Special Issue Published in International Journal of Trend in Research and Development (IJTRD), ISSN: 2394-9333, www.ijtrd.com

Knowledge Discovery of Exact Image / Data Access through Internet Using Optimal Content Based Video Indexing With Motion Compensated Predictive Video Compression Technique

[1]Dr.V.Joseph Emmanuvel and [2]Dr.S. Vetrivel
[1,2]Assistant Professor, Department of Computer Science
[1]Mahendra Arts and Science College, Kalippatti, TamilNadu, India
[2]Joseph Arts and Science College, Thirunavalur, Villupuram, TamilNadu, India

Abstract- Existing Video Indexing Models are analyzed and a practical approach to the optimal Video Indexing is introduced. It is studied under all the phases of video indexing processes like segmentation, indexing, database storage, query based access, browsing and video clip retrieval, etc… The main aim is to easily parse the video stream into meaningful scenes, maintain them in an effective database with minimal data repetitions, efficient query handling and user friendly browsing capabilities. Conceptual Graph, Motion Estimation, Mean Absolute Frame Difference, Displaced Frame Difference, Dublin Core and other important existing techniques are utilized in this model. The main aim is to reduce the memory storage of video clippings without visible loss in quality by using a predictive video compression technique. Today almost all video clippings face a compromise between their quality and memory size. Even Video clippings are not advised to include in the web pages because of their downloading time and its memory size. To try to rectify it and to take positive measures to convert a video clipping file similar to *.swf (flashplayer's schockwave files) is the aim of this presentation.

I. INTRODUCTION

In fields like Meteorology (satellite image sequence processing), Road Traffic Surveillance, Medical Imaging, TV News Broadcasting, Movies, Documentaries, Distance education Programmes (eg. UGC Programmes in DD1)etc… include Video Documents, which are increasing in enormous volume such that manual indexing, annotations, database management, search and retrieval of these data at the time of user's demand is almost impossible. Technically speaking, if automation is achieved, then it may be called as CONTENT BASED ACCESSING of Video. One important A.I. concept in the field of Multimedia is that Computer aided vision techniques have to answer the following query of end users even without seeing it: - "WHAT IS IN A VIDEO CLIPPING ?", "WHAT IS HAPPENING IN A VIDEO CLIPPING ?" Or "WHAT IS THE SUMMARY OF A VIDEO CLIPPING?" These questions are reasonable to be asked by an end-user. But difficult to apply a computer aided technique. Let us concentrate on the first question: - "WHAT IS IN A VIDEO CLIPPING?" using the Content based video Indexing. Video indexing is the process of attaching content-based labels to video. Existing literature on video indexing implicitly defines video indexing as a continuous process of extracting video data from the raw video streams having the temporal location of a feature and its value. The indexing effort is directly proportional to the granularity of video access. Today Users demand finer grain access to video. So, automation of video indexing process becomes essential. The entire discussion below only concentrates on Video data in a video clipping and there is a separate field of Video Indexing via Audio stream by using voice recognition, Background audio music detection and indexing, etc…, which are not touch through the course of this presentation. The storage of uncompressed graphics, audio and video is not feasible even with DVDs (which can store about 20 GB of information). Uncompressed graphics also require high bandwidth for transmission, meaning that the number of bits to be transferred per second is very high. This we may probably experienced before, especially when we are streaming a movie online and are trying to load another page with lots of graphics on another window at the same time - you notice that the graphics don't load as fast as they would if you weren't streaming the movie, because the internet connection probably allows to transfer (upload or download) only a fixed number of bits per second, and the movie is taking up most of this allotted bandwidth. This also happens if we are using software like Kazaa or download accelerator to download an mp3 file when we try to upload a movie file from the same system. if there are too many users trying to access the same files, most of the bandwidth is expended in them uploading files and download becomes slow.

II. WITHOUT COMPRESSION

- A text file (let's say it has allotted 1 byte to each character, and there are 55 lines per screen with 80 characters per line) which is about ten pages will be about 44 KB in size
- A still image (even low resolution images, 640x480 pixels, with more than 1 byte required per pixel) would be about 30 KB (the size would be a lot larger if it was a true color image, where you use 8 bytes for each of three colors per pixel!)
- Each second of an audio signal (sampled at 44.1 kHz and quantized at 16 bits per sample, 2 channel stereo) would require about 180 KB, and Falco's Rock Me Amadeus (length = 3:15) would be about 35 MB in size - this, in fact, is about the size of the .wav or .cda files that exist on your music CDs. Compare this to the mp3 file, which is about 3 MB!
- NTSC (the format used in the United States for TV broadcasting) TV signals have bandwidths of about 10 MBps. This means that a TV episode of about 25 minutes will need 15 GB!
- HDTV (High Definition TV) doubles the number of lines from NTSC and uses an aspect ratio of 16:9 compared to the NTSC value of 4:3, a factor of 1.33 higher, and has a bandwidth of about 50 MBps (it can be as high as 1 GBps) this only makes matters worse, even this bandwidth is too high to actually achieve for regular transmission.

Digital video (let's say its low resolution again, 640x480 pixels), running at 29.9 frames per second (fps), encoded with

True Colour, would require a bandwidth of about 900 MBps! A typical Britney Spears music video (length about 3.5 minutes) needs about 190 TB!!

So, there are three things to think about here -

1. How is it that we can reduce the size of a file and still retrieve all of the original information on uncompressing?
2. Why is there a need for compressing audio, video and images? I heard that TV broadcasts also consist of compressed data! Is this true? How does that work?
3. Okay, I understand how this can work for text files, but what do you do in case you have images, audio and/or video?

III. VIDEO INDEXING

The entire Video Indexing Process can be classified as:-

1. Video Parsing
2. Video Indexing
3. Database Management &
4. Retrieval and Browsing

Basically Content Based Video Indexing can be classified as:-

1. Low Level Indexing
2. High Level Indexing and
3. Domain Specific Indexing

Video parsing is the process of detecting scene changes or the boundaries between camera shots in a video stream. The video stream is segmented into generic clips. These clips can be the elemental index units stored in a video database, just like a word in a text database. Then, their key frames will represent each of these clips visually. To reduce the requirements for large amount of storage, only the key frames of these clips will be stored into the database. There are two type of transitions namely, abrupt transitions (or) camera break and gradual transitions e.g., fade-in, fade-out, dissolve, and wipe. Using these transitions also Video Stream can be clipped to pieces. Indexing tags will have small pieces of video clips, while the system inserts them into the database. Along with the clips indexing tag also includes information based on a knowledge model that guides the classification according to the semantic primitives of the images. Indexing is done by using the image itself. Any semantic descriptors provided in the model. For this, two types of indices, text-based and image-based, are needed. The text-based index is typed in by human operator based on the key frames using a content logger. The image-based index (small Gif file) is automatically constructed based on the image features extracted from the key frames. Low Level Indexing deals with techniques that provide access to video, based on properties like color, texture etc. High Level Indexing uses a set of predefined index terms for annotating video. The index terms are organized based on high-level ontological categories like action, time, space, etc. The high level indexing techniques are primarily designed from the perspective of manual indexing or annotation. This approach is suitable for dealing with small quantities of new video and for accessing or verifying the auto-indexed databases. Domain Specific Indexing uses the high level structure of video to constrain the low level video extraction and processing. These techniques are effective in their intended domain of application. The primary limitation of these techniques is their narrow range of applicability.

IV. DATABASE MANAGEMENT

Retrieval and browsing, where users can access the database through queries based on text and/or visual examples or browse it through interaction with displays of meaningful icons. Users can also browse the results of a retrieval query. It is important that both retrieval and browsing appeal to the user's visual perception. By using visual query, users can find video shots that look similar to a given example. In concept query, users can find video shots having specific objects or events in them. Visual query can be implemented by directly comparing low level visual features like color, texture, shape and temporal variance of video shots or their representative frames (i.e. key frames). But the concept query depends on object detection, tracking and recognition. So, fully automatic object extraction is still impossible, some extent of user interaction is necessary in this process. But as we will discuss later, manual indexing labor can be greatly reduced with the help of video analysis techniques. The driving force behind this group of techniques is to extract data features from the video data, organize the features based on some distance metric and to use similarity based matching to retrieve the video. Their primary limitation is the lack of semantics attached to the features. Video Indexing, then, can be explained how the automatic video analysis techniques, such as scene cut detection, key frame selection, visual/audio feature extraction, object recognition, text and speech identification, can be used in the content based video indexing process. There are several approaches against Content Based Video Indexing. They are:-

- Video Indexing using Motion Estimation for Serials, Movies, etc...
- Dublin Core extensions and the Resource Description Framework Video Indexing for News, Documentary, etc…
- Video Preprocessing for Video Indexing
- Knowledge based Conceptual Graph method
- Video Text Model of Video Indexing
- Optimal Content based Video Indexing – a practical approach (which we found optimal than the existing methods).

Data mining Concept is used to implicit, previously unknown, and potentially useful information from database. A sequential pattern is an ordered list (ordered by increasing event's time.) of events. The main contribution of this thesis is the development of general model of sequential patterns as well as algorithms for discovering such patterns. We investigate the generalized sequential ' pattern in which time isn't just part of constraints but also a part of discovered pattern. We have developed several algorithms to discover the generalized sequential patterns from a given temporal database.

V. VIDEO INDEXING USING MOTION ESTIMATION

This algorithm generates a video index which assumes that a video sequence has already been temporally segmented into individual shots using the edit effect detection algorithm. Then, estimates of the dominant motion between each frame pair are used to identify shots containing significant camera motion that may require more than one key frame to represent their content. A weighted directed graph can be formed for each shot where the vertices represent the frames in the shot and the weight on each edge is a measure of similarity between the contents of each frame pair. The frames corresponding to the vertices forming the shortest path through the graph are used as representative key frames for each shot. The extracted key frame give a graphic, sequential depiction of the narrative; analogous to a storyboard. Thus, a small subset of frames can be used to retrieve information from the video and enable content-based video indexing. The algorithm uses two thresholds, T_{min} and T_{max}, which define the minimum and maximum amount of overlap allowed between the contents of two key frames respectively. These parameters can be set according to user preference. The algorithm is extended to characterize and textually annotate the apparent camera motion contained within each shot to augment the indexing and searching process. It uses a top-down approach starting with a crude initial guess of the overall motion, which is then recursively refined using the Douglas-Peucker Line Simplification Algorithm.

CONCLUSION

Recent progress in digital technology has made the widespread use of compressed digital video signals practical. Standardization has been very important in the development of common compression methods to be used in the new services and products that are now possible. In this proposed optimal based video compression technique, the motion compensated prediction step is very computationally intense. Motion compensation is important because it can provide a lower data rate than using just intra frame approaches for a fixed image quality.

References

[1] Alen V.Oppenheim, Renold W.Schafer" Digital Signal Processing Principles Algorithms and Applications, Prentice Hall 1997.

[2] S.Paris and F.Durand, " A Fast Approximation of the Bilateral Filter Using a Signal Processing Approach", Proc. European Conf. Computer Vision, 2006.

[3] James S. Walker, Wavelet-Based image Compression,Sub-chapter of CRC Press book Transforms and Data Compression

[4] Barry G. Haskell, Atul Puri, Arun N. Netravali, Digital Video: An Introduction to MPEG-2, Chapman and Hall, 1997.