Multi-view Video Coding System for Wireless Channel

Nurulfajar Abd Manap[ab], **Gaetano Di Caterina**[a], **John Soraghan**[a]

[a]Centre for Excellence in Signal and Image Processing (CeSIP), Electronic and Electrical Engineering Department, University of Strathclyde, Glasgow, UK.
[b]Faculty of Electronic and Computer Engineering, Universiti Teknikal Malaysia Melaka, Malaysia.
fajar@eee.strath.ac.uk, gaetano-dicaterina@strath.ac.uk, j.soraghan@eee.strath.ac.uk

Keywords: multi-camera, multi-view video coding, image processing, prediction structure, H.264/AVC.

Abstract

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 feature 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. 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). One of the highlights in this paper is the low cost implementation of a multi-view video system, which using only typical web cameras attached to a single PC.

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. This technology provides a good platform for new applications to emerge such as 3D scene communication. 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 [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. These redundancies can be exploited, where images are not only predicted from temporal neighbouring images but also from corresponding images in adjacent views, referred to as multi-view video coding (MVC).

2 Simulcast and Multi-view Video Compression

Many 3D video 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. And 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 standard can be found in [5].

In multi-view video system, the video streams must be synchronized to ensure that all the cameras' shutters open at the same instant time when they are sampling the scene from different angles. Video captured from different cameras in used together with timing information to create novel views in multi-view video. The input from the cameras can be synchronized using external sources such as a light flash at periodic intervals. External synchronization can slow down the frame rate considerably.

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. In multi-view coding, correlations between adjacent cameras are exploited in addition to temporal correlations within each sequence: the inter-view direction.

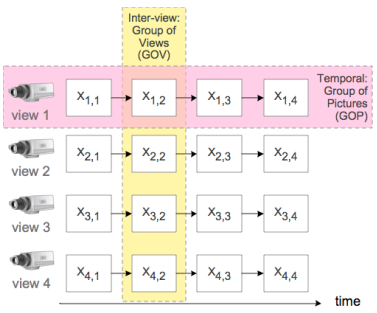


Figure 1. Matrix of pictures (MOP) with 4 camera views

In order to understand the temporal and inter-view correlation and redundancies between adjacent camera views and each video, a simpler version of temporal and inter-view prediction structure is shown in Figure 1. The classification of the redundancies based on the normal arrangement of multi-view video images into a matrix of pictures (MOP) [6]. Each row holds temporally successive pictures of one view, and each column consists of spatially neighbouring views captured at the same time. It depicts a matrix of pictures for $N = 4$ image sequences, each composed of $K = 4$ temporally successive pictures. $N = 4$ views form a group of views (GOV), and $K = 4$ temporally successive pictures form a temporal group of pictures (GOP). For example, the images of the first view sequence are denoted by $x_{1,k}$, with $k = 1,2,...,K$.

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 (the temporal group of pictures, GOP) is coded independently, just like a normal video stream as shown in Figure 1. 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. Meanwhile, the multi-view coding should reduce redundancies in information from multiple views as much as possible to provide high degree of compression. These redundancies can be exploited with temporal (GOP) and inter-view prediction (GOV) combination.

H.264/AVC is the state-of-the-art video coding standard for monoscopic video. Most of the representations of 3D videos are coded using variants of this codec. It uses prediction structure of hierarchical B pictures for each view in temporal direction [2]. The concept of hierarchical B pictures was introduced by [3]. A typical hierarchical prediction structure with three stages of a dyadic hierarchy is depicted in Figure 2, where I (intra-coded pictures) with P or B (inter-coded pictures) for temporal prediction in video coding. The first picture of a video sequence is intra-coded as IDR picture and so-called key pictures are coded in regular intervals. A key picture and all pictures that are temporally located between the key picture and the previous key picture are considered to build a group of pictures (GOP), as illustrated in Figure 2 for a GOP of eight pictures.

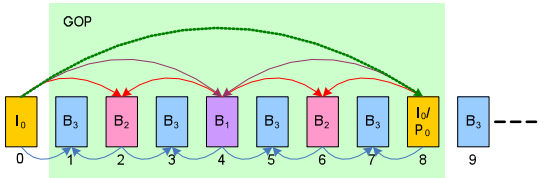


Figure 2. Hierarchical reference picture structure for temporal prediction

For this research, the multi-view video coding has been selected since 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. The next section will discuss the multi-view video coding system developed for the wireless channel with different multi-view modes of operation.

3 Multi-view Video Coding System

3.1 System Architecture

The proposed multi-view video system shown in Figure 3 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.

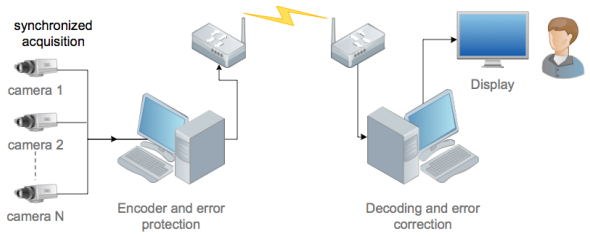


Figure 3. General scheme of the proposed system

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. 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 positioned in a regularly spaced linear array. The distance between neighbouring camera positions was set to 10cm.

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.

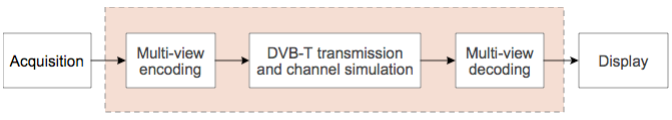


Figure 4. Block diagram of system design

The whole system can be simplified as shown in Figure 4, 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.

3.2 Multi-view Modes of Operation

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 multi-view video coding. It uses prediction structure of hierarchical B pictures for each view in temporal direction [3] as illustrated in Figure 5, for a sequence with 7 cameras and a GOP length of 8, where S_n denotes the individual view sequences and T_n the consecutive time points.

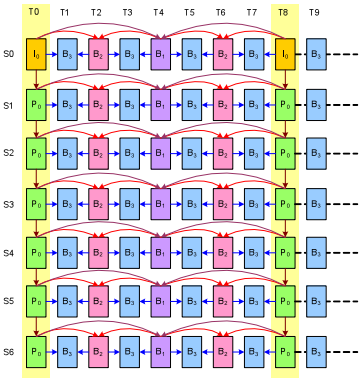
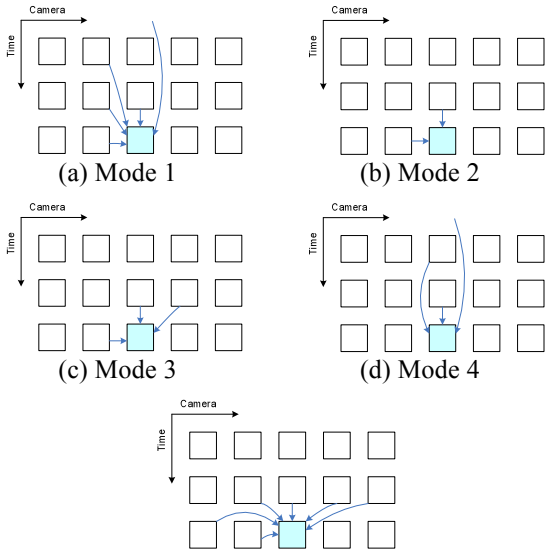


Figure 5. Multi-view video coding prediction structure

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 as shown in Figure 5. 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 [7].

Five different mode of operation of the H.264/AVC based on multi-view video coding scheme illustrated in Figure 6. A block diagram of Mode 1 of operation is shown in Figure 6(a), where the previous frames of closest camera sequence in addition to previous frames of the encoded camera sequence. Figure 6(b) shows Mode 2 operation, where the latest frame from one nearby camera and latest frame from encoded camera sequence are used. For Figure 6(c), the latest frames from two nearby cameras and latest frame from encoded camera sequence are used. Meanwhile Figure 6(d) illustrates the only the previous frames of the encoded camera sequence are used in Mode 4. Lastly, in Mode 5, which is shown in Figure 6(e), the latest frames from all the cameras in addition to one more frame from one of the closest cameras are used.

The next section will discussed some results and simulation based on the H.264/AVC reference software with five different modes of operation.



(a) Mode 1 (b) Mode 2

(c) Mode 3 (d) Mode 4

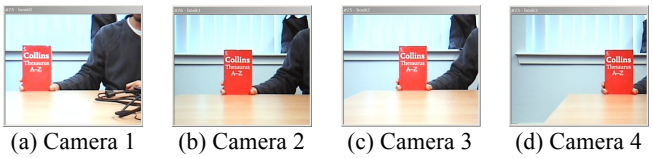
(e) Mode 5

Figure 6. Reference modes in H.264/AVC multi-view extension codec

4 Results and Discussion

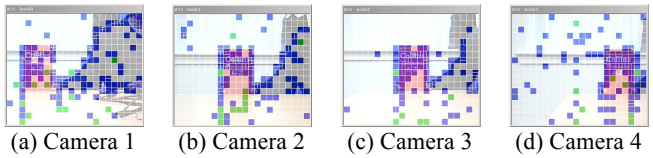
The first goal of the test is to ensure that the H.264/AVC reference software could handle the multi-view video streams. Every view of the cameras contains 50 frames in YUV format and CIF size captured at 15 fps. The simulcast coding is achieved by coding each view sequence separately using H.264/AVC 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. To illustrate the nature of the captured data sets, frames 25 of the four cameras from 'Book' sequences are shown in Figure 7.



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

Figure 7. Frame 25 of the four cameras from Book sequences



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

Figure 8. Output of the coded multi-view video with H.264 Analyzer at frame 25

With H.264 Analyzer released by MMRG team, the output of the coded multi-view video can be shown in Figure 8. The tool simply labels the macroblocks with selected colours to distinguish which camera sequence they are referenced from:

inter-view or intra-view camera. The frame displayed in block of colours that shows the referenced video sequences in the multi-view configuration. The colours for the corresponding cameras customized in the H.264 Analyzer for intra-coded and skipped blocks of the sequences.

Table 1 provides the simulation results for the Book sequences by using simulcast coding and Table 2 for 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 result for the simulcast coding in Table 1 obtained from each camera views where the video coded independently. The total encoding time is between 720 to 800 seconds for each camera views.

Camera Views	Camera 1	Camera 2	Camera 3	Camera 4
Σ encoding time (sec)	760	772	788	726
SNR Y (dB)	39.82	40.75	41.14	41.66
Σ bits	325,192	211,160	187,752	150,312
Bit rate (kbit/s)	195.12	126.70	112.65	90.19

Table 1. Simulation Results for Simulcast Coding

In Table 2, the results obtained based on the multi-view video coding for different reference modes. The different parameter used between the result of Table 1 and Table 2 is the total number of frames. In simulcast coding, the total number of frames for each camera views is 50 frames. Meanwhile, for multi-view video coding, the total number of frames is 200 frames for each mode, which is by combining all frames of the four cameras (50 frames for each view).

Reference Mode	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5
Σ encoding time (sec)	1407	605	864	770	1842
SNR Y (dB)	40.53	40.51	40.50	40.92	40.49
Σ 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

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

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 2 produce faster encoding time, which is 605 seconds (with the total number of 200 frames for each mode). 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.

5 Conclusion

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. 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.

Acknowledgements

The main author wishes to thank the Ministry of Higher Education Malaysia and Universiti Teknikal Malaysia Melaka for the sponsorship.

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

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