Computer Vision
Homework 08: Intro to Recognition
CS 670, Fall 2019

Name: Kunjal Panchal
Student ID: 32126469
Email: kpanchal@umass.edu

November 9, 2019

Contents

1 Problem 1

A CHANCE PERFORMANCE is what you would obtain if you performed at random. For example, if you answer a multiple choice test randomly, you will (on average) get a score that is not zero. So, being only just above chance performance means "being only marginally better than random". Thus, it is not very good.

Suppose one has a BALANCED CLASSIFICATION problem (50% of 0's and 50% of 1's). In such a case, the so called chance-level accuracy of classifier would be 50%.

Model-A is trained to classify pictures of dogs versus cats. Its accuracy on the cats versus dogs test set is 55%.

In a two-class (or four-class) classification problem, the chance levels are thus 50% (or 25% respectively).

So the Chance performance of Model-A is 50%, but the accuracy of classifier is 55%.

∴ Model-A performs 5 percent-point better than random guessing. Thus, we can say that the model accuracy increased by 10% [((55-50)/50)*100% = 10%] or 5 percent-points, which is marginally only a little better.

Model-B is trained to classify digits 0,...,9; total 10 digits. Its accuracy on digits test set is 55%.

So the Chance performance of Model-B is 10% [100%/#classes = 100%/10 = 10%], but the accuracy of classifier is 55%.

∴ Model-B performs 45 percent-point better than random guessing. Thus, we can say that the model accuracy increased by 450% [((55-10)/10)*100% = 450%] or 45 percent-points, which is marginally significantly better.

2 Problem 2

2.1 (a) No Compression

Each pixel has 3 values: the values of Red, Green and Blue channels.

And each of those channel values are stored with 8 bits of precision.

Thus, each pixel stores $3 * 8 = 24$ bits of information.

So, N pixels will store $24N$ bits of information.

$\boxed{\therefore \text{ To transmit whole image in its original form takes } \textbf{24N} \text{ bits.}}$

2.2 (b) Identity of Cluster-Assignments

If we run K-means algorithm on the image containing N pixels, each pixel would be assigned to **ONE** out of **K** clusters.

The RGB values of each pixel will be the same as the cluster centroid of the cluster the pixel is in.

Thus, there will be at max, K values of each of Red, Green and Blue channels.

One pair of RGB values take $3 * 8 = 24$ bits to get stored.

K values will have $24K$ bits of color informations.

And each pixel will have **ONE** value associated with it. [it's cluster identity].

So, there will be total N pixel values transmitted, and $24K$ bits of color information.

Each of the N pixels needs to specify which of the K clusters it is using, so the value of bits will be $Nlog_2K$

As we need log_2K bits to represent K clusters. And we need to represent N pixels' identity.

$\therefore$ To transmit only the identity of the cluster-assignments of each pixel, the cost of transmission is **N** log_2**K**;
To pass it with color information [codebook] of each cluster, it takes **N** log_2**K + 24K** bits.

2.3 (c) Codebook Size

As already shown in section 2.2, the each codebook entry will have $3 * 8 = 24$ bits for color information.

And there will be K such entries in the codebook.

$\therefore$ To pass the codebook with color information of each cluster, it takes **24K** bits.

3 Problem 3

Suppose we want to perform LOSSY COMPRESSION of some real-valued pixels, N. A very simple approach to this is to use vector quantization.

The basic idea is to replace each real-valued pixel N with a discrete symbol $\mu_k \in 1, ..., K$, which is an index into a codebook of K prototypes.

Each data vector is encoded by using the index of the most similar prototype, where similarity is measured in terms of Euclidean distance.

We can define a cost function that measures the quality of a codebook by computing the RECONSTRUCTION ERROR OR DISTORTION it induces:

$$\frac{1}{N}\sum_{i=1}^{N}||x_i - \mu_k||^2$$

We perform K-mean clustering for $K \in 2, 4, 8, 16$ and quantize the RGB pixel values with the mean vector of that pixel's assigned cluster.

3.1 Quantized Images

Color quantization is the process of reducing the number of distinct colors in an image.

Normally, the intent is to preserve the color appearance of the image as much as possible, while reducing the number of colors, whether for memory limitations or compression.

Any given 24-bit RGB image has 256 x 256 x 256 possible colors. And sure, we can build standard color histograms based on these intensity values.

But another approach is to explicitly quantize the image and reduce the number of colors to say, 16 or 64. This creates a substantially smaller space and (ideally) less noise and variance.

Figure 1: Quantized image when $K = 2$

Figure 2: Quantized image when $K = 4$

Figure 3: Quantized image when $K = 8$

Figure 4: Quantized image when $K = 16$

3.2 Reconstruction Error

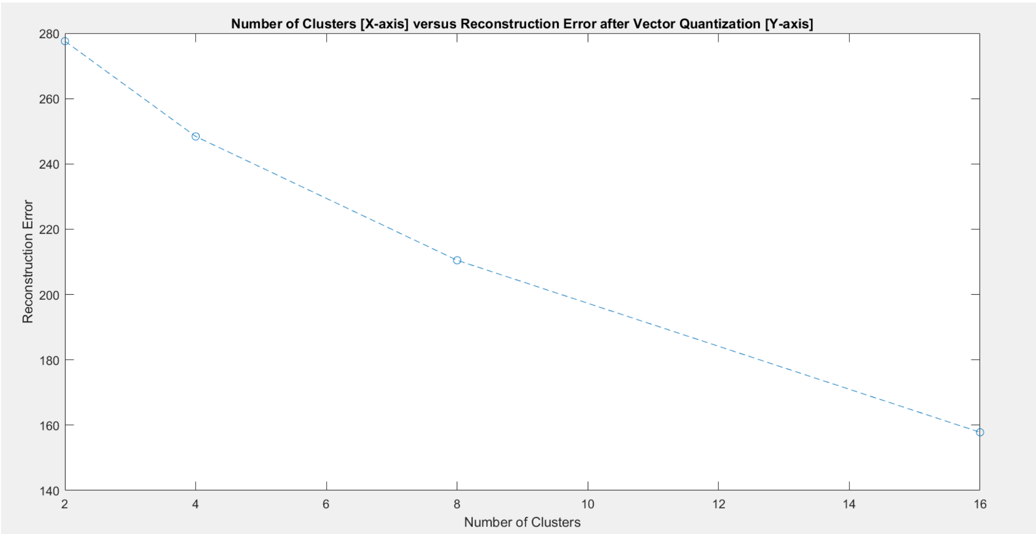

Figure 5: Reconstruction Error

References

[1] Subhransu Maji, *Image Alignment* https://www.dropbox.com/s/5flemuwc73wv7o9/lec11\%2B12\%2B13_image_alignment.pdf?dl=0