IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

An Key Frame Extraction using Euler Distance

Priyanka D.Khemalapure[1], Vidya V.Khandare[2], Poonam S.Shetake[3]

[*1,2,3]Dept.of Electronics and Telecommunication Engineering Bharati Vidyapeeth's College of Engineering Kolhapur, India

sachupadhye@gmail.com

Abstract

Key frame extraction has been recognized as one of the important research issues in video information retrieval .Although progress has been made in key frame Extraction the existing approaches are either computationally expensive or ineffective in capturing salient visual content. Key frame extraction aims to reduce the amount of video data, and the frame sequence must preserve the overall contents of the original video. We propose a new method for extracting key frames from a MPEG. Our proposed approach consists of two steps. In the first step, we propose a method of frame segmentation and second step is key frame extraction. In this given video sequence is segmented into number of frames, Euler distance difference method is used to extract key frames. Experimental results show that the extracted key frames can summarize the salient content of the video and the method is of good feasibility, high efficiency, and high robustness.

Keywords: Key frame, segmentation, block correlation, Euler distance

Introduction

frame extraction is the process of summarizing a video by producing a summary of salient key frames that could convey the overall content of the video to the viewer so that a viewer could understand the content of the video without actually watching the video fully. A video summary conveys the actual content of the video using few most representative frames. A basic rule of key frame extraction is that key frame extraction would rather be wrong than not enough. So it is necessary to discard the frames with repetitive or redundant information during the extraction.

Video segmentation and key frame extraction are the bases of video analysis and content-based video retrieval. Key frame extraction[1-2], is an essential part in video analysis and management, providing a suitable video summarization for video indexing, browsing and retrieval. The use of key frames reduces the amount of data required in video indexing and provides the framework for dealing with the video content[3]. A new algorithm of key frame extraction from compressed video data is presented in this paper. We analyze the features of compressed data and finally obtain the key frames.For video, a common first step is to segment the videos into temporal "shots," each representing an event or continuous sequence of actions. A shot represents a sequence of frames captured from a unique and continuous record from a camera. Then key frames are to be extracted. Video segmentation is the premise of key frame extraction, and key frames are the salient content of the video (key factors to describe the video contents).

Related Work

Different methods can be used to select key frames. In general these methods assume that the video has already been segmented into shots (continuous sequences by a shot detection algorithm, and extract the key-frames from within each shot. One of the possible approaches to key frame selection is to take the first frame in the shot as the key frame [4]. Ueda et al [5] and Rui et al. [6] use the first and last frames of each shot. Other approaches time sample the shots at predefined intervals, as in Pentland et. al [7] where the key frames are taken from a set location within the shot, or, in an alternative approach where the video is time sampled regardless of shot boundaries [6]. These approaches do not consider the dynamics in the visual content of the shot but rely on the information regarding the sequence's boundaries. They often extract a fixed number of key frames per shot. In Zhonghua et al. [8] only one key frame is extracted from each shot: the frames are segmented into objects and background, and the frame with the maximum ratio of objects to background is chosen as the key frame of the segment since it is assume to convey the most information about the shot. Other approaches try to group the key frames (in each shot, or in the whole video) into visually similar clusters.

In Arma et al. [9] the shot is compacted into a small number of frames, grouping consecutive frames together. Zhuang et al. [10] group the frames in clusters, and the key frames are selected from the largest clusters. In Girgensohn et al. [11] constraints on the position of the key frames in time are also used in the clustering process; a hierarchical clustering reduction is performed, obtaining summaries at different levels of abstraction. In Gong et al. [12] the video are summarized with a clustering algorithm based on Single Value Decomposition (SVD). The video frames are time sampled and visual features computed from them. The refined feature space obtained by the SVD is clustered, and a key frame is extracted from each cluster.

The drawback to most of these approaches is that the number of representative frames must be set in some manner a priori depending on the length of the video shots for example. This cannot guarantee that the frames selected will not be highly correlated. It is also difficult to set a suitable interval of time, or frames: large intervals mean a large number of frames will be chosen, while small intervals may not capture enough representative frames, those chosen may not be in the right places to capture significant content.

Implementation

The implemented system concentrated on frame difference approaches and key frame extraction which is used for the compression technique. This system is implemented via many stages as illustrated in figure 1 and these stages are listed below:

To fully understand the issues involved with this type of video compression it is necessary to examine each of the stages in detail. These stages can be described as:

- Frame segmentation
- Block correlation
- Euler distance
- Frame difference mean
- Frame difference variance
- Key frame extraction

A. Frame Segmentation

The current frame of video to be compressed is divided into equal sized non-overlapping rectangular blocks. Ideally the frame dimensions are multiples of the block size and square blocks are most common. Chan etc. used rectangular blocks of 16 x 8 pixels, claiming that blocks of this shape exploit the fact that motion within image sequences is more often in the horizontal direction than the vertical .

Block size affects the performance of compression techniques. The larger the block size, the fewer the number of blocks, and hence fewer motion vectors need to be transmitted. However, borders of moving objects do not normally coincide with the borders of blocks and so larger blocks require more correction data to be transmitted . Small blocks result in a greater number of motion vectors, but each matching block is more likely to closely match its target and so less correction data is required. Lallaret et al. found that if the block size is too small then the compression system will be very sensitive to noise . Thus block size represents a tradeoff between minimizing the number of motion vectors and maximizing the quality of the matching blocks. The relationship between block size, image quality, and compression ratio has been the subject of much research and is well understood.

B. Blocked Correlation

In block motion compensation (BMC), the frames are partitioned in blocks of pixels (e.g. macroblocks of 16×16 pixels in MPEG). Each block is predicted from a block of equal size in the reference frame. The blocks are not transformed in any way apart from being shifted to the position of the predicted block. This shift is represented by a *motion vector*.

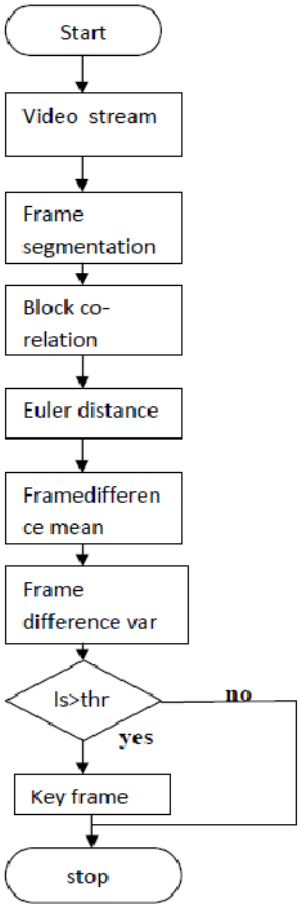

Figure 1.Flow chart of proposed algorithm

To exploit the redundancy between neighboring block vectors, (e.g. for a single moving object covered by multiple blocks) it is common to encode only the difference between the current and previous motion vector in the bit-stream. The result of this differencing process is mathematically equivalent to a global motion compensation capable of panning. Further down the encoding pipeline, an entropy coder will take advantage of the resulting statistical distribution of the motion vectors around the zero vector to reduce the output size. It is possible to shift a block by a non-integer number of pixels, which is called sub-pixel precision.The in-between pixels are generated by interpolating neighboring pixels. Commonly, half-pixel or quarter pixel precision (Qpel, used by H.264 and MPEG-4/ASP) is used. The computational expense of sub-pixel precision is much higher due to the extra processing required for interpolation and on the encoder side, a much greater number of potential source blocks to be evaluated.

Block motion compensation divides up the current frame into non-overlapping blocks, and the motion compensation vector tells where those blocks come from (a common misconception is that the previous frame is divided up into non-overlapping blocks, and the motion compensation vectors tell where those blocks move *to*). The source blocks typically overlap in the source frame. Some video compression algorithms assemble the current frame out of pieces of several different previously-transmitted frames.

Frames can also be predicted from future frames. The future frames then need to be encoded before the predicted frames and thus, the encoding order does not necessarily match the real frame order. Such frames are usually predicted from two directions, i.e. from the I- or P-frames that immediately precede or follow the predicted frame. These bidirectionally predicted frames are called *B-frames*. A coding scheme could, for instance, be IBBPBBPBBPBB.

The underlying supposition behind motion estimation is that the patterns corresponding to objects and background in a frame of video sequence move within the frame to form corresponding objects on the subsequent frame. The idea behind block matching is to divide the current frame into a matrix of 'macro blocks' that are then compared with corresponding block and its adjacent neighbors in the previous frame to create a vector that stipulates the movement of a macro block from one location to another in the previous frame. This movement calculated for all the macro blocks comprising a frame, constitutes the motion estimated in the current frame. The search area for a good macro block match is constrained up to *p* pixels on all fours sides of the corresponding macro block in previous frame. This '*p*' is called as the search parameter. Larger motions require a larger *p*, and the larger the search parameter the more computationally expensive the process of motion estimation becomes. Usually the macro block is taken as a square of side 16 pixels, and the search parameter *p* is 7 pixels. The idea is represented in Fig 2. The matching of one macro block with another is based on the output of a cost function. The macro block that results in the least cost is the one that matches the closest to current block .

C. Euler Distance

The similarity of images is measured by the distance Between two images. The smaller the distance is, the bigger the similarity is. It is important to select a suitable similarity measurement method which makes the distance of feature match with the similarity of images possibly.

So Euler Distance is adopted in this paper to find the difference between frames.The difference between frames can be calculated by

$$Diff = Xn1 - Xn \quad (1)$$

Where Xn1 and Xn are two consecutive frames.Euler distance difference can be written as,

$$\text{Euler distance} = \sqrt{\sum_{n=1}^{N} (Xn\ 1 - Xn)^2} \quad (2)$$

Where N is the total number of frames .

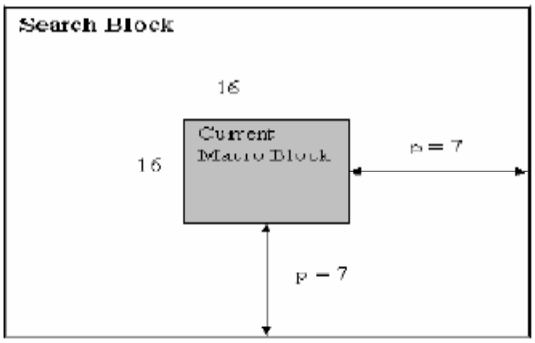

Figure 2.Block Matching a macro block of side 16 pixels And search parameter *p* of size 7 pixels.

D. Frame Difference Mean

In this method, the mean value of the frames difference is calculated. Mathematically the mean (average) is obtained by dividing the sum of the observed values of frames difference by the number of observations. Then remove the frames where the

frames difference between any consecutive frames is lower than the mean value of the frames difference.

E. Frame Difference Variance

Frame difference method carries out frame difference operationof two adjacent frames in a collection of image sequences(usually decomposed from a video). To effectively avoid the high time cost, we use inferior moving object detection of traditional frame difference method for motion detection to extract the key frame, that is, the frame corresponding to a fast moving object.

Through a series of experiments we found that compared to frame difference-Euler method, frame difference-variance method is less sensitive to noise and can adapt to various dynamic environments with good stability.Figure 3 shows frame difference variance curve.

Thus we take frame difference-variance as evaluation criteria to quantify the frame difference. To accelerate the computation efficiency, we set sampling extraction interval to extract one frame from each several frames, in which way we compress the video. Also we set the number of key frames to be extracted according to users demands. Then we subtract adjacent frames in image sequences and calculate frame difference variance in high-dimensional space. Afterwards, we draw the frame difference-variance curve and find the local maximum point in each certain length.

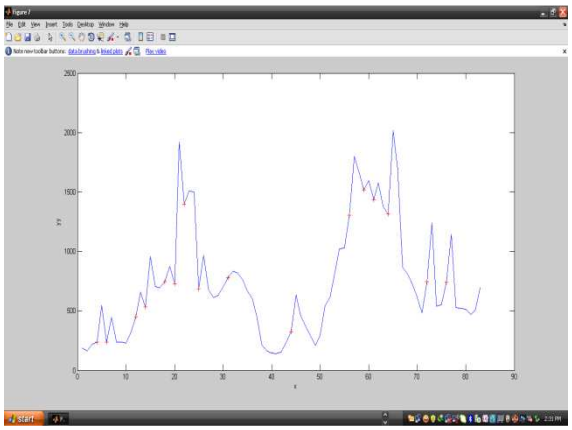

Figure 3.The frame difference-variance curve.

Experiment Test

A segment of video is selected to do the experiment. Firstly, let the first frame as a key frame, and the ratios are calculated according to equation . The frame where a ratio peak occurs is extracted as a key frame. If there are no transitions in a shot, the frames in the shot have high similarity, and there is no significant change among the characteristic curves. Then the first frame can be extracted as a key frame, and finally the key frames of the video can be obtained. The experimental result is as follows:

Figure 4.The result of the key frames extraction from video stream

Figure 5.The result of key frame extraction from video stream

The algorithm is based on shot. There is only one shot in the tested video sequence, and there are no gradual transitions and abrupt transitions. So only a shot is obtained after the shot segmentation. The first frame is extracted as a key frame, and other key frames are calculated according to macro block motion when abrupt transitions occur.

Result And Discussion

Key frame extraction aims to reduce the amount of video data, and the frame sequence must preserve the overall contents of the original video. Whether the key frames can be accurately detected and extraction is the fundamental rule to measure the validity of the algorithm. Current measurement mainly relies on eye observation. If we simply use this method, it is time-consuming in the case of large scale video data.

A video summary should not contain too many key frames since the aim of the summarization process is to allow users to quickly grasp the content of a video sequence. For this reason, we have also evaluated the compactness of the summary (compression ratio). The compression ratio is computed by dividing the number of key frames in the summary by the length of video sequence. For a given video sequence **St**, the compression rate is thus defined as:

$$CRatio(S_t) = 1 - \frac{\gamma_{NKF}}{\gamma_{NF}} \quad (3)$$

Where *NKF* is the number of key frames in the summary, and *NF* is the total number of frames in the

video sequence. Ideally, a good summary produced by a key frame extraction algorithm will present both high quality measure and a high compression ratio (i.e. small number of key frames).

Video sequences	Total frames	Key frames	Compression ratio (%)
Video 1	220	4	98.2
Video 2	1283	5	99.6

Table 1.The results of the experiment

From Table 1, the average compression ratio of the new algorithm is 99.9%. From the experiment, we can see that the representative key frames can be extracted accurately and semantically from long video sequences or videos with more transitions, reflecting the video content objectively. Figures 4 and 5 shows the results of key frame extraction with the proposed algorithm. The video depicts the process moving car, road, and tree. Five key frames are extracted from the video with the algorithm. From the key frames, the video content can be clearly acknowledged, including car, tree, road etc. The result illustrates that the algorithm is valid to segment the shot and extract the key frames and it is of good feasibility and strong robustness.

Conclusion And Future Work

We have proposed a new algorithm for key frame extraction. It compensates for the shortcomings of other algorithm and improves the techniques of key frame extraction based on MPEG video stream. We conducted a serial of experiments and the results showed that our proposed system has strong robustness and effectiveness to detect moving object in various illumination conditions and being computationally inexpensive.

The system can be enhanced by allocating more key frames representing shots that involve more action. This requires an efficient shot boundary detection method. These are the enhancements that could be added to the system in future. The system has been tested with web videos of very short duration(less than one minute) that have no major change in background. However, long videos with multiple shots and scenes will need appropriate shot segmentation step.

References

[1] D.Feng, W.Siu and H. Zhang, “Multimedia information retrieval and management: Technological Fundamentals and Applications, ” Springer, pp.44, 2003.

[2] Costas Cotsaces, Nikos Nikolaidis, and Ioannis Pitas, “Video shot detection and condensed representation: a review,” IEEE Signal Processing, vol. 23, no. 2, pp. 28-37, 2006.

[3] T. Liu, H. Zhang, and F. Qi, “A novel video key-frame-extraction algorithm based on perceived motion energy model,” IEEE Transactions On Circuits And Systems For Video Technology, vol. 13, no. 10, pp. 1006-1013, 2003.

[4] Tonomura Y., Akutsu A., Otsugi K., and Sadakata T. VideoMAP and VideoSpaceIcon: Tools for automatizing video content. Proc. ACM INTERCHI ’93 Conference, 1993:131-141.

[5] Ueda H., Miyatake T., and Yoshizawa S. IMPACT: An interactive natural-motion-picture dedicated multimedia authoring system. Proc. ACM CHI ’91 Conference, 1991:343-350.

[6] Rui Y., Huang T. S. and Mehrotra S. Exploring Video Structure Beyond the Shots. Proc. IEEE Int. Conf. on Multimedia Computing and Systems (ICMCS), Texas USA, 1998:237-240.

[7] Pentland A., Picard R., Davenport G. and Haase K. Video and Image Semantics: Advanced Tools for Telecommunications. IEEE MultiMedia,1994;1(2):73-75.

[8] Zhonghua Sun, Fu Ping. Combination of Color and Object Outline Based Method in Video Segmentation. Proc. SPIE Storage and Retrieval Methods and Applications for Multimedia, 2004;5307:61-69.

[9] Arman F., Hsu A. and Chiu M.Y. Image processing on Compressed Data for Large Video Databases. Proc. ACM Multimedia '93, Annaheim, CA,1993:267-272.

[10] Zhuang Y., Rui Y., Huang T.S., Mehrotra S. Key Frame Extraction Using Unsupervised Clustering. Proc. of ICIP’98, Chicago, USA, 1998;1:866-870.

[11] Girgensohn A., Boreczky J. Time-Constrained Keyframe Selection Technique. Multimedia Tools and Application, 2000;11:347-358.

[12] Gong Y. and Liu X. Generating optimal video summaries. Proc. IEEE Int. Conference on Multimedia and Expo, 2000;3:1559-1562.