[60,320]);

colormap gray

imagesc(reshape\_first\_4)

title('reshaped first 4 eigenvectors')

fif\_hund = zeros(4800,2);

fif\_hund(:,1) = eigen\_vectors(:,50);

fif\_hund(:,2) = eigen\_vectors(:,100);

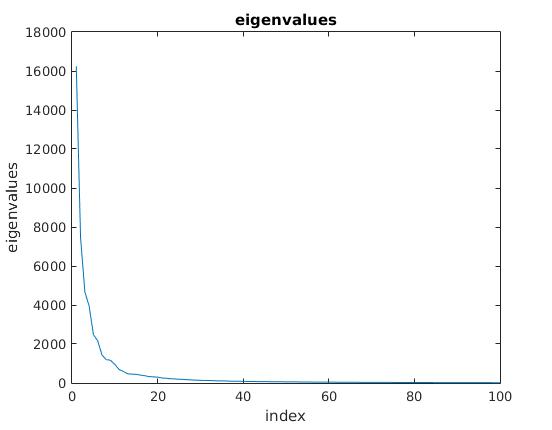
reshape\_fif\_hund = reshape(fif\_hund,[60,160]);

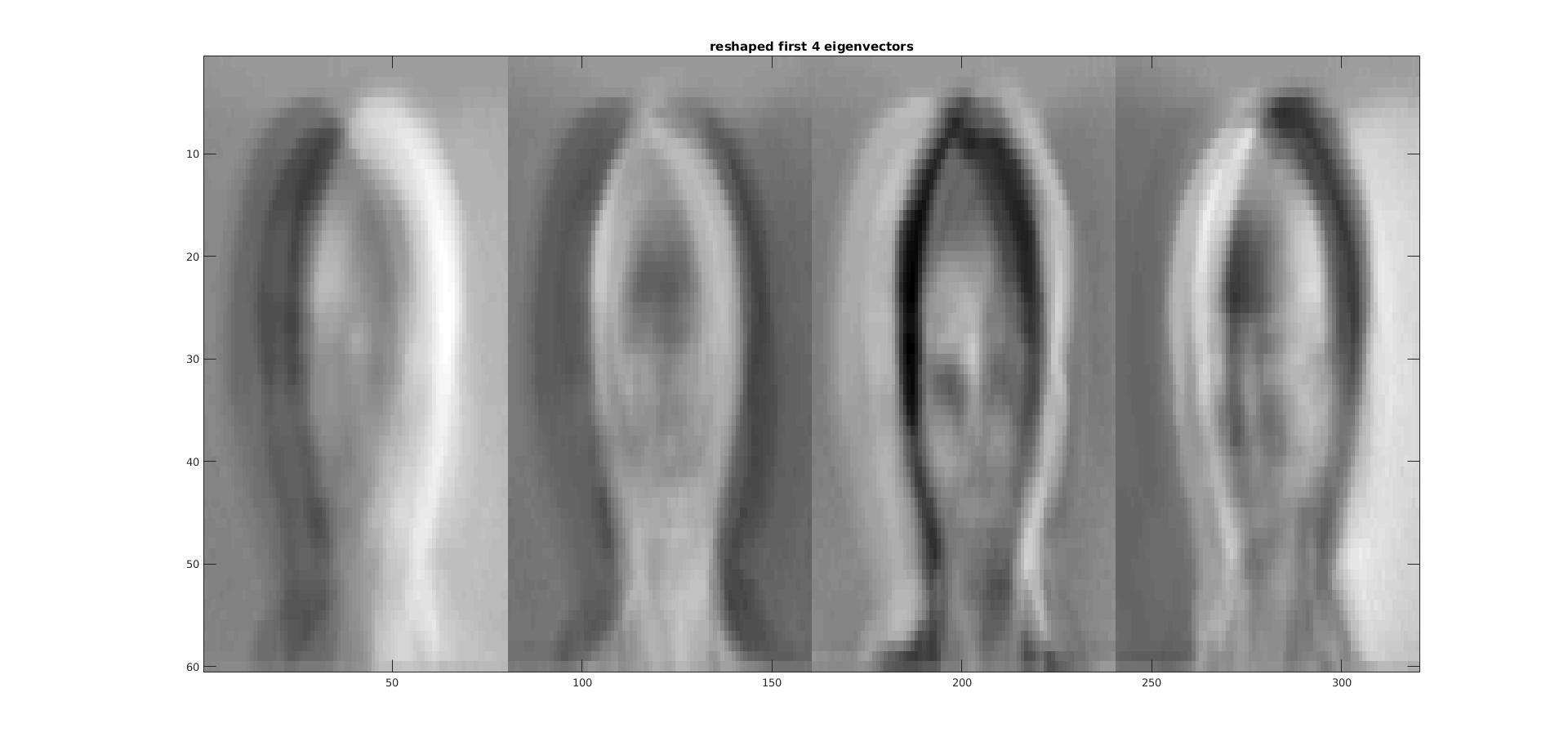
figure(3)

colormap gray

imagesc(reshape\_fif\_hund)

title('reshaped 50th and 100th eigenvectors')







The first 20 eigenvalues are much greater than the rest of the eigenvalues.

We can see 4 almost clear figures from the first 4 eigenvectors.

We see two figures from the 50th and 100th eigenvecto, but they are much more blurry than images from first 4 eigenvectors. For n >= 50, the eigenvectors should have really low magnitude, which almost equal 0.

Report Item 8:

function pca\_basis = PCA\_transform(mean,V,x\_orig)

x\_orig\_til = x\_orig-mean;

pca\_basis = V' \* x\_orig\_til';

end

function orig = invPCAtransform(mean,V,pca)

orig = V \* pca + mean';

end

Report Item 9:

image = imread('noisy\_face.tiff');

mean = (computeMeanVec(X));

x\_orig = reshape(image,[1,4800]);

pca = PCA\_transform(mean,U(:,1:150),double(x\_orig));

new = invPCAtransform(mean,U(:,1:150),pca);

imagesc(reshape(pca,[5,30]))

colormap gray

title('new basis')

figure(2)

colormap gray

imagesc(reshape(new,[60,80]))

title('original basis')

