

Audio and Video Signatures for Synchronization

Regunathan Radhakrishnan, Kent Terry and Claus Bauer
Dolby Laboratories Inc
100 Potrero Ave, San Francisco, CA
Email: {regu.r,kbt,cb}@dolby.com

Abstract—We propose a framework based on signatures extracted from audio and video streams for automatically measuring and maintaining synchronization between the two streams. The audio signature is based on projections of a coarse representation of the spectrogram onto random vectors. The video signature is based on projections of a coarse representation of the difference image between two consecutive frames onto random vectors. The time alignment present at the signature generator between the two streams is recorded by combining audio and video signatures into a combined synchronization signature. At the detector after video and audio streams go through different processing operations, we extract the signatures again. The signatures extracted before and after processing from the audio and the video are compared independently using a Hamming distance based correlator to estimate the relative misalignment introduced due to processing in each of the streams. Then, the estimated relative misalignment between the audio and video streams is used to preserve the same alignment between the streams that was present before processing. Our experimental results show that we can achieve $> 93.0\%$ accuracy in synchronization.

I. INTRODUCTION

Audio and video streams though initially synchronized in the front-end of a broadcast system can often go out of synchronization after going through different signal processing operations independently, resulting in lip synchronization errors. It is impractical to measure and maintain synchronism between these two streams manually. Hence, there is a need for an automatic mechanism to achieve this.

Past work on synchronizing audio and video using features has relied on first extracting certain audio or video features and then synchronously attaching these to the other signal. This could be achieved by watermarking and hence limited by the robustness of the watermarking method to recover the synchronization data [10].

In this paper, we propose a framework based on audio-visual signatures for automatically measuring and maintaining the time alignment between the audio and video streams. In this framework, the alignment present at the signature generator between the audio and video streams is recorded by combining audio and video signatures into a synchronization signature. It extracts certain video signatures and audio signatures at a point when the two streams are assumed to be in time alignment. The audio signatures are based on the projections of a coarse spectrogram onto random vectors [9] and the video signatures are based on the projections of a coarse difference image onto random vectors [8]. The extracted signatures from the audio and the video are content-based and do not change drastically as the audio and video streams go

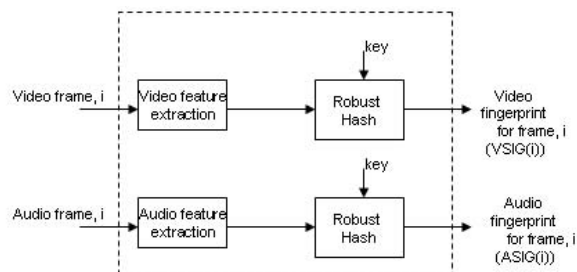


Fig. 1. Audio and Video Signature Generation

through signal processing operations that preserve the content. During synchronization, the same signatures are extracted and compared with the reference signatures to estimate the relative temporal misalignment between the two streams. Then, the estimated relative misalignment is used to achieve the same alignment between the audio and video streams that was present before processing. Note that the reference signatures for audio and video and their alignment information, do not need to follow the signal processing paths that the audio and video streams follow. They are assumed to arrive intact at the detector through a reliable meta-data path.

II. PROPOSED FRAMEWORK

Figure 1 shows the procedure for signature generation for audio and video. The extracted signatures are referred to as reference signatures. In practical applications, the audio and video streams are assumed to have the correct relative synchronization in terms of time alignment. Even if they are not aligned, the proposed system would measure and preserve whatever the alignment is between the audio and video streams at the signature generation end. The alignment information is recorded by combining audio and video signatures that occur together.

Figure 2 shows the procedure for synchronization at the detector. First, signatures are extracted from processed audio and video data. Then, the extracted signatures are compared with corresponding reference signatures using a Hamming distance based correlator. The output of hamming distance based correlator is estimated delay are relative to each of the corresponding reference signature. Finally, the estimated delays are used to correct the relative misalignment between the audio and video streams.

In the following subsections, we describe each of the components of the proposed framework.

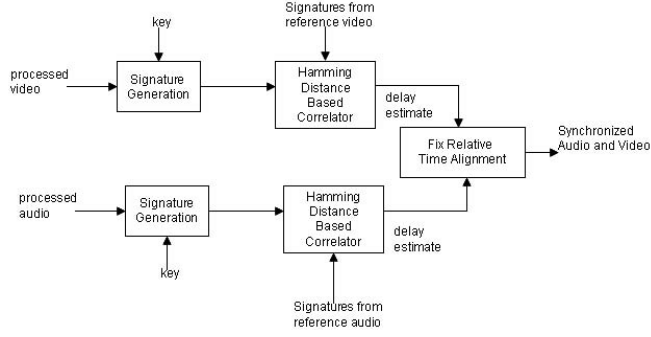


Fig. 2. Synchronization procedure at the detector

A. Audio Signature Extraction

The goal of audio feature extraction is to create a signature that is robust against a wide set of processing applied to the audio stream such as compression, time scale modification etc. The proposed method consists of two steps: Feature extraction is explained here and calculation of the signature bits is described in section II-C.

We use our audio signature extraction method proposed in [9]. The method in [9] creates a low-dimensional representation of a coarse spectrogram of the input audio frame by projecting the spectrogram onto random vectors, which can be thought of as basis vectors. For the sake of completeness, we provide a brief description here. In general, we could use any of other audio signature extraction methods proposed in literature [2],[5],[3],[4].

The generation of the coarse spectrogram itself is illustrated in Figure 3. The input audio is first divided into chunks of duration T_{ch} with an overlap of T_o between adjacent chunks. For each chunk of audio data (X_{ch}) we compute a spectrogram with certain time resolution and frequency resolution. Then, we mark time-frequency blocks within the computed spectrogram for X_{ch} . Finally, we sum up the magnitude of the spectrum within each of the time-frequency blocks to obtain a coarse representation of the spectrogram. Let us represent the spectrogram by S . We obtain a coarse representation (Q_a) of S by averaging the magnitude of frequency coefficients in time-frequency blocks of size $W_f \times W_t$ such that $F \times W_f = 129$ and $T \times W_t = 142$. F is the number of blocks along frequency axis and T is the number of blocks along time axis and hence Q_a is of size $(F \times T)$. Q_a is computed as given below:

$$Q_a(k, l) = \frac{1}{W_f * W_t} \sum_{i=(k-1)W_f}^{kW_f} \sum_{j=(l-1)W_t}^{lW_t} S(i, j)$$

$$f = 1, 2, \dots, F; t = 1, 2, \dots, T$$

Here, i and j represent the indices of frequency and time in the spectrogram and k and l represent the indices of the time-frequency blocks in which the averaging operation is performed. Note that the size of T_o (typically 10 ms) relative to inter frame duration (typically 30ms) in video determines the accuracy of alignment.

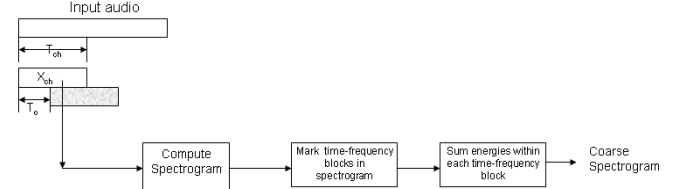


Fig. 3. Audio Feature Extraction

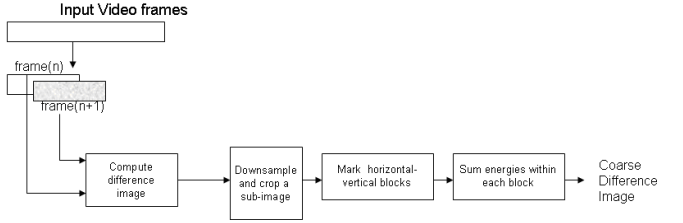


Fig. 4. Video Feature Extraction

B. Video Signature Extraction

The goal of the video feature extraction block is to come up with robust features that are invariant to certain signal processing operations on the video. This set of features would have to survive a variety of processing of the video, including compression, color space conversion, intensity adjustment, addition of computer generated graphic objects, format conversions etc. In this section, we use the video signature extraction method proposed in [8]. We provide a brief description below for the sake of completeness. In general, any of the other video signature methods proposed in literature could be used [6],[7].

Figure 4 illustrates the proposed video feature extraction approach in [8]. An absolute difference image is computed between adjacent video frames ($frame(n)$ and $frame(n+1)$). Then, the absolute difference image is downsampled and cropped so that the extracted features are invariant to interlaced and progressive video formats and addition of graphics and letterboxes on the corners of the individual frames. Then, the cropped absolute difference image is tiled horizontally and vertically. Finally, the intensities of the absolute difference image within each tile are summed up to obtain a coarse absolute difference image. Let us represent the cropped absolute difference image by Δ . We obtain a coarse representation (Q_v) of Δ by averaging pixel intensities in image blocks of size $W_x \times W_y$ such that $K \times W_x = 120$ and $L \times W_y = 160$. Q_v is of size $(K \times L)$

$$Q_v(k, l) = \frac{1}{W_x * W_y} \sum_{i=(k-1)W_x}^{kW_x} \sum_{j=(l-1)W_y}^{lW_y} \Delta(i, j)$$

$$k = 1, 2, \dots, K; l = 1, 2, \dots, L$$

Here i and j represent the indices for the horizontal and vertical dimensions for the absolute difference image Δ . k and l represent the indices of the sub-blocks of the absolute difference image Δ . This coarse representation (Q_v) helps us

achieve robustness by allowing for certain variations within a block while preserving the average intensity within a block.

C. Robust Hash

This block takes as input the matrix, Q ($Q = Q_v$ for video and $Q = Q_a$ for audio), and generates the signature by generating K hash bits. We use a robust hash function for this purpose as small perturbations in the audio and video features caused by signal processing operations such as compression, filtering etc would not change the hash bits drastically. The Robust hash function also serves to reduce the bit-rate of the signature stream. Let us represent the dimensions of the matrix Q by $(M \times N)$. By using a robust hash, instead of sending $(M \times N)$ values we only send few bits. A robust hash function is unlike a regular cryptographic hash function. A cryptographic hash function changes its output for every single bit change in the input. However, we would like our hash output to change slowly with small changes in features. This would enable us to allow for certain signal processing operations on the content which do not change the content but only slightly disturb the features. We use one such robust hash function proposed in [1] for generating the hash bits from the feature matrix, Q . We generate K random matrices each with the same dimensions as the matrix, Q ($M \times N$). The matrix entries are uniformly distributed random variables in $[0, 1]$. The state of the random number generator is set based on a key. Let us denote these random matrices by P_1, P_2, \dots, P_K each of dimension $(M \times N)$. We compute the mean of matrix P_i and subtract it from each matrix element in P_i (i goes from 1 to K). Then, the matrix Q is projected onto these K random vectors as shown below:

$$H_k = \sum_{i=1}^M \sum_{j=1}^N Q(i, j) * P_k(i, j)$$

Here H_k is the projection of the matrix Q onto the random vector P_k . Using the median of these projections ($H_k, k = 1, 2, \dots, K$) as a threshold, we generate K hash bits for the matrix Q . We generate a hash bit '1' for k^{th} hash bit if the projection H_k is greater than the threshold. Otherwise, we generate a hash bit of '0'.

D. Hamming Distance based Correlator

The goal of the Hamming distance based correlator is to compare a sequence of reference signatures (R) and a sequence of attacked signatures (A) extracted from processed audio or video and estimate the relative temporal alignment (delay) between the two. Let us represent the reference signatures for video as R_v and the reference signatures for audio as R_a . These signatures are extracted at a point where the audio and video are assumed to be temporally aligned. The audio and video may go through different signal processing operations and incur different delays δ_v and δ_a . Then, at the detector we extract the attacked signatures. Let us represent the attacked signatures for video as A_v and the attacked signatures for audio as A_a . The Hamming distance based correlator compares R_v and A_v and outputs a delay estimate γ_v of δ_v .

It also compares R_a and A_a and outputs a delay estimate γ_a of δ_a . Then, by using the delay estimates (γ_v and γ_a), we can achieve the same alignment between the audio and video streams that was present while the reference signatures were extracted.

Now let us look at how the Hamming distance based correlator computes the delay estimates. We will drop the suffices v and a from R_v, R_a, A_v and A_a as the Hamming distance based correlator is the same for comparing audio and video signatures. The inputs to the correlator are two set of signatures: R (reference) and A (attacked). The correlator has a search range of L signatures. For every signature from $R(i)$ from R , it tries to search within L a matching window (W) of signatures from A by computing the following score:

$$D(m, i) = \sum_{j=0}^{j=W} \text{Hammingdist}(R(i+j), A(m+j))$$

$$m = i - L, \dots, i + L; i = 1, 2, \dots, I$$

Here I is the total of number signatures in R and Hammingdist refers to the Hamming distance between two signatures. By computing this score, we are searching for a matching sequence of signatures $R(i)$ through $R(i+W)$ in the reference signature, a corresponding sequence of signatures $A(m)$ through $A(m+W)$ in A . $\gamma_i = \arg \min_m D(m, i); m = i - L, \dots, i + L$; is the index 'm' for which the score is minimum within the search window. This means that we found a matching signature at index 'm' in A for the reference signature at index 'i' in R . If there is no processing delay, then the matching index γ_i would be same as i . If there is a processing delay equal to 3 frames, then the matching index γ_i would be $i + 3$. Therefore, the relative delay estimate for the signature for frame 'i' is the offset value between 'i' and ' γ_i '.

III. EXPERIMENTAL RESULTS

A. Test Content

The test content used for the performance assessment of the proposed signature extraction based audio/video synchronization consisted of 36 one-minute audio video clips of a variety of content types in various formats. The video formats include standard broadcast formats such as Standard Definition 480/30i, High Definition 1080/30i and 720/60p. The audio for the content was both stereo and 5.1 channel formats at 48khz. Each of the test clips were processed in a variety of ways to simulate the processing in modern broadcasting and post production facilities. For video, the considered processing operations include: Fast/Slow play by 5%, Brightness modification by 10%, Median noise reduction, Addition of random film grain type noise, MPEG compression and decompression at 2,4,8 and 12Mbps, Down conversion from HD to SD, Logo and Graphic insertions. For audio, the considered processing operations include: 5% speedup/slow down without pitch correction, 5% speedup/slowdown with pitch correction, Down mixing, Gain modification by 15db, Equalization, Dynamic Range Compression, Dolby Digital

Video Source	Accuracy
Standard Definition	94.0%
High Definition (1080i)	97.0%
High Definition (720p)	93.0%

TABLE I
PERFORMANCE OF VIDEO SIGNATURE

Encoding/Decoding (320,480,640 kbps for 5.1 channel and 192 and 128kbps for stereo), MPEG1 layer II encoding and decoding at 128 and 256kbps for stereo. Overall, we created 432 test cases for video and 645 test cases for audio.

B. Parameter Settings

The signature extraction parameters for video were set as described below. The size of coarse difference image was set to be 8×9 ($K = 8, L = 9$). The number of random vectors for the hash was chosen to be 36. This means for every pair of frames we create a signature of length 36 bits. The parameters for comparing two video signatures at the hamming distance based correlator were set as described below. The search range was set to be 3 seconds and the window (W) was set to be one second worth of signatures.

The signature extraction parameters for audio were set as described below. The time-frequency resolution for the spectrogram was set to be 10×20 ($T = 10, F = 20$). The number of random vectors for the hash was chosen to be 18. The chunk size (T_{ch}) was chosen to be 90ms and the step size (T_o) was set to be 512 samples at 48kHz sampling rate. The parameters for comparing two audio signatures at the hamming distance based correlator were set as described below. The search range was set to be 2 seconds and the window (W) was set to be 0.25 second worth of signatures. These parameters were chosen empirically based on performance on a wide variety of content.

C. Algorithm Performance

Before we present the results, we first describe the criteria for success in Audio/Video synchronization. We consider it a success if the estimated relative alignment between the reference signature of source audio and the extracted signature of processed audio is within 10ms of the actual relative alignment between the two. For video, we consider it a success if relative alignment between the reference signature of source video and the extracted signature of processed video is within 1 frame duration of the actual relative alignment between the two. As long as we can accurately estimate these relative alignments for audio and video signatures in the hamming distance based correlator, we would be able to correct the temporal misalignment between audio and video. The synchronization accuracy is therefore measured in terms of percentage number of times for which the estimated relative alignment is within actual relative alignment.

Tables I and II present the accuracy of the alignment estimates for video and audio respectively. Note that we can achieve synchronism only when both the audio and video

Audio Source	Accuracy
Stereo	94.0%
Multi-Channel	97.0%

TABLE II
PERFORMANCE OF AUDIO SIGNATURE

alignment estimates are correct. In other scenarios, a higher level of logic is needed that combines them intelligently. Our initial results based on filtering out spurious alignment estimates provide slightly better results than those in Table I and II.

IV. CONCLUSION

We proposed an audio/video signature extraction based framework for measuring and maintaining synchronism between audio and video streams. The proposed system extracts reference signatures from audio and video at a point where the two streams are assumed to be aligned. Then, the audio and video streams take independent processing paths. The reference signatures along with the alignment information is available at detector where they are to be re-synchronized. At the detector, audio and video signatures are extracted from processed/modified content and compared with the reference signatures to extract the relative misalignment between audio and video. This information is used to preserve the same alignment between the two streams that was present when the reference signatures were extracted. We presented experimental results on a wide variety of content with different kinds of processing on the original source. For video, we were able to achieve 93%-97% accuracy in estimating the relative delay between the reference signature and the attacked signature. Similarly, for audio, we were able to achieve 94%-97% accuracy. In our future work, we would explore the tradeoff in accuracy and number of signature bits.

REFERENCES

- [1] J.Fridrich and M.Goljan, "Robust Hash Functions for Digital Watermarking", Proc. of ITCC, 2000.
- [2] Burges, C.J.C. Platt, J.C. Jana, S, "Distortion discriminant analysis for audio fingerprinting", IEEE Transactions on Speech and Audio Processing, May 2003.
- [3] Chun-Shien Lu, "Audio fingerprinting based on analyzing time-frequency localization of signals", IEEE Workshop on Multimedia Signal Processing, 2002.
- [4] E.Battle, J.Masip, E. Gaus, "Automatic Song Identification in Noisy Broadcast Audio", Proc. Of SIP, Aug 2002.
- [5] J.Haitsma and T.Kalker, "A Highly Robust Audio Fingerprinting System", In Proc. ISMIR 2002
- [6] Kozat, S.S. Venkatesan, R. Mihcak, M.K. , "Robust perceptual image hashing via matrix invariants", ICIP 2004.
- [7] A.Swaminathan, Y.Mao and Min Wu, "Image Hashing Resilient To Geometric and Filtering Operations", MMSP 2004.
- [8] R.Radhakrishnan and C.Bauer, "Content-based video signatures based on projections of difference images", MMSP, 2007.
- [9] R.Radhakrishnan, C.Bauer, C.Cheng and K.Terry, "Audio signature extraction based on projections of spectrograms", ICME, 2007.
- [10] US patent 6211919.