SPATIAL ANALYSIS IN KEY-FRAME EXTRACTION USING VIDEO SEGMENTATION

J. Calic, B. T. Thomas
{janko, barry}@cs.bris.ac.uk
Dept. of Computer Science, University of Bristol, Bristol BS81UB, UK

ABSTRACT

Though being a vital part of video indexing and retrieval systems, key-frame extraction algorithms have been based mainly on the analysis of various frame similarities and their later clustering. This work broadens the spectra of the analysis by focusing on the spatio-temporal region relations present in the scene to determine the most representative frame in the shot. It applies efficient video segmentation to a low-resolution sequence representative utilising a novel iterative unsupervised segmentation algorithm based on the anisotropic diffusion paradigm and k-means clustering. Key-frames are determined by applying a set of heuristic rules to the behaviour and features of extracted spatio-temporal regions. Experimental results are given.

1. INTRODUCTION

In order to facilitate a meaningful use of all-pervasive digital video media, the field of content based video indexing and retrieval faces a problem of bridging the semantic gap between user's need for intuitive retrieval and limited capability of the available low-level media descriptions.

In the media processing chain of a video indexing and retrieval system, key-frame extraction is a vital task. By determining the most representative frame of the analysed shot, key-frame extraction interprets the gist of that shot on every level of signification, from the perceptual low-level description to the high-level semantics. Its relevance is even more evident if we know that the majority of metadata describing the shot is being extracted from the chosen key-frame.

Nevertheless, the definition of the key-frame is yet vague. Whether it is the best representative regarding the perceptual similarity of the abstract metadata representation, evocative semantic summary or the subjective impression to the professional user, the importance of the spatial relations present in the shot is substantial. Not only because it can take into account events and relationships important for the subjective criterion of the frame significance, but because this method follows the new computational media aesthetics paradigm that has set the editing rules as fundamental in the video indexing and retrieval process. Therefore, a person entering the scene, a close-up to some object or tiger jumping on his prey are events that distinguish semantically important frames from unimportant ones.

Various techniques tackled this problem. By analysing the frame difference metric, researchers proposed a range of initial methods [1]. More recently, a clustering based technique classified frames by their global similarities and decided upon the optimal key-frames by choosing the ones closest to the cluster centeriods [2]. Some authors analyse the content behaviour by extracting motion or shot activity measures [3]. However, none of the presented algorithms endeavour to analyse the object spatial relations due to the high complexity of the algorithms involved.

This paper presents work that tackles the spatial relationship criterion in the key-frame extraction task by segmenting the low-resolution frame representation into regions and analyses the region behaviour using a set of heuristic rules, as described in Section 4 of the paper. Complexity reduction is achieved by analysing compressed domain features, as presented in the Section 2. Experimental results and conclusions are given in Sections 5 and 6 respectively.

2. COMPLEXITY REDUCTION

The starting assumption of the key-frame extraction module is that the shot boundaries are known. There are various approaches to efficient shot detection [1,4] that extract a one dimensional frame difference metric and by thresholding it, detect positions of the shot boundaries.

The analysed features are extracted directly from the compressed domain video stream. The motive to analyse the video in the compressed domain is in its low computational requirements. Since the segmentation algorithm needed for spatial analysis requires heavy computation the compressed domain processing will try reduce the complexity of data passed on to the segmentation module. The compression formats used are MPEG and H.23x, because of their DCT based compression.

The first step is to extract a low-resolution representative of a frame, known as DC sequence. DC coefficients, as the main constituents of the DC sequence, are readily

accessible for I frames, but since P and B frames are represented by the residual error after prediction or interpolation, their coefficients need to be estimated. To calculate the coefficients of an Macro Block (MB) in a P frame or B frame, the DCT coefficients of the 16x16 area of the reference frame from which the current MB was predicted need to be calculated. Let us call this area the reference MB (though it is not an actual MB). Since the DCT is a linear transform, the DCT coefficients of the reference MB in the reference frame can be calculated from the DCT coefficients of the four MBs that overlap this reference MB.

Figure 1 shows a MB in a P frame, MB_{Cur}, being predicted from a 16x16 area denoted by MB_{Ref}. During encoding, only the residual error of MB_{Cur}, with respect to MB_{Ref} is stored. The DC coefficients of MB_{Ref} can be calculated from the DCT coefficients of four MBs.

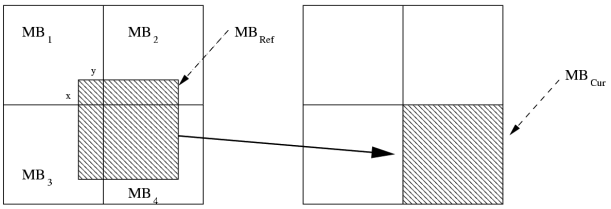

Figure 1. DC sequence generation

To avoid expensive computation, the DC coefficient alone is approximated by a weighted sum of the DC coefficients of the four MBs, with the weights being the fractions of the areas of these MBs that overlap the MB_{Ref}, i.e.,

$$DC(MB_{Ref}) = \sum_{i=1}^{4} \omega_i \cdot DC(i)$$

In addition to the complexity reduction by representing the frames as their low-resolution representatives, temporal complexity reduction is applied. Due to the fact that the events that occur in a continuous shot have to appear in multiple frames to be visible and taken into account in the key-frame extraction, only a subset of frames is selected as key-frame candidates. Frame candidate selection is done in two ways. One is by applying simple time decimation and taking only every n-th frame into consideration ($n \in [3,15]$). Another method is to calculate a cumulative frame-to-frame difference metric calculated during the shot detection process and select frames that accumulate constant measure of the frame difference between them. By applying this method, more frames are taken as key-frame candidates during the shot parts with a higher visual activity. The nature of the visual activity involved depends on the frame-to-frame metric; in our case this is block based colour histogram χ^2 distance. Due to its adaptability to different shot activities, this method reduces a lot of processing load during the static shots and speeds up the processing.

3. SEGMENTATION ALGORITHM

Having reduced the complexity of the data involved, an efficient unsupervised video segmentation algorithm is applied to the low-resolution shot representation. It segments 3-dimensional video signal into spatio-temporal regions by applying nonlinear filtering using anisotropic diffusion paradigm and later two stages of hierarchical k-means clustering. Various other segmentation methodologies were investigated, like nCut [5] and other eigen-decomposition based algorithms, but were abandoned due to their processing inefficiency.

3.1. Anisotropic Diffusion

The segmentation algorithm is based on the anisotropic diffusion approach [6]. This nonlinear filtering technique shows an extremely interesting property from the point of view of segmentation [7]: the smoothing is selective, being encouraged in homogeneous regions and inhibited across region boundaries. Thus, noise and irrelevant image details can be filtered out, making it easier for a segmentation algorithm to achieve spatial compactness while retaining the edge information.

Colour components are treated separately in HSV space, because of its good perceptual representation. Considering the anisotropic diffusion equation:

$$I_t = \mathrm{div}\left(c(x,y,t)\nabla I\right) = c(x,y,t)\nabla I + \nabla c \cdot \nabla I$$

the applied anisotropic diffusion has conductivity function c(x,y,t) as follows:

$$c(x,y,t) = g\left(\left\|\nabla I(x,y,t)\right\|\right) = e^{-\frac{\nabla I(x,y,t)}{K^2}}$$

where K is the diffusion coefficient.

Diffusion algorithm is an iterative process, where in each iteration t, a new level of image simplification is achieved. For the numerical implementation of the diffusion process, a simple discretisation scheme described in [3] with $N_i = 80$ iterations and the diffusion coefficient K selected as 5% of the maximum value of $|\nabla I(x,y,t)|$. In most cases, the diffusion ends up with 2-5 visually homogeneous areas.

3.2. Clustering Stage

After applying the diffusion process, hierarchical k-means clustering is applied to segment the 3D video space into predefined number of regions. At present the number of final regions in defined empirically (approx.: 5-9 regions).

The k-means clustering procedure iteratively minimizes the sum of squared distances between all points in the cluster and the cluster centre. For detailed description of the k-means clustering algorithm refer to literature [8].
Two stages of clustering hierarchy are applied. In the first stage, pixels are clustered only by their colour similarity. The distance function is defined as the Euclidean distance between the colour components in a given colour space. The best results are achieved in HSV colour space, though the choice of the colour space is left to the user.
In the second clustering stage, the procedure is repeated on the calculated clusters from the first stage as inputs. Each input cluster is defined by its geometrical centre coordinates and its colour components calculated as the average values of the member colour components. Therefore the metric distance in the second clustering stage is weighted Euclidean distance between the colour components and geometrical centres of input clusters.
This procedure results in segmented 3D video space, where each region represents a rough approximation of the visual object present in the shot. However, these regions cannot represent meaningful semantic objects but will convey the spatial and temporal composition of the analysed shot.

4. KEY-FRAME EXTRACTION

The key-frame extraction task is done by applying a set of heuristic rules to the information extracted from a number of units present in system. As mentioned before, the main driving factor of the decision process is visual continuity. Therefore, if a region disappears from the scene, merges with another region, enlarges over the whole screen, etc. this is considered as an important event. The block diagram of the key-frame extractor is given in Figure 2.
The initial assumptions are that the *Shot Detection Module* and the *Shot Type Classifier* are modules external to the key-frame extractor. The *Shot Detection Module* labels the shot boundaries and outputs frame-to-frame difference metric utilised in the temporal complexity reduction algorithm, as described in Section 2. *Shot Type Classifier* assigns a class to each shot from a predefined set of shot classes. Though this module is still being developed, a ground truth from a manually labelled metadata is used to classify shots into 10 categories, as given in Table 1.

BCU = BIG CLOSE UP	**LS** = LONG SHOT
CU = CLOSE UP	**CA** = CUT AWAY
MCU = MEDIUM CLOSE UP	**WS** = WIDE SHOT
MS = MID SHOT	**ZOOM** = CAMERA ZOOM
MLS = MEDIUM LONG SHOT	**PAN** = CAMERA PAN

Table 1. Shot Type Categories

Region Features and *Region Relations* are determined during the video segmentation procedure. Each region is described with a set of features: colour, geometric centre and area. Furthermore, a set of region relations is assigned to each region pair: adjacent, enclosed by, merging with, out of bounds, etc. all of which are assigned by analysing regions' relative positions, sizes and neighbouring relations.

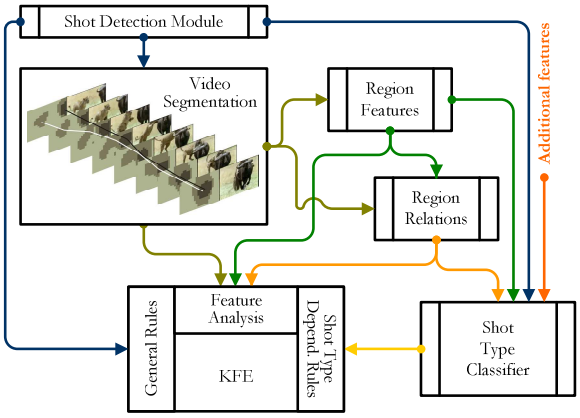

Figure 2. System block diagram

KFE module is taking into account region features and relations, and depending on a type of shot, applies both *General* and *Shot Type Dependant Rules* to input information and decides upon the best frame. In fact, the module calculates a linear combination $\Omega(i)$ of values assigned to relations $\rho(i)$ and features $\varphi(i)$ for each region R_i, while the coefficients $\alpha(i,T)$ and $\beta(i,T)$ are determined by the rules for a given type T, as in:

$$\Omega(i) = \sum_{\forall R_i} \alpha(i,T) \cdot \varphi(i) + \beta(i,T) \cdot \rho(i)$$

The frame having maximum Ω value is chosen to be a key-frame.
At present, coefficient tables are determined empirically, though there is a possibility to train the algorithm on a manually extracted set of key-frames and optimize the coefficients. However, some of the rules are pretty obvious. For example, if the geometric centres of two regions are positioned relatively in the middle of the frame and regions are not out of bounds (i.e. fraction of the region touching frame boundary is insignificant), the frame has higher probability to become a key-frame, given these two general rules. In addition, if two regions merge and the shot is a medium close up, like in the example in Figure 4, the moment they merge is very likely to become a key-frame of the shot.

5. RESULTS

Experimental results were conducted on a large dataset as a part of Intelligent Content Based Retrieval project at University of Bristol, courtesy of Granada plc. Media vault consists of 2000h of MPEG4 QT wrapped video of wildlife footage with rich human annotated metadata. This unique annotated multimedia database opens many opportunities to tackle the problem of the semantic gap in content based video retrieval.

Due to the heavy complexity reduction and compressed domain feature analysis, this procedure runs in real time for CIFF (352x264) MPEG-4 compressed video on a Pentium 2.4MHz.

The example of DC sequence extraction, anisotropic diffusion and two stages of clustering are depicted in Figure 3. The image distortion in the DC downsampling step is strong, though the major features and the image layout are saved. Nevertheless, however distinctive an object is, if it's relatively small in comparison to the neighbouring object and the number of final regions is small, it will me merged with the major object, as it happened with the small tree in the example. Finally, the segmentation result keeps the configuration of the major image layout, and the same conclusion can be applied to segmented 3D regions from a video clip.

Figure 3. Original frame, its DC image, five stages of diffusion process and two stages of agglomerative clustering

Figure 4 shows the region behaviour through the shot. Trajectories of the two main objects are marked and the chosen key-frame is labelled. The particular set of rules/coefficients used for this shot is given in the previous Section.

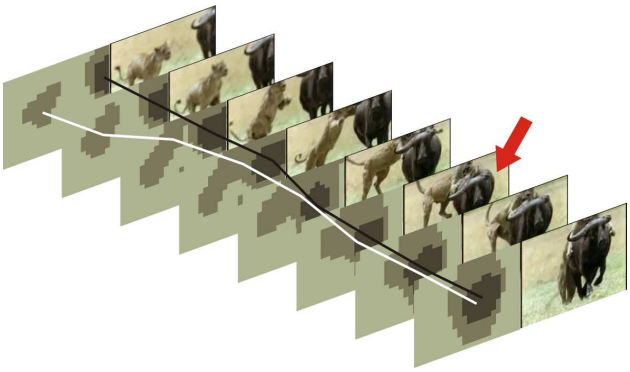

Figure 4. Original frames and the segmented regions with region trajectories throughout a wildlife footage shot

6. CONCLUSIONS

This paper presents an efficient algorithm for key-frame extraction that analyses behaviour of spatio-temporal regions present in a video sequence. Results show a good abstract representation of a shot while maintaining the real time processing capability. Future work will be directed towards development of the self-learning rule based decision process for key-frame extraction.

7. REFERENCES

[1] J. Calic and E. Izquierdo, "A Multiresolution Technique for Video Indexing and Retrieval", in Proc. ICIP 2002, Rochester, New York, USA

[2] Andreas Girgensohn, John S. Boreczky, "Time-Constrained Keyframe Selection Technique", Multimedia Tools Appl. 11(3): 347-358 (2000)

[3] Divakaran, A., Peker, K.A., Radharkishnan, R., Xiong, Z., Cabasson, R., "Video Summarization Using MPEG-7 Motion Activity and Audio Descriptors", Video Mining, ed. Rosenfeld, A., Doermann, D., DeMenthon, D. October 2003., Kluwer Academic Publishers, NY

[4] B. L. Yeo, B. Liu, "On the Extraction of DC Sequence from MPEG Compressed Video", Proceedings of IEEE ICIP, 1996.

[5] J. Shi, S. Belongie, T. Leung and J. Malik, "Image and Video Segmentation: The Normalized Cut Framework", ICIP 1998, Chicago, IL, pp. 943-7 vol.1.

[6] P. Perona, J. Malik, "Scale-space and Edge Detection Using Anisotropic Diffusion", IEEE PAMI, July 1990, Vol. 12, No. 7.

[7] L. Lucchese and S.K. Mitra, "Color Segmentation Based on Separate Anisotropic Diffusion of Chromatic and Achromatic Channels," IEE Proceedings Vision, Image, and Signal Processing, Vol. 148, No. 3, pp. 141-150, June 2001.

[8] A.K. Jain, M.N. Murty, P.J. Flynn, "Data clustering: a review", ACM Computing Surveys, Vol. 31, N.3, pp 264-323, 1999.