PROJECT REPORT

Project Partners :

Jaspreet Singh – jaspreet@usc.edu (USC Id:7464259281)
Vishnu Vardhan Vedula – vedula@usc.edu (USC ID:6920265978)

Video Summarizer:

We have used two hashmaps – shotHashMap and frameHashMap of type <Long, ArrayList<Double>>. We store information about all frames in frameHashMap and about in shotHashMap.
We will look to display 120 non-overlapping frames for every key frame, unless there is an overlap.

Steps For Compression

1. To begin with, we use the input from the user – wave file path, video file path and the shortening ratio.
2. Based on the required shortening ratio we determine our thresholds for selecting key frames(KFs) generating new shots and scenes
3. Once we get input files, we compute scores of the files based on the metrices of color(entropy), motion and audio volume.
4. On the first pass, we read the audio file to populate our frameHashMap with each entry of the audio volume, corresponding to one frame of the video.
5. In the next pass, we now begin the creation of shots and scenes.
    - For this, we begin with calculating the color histogram for every frame. We vary the number of bins of the histogram according to compression required. We calculate the entropy based on the histogram.
    - Then, we read the audio volume from frameHashMap for each frame.
    - For efficiency, we only calculate the motion vector when one of the criteria among entropy or audio volume is greater than the threshold.
    - The frame is considered key frame if at least two of the above three criteria are satisfied.
    - Creation of a new shot is done only when the difference between the entropy values of two successive key frames is greater than 70 percent.
6. Now we have a list of all key frames from various shots. If the required number of key frames to be displayed is equal to the number of frames that we have (we consider an error threshold value of 4 percent higher or lower to that of actual value – we raise it to 10% if compression requirement is low, say, input factor is 0.8), then we do not manipulate with our set of frames. If the value of the required number of frames is lower or higher than that of the frames that we have, we use the following algorithms to either increase or

decrease the number of frames. In these algorithms, to begin with, we sort shots by score.

Algorithm to increase shots :

- Start with shot of lowest score : (we do this because in case of extra time we want that to go the highest rated shot first).
- We'll look for Key Frame that give us 120 frames already as they are only ones who we can extend .
- We multiply this with ratio to be increased: if this number does not lead to overlap, we extend that time. Else, we save difference of frames to be further added and saved for addition to the next Key Frame in shot.
- If we still have time, we put that at end of shot overlap permitting, if shot can't get all in, we move that difference to new shot , we take all possible extra frames we can get.

Algorithm to decrease shots:

- Start from the shot with the smallest score.
- Remove all shots for which the length of the frames is less than 60.
- Continuously check the shot length and stop if required compression has reached.
- Find Key Frames with the length less than 20.
- Continuously check the shot length and remove if length is now<60 and stop if required compression has reached.
- Remove Key Frame with lowest score from shot .

7. Now, we have the final list of video and audio frames that are to be displayed.

Algorithms :

1. Shot boundaries – This is done based on the change of color entropy. We create a new shot if entropy difference is greater than 70 percent.
2. Color-base selection – We create new key frame in same shot if change in entropy is greater than 30 percent and less than 70 percent.
3. Motion vector analysis – We use the Logarithmic search method to calculate the motion vector between two frames.
4. Audio levels – To get volume, we take average of normalized data in all audio frames needed to make one video frame, which is = Audio_Fps / Video_fps = 22050/24. Now if this volume is too high than a new keyframe is setup, and also if its changed too much from last key frame