Ans 1)

Functions:

getNeighbours(x,y,dist,h,w): get neighbours of pixel x,y at distance dist. H,w are height and width of the given image

valid(i,h,w) : check if point i is in the bounds of h and w

Procedure:

1. We read the image and scale it to 15% to make the computations faster
2. We then find the unique colors in the image
3. For every distance K, and for every color C, we calculate the neighbours that have the same colour, and save the count in a dictionary X, along with the total count of that colour T
4. Final feature added has the value X/(T*8*distance)
5. For calculating the distance, we find the difference in the value of features divided by the sum of these features +1. We do this for all distances and colour
    a. (f1+f2)/(1+f1+f2)
6. For cases when colour is not present, we take fx to be 0
7. We then read the query file and retrieve the feature of the given image.
8. We then calculate the score of image with every other image.
9. We calculate precision and recall for all the 3 classes(good,ok,junk)

Statistical Summary

Based on 7 queries,
Precision good = 0.028
Precision ok = 0
Precision junk= 0

Recall good = 0.55
Recall ok = 0
Recall junk = 0

Ans 2)

Functions:

extractFeatures(img): This function is used to find LoG of the image at 11 different scales. Features are then made square to do approximation of gaussian smoothening

Procedure:

1. We read the images, and convert them to greyscale.
2. We extract the features using extractFeatures Function

3. We interpolate over the image, and find the point which has the largest value among the neighbours across all the scales.
4. We perform thresholding and remove weaker points
5. We extract the centre and radius of the blobs and plot them

References for procedure and the exact values of variables:

https://www.di.ens.fr/willow/teaching/recvis10/assignment1/
http://www.cs.unc.edu/~lazebnik/spring11/lec08_blob.pdf
http://www.cs.utoronto.ca/~fidler/slides/CSC420/lecture7.pdf