

# A Novel Method to Search for Similar Video Clips within a TV Broadcast

Nael Hirzallah

**Abstract**—Marketing executives are among other groups who may be interested in matching a specific video commercial or other video clips within a certain broadcast channel. This paper presents an algorithm to recognize a certain video sequence within a live video stream. The algorithm features a way to create a small sized digital signature from a target video clip. Uploading a number of these signatures into the proposed system to automatically watch a TV channel and record or mark the times any of the video clips in question have been displayed would save time, cost and efforts that would have been invested in such a job if done manually. The proposed algorithm is based on selecting key frames, processing these frames, and performing a number of comparisons that involve some image parameters.

**Index Terms**—Camera Cuts, Commercials, Image Processing, Multimedia, Scene Detection, Video Sequence.

## I. INTRODUCTION

The idea of searching for a video clip or sequence that is already known within a video stream is not that common. However, demands for such a task have increased with groups of people such as news agencies and marketing executives.

A reliable and automatic recognition of video sequences has applications in a number of areas. One application is the spotting of a specific commercial within TV channel streams. Marketing executives, for example, may wish to know when and how often a particular commercial has been broadcast. One possibility would be to store the entire commercial and continually compare broadcast signals against the stored video sequences in the library through a frame by frame comparison. This solution is not very practical since it would require a prohibitive amount of memory and processing power to implement. Besides, making this manual would drive searchers crazy when they try to locate a piece of a video with the absence of proper indexing information.

Thus, instead of a frame by frame comparison, selecting specific frames in both the video sequence and the live video stream then comparing these corresponding frames would reduce the processing power, number of comparisons, and the size of the video sequence in question to be stored in the library. However, the question that may be raised at this point is how to select these specific frames or key-frames. Selecting key-frames from a video stream that may contain all sorts of images is not an easy task. Furthermore, having this process to satisfy the real-time requirements is another

story.

This paper proposes a system that works to spot a known video sequence, such as a specific commercial, within a live video stream. The system is split into two parts: one to generate a digital signature out of the video sequence in question and another to search for this sequence within a live video stream such as that coming from a TV channel. The system satisfies the real-time requirements and consumes low processing power to the extent that more than one TV channel may be monitored for one or more of such video sequences.

In the following section, an overview on video sequence recognition algorithms is presented along with a fair amount of detailed description to a cut detection algorithm that exists in the market, on which the proposed system is based. Next in section three, the proposed system will be explained. Following that, section four describes a typical experiment that is used to test the proposed system. Finally the paper concludes in section five.

## II. OVERVIEW

Storing an entire video sequence and continually comparing the broadcast signals against a library of such stored sequences using frame by frame comparisons is not very practical since it would require a prohibitive amount of memory and processing power to implement. Ref [1] teaches the use of digital signatures to match video sequences. However, such signatures that for example represent the times between black scenes or color changes are insufficient to permit fast and accurate matching of complex video sequences. Ref [2] describes a method of video recognition that involves creating digital signatures for all the frames in a sequence. This method involves an undue amount of processing power.

An object of the proposed system is to alleviate these problems, namely accuracy and time efficiency, and provide a practical method of recognizing a predetermined video sequence that is reasonably accurate yet at the same time requires an acceptable amount of processing power. The proposed system is a modified version of an earlier system that was issued in Ref. [3] yet the focus in this paper is on commercial types of video sequences. This required few modifications mainly in the search engine which will be highlighted in this paper in section 4. Yet, the core of the system is based on cut or scene detection algorithms to filter the video into key-frames. These key-frames which include the first frame in the scene after a cut, as well as other frames as it will be explained in section 3.

Selecting key-frames to transmit over a network, save on a hard disk, or use for browsing certainly reduces bandwidth,

capacity and time than considering the whole video data. Furthermore, detecting key-frames from a video sequence and letting these time-stamped key-frames to represent the sequence for the sake of spotting similar sequences within a video stream would reduce the number of comparisons to perform. However, cut detection is yet a difficult process when considering various types of camera breaks and operations. A typical simple camera cuts detection algorithm may result in detecting false cuts or missing true cuts. False cuts may result from certain camera operations, object movements or flashes within a video clip; while missed ones may result from gradual scene changes.

Video parameters to be considered by such algorithms may include intensity, red-green-blue (RGB), hue-value-chroma (HVC), and motion vector. A basic approach [4] to detect cuts is to compare the values of one or more of these parameters, such as luminance of the corresponding pixels in a pair of consecutive frames. In simple words, if the number of pixels whose luminance values have changed from one frame to the next exceeds a certain threshold, a cut is detected. Although, the solution is quite simple, but it does not usually result in high detection rates.

SmartFrame, was invented in [5] to achieve a higher video cut detection rate, satisfy real-time requirements, and use low resources. SmartFrame is currently being applied in the AccesTV application, a product of a Canadian company, named Televitesse.

Depending on SmartFrame or similar algorithms to filter a video sequence into scene key-frames, the proposed system takes the process further and processes these key-frames into a small size digital signature to represent the video sequence in question.

### III. THE PROPOSED SYSTEM

The proposed system is composed of two, parts as shown in Fig. 1.

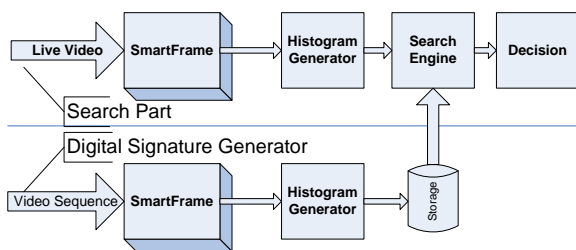


Fig. 1. Proposed system block diagram

The Digital Signature Generator part of the system is to process the video sequence in question. The first step is to select key-frames out of the whole sequence. These time-stamped key-frames will be categorized under one of the following classes:

- 1) Cut: Is a frame that is marked so by the cut detection algorithm used, such as SmartFrame; see column five in Table I.
- 2) Time-out: Is a frame which occurs after a predetermined period of time in the sequence when no "cut" has been detected
- 3) Initial Frame: the first frame in the video in question.

For each key-frame selected, the color table is initialized. Among the color parameters, luminances of values that range typically from 0 to 256 are selected to represent each pixel. A Luminance histogram will then be generated. The histogram will express the amount of pixels whose color lies within that value in the current frame in terms of percentages. Finally, the length of the video sequence is also recorded.

In summary, the digital signature of a video sequence will be composed of Histograms of a limited number of frames in the video sequence. These frames are preferably frames having a particular significance, namely "Cut", "Time-out", and "Initial-frame, along with their relative timestamps. Such information is small in size, yet condensed in information.

Fig. 2 illustrates how Digital Signature Generator works. The histogram of the first frame in the sequence is created. However, if this frame represents a complete black frame which usually exists between commercials and just before a commercial break in some TV channels, the system then waits for this break to end before capturing the first frame in the sequence. As mentioned in Table 1, gradual Fade-out cuts are also referred to as cuts, but of a gradual type. Thus, both gradual and sharp cuts are considered as simply cuts. The system then waits for the subsequent frame corresponding to a cut, and stores its histogram as the next in line unless a certain time, say two seconds, has elapsed without a cut occurring, in which case the histogram of the next frame is taken and tagged as a "time-out". This is to ensure that histograms are stored even when a sequence does not include cuts.

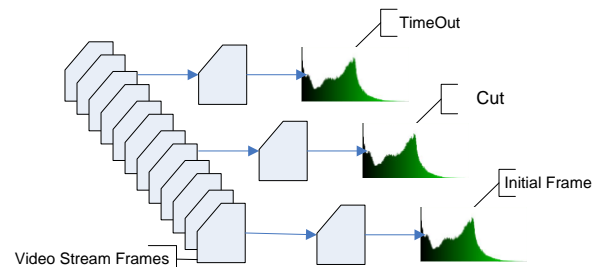


Fig. 2. Digital signature generator

The second part of the system is the search engine. Figure 3 depicts the flow chart of the search engine part of the system. It is important to note that when comparing two histograms of supposedly to be same frames, one from the live stream and another from the video sequence in question; it may result in some difference. These results are compared against pre set threshold values. Thus exact matches between histograms are not the target. This is because of many reasons that include the following:

- 1) A matching pair of histograms may belong to two frames that may not have the same exact timing due to a small shift in the sequence initial frame.
- 2) The broadcasting noise or lossy codec that may have been introduced to the broadcast video stream

In the proposed system, two values of thresholds are selected, low and high. The high threshold refers to a more tolerant match. It is used for comparing between histograms of initial frames, while lower value thresholds are used for comparing between histograms of cuts frames. This is

because the frames that belong to a cut is expected to be the same in both the Signature and live stream.

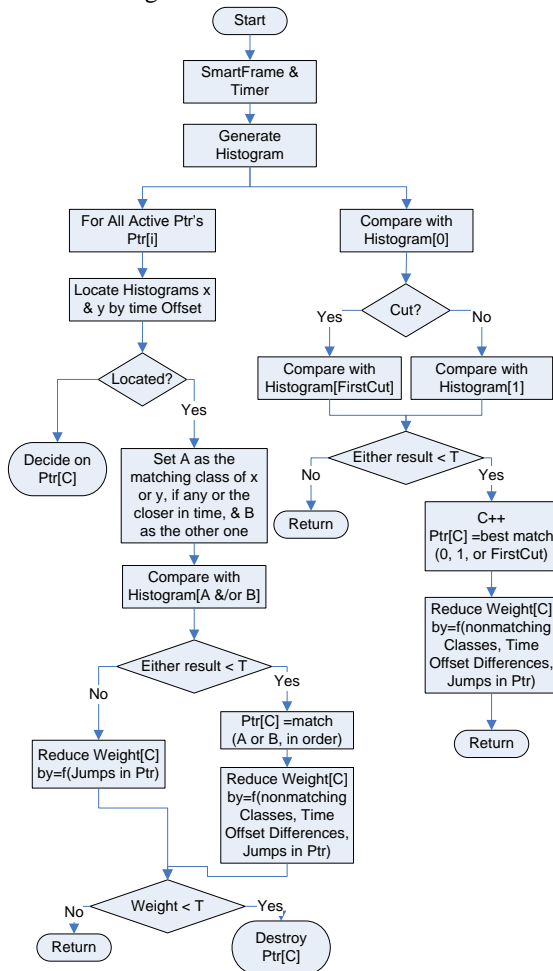


Fig. 3. Search engine flow chart

The timeout frames behave sometimes like cuts and sometimes like initial frames. If a timeout frame is preceded directly or indirectly by a cut, then it should be dealt with as a cut. This is because the cut synchronizes the sequence in question with the candidate sequence. While if the video sequence contains no cuts prior to a timeout frame, then the timeout frame should be dealt with as an initial frame and the higher value threshold will be used. This is because, as can be seen from the flow chart in Figure 3, a negative result test is more critical to making a quick mismatching decision than a positive result test. Positive test will be followed by more tests before making a decision. Moreover, when comparing histograms of mismatching classes the lower value threshold (less tolerant) is used. This is because these two histograms of mismatching classes should not be compared at the first place.

Looking at the right hand side of the flow chart, one may notice that each histogram generated from the video stream is compared against both the first, and either the second or the first cut histograms (in case the histogram in hand is a cut) of the signature. A pointer will be created at the better match, if any. This pointer will be considered active till it gets destroyed.

The histogram in hand is yet to be compared against all active pointers, as stated in the left hand side of the flow chart. In order to do that, an attempt to advance each pointer by the amount that is equal in time to the elapsed time since

the last advancement of the point in question. This advancement may result in one of the four following cases: lying between two signature histograms (which we call in the flow chart as x and y), lying before the next histogram, lying beyond the last histogram, or no more histograms in the signature file to compare with. In the first three cases, the histogram in hand is compared against the close by histogram or histograms. The pointer will then move to the position of the signature histogram if a match has occurred, giving a priority to the same class histogram, the closer one, then the better matching one. Upon advancing the pointer, the matching weight will be reduced based on three factors listed in order as follows: the number of histograms that have been skipped in the signature file, the difference in the classes of matching histograms, and the difference in the elapsed time. The weight is kept monitored, and the pointer gets destroyed if it exceeds a certain value.

In the last case, when no more histograms in the signature file to compare with exist, the active pointer in question will represent a matching complete signature.

It should also be noted that the original frame size and sample rate do not have to match the video frame size and sample rate of the candidate sequence, which may, for example, be a broadcast signal. Thus video signature matching will tolerate varying broadcast quality.

#### IV. TESTING

One of OfficeMax commercials (Fig. 4 shows a screen snapshot) was used as a typical commercial to generate a signature file for by the proposed system. It has a length of 30 seconds with 11 scene changes, which represent either sharp or gradual cuts. In this example, the computer considers only 17 frames, of which 11 represent cuts, one represents the first frame, and five represent timeouts, i.e. where no cut has been detected after a predetermined period of time. The signature for this sequence would contain 17 tables (histograms), their timestamps, and their classes.

The use of tagged histograms, according to whether they relate to frames that are Cuts, Initial frames, or Time-outs, considerably enhances the efficiency of the search routine.



Fig. 4. OfficeMax commercial snapshot

The search engine was tested on a long stored video stream that included a piece of a basketball game (characterized by an action-full video type), a long piece of a movie (characterized by little action), and a news program from CNN. The stream was interrupted by 25 commercials (few are repeated) spread over five commercial breaks. Three of these break the OfficeMax commercial in question existed. There was also a similar OfficeMax commercial but

not the one in question. Two of these three commercials were distorted by cutting 3 second from either the beginning or the trail of the commercial (the scenarios that marketing executives look for to request a rebroadcast). The proposed system was tested on this video and all three commercials were successfully spotted.

The same experiment was repeated but after splitting the stream into four parts. The system ran on these four parts simultaneously not on a Core 2 Duo but a Celeron PC. All three commercials were successfully detected while running a smooth video playback each of a 240x180 resolution, as seen in Fig. 5.

The proposed system permits video sequences to be matched with good accuracy in real time with the processing power available on a standard PC-based system and without having prohibitive memory requirements.

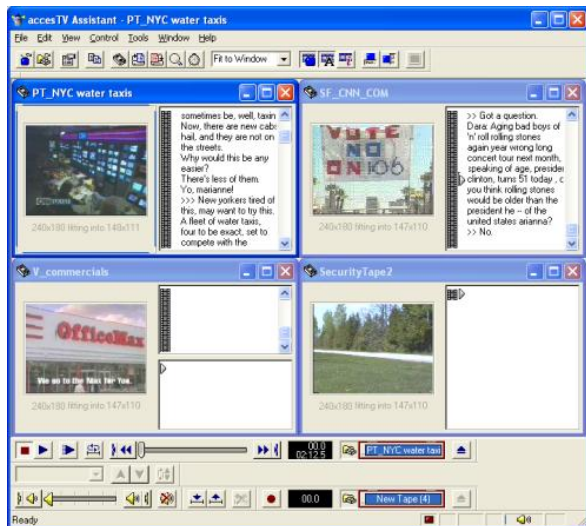


Fig. 5. Simultaneous search

## V. CONCLUSION

The paper proposed a system to spot a specific video sequence, such as a specific commercial, within a live video stream. The system composed of two parts; one to create a digital signature out of the video sequence in question, and another to search for this sequence within a live video stream. The proposed system can offer a high percentage hit rate. The stored video signature frame size and sample rate do not have to match the video frame size and sample rate of

the candidate sequence, which may, for example, be a broadcast signal. Furthermore, the video signature matching will tolerate varying broadcast quality.

The alert signal can take any form. It can be a visual or audible alert, or it can be used to take some action such as increment a counter. It will be understood, however, that the system described in this paper can load a number of signature files simultaneously, in which case the multiple incidences of the search algorithm are run on each signature file in parallel. Also it may run on a multiple video stream at the same time.

## REFERENCE

- [1] D. Michael *et al*, "Method and system for recognition of broadcast segments Ellis," U. S. Patent 5436653, 1995.
- [2] G. W. Schober, R. Chadwick, and A. R. Francis, "System and method for identifying a television program," U. S. Patent 5404160, 1995.
- [3] Nael Hirzalla, Paul Streach, Roger Maclean, and Robert Menard, "Method and apparatus for recognizing video sequences," U. S. Patent 6633651, 2003.
- [4] N. Hirzalla and A. Karmouch, "Detecting Scene Boundaries For Video Indexing," *Advanced Digital Library Forum'95*, May 15-18, 1995.
- [5] N. Hirzalla, Paul Streach, Roger Maclean, and Robert Menard, "Method Of Processing A Video Stream," U. S. Patent 6,415,000, 1997.
- [6] S. Raaijmakers, J. den Hartog, and J. Baan, "Multimodal topic segmentation and classification of news video," *ICME*, vol. 2, pp. 33-36, Aug 26-29, 2002.
- [7] N. Hirzallah and A. Karmouch, "Detecting Cuts By Understanding Camera Operations," *Journal Of Visual Languages And Computing*, vol 6, no 4, pp. 385-404.
- [8] N. Hirzallah and A. Karmouch, "Automatic Cut And Camera Operation Detection For Video," *International Conference On Consumer Electronics'95, Chicago, Illinois*, June 7-9, 1999.



**Nael Hirzallah** has finished his Masters and PhD Degrees from the University of Ottawa, Canada, in 1993 and 1997, respectively. He then worked in the industry in leading software and semiconductor companies in both Canada and USA. He joined Applied Science University in Jordan in 2002 to work as an Assistant Professor in the faculty of IT. In 2007 he was promoted to Associate. He became its Computer Center Director in 2003. For three years starting 2008, he worked as a department chairman and Acting Dean in Fahad Bin Sultan University in Tabuk, KSA. Since 2011, he returned to Applied Science University to assume the position of the Faculty of Information Technology Dean. His research interests are in eLearning, eCommerce, Multimedia and image processing.