Elimination of Duplicate Videos in Video Sharing Sites
Narendra Kumar S, Murugan S, Krishnaveni R

Abstract - In some social video networking sites such as YouTube, there exists large numbers of duplicate video in the database. It is very difficult for the users to open each video and find whether they are duplicate. If a user is having limited usage of internet connectivity, it will be a waste of cost as well as time to view the duplicate videos. Hence in this project, to balance the speed and accuracy aspects, we combine the contextual information such as time duration, number of views and content information such as color and local points to achieve real-time duplicate video elimination. We are going to select a group of similar videos as dominant group based on time duration and select a seed video from the dominant group. The videos are segmented into frames in the form of JPEG images and the frames of the seed video are compared with the frames of the other videos. The duplicate videos are thus eliminated from the database. The videos used are in the .avi format. We create a model social video networking site and use some of the videos present in the database to illustrate the example for duplicate video elimination using content information and context information. This is a very effective method with just a slight loss in the performance.

Keywords - Content, Context, Dominant video, Duplicate, Frame comparison, Web Video.

I. INTRODUCTION

With the exponential growth of social media in Web 2.0, the huge volume of videos being transmitted and searched on the Internet has increased tremendously. Users can capture videos by mobile phones, video camcorders, or directly obtain videos from the web, and then distribute them again with some modifications. For example, users upload 65,000 new videos each day on video sharing website YouTube [1][2]. Among these huge volumes of videos, there exist large numbers of duplicate and near-duplicate videos.

Near-duplicate web videos are identical or approximately identical videos close to the exact duplicate of each other, which have similar time duration/length, but different in file formats, encoding parameters, photometric variations (color, lighting changes), editing operations (caption, logo and border insertion), and certain modifications (frames add/remove). A video is a duplicate of another, if it looks the same, corresponds to approximately the same scene, and does not contain new and important information.

For videos having remarkably different time durations, e.g. full version and short version, by definition, they will not be regarded as near-duplicates. A user would clearly identify the videos as “essentially the same.” Two videos do not have to be pixel-wise identical to be considered duplicates. A user searching for entertaining video content on the web, might not care about individual frames, but the overall content and subjective impression when filtering near-duplicate videos [3]. Exact duplicate videos are a special case of near-duplicate videos, which are frequently returned by video search services [4].

II. RELATED WORK

In the existing work, the frames of seed video extracted from middle are checked for similarity with the thumbnail images of the remaining videos in the Dominant Version set. If the thumbnail image matches with the frames of the seed video segmented frames, then the videos are identified as duplicate videos.

A hierarchical approach combining global signatures and local feature based pair wise comparison were used to detect duplicate web videos. Global signature[5] is nothing but computing color histogram for each frames present in the video. Since there is possibility for different images producing same histogram, this method is not efficient. For videos which cannot be determined novel or near duplicate using global signature, the similarity of frames are measured by pair wise comparison of frames from two videos. Then the redundancy is determined by comparing the ratio of number of similar frames.

In pair wise comparison among frames, salient regions in each frame is extracted using local point detectors. The Difference of Gaussian is an operator used to detect edges in an image. The image is smoothed before applying the Gaussian operator to remove the noises. Salient Regions are regions in the image that are either brighter or darker than the surrounding. This way of identifying duplicate videos by pair wise comparison among frames is time consuming.

Although local point based feature extraction is reliable, matching large number of local points is a difficult problem. Since the number of frames could exceed one hundred for a four minute video, and the number of local points could reach many hundreds per frame. In the work, a text file is associated for each video. The similarity of the frame is found by comparing the text files of the video. Two methods were explored to find a short video clip in a stream of video. One approach is ‘baseline method’ i.e. using histograms to compare images in the clip to those in the video, to find the
regions where the sequences of images are most similar. This method works well when video and clip are in the same format, but is slow, when formats vary. The other approach was a ‘new method’ which is based on the sequence of cuts or edits in the video. This method is very much faster for videos having higher edits per hour, but effectiveness is poor when the number of cuts in the clip is too small.

Also two new approaches were proposed for video matching. In one of the approach simple measure is used to find how far particular regions have been shifted from one image to next. In another approach, the change in color from one image to another is computed. Many more approaches for content based video retrieval were also proposed. However comparing video sequences using color histogram for each frame tends to be more efficient, and in some cases more accurate method for retrieving similar video material.

The global feature represents the name, production date and some comments of video. The local feature specifies the content of the video. Two video sequences are said to be similar if they share a significant portion of visually similar frames. In this approach a Video Signature Method is followed. Here a small summary or signature for each video is formed, which consists of a small number of its frames that are most similar. By comparing the signature the similarity of the video is detected.

The videos are checked for similarity based on the video sequences. The video sequences are checked for similarity based on the levels of temporal resolution, the temporal order, and the temporal duration. In the levels of temporal resolution, video sequences may be the shot, scene or a segment is compared for similarity. In temporal order, the similarity between two video sequences is measured by the number and amount of similar image feature. In temporal duration, the shots with similar duration are compared for similarity. Compared to above work our work differs in the following way.

1) The frames are compared based on the edge value detected from the images and by based on the mean value of the images.

2) Pair wise comparison of frames is performed rather than comparing the thumbnail nail of the images with the frames of the seed video.

### III. SYSTEM ARCHITECTURE

The main aim of this project is to identify and eliminate duplicate videos on the Web with the help of context and content information of the video.

#### A. Overall System Architecture

The architecture depicts various process involved in the duplicate video elimination. The context information of the video such as time duration, thumbnail image and view count of the video are used to precede the process. Figure 1 represents the overall system architecture.

#### B. Description of Modules

The system mainly consists of three modules.

(i) Dominant Version Identification

(ii) Seed Video Selection

(iii) Frames Comparison

The Dominant Version Identification module and Seed Video Selection module makes use of context information. The Frames comparison module makes use of content information.

1. Dominant Version Identification:

**INPUT**: Query Related Video List

**OUTPUT**: Set of Video’s having Similar Duration

The query related video list are taken as input. Fig.2 represents Dominant Version Identification module.

![Fig 2 Dominant Version Identification](image)

For each query, dominant version identification is performed by collecting the time duration of all videos [5]. That is, videos having similar time duration are gathered as a set. If ‘d’ is the multiple duration for a query, then videos having duration of ‘d ± α’ is gathered, where α can be the range of 2 to 4sec.

**Algorithm:**

Step 1: For the given query collect the relevant videos from the database.

Step 2: From the relevant videos gather videos having more or less similar time duration.

Step 3: Stop

2. Seed Video Selection:

**INPUT**: Set of Video’s having Similar Duration

**OUTPUT**: Video Having Large Number of Views

Fig 3 represents Seed Video Selection. The set of videos having similar duration are taken as input. The thumbnail images of the videos present in the Dominant Version Set are checked for similarity using the Euclidean distance of the color histogram.
An arbitrarily selected video is used as a reference video. The distance\[6\] between the thumbnail image of the reference video and remaining videos are calculated as:

$$\text{Distance} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \left( (r_1-r_2) \cdot (r_1-r_2) + (g_1-g_2) \cdot (g_1-g_2) + (b_1-b_2) \cdot (b_1-b_2) \right)$$

where \(M,N\) represents height and width of the image, \(r_1, r_2, g_1, g_2, b_1, b_2\) represents the RGB values of the images being compared.

**Algorithm:**

Step 1: The thumbnail images of videos in the Dominant Version Set are checked for similarity.

Step 2: Images having similar thumbnail images are taken as a set.

Step 3: From the set videos having large number of views are selected as seed video.

Step 4: Stop.

3. Frames Comparison:

**INPUT**: Frames of Seed Video

**OUTPUT**: Result showing redundant videos
The seed video and remaining videos present in the Dominant Version set are segmented. The frames of seed video are taken as taken as input. The comparison of frames is shown in Fig 4.

After segmentation, the frames of the seed video are compared with the frames of remaining videos present in the set. The frames are compared based on the edge values and mean value of images.

Algorithm:
Step 1: Segment the seed video into frames.
Step 2: Take a video from the set and segment it into frames.
Step 3: Compare the frames of these videos based on the edge value and mean value of images.
Step 4: If both the results are true, then the frames are said to be similar else they are not similar.
Step 5: If all the frames are similar, then the videos being compared said to be duplicate of each other.
Step 6: Take another video from the set. Repeat step 3 to step 5.
Step 7: Repeat the above process for all the videos present in the set.
Step 8: Stop.

4. Edge Detection:

INPUT : An image
OUTPUT: Edge values of image

Edges and lines in images carry more useful information than other types of features such as texture, color. Edge detection is a terminology in image processing, mainly for feature detection and feature extraction [7]. This aims at identifying points in a digital image at which the image brightness changes sharply. The discontinuities in image brightness are likely due to discontinuities in depth, discontinuities in surface orientation, changes in material properties and variations in scene illumination.

Algorithm:
Step 1: Take a image.
Step 2: Compute the RGB value for each pixel in the image.
Step 3: From the RGB value compute the luminance value for each pixel.
Step 4: Compare the luminance value of a pixel with its neighbor value.
Step 5: If the luminance value exceeds the threshold value, then an edge is considered at the respective pixel position.
Step 6: Repeat this procedure for all the pixels present in the image.
Step 7: Repeat step 2 to step 6 for all the frames present in the video.
Step 8: Stop

5. Mean Computation:

INPUT : An image
OUTPUT: Mean value of image

The mean value of an image is used along with edge value detection to perform the comparison. The mean of an image is the ratio of sum of its pixel value to its image dimension.[6]

\[
\text{Mean} = \frac{\text{Sum of the pixel values}}{H \times W}
\]

Where, \(H, W\) are height and width of the image.

Algorithm:
Step 1: Take an image.
Step 2: Compute the RGB value for each pixel in the image.
Step 3: Find the sum of pixel value.
Step 4: Compute the ratio of pixel value to the image dimension.
Step 5: Stop.

The edge values and mean value computed are used to perform pair wise comparison of frames of the video.

IV. CONCLUSION AND FUTURE WORKS

The duplicate video elimination of the video with help of content and context is performed. The time duration acts as a filter to avoid unnecessary computation. If the difference of time duration is large, the videos are not compared and regarded as dissimilar.

With the thumbnail images, the distance of the color histogram is computed. These steps lead to pair wise comparison of frames of videos. The mean value of the frame and edge value are used to check the similarity.

This project may be extended in the future, by focusing on the following aspects. Using the user groups, usernames, related/relevant videos, user relationship, and relevance relationship among videos as the contextual information for web video search and retrieval on the Social Web. Using this
information to identify and eliminate duplicate videos on the Social Web.

REFERENCES


S.Narendra Kumar, Chennai, 06.08.1989, M.E computer science and engineering, School of computing sciences and engineering, Hindustan University, Chennai, Tamil Nadu, India.
S.Murugan, Chennai, 09.09.1988, M.E computer science and engineering, School of computing sciences and engineering, Hindustan University, Chennai, Tamil Nadu, India.
R.Krishnaveni, Chennai, 25.02.1969, Associate professor, School of computing sciences and engineering, Hindustan University, Chennai, Tamil Nadu, India.