International Journal of Applied Engineering Research
ISSN 0973-4562 Volume 10, Number 6 (2015) pp. 15571-15580

http://www.ripublication.com

A Time Delay Based Efficient Approach For The Synchronization of The Audio and Video Streams

S.Vigneswaran*
Research Scholar, Anna University Regional Centre Coimbatore,
vickymca05@gmail.com.
Dr.A.Leelamani
Asst Prof of Mathematics, Anna University Regional Centre Coimbatore,
aleelamani@gmail.com.
K. Divya
Senior Research Fellow, Tamilnadu Agricultural University,Coimbatore ,
kdivyyaa@gmail.com.

Abstract

The most obvious result of audio to video mismatch are visible "lip synchronization" and Content Synchronization errors. This problem certainly can and does happen in today's scenario, with the frequency of occurrence at particular intervals so it’s becoming a significant concern to advertisers for station management. The mistiming of audio and video will always cause a subconscious degradation of the program's entertainment quality as perceived by the home viewer when the audio is advanced with respect to the video or vice versa. The cause of this effect is believed to be the unnatural relationship between the streams. In this natural environment we are used to hear the audio slightly delayed with respect to video due to the slower speed of propagation of sound waves as compared to light. In today's systems however, it is the video which is delayed thus causing the sound to arrive at the viewer's ears before the corresponding visual sensation. This paper presents an algorithm for minimizing the audio to video synchronization error correction mainly in Visual multimedia systems such as television systems. Some of the more common sources of errors are described; the problems which the errors create and solutions to the error causes are outlined.

Keywords: Multimedia , Synchronization ,Time delay ,Audio, Video.

Introduction

Recently, many researchers in the field of hand-held consumer electronics have focused on incorporating various kinds of multimedia and supplementary services into

portable devices. Reflecting this trend, many of the contemporary cellular phones are implemented with applications like fingerprint recognition, speech recognition, video-telephony (VT) [2], and video-on-demand (VOD). Among them, with the recent advent of Wideband Code-Division Multiple Access (WCDMA) services, VT system has begun to draw significant attention. In order to transmit payload in real-time for VT systems, Real-time Transport Protocol (RTP) is usually employed. In pair with RTP, we always use RTP Control Protocol (RTCP) for the purpose of quality control. In order to synchronize between different Medias, we inspect every RTCP Sender Report (SR) to find the reference time corresponding to RTP timestamp conveyed on RTP packet. We propose an efficient audio/video (A/V) synchronization method. In this method, we do not need to process RTCP SR packet for synchronization. Additionally it does not require any floating-point operations or any divisions at all. Moreover, the decision criterion for synchronization can be compactly described just in a single equation. Through extensive simulations, the proposed algorithm shows noticeable advantages in terms of required computation load and simplicity of software structure.

Viewer Perception Problems

Viewing programs in portable device or television programs with advanced audio is unnatural for the viewer, and is believed to cause many errors and stress. Tests at Stanford University (1) demonstrate that viewers who watch commercials programs are having advanced audio than the same commercials which were played with the audio in sync with the video. It was discovered that this effect takes place with relatively small audio advances where the mere existence of an audio problem was detected by very few average viewers. In addition to the negative perception of the commercials in the presence of advanced audio, there was also evidence this caused the test subject's memory of the negative aspects of the commercial to be remembered longer than normal. The least possible scenario takes place, the viewer perceives the advanced audio commercial in a bad light, and also remembers it longer than a commercial which is properly presented. Obviously, such problems can cause a great deal of concern for television advertisers.

CCD Camera Generated Vision Delays

Audio to video synchronization errors are becoming more troublesome as television technology progresses. The wide use of cameras having CCD sensors is having this synchronization problem. All CCD sensors have a basic visual delay mechanism. Depending on the sensor type, the visual delay may be several fields for newer camera types. In particular, the liberal use of digital frame store based image processing in newer cameras is creating previously unknown vision delays of various fields, with a four field delay not being rare.

Variable Temporal Resolution in the CCD

It would be worthwhile to mention the effect that variable shutter speeds has on temporally sampling the image. At high level, that is a 1 frame shutter speed, the

image is combined over the entire frame, tending to blur any motion in the image and making it difficult for page 2 the viewer to distinguish precisely such events as lip movement. This blurring was normal with tube based cameras which were continuously exposed to light.

With a fast shutter speed of CCDs, the image is integrated over a relatively short time, for example 100As for a 1/10,000 second exposure. In television systems, the frame rate (assuming a frame rate CCD exposure) is equivalent to the sampling rate in sampling theory. The exposure time is equivalent to aperture time. The ratio of exposure time to frame rate is the aperture ratio. It is known from sampling theory that the aperture ratio effect on frequency response, which in this case is the ability to accurately convey motion. For short exposures, the ability to convey motion to the viewer increases dramatically. The shorter exposure time gives brighter and less blurred moving edges which result in the viewer's improved ability to perceive motion. The CCD camera induced improved motion perception aggravates the corresponding increased image delay time, and makes any audio to image mismatch easier for the viewer to consciously or subconsciously detect.

Video Processing Delays

Video signals are often passed through special effects generators, color correctors, noise reducers, frame synchronizers and a variety of other editing and image processing functions. As memory costs continue to degrade, complexity is increased in these devices, and many incorporate frames based processing functions which add delays which are switched in and out. Unlike the past where video delays slowly drifted due to differing sync generator phases, the video delay in many of current interactive multimedia systems take instant jumps of one or more frames, as editors and other operators can choose from different processing modes. This situation is especially true of many current noise reduction and color correction products where extra frames of delay are added for each additional selected function. This instant change of delay length poses special challenges for the corresponding audio synchronizer which must keep up with these instant large changes in video delay.

Audio/Video Synchronization Algorithm

Conventional Synchronization Algorithm

RTP has supplementary information like sequence number and RTP timestamp in its header to facilitate real-time transmission [3]. In the VT system, synchronization between audio and video data is a crucial issue, since audio and video data are transmitted in separate RTP streams. However, we cannot directly use RTP timestamp to synchronize data conveyed in different RTP sessions for the following two reasons. Firstly, RTP timestamp begins at a random number [3]. Thus, the initial RTP timestamps for audio and video sessions are different, but they are actually sampled at the same time. Secondly, RTP timestamp increases in proportion to the sampling rate of media. Usually the sampling rates of audio and video data are quite different.

Thus, the rates of increase in RTP timestamp for audio and video sessions are not the same. To circumvent these two problems, RTCP SR packets carrying both the RTP and the Network Time Protocol (NTP) timestamp are generally employed. NTP timestamp provides absolute time information specified by RFC1305 [4]. Fig. 1 illustrates the streams of RTP and RTCP packets for a certain media. Without loss of generality, let us assume that this media stream constitutes an audio session. For a specific RTP packet shown(highlighted) in Figure 1,let us assume this RTP packet is located between i + 1 th and i+ 2 th RTCP SR packets in time order. If we let T^A be the reference time in second corresponding to the RTP timestamp, M^A , of this specific RTP packet, we obtain the following relation:

$$\frac{T^A - T^A_{Sr}(i+1)}{M^A - M^A_{Sr}(i+1)} = \frac{T^A_{Sr}(i+1) - T^A_{Sr}(i)}{M^A_{Sr}(i+1) - M^A_{Sr}(i)} \quad (1)$$

The superscript A in each term is added to show that they are related to audio session. In (1), T^A_{SR} (i) and T^A_{SR} (i+1) denote NTP timestamps conveyed by i + 1^{th} and i + 2^{th} RTCP SR packets, respectively. Even though NTP timestamp is a 64-bit integer [4], T^A , T^A_{SR} (i), and T^A_{SR} (i+1) in (1) are floating-point number in the unit of second. In the same manner, and are RTP timestamps carried with i + 1^{th} and i + 2^{th} RTCP SR packets, respectively. We can rearrange (1) to obtain the reference time, T^A as follows [7]:

$$T^A = T^A_{Sr}(i+1) + \frac{T^A_{Sr}(i+1) - T^A_{Sr}(i)}{M^A_{Sr}(i+1) - M^A_{Sr}(i)} (M^A - M^A_{Sr}(i+1)) \quad (2)$$

The same procedure can be applied to video session to obtain T^V , the reference time in second corresponding to RTP time stamp in video RTP packet. Comparing the obtained T^A and T^V values, we find out whether video data being decoded is relatively fast or slow compared to the audio data.

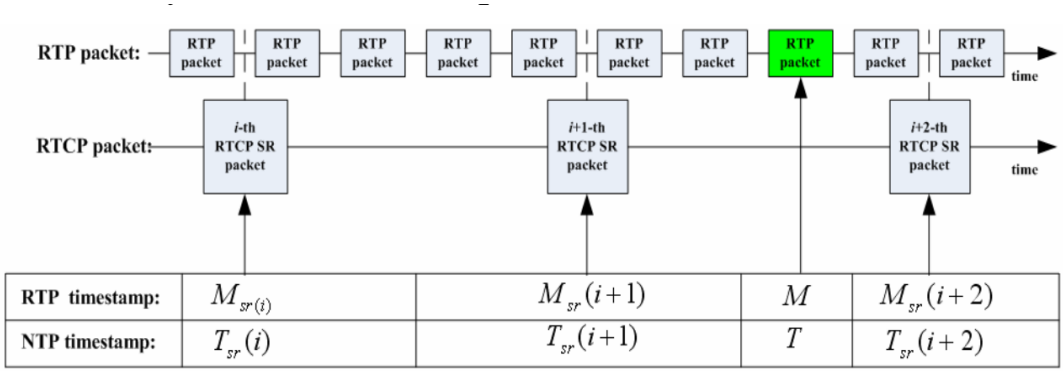

Figure 1: RTP and RTCP packets.

Proposed Synchronization Algorithm

If the sampling rate of audio data, R^A, is constant, we can simplify (1) into

$$T^A = T^A_{Sr}(0) + \frac{M^V - M^V_{Sr}(0)}{R^A} \quad (3)$$

by noting that

$$\mathbf{R^A = \frac{M^A_{Sr}\ (i+1) - M^A_{Sr}\ (i)}{T^A_{Sr}\ (i+1) - T^A_{Sr}\ (i)}} \quad \mathbf{(4)}$$

In the same manner, we can obtain reference time in second T^V , which corresponds to the RTP time stamp in a specific RTP packet in a video session by: M^V

$$\mathbf{T^V = T^V_{Sr}\ (0) + \frac{M^A - M^V_{Sr}\ (0)}{R^V}} \quad \mathbf{(5)}$$

By subtracting (5) from (3) and some arithmetic, we can obtain the following decision rule:

$$\mathbf{R^A\ M^V\ -\ R^V\ M^A\ <\ \eta\ ,\ if\ Audio\ is\ too\ fast} \quad \mathbf{(6)}$$

$$\mathbf{R^A\ M^V\ -\ R^V\ M^A\ >\ \eta\ ,\ if\ video\ is\ too\ fast} \quad \mathbf{(7)}$$

Where the threshold , η IS GIVEN BY

$$\mathbf{\eta\ =\ R^A\ R^V (T^V_{Sr}\ (0) - T^A_{Sr}\ (0) + R^V M^A_{Sr}\ (0)\ - R^A M^V_{Sr}\ (0)} \quad \mathbf{(8)}$$

In (6), R^V and R^A can be obtained during the Session Description Protocol (SDP) [8] negotiation process, and as previously mentioned, they are integer values representing the sampling rates of each media. M^A and M^V are RTP timestamp contained in each RTP packet, which is a 32 bit integer value as specified in [3]. Since all of the R^V , R^A , M^A , and M^V values in (6) are fixed point numbers themselves, there is no need to utilize floating point operations at all. Additionally, (6) does not require any division operations unlike the case of (1), (2), and (4). Obviously, this is a clear advantage for embedded processors, which usually do not have floating point units. For ARM processors, avoiding division is also advantageous aspect. To implement (6), all we need to do is two fixed-point multiplications, one subtraction, and one comparison operation.

System Implementation and Experimental Results

The proposed method is incorporated in a prototype VT system. Fig. 2 shows the hardware structure of the developed system. We adopt H.263 video codec for video processing, and Qualcomm Code Excited Linear Prediction (QCELP) for audio processing. Video codec operates on top of TI Open Multimedia Application Platform (OMAP) 1510 processor [6], which incorporates an ARM925T core and a TMS320C5510 Digital Signal Processor (DSP). We adopt Qualcomm Mobile Station Modem (MSM) 5500 as a baseband modem. In this system, OMAP processor acts as the host processor, while MSM 5500 processor operates at the slave mode. For OMAP processor, we use Nucleus PLUS as an RTOS for the ARM core.

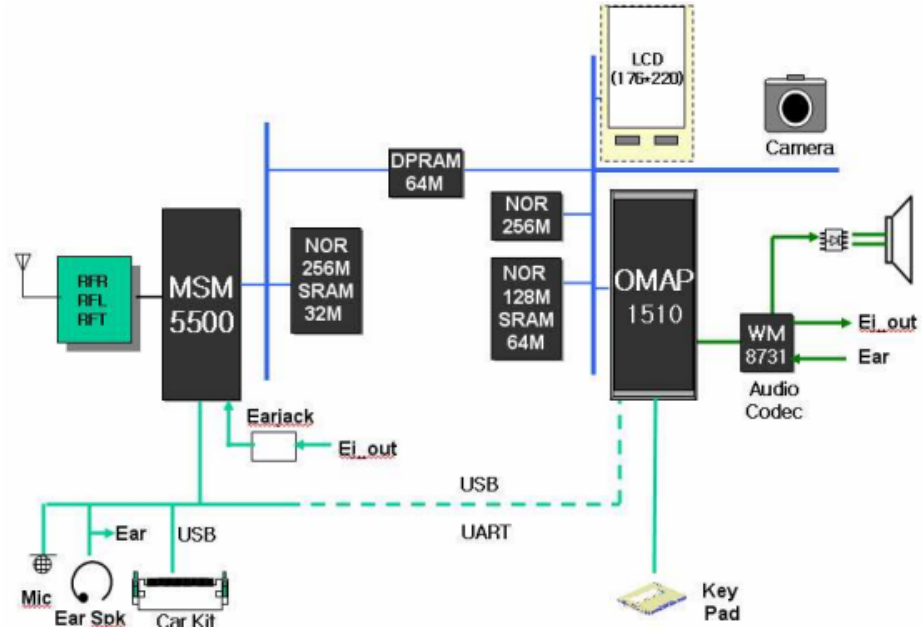

Figure 2: Hardware Block Diagram For The Developed VT System

Table I shows comparison results of required computational amount in clock cycles for the case of ARM925T processor. As shown in the table, the proposed system can reduce the computational amount required for synchronization by more than one-tenths. The simplified method in this table is comprised by (3) and (5). Finally, the result for the proposed method is obtained using (6) and (7). In the case of conventional and simplified methods, we obtain these results after scaling into fixed-point arithmetic.

Table 1: Comparison of Required Clock Cycles For Different Approaches

Required Clock Cycles	Conventional Method	Simplified Method	Proposed Method
Decision rule	145 cycles	72 cycles	8 cycles

Setting Performance Standards

Several standards committees have set standards or guidelines for audio to video synchronization errors. The Radio communication Study Groups of The International Telecommunication Union states "Given the operating practices employed in the United States and the requirement that a single picture and sound service may reach the consumer in different forms and via different paths, the list of preferred points should be as noted above and the tolerances required at each of the points should be the same (+1lfield, -2 fields) with the understanding that these tolerances are absolute, are not accumulative, and apply to the overall system".

The International Telecommunication Union in the Draft New Recommendation [DOC. 11/59] reports those errors of and greater than +20 and -40 ms are detectable and errors of +40 and -160 ms are "subjectively annoying" (+ numbers indicate sound advanced with respect to video). The draft recommendation states: A tighter tolerance on the range of values in the studio and production paths would be required to allow this (partitioning of tolerances]. The situation might look something like this:

+20 ms -40 ms Overall tolerance
+10 ms -30 ms Production/presentation
+10 ms -10 ms Distribution/transmission
+2 ms -2 ms Per codecpage 3

EIA/TIA-250-C standards call for a +25 to -40 ms specification end to end for transmission facilities. Given the inherent video delays in CCD cameras, very little additional delay can be tolerated in the rest of the system.

Measuring The Video Delay

Clearly, television facilities need to be designed with audio synchronization in mind. It is impractical to remove the offending video delays, so the only remaining solution is to ensure that the program audio receives the same delay as the associated video. Part of the solution is to measure the video delay at each significant delaying device so that a corresponding audio delay can be inserted at that point. Several video synchronizer manufacturers have a digital delay output (DDO) which provide a current video delay value signal for use by a companion audio synchronizer. Additionally, video delay detectors are available for devices which do not provide DDO signals. The audio synchronizer receives the DDO signal and automatically delays the audio signal by a corresponding amount.

Delay detectors for video devices without DDOs operate by storing a given input video frame and comparing all output frames to the stored frame. By counting the number of frames which pass until the previously input frame is output, the video delay is obtained. These devices are easy to add to an existing system, requiring only that input and output video be looped through their inputs. They provide a DDO signal which may be utilized by a companion audio synchronizer to make appropriate corrections.

The Second Generation Audio Synchronizer

It should be noted that all currently viable solutions to the audio to video synchronization problem utilize adjustable audio delays at some point in the system to delay the audio to match the delayed video. The adjustable audio delay remains a key element in system designs, and second generation synchronizers are challenged with the problem of making adjustments to the delay length which are imperceptible to the viewer.

As video delay values take jumps of one or more frames, the audio delay is required to take on the new, greatly different delay value without disrupting the audio. old style audio delays often operated by dropping or repeating audio samples, and relied on slowly changing video delays to operate properly. The occasional sample manipulation usually went unnoticed by the home viewer. When faced with instant

delay jumps of a frame or more, these old devices required several seconds or even minutes to attain new delay values, with the sample manipulation creating noticeable distortion the whole time. Consequently, the audio would be both out of sync and noticeably degraded for the duration of the time to make the change. In systems where large jumps in delay are frequently made, this is unacceptable performance.

In over to overcome the problems inherent with sample manipulation, and more importantly to preserve the integrity of AES/EBU digital audio, it is necessary to have 1:1 correspondence between input and output samples in the audio synchronizer.

The audio delay memory must store every audio sample which is taken by the A-D, or received on the digital input, and read every stored audio sample once and only once. In order to accomplish this task, the memory must have completely decoupled and asynchronous reading and writing, so that the reading rate can be faster or slower than the storing rate. By varying the reading rate with respect to the storing rate the delay time can be controlled, by causing the reading to catch up with the storing (to decrease delay) or to lag behind the storing (to increase the delay). In digital systems, this must be performed with the obviously inconsistent requirement of maintaining the clock rate at the correct frequency. Varying the reading rate with respect to the storing rate creates an annoying pitch change artifact, and requires re-clocking audio to maintain the proper output clock rate for digital audio.page 4 In theory, to make the pitch change resulting from the memory read rate change indistinguishable to the viewer, it is necessary to limit the differential rate between memory storing and reading to keep the associated audio pitch change very small. Unfortunately, if the differential rate between memory storing and reading is small, the amount of time required to change delay settings is correspondingly large.

It would be possible to modulate the relative reading rate in response to the audio signal content since larger ratios may be tolerated if no high frequency audio is present, or if there are periods of silence. Modulating the rate with the audio content does not provide a consistent significant improvement however, and frequently is of no advantage for any program material having a musical background.

In order to minimize perceptible pitch shifts during delay changes, to facilitate rapid large delay changes and to maintain proper clock frequencies for correction of AES/EBU digital audio, it is necessary that the audio delay incorporate a pitch correction circuit. With pitch correction, it is possible to make rapid delay changes and maintain proper output clock frequency with the pitch correction circuit removing corresponding audio pitch artifacts so they are unnoticed by the viewer.

One commercial product which incorporates pitch correction is the AD-3100 manufactured by Pixel Instruments Corp. of Los Gatos, CA. This device has selectable analog and AES/EBU digital inputs and simultaneous analog and digital outputs. It receives a DDO signal from a video instrument and adjusts the reading rate of the internal memory to increase or decrease the delay while at the same time providing digital signal processing pitch correction to maintain both proper pitch and output sample rate. In this device, multiple frame delay changes can be made in a matter of milliseconds without introducing artifacts or losing proper synchronization.

Conclusions

A major feature that differentiates multimedia applications from other traditional applications is the integration of various media streams that have to be presented in a synchronized fashion. One of the main problems that have to be addressed in a multimedia system is the way media streams are synchronized when they are presented to the users. In order to guarantee high-quality multimedia presentations, real-time support from the network and operating system is important, for this we concentrate on synchronization algorithms that consider best-effort environments, for two reasons: first, algorithms designed for non-real-time environments can also work in real-time environments. In this paper, we describe an efficient A/V synchronization algorithm that is quite useful for video telephony applications. The proposed method reduces the gap between temporal and special gaps between the multimedia media frames(Audio and video) and requires far less computation compared to the conventional algorithm. As shown in (6), the obtained decision rule is very easy to implement. Further the same algorithm can be implemented to other multimedia streams.

References

[1] C. Kim and K.-d. Seo, "An apparatus for synchronizing audio/video for hand-held devices," Korea Patent Application, P04-046697.

[2] C. Kim, S. Park and K.-d. Seo, “An efficient audio/video synchronization method for video telephony,” KISS Korea Computer Congress, July 2005.

[3] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, “Real-time transport protocol,” RFC 3550, IETF, July 2003.

[4] D. Mills, “Network time protocol specification, implementation and analysis,” RFC 1305, IETF, Mar. 1992.

[5] S. Furber, ARM System Architecture, Harlow, UK: Addison-Wesley, 1996.

[6] OMAP1510 Multimedia Processor (Technical Reference Manual), Dallas, TX: Texas Instruments, June 2002.

[7] C. Perkins, RTP: Audio and Video for the Internet, Boston, MA: Addison-Wesley Professional, 2003.

[8] M. Handley, V. Jacobson, “Session description protocol,” RFC 2327, IETF, Apr. 1998.

[9] Dr. Byron Reeves & Dave Voelker, research report Effects of Audio-Video Asynchrony on viewer's Memory, Evaluation of Content and Detection Ability (1993)

[10] International Telecommunication Union Document 10OC/32-E, 11A/43-E, 11C/40E, CMTT-C/18-E 5 October 1993

[11] International Telecommunication Union Document 11A/47-E, 13 October 1993.

[12] NAB Engineering Handbook, Television signal Transmission Standards (Washington, D.C.: National Association of Broadcasters), 621.

[13] E. Damiani, S. D. C. di Vimercati, S. Paraboschi, and P. Samarati. A fine grained access control system for xml documents. ACM Transactions on Information and System Security, 5, 2002.

[14] E. Damiani and S. D. C. di Vimercati. Securing xml based multimedia content. In 18th IFIP International Information Security Conference, 2003.

[15] E. Bertino, M. Braun, S. Castano, E. Ferrari, and M. Mesiti. Author-x: A java-based system for XMLdata protection. In IFIP Workshop on DatabaseSecurity, pages 15–26, 2000.

[16] Ramazan Sava Aygijn. An Integrated Framework for Interactive Multimedia Presentations in Distributed Multimedia Systems. In ACM, pages 1-581, '01 Proceedings of the ninth ACM international conference on Multimedia October 5 2001.

[17] Kuo-Yu Liu and Herng-Yow Chen. Exploring Media Correlation and Synchronization for Navigated Hypermedia Documents. annual ***ACM*** symposium on User interface software ACM 1-59593-044-2/05/0011 November 2005.

[18] Martin Hepp. Semantic Web and Semantic Web Services, IEEE Internet Computing, April 2006.

[19] Elina Megalou and Thanasis Hadzilacos. Semantic Abstractions In the Multimedia Domain IEEE Transactions on Knowledge and Data Engineering, vol. 15, no. 1, January/February 2003.

[20] Emilia Stoica, Hussein Abdel-Wahab and Kurt Maly. Synchronization of Multimedia Streams in Distributed Environments., IEEE, 0-8186-781949, 1997.