Q4:

conv([1 3 3 1],[1 3 3 1],'full');

% a ans = [1 6 15 20 15 6 1]

conv2([1 3 3 1],[1 3 3 1]','full');

% b ans =

% 1 3 3 1

% 3 9 9 3

% 3 9 9 3

% 1 3 3 1

b involves the property of separability.

a involves the property of cascading.

Q5:

clear all; close all; clc;

image = double(imread('lena.tif'));

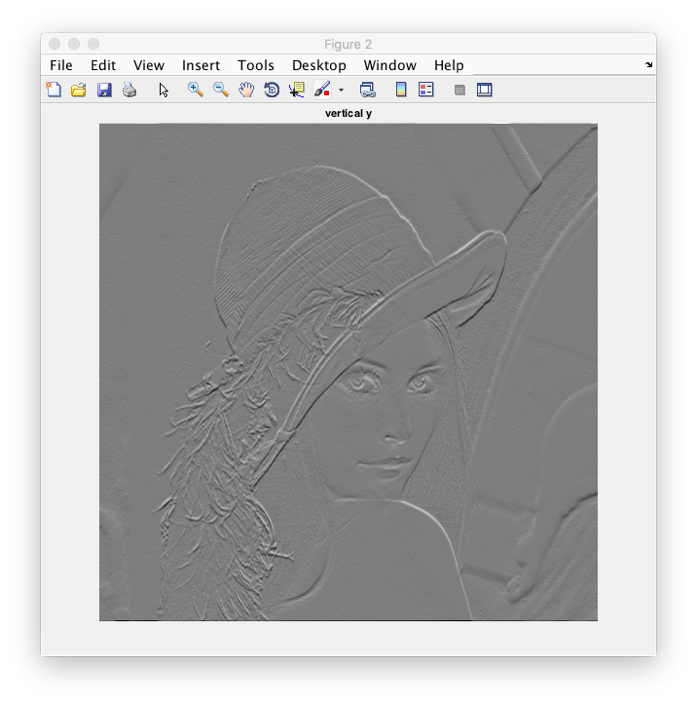
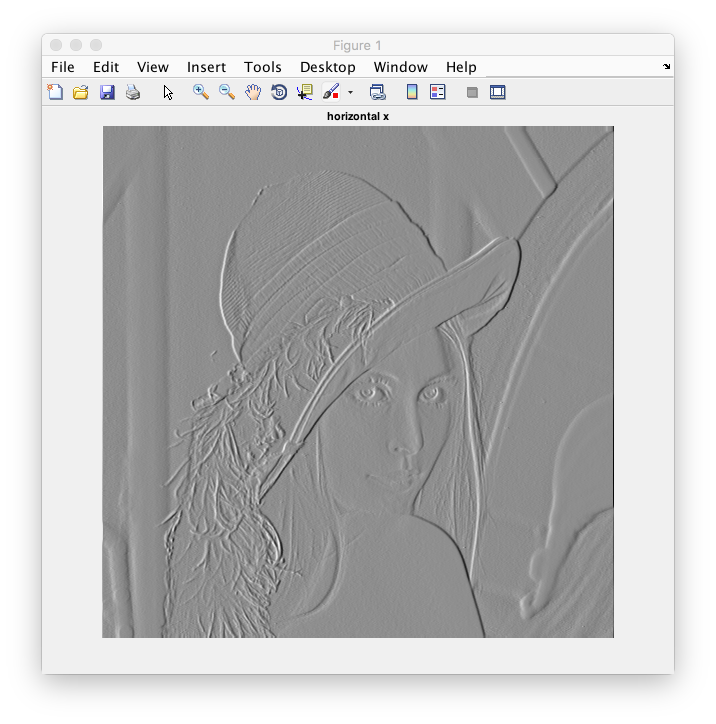
% (1). Use the Matlab function imfilter() to compute the horizontal (x) and vertical (y) derivatives on the image, without smoothing.

hx = imfilter(image,[1 0 -1],'conv','same');

figure; imshow(hx,[]); title('horizontal x');

vy = imfilter(image,[1 0 -1]','conv','same');

figure; imshow(vy,[]); title('vertical y');



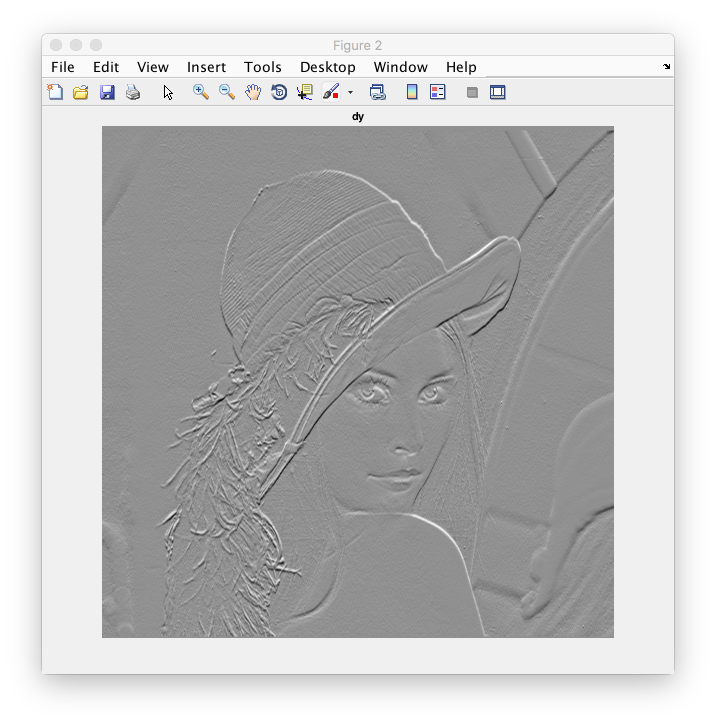
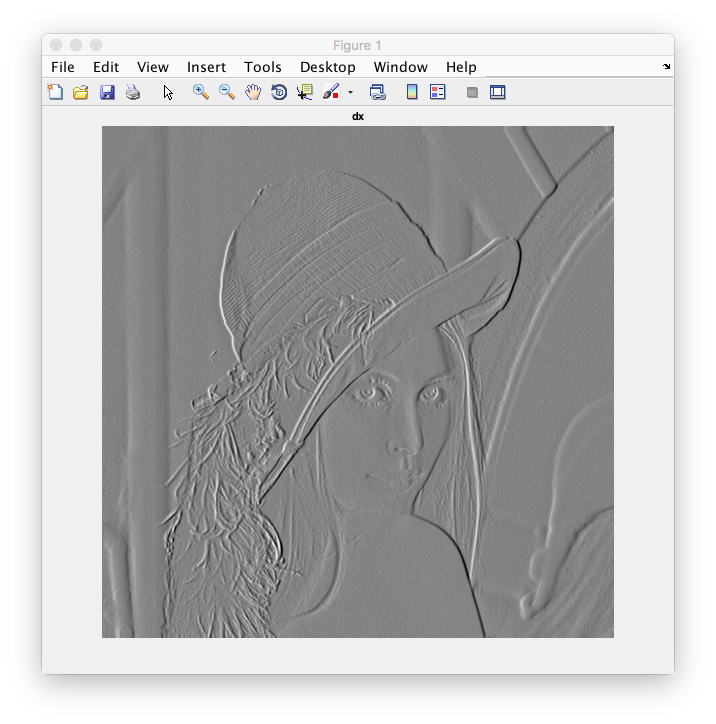
% (2). Compute the magnitude and orientation of the gradient.

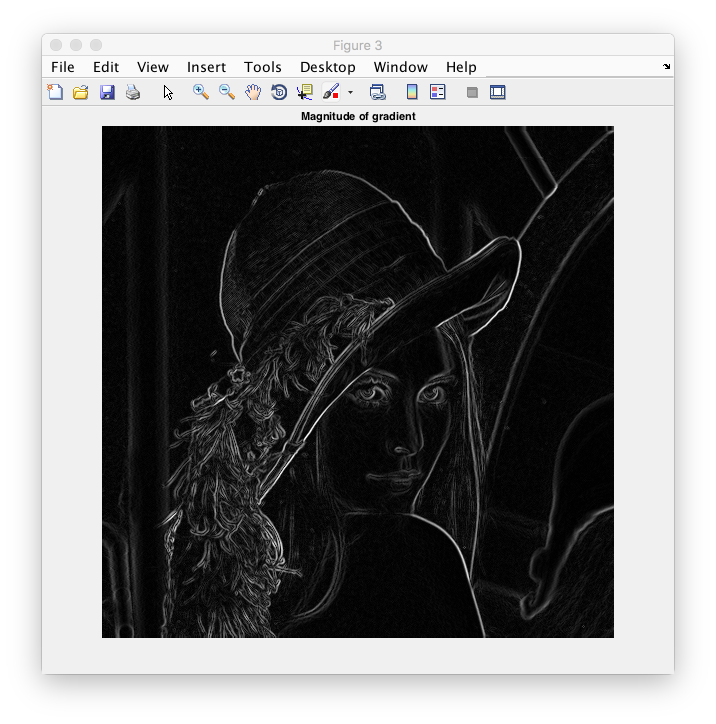
[dx dy] = gradient(image);

figure; imshow(dx,[]); title('dx');

figure; imshow(dy,[]); title('dy');

figure; imshow(sqrt(dx.^2+dy.^2),[]); title('Magnitude of gradient');





% (3). Crop a small image patch. Visualize the gradient vectors of the patch using the Matlab function quiver().

im = image;

figure; imshow(im,[]);

im = imcrop;

im = double(im)/255;

[dx dy] = gradient(im);

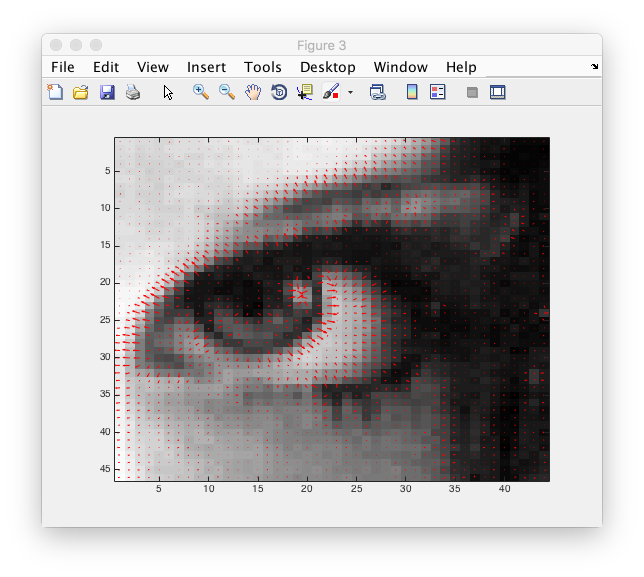
[tmpx, tmpy] = meshgrid(1:size(im,2),1:size(im,1));

figure; imagesc(im); colormap(gray); hold on;

% contour(im,'LineWidth',2,'Color','b');

quiver(tmpx,tmpy,dx,dy,'Color','r');

hold off;



% (4). Apply a Gaussian filter with sigma = 2 to the ?lena.tif? and then compute its gradient vector. Repeat the visualization in step 3.

im = image;

sigma = 2; halfwid = 3\*sigma;

[xx,yy] = meshgrid(-halfwid:halfwid,-halfwid:halfwid);

gau = exp(-(xx.^2+yy.^2)/(2\*sigma^2));

im = imfilter(im,gau/sum(gau(:)),'same',0);

figure; imshow(im,[]); title('Gaussian smoothing with sigma = 2');

im = imcrop;

im = double(im)/255;

[dx dy] = gradient(im);

[tmpx, tmpy] = meshgrid(1:size(im,2),1:size(im,1));

figure; imagesc(im); colormap(gray); hold on;

% contour(im,'LineWidth',2,'Color','b');

quiver(tmpx,tmpy,dx,dy,'Color','r');

hold off;

