Journal of Electronic Imaging 11(3), 375–380 (July 2002).

Conformance test of simple profile MPEG-4 texture decoding

Woo-Shik Kim
In-Su Park
Sogang University
Department of Electronic Engineering
C.P.O. Box 1142
Seoul 100-611
South Korea

Su-Hyun Park
Ewha Womans University
Department of Information Electronics
Seoul 120-750
South Korea

Byung-Uk Lee
Ewha Womans University
Department of Information Electronics
Seoul 120-750
South Korea
E-mail: bulee@ewha.ac.kr

Rae-Hong Park
Sogang University
Department of Electronic Engineering
C.P.O. Box 1142
Seoul 100-611
South Korea

Abstract. *Moving picture experts group (MPEG)-4 visual conformance standard specifies methods to verify whether bitstreams and decoders meet the requirements at a specified profile and level. The test of decoders can be divided into two parts: static and dynamic tests. The static test can be performed by comparing images from a decoder under test with those from a reference decoder using various test bitstreams. This paper proposes design methodologies of test bitstreams for a simple profile MPEG-4 texture decoder, and shows experimental results on the discrete cosine transform (DCT) scan type, DC/AC coefficient prediction, inverse quantization, inverse DCT, and various macroblock-type verifications.* © 2002 SPIE and IS&T. [DOI: 10.1117/1.1477443]

1 Introduction

With the establishment of the MPEG-4 standard, many changes are expected in related multimedia applications. MPEG-4 provides a means of mobile multimedia communications at very low bitrate with error robustness. Moreover, the object oriented coding in MPEG-4 enables user interaction, which will bring new applications in digital broadcasting and multimedia.[1]

For mass production of MPEG-4 decoders, it must be assured that they are free from malfunctions. Conformance tests[2] verify whether MPEG-4 decoders are compliant with the MPEG-4 standard.[3] These tests consist of static and dynamic tests; the former verifies bitstream parsing and functionality, while the latter is concerned about timing such as buffering verification and decoding order.[2,4,5]

Since MPEG-4 integrates many tools, the conformance test is partitioned into several parts. For static test of the visual part of the MPEG-4, several types of test bitstreams, such as general, shape coding,[6] and error resilience are employed to verify the decoder under test. The general test bitstreams cover texture coding, motion compensation, variable length coding. Each test bitstream is designed according to a specific profile and level. We confined our test scope to texture decoding procedure of simple and core profiles. The texture decoding procedure is shown in the block diagram of Fig. 1.[7] Conformance test for shape coding can be found in other literature.[6]

Paper JEI 20078 received Oct. 30, 2000; revised manuscript received Jan. 24, 2002; accepted for publication Feb. 25, 2002.
1017-9909/2002/$15.00 © 2002 SPIE and IS&T.

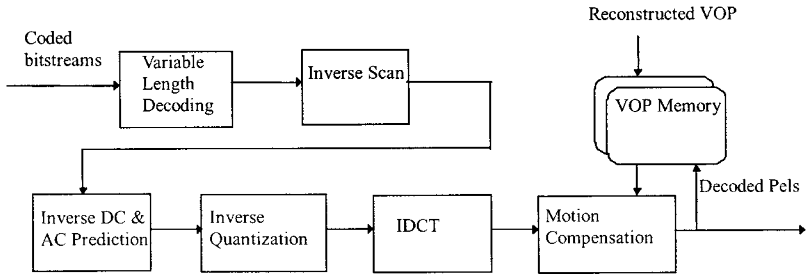

Fig. 1 MPEG-4 video texture decoding process.

We modified a reference software encoder[8] to create test bitstreams for texture decoding and macroblock (MB)-type verification, which is a basic coding unit of an MPEG video bitstream. Internal encoding parameters were designed, and special purpose test images were created for the test bitstreams. Additionally, we developed a bitstream analyzer to extract encoding parameters and data interactively. Conformance test results are determined by comparing the reconstructed images of the reference decoder and the output of a decoder under test. Any deviation from the standard will result in discrepancies of the output images. To guarantee the conformance, it is important to cover all possible cases of the encoding schemes. We will present the design principles for the test bitstreams.

2 Inverse Scan Test

The inverse scan constructs a two-dimensional array of 8 $\times$8 discrete cosine transform (DCT) coefficients from a one-dimensional sequence. The scan direction depends on several variables such as *alternate_scan_flag* and *AC_pred_flag*.

There are three scan methods: zigzag scan, alternate vertical scan, and alternate horizontal scan. The choice of a scan method depends on AC prediction. If AC prediction is not used, the zigzag scan is used. Otherwise, alternate vertical or horizontal scan is chosen depending on the direction of the ac prediction. In the interlaced mode, the alternate vertical scan can be chosen regardless of AC prediction by setting the *alternate_scan_flag*.

2.1 *Zigzag Scan Test*

We disabled the *AC_pred_flag* to generate a zigzag test bitstream. We decoded the generated test bitstream using a reference decoder and abnormal decoders that can scan with only a fixed scan method.

Figure 2(a) shows the zigzag scan test image decoded by a reference decoder. Figure 2(b) shows an image decoded by zigzag scan, which confirms that all the blocks are scanned in zigzag scanning order. Figures 2(c) and 2(d) show erroneous images from a decoder with wrong scanning directions.

2.2 *Alternate Scan Test*

When the *AC_pred_flag* is enabled, DCT coefficients are alternately scanned. The alternate vertical or horizontal scan is determined by the direction of AC prediction, which depends on the direction of DC prediction. We designed test images, with DC coefficients of each block increasing monotonically either from top to bottom or from left to right. In other words, the blocks of the test images are becoming brighter as they go downward or to the right side. A reference decoder detects the DC prediction direction and then the AC prediction direction, which will determine the direction of the alternate scan. Figure 3 shows alternate vertical scan test images. Note that the bottom of the image is brighter than the top, which results in vertical DC prediction. Figure 3(a) shows the image from a reference decoder with alternate vertical scan. Figures 3(b)–3(d) are images from erroneous software decoders with fixed scanning directions.

Figure 4(a) shows an alternate horizontal scan test bitstream decoded by a reference decoder. Note that the right side of the image is brighter than the left side. Figures 4(b)–4(d) are images obtained from hypothetical decoders with fixed scanning directions.

3 DC/AC Prediction Test

In an encoder, DCT coefficients are compressed by predictive coding. The prediction direction depends on the horizontal and vertical DC gradients between adjacent blocks. If the horizontal DC gradient is less than that of the vertical direction, horizontal prediction is chosen. AC prediction is performed when *AC_pred_flag* is set to one. The direction of AC prediction is the same as that of the DC prediction.

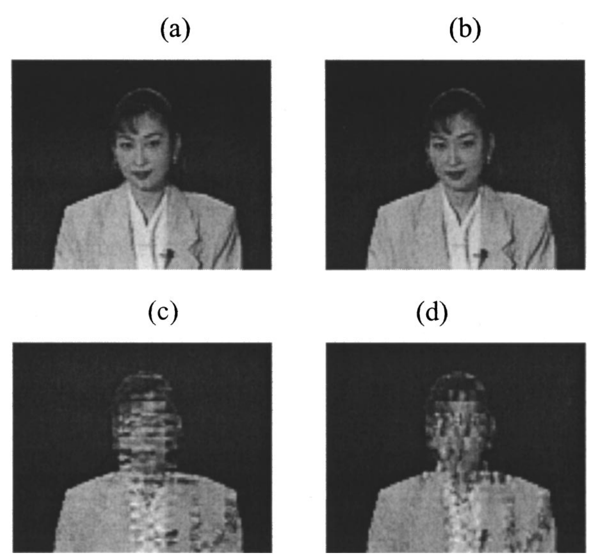

Fig. 2 Zigzag scan when AC prediction is disabled. (a) Reference decoder, (b) zigzag scan, (c) alternate vertical scan, (d) alternate horizontal scan.

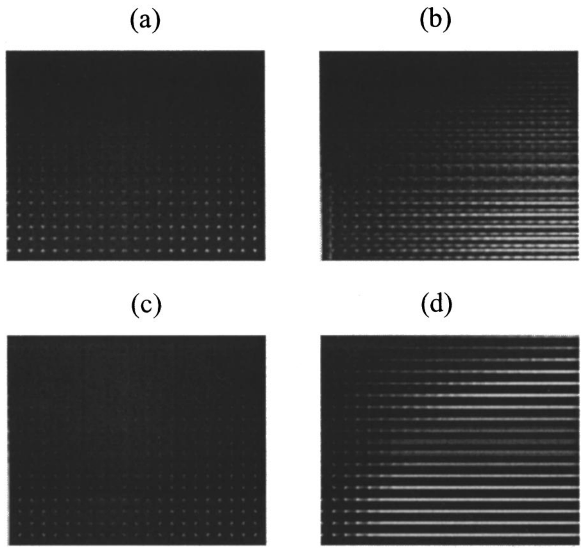

Fig. 3 Alternate vertical scan test. (a) Reference decoder, (b) zigzag scan, (c) alternate vertical scan, (d) alternate horizontal scan.

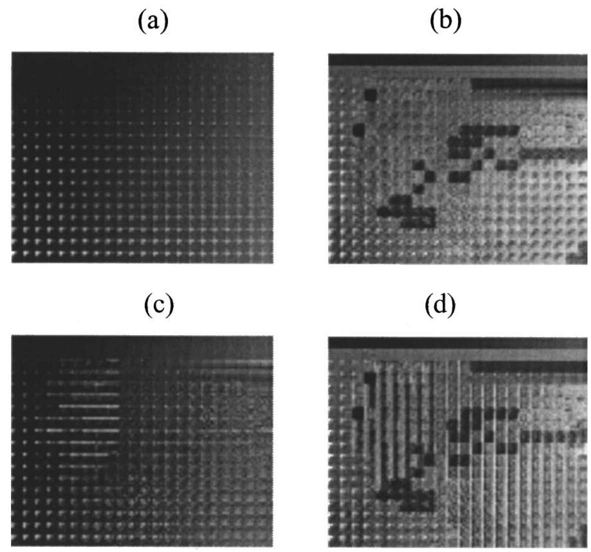

Fig. 5 DC/AC prediction test. (a) Input frame, (b) incorrect DC prediction, (c) incorrect AC prediction, (d) incorrect DC/AC prediction.

Coefficients from the first row or the first column of the neighboring block are used for prediction of the current block depending on the prediction direction.

A test bitstream was designed to verify whether a decoder performs DC/AC prediction correctly, and to detect any error by comparing the decoded images. Test patterns for DCT coefficients were designed with specific DC and AC values. Figure 5(a) shows one of the test patterns, where the DC coefficients increased from left to right and from top to bottom, while they had the identical AC coefficients in the same row. We could verify the same patterns from the reference decoder. Figures 5(b)–5(d) show images obtained from abnormal decoders with fixed prediction direction. Figure 5(b) is a decoded image when the DC prediction direction was set incorrectly to the vertical direction, Fig. 5(c) shows a result when the AC prediction direction was set incorrectly to the horizontal direction, and Fig. 5(d) is a case when both DC and AC prediction directions were wrong. By comparing the images of Figs. 5(b) and 5(c), we observed that DC prediction errors were more prominent than AC prediction errors, since human eyes are more sensitive to low spatial frequency components.

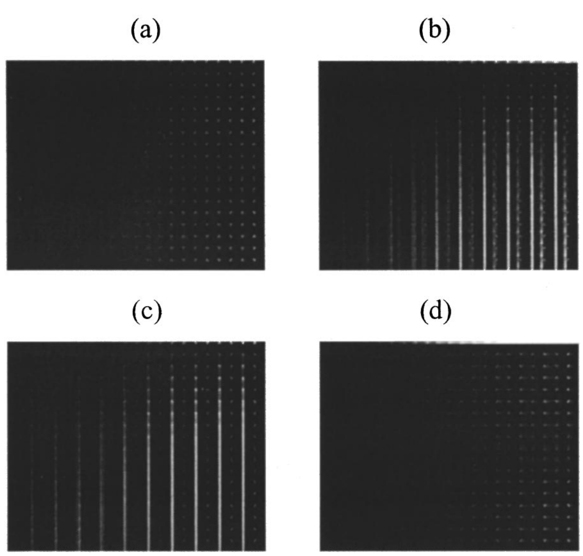

Fig. 4 Alternate horizontal scan test. (a) Reference decoder, (b) zigzag scan, (c) alternate vertical scan, (d) alternate horizontal scan.

4 Inverse Quantization Test

Quantization reduces the amount of bits to represent DCT coefficients by dividing them with quantization parameters. MPEG-4 adopts two different quantization methods: MPEG-1 and H.263. In the H.263 quantization method, coefficients are quantized using the same quantization parameter for all AC coefficients, while the different values are used for each AC coefficient in the MPEG-1 method.

In a decoder, an 8×8 array of quantized DCT coefficients, $QF[u][v]$, is inverse quantized, resulting in the reconstructed DCT coefficients, $F[u][v]$. The weighting matrix dictates the step size for each DCT component, and a scale factor changes the whole matrix. There are two inverse quantization methods, which are determined by *quant_type*. The first method is used when *quant_type* is set to 1. It uses both the weighting matrix and the scale factor. Otherwise, only the scale factor is applied and the weighting matrix is not used.

4.1 *Intra DC Coefficients*

In both quantization methods, inverse quantization of DC coefficients is different from that of AC coefficients. DC coefficients are reconstructed as

$$F[0][0]=dc\_scaler\times QF[0][0].$$

To verify the inverse quantization, we designed a test bitstream using 31 frames of Akiyo sequence. Each frame was encoded with a fixed *quantizer_scale* value from 1 to 31, respectively, which encompasses the whole range. If a decoder has a defect in determining *dc_scaler*, an error will

Fig. 6 Inverse quantization test. (a) $QP=1$, $dc\_scaler=2$, (b) $QP=29$, $dc\_scaler=58$.

be detected in images from the decoder under test. Two images are shown in Fig. 6 among 31 decoded images from a hypothetical decoder, in which the *dc_scaler* values were miscalculated as $2\times quantizer\_scale$. When *quantizer_scale* was 1 and 29, the used *dc_scaler* was 2 and 58 at the hypothetical decoder, instead of 8 and 23, respectively. In Fig. 6(a), a decoded image from the hypothetical decoder had smaller *dc_scaler* value than the original one, so it became obscure, and in Fig. 6(b), a decoded image with larger *dc_scaler* value than the original one shows an exaggerated blocking effect.

4.2 *Intra AC and Inter DC/AC Coefficients*

To generate test bitstreams for inverse quantization for intra AC and inter DC/AC coefficients, each element of the inverse quantization matrix was emphasized. Each MB has only one dominant coefficient to assess whether the weighting matrix in the decoder is the same as the standard matrix and the scan order is correct. Figure 7 shows differences between P-frame image from a reference decoder and that from a hypothetical decoder that used the intra matrix instead of inter matrix.

4.3 *Saturation Test*

The saturation process imposes a limit to the inverse quantized DCT coefficients within a fixed range of [− 2048–2047] for the intra MB, and [−127–127] for the inter MB.

To conform the saturation process of an intra MB, we designed a bitstream with DCT values of −2050, −2048, −2040, 2040, 2047, and 2050. Two values, −2050 and 2050, exceeding the saturation range and two threshold values, −2048 and 2047, were included to verify the correctness near the threshold. The test was performed by comparing the decoded images of the reference decoder with that of a decoder under test.

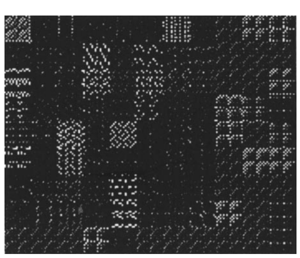

Fig. 7 Difference picture between the output of the reference decoder and that of the hypothetical decoder for a P frame.

4.4 *Mismatch Control Test*

A mismatch control is used only for the first inverse quantization method. It removes errors in decoded images, since encoders and decoders may have various implementations of DCT with different precisions. Small nonzero input values to the IDCT process may result in zero output values for compliant inverse discrete cosine transform (IDCT). In that case, mismatches may occur among compliant decoders with different precision IDCT implementations. Moreover, large and visible mismatches may accumulate between the encoder and decoder without the mismatch control.

To generate a test bitstream for mismatch control, the following pseudo C codes were added:

$$if(Block\_No==0)\{sum=odd;\ F'[7][7]=odd;\}$$

$$else\ \ if(Block\_No==1)\{sum=odd;\ F'[7][7]=even;\}$$

$$else\ \ if(Block\_No==2)\{sum=even;\ F'[7][7]=even;\}$$

$$else\ \ if(Block\_No==3)\{sum=even;\ F'[7][7]=odd;\}$$

Therefore, it is possible to verify that a test decoder performs the mismatch control process correctly by comparing the output of the normal decoder and that of the test decoder.

5 IDCT Test

IDCT reconstructs pixel values from DCT coefficients. The IDCT function used in the decoding process generates a value in the range of [−256, 255], and the range becomes [−384, 383] when input blocks of DCT coefficients use all the 64 values in a block. We designed a test bitstream with all possible values of IDCT in the range of [−384–383]. We generate all possible values of DCT coefficients. By comparing the output of the reference decoder with that of a test decoder, we can assess the conformance of the test decoder.

6 Macroblock-Type Decision Test

The MB is a coding unit for combined motion, shape, and texture coding for luminance and chrominance blocks. For example, in 4:2:0 chrominance format, a MB has four luminance blocks and two chrominance blocks. The MB layer has data for motion vector difference, difference quantizer, AC prediction, DCT coefficients of each block, and MB type.

The information contained in the MB layer varies according to the MB type. It is determined by video object plane (VOP) type, namely, I- and P-VOP, where I denotes intra, P represents inter. For I-VOP, all the MBs have intra mode, therefore, they do not have motion vectors. There are two intra MB types depending on the usage of difference quantization, which changes the quantization parameter for rate control.

Table 1 Decoded macroblock types.

VOP type	MB type	Name
P	Not_coded	···
P	0	INTER
P	1	INTER+Q
P	2	INTER4V
P	3	INTRA
P	4	INTRA+Q
P	Stuffing	···

In P-VOP, MBs can be either inter or intra mode, and the usage of difference quantization affects the MB type. When the 8×8 prediction mode is used, the MB type becomes four motion vector mode. When not_coded type is sent, the MB from the previous frame at the same position is duplicated.

Table 1 shows all types of MBs for the P-VOP. Stuffing is bit insertion, which is employed to increase the bitrate in order to prevent underflow at the decoder. A decoder simply discards the stuffing bits.

In the following subsections, a procedure to determine the MB type is briefly explained and simulation results are illustrated. The test bitstreams are generated by changing MB types. We used 176×144 quarter common intermediate format Akiyo test sequence with 4:2:0 chrominance format, consisting of 11×9=99 MBs. The test sequence is coded as P-VOP with binary shape in progressive frame mode. Various combination of all MB types are tested.

6.1 *MB Type Not_Coded*

MB type not_coded can be used in P-VOP only. When the variable named COD is set to 1, no further process is required and all the information of the MB is assumed to be the same as that of the MB at the same position in the previous frame.

We generated the MB not_coded test bitstream by modifying a software encoder. Figure 8(a) shows the output of the bitstream analyzer, which verifies that all MBs in the bitstream are coded as not_coded. In the output of the bitstream analyzer, there are only 9×7=72 MBs, because the object defined in shape coding is limited by a bounding rectangle. The MB outside the shape mask is noted as "T," which stands for transparent, and not coded MB is represented as "N."

6.2 *MB-Types 0, 1, and 2: INTER, INTER+Q, and INTER4V*

For motion compensation in P-VOP, there are two types of motion vectors: 16×16 and 8×8 motion vectors, where the number indicates the block size for the motion compensation. For 8×8 motion vector mode, each MB has four motion vectors, which improves the accuracy of motion compensation. The MB of type 0 uses 8×8 motion vectors. With 16×16 motion vectors mode, the MB type is either 1 or 2 depending on the usage of difference quantization. Figure 8(b) shows a result when MB type is 0, whereas Fig. 8(c) is the result when MB type is 2.

(a)

T	T	T	N	N	N	T	T	T
T	T	T	N	N	N	T	T	T
T	T	T	N	N	N	N	T	T
T	T	T	N	N	N	N	T	T
T	T	N	N	N	N	N	N	T
T	N	N	N	N	N	N	N	N
N	N	N	N	N	N	N	N	N

(b)

T	T	T	0	0	0	T	T	T
T	T	T	0	0	0	T	T	T
T	T	T	0	0	0	0	T	T
T	T	T	0	0	0	0	T	T
T	T	0	0	0	0	0	0	T
T	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

(c)

T	T	T	2	2	2	T	T	T
T	T	T	2	2	2	T	T	T
T	T	T	2	2	2	2	T	T
T	T	T	2	2	2	2	T	T
T	T	2	2	2	2	2	2	T
T	2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2	2

(d)

T	T	T	4	4	4	T	T	T
T	T	T	4	4	4	T	T	T
T	T	T	4	4	4	4	T	T
T	T	T	4	4	4	4	T	T
T	T	4	4	4	4	4	4	T
T	4	4	4	4	4	4	4	4
4	4	4	4	4	4	4	4	4

Fig. 8 Macroblock-type test (a) MB type not_coded, (b) MB-type 0 (INTER), (c) MB-type 2 (INTER4V), (d) MB-type 4 (INTRA+Q).

6.3 *MB-Types 3 and 4: INTRA and INTRA+Q*

For I-VOP all MBs are coded in intra mode with MB type equal to 3, INTRA, or 4, INTRA+Q, depending on the usage of difference quantization. In P-VOP, when a MB has large motion compensation error, the MB is coded as an intra block with MB type equal to 3 or 4. The test bitstream is generated with P-VOP in which the MB type is all set to 3 or 4. Figure 8(d) shows the result in the case of MB type 4.

7 Conclusion

This paper presents design methodologies of MPEG-4 texture decoding conformance tests and experimental results on inverse scan, DC/AC prediction, inverse quantization, IDCT, and MB types. We explain design principles of the test bitstreams and compare the output of the reference decoder with the images decoded by abnormal decoders. Test patterns are designed to include all possible cases specified in the MPEG-4 standard. The test bitstream is verified using the bitstream analyzer, which can verify the desired information in the bitstream. This test can be easily applied during the R&D and mass production of the decoders. Future work includes the dynamic test, which is related to rate control, buffer verification, memory bandwidth, and automatic detection of the fault.

Acknowledgments

This work is supported by Hyundai Electronics Industries Co., and in part by the Brain Korea 21 Project.

References

1. T. Ebrahimi and C. Horne, ‘‘MPEG-4 natural video coding—An overview,’’ *Signal Process. Image Commun.* **15**(4–5), 365–385 (2000).
2. ISO/IEC 14496-4, *Information Technology—Very Low Bitrate Audio-Visual Coding: Conformance (Committee Draft)* (1998).
3. ISO/IEC JTC1/SC29/WG11 N3156, *Overview of the MPEG-4 Standard* (1999).
4. C. M. Kim, B. U. Lee, and R.-H. Park, ‘‘Design of MPEG-2 video test bitstreams,’’ *IEEE Trans. Consum. Electron.* **45**(4), 1213–1220 (1999).
5. T. F. Settle and L. Lu, ‘‘Digital bit stream generator for testing MPEG video decoder,’’ *IEEE Trans. Consum. Electron.* **42**(3), 422–430 (1996).
6. H. J. Hwang, I. S. Park, S. H. Park, B. U. Lee, and R.-H. Park, ‘‘Conformance test of MPEG-4 shape decoding,’’ *Proc. Int. Conf. Consumer Electronics 2001*, 146–147 (2001).
7. ISO/IEC 14496-2, *Information Technology—Generic Coding of Audio-Visual Objects. Part 2: Visual (Final Draft of International Standard)* (1998).
8. ISO/IEC JTC1/SC29/WG11 MPEG99/N2687, *MPEG-4 Video Verification Model Version 13.0* (1999).

Woo-Shik Kim received his BS degree in electronic engineering from Sogang University, Seoul, Korea, in 1999. He is currently working toward his MS degree in electronic engineering at Sogang University. His current research interests are video communication, signal processing, and computer vision.

In-Su Park received his BS degree in control and instrumentation engineering from Chosun University, Kwangju, Korea, in 1999. He is currently working toward his MS degree in electronic engineering at Sogang University, Seoul, Korea. His current research interests are video coding and multimedia communication system.

Su-Hyun Park received her BS degree in electronic engineering from Ewha Womans University, Seoul, Korea, in 1999. She is currently working toward her MS degree in information electronics at Ewha Womans University. Her current research interests are video communication and computer vision.

Byung-Uk Lee received his BS degree in electronics engineering from Seoul National University in 1979, his MS degree in electrical engineering from Korea Advanced Institute of Science and Technology in 1981, and his PhD in electrical engineering from Stanford University in 1991. He has worked for the Daewoo Electronics Co. from 1981 to 1985 and from 1991 to 1995. He joined the Department of Information Electronics, Ewha Womans University in 1995, where he is currently an associate professor. His current research interests include image compression and computer vision.

Rae-Hong Park received his BS and MS degrees in electronics engineering from Seoul National University, Seoul, Korea, in 1976 and 1979, respectively, and his MS and PhD degrees in electrical engineering from Stanford University, Stanford, CA, in 1981 and 1984, respectively. He joined the faculty of the Department of Electronic Engineering at Sogang University, Seoul, Korea, in 1984, where he is currently a professor. In 1990, he spent his sabbatical year at the Computer Vision Laboratory of the Center for Automation Research, University of Maryland, College Park, MD, as a visiting associate professor. He received a scholarship from the Korean Government, Ministry of Education, from 1979 to 1983, and a postdoctoral fellowship from the Korea Science and Engineering Foundation (KOSEF) in 1990. He received the Academic Award in 1987 from the Korea Institute of Telematics and Electronics (KITE). He also received the First Sogang Academic Award in 1997 and the Professor Achievement Excellence Award in 1999, all from Sogang University. He served as editor for the *KITE Journal of Electronics Engineering* in 1995 and 1996. He is a senior member of the IEEE. His current research interests are video communication, computer vision, and pattern recognition.