Laplacian Blob Detection

INTRODUCTION

- Blob detection refers to visual modules that are aimed at detecting points or regions in the image that are either brighter or darker than the surrounding.
- In order to automatically detect blobs of different sizes, a multi scale approach is necessary.
- According to the scale-space theory the multiple-scale LoG blob detector can locate blobs of different scales by detecting local extrema of the LoG scale-space representation after the scale-normalized LoG operation, where the scale of the detected blob is determined by selecting the one at which the maximum filter response is assumed

STEPS OF EXECUTION:

ALGORITHM AND METHODOLOGY USED:

1) Generating a Laplacian of Gaussian filter

a) The Log filter is generated using the equation:

$$LoG(x,y) = -\frac{1}{\pi\sigma^4}\left[1 - \frac{x^2+y^2}{2\sigma^2}\right]e^{-\frac{x^2+y^2}{2\sigma^2}}$$

(Where, (x,y) are the coordinates of the kernel, where x goes from –M/2 to M/2 and y goes from – N/2 to N/2. (M,N)=size of kernel.(In this program only odd sized kernels are used). Sigma=standard deviation of Gaussian.)

b) This kernel is generated by the function defined as : coeff(size,s)

c) The function takes in the size of kernel and sigma(s) as input and outputs the kernel matrix.

2) Laplacian Scale space

The following steps are performed 12 times to get 12 layers in the Laplacian scale space.

a) The initial scale is taken as k=1.32 and Initial sigma=2.

b) The kernel is obtained using coeff(size,s) function for given sigma and size Size of kernel=2*ceil(sigma*3)+1

c) Normalized Laplacian is obtained as sigma^2 times the above kernel. d) Filtering:

The image is filtered with the kernel obtained above. This is done using the modified function from project2. The out=frequency_filter(image,kernel) function performs convolution in the frequency domain. This is done in the following steps:

- The function takes image and kernel as inputs.
- 2-D DFT is performed on the image using the defined function D_2_fft(image).
- Make the kernel size equal to size of image by zero padding and using ifftshift.
- 2-D DFT is performed on the new kernel using D_2_fft() function.

- The 2 DFTs are multiplied and inverse dft is taken from 2_D_fft function to obtain the filtered image.

e) The Laplacian obtained is squared and saved in a 3D scale space.

f) Sigma is increased by a scale of k for each iteration

g) The above steps are performed for n=12 iterations to obtain 12 layers in the Laplacian scale space.

3) Non Maxima Suppression

a) The maxima is calculated for each element of the matrix with respect to the eight neighbors.

b) This max element is calculated for all 12 Laplacian scale space layers.

c) Maximum value of each pixel is calculated considering all the layers(in the 3D scale space).

d) We obtain matrices of 0s and max values using the maximum values created above for all the twelve 3D scale spaces.

4.) Displaying circles to represent the blobs

a) The non-maxima values are replaced by 0 and others by using a threshold=0.007 for all the layers in the scale space.

b) The radius is calculated for different values of sigma as r=1.414*sigma and stored.

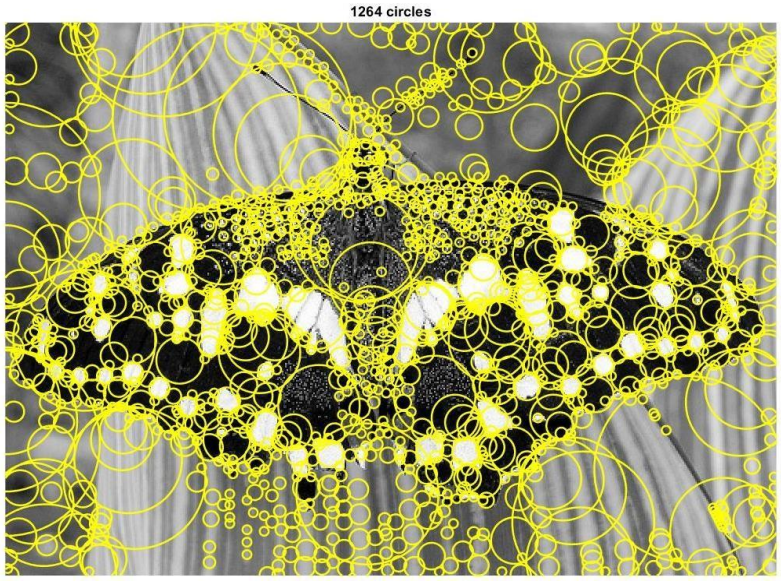
c) The coordinates of maxima are found and stored.

d) The values of coordinates of the center of maxima on the image and the radius of the circle is used to draw the circles using the function defined as display_circles(I,cx,cy,radius1), where I=image,(cx,cy)=coordinates of centre of the blobs, radius1=radius of the blob.

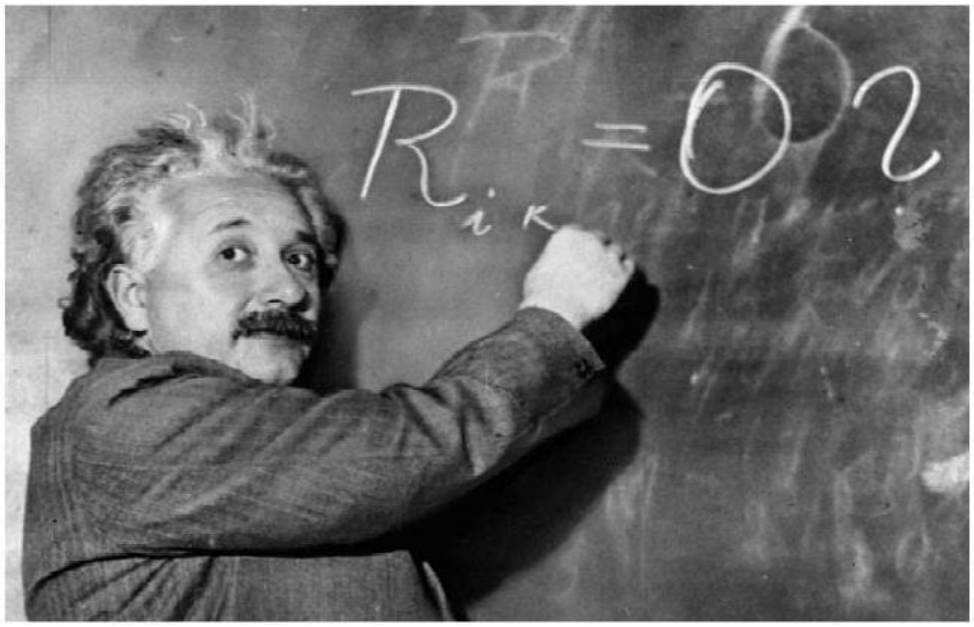
e) The number of circles (blobs) is also displayed.

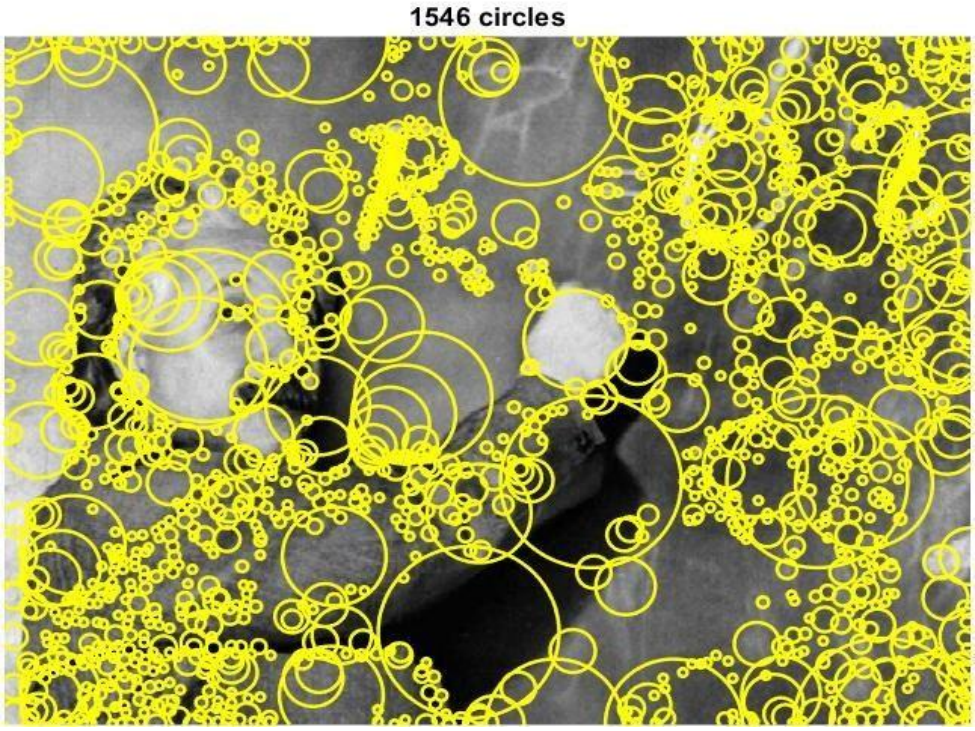
BLOB DETECTION RESULT-INPUT vs OUTPUT IMAGES:

1)Butterfly:

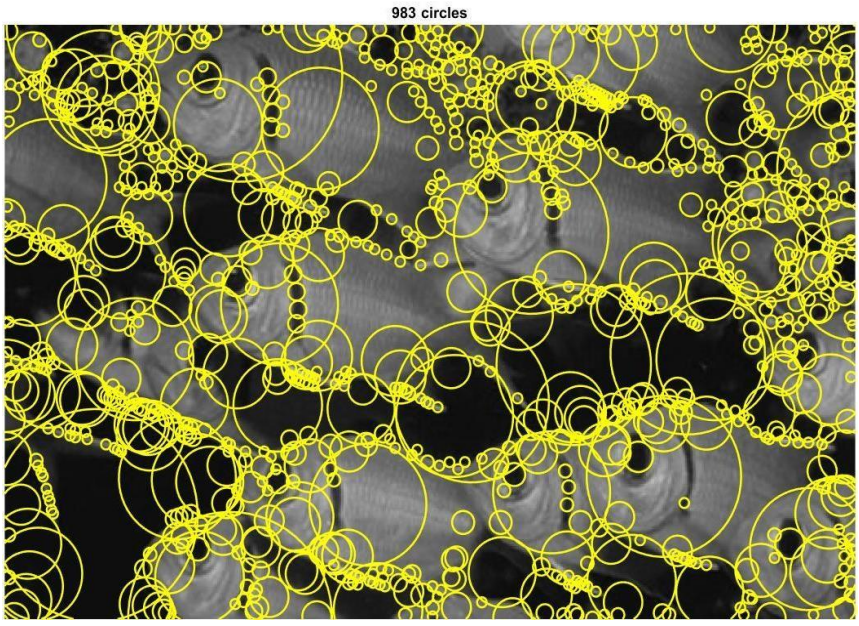


2) Einstein:

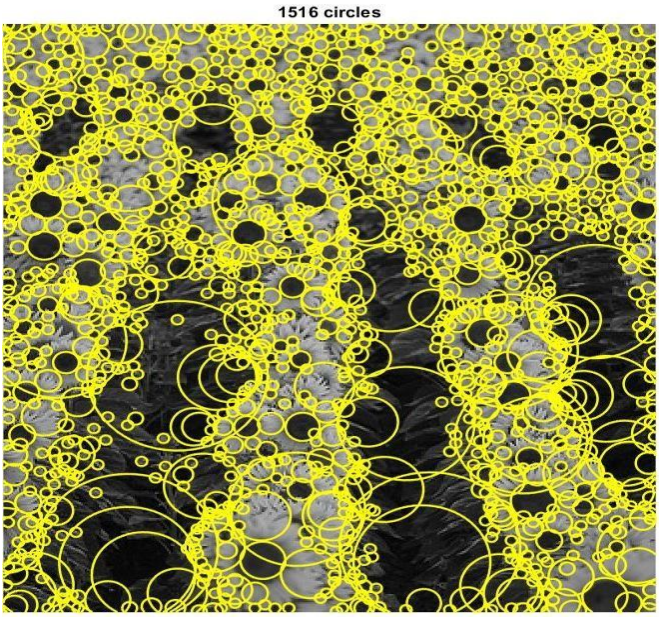




3)Fishes:

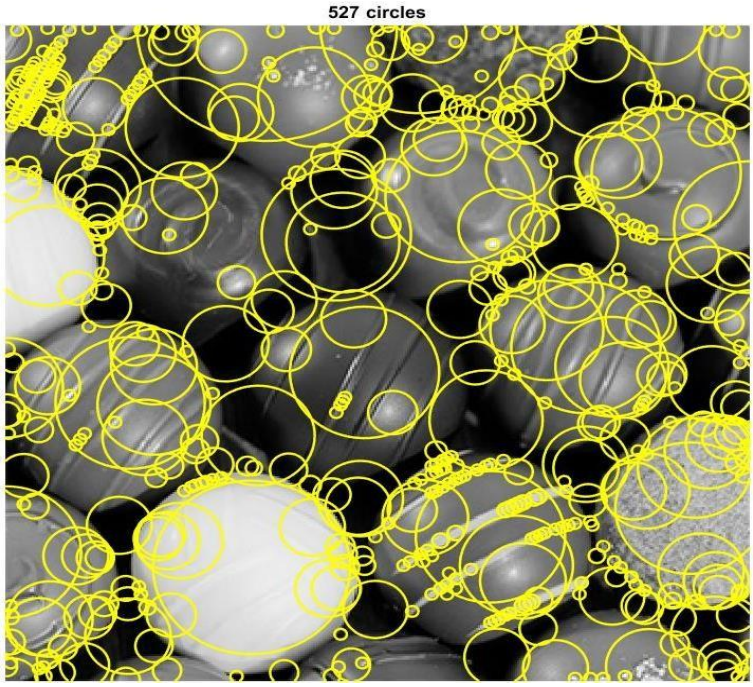


4)Sunflowers:

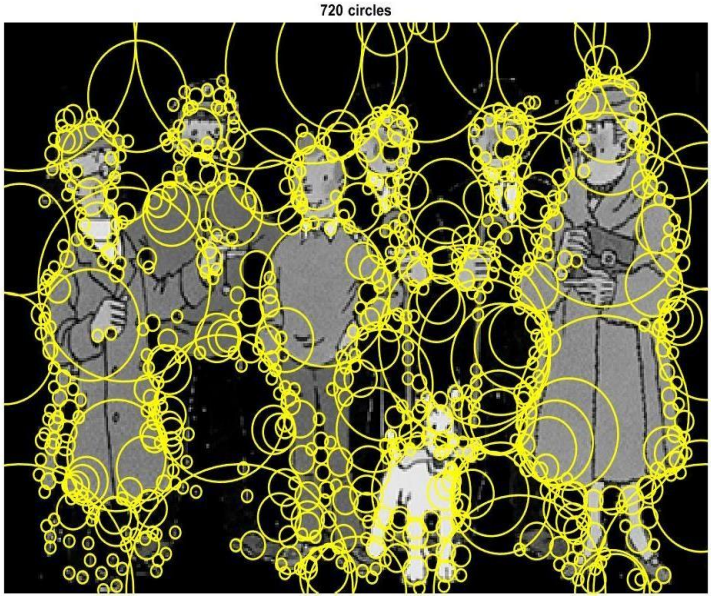


5.Cupcakes:

6.Chocolates:



7.Tintin:



8.Lena:

Average running time for above 8 images=37.26sec