CORE
Metadata, citation and similar papers at core.ac.uk
Provided by Universiti Teknikal Malaysia Melaka (UTeM) Repository

LOW COST MULTI-VIEW VIDEO SYSTEM FOR WIRELESS CHANNEL

Nurulfajar Abd Manap, Gaetano Di Caterina, John Soraghan

Centre for Excellence in Signal and Image Processing (CeSIP), EEE, University of Strathclyde, UK

ABSTRACT

With the advent in display technology, the 3DTV will provide a new viewing experience without the need of wearing special glasses to watch the 3D scenes. One of the key elements in 3DTV is the multi-view video coding, obtained from a set of synchronized cameras, capture the same scene from different view points. The video streams are synchronized and subsequently used to exploit the redundancy contained among video sources. A multi-view video consists of components for data acquisition, compression, transmission and display. This paper outlines the design and implementation of a multi-view video system for transmission over a wireless channel. Synchronized video sequences acquired from four separate cameras and coded with H.264/AVC. The video data is then transmitted over a simulated Rayleigh channel through Digital Video Broadcasting –Terrestrial (DVB-T) system with Orthogonal Frequency Division Multiplexing (OFDM).

Index Terms— Multi-view video, H.264/AVC, wireless channel, low-cost implementation.

1. INTRODUCTION

The demand for multi-view video coding is driven by the development in new 3D display technologies and the growing use of multi-camera arrays. A variety of companies are starting to produce 3D display technologies that do not require special glasses and can be viewed by multiple people simultaneously. This technology provides a good platform for new applications to emerge such as 3D scene communication [1]. Even with 2D displays, multi-camera arrays are increasingly being used to capture a scene from many angles. The resulting multi-view data sets allow the viewer to observe a scene from any viewpoint and serve as another application of multi-view video compression.

Multiple camera views of the same scene require a large amount of data to be stored or transmitted to the user. Furthermore for real-time multi-view video processing it demand extensive processing capabilities. Multi-view video data is also predicted to consume a large portion of the bandwidth available in the future [1]. Therefore, efficient compression techniques are essential. The simplest solution for this would be to encode all the video signals independently using a state-of-the-art video codec such as H.264/AVC [2, 3]. However, this is inefficient as it does not exploit the correlation or inter-view statistical dependencies that exist in the multi-views. As the video data is taken from the same scene, the correlations of the multi-view scenes can be exploited for efficient compression. The correlations and redundancies can be categorized into two types, inter-view redundancy between adjacent camera views and temporal redundancy between temporal successive images of each video. These redundancies can be exploited, where images are not only predicted from temporal neighboring images but also from corresponding images in adjacent views, referred to as multi-view video coding (MVC).

In this paper, a multi-view video system for wireless applications will be presented. The system consists of components for data acquisition, compression, transmission and display. The main features of the system includes wireless video transmission system for up to four cameras, by which videos can be acquired, encoded and transmitted wirelessly to a receiving station. The video streams can be displayed on a single 3D or on multiple 2D displays. The encoding for the multi-view video through inter-view and temporal redundancies increased the compression rates. The H.264/AVC multi-view compression techniques has been exploited and tested during the implementation process. One of the highlight in this paper is the low cost implementation of a multi-view video system, which using only a typical web cameras attached to a single PC.

2. MULTI-VIEW VIDEO COMPRESSION

Many 3DTV systems are based on scenarios, where a 3D scene is captured by a number of N cameras [4]. The simplest case is classical stereo video with two cameras. More advanced systems apply 8, 16 and more cameras. Some systems traditionally apply per sample depth data that can also treated as video signals. An overview of compression algorithms and standards can be found in [5], which includes the conventional stereo video coding, video plus depth data and multi-view video coding. Depending on the degree of common content shared by a subset of the cameras, a coding gain can be achieved in comparison to single-view coding.

Multi-view video compression algorithms aim to reduce redundancy in information from multiple views, to provide a high degree of compression. As multi-view videos capture the same dynamic 3D scene, similarities and redundancy exists among frames. The redundancy in multi-view video streams consists of inter-view redundancy (between adjacent camera views) and temporal redundancy (between temporally successive frames of each video) [1]. These types of redundancy can be exploited to obtain a combined temporal/inter-view prediction. Other types of redundancy include transform redundancy and redundancy of human visual system [4].

Encoding and decoding separately each view of a multi-view video data separately can be done with any existing standard, such as with H.264/AVC, where each camera view of the sequence is coded independently, just like a normal video stream. This technique is referred to as simulcast coding [6]. This would be a simple, but inefficient way to compress multi-view video sequences, because inter-view statistical properties are not taken into consideration.

In multi-view coding, correlations between adjacent cameras are exploited in addition to temporal correlations within each sequence. The multi-view video coding adds another compression dimension on the top of single-view coding: the inter-view direction. Exploiting redundancies among the multi-view video images is the key to efficient compression. Therefore, for this research, the multi-view video coding has been selected due to the fact that this technique provides a high compression rates compared to the simulcast coding. The video data will be compressed with H.264/AVC algorithms before transmitted over the wireless channel.

In the next section, the implementation and system architecture of multi-view video for wireless channel will be described.

3. SYSTEM ARCHITECTURE

The proposed multi-view video system shown in Figure 1 mainly consists of four video cameras, one acquisition PC, multi-view codec with error protection and correction, transmission, reception and display. These components can be classified into four modules: acquisition, data encoding and decoding, error protection and correction, and lastly the display.

The acquisition stage consists of an array of hardware synchronized cameras. All the cameras are connected to a single PC through the USB connection. The PC captured live and uncompressed video streams through Matlab environment. The captured video streams can be played within the Matlab player. The video streams encoded by using standard H.264/AVC. The compressed video streams can be broadcast on separate channels over a transmission on network. In this project, it was transmitted over the wireless channel. Error protection and correction was simulated through the DVB-T standard with Rayleigh channel. For initial implementation, each video streams will be encode/decode in the same PC and displayed at 2D display due to the limitation of the equipment.

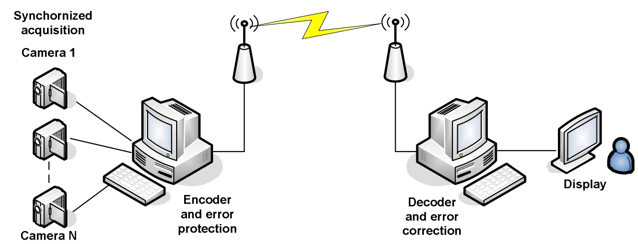

Figure 1 - General scheme of the proposed system

3.1. Test Sequences

Multi-view video sequences were captured using four cameras in the lab room. Each camera captures progressive video in real-time simultaneously through Matlab. There are four Creative Live! Color cameras that provide 800x600 resolution and provide output up to 30 frames per second at full resolution. The cameras were connected by USB port to the computer and synchronized by software controller through Matlab command source code. The video sequences were stored in a hard disc in the single computer. For the initial stage condition, only the normal lighting in the lab room will be used. All cameras were color calibrated by white balancing by the Creative Live! Camera Manager Auto tuning software, which automatically adjusts lighting and brightness. The cameras were positioned in a regularly spaced linear array. The distance between neighboring camera positions was set to 10 cm.

As mentioned earlier, H.264/AVC is the state-of-the-art video coding standard for monoscopic video. Most of the representations of 3D video are coded using variants of this codec. Simulcast coding uses several streams that encoded by H.264/AVC independently. The multi-view coding encoder implementation used in the system is JM H.264/AVC software version 10, which is the reference software for MVC. It uses prediction structure of hierarchical B pictures for each view in temporal direction [7].

The multi-frame referencing is the key property of the H.264/AVC standard that enables prediction of blocks of a P-frame being coded using a previous I-frame of multiple previous coded P-frames. The fact that there are high correlations among different views of a multi-view sequence led the development of a H.264/AVC based on multi-view video coding technique with 5 modes of operation by

MMRG H.264 Multi-view Extension Codec developed by the team of Multimedia Research Group at METU [8].

3.2. Compression, Transmission and Display Testbed Design

The system developed not in a real-time implementation. Therefore, the whole process from acquisition to display could be divided into several steps; each of it could be implemented in a software module written in Matlab. The steps identified within the process were acquisition of two to four synchronized video streams, encoding of the video streams using multi-view with the H.264/AVC format, error protection applied to the single video stream as defined in DVB-T standard, transmission (with the simulation of the effect of a Rayleigh channel on the error protected data stream), error correction in the received data stream, decoding of video stream received and display the video streams.

The whole system can be simplified as shown in Figure 2, which consist all the modules. The system divided into several modules for some practical reasons. Such modules match naturally the functions provided by real devices used in a real implementation of the system. The division of a major problem into a set of smaller problems reduces design and implementation complexity of the original problem. By defining self-contained modules, it helped the debugging process and allows reusability. In addition, it also simplifies code maintenance and modification.

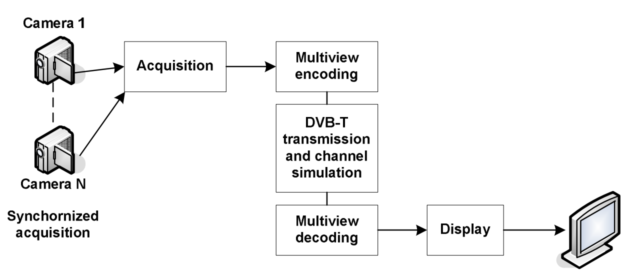

Figure 2 – Block diagram of system design

4. RESULTS

The first goal of the test is to ensure that the H.264/AVC reference software could handle the multi-view video streams. Every view of the cameras contains 50 frames in YUV format and CIF size captured at 15 fps. The simulcast coding is achieved by coding each view sequence separately using H.264/AVC standard. The quality of the encoded sequences was measured by the average PSNR of their frames.

The parameters that have been set for the encoding process were 352x288 image format, 16 search range, IPPP sequence type and full Motion Estimation scheme search. A set of multi-view video sequences, called 'Book' was captured. It contains of four views of 50 frames each at 15 fps. The sequences were set to produce CIF size sequences. To illustrate the nature of the captured data sets, frames 25 of the four cameras from 'Book' sequences are shown in Figure 3. With H.264 Analyzer released by MMRG team, the output of the coded multi-view video was shown below. The tool simply labels the macroblocks with selected colors to distinguish which camera sequence they are referred from.

(a) Camera 1 (b) Camera 2 (c) Camera 3 (d) Camera 4

Figure 3 - Frame 25 of the four cameras from Book sequences.

Table 1- Simulation Results for Simulcast Coding

Camera Views	**Book 0**	**Book 1**	**Book 2**	**Book 3**
Total encoding time (fps)	0.07	0.06	0.06	0.07
Total ME time for sequence (sec)	737.630	750.381	764.662	705.766
SNR Y (dB)	39.82	40.75	41.14	41.66
Total bits	325,192	211,160	187,752	150,312
Bit rate (kbit/s)	195.12	126.70	112.65	90.19

Table 1 illustrates the output through simulcast coding. Meanwhile Table 2 provides the simulation results for the Book sequences by using different reference modes of the multi-view video coding H.264/AVC. Four input files (from four different camera inputs) with the YUV 4:2:0 format coded with the search range of the macroblock by 16. The total number of frames is 200.

Table 2 – Simulation Results for Multi-view Coding with Different Modes

Reference Mode	**Mode 1**	**Mode 2**	**Mode 3**	**Mode 4**	**Mode 5**
Total encoding time (fps)	0.14	0.33	0.23	0.26	0.11
Total ME time for sequence (sec)	1343.474	549.732	807.381	698.489	1768.547
SNR Y (dB)	40.53	40.51	40.50	40.92	40.49
Total bits	905,528	939,104	936,696	1,140,888	937,120
Bit rate (kbit/s)	135.83	140.87	140.50	171.13	140.57

From the results, it shown that different reference mode yields a difference performance. The bit rate for the Mode 4 higher compared to the remaining mode, even though the SNR of each mode almost similar. The total encoding time for the Mode 1 produce faster encoding time, which is 0.14 fps. The total bits for the simulcast coding seem to be smaller compared to the output in inter-view coding because it was only for single sequence. The total number of bits will increase due to the summation of all the compressed data 4 views to transmit or storage. This is inefficient way to compress the multi-view video because it did not exploits the redundancies between the multiple views.

The reference software will be modified in the next future work to enable the sequence type from IPPP to the other prediction structure. The Da Vinci technology in the Texas Instruments DSP processor will be used as the main platform for the multi-view video coding implementation, particularly for encoding/decoding process. With the new system, the real-time implementation can be achieved.

5. CONCLUSIONS

A multi-view video coding simulation based on H.264/AVC for wireless channel has been presented. The coding scheme processed the frames of sequences captured by multiple cameras from a scene. The codec is based on the JM H.264/AVC software version 10. Five modes of operation are simulated based on the MMRG H.264 Multi-view Extension. The acquisition stage consists of an array of synchronized cameras that are connected to a single PC through the USB connection. One of the highlight in this research was that the implementation cost for this system is quite low since it used a typical web camera attached to the PC. The system can be upgraded to higher state with better specification of the equipment from acquisition to display.

6. ACKNOWLEDGEMENTS

Our thanks to Multimedia Research Group at Middle East Technical University (METU) of Ankara for providing the MMRG H.264 Multi-view Extension software and Karsten Sühring from Fraunhofer Institute for Telecommunications, Heinrich-Hertz-Institut for H.264/AVC source code.

7. REFERENCES

[1] M. Flierl and B. Girod, "Multiview Video Compression," *Signal Processing Magazine, IEEE*, vol. 24, pp. 66-76, 2007.

[2] P. Merkle, A. Smolic, K. Muller, and T. Weigand, "Efficient Prediction Structures for Multi-View Video Coding," *IEEE Transactions on Circuits and Systems for Video Technology*, 2007.

[3] G. Li and Y. He, "A Novel Multi-View Video Coding Scheme Based On H.264," *Proceedings of the Joint Conference of the Fourth International Conference on Information, Communications and Signal Processing,* vol. 1, pp. 15-18, 2003.

[4] I. Ahmad, "Multi-View Video: Get Ready for Next-Generation Television," *Distributed Systems Online, IEEE*, vol. 8, pp. 6-6, 2007.

[5] H. Y. Shum, S. B. Kang, and S. C. Chan, "Survey of Image-Based Representations and Compression Techniques," *IEEE Trans. Circuits Systems Video Technology*, vol. 13, pp. 1020-1037, Nov. 2003.

[6] M. Flierl, A. Mavlankar, and B. Girod, "Motion and Disparity Compensated Coding for Multiview Video," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 17, pp. 1474-1484, 2007.

[7] H. Schwarz, D. Marpe, and T. Wiegand, "Analysis of Hierarchical B Pictures and MCTF," *International Conference on Multimedia and Expo*, 2006.

[8] C. Bilen, A. Aksay, and G. B. Akar, "A Multi-view Video Codec Based on H.264," *IEEE ICIP 2006*, Oct. 2006.