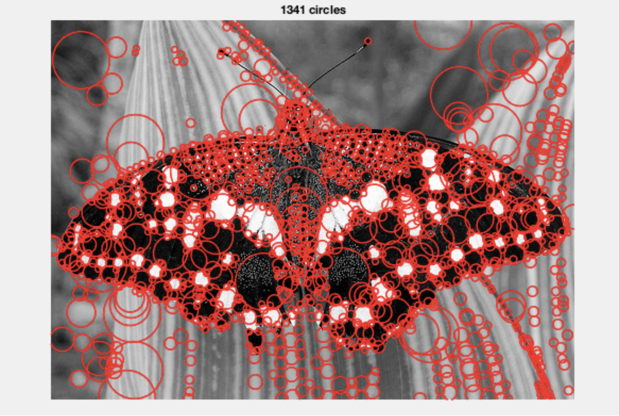
Name: Srinivas Ravi
UB #50244669
CSE573: Computer Vision and Image Processing
HW2 : Scale-Space Blob Detection

1. The output of circle detector on all images

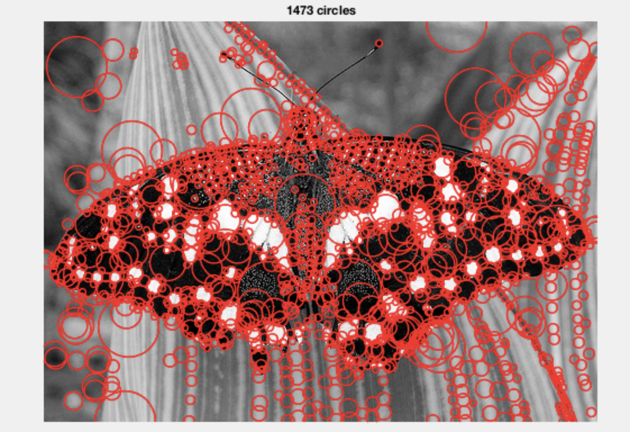
Method 1: kernel of increasing size by a factor of k (less efficient)

Method 2: Downsample the image by a factor of 1/k (more efficient)

1

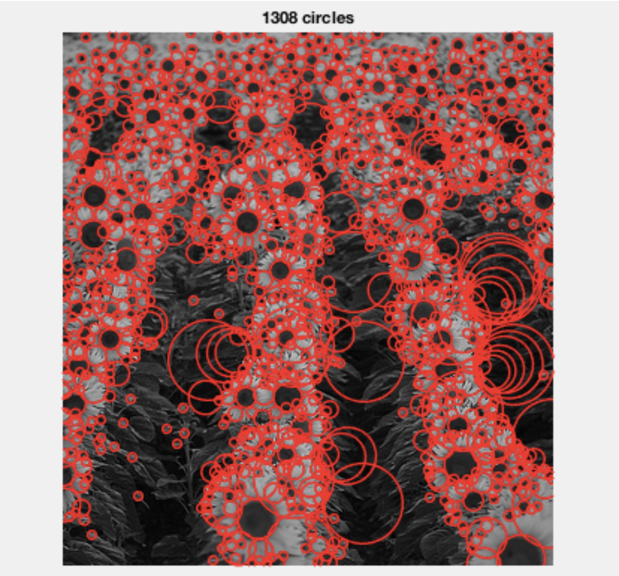


Runtime: 0.89 sec

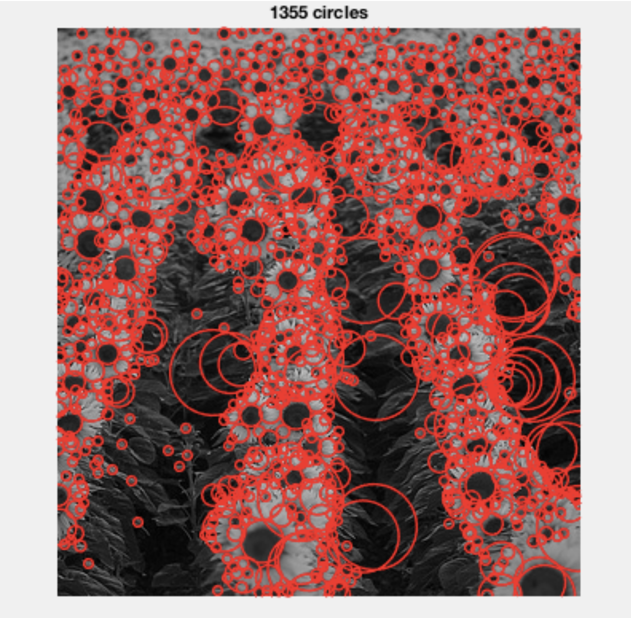


Runtime: 0.10 sec

2

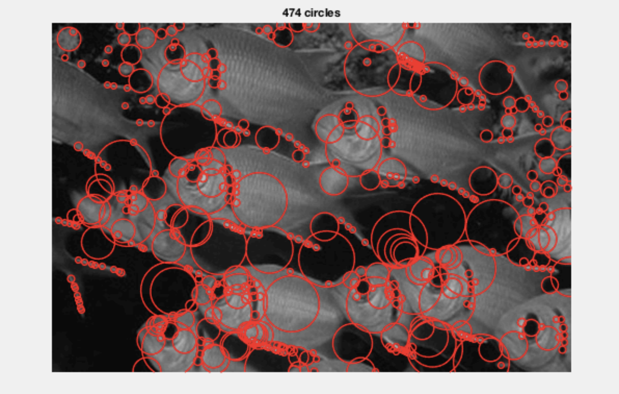


Runtime: 0.63 sec

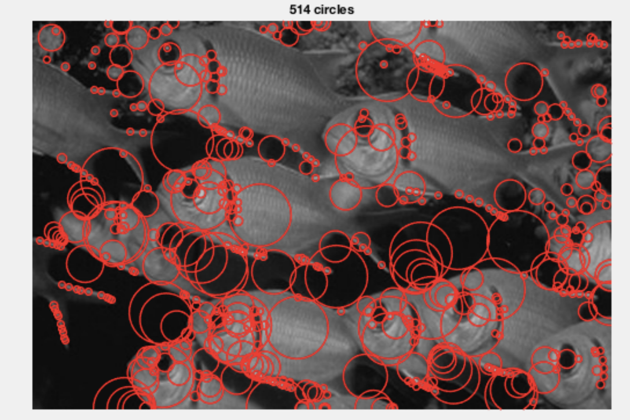


Runtime: 0.06 sec

3

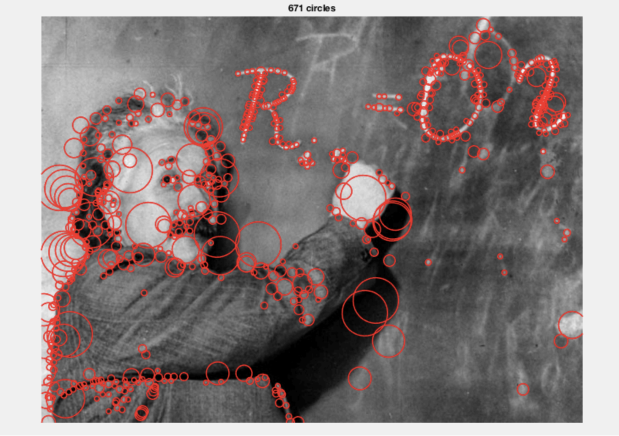


Runtime: 0.84 sec

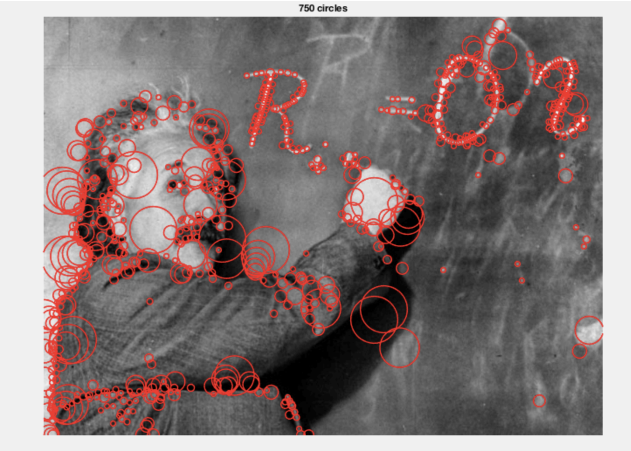


Runtime: 0.09 sec

4

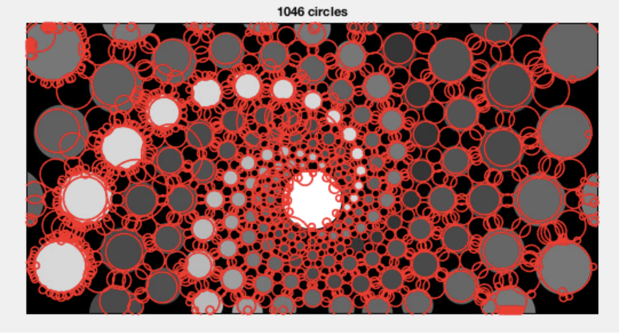


Runtime: 1.46 sec

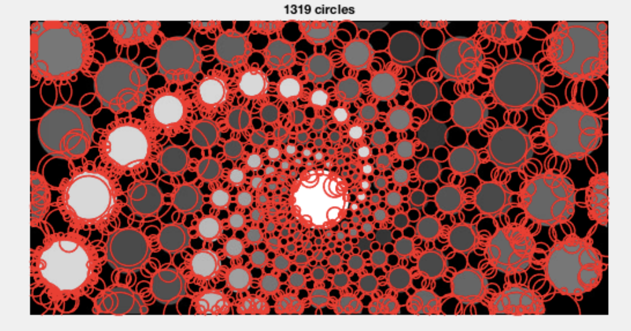


Runtime: 0.17 sec

5

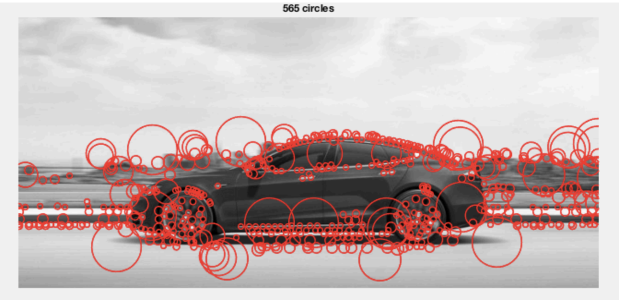


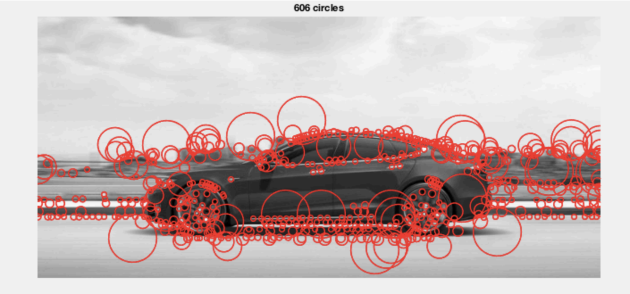
Runtime: 0.79 sec



Runtime: 0.07 sec

6

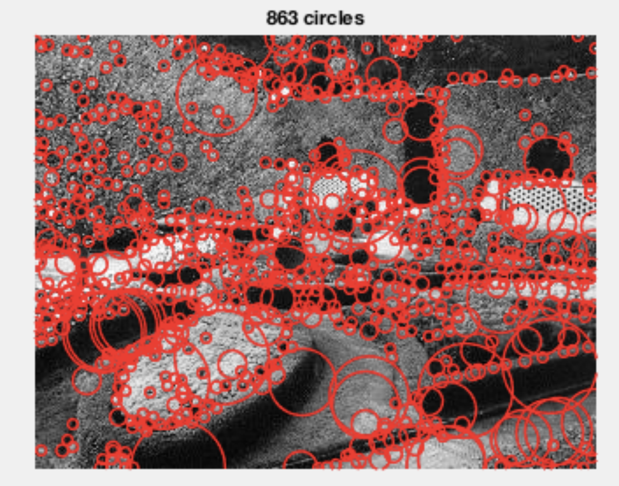


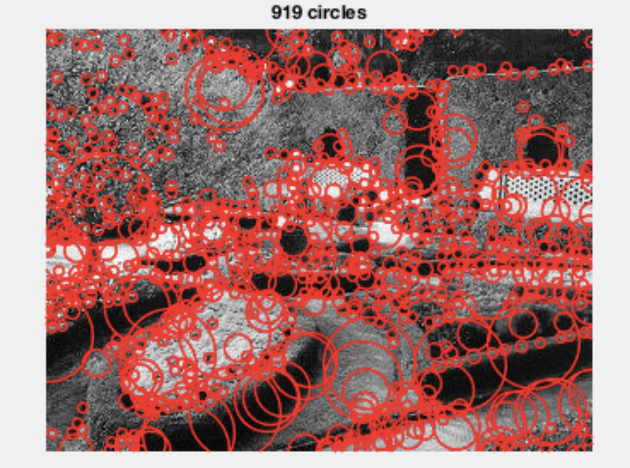


Runtime: 0.987 sec

Runtime: 0.12 sec

7

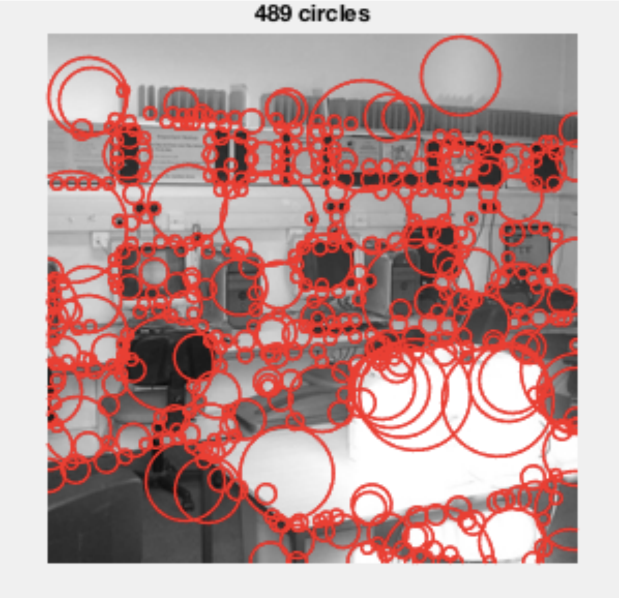


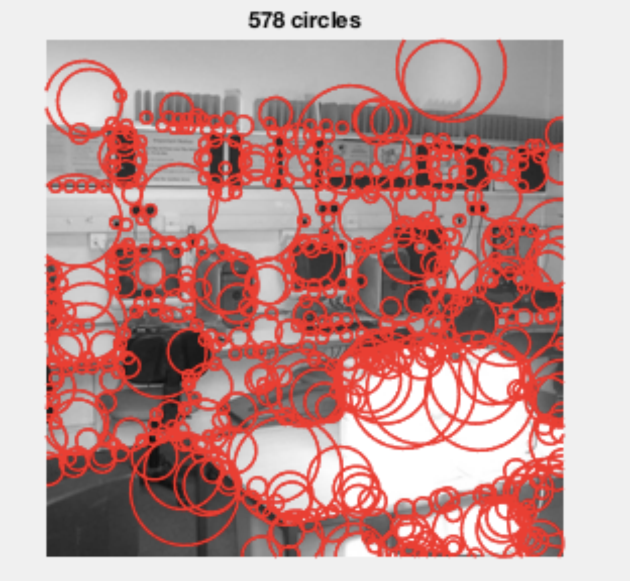


Runtime: 0.59 sec

Runtime: 0.07 sec

8





Runtime: 0.49 sec

Runtime: 0.05 sec

2. Explanation of any interesting implementation choices made

I) Implemented Scale-space blob detection using two methods:

i. Repeatedly filter an image with a kernel of increasing size by a factor of k (avg running time: 0.834)

ii. Downsample the image by a factor of 1/k (avg running time: 0.091) (**more efficient**)

II) For the non-maximum suppression within a scale, I first tried colfilt and later tried ordfilt2. The latter is faster to compute and work with.

III) I replaced the manual iterative processing of each pixel for non-maximum suppression between the scales and used max() for the 3rd dimension to make the implementation faster.

3. Explanation of parameter values tried and which ones you found optimal

I tried modifying the below values:

n – This value I played around by changing the max_sigma value (max_sigma = sigma * k^n). This value ensures that the LoG filter applied indeed gives some information about the image edges (or areas of rapid changes). A very high max_sigma would result in loss of information and a very low sigma value would mean lesser number of scales and also more sensitive to features that are not treated important by the human eye. I tried with n = 10(max_sigma = 12) to 17(max_sigma = 25) and decided to fix n =14(**max_sigma = 20**) as that ensured optimal information capture of areas of rapid change.

Initial scale **(sigma) = 2**. Changing this value to 3 or more meant more information that could be captured is not captured. A lower value than 2 would require substancially more processing hence chose the value of 2.

k – This determined what would be the step increase in the value of sigma for the different scales. A very high step would mean that very different edges will be compared in the adjacent scales and a very small step would lead to more n and thus more computation. Thus I chose **k = 1.189**.

Threshold – This decided which intensity values that were maximum in their locality would make it to the final blobs. Increasing the value gradually from 0.001 to 0.03 ensured lesser blobs are visible subsequently. I went for **0.01** as it ensured most of the circles drawn accurately represented the regions without cluttering the other circles.