EE678 Wavelets Application Assignment, April 2005

Wavelet Based Video Segmentation and Indexing

Group Members:

Kashinath Murmu	cashee@ee.iitb.ac.in	01D07038
Vineet Kumar	vineetk@ee.iitb.ac.in	01D07001

Abstract

Video segmentation is a basic step in many video processing applications and automatic video indexing is an important feature in the design of a video database. Wavelet transform has emerged as a powerful tool for analysis and efficient compression of visual information. In this application presentation we review various wavelet based techniques for video segmentation and indexing reported in the recent literature.

Index Terms

Video database, Video indexing, Content based retrieval, Shot transition, Wavelet Transform

I INTRODUCTION

The adoption of video compression standards such as MPEG has accelerated the widespread use of video in many applications. These applications require effective techniques to manage the large quantity of video data. As video is a continuous time-based medium, it is important to first divide the video stream into temporal segments called shots. A shot is an atomic self-contained entity, which can be used as basic building blocks to support many high-level video access operations. The process of temporal partitioning of video into shots has been termed differently in the literature such as video segmentation, shot boundary detection, video parsing and scene change detection. In general, video shots are combined into a coherent sequence based on some video editing rules. There are two main types of video transitions based on the editing techniques used. If the two video shots are simply concatenated directly, the transition is termed as abrupt transition, which we denote as CUT. On the other hand, if the video shots are merged based on some special techniques to make the join appears visually smooth, the join will be a gradual transition and is denoted by GT. Because there are different types of GT, such as dissolve, fade in/out, wipe, morph, etc, and they may have varying temporal duration, most existing algorithms employ different techniques to handle these transitions. The goal of indexing is to provide the ability to store and retrieve images and video based on their contents. Potential applications of image and video-indexing techniques include multimedia information systems, digital libraries, remote sensing, and management of natural resources, movie industry, and video on demand.

Since the frames within a shot are similar, one of the frames is chosen as the key frame to represent the shot. The spatial characteristics (image indices) of the key frame, such as color, texture, and shape are then captured, which corresponds to the spatial content within a shot. The next step is to characterize the video dynamics such as object motion and camera operation, which essentially constitute the temporal content within each shot. The combination of the extracted spatial and temporal features (spatiotemporal index) provides a succinct representation of the visual content within a shot. The string of spatio-temporal indices of various shots within a sequence constitutes the video index.

A. Other approaches

A number of researchers have addressed video segmentation and querying. A survey of parsing and indexing technologies by Ahanger and Little discusses methods mainly in the pixel domain. Most of previous approaches to video indexing in the pixel domain have focused on color histograms. Color-histogram-based methods detect cuts by comparing the similarity between distributions of colors in the frames. Gargi et al. present an overview of how well histogram methods work in different color spaces using a variety of metrics. Color-histogram features have been combined with motion, texture, shape, and edge features to detect cuts, as well as represent the content of a shot.

II BACKGROUND THEORY

A. Multi-resolution Analysis

It is a self similar decomposition of $\mathcal{L}^2(\mathbb{R})$ into a ladder of subspaces. A function in any of these subspaces looks like a function in the other when expanded or contracted by a power of 2.

Axioms for MultiResolution analysis of square integrable ($\mathcal{L}^2(\mathbb{R})$) functions

The space of $\mathcal{L}^2(\mathbb{R})$ is a ladder of subspaces as shown below

$0 \subset \ldots\ldots\ldots\ldots\ldots V_{-2} \subset V_{-1} \subset V_0 \subset V_1 \subset V_2 \subset \ldots\ldots\ldots\ldots\ldots \mathcal{L}^2(\mathbb{R})$

(i) $\bigcup_{i \in Z} V_i = \mathcal{L}^2(\mathbb{R})$

(ii) $\bigcap_{i \in Z} V_i = \{0\}$

(iii) If $f(t) \in V_0$ then $f(t-n) \in V_0 \ \forall n$

(iv) If $f(t) \in V_0$ then $f(2^j t) \in V_j \ \forall j$

(v)There exists a function ϕ(t) $\in$ V_0 so that the translates {ϕ(t-n)} span V_0 orthonormally. i.e., the inner product <ϕ(t-n), ϕ(t-m)>=δ(n-m)

MRA Theorem

Under axioms (i) to (v) of a dyadic MRA, there exists a function $\psi(.)$, so that $\left\{\psi\{2^m t-n)\right\}_{n\in\mathbb{Z}}$ span $L^2(\mathbb{R})$

B. Continuous Wavelet Transform (CWT)

Continuous Wavelet transform of a square integrable functions x(t) is the dot product of x(t) with dilated and translated versions wavelet function $\psi(t)$.

The continuous wavelet transform is given by equation:

$$W_\psi f(b,a)=\frac{1}{\sqrt{a}}\int_{-\infty}^{+\infty} f(t)\psi(\overline{\frac{t-b}{a}})dt$$

The Parsval's like identity for Wavelet Transform is given as

$$\int_{-\infty}^{\infty}\int_{0}^{\infty} W_\psi f(b,a)\overline{W_\psi g(b,a)}\frac{da}{a^2}db=C_\psi\int_{-\infty}^{+\infty} f(t)g(t)dt$$

where,

$$C_\psi=\int_{-\infty}^{+\infty}\frac{\left|\psi(w)\right|^2}{w}dw<\infty$$

III APPLICATION.

A. Wavelet-based video indexing and querying

Authors have presented several algorithms suitable for analysis of broadcast video. First, they have shown how wavelet analysis of frames of video can be used to detect transitions between shots in a video stream, thereby dividing the stream into segments. Next they have described how each segment can be inserted into a video database using an indexing scheme that involves a wavelet-based signature. Finally, they have shown that during a subsequent broadcast of a similar or identical video clip, the segment can be found in the database by quickly searching for the relevant signature. The method is robust against noise and typical variations in the video stream, even global changes in brightness that can fool histogram-based techniques.

The effectiveness of above method derived from several desirable properties of wavelets. Wavelet decompositions allow for very good image approximation with just a few coefficients. This property has been successfully exploited for lossy image compression

Wavelet decompositions can be used to extract and en- code edge information. Edges provide important visual cues in differentiating images. The coefficients of wavelet decomposition provide information that is independent of the original image resolution. Thus, a wavelet-based scheme allows us to easily compare images of different resolutions. Wavelet decompositions are fast and easy to compute, requiring linear time in the size of the image and very little code.

To create a video database for querying, video is first divided into shots. In cinematography, shots are composed of all the frames filmed by one camera in a continuous sequence. For above purposes, shots are detected by searching for abrupt changes between consecutive video frames. In this section, a metric is developed for measuring these abrupt changes. Later, a second metric will be required for video querying, to measure the similarities between the querying shot and shots in the video library. The goal is to construct two metrics that are effective for video analysis, fast to compute, and require little storage space.

1) Creating the signature

First a method is established for forming a signature unique to a frame or sequence of video. For the above application, all comparisons are performed using only the luminance channel Y from the television standard color space, YIQ .It is found experimentally that using only the Y- channel is sufficient and improves the performance of the application. Rather than working with the 640X480 pixels available in a frame of NTSC video, each frame is subsampled to 1616, greatly accelerating the process that follows. The standard tensor product Haar wavelet transform is used . In the case of a single frame, the 16X 16 pixel array is transformed using 2D wavelets. In the case of a video segment, a 16X 16X16 block of pixels (arising from 16 consecutive frames, or roughly half a second of video) is transformed using 3D wavelets. Of the wavelet coefficients produced by transforming a frame or video segment, only the largest M are kept, truncating the rest to 0. The large coefficients correspond to the most significant features in the image (or video sequence). Truncation accelerates the algorithms and reduces the storage required for each video signature. It also improves the discriminatory power of the metric by ignoring the smaller coefficients, which contain most of the noise. The wavelet coefficients remaining after truncation are quantized, setting positive coefficients to +1 and negative coefficients to -1. Quantization accelerates the computation, reduces the storage cost, and improves the discriminatory power of the metric. The final, truncated, quantized decomposition is our wavelet-based signature for video frames or shots. The signatures for three frames of video are shown in Fig. 1.

2) Video indexing

The video database is composed of shots that are indexed by their wavelet signatures. The first challenge is to detect video transitions for shot segmentation. Next, the database is built by computing wavelet signatures and storing them in method which is easily and speedily queried. The shot transition detection algorithm is a straightforward implementation of the transition-detecting metric. The input is a sequence of video frames. For each frame, a signature is constructed as described in the previous section.

Then the difference is compared between consecutive frames. When the score is larger than a user-defined threshold, this is flagged as a shot transition.

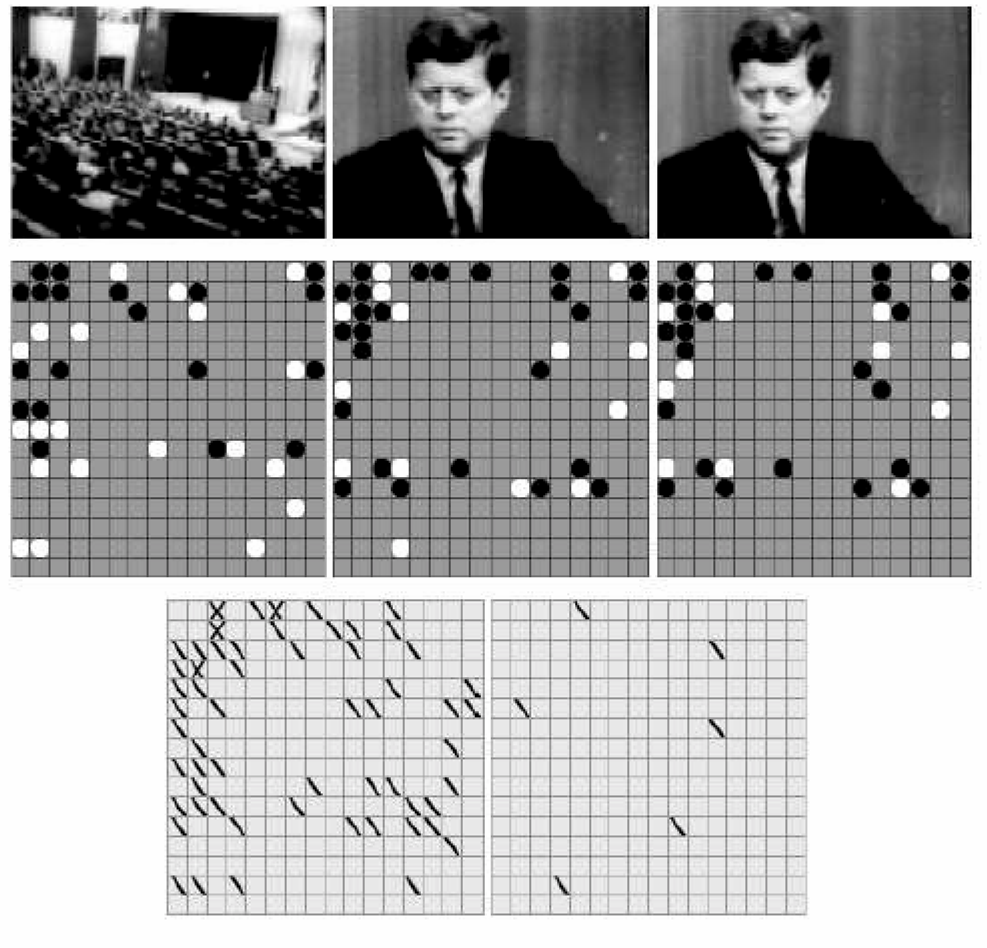

Fig.1.Finding The shot Transitions

3) Building a video database:

After isolating a shot, a wavelet signature is computed for it. The signatures are combined to form a video database. This algorithm does not require the actual video shots to be stored, so the decision whether to store the full video can be made based on the application than the original video. Following four methods have been identified for constructing a signature of a shot.

(a)Keyframe signature: By using the nth frame following a transition as a keyframe, we can use the keyframe's 2D wavelet decomposition, computed already by the transition detection algorithm, as a signature. This keyframe method is the easiest to compute, but is the most dependent on recognizing transitions accurately. If a transition is detected incorrectly by one frame, the keyframe may be different enough to have a completely different signature.

(b)Average signature: A slightly more robust process averages the 16 frames directly following a transition and utilizes the 2D wavelet de- composition of the average as a signature for the shot. Even if a transition is located incorrectly by a few frames, the signature of the average of all 16 frames will still be similar. However, by taking only an average of several frames, we do not consider temporal changes in the video. In addition, different shots with the same background may have similar averages, and therefore share similar signatures.

(c)Initial motion signatures: Rather than simply averaging the 16 frames following a cut, we can use the 16 frames as a 3D block of pixels and find a 3D signature using the 3D wavelets. These wavelets will capture any significant motion information at the beginning of the shot.

(d)Distributed motion signatures: Since the first 16 frames (roughly half a second) in a shot may not contain much motion information, we have also implemented a version of the 3D signatures that samples 16 frames evenly distributed over the shot. This method more accurately captures motion. However, it cannot be computed until the end of the shot and is more susceptible to failures in the cut detection mechanism.

(e) Storage of signatures: These signatures are stored in a database conducive to a rapid query algorithm. Associated with each possible location for a coefficient in a signature are two buckets (Y+ and Y-). (These buckets are entries in a 2D array if 2D signatures are used, and a 3D array if 3D signatures are used.) Each Y+ bucket contains a list of all shots that have a positive coefficient at that position; each entry of the Y- bucket contains a list to all the shots that have a negative coefficient at that position. Because all but the M most significant coefficients are truncated from each shot, each array entry will only list a fraction of the total number of shots in the database.

4) Video querying algorithm:

The video-querying algorithm is a straightforward embodiment of the video-query metric. As expected, video retrieval occurs one shot at a time. The computational complexity of the algorithm is linear with respect to the number of shots in the video database. To compute a similarity score between a query Q and every shot in the database, we first compute and store the difference between the query 3D signature's average intensity $Q[0,0,0]$ (or $Q[0,0]$ using 2D signatures) and those of the database signatures, $T[0,0,0]$ (or $T[0,0]$ in the case of 2D signatures). As a final step, the algorithm passes over the list of scores, finding the lowest score, which is considered to be the closest match for the querying signature.

IV. CONCLUSION

The wavelet-based method fared substantially better than the histogram method in terms of the numbers of both false positives and false negatives. Furthermore, while both methods have an asymptotic that is linear in the number of pixels per frame, it seems that the wavelets method has substantially lower constants. For cut detection and searching in broadcast video, it would be fruitful to analyze the sound stream in addition to the image stream to provide a better measure of scene cuts. Furthermore, it may be possible to consider closed-caption text as well (especially for database lookup) on systems where this information is available. Other metrics may work better for certain kinds of video, so a high-level analysis involving multiple measures over multiple media could yield a more effective tool for indexing and querying video. It will be possible to make the searching method much more robust by combining the results of several queries. If, for example, three out of four consecutive queries agree about a sequence within a scene, perhaps the disagreeing query may be ignored. Furthermore, information from the frame just *prior* to a cut could be used in addition to the frames just after the cut.

V. ACKNOWLEDGEMENT

We take this opportunity to express our sincere gratitude to our course instructor **Prof. V. M. Gadre** for his valuable guidance and constant encouragement rendered to us through out the course.

References

[1] Tai-Seng Chua, Mohan Kankanhalli and Yi Lin, *A General Framework For Video Segmentation Based On Temporal Multi-Resolution Analysis* Proceeding of Int'l Workshop on Advanced Technology,Japan.Jan2000.119-124

[2] Xiaodong Wen, Theodore D. Huffmire, Helen H. Hu, Adam Finkelstein, *Wavelet-based video indexing and querying* Multimedia Systems 7:350-358

[3] M.K. Mandal, S. Panchanathan, *Video Segmentation in the Wavelet Compressed Domain* Journal of Visual Communication and Image
Representation 12,17-28.2001

[4] Tat-Seng Chua, HuaMin Feng and Chandrashekhara A, *An Uni_ed Frameowrk For Shot Boundary Via Active Learning* ICASSP 2003.845-
848