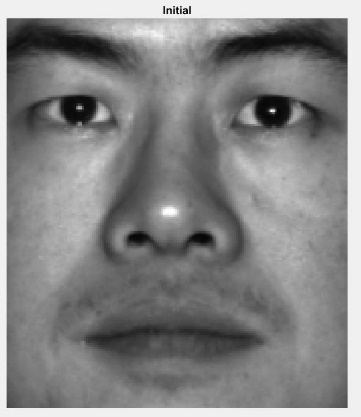
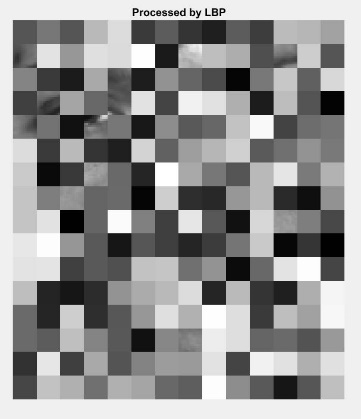
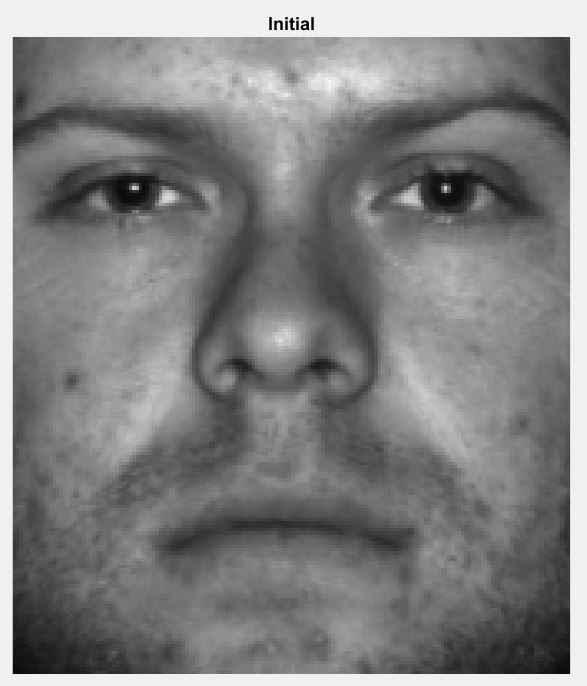
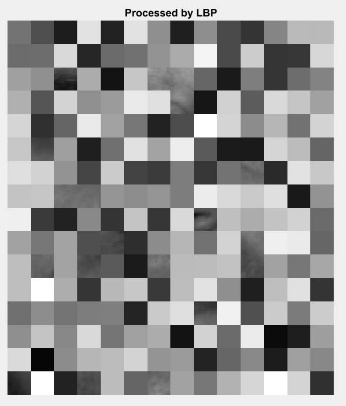
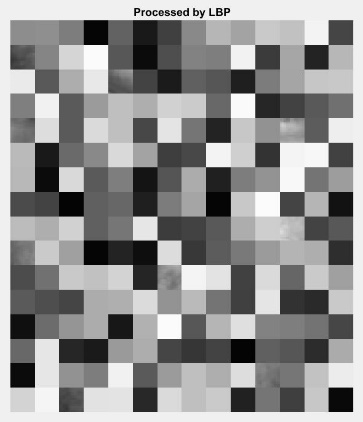
a) Display any 5 original faces using the codebook of 256+1 codewords that are associated with the Bag of Words referred to above. Do this by assigning the appropriate codeword color (1 out of 257) to each pixel in each image.

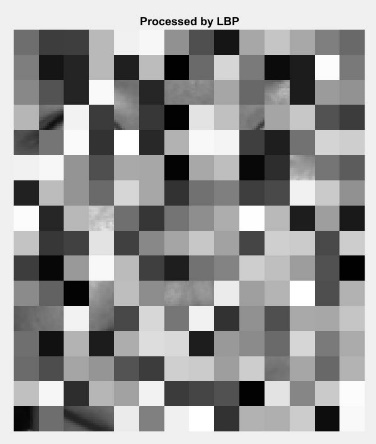
LBP method. The higher frequency codewords are, the darker color of the patch is. Therefore, the most frequent codeword has a value of 1, whereas the least frequent codeword has a value of 256. All of the other codewords outside of the most frequent 256 ones are assigned with the original/old pattern.

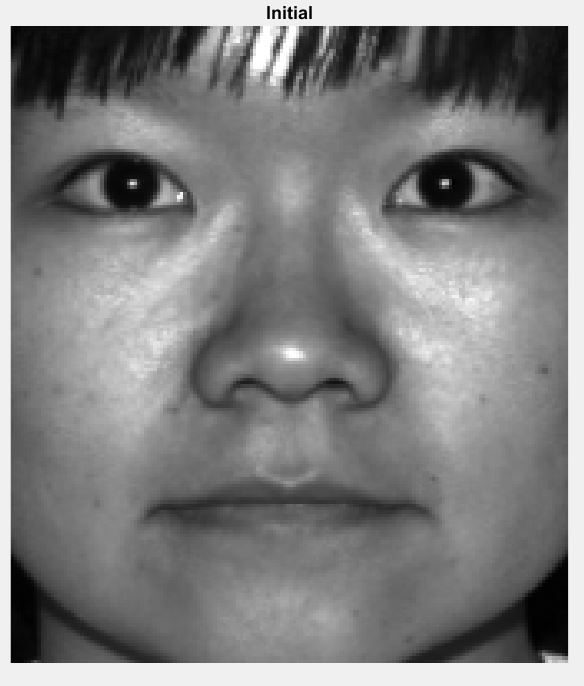












**Due to the issues encountered during the programming, the output for Dense SIFT and HoG are not successfully generated. However, these problems are in the few last steps in the code. The codes for these two are available in the zip file attached.**

b) Given the results in a), contrast and compare the three feature detectors in light of them being used as potential data inputs for a face detector. Which features would you use and why? Are 256 codes adequate for this task?

The LBP compares each pixel with its neighboring 8 pixels. Based on the comparing results, it is able to generate 256 different outputs for every single pixel. By using a uniform LBP operator, the dimension of each feature descriptor is 59.

The Dense SIFT operator firstly creates a Laplacian image pyramid, and finds the location of the maximum and minimum generated. After interpolating locations, it eliminates those locations where Laplacian values are below a threshold. Ultimately, it gets rid of locations if the locations are similar to edges and compute the orientation of the SIFT keypoint. The dimension of each feature descriptor is 128.

Lastly, the HoG operator computes the histogram of gradient orientations on each cell. Instead of computing the gradient magnitude, which is sometimes affected by noise, the direction of the gradient is computed because it is hard to be effected by other factor such as illumination. HoG computes edge orientation histograms, which is magnitude-weighted.

In terms of facial detection, HoG has no compensation for scale, which makes it better than SIFT for such tasks due to little change in scale. SIFT descriptor is computed around keypoints, whereas HoG descriptor is computed on a dense grid of uniformly spaced windows. The biggest difference between them is the descriptor blocks, HoG uses more cells than SIFT to cover large image patches. These cells are grouped in descriptor blocks, which overlap with each other.

On the other hand, LBP has the advantage of simple computation, which is faster than SIFT and HoG. This matters when speed is important. By configuring cell size and sampling radius, it can successfully and accurately detect flat areas, edges, and corners, which are the important features on faces.

I would like to use LBP method due to reasons above. Its computation complexity is low, which is increasingly important when the number of face images processed increases. It has good discriminative power and is very robust to monotonic intensity transformations. Even though most of the values in the histogram are equal to zero, it is still very handy and powerful method if some compression is applied.

256 code is not adequate for this problem, because the number of testing images is large. The testing images have unique features differing from each other, which results in a large number of codewords that are not successfully captured due to the limitation size of the codebook. By having a larger number of codes, (300 for example), more visual words can be included in the codebook and therefore having a sufficiently large coverage.

Reference:

1. ECSE-415 Introduction to Computer vision, Image Feature, by Professor Arbel
2. Lowe, D. G., “Distinctive Image Features from Scale-Invariant Keypoints”, International Journal of Computer Vision, 60, 2, pp. 91-110, 2004.