TEMPORAL VIDEO REGISTRATION FOR WATERMARK DETECTION

Bertrand Chupeau, Lionel Oisel, Pierrick Jouet

Thomson R&D France
1, Avenue Belle-Fontaine – CS 17616
35576 Cesson-Sévigné – France

ABSTRACT

Video frame alignment with the original source signal is required in a variety of applications, for instance non-blind watermark detection. Robustness to temporal cropping, scene removal, frame-rate changes and other non-linear temporal distortions are required. This paper presents an algorithm for locating a video segment within a long video document with frame-accurate alignment. The video sequences are first summarized into one-dimensional "temporal profile" signals. Then frame-to-frame correspondence is obtained by a dynamic programming approach. An automatic assessment of the matching result is proposed. Extensive test results achieved 77% success rate.

1. INTRODUCTION

Throughout its life from origin to final display, a video content undergoes processing of various kinds. Large amount of photometric, geometric and temporal distortions can occur, in a way that the final document is very different, in terms of pixel values, compared to the original, while retaining an acceptable quality for the viewer. A typical example of such a processing chain is the illegal camcorder capture of a movie projected in a theater, followed by DivX compression and distribution on the Internet. In this case the (non-exhaustive) processing list includes frame rate and sampling grid conversions, cropping, deflickering, compression, etc.

If an application makes use of the original video signal or some properties derived from it as a reference to process the distorted signal at the other end of the image chain, some kind of re-synchronization may be necessary. It could for example be a non-blind watermark detector, subtracting the original unmarked signal prior to detection [4]. In this case, inversion of both geometric and temporal distortions are needed. But it can also more simply be any decoding process requiring frame synchronization with the original time reference.

This paper focuses on compensating distortions due to temporal structure modification, which includes frame rate modification, scene removal, temporal cropping, etc. More precisely we tackle the specific problem of locating a given reference video segment within a longer document and performing frame-to-frame alignment.

In section 2, we present the state-of-the-art in video alignment. The proposed algorithm is detailed in section 3. An extensive set of results is discussed in section 4.

2. STATE OF THE ART

Video alignment is usually needed when the same dynamic scene is recorded by different uncalibrated stationary cameras, generating two or more unsynchronized video sequences. In this context, a 1-D affine transformation can accurately model time-shift (offset) and differences of frame rates between the video cameras [2,7]. This assumption is not valid, however, in our case, as temporal cropping and other non linear transforms cannot accurately be described with a single parameter model over the whole sequence.

Delannay [5] proposes a temporal alignment of video sequences for watermarking systems, by matching a restricted number of frames among the whole sequences. The first step extracts key-frames, separately on the two (original and copy) videos. The second step locally matches the two sequences of images using a sliding window (in a given window, frame rates are assumed constant). In addition to providing sparse alignment only, this approach fails when too many key-frames were suppressed, or with high amount of motion activity (in this case the key-frame selection process is not likely to give the same results for reference and copy videos).

To obtain a full video alignment, we propose a method based on dynamic programming. Dynamic programming forms the core of many sequence analysis methods, including classical methods for sequence-to-sequence comparison and alignment [8], as well as more recent methods such as linear hidden Markov models (HMM) [6]. It is also adopted in computer vision, for example to perform stereo matching [1].

3. ALGORITHM OVERVIEW

The goal is to determine the location of a "reference" video segment within a longer "target" video, and to perform a frame-to-frame registration. The reference and target frame rates may be different.

This temporal registration process involves two steps. First a "temporal profile" is extracted from the two videos. In a second step, the temporal profile of the reference segment is matched against the temporal profile of the target video. The output of this matching step is a list of frame-to-frame correspondences.

3.1 Video temporal profile generation

The aim is to transform the image sequence into a compact one-dimensional signal, easier to manipulate, while being representative of the video content, robust to format conversions, illumination variations, and other impairments of the video signal. This one-dimensional signal will be referred to as the "temporal profile" in the following.

The temporal profile is a sequence of distances between successive image features, rather than a sequence of intra-frame features, in order to achieve robustness against picture appearance changes between the original and the copy, due to illumination variations. To be also robust to geometric distortions we chose as image feature a global statistics on pixel values, in the form of a histogram.

We compute for each frame a 512-bin color histogram (in YCrCb color space for the sake of decoding simplicity). A detailed comparison of other possible color descriptors to represent the image content, in a query-by-example context, can be found in [3]. The use of color information provides useful additional information in some critical cases, compared with mere luminance histograms. Other image features capturing texture properties, such as color distribution weighted by gradient magnitude, histograms of gradient orientations or energy in wavelet sub-bands, were found to be less effective.

The temporal profile value *p(n)* at frame *n* is the distance between color histograms extracted from successive frames *n-1* and *n*. The distance measure we employ to compute the dissimilarity between statistical distributions is the Bhattacharyya distance. This distance, given in Equation (1) for two N-bin histograms *H(i)* and *K(i)*, captures the temporal variations between successive color histograms with significantly less disturbances than a simple L_1 distance.

$$d_{Bhat}(H,K) = -\log\left(\sum_{i=1}^{N}\sqrt{h_i k_i}\right) \qquad (1)$$

Examples of such temporal profiles are given in Figures 1 and 2, on a 750-frame long reference segment, and part of a target video, respectively (the full-length target video is 85764 frame long). For the sake of illustration the temporal axis is rescaled in Figure 2. The peaks correspond to shot boundaries, but other more complex temporal variations of the video signal are also captured in this signal. The goal is now to find a subpart of the target profile, depicted in Figure 2, which matches in some optimal sense the reference profile of Figure 1.

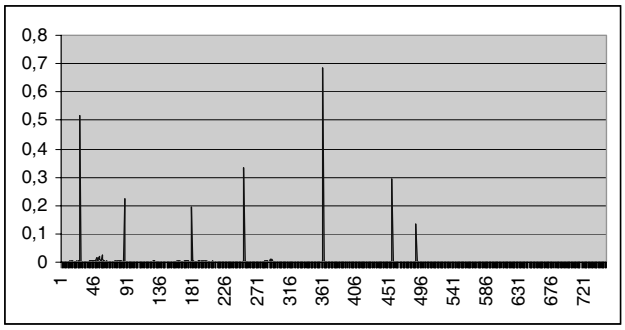

Figure 1: Reference segment profile

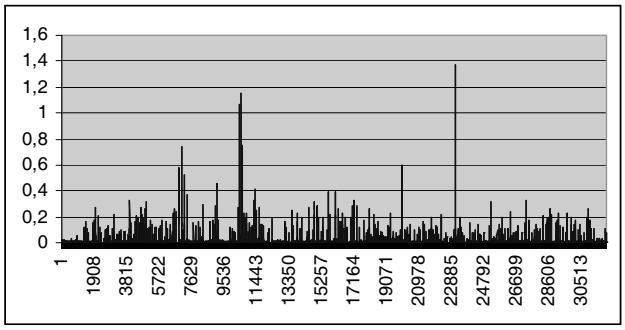

Figure 2: (Part of) a target video profile

3.2 Frame matching by dynamic programming

The frame-to-frame correspondence is found by a dynamic programming approach. The goal is to completely match two sequences of observations (each element of one sequence matches at least one element of the other sequence).

The dynamic programming method is based on the computation of a 2D array. An element *A(i,j)* of the array gives the minimal matching cost between two subsequences leading to matched elements *j* in sequence 1, and *i* in sequence 2 (in other words the best way to match sub-sequence *[0..j]* with *[0..i]*). This matching cost is computed as the sum of:

- the distance between feature vectors (temporal profile values in our case) associated with elements *i* and *j*,
- the cost of optimal path through *A* leading to *(i,j)* cell.

This minimal matching cost can be expressed recursively as:

$$A_{i,j} = Min\left(A_{i-1,j-1}, \omega_h A_{i,j-1}, \omega_v A_{i-1,j}\right) + dist(i,j) \qquad (2)$$

where ω_h and ω_v are weighting penalties associated with horizontal and vertical transitions, corresponding to several

elements of one sequence being associated with only one element of the second.

In our case, the feature vectors associated with elements *i* and *j* are one-dimensional (floating point) temporal profile values, $p_1(i)$ and $p_2(j)$, as explained in previous subsection. Our experiments lead us to conclude that a χ^2 distance performs better with such signals with high dynamics, than a simple absolute value of difference:

$$dist(i,j) = \frac{(p_1(i) - p_2(j))^2}{(p_1(i) + p_2(j))} \quad (3)$$

Finding the optimal correspondence involves first computing the whole matrix (filled from the top left to the bottom right, using equation (2)) and then determining the path giving the minimal final matching cost. A scheme presenting the mechanism of the dynamic programming is illustrated in Figure 3, with an example of optimal path.

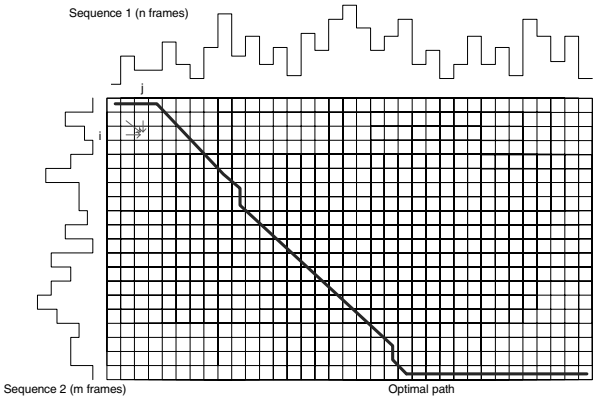

Figure 3: Frame matching by dynamic programming

3.3 Shot-based matching adjustment

In most cases, the dynamic programming approach provides accurate frame alignment. But in cases of low temporal evolution of image content (e.g., static scenes), the distance between successive color histograms is close to zero in-between two shot boundaries. In such cases, the dynamic programming approach favors the diagonal path (i.e. the default linearity hypothesis). Matching shot boundaries is then a way to recover accurate frame-to-frame correspondence. For this purpose, shot boundaries are detected in the reference and matched segments. This is easily performed by thresholding their temporal profiles. An example is depicted in Figure 4, with the reference segment at the top and the matched segment in the lower part: there are 9 scene cuts in the reference segment, which all find a correspondence in the matched segment that also contains false detections due to signal noise. Then, for each shot boundary of the reference segment, a corresponding shot boundary, if any, is searched for in a small temporal neighborhood in the other sequence. In-between each two successive matched shot-boundaries, a linear transform $T_2=\alpha T_1+\beta$ is applied to align the frame-to-frame correspondences with the shot-boundary correspondences.

Figure 4: Two shot-boundary sequences

In most cases the (α, β) parameters slightly vary with every shot along the sequence, and an accurate frame correspondence is obtained by applying the above-described shot-based linear transforms. However, in presence of a temporal crop, larger variations are detected, and it is preferable to rely on the result of dynamic programming.

3.4 Matching confidence

We observe that the percentage of matched shot boundaries in the previous adjustment step is a very good confidence measure of the success of temporal registration. In case of erroneous matches, the ratio of matched shot boundaries drops below 50% or even less. Otherwise it is close to 90% or even more (see Figure 5 in next results section).

We use this ratio as an indicator to drive a fall-back registration process in case of an erroneous result produced by the direct matching process.

3.5 Fall-back registration process

If an incorrect match is detected, a more computationally consuming process is activated. The target video is split into smaller overlapping segments and the earlier described registration process is performed on those segments. This gives as many candidate solutions as target segments, amongst which the one maximizing the frame-to-frame correlation between temporal profiles is chosen. This fallback process proves useful in recovering the correct match in some critical cases (see the following section discussing the results).

4. RESULTS

This algorithm was tested on a large set of real data. The reference data set consists of 15 feature movies (frame rate is 24Hz). The target data set consists of camcorded copies of those original movies: there are 154 such copies in all (2/3 of which at 29.97Hz, 1/3 at 25Hz). The average length of target videos ranges from 30 mn to 90 mn, as some copies are split into several files. The quality ranges from clean copies to highly distorted ones, with a large amount of cropping, trapezoidal distortion and compression artifacts. Image dimensions vary from 352×240 to 720×576 pixels.

For the sake of demonstration, we suppose that a given portion of the reference videos carry marks (e.g., forensic

tracking watermarks) and we wish to recover those marks in the camcorded copies. Temporal registration of such marked sections is therefore a prerequisite for mark decoding. For testing the registration algorithm we set the duration of the reference sections to 30 seconds, as a good compromise between too long and too short. Once the temporal profiles are computed offline and stored, only a few CPU seconds are necessary to perform the frame alignment of such a 30 second segment with a 1 hour long video.

A successful accurate frame-to-frame correspondence was achieved for 118 of 154 distorted copies (77%). Failures occur in case of a reference section located in too dark parts of the movie, in the presence of too many compression artifacts, high instability along the time-axis, frame-to-frame feature difference capturing compression noise rather than temporal content evolution.

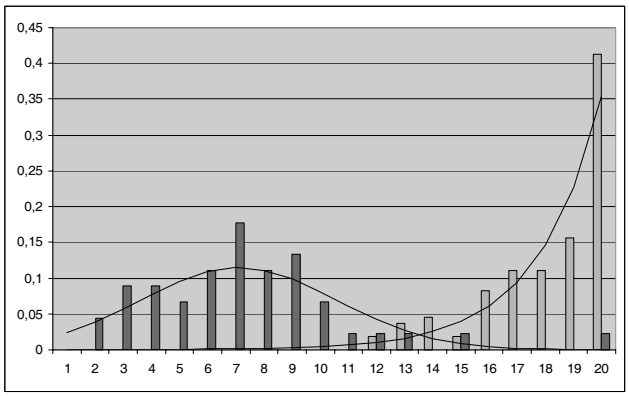

Figure 5: Registration confidence distributions

Furthermore, the ability of the confidence measure to assess registration accuracy was experimented with. In Figure 5, the normalized histograms of such confidence measures are depicted for both correct (right distribution – light grey bars) and erroneous matches (left distribution – dark grey bars). Each histogram bin represents a 5% wide interval within the [0%,100%] confidence range. The two distributions are relatively well separated: we modeled the left probability density with a Gaussian law ($\mu = 33.133, \sigma = 17.382$), and the right one with an exponential law ($a = 0.011$), both represented with continuous curves in the figure. This allows invoking a Neyman-Pearson test for deciding whether or not to activate the fall-back registration process. Setting a threshold at twice the standard deviation of the Gaussian distribution, that is a 68% confidence value, reduces the probability of type I errors (i.e. deciding the registration is correct when it is not) to 2.2%.

In 9 cases, a correct match was obtained when performing the fall-back registration process described in section 3, after detecting a first-pass registration error with a confidence measure below the threshold. The number of false matches was thus decreased by 20%, from 45 to 36.

5. CONCLUSION

An algorithm for locating a reference video segment within a longer video document and achieving frame-accurate alignment is described. It is based on the computation of compact and robust one-dimensional temporal video profiles: after a thorough comparison of available image features the choice was made to use the distance between color histograms of successive frames. A dynamic programming method was modified to obtain frame-to-frame alignment. Good performance was measured on a significant set of very noisy real data with 77% success rate on 154 different registration experiments.

One limitation is the computational demand of dynamic programming: if the length of the reference segment is increased too much, required memory as well as processing time become prohibitively large. For that reason a hierarchical approach could be resorted to when aligning two long video documents: a raw sequence alignment being achieved at the shot level in the first step, followed by a frame-accurate alignment between matched shots. Also, performing in parallel the same registration process on video and audio signals would increase the robustness.

6. REFERENCES

[1] H.H. Baker, and T.O. Binford, "Depth from edge- and intensity- based stereo", *Proc. 7th Int. Joint Conference on Artificial Intelligence*, pp. 631-636, August 1981.

[2] Y. Caspi, and M. Irani, "A step towards sequence-to-sequence alignment", *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, June 2000.

[3] B. Chupeau, and R. Forest, "Evaluation of the effectiveness of color attributes for video indexing", *Journal of Electronic Imaging*, Vol. 10, pp. 883-894, October 2001.

[4] I. Cox, M. Miller, J. Bloom, "Digital watermarking", Morgan Kaufmann, 2001.

[5] D. Delannay, C. de Roover, and B. Macq, "Temporal alignment of video sequences for watermarking", *IS&T/SPIE's 15th Annual Symposium on Electronic Imaging*, Santa Clara, California, USA, Proc. Vol. 5020, pp. 481-492, January 2003.

[6] A. Krogh, M. Brown, I.S. Mian, K. Sjölander, D. Haussler, "Hidden Markov models in computational biology: Applications to protein modeling", *Journal of Molecular Biology,* Vol. 235, pp.1501-1531, 1994.

[7] S. Kuthirummal, C.V. Jawahar, and P.J. Narayanan, "Video frame alignment in multiple views", *Proc. Int. Conf. on Image Processing (ICIP)*, Rochester, USA, September 2002.

[8] T. Smith, and M. Walterman, "Identification of common molecular subsequences", *Journal of Molecular Biology*, 1981.