Exploration of the TI AM5728 Audio/Video Subsystem

Dennis Joosens
Department of Applied Engineering - Electronics-ICT
University of Antwerp
Antwerp, Belgium
dennis.joosens@student.uantwerpen.be

Theo Debrouwere
Televic
Izegem, Belgium
t.debrouwere@televic.com

Walter Daems
University of Antwerp
Antwerp, Belgium
walter.daems@uantwerpen.be

***Abstract*—Video conferencing in Full and Ultra High Definition with zero latency has become an emerging research area. In this survey paper we discuss and compare the audio/video encoders and decoders which are available in the Software Development Kit of the Texas Instruments AM5728 Evaluation Module. Our goal is to set up a proof of concept HD video conferencing system with a minor latency. Since the SDK does not provide a lot of options, our choice is made quite swiftly. We conclude that it should be viable to set up a video conferencing system having a certain end-to-end delay.**

I. Introduction

Over the past decades digital video has become mainstream and is being used in a wide range of applications such as video conferencing. If we would send raw video data over a network we would bump into some problems. For instance, if we have a screen resolution of 1920x1080 pixels with a frame rate of 30 fps, 60 minutes full color video, that would give a data quantity of approximately 671.85 GB. This translates to about 11.2 GB per minute and equals a data rate of about 1.49 Gbps. That amount of data would cause problems with the memory and storage capacity of many devices and will cause issues with the bandwidth of certain transmission channels [1]. The data needs to be processed in a certain way to eliminate these problems. Therefore several digital video and audio compression standards have been developed and are still being developed. These compression standards use certain techniques and algorithms to decrease the data quantity and still provide an acceptable video and audio quality. In collaboration with the company Televic, we will develop a proof of concept video conferencing system and attaining a low latency. For the execution of the project we will be using the Texas Instruments AM572x EVM board using the TI AM5728 Sitara processor. The evaluation module is based on a BeagleBoard-X15 extended with a 7 inch LCD capacitive touchscreen, a few buttons and a camera module. For the development we will be using the standard SDK that is available. The SDK has a limited support for codecs that can be used. In this paper we will discuss and compare the most used video and audio codecs which are supported by the development board.

II. H.264 and MPEG-4

MPEG-4 is a ISO/IEC video compression standard developed by the Moving Picture Experts Group. The first version of the MPEG-4 standard was finalized in 1998 [2]. Until this day, new parts are continuously added. The H.264/AVC standard, also called ISO MPEG-4 Part 10 was developed a few years later by the Joint Video Team. A collaboration between the ISO/IEC MPEG working group and the ITU-T Video Coding Experts Group. The standard was released in 2003 and was developed due to the need for higher coding efficiency which is caused by the increasing number of services and growing popularity of high definition applications [3]. Due to the fact that H.264 supports low-and high bitrate video coding the standard has become one of the most commonly used codecs for a widespread variety of products, applications and services [4], e.g. for recording, compression and distribution of video content over mobile networks as well for video conferencing [5]. H.264 provides good video quality at lower bit rates and claims to double the coding efficiency compared to previous standards [5], [6]. In general it has a smaller end-to-end delay compared to MPEG-4 [7]. The TI AM5728 EVM only supports MPEG-4 and H.264 encoding. We conclude that we will be using the H.264 codec for our application. The H.264 standard outperforms MPEG-4 part 2 in many ways [2], [8]. The standard is newer and has some significant improvements in coding efficiency and error robustness compared to MPEG-4.

III. MPEG-2 BC and MPEG-2 AAC

MPEG-2 audio compression consists out of two lossy coding standards which are slightly different. We have MPEG-2 BC which is backwards compatible with the MPEG-1 Audio Layers and MPEG-2 Advanced Audio Coding which is not constrained to this backwards compatibility and thereby has a better coding efficiency. In contrast to MPEG-2 BC, AAC supports more channels and more sampling rates. The AAC standard has been developed to be the successor of MPEG-1 audio, but in practice MPEG-1 audio compression is still used a lot [9]. Another advantage of AAC is that we can attain improved coding efficiency at very low data rates. The evaluation board that we will be using supports MPEG-2 and MPEG-2 AAC, thereby MPEG-4 and H.264

both use MPEG-2 AAC so the choice for the type of audio compression that we will be using is clear [9].

IV. H.265

It is quite common that video standards are further developed and improved until the standard reaches its limits and eventually will become obsolete [4]. As a result, several working groups are still innovating and developing new standards that will be used in the future. For example, after the release of the H.264 standard, the Joint Collaborative Team on Video Coding which was formed by the same organizations who created H.264, started with the development of a new video compression standard called H.265. This new standard also known as MPEG-H Part 2 / High Efficiency Video Coding is the successor of H.264 and will eventually replace it completely. The goal was to increase coding efficiency and doubling the video compression performance compared to H.264 while maintaining the same level of video quality [10]. The inner working of H.265 is based on H.264 but has many incremental improvements. There are essentially two major improvements that are very important to get higher compression rates. HEVC uses more efficient and flexible macroblocks called Coding Units and provides more intra-predictive directions which decreases the bit rate but having a higher picture quality. In general, the bit rate reduction at the same level of visual quality compared to H.264 is between 40% and 50%. In practice, there can be an average bit rate saving between 39.3% to 59% [4], [11]. In the end it all depends on the characteristics of the used video content [10], [11]. However H.265 comes with a cost. It is more difficult to encode and requires more processing power [10], [12]. This eventually can be marginalized because computers have become more powerful since the release of the H.264 standard. Eventually, H.265 will become the new leading standard. Although to this point in time it is not, H.264 has been approved in 2004 and is now widely used. H.265 is released in 2013 and it takes a few years until it becomes widely used. Manufacturers have already started implementing the new standard. It will be mainly used for 4K/Ultra HD broadcasting and streaming applications. Some applications that already make use of H.265 are Netflix, Amazon and the Xbox One. Nonetheless, the standard has a high potential but we will not be using it due to the fact that it is not completely supported by the TI module.

V. Latency

When using real time audio/video applications we do not want to wait for several seconds to interact with the person on the other side. When a latency becomes noticeable for an end user, it can make a conversation troublesome and in some cases even impossible. It is crucial to minimize the overall latency to preserve a good user experience. Latency is declared as the amount of time between a captured frame and the time when the frame is displayed and is usually measured in milliseconds. In video conferencing systems there are three main components that form the end-to-end latency [13]. Namely, the encoding delay that compresses the audio and video data, the network delay and the decoding delay that originates from the decompression process. A delay smaller than 100 ms is considered as low latency and is not perceived by humans [14]. A latency between 100 - 150 ms is perceived as instantaneous for an end user. When the delay gets higher than 150 ms, it becomes noticeable. A delay bigger than 300ms is unacceptable and considered aggravating. Other sources say that the end-to-end delay should not be bigger than 250 ms [7], [15]. In many cases a small noticeable delay can be acceptable. However in applications where machines or people interact with video e.g. in the automotive industry, medical systems or video conferencing, we want a low value latency. At this time of writing one of the most used video codecs for streaming video content is H.264/AVC. For the implementation of the project we will be using the H.264 baseline profile. H.264 has been designed with low latency applications in mind and should be resilient to packet losses. The baseline profile only uses I- and P-frames which makes it ideal for low latency applications. We can reduce the latency further by using a smaller screen resolution and applying a higher frame rate. The expected latency without modifications will be between 100 ms and 200 ms [15] without modification of the codec. In this project we want to keep the end-to-end latency beneath 25 ms which will be hard to accomplish. However there are several successful implementations where the end-to-end latency is beneath 10 ms, so called zero latency. One of them is the Taos H.264 codec from W&W Communications. The Taos architecture can encode and decode a 1080p60 video simultaneously. It starts encoding and decoding when just a line of pixels is received and does not wait as common codecs for a full frame. This way of processing decreases the encode-decode latency to sub 2 ms while not compromising the video quality [16].

VI. Conclusion

For the development of our video conferencing system we can conclude that we will be using H.264 to encode and decode the data stream in real time. The supplied SDK of the board supports a few codecs e.g. MPEG-4, H.264, VC-1. We choose the H.264 due to its supreme compression and versatility in contrast to the other codecs. H.265 should be even better but until this moment, the codec is not fully supported by the TI module. For the encoding and decoding of the audio we will be using AAC which is integrated into the H.264 standard and is fully supported by the development board. To decrease the end-to-end latency we will try to modify and implement several changes into the H.264 codec.

References

[1] S. Ponlatha and R. S. Sabeenian, "Comparison of Video Compression Standards," *International Journal of Computer and Electrical Engineering*, vol. 5, no. 6, pp. 549–566, 2013.

[2] D. Dembla, B. Patel, A. Kumar, and D. Y. Bhomia, "Comparison of H.264 and MPEG-4 Codec Based on PSNR- Peak Signal to Noise Ratio Algorithm," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 3, no. 3, pp. 365–370, 2013.

[3] T. Wiegand, G. J. Sullivan, and G. Bjøntegaard, "Overview of H.264 / AVC Video Coding Standard," vol. 13, no. July, pp. 560–576, 2003.

[4] D. Grois, D. Marpe, A. Mulayoff, B. Itzhaky, and O. Hadar, "Performance comparison of H.265/MPEG-HEVC, VP9, and H.264/MPEG-AVC encoders." *Pcs*, vol. 2013, no. Pcs, pp. 394–397, 2013.

[5] R. Bahirat and A. Kolhe, "Video Compression using H.264 AVC Standard," vol. 9359, no. 2, pp. 31–37, 2014.

[6] J. Zeng, O. C. Au, W. Dai, Y. Kong, L. Jia, and W. Zhu, "A Tutorial on Image / Video Coding Standards," 2000.

[7] S. Rowshanrad, S. Namvarasl, B. Jamasb, and M. Keshtgary, "Video Codec Standards Comparison for Video Streaming Over SDN," vol. 5, no. 1, pp. 10–15, 2015.

[8] O. Nemčić, M. Vranješ, and S. Rimac-Drlje, "Comparison of H.264/AVC and MPEG-4 Part 2 coded video," *Proceedings Elmar - International Symposium Electronics in Marine*, no. September 2014, pp. 41–44, 2007.

[9] K. Brandenburg, "MP3 and AAC Explained," *Audio Engineering Society, 17th International Conference 2004 October 2629*, pp. 99–110, 1999.

[10] M. Uhrina, J. Frnda, L. Sevcik, and M. Vaculik, "Impact of H.264 / AVC and H.265 / HEVC compression standards on the video quality for 4K resolution Impact of H.264 / AVC and H.265 / HEVC Compression Standards on the Video Quality for 4K resolution," no. October 2015, 2014.

[11] T. K. Tan, S. Member, R. Weerakkody, M. Mrak, S. Member, N. Ramzan, S. Member, V. Baroncini, J.-r. Ohm, G. J. Sullivan, A. The, H. Efficiency, and V. Coding, "Video Quality Evaluation Methodology and Verification Testing of HEVC Compression Performance," vol. 26, no. 1, pp. 76–90, 2016.

[12] M. Semsarzadeh, H. Roodaki, A. Aminlou, and M. R. Hashemi, "A Receiver Aware H.264 / AVC Encoder for Decoder Complexity Control in Mobile Applications," no. November, 2016.

[13] R. M. Schreier and A. Rothermel, "A Latency Analysis on H.264 Video Transmission Systems," *2008 Second International Conference on Electrical Engineering*, pp. 1–2, 2008.

[14] M. M. Al-n, "A Guideline to Video Codecs Delay," vol. 4, no. 2, pp. 5–8, 2014.

[15] J. Wu, J. Yang, X. Wu, and J. Chen, "A Low Latency Scheduling Approach for High Definition Video Streaming over Heterogeneous Wireless Networks," *Globecom2013-Communication Software, Services and Multimedia Symposium*, no. December, pp. 1723–1729, 2013.

[16] Taos, "Taos - A Revolutionary H.264 Video Codec Architecture For 2-Way Video Communications Applications," 2008.