Classifying Actions

Programming Assignment #7
CS253 Spring 2014
Due 6:00pm Wednesday, April 6th, 2016

Motivation

Now it time to achieve the main goal of this semester's project: recognize actions in videos.

Task

In PA6, your input was two Kinect file names and a distance measure. In PA7, your program takes an arbitrary number of inputs. The first input is the name of the 'long' Kinect video. This is the video you are looking for actions in. This is followed by an arbitrary number of filenames, each of which contains a short Kinect video. Finally, the last argument specifies a distance measure (as in PA6). Your task is to compare every short video to the long video using the distance measure specified, and to output the name of the closest (most similar) short video to std::cout.

You should output nothing to std::cout except the name of the closest video. It is an error if any of the 'short' videos are longer than the target 'long' video, and it is an error if there are no short video filenames or the last argument is not a valid distance measure.

Submitting your program

You will submit your program through Canvas. You should submit a single tar file, and this file should contain all the source files of your program *and a makefile*, but no object or executable files. The makefile should create an executable program called PA7. To grade your programs, the GTAs will write a script that untars your file in a clean directory, runs 'make', and then tests PA7 on novel inputs. If your program does not compile, whether because of an error in your makefile or an error in your code, you will receive no credit on any of the test cases. Note that as always, we will test your code on the department Linux machines.

Grading

As always, most of your grade will be determined by how well your program performs on (novel) test cases. However, a small amount of points are awarded for other factors, such as whether your code compiles. For PA7, like PA7, we may use valgrind to check for memory errors. Programs that leak memory, read uninitialized memory, access out-of-bounds data, or have similar memory problems will loose points.

Hints

1. This assignment is mostly like calling PA6 multiple times.
2. Be aware that PA8 will be graded on efficiency
3. As before, I strongly suggest writing unit tests. It is getting harder and harder to verify your code using only end-to-end testing.