Image processing - 046200

Homework #4

Alexander Shender 328626114

Sahar Carmel 305554453

Technion - Israel Institute of Technology

I. Question 1.

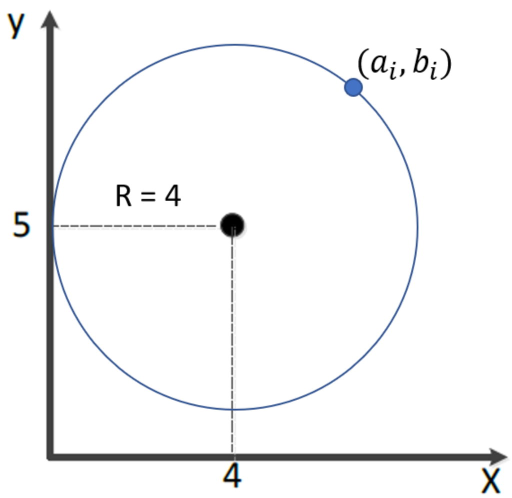
A.

Since the radius is constant, the parameters space is decreased to 2 parameters: a and b. The dimensions are 2D:

$$\{a, b\} \in R^2$$

B.

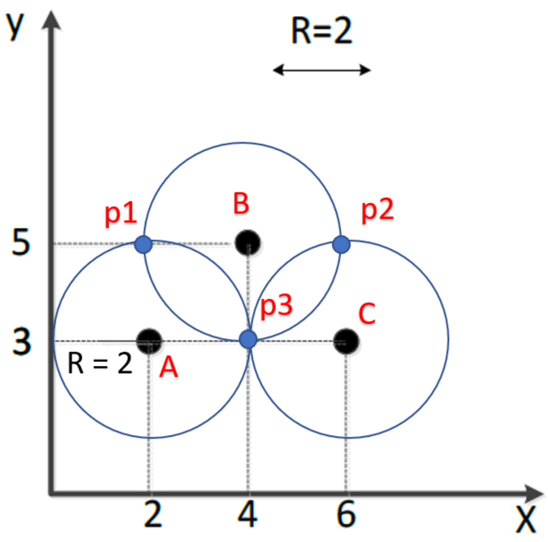
Given R=4, we are mapping all the possible circles with this radius which may have this point (which is given on the graph) on its perimeter. Meaning that any point on the circle which we drew (e.g. a_i, b_i) may be the center of a circle, which will include the main point given.



C.

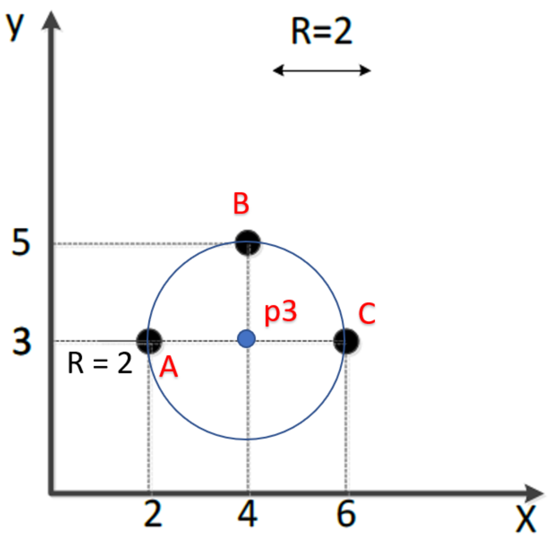
For each of those points A, B, C - we draw a circle around with a given radius R=2. We can see that there are 3 intersections between the circles, marked as p1, p2, p3. Which means:

1. p1 - can be a center of a circle which includes points A and B. It got 2 votes
2. p2 - can be a center of a circle which includes points B and C. It got 2 votes
3. p3 - can be a center of a circle which includes points A, B and C. It got 3 votes



D.

Yes, as we have seen, point p3 can be the center of a circle which includes all 3 points. The coordinates of p3 are (4,3), and the visualization of a circle is the following:



E.

The algorithm is divided into 2 parts . First part is the CHT transform on the image. Let us assume we have N points, and we do the transform on all of them:

for i = 1 **to** N points **do**

    for $\theta = 0$ **to** 360 deg (step = $\delta\theta$) **do**

$$a = x_i + R \cdot cos(\theta)$$
$$b = y_i + R \cdot sin(\theta)$$
$$A[a, b] + = 1$$

    end for

end for

Then, for the maximum points extractrion, we have to decide what is maximum. Points which lay on the same circle will create high value for the coordinates of the center of this point. Let us denote *threshold* as a threshold which the value has to overcome to be counter as 'maximum'.

for a = 0 **to** a_{max} **do**

    for b = 0 **to** b_{max} **do**

        if A [a,b] > threshold **then**

            Max points $\Leftarrow$ A [a,b]

        end if

    end for

end for

F.

The system H_1 is performing the binarization of the image, where a new value to a pixel is given based on its current value. The threshold is being chosen based on the histogram. By looking at the histogram provided, we can see that the background consists mostly of the very bright pixels, and the coins are darker. Putting the threshold to a value of 180 (where the least amount of similar pixels would get different values) would be reasonable. Mathematically:

$$g(m,n) = \begin{cases} 0: & iff(m,n) > threshold(180) \\ 2: & else \end{cases}$$

But after this binarization, the image may contain dark spots inside the coins, which in the original image had gray level beyond the threshold (we can spot some visually). Thus, to eliminate those spots, we perform the morphological 'Close' operation to eliminate dark dots inside the bright coins.

Thus, the system H_1 consists of Binarization and morphological operation 'Close'. This system is non-linear, since both binarization and 'close' are non-linear transformations.

G.

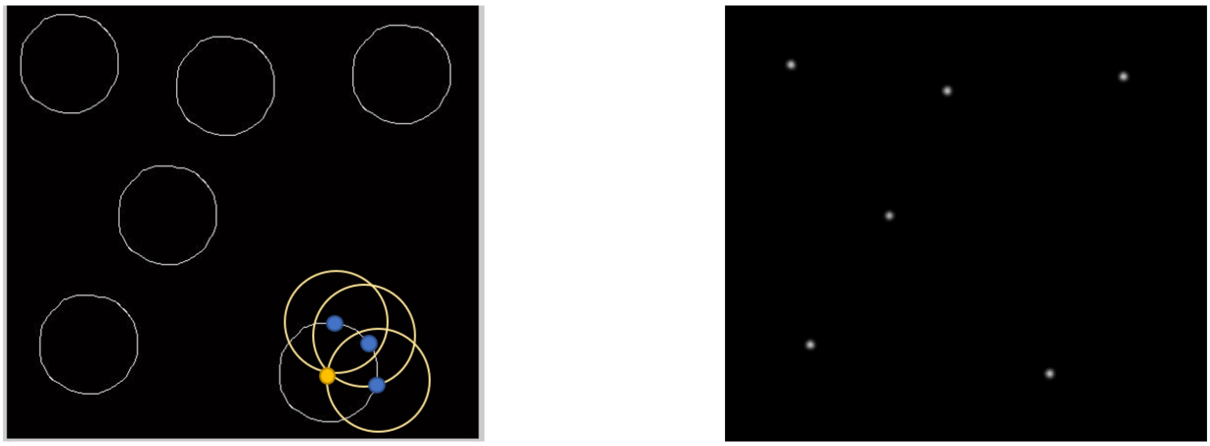
The system H_2 which performs the described operation can be done with morphological operations. We suggest the following system:

$$v(m,n) = g(m,n) - (g(m,n) \Theta SE)$$

where the erosion is made over the SE, which is a small disk. What yielded the similar result in MATLAB is the SE defined as:
SE = strel('disk',1,4);
which a disk of 1 pixel in radius and 4 line structuring elements. This system is as well non-linear, since the 'erosion' is a morphological non-linear operator.

H.

Let us take the bottom right circle. CHT transform creates a circle around each line of radius of 40 pixels. All the pixels on this circle get a vote in the CHT plane. Then, the maximum points on the CHT place are the centers of the circles. Left image shows some circles, while right one 'recreates' the CHT plane. We can see than because the circles are not perfect in an input image, so the bright spots in CHT plane are not single pixels, but a bit blurred.



(a) CHT algorithm on pixels

(b) CHT plane

Figure 1: Finding the circles on an image through CHT transform

II. Question 2.

A.

The correspondence is the following:

1. $g_1 \rightarrow Hf$ - is the following Blur operation. No noise.
2. $g_2 \rightarrow f + N$ - only Noise is added
3. $g_3 \rightarrow f$ - is the same as input
4. $g_4 \rightarrow Hf + N$ - input is blurred, and noise is added.

B.

Both quality measures E_b and E_a measure the norm of the gradient in an image. First, two major points which help us to solve the question:

1. Blurring - *decreases* the absolute value of the gradient. Thus, it will decrease both the first and second norm of the gradient.
2. Noise - is a random signal which is added. It has no significant effect on the first norm of the gradient (because gradients of the different directions cancel out), but it increases a lot the second norm of the gradient (since every gradient values are being squared, thus also negative gradients are being summed up)

Thus, the result is the following:

$$\begin{cases} E_a(g_1) \approx E_a(g_r) < E_a(g_2) \approx E_a(g_3) \\ E_b(g_1) < E_a(g_3) < E_b(g_4) \approx E_b(g_2) \end{cases}$$

III. Question 5.

We are looking for a way to maximize each of the priors (those are PDFs).
The correspondence is the following:

1. what is measured here is the gradient value in the X direction, multiplied by the pixel value.
    - the most probable priors (which maximizes this expression) are images A,B (has no gradient, all D_x values are 0).
    - The most unprobable prior is image F which has a lot of sharp gradients. (where it passes from black to white the pixel value is 1)
2. here both gradients are taken into consideration, and also the pixel value.
    - the most probable priors is image C, which has no gradients, and its pixel values are 0. This whole expression will be 0 for it.
    - The most unprobable prior is image A. Its pixel values are at maximum, and also it has no gradients (which come here with opposite sign, which means that more gradients is good for probability)
3. similarly to case 1, we look here at the gradients in the direction Y.
    - Similarly to 1, the most probable images are the ones with no gradients in the Y direction, which are A, C, F
    - The most unprobable prior is now G (E may have approximately same value), which has gradients on the pixels which have non-zero value, meaning it will decrease the probability.
4. Here we are looking for the biggest pixel values.
    - Image A is the most probable, it has the maximum amount of biggest value (white=1) pixels
    - Image B is the most unprobable.