# Algorithms

## Preprocessing

1. To save data and memory space and expedite the processing time, convert the color of each pixel in each frame for all videos in the database.
2. Calculate the mean value of the gray scale of each frame for all database videos.
3. Convert the mean of each frame to a md5 hash code and store the hash codes extracted from all the frames for all videos into files in order for future quick access.
4. Also, starting from the 0,1,2,...,9th frame, store the hash code every 10 frames, so we are getting 10 separate files stored for each database video respectively. Each of these files is 1 frame off its preceding or succeeding files, and they all together make up the whole large file obtained from step 3. This step is efficiently useful for the initial step when comparing with the data from query videos.

## Query Matching

1. Use the same method as used in step 3 from preprocessing to acquire a complete array of hash codes for all the frames of the query video.
2. Sample the frames with a ratio of 1:10 to keep only 1/10 of the hash codes.
3. Compare the hash code array obtained from step 2 with the 10 separate arrays obtained from step 4 of preprocessing each one by one to check if the 1/10 hash codes contain an exact copy of the array obtained from step 2 for the query video. Since both arrays include only 1/10 of the data, this step can be executed very quickly.
4. If there is an exactly match, record the starting index of the frame in the 1/10 hash code array for the database video, then the original position of the first frame of the query video should be located in the matching database video at a frame index of 10 * recorded starting index + offset. The offset is the initial frame number selected at the beginning of step 4 of preprocessing.
5. After we have got the corresponding frame index in the original database video regarding the first frame of the query video, we just need to compare all the frames for both videos one by one to make sure this is a correct match, by comparing the array obtained from step 1 for the query and step 3 of preprocessing starting from the corresponding frame index.
6. If all hash codes match for the whole length of the query video, we determine that this specific database video contains an exact same part as the query video.
7. If not found, iterate to check other videos in the database accordingly.
8. Record the time elapsed during the whole process of searching and matching.