End Sem

Special Topics in Computer Vision

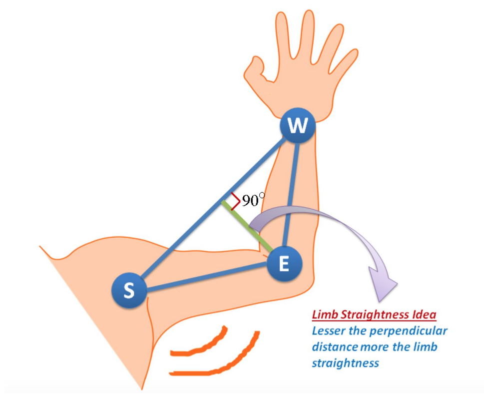
Gagandeep Singh: 2016037

Question 1

Given **S**(sx, sy), **E**(ex, ey) and **W**(wx, wy) to be the coordinates of the shoulder, elbow, and wrist of a limb, respectively.

The following features could be used to detect if the limb is straight or not:

1. We could calculate the slope of the line segments between points **S-E** and point **E-W.** If the absolute value of the difference between their slopes is less than a threshold, we can claim the limb to be straight.
2. We can consider the line between the shoulder and wrist, i.e., between points **S-W** and find this lines distance from the elbow point. The smaller the distance, the higher the probability of the limb being straight



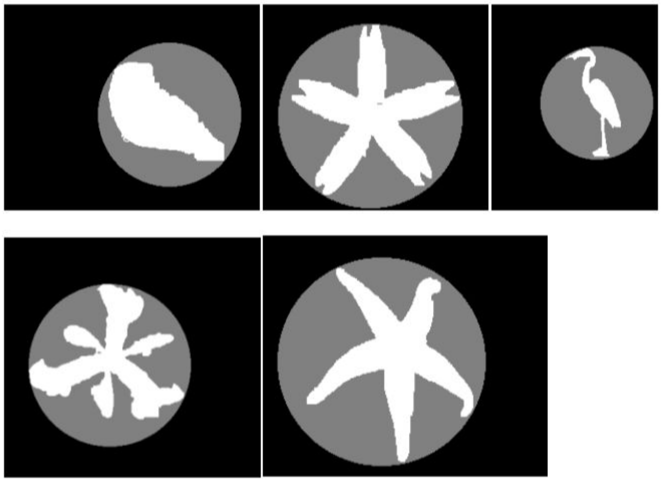
*taken from slides

3. We could also calculate the angle of the limb at the elbow joint. The smaller the angle, the higher the probability that the limb is not straight.

4. We can also find the distance between the wrist joint and the shoulder joint and if this distance is smaller than a threshold, we could consider it to be a bent limb. This feature would not work well with people having quite a huge amount of variation in their arm size.

Question 2

Given a segmented region in the image we can devise the following way to compute a circle that tightly fits the region as shown below:

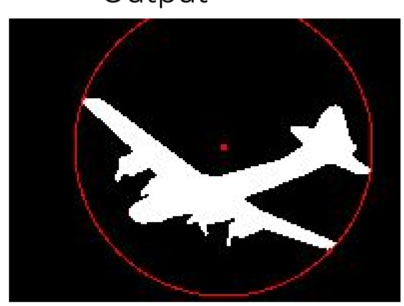


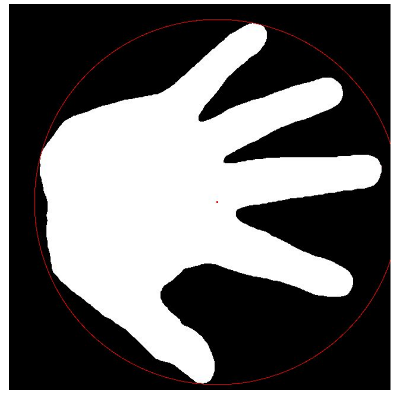
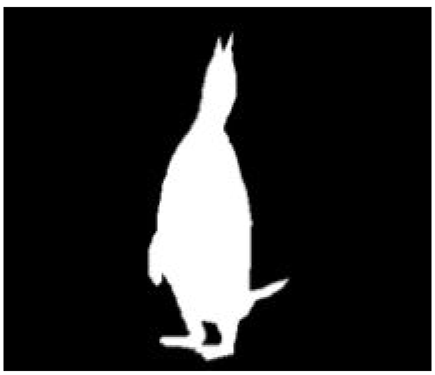
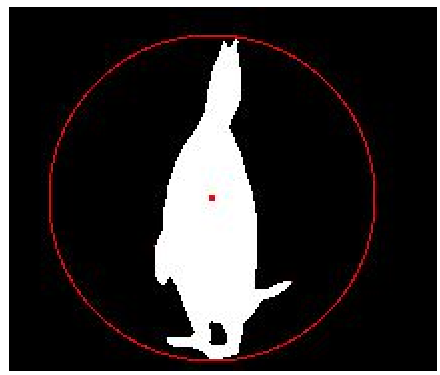
We can follow the given steps to get a circle that tightly fits the given segmented object:

1. Apply Otsu's thresholding to convert the segmented object into a binary image.
2. We could loop over the given segmented region and find the **leftmost(l)**, **rightmost(r)**, **topmost(t)** and the **bottommost(b)** point of the white object we are willing to encircle.
3. We then iterate over all the pixels occurring in the region between the points **r, t, b, l.**
4. At each pixel, we find the **maximum** distance (<u>in terms of location and not the color of the pixel</u>) of that pixel from all the 4 points r, b, l and t.
5. Whichever pixel has its maximum distance as minimum among all other pixels, we declare that pixel as the **center** of the circle and its maximum distance from the 4 points as the **radius** of the circle

Input

Output



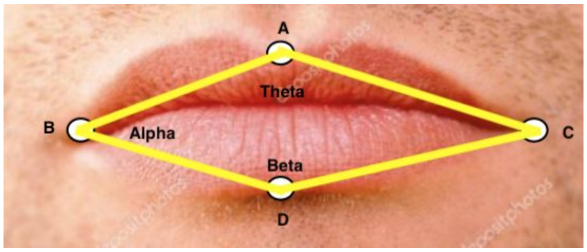
Note: Code is also attached with the submission.

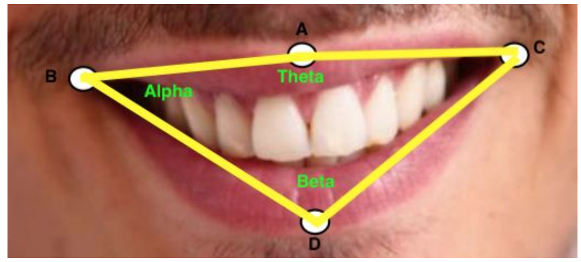
Question 3

The following three features could be used to help us monitor if a person is smiling or not:

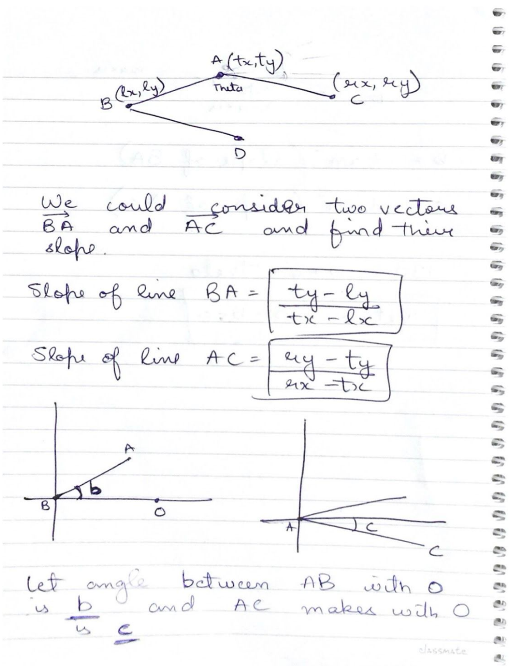
Keywords:

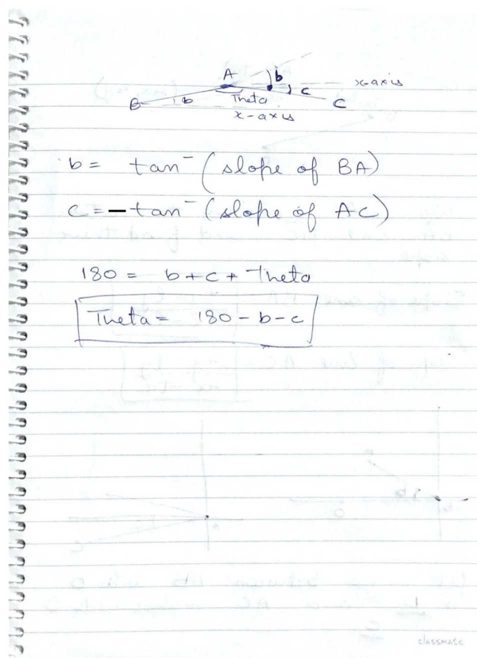
a. **Top** : The top point in the lip
b. **Left** : The leftmost point in the lip
c. **Right** : The rightmost point in the lip
d. **Bottom**: The bottommost point in the lip
e. **Alpha** : The angle between Top-Left-Bottom angle (as shown below)
f. **Theta** : The angle between Left-Top-Right angle (as shown below)
g. **Beta** : The angle between Left-Bottom-Right (as shown below)





1. The more the angle **Theta** is close to 180 degrees, the better we could conclude that the person is smiling.
   The following image shows how to calculate **Theta:**





   We could check if **theta** ranges from 180-**thresh** to 180+**thresh**, then we can consider that the person is smiling.

2. The second feature we could consider is the distance between the topmost and the bottommost points, as if the person is laughing with their mouth wide open, the angle **theta** would be really close to 180 degrees. In those cases, we could also consider the distance between the top and bottom point, and check is that distance is greater than the threshold, along with **theta** being in range near 180 degrees where **thresh**(as describe in previous point) for theta is a bit higher, even then we could consider person to be smiling.

Combining these two features would be essential to eliminate false positives where even if the person is crying or shouting, considering only the distance between the top and the bottom point would yield false results.

3. Another feature which could help us monitor if a person is smiling is by calculating the ratio of the angles **theta** and **beta.**
   Seeing how the human mechanics work, as theta grows(i.e. person starts to smile), the value of beta would decrease, thus increasing the ratio **theta/beta.**

   So if the ratio falls below 1, i.e. beta>theta , then the person is not smiling, else if the ratio rises above 1, i.e. theta>beta, then we can conclude the person is smiling.
   Both angles theta and beta could be calculated as shown in feature 1.

4. We could also consider the distance between the left and right point as when the person is smiling, this distance would increase. We can threshold it to check if a person is smiling or not.