The aim of question 1.1 was to implement a Gaussian filter in MATLAB with varying standard deviation sigma as user input. The code for this question is given below.

% Implementation of a Gaussian Filter.

%Reading Image.

I = imread('image1.tiff');

%Asking user for input of sigma.

sigma = input('Please Enter the value of sigma:');

%Assuming size of gaussain matrix as 15 by 15.

hsize = [15 15];

%Creating a two-dimensional Gaussian filter h.

h = fspecial('gaussian', hsize , sigma);

%Convolving the 2D image with the gaussian filter.

I2 = conv2(double(I),double(h));

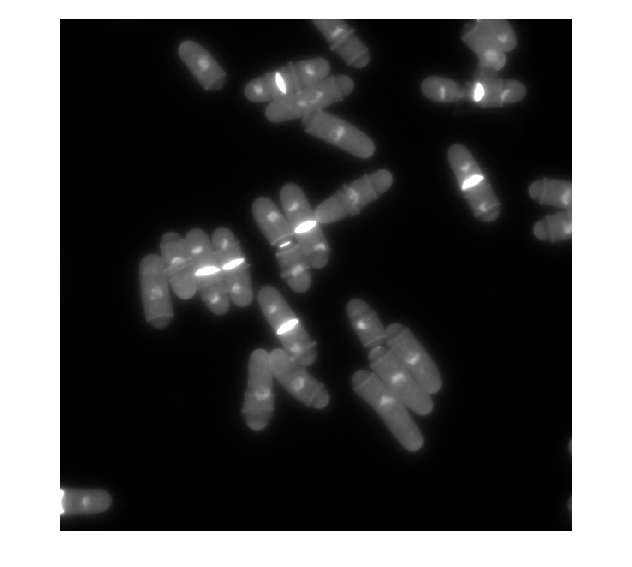
I2 = double(I2);

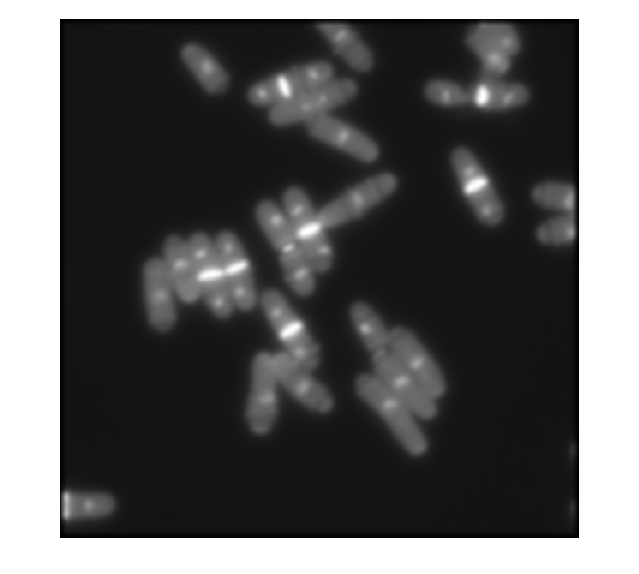
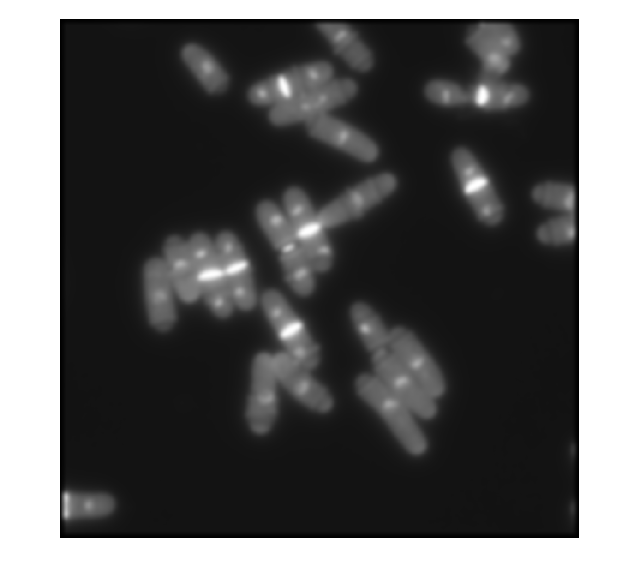
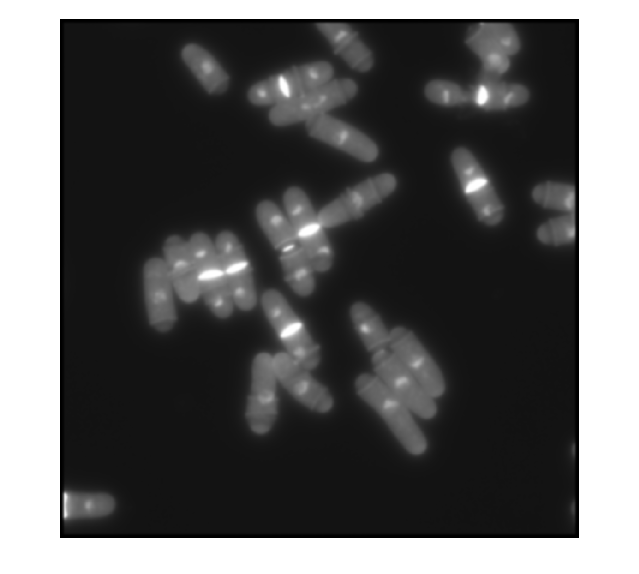
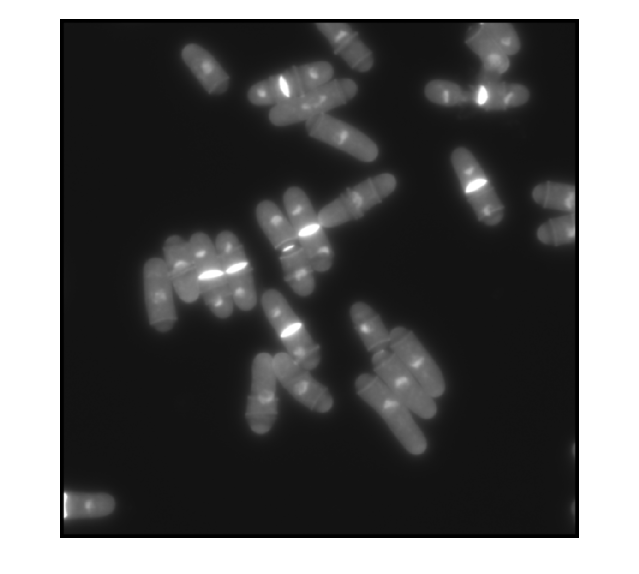
% Displaying the original image, then after getting sigma input, displaying

% filtered image.

figure, imshow(I,[])

figure, imshow(I2,[])

The original image is shown below:

The images below are the ones with sigma = 1,2,5,7 respectively.

As sigma increases, the images get blurrier. A Gaussian filter is essentially a low pass filter. Thus, it removes noise but also removes detail. Large sigma means a wider Gaussian filter that in turn means greater smoothening. Thus, as sigma goes from 1 to 7, a greater blur is seen as greater amount of detail is removed.

The aim of question 1.2 was to calculate image derivates. The original image used was the same as question 1.1. The code is given below

% Implementation of image derivatives.

%Reading Image.

I = imread('image1.tiff');

%Asking user for input of sigma.

sigma = input('Please Enter the value of sigma:');

%Assuming size of gaussain matrix as 15 by 15.

hsize = [15 15];

%Creating a two-dimensional Gaussian filter h.

h = fspecial('gaussian', hsize , sigma);

%hx is gradient along x-axis direction and hy is gradient along y-axis

%direction

[hx,hy] = gradient(h);

%Convolving the 2D image with the gaussian filter in the x direction.

I2 = conv2(double(I),double(hx));

I2 = double(I2);

%Convolving the 2D image with the gaussian filter in the y direction.

I3 = conv2(double(I),double(hy));

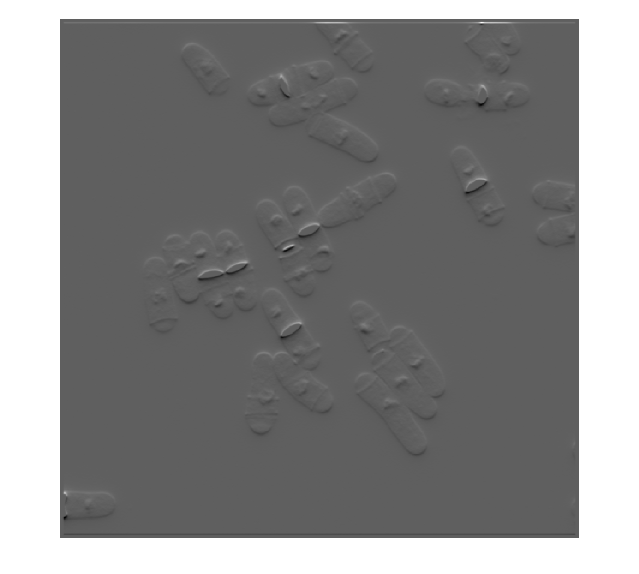
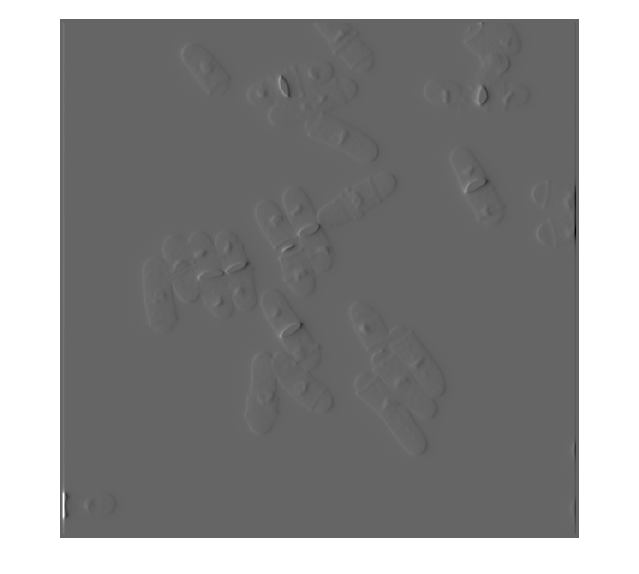
I3 = double(I3);

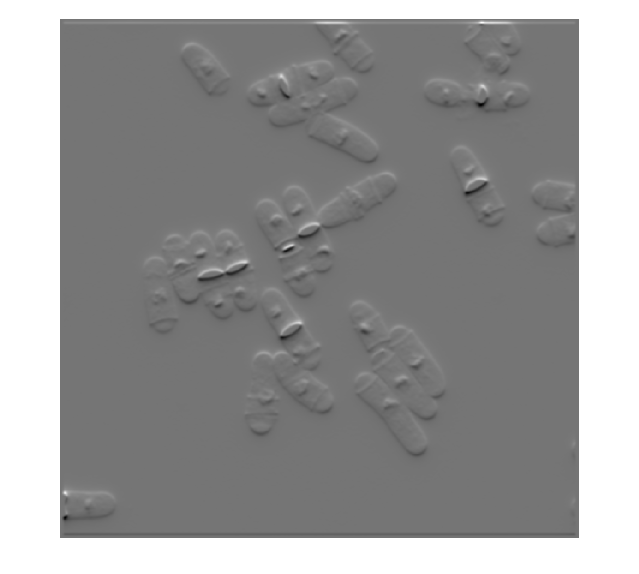
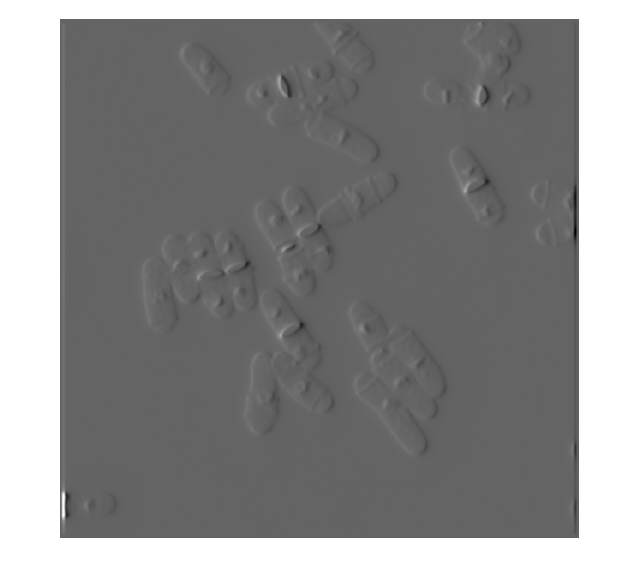
%Displaying original image, followed by derivative images.

figure, imshow(I,[])

figure, imshow(I2,[])

figure, imshow(I3,[]);

For sigma =1 , the derivative in the horizontal direction and derivative in the vertical direction are shown below respectively.

Similarly for sigma = 2,

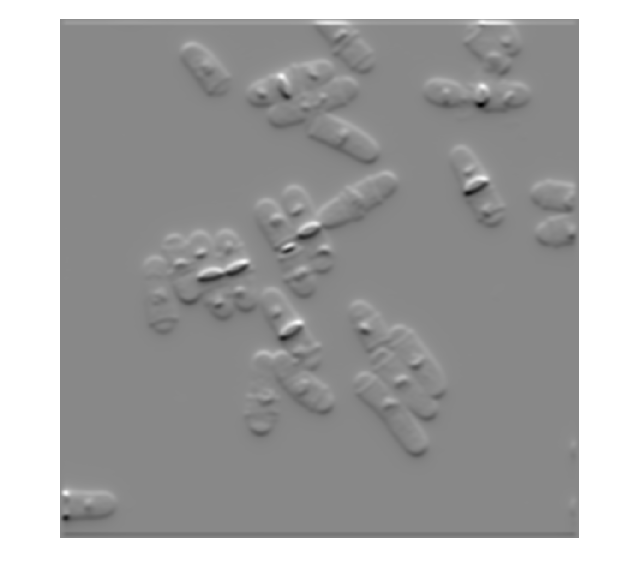
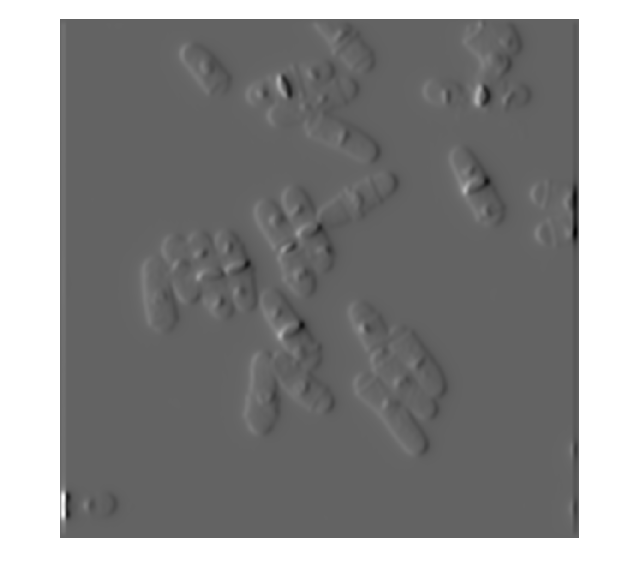
And for sigma = 5 ,

Image gradients measure the change in intensity levels of an image. For the images on the left, as derivatives are taken along horizontal direction, any change in intensity level along the vertical can be seen prominently. Conversely, for the images on the right, the derivatives are taken along the vertical direction and changes in intensity levels along the horizontal are prominent.