CS131: Computer Vision: Foundations and Applications
Sample Final Exam Problems Solutions

December 4, 2015

Disclaimer: The sample problems are not representative of the true length of the final exam. It is intended to provide you an idea of the format of the questions you are going to have in the final.

1 Multiple Choice

Mark one best answer, unless otherwise indicated

1. In Canny edge detection, we will get more discontinuous edges if we make the following change to the hysteresis thresholding:

   (a) increase the high threshold

   (b) decrease the high threshold

   (c) increase the low threshold

   (d) decrease the low threshold

   (e) decrease both thresholds

   Solution c

2. (Mark all that apply) Which of the following are true about PCA?

   (a) Can be used to effectively detect deformable objects.

   (b) Invariant to affine transforms.

   (c) Can be used for lossy image compression.

   (d) Not invariant to shadows.

   Solution c, d

2 True or False

For these problems no explanation is required; simply write True or False.

1. The Canny edge detector is a linear filter because it uses the Gaussian filter to blur the image and then uses the linear filter to compute the gradient.

Solution False. *Though it does those things, it also has non-linear operations: thresholding, hysteresis, non-maximum suppression.*

2. It is possible to blur an image using a linear filter.

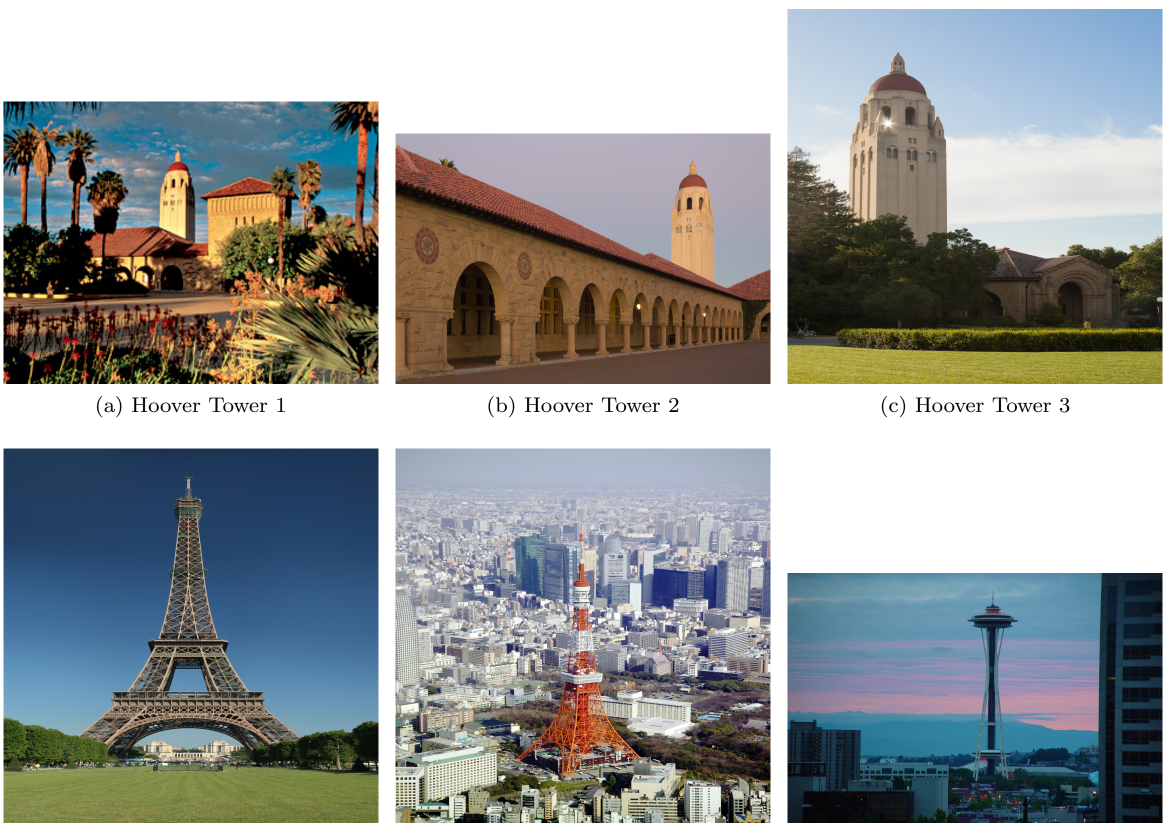
Solution True.

3. Fisherfaces works better at discrimination than Eigenfaces because Eigenfaces assumes that the faces are aligned.

Solution False. *Both assume the faces are aligned.*

3 Short Answers

1. Given a dataset that consists of images of Hoover tower and some other towers, you want to use PCA (Eigenface) and the nearest neighbor method to build a classifier that predicts whether new images depict Hoover tower. A sample of your input training images are given in Figure 1. In order to get reasonable performance from the Eigenface algorithm, what preprocessing steps will be required on these images?



(a) Hoover Tower 1 (b) Hoover Tower 2 (c) Hoover Tower 3

(d) Other Tower 1 (e) Other Tower 2 (f) Other Tower 3

Figure 1: Tower dataset

Solution

- Align the towers to be in the same position in the image.

- Scale or crop all images to the same size.

2. Given a dataset that consists of following points

$$\begin{aligned} x_1 &= (-1, 1) \\ x_2 &= (1, 1) \\ x_3 &= (-1, -1) \\ x_4 &= (1, -1) \end{aligned}$$

We want to do K-means clustering using Euclidean distances when $K = 2$. We start by randomly picking two points as the cluster centroids.

(a) What are all possible clustering results?

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

(b) Among all possible clustering results, which have the smallest cost, as measured in total distance (defined as follows)?

We are given points $x_1, \ldots, x_4 \in \mathbb{R}^n$ and we wish to organize these points into 2 clusters. This amounts to choosing labels $\ell_1, \ldots, \ell_4$ for each points, where each $\ell_i \in \{1, 2\}$. The total distance is defined as

$$\sum_{i=1}^{4} d(x_i, \mu_{\ell_i}) \tag{1}$$

where μ_c is the center of all points x_i with $\ell_i = c$ and $d(x, y)$ is the Euclidean distance between the points x and y.

Solution: The last four clustering results have the smallest cost.

(c) In a situation like above, how could you modify the K-means algorithm to output a clustering result that has relatively small total distances (defined above)?

Solution: Try multiple random initializations, and choose the result with the smallest total cost to output.

3. Suppose we take the (full) SVD of a size 12000 x 30 matrix A. Give the sizes and special properties of the resulting matrices.

Solution:

$$A = U\Sigma V^T$$

U is 12000 x 12000
Σ is 12000 x 30
V is 30 x 30
U and V are unitary matrices (i.e. rotation matrices), meaning their columns are orthogonal (mutually perpendicular) unit vectors.
Σ is a diagonal matrix.