Daniel McDonough
HW10
CS534
11/17/19

1.
    1. The model should be represented as a set of edges and regions. The contours provided by the set of edges can determine the set of pixels that may determine a face and the edges in the regions can be segmented into further parts; eyes, nose, mouth.
    2. The image should be processed using a feature detector such as SIFT to detect regions such as the eyes and head, and edge detectors such as Sobel to detect the mouth and nose curves.
    3. Matching should be performed via a mask passing through and classified over a threshold.
    4. This is invariant to translation but not to rotation or scale
    5. To handle partial occlusion, the algorithm could propose a case of the thresholding such that if a hard edge is detected and the region beyond that edge is not part of the class face, it is part of the class obstruction. And a new threshold will be calculated to account for the occlusion.
2.
    1. We match the model to the image as the model can be simplified to compare to larger number of images edges, the number of model edges will be lower than the image edges, therefore being a faster computation.
    2. No, as the base model will already include the null element and “objects” are grouped based on irrelevance given the model. Comparing the model to the images the model would only have to include E^{M+1} matches.
3.
    1. As $(|I_1-I_2|)^2 = I_1^2 - 2I_1I_2 + I_2^2$
    2. and $I^2 = sum(i_{jk}^2)$
    3. then $(|I_1-I_2|)^2 = sum(i_j^2) - 2I_jI_k + sum(i_k^2)$
    4. such that $-2I_jI_k$ is equivalent to the overlap of I1 and I2
    5. therefore $(|I_1-I_2|)^2 = sum(i_j^2) + sum(i_k^2)$ – pixels where I1 == I2

4.
    1. Given a set of equally likely points Nearest mean classifier (NMC) would choose k random starting points and compute class labels per class mean. Set of points in the class would also keep track of the variance of the point from the mean. Iteratively NMC then calculates what points belong closer to the set such that the mean-variance is minimized in each class. Given a binary classifier A/B such that classes A and b are equally likely and the line segment between the mean points of classes A and B, then the decision line must be perpendicular to the line segment between the mean points. the decision plane is not effected by the length of the line segment ,as it will always lie in the middle, between Mean A and mean B but rather the angle of the line. If the decision line is not perpendicular then the points that lie in the offset angle would not be classified correctly as the variance of those points to the incorrect class would be too high and the algorithm would alter the angle of the decision boundary to minimize the variance. Therefore the decision boundary lies midway, perpendicular to the the line segment of mean A and mean B.