Frank Stajano
Hyoung Joong Kim
Jong-Suk Chae
Seong-Dong Kim (Eds.)

LNCS 4412

Ubiquitous Convergence Technology

First International Conference, ICUCT 2006
Jeju Island, Korea, December 2006
Revised Selected Papers

Springer

Lecture Notes in Computer Science 4412

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Frank Stajano Hyoung Joong Kim
Jong-Suk Chae Seong-Dong Kim (Eds.)

Ubiquitous Convergence Technology

First International Conference, ICUCT 2006
Jeju Island, Korea, December 5-6, 2006
Revised Selected Papers

Springer

Volume Editors

Frank Stajano
University of Cambridge
Cambridge CB3 0FD, United Kingdom
E-mail: fms27@cam.ac.uk

Hyoung Joong Kim
Korea University
Seoul 136-701, Korea
E-mail: khj-@korea.ac.kr

Jong-Suk Chae
ETRI
Daejeon, 305-700, Korea
E-mail: jschae@etri.re.kr

Seong-Dong Kim
KETI/Ubiquitous Research
Gyeonggi-do 463-816, Korea
E-mail: sdkim@keti.re.kr

Library of Congress Control Number: 2007923849

CR Subject Classification (1998): C.2, C.3, D.4, D.2, K.6.5, H.5.3, K.4

LNCS Sublibrary: SL 3 – Information Systems and Application, incl. Internet/Web and HCI

ISSN 0302-9743
ISBN-10 3-540-71788-9 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-71788-1 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media

springer.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 12044061 06/3180 5 4 3 2 1 0

Preface

Ubiquitous computing is already with us and is changing our lifestyle, way of thinking and quality of life. Everyday objects with embedded computing capabilities are now commonplace and, between mobile phones and RFID tags, further deployment proceeds at an unstoppable pace. The next major step of the ubiquitous computing evolution is the move, already partly underway, from isolated smart objects to distributed systems of smart objects and appropriate back-end infrastructure: microelectronics and communication technology converging with healthcare technology, communication technology, sports and entertainment, housing, vehicular technology, middleware, sensor networks and so on.

You will have noticed that many people in the field now use the word "ubiquitous" not to mean "present everywhere" but as a shorthand for "ubiquitous computing and communications"—leading to otherwise inexplicable locutions such as "the ubiquitous society". Rather than continuing to fight this synecdochical use we have chosen to go with the flow, in so far as the change in language is an indication of the global spread of the meme. We have therefore chosen "ubiquitous convergence" as a concise description of the above view: a systems-oriented perspective encompassing both the technology and its applications.

The First International Conference on Ubiquitous Convergence Technology (ICUCT) was held on Jeju Island, Korea on December 5–6, 2006. This was the first conference organized by the Institute of Electronics Engineers in Korea (IEEK) to celebrate its 60^{th} anniversary. This conference was organized to pave the way for the ubiquitous society by contributing to the development of ubiquitous technologies and their integration in the appropriate application domains. This volume collects the post-proceedings of the conference.

At ICUCT 2006 we accepted only 30 papers from around 640 submissions. We believe the acceptance rate of less than 5% is a clear indication of our commitment to ensuring a very high quality conference. This would not have been possible without the support of our excellent Technical Program Committee members who accurately reviewed and ranked an extraordinarily high number of papers under pressing deadlines. We express our extreme gratitude to all the Program Committee members for their dedication and hard work.

Due to the overwhelming number of submissions, it was impossible to evaluate all papers in one pass in the usual way. Thus, the evaluation process was twofold. In the first round, each reviewer reviewed and classed around 30 papers. After the first round of evaluation, 90 papers were selected. In the second round, 30 papers were accepted. One no-show paper was excluded from the post-proceedings. In addition, we invited Hide Tokuda and Yo-Sung Ho to deliver keynote talks, and we thank them for their valuable contributions. Yo-Sung Ho also wrote up his talk as an invited paper. This volume therefore contains one invited paper and 29 refereed papers.

All accepted authors were asked to revise and update their papers after the conference based on the written comments from the reviewers and on the formal and informal feedback they received at the conference from other attendees following their presentation. In choosing which papers to accept we tried to achieve a balance among important topics while keeping the paper quality high. Mobile and wireless communication techniques, multimedia technologies, security issues, RFID, sensor networks, applications and convergence aspects of relevant technologies are covered in this conference. These papers address both theoretical and practical issues which, we believe, are of broad interest to our community.

We hope the reader will find this volume to be a timely collection of quality papers that will help to advance the field of ubiquitous convergence technology.

December 2006

Frank Stajano
Hyoung Joong Kim
Jong-Suk Chae
Seong-Dong Kim

Conference Organizers

Conference Chair

Injung Park (DKU, Korea)

Technical Program Committee Chairs

Frank Stajano (University of Cambridge, UK)
Hyoung Joong Kim (Korea University, Korea)

Financial Committee

Kyle J. Kim (KETI, Korea)

Publications Committee

Soo-Hyun Park (Kookmin University, Korea)
Beongku An (Hongik University, Korea)

Local Arrangement Committee

Sang Yep Nam (Kyungmoon College, Korea)
Do Hyun Kim (Jeju National University, Korea)

Technical Program Committee

Beongku An, Hongik University, Korea
Alastair Beresford, University of Cambridge, UK
Ki-Seok Chung, Hanyang University, Korea
Jon Crowcroft, University of Cambridge, UK
Anind Dey, Carnegie Mellon University, USA
Nigel Davies, Lancaster University, UK
Alois Ferscha, Johannes Kepler Universität Linz, Austria
Christian Flörkemeier, ETH Zurich, Switzerland
Enrico Gregori, CNR, Italy
Marco Gruteser, Rutgers University, USA
Yo-Sung Ho, GIST, Korea
Yu Huihua, Fudan University, China
Young Huh, KERI, Korea
Sozo Inoue, Kyushu University, Japan
Jae JI Jung, Hanyang University, Korea

Daeyoung Kim, Information and Communications University, Korea
Hyoung Joong Kim, Korea University, Korea
Seong-Dong Kim, KETI, Korea
Ho Yeol Kwon, Dangwon National University, Korea
Yong Joon Lee, ETRI, Korea
Petri Mähönen, Aachen University, Germany
Cecilia Mascolo, University College London, UK
Young Shik Moon, Hanyang University, Korea
Jae In Oh, Dankook University, Korea
Shingo Ohmori, NICT, Japan
Neeli Prasad, Aalborg University, Denmark
Kasim Rehman, Cambridge Systems Associates, UK
Tapani Ryhänen, Nokia Research Center, Finland
Alberto Sanna, Scientific Institute H San Raffaele, Italy
Ichiro Satoh, National Institute of Informatics, Japan
Hideyuki Tokuda, Keio University, Japan
Pim Tuyls, Philips Research, Netherlands
Pablo Vidales, Deutsche Telekom Laboratories, Germany
Dirk Westhoff, NEC Laboratories, Germany

Table of Contents

Mobile, Wireless, and Ad Hoc Networking

Smart Sensors and Sensor Networks

Video Coding Techniques for Ubiquitous Multimedia Services

Yo-Sung Ho and Seung-Hwan Kim

Gwangju Institute of Science and Technology (GIST)
1 Oryong-dong, Buk-gu, Gwangju, 500-712, Korea
{hoyo, kshkim}@gist.ac.kr

Abstract. Emerging ubiquitous multimedia services are expected to be available anytime, anywhere, and using different computing devices. Video compression is necessary for transmission of digital video over today's band-limited networks, or for storage constrained applications. This paper gives a short overview over previous video coding standards and analyzes in more detail H.264, which is the latest international video coding standard. Since scalable video coding (SVC) provides the capability of reconstructing lower resolution or lower quality signals from partial bitstream, it is a good paradigm to the streaming video application for ubiquitous multimedia service. Hence, we also discuss several coding techniques and frameworks for SVC including fine granular scalability (FGS).

Keywords: Video coding standard, ubiquitous, H.264, scalable video coding, fine granular scalability.

1 Introduction

The deployment of multimedia services such as audio/video-on-demand, digital library, remote camera surveillance, and distributed visual tracking is becoming ubiquitous. Since limited transmission bandwidth or storage capacity stresses the demand for higher video compression ratios, video compression has been a critical component of many multimedia applications available today. Meanwhile, international study groups, VCEG (Video Coding Experts Group) of ITU-T (International Telecommunication Union - Telecommunication sector) and MPEG (Moving Picture Experts Group) of ISO/IEC, have researched the video coding techniques for various applications of moving pictures since the early 1990s.

ITU-T developed H.261 as the first video coding standard for videoconferencing application. H.261 [1] supports video telephony and videoconferencing over ISDN circuit-switched networks. These networks operate at multiples of 64 kbps and the standard was designed to offer computationally simple video coding for these bitrate. The coding algorithm is a hybrid of transform coding and inter-picture prediction with an integer-accuracy motion compensation.The block-based inter-picture prediction is used to removes temporal redundancy between consecutive frames. If the time domain data is smooth with little variation then frequency data will make low frequency data larger

F. Stajano et al. (Eds.): ICUCT 2006, LNCS 4412, pp. 1–10, 2007.

and high frequency data smaller. Hence, discrete cosine transform (DCT) is used to convert data in time domain to data in frequency domain. In order to remove any further statistical redundancy in the motion data and transformed coefficients, variable length coding (VLC) is used.

The first MPEG standard, MPEG-1 video [2] was developed for the specific application of video storage and playback on Compact Disks. MPEG-1 video was conceived to support the video CD, a format for consumer storage and playback that was intended to compete with VHS videocassettes. The standard uses block-based motion compensation, DCT and quantization and is optimized for a compressed video bitrate of around 1.2 Mbps. MPEG-1 video is still widely used for PC and web-based storage of compressed video files.

Following on from MPEG-1 video, MPEG-2 video [3] (ITU-T adopted it as H.262) standard aimed to support a large potential market, digital broadcasting of compressed television. MPEG-2 video was a great success, with world wide adoption for digital TV broad-casting via cable, satellite and terrestrial channels. For several years, MPEG-2 video has been improved, but it is reaching its theoretical limitations. Additional improvements were attempted, using other techniques, such as fractals and wavelets, with no significant improvement in video results. The original MPEG-4 Visual[4] standard attempted to bring the object-oriented perception into the compression world, with limited success, due to its complexity and overhead. In order to cover the very wide range of applications such as shaped regions of video objects as well as rectangular pictures, MPEG-4 Visual [4] standard was developed. This includes also natural and synthetic video/audio combinations with interactivity built in.

In an attempt to improve on the compression performance of H.261, the ITU-T working group developed H.263 [5]. This provides better compression than H.261, supporting basic video quality at bitrate of below 30 kbps, and is part of a suite of standards designed to operate over a wide range of circuit-switched and packet-switched networks. The coding algorithm used in H.263 is similar to that used by H.261, however with some improvements and changes to improve performance and error recovery. Half pixel based motion compensation technique is used and some parts of the hierarchical structure of the data stream is provided optionally. There are now four negotiable options included to improve performance: Unrestricted Motion Vectors, Syntax-based arithmetic coding, Advance prediction, and forward and back-ward frame prediction. After finalizing the original H.263 standard for video telephony in 1995, the ITU-T Video Coding Experts Group (VCEG) started working on a long-term effort to develop a new standard for low bitrate visual communications. This effort leads to the H.26L standard draft, offering significantly better video compression efficiency than previous standards [6].

The organization of the paper is as follows. We first explain several key features of H.264 in Section 2 and present the basic coding structure for MPEG-4 FGS and the scalable extension of H.264 (JSVM) which is the newest SVC standard in Section 3. In Section IV, we show two kinds of experimental results: One is related to the comparison of coding efficiency between H.264 and MPEG-4 visual and the other one is related to between MPEG-FGS and JSVM. Section V draws conclusions and summarizes future perspectives of video coding techniques for ubiquitous multimedia services.

2 Overview of H.264/AVC

H.264 video coding standard has been developed to satisfy the requirements of applications for various purposes, better picture quality, higher coding efficiency, and more error robustness. In this Section, we describe an overview of H.264.

2.1 Profiles and Levels

A Profile specifies a subset of entire bitstream of syntax and limits that shall be supported by all decoders conforming to corresponding Profile. There are three Profiles in the first version: Baseline, Main, and Extended. Baseline Profile is to be applicable to real-time conversational services such as video conferencing and videophone. Main Profile is designed for digital storage media and television broadcasting. Extended Profile is aimed at multimedia services over Internet. Also there are four High Profiles defined in the fidelity range extensions [7] for applications such as content-contribution, content-distribution, and studio editing and post-processing : High, High 10, High 4:2:2, and High 4:4:4. High Profile is to support the 8-bit video with 4:2:0 sampling for applications using high resolution. High 10 Profile is to support the 4:2:0 sampling with up to 10 bits of representation accuracy per sample. High 4:2:2 Profile is to support up to 4:2:2 chroma sampling and up to 10 bits per sample. High 4:4:4 Profile is to support up to 4:4:4 chroma sampling, up to 12 bits per sample, and integer residual color transform for coding RGB signal.

The Profiles have both the common coding parts and as well specific coding parts as shown in Fig. 1. For any given Profile, Levels generally correspond to processing power and memory capability of a codec. Each Level may support a different picture size - QCIF, CIF, ITU-R 601 (SDTV), HDTV, S-HDTV, D-Cinema [7]. Also each Level sets the limits for data bitrate, frame size, picture buffer size, etc [7].

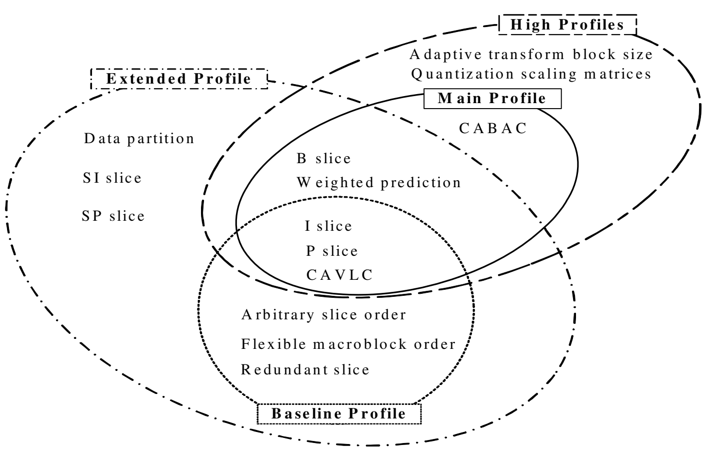

Fig. 1. Profiles and coding tools in H.264/AVC

2.2 Video Coding Algorithm

H.264 improves the rate distortion performance by exploiting advanced video coding technologies, such as variable block size motion estimation, multiple reference prediction, spatial prediction in intra coding, context based variable length coding (CAVLC) and context-based adaptive binary arithmetic coding (CABAC). The testing results of H.264/AVC show that it significantly outperforms existing video coding standards in both peak signal-to-noise ratio (PSNR) and visual quality [6].

For encoding a block or macroblock in intra-coded mode, H.264 predicts a block based on previously reconstructed blocks. The residual signal between the current block and the prediction is finally encoded. For the luminance samples, the prediction block may be formed for each 4x4 block, each 8x8 block, or for a 16x16 macroblock. One case is selected from a total of 9 prediction modes for each 4x4 and 8x8 luminance blocks; four modes for a 16x16 luminance block; and four modes for each chroma blocks.

Inter prediction is to reduce the temporal correlation with help of motion estimation and compensation. In H.264, the current picture can be partitioned into block sizes up to 4x4. For 16x16macroblock mode, there are four cases: 16x16, 16x8, 8x16 or 8x8, also four cases: 8x8, 8x4, 4x8 or 4x4 for 8x8 mode. Hence, the inter prediction process can form segmentations for motion representation as small as 4x4 block in size, using motion vector accuracy of one-quarter of the sample. Sub-pel motion compensation can provide significantly better compression performance than integer-pel compensation [7]. The process for inter prediction also involve the selection of the pictures to be used as the reference pictures from a number of stored previously-decoded pictures.

After inter prediction or intra prediction, the resulting prediction residual in a macroblock is split into small blocks according to the size of transform. H.264 uses also an adaptive transform block size, 4x4 and 8x8 (High Profiles only). In general transform and quantization require several multiplications resulting in high complexity for implementation. So, for simple implementation, the exact transform process is modified to avoid the multiplications. Then the transform and quantization are combined by the modified integer forward transform, quantization, scaling. For improved compression efficiency, H.264 also employs a hierarchical transform structure, in which the DC coefficients of neighboring 4x4 transforms for the luminance signals are grouped into 4x4 blocks and transformed again by the Hadamard transform. In order to utilize correlation among transform DC coefficients of neighboring blocks, the standard specifies the 4x4 Hadamard transform for luminance DC coefficients for 16x16 Intra-mode only, and 2x2 Hadamard transform for chroma DC coefficients.

Unlike fixed tables of variable length codes used in previous standards such as MPEG-1, 2, 4, H.261, H.262 and H.263, H.264 uses different VLCs in order to match a symbol to a code based on the context characteristics. In Baseline profile, all syntax elements except for the residual data are encoded by the Exp-Golomb codes and residual data is coded with more sophisticated entropy coding method called context-based adaptive variable length coding (CAVLC). In Main and High profiles, context-based adaptive binary arithmetic coding (CABAC) is can be used for all syntax elements including residual data. CABAC has more coding efficiency but higher complexity compared to CAVLC.

H.264 may suffer from blocking artifacts due to block-based transform in intra and inter prediction coding, and the quantization of the transform coefficients. The deblocking filter reduces the blocking artifacts in the block boundary and prevents the propagation of accumulated coded noise. H.261 has selectively suggested similar deblocking filter which was beneficial to reduce the temporal propagation of coded noise. However, MPEG-1, 2 did not use the deblocking filter because of high implementation complexity. However, H.264 uses the deblocking filter for higher coding performance in spite of implementation complexity. Filtering is applied to horizontal or vertical edges of 4 x 4 blocks in a macroblock. The luminance deblocking filter process is performed on four 16-sample edges and the deblocking filter process for each chroma components is performed on two 8-sample edges.

Table 1. Comparison of standards MPEG-2, MPEG-4 Visual and H.264

Feature	MPEG-2	MPEG-4 part 2	H.264/AVC
ME block size	8×8	16×16, 16×8, 8×8	16×16, 8×16, 16×8, 8×8, 4×8, 8×4, 4×4
Intra prediction	No	Transform Domain	Spatial Domain
Transform	8×8 DCT	8×8 DCT	8×8, 4×4 integer DCT 4×4, 2×2 Hadamard
Entropy coding	VLC	VLC	VLC, CAVLC, CABAC
Fractional ME	½-pel	¼-pel	¼ -pel
Reference picture	One	One	Multiple
In loop Deblocking filter	No	No	Yes
Picture types	I, P, B	I, P, B	I, P, B, SI, SP
Profiles	5 profiles	8 profiles	7 profiles
Transmission rate	2-15Mbps	64kbps - 2Mbps	64kbps - 150Mbps
Complexity	Medium	Medium	High

3 Scalable Video Coding

In ubiquitous environment, many challenges rise from the heterogeneity in multimedia client and server capabilities, and their end-to-end resource availabilities. For example, clients of a multimedia service may range from supercomputers to commodity PCs and smart handheld devices such as palm-tops. The network connections between the server and clients may range from high speed LANs to low speed dial-ups, from wire-line to wireless. Furthermore (and less addressed), even for clients with the same machine type and connection type, the amounts of resources available to each of them may still vary, depending on their location, workload, and the time they make service requests. In particular, the bottleneck resource in each client's resource requirement may be different. Therefore, to deal with the heterogeneity problem, any solution that only targets one specific type of bottleneck resource (for example, the network) may not be effective in all situations. Since scalable video coding (SVC) provides the capability of reconstructing lower resolution or lower quality signals from partial bitstream, it is a good paradigm to the streaming video application for ubiquitous multimedia service.

Early video compression standards such as ITU-T H.261 [1] and ISO/IEC MPEG-1 [2] did not provide any scalability mechanisms. MPEG-2 was the first standard to include implementations of layered coding, where the standalone availability of enhancement information (without the base layer) is useless, because differential encoding is performed with reference to the base layer. All dimensions of scalability as mentioned above are supported (spatial, temporal, SNR); however, the number of scalable bitstream layers is generally restricted to a maximum of three in any of the existing MPEG-2 profiles. The video codec of the ISO/IEC MPEG-4 standard [5] provides even more flexible scalable profile called MPEG-4 FGS, including spatial and temporal scalability within a more generic framework, but also SNR scalability with fine granularity and scalability.

3.1 MPEG-4 FGS

As shown in Fig. 2, the basic information of the input signal is coded in the same way as the traditional block-based coding method in the base layer. In the enhancement layer, the residual signal that is not coded in the base layer is divided into 8x8 blocks and each block is DCT transformed. All the 64 DCT coefficients in each block are bitplane coded using four VLC tables [8].

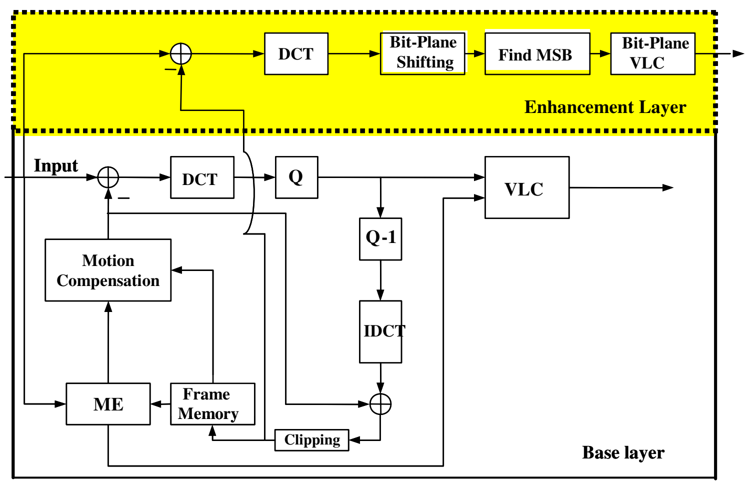

Fig. 2. MPEG-4 FGS encoder

There are many advantages of using FGS for Internet streaming video applications: it allows separation of encoding and transmission, the server can transmit enhancement layer at any bit rate without transcoding, it enables video broadcast on the Internet to reach a large audience, and it provides a solution to the video server overload problem.

However, compared with nonscalable coding, which is the upper bound for any scalable coding techniques, FGS is about 2-dB worse at the high end of the bitrate range.

3.2 Joint Scalable Video Model

In order to support fine granular SNR scalability, JSVM adopted progressive refinement (PR) slices [9]. Each PR slice is regarded as a FGS layer and coded with cyclical block coding as depicted in Fig. 3.

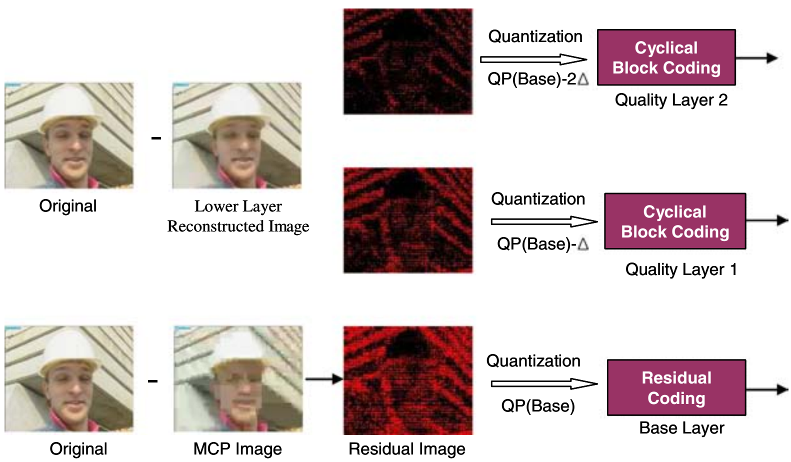

Fig. 3. FGS coding structure in JSVM

In the cyclical block coding scheme, the coding is basically partitioned into two passes, the significant and refinement passes. The significant pass first encodes the insignificant coefficients that have values of zero in the subordinate layers. Then, the refinement pass refines the remaining significant coefficients with range from -1 to +1. During the significance pass, the transform blocks are coded in a cyclical and block interleaved manner. On the other hand, the coding of the refinement pass is conducted in a suband-by-subband fashion [9] [10] [11].

In cyclical block coding, for each cycle, the coding of a block is continued until a non-zero coefficient in zigzag order is coded. Particularly, the coding of each cycle in a block includes an EOB symbol, a Run index and a non-zero quantization level. The EOB symbol is coded prior to the other symbols for signaling whether there are nonzero coefficients to be coded in a cycle. In addition, the Run index, represented by several significance bits, is used for recording the location of a non-zero coefficient. To further reduce the bit rate, each symbol is coded by a context-adaptive binary arithmetic coder [12] [13].

In Fig. 3, each FGS layer is represented as a group of multiple bit-planes. However, these bit-planes are coded by a cyclic block coding instead of traditional bit-plane coding used in MPEG-4 FGS. The coding order of transform coefficient levels has been modified. Instead of scanning the transform coefficients macroblock by macroblock as it is done in the ”normal” slices, transform coefficient blocks are scanned in several paths, and in each path only a few coding symbols for a transform coefficient block are coded. Therefore, quality of the SNR base layer can be improved in a fine granular way. With the exception of the modified coding order, the CABAC entropy coding is reused, as specified in H.264/MPEG4-AVC [7].

4 Experimental Results

In the experiment, we first compare the coding efficiency between H.264 and MPEG-4 Visual. The basic test conditions for H.264 are set as follows:

1) MV search range is 16 pixels for CIF
2) RD optimization is enabled
3) Reference frame number equals to 1
4) GOP structure is IPPPP

Fig. 4 shows some comparisons of the coding efficiency of MPEG-4 Visual ASP and H.264 for the FOREMAN test sequence of CIF (352288) format. In these simulations, no rate control was used and Rate-Distortion (R-D) curves corresponding to encoding with different standards are presented. These are example plots and the results will vary from one encoder to another and from one test video sequence to another. From these plots we see that H.264 baseline profile provides about 2dB higher PSNR value over MPEG-4 Advanced Simple Profile.

In the second experiment, we have tested the coding efficiency of JSVM (version 5.2) according to the size of GOP. For GOP size of 2, we obtain the well-known prediction

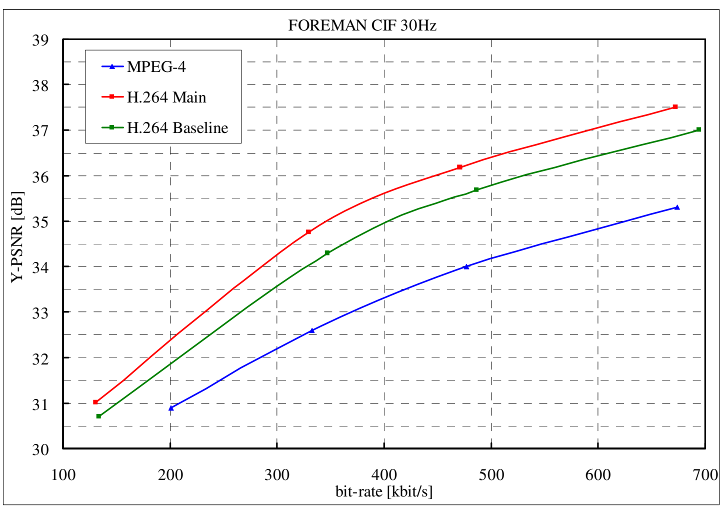

Fig. 4. Comparison of coding efficiency between MPEG-4 ASP and H.264

structure (IBPBPB) where one B-frame is encoded between two P or, alternatively, I-frames. For GOP size of 4, the coding structure of picture type is as follows: I B B B P B B B P. Fig. 5 shows some comparisons of the coding efficiency for GOP size of one, two, four, and eight. From Fig.5, we can know that the larger size of GOP provides better coding efficiency.

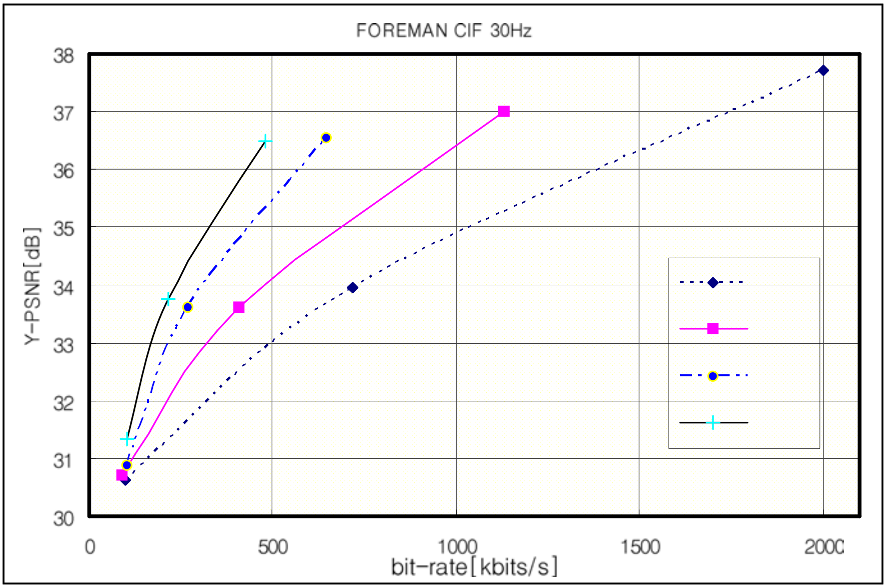

Fig. 5. Comparison of coding efficiency of JSVM according to GOP size

5 Conclusions

This paper gives a short overview over previous video coding standards. The new video standard known as H.264/AVC presents a rich collection of state-of-the-art video coding techniques and it can provide interoperable video broadcast or communication with degrees of capability that far surpass those of prior standards. Since the scalable extension of H.264 (JSVM) also provides fine granular scalable functionality and good coding efficiency, it is also a good paradigm to the streaming video application. Therefore, we believe these video coding technologies provide a powerful impact on ubiquitous multimedia services in the years to come.

Acknowledgements

This research was supported by MIC, Korea, under the ITRC support program supervised by IITA (IITA-2005-C1090-0502-0022) through the Realistic Broadcasting Research Center (RBRC) at the Gwangju Institute of Science and Technology (GIST).