Homework 3.

Assem Kussainova. 20153579 8.

1) To find a bounding box we can project the filter onto the image using the affine model of the desired area of the image. To get the indicated affine model, we firstly need to limit the area of the person in the images. Since the face is not covered with clothing, we segment the parts of the human body with an open surface, that is the parts where the human skin is visible. After that you need to extract the necessary features that will help determine whether the segment is a person's face. To do this, we can use SIFT method, since unlike the histograms of oriented gradients, this method doesn't require windows to study the entire image, but uses interest points, therefore it is more convenient in this case. Thus, we find keypoints on the surface of the skin and extract features of each segment. After that, we send received features to the affine model, and compare it with the affine model of facial features, and those points that are most suitable and similar designate the face area.

2) $(E(x_c, y_c, r))' = -2\sum_{i=1}^{m}\left(\sqrt{(x_i-x_c)^2+(y_i-y_c)^2} - r\right)$

$$-2\sum_{i=1}^{m}\left(\sqrt{(x_i-x_c)^2+(y_i-y_c)^2} - r\right) = 0$$

$$-2\sum_{i=1}^{m}\sqrt{(x_i-x_c)^2+(y_i-y_c)^2} + 2\sum_{i=1}^{m} r = 0$$

$$-2\sum_{i=1}^{m}\sqrt{(x_i-x_c)^2+(y_i-y_c)^2} = -2\sum_{i=1}^{m} r$$

$$\sum_{i=1}^{m}\sqrt{(x_i-x_c)^2+(y_i-y_c)^2} = m\cdot r$$

$$r = \frac{1}{m}\sum_{i=1}^{m}\sqrt{(x_i-x_c)^2+(y_i-y_c)^2}$$

b) $N = \dfrac{\log(1-p)}{\log(1-(1-e)^s)}$

$p = 0{,}2$

$e = 0{,}3 \Rightarrow 1-e = 1-0{,}3 = 0{,}7$

$s = 3$, since we need to choose x, y and r

$N = \dfrac{\log(1-0{,}2)}{\log(1-0{,}7^3)}$

3) The performance is suffering since there is a problem inherent to performing this task, which is a recognition of Hoover tower with images where vary the angle of view on an object. As this object is not aligned it exhibits different spatial features and cause errors because of shape and illumination differences. Therefore, images alignment should be done before PCA.

4) The input to the K-means algorithm is the color of pixels. However, color, brightness, position alone are not enough to distinguish all clusters. Therefore, we can group pixels based on both color + position similarity, thereby transferring position features with color features into the input of algorithm to get proximity, which helps to overcome given problem.

5) a) 1: x_1 2: x_2, x_3, x_4 1: x_1, x_2 2: x_3, x_4

1: x_2 2: x_1, x_3, x_4 1: x_1, x_3 2: x_2, x_4

1: x_3 2: x_1, x_2, x_4

1: x_4 2: x_1, x_2, x_3

b) 1. $\sum 0 + \frac{2\sqrt{5}}{3} + \frac{2\sqrt{5}}{3} + \frac{2\sqrt{2}}{3} = \frac{4\sqrt{5}+2\sqrt{2}}{3}$ for clusters where 1 point is in one cluster and the rest points in second clusters;

2. $\sum 1+1+1+1 = 4$ for clusters where points divited by two

$3{,}924 < 4$, therefore the smallest cost will have following clusters:

1: x_1 2: x_2, x_3, x_4

1: x_2 2: x_1, x_3, x_4

1: x_3 2: x_1, x_2, x_4

1: x_4 2: x_1, x_2, x_3

c) So, if similar to this conditions will be given in K-means clustering, it will be needed to initialize several clusters options like in Step A, and calculate their costs like in Step B, and then remain only those clusters which have smallest costs, for further output.

6) There are two variables which are changed while performing k-means clustering: mean and centroid of clusters. If the algorithm terminates in a finite number of iterations, it is needed that these two operations never increase the value of L. Lets talk about mean of clusters. It uses value of current centroid for computation. New centroid is chosen in a way that distance from each point in cluster to centroid should be less than previous. distance to old centroid, therefore changing centroids can only decrease L. So, as this centroid is used in mean calculation, this will also minimise squared distances. As this is true, after the first iteration of data, there are only finite possible assignments of mean and centroid of clusters. In addition, L is lower-bounded by 0. Therefore, we can make a conclusion that L cannot decrease more than a finite number of times, and will terminate at one point. ■

7) a) $$A(i,j) = e^{-\frac{\|f(i)-f(j)\|^2}{2\sigma_I^2}} \cdot \begin{cases} e^{-\frac{d(i,j)^2}{2\sigma_X^2}}, & \text{if } d(i,j) < r \\ 0, & \text{otherwise} \end{cases}$$

This affinity measure reflects the likelihood of the two pixel intensities which belong to one object. It takes pixels that are in a close distance to each other which is denoted by condition $d(i,j) < r$, therefore affinity measure for pixels whose distance exceeds some r is taken as 0.

b) Stage 1: use affinity measure for each of the sub-image, apply NCut, ~~merge~~ the results and compute components which are close to each other;

Stage 2: create new graph with each component from stage 1 representing unique node, calculate affinity measure for adjacent nodes, apply NCut