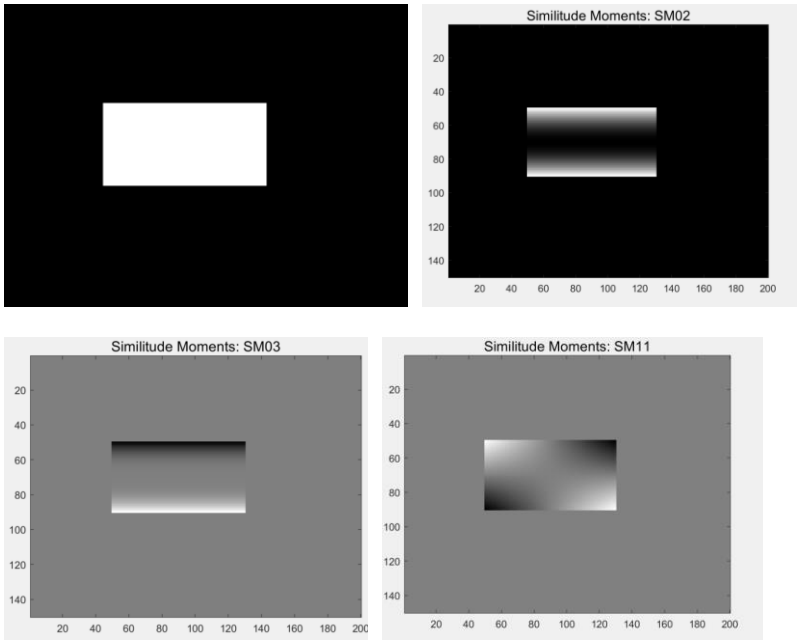
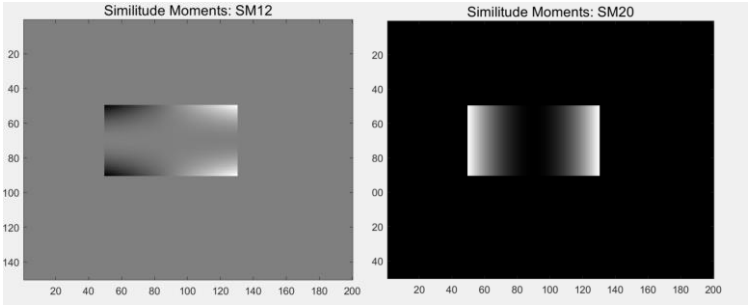
Yukun Duan
Computer Vision for HCI AU'22
Homework Assignment #4

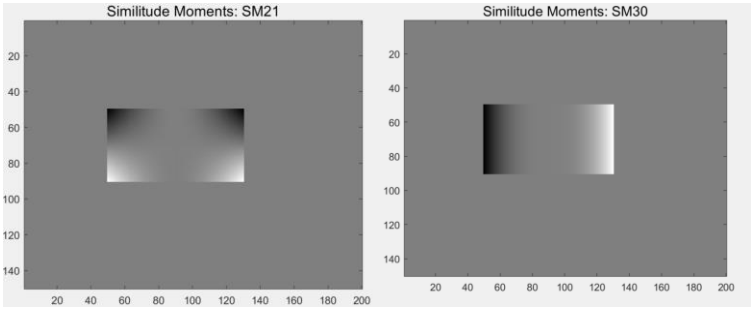
1.
Here is the result I get for similitude momentum:

IMG\SM	02	03	11	12	20	21	30
boxIm1.bmp	0.1653	0	0	0	0.6455	0	0
boxIm2.bmp	0.1653	0	0	0	0.6455	0	0
boxIm3.bmp	0.1658	0	0	0	0.6435	0	0
boxIm4.bmp	0.6455	0	0	0	0.1653	0	0

The result implied that image 1-3 has the same similitude on all momentum, which can be because they look similar in shape. Image 2 has the same size as image 1, but it's in a different location. Image 3 has the same shape as image 2, but has a different size. From the result above, we can see that similitude moment is invariant to scaled and translated images. Image 4 has the same shape and size as image 1 but it is rotated by 90 degrees. Thus, the momentum 02 and 20 are switched since x, and y is switched.
Here is the origin image of boxIm1.bmp vs. similitude moment (before sum):



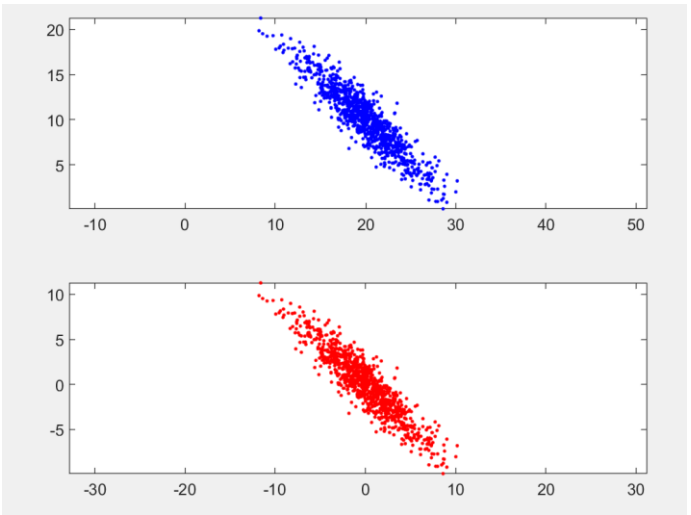




0 represents pixels that are outside the outline (rectangle) of the image.

2.

Here is the image I get:

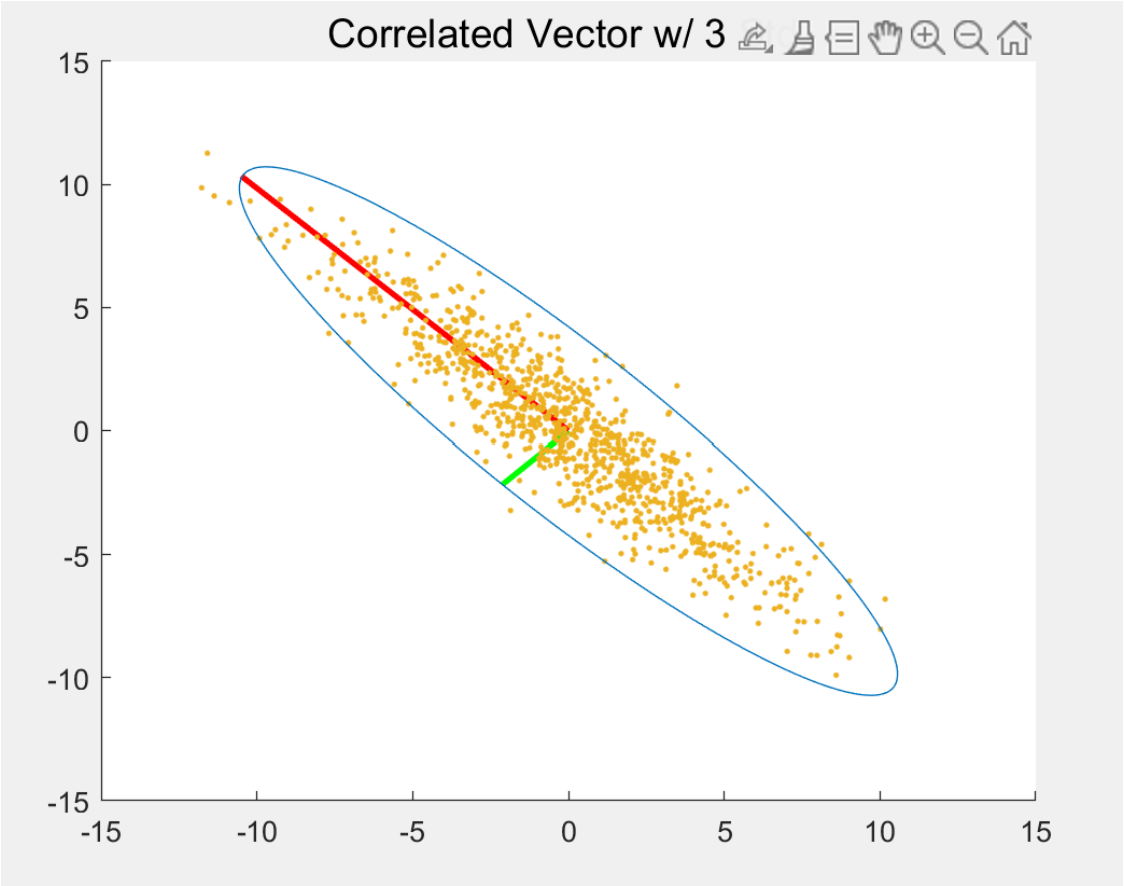


3.

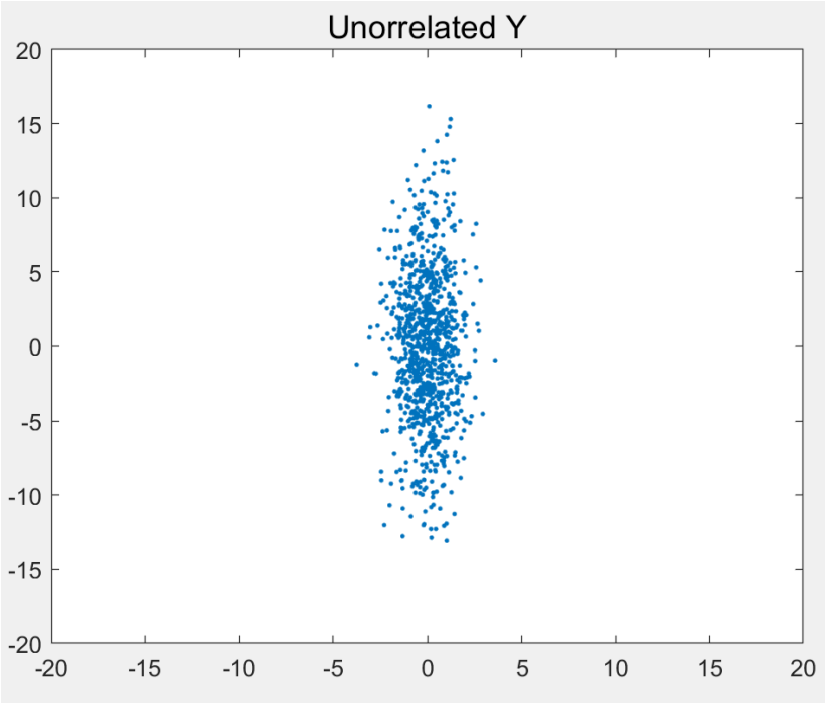
The eigenvalue and eigenvector are calculated using built-in function. MATLAB sorts the eigenvalue in ascending order, so it looks different from the example given on the note. The eigenvectors point to the axes of the eclipse, and the length of vector is 3σ.

Result: U= [-0.7016, -0.7126; -0.7126, 0.7016],

V = [1.0142, 0; 0, 24.1385].
Here is the plot I drew:

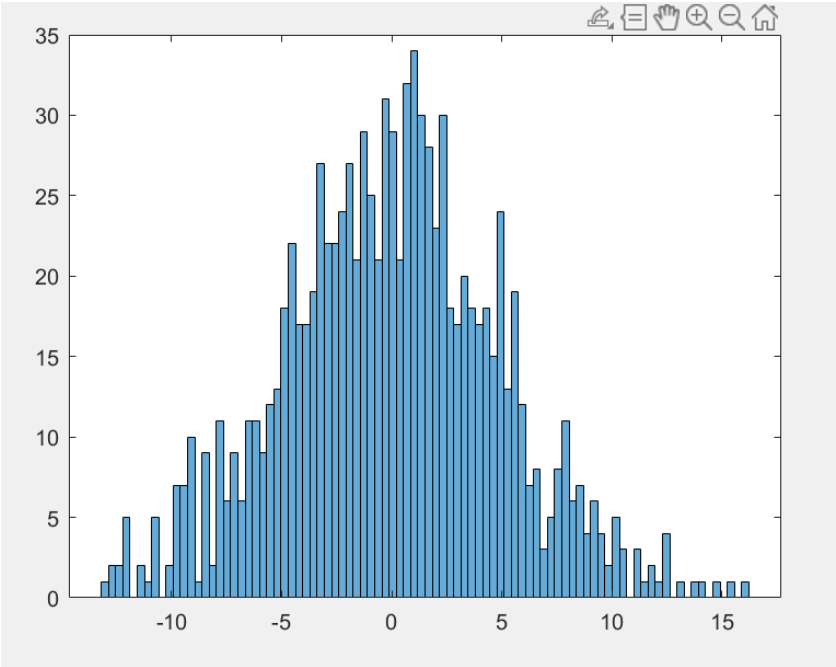


4.
The plot is shown below. Dots are projected by multiplying the eigenvector. The resulting Y are decorrelate. Obviously, the dots are aligned across the Y axis, and the eclipse is rotated:



5.

This is the 1-D histogram I plot:



It does look like 1-D Gaussian.

Attached Code:

```
% Yukun Duan
% CSE5524 - HW4
% 9/17/2022


%% Problem 1
for i = 1:4
    % read image (in double): boxIm[1-4].bmp
    boxIm= double(imread(sprintf('boxIm%d.bmp',i)));
    % display 7 similituyde moment for each of image
    disp(similitudeMoments(boxIm));
end


%% Problem 2

% Load the data
clear; close all;
load 'eigdata.txt';
```

```
X = eigdata;
subplot(2,1,1);
plot(X(:,1),X(:,2),'b.');
axis('equal');
% mean-subtract data
m = mean(X);
Y = X - ones(size(X,1),1)*m;
subplot(2,1,2);
plot(Y(:,1),Y(:,2),'r.');
axis('equal');
pause;


%% Problem 3
close all;
K = cov(Y);
[coeff,score,latent,~,explained] = pca(Y);
[U, V] = eig(K);
c = 9;
len1 = sqrt(c*V(1,1));
len2 = sqrt(c*V(2,2));
hold on
plot([U(1,1),0]*len1,[U(2,1),0]*len1, 'g','LineWidth',2)
plot([U(1,2),0]*len2,[U(2,2),0]*len2, 'r','LineWidth',2)
plot(Y(:,1),Y(:,2),'.')
title('Correlated Vector w/ 3 Std.','FontSize', 14)
ellipse(len1,len2,atan(U(1,1)/U(2,1)),0,0)
pause;




%% Problem 4
close all;
Y2 = Y * U';
plot(Y2(:,1),Y2(:,2),'.')
axis([-20 20 -20 20])
title('Unorrelated Y','FontSize', 14);
pause;



%% Problem 5
Y3=Y2(:,2);
```

```
histogram(Y3,90);
pause;



%% Problem 1--Similitude Function

function Nvals = similitudeMoments(boxIm)
    Nvals = [];
    % initialize matrix for row index, col index, x average and y average.
    xIndex = repmat(1:size(boxIm,2),size(boxIm,1),1); % col => x
    yIndex = repmat((1:size(boxIm,1))', 1, size(boxIm,2)); % row => y
    m00 = sum(boxIm, 'all');
    m10 = sum(xIndex.*boxIm, 'all');
    m01 = sum(yIndex.*boxIm, 'all');
    xbar = ones(size(boxIm)) * m10/m00;
    ybar = ones(size(boxIm)) * m01/m00;
    % iteratively calculate 7 similitude moments
    for i = 0:3
        for j = max(0,(2-i)):(3-i)
            % 2 <= (i+j) <= 3
            nij = sum(((xIndex - xbar).^i).*((yIndex - ybar).^j).*boxIm,
'all')/(m00.^((i+j)/2+1));
            img = ((xIndex - xbar).^i).*((yIndex-
ybar).^j).*boxIm/(m00.^((i+j)/2+1));
            imagesc(img)
            colormap('gray');
            title(sprintf('Similitude Moments: SM%d%d',i,j), 'FontSize',
14);
            pause;
            Nvals = [Nvals, nij];
        end
    end
end
```