CS 675 – Computer Vision – Fall 2007

Instructor: Marc Pomplun

Assignment #4

Posted on November 28
Programming question (Q1) due by December 21
Other questions due by December 4 before class

Question 1: The Man Who Mistook His Watch for a Boat

In this final work on your computer vision program, your task is to add an object classification function. Create a button labeled "Object Classification." When the user presses this button, the image currently loaded and shown on the left side is being analyzed and classified. A window will pop up saying whether this image was classified as a face, a motorbike, a watch, or a boat.

To implement this function, please use the images in ImageDatabase.zip available from the course homepage. It contains 100 images for each of the four classes. Your classifier does not need to be able to recognize any face or boat, but just those of the type shown in these images. Please try to find image features that provide information on the class of the image, i.e., that tend to have different values for different classes (face, motorbike, watch, or boat). You can use any image processing and feature detection techniques you like. It would be a good idea to write a function that reads all 400 images and computes a set of features – the ones you are currently investigating - for all of them. Once you have identified a suitable set of features, use these features to classify new images.

As we discussed in class, one simple way to do this might be to divide your image into rectangular pieces and construct a feature vector consisting of, for example, the average intensity in each piece. So if you divide each image into 5 by 5 pieces, this method will give you a 25-element vector describing the distribution of intensity in the image. You can combine multiple different features and their distributions to form even larger feature vectors. Then you just need to find a way to determine for the vector of a new image whether it is more similar to the vectors of the face, motorbike, watch, or boat images. To do that, you can define a distance metric for your vectors, for example, the Euclidean distance in their vector space. Feel free to use any features that you like, maybe edges, Fourier coefficients, or Hough transform parameters.

Your final program should not perform the file reading and feature computation for all 400 database images during runtime. If the classification of an image takes more than 30 seconds, it will be difficult for me to test your program thoroughly.

Your algorithm does not have to be perfect, but it should yield a proportion of correct classification that is clearly above 25% (75% would be good), otherwise it is not a useful classifier. I will use different images taken from the same source to test your algorithm. Describe in detail which features you compute, how you determined them, and how you use them to do the classification task.

Question 2: The Failure of k-Means

Usually, the k-means algorithm provides good results, and it is a very popular clustering method due to its simplicity and efficiency. However, it can sometimes yield results, i.e., group input data into clusters, which are quite different from what a human observer would consider the optimal clustering.

Construct a configuration with at least 3 cluster centers and any number of data points that would lead to such an implausible clustering. You can just show the final, converged state or several snapshots showing the configuration at different points in time during the computation. You do not have to provide exact coordinates and computations of means, but just draw one or more sketches.

Question 3: Differential Motion Analysis

Below you can see the first and the last frame of a (not too exciting) movie:

0	0	0	0	0	0	0	0
0	100	100	0	0	0	0	0
0	100	100	0	0	0	0	0
0	0	0	0	0	0	0	0

First frame

0	0	0	0	0	0	0	0
0	0	0	0	0	100	100	0
0	0	0	0	0	100	100	0
0	0	0	0	0	0	0	0

Last frame

(a) There are a total of 5 frames, and in each frame the gray square moves one pixel to the right (no surprises here). Compute and write down the cumulative difference image of this sequence with $a_k = 0.2 \cdot k$ for $k = 1, \ldots, 5$.

(b) This time we have the same first and last frames; however, the sequence is only 3 frames long, with the gray square moving two pixels to right in each frame. Compute and write down the cumulative difference image of this sequence with $a_k = 0.2 \cdot k$ for $k = 1, \ldots, 3$.