3–23 Static Test of MPEG-4 Visual Decoders

Woo-Shik Kim†, In-Su Park†, Su-Hyun Park††, Byung-Uk Lee††, and Rae-Hong Park†
†Dept. of Electronic Engineering, Sogang University, C.P.O. Box 1142, Seoul 100-611, Korea
††Dept. of Information Electronics, Ewha Womans University, Seoul, Korea
E-mail: †kws@eevision1.sogang.ac.kr, ††bulee@mm.ewha.ac.kr

Abstract

Moving picture experts group (MPEG)-4 visual conformance standard specifies methods to verify whether bitstreams and decoders meet the requirements at the specified profile and level. The test of decoders can be divided into two parts: the static test and the dynamic test. The static test can be performed by examining the decoder output with that of a reference decoder using test bitstreams. This paper proposes design methodologies of MPEG-4 visual test bitstreams, and presents experimental results on DCT scan type, DC/AC coefficient prediction, inverse quantization, and various macroblock type verifications.

1 Introduction

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

The MPEG-4 standard consists of system, visual, audio, conformance test, reference software, and delivery multimedia integration framework (DMIF) [2]. This paper deals with the conformance test [3], which consists of static tests [4] and dynamic tests [5]. The conformance test is to verify that the MPEG-4 decoder or bitstream is compliant to the MPEG-4 standard [6].

Static tests can be performed by verifying that a decoder under test decodes given test bitstreams precisely. There are several types of test bitstreams such as general, shape coding, scalability, error resilience, scalable still texture, and sprites [3]. The general test bitstreams cover texture coding, motion compensation, motion prediction bandwidth, variable length coding (VLC), and bitstream parsing. Each test bitstream is applied according to a specific profile and level.

The texture decoding is composed of VLC, inverse scan of discrete cosine transform (DCT) coefficients, DC/AC coefficient prediction, inverse quantization, and inverse DCT, as shown in Fig. 1 [6].

This paper proposes test bitstreams for texture decoding procedure and verification of type of macroblock (MB) which is basic unit of MPEG video bitstreams. The test bitstreams are made by modifying a software encoder [7], changing the conditions or flags of the encoder or by using test images of special patterns as input. The test result is checked by examining the reconstructed images of the reference decoder and the output of a bitstream analyzer.

2 Inverse Scan Test

The inverse scan constructs an array of 8×8 DCT coefficients from one dimensional sequence. The scan direction is not directly determined from the bitstream. It 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

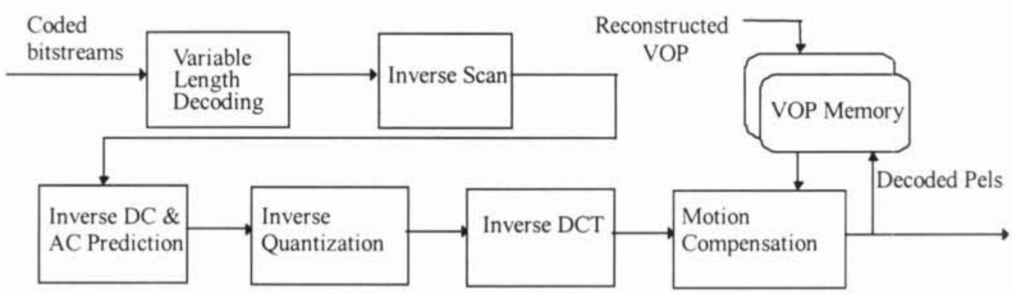


Figure 1: MPEG-4 video texture decoding process.

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 AC prediction direction. In the interlaced mode, the alternate vertical scan can be chosen regardless of AC prediction.

2.1 Zigzag scan test

We disable the *AC_pred_flag* to generate a zigzag scan test bitstream. If *AC_pred_flag* is disabled even though the AC prediction is performed, the reference decoder cannot reconstruct the image correctly. Therefore the encoder is modified not to perform the AC prediction, regardless of the condition whether the prediction is performed or not.

Fig. 2(a) shows the zigzag scan test image decoded by each scan method. The left one shows an image decoded by zigzag scan. We can see that all the blocks are reconstructed correctly. The middle one and the right one show images from a wrong decoder with scanning error.

2.2 Alternate scan test

When the *AC_pred_flag* is enabled, DCT coefficients are alternately scanned. The alternate vertical or horizontal scan is chosen according to the direction of AC prediction, which is determined by the direction of DC prediction. We generate test bitstreams from test images with DC coefficients of each block increasing monotonically from top to bottom and from left to right. In other words, the blocks of the test images are becoming brighter as one goes down or to the right. A reference decoder detects the DC prediction direction and the AC prediction direction, which will determine the alternate scan direction. Fig. 2(b) shows alternate vertical scan test images. Note that the lower side of the input image is brighter than the upper side, which results in the vertical DC prediction. The middle one is reconstructed correctly by alternate vertical scan and the others show the incorrect images from abnormal decoders.

Fig. 2(c) shows an alternate horizontal scan test bitstream decoded with each scan method. Note that the right side of the input image is brighter than the left side for the horizontal DC prediction. The left one shows the correct image with an alternate horizontal scan and the others are images obtained from abnormal decoders.

3 DC/AC Prediction Test

During encoding, DCT coefficients are compressed by predictive coding. The prediction direction depends on the horizontal and vertical DC gradients around the block. If the horizontal DC gradient is less than the vertical one, horizontal prediction direction is selected. AC prediction is performed when *AC_pred_flag* is set to one. Either coefficients from the first row or those from the first column of the previous block are used for the prediction of the current block. The direction of AC prediction is the same as that of DC prediction.

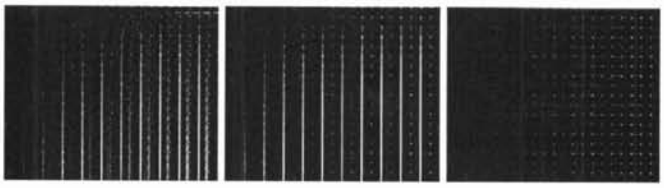
We design a test bitstream to verify whether a decoder performs DC/AC prediction correctly, and we can detect an error by comparing the decoded images. We generate a test DCT coefficient pattern with specified DC and AC coefficients. Fig. 3(a) is one of the test patterns, where the DC coefficients increase from left to right and from top to bottom, while it has the identical AC coefficients in the same row.

We can verify the same pattern from the normally decoded image. Figs. 3(b) and 3(c) show the images from abnormal decoders with fixed prediction direction. Fig. 3(b) is the result when the DC prediction direction is set to the vertical direction, and Fig. 3(c) is the result when AC prediction direction is fixed to horizontal direction. Comparing the result of Figs. 3(b) and 3(c), DC prediction error is more prominent than AC prediction error, since human eyes are more sensitive to low spatial frequency.

<zigzag> <altern. vertical> <altern. horizontal>

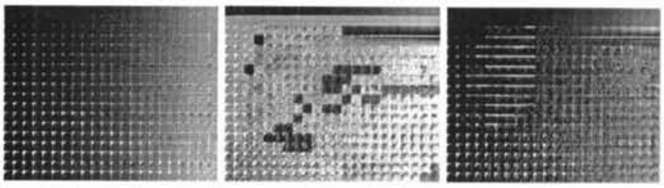
(a) zigzag scan test when AC prediction is disabled

(b) alternate vertical scan test



(c) alternate horizontal scan test

Figure 2: Scan type test.



(a) input frame (b) incorrect DC pred. (c) incorrect AC pred.

Figure 3: DC/AC prediction test.

Table 1: Nonlinear scaler for DC inverse quantization.

Component: Type	dc_scaler for qs range			
	$1 \sim 4$	$5 \sim 8$	$9 \sim 24$	$25 \sim$
Luminance: Type1	8	$2 \times qs$	$qs+8$	$2 \times qs-16$
Chrominance: Type2	8	$(qs+13)/2$		$qs-6$

(a) I-VOP, QP=1 (b) P-VOP, QP=1 (c) P-VOP, QP=29

Figure 4: Inverse quantization test.

4 Inverse Quantization Test

The 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 gives 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 a parameter, *quant_type*.

The first method is used when the *quant_type* is 1, and both the weighting matrix and scale factor are used. On the other hand if *quant_type* is 0, only the scale factor is used. In both methods, the DC coefficients are inverse quantized in a different way from other AC coefficients. The DC coefficients reconstruction is computed as follows,

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

where *dc_scaler* is defined in Table 1, with qs denoting an abbreviation of *quantizer_scale*.

To verify the inverse quantization, we fix the *quantizer_scale* for each picture. Fig. 4 shows the decoded images where Fig. 4(a) is coded as intra frame, while Figs. 4(b) and 4(c) are inter frame pictures. The *quantizer_scale* is set to 1 in Figs. 4(a) and 4(b) while it is set to 29 in Fig. 4(c). As the *quantizer_scale* increases, the reconstructed image becomes darker, because the DC coefficient decreases.

5 Macroblock Type Decision Test

The MPEG-4 data structure classifies objects and each object can be partitioned into several sub-structures. In the case of video objects, the lowest structure is a block, which is the basic unit for the DCT. The block has luminance type and chrominance type. The MB consists of luminance block and chrominance block, and it is the basic unit for combined motion, shape, and texture coding. For example, in 4:2:0 chrominance format, an MB has 4 luminance blocks and 2 chrominance blocks. The MB layer has the information about the alpha block which contains shape information, motion vector difference, difference quantizer, AC prediction, DCT coefficients of each block, MB type, etc.

Table 2: 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	–
I	3	INTRA
I	4	INTRA+Q
I	stuffing	–

The information in MB layer varies according to the MB type. It is determined by video object plane (VOP) type, namely, I-, P-, S-VOP, where I represents intra, P specifies inter, and S denotes sprite. We confine our scope to simple and core profiles, which do not cover S-VOP. For I-VOP, intra mode is used for all the MBs, therefore they do not need motion information. There are two intra MB types depending on the usage of difference quantization.

In case of P-VOP, both inter and intra mode MBs can be used, 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 used, an MB from the previous frame at the same position is duplicated.

Table 2 shows various types of MBs. Stuffing is bit insertion, which is employed to increase the bitrate at the encoder in order to prevent underflow at the decoder. The decoder simply discards the stuffing bits.

In the following subsections, the procedure to determine the MB type is briefly explained and the results of the proposed test bitstreams are illustrated. The test bitstreams are generated by changing MB types. We use the Akiyo test sequence. The image format is quarter common intermediate format (QCIF), 176×144 pixels, 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. We have developed a bitstream analyzer to extract and display parameters from an MPEG 4 video bitsteam. Fig. 5 shows decoded images from a reference decoder and a bitstream analyzer output.

5.1 MB type: not coded

MB type not coded is used only in P-VOP. When the variable named *COD* is set to 1, no further process is made

T T T N N N T T T	T T T 2 2 2 T T T	T T T 4 4 4 T T T
T T T N N N T T T	T T T 2 2 2 T T T	T T T 4 4 4 T T T
T T T N N N N T T	T T T 2 2 2 2 T T	T T T 4 4 4 4 T T
T T T N N N N T T	T T T 2 2 2 2 T T	T T T 4 4 4 4 T T
T T N N N N N N T	T T 2 2 2 2 2 2 T	T T 4 4 4 4 4 4 T
T N N N N N N N N	T 2 2 2 2 2 2 2 2	T 4 4 4 4 4 4 4 4
N N N N N N N N N	2 2 2 2 2 2 2 2 2	4 4 4 4 4 4 4 4 4

(a) not coded (b) INTER4V (c) INTRA+Q

Figure 5: Macroblock type test (top: decoded image, bottom: bitstream analyzer output).

and all information about the MB is assumed to be the same as the MB in the previous frame at the same position, which means that there is no motion vector or motion compensation.

We generate MB not coded test bitstream from a software encoder [7]. Fig. 5(a) is the decoded image by a reference decoder which shows that the decoder can reconstruct correctly when the MB type is not coded, and 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 the shape coding is limited by a bounding rectangle. The MB outside the bounding rectangle is denoted as 'T' and not coded MB is denoted as 'N'.

5.2 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 motion vectors or 8×8 motion vectors, where the number indicates the size of the block for motion compensation. When 8×8 motion vectors are used, the number of motion vectors needed per block is 4, which improves the accuracy of motion compensation. The MB of type 0 uses 8×8 motion vectors. When 16×16 motion vectors are employed, the MB type is either 1 or 2 depending on the usage of difference quantization. Fig. 5(b) shows the result when MB type is 2.

5.3 MB types 3 and 4: INTRA and INTRA+Q

When I-VOP is coded, all MBs are coded in intra mode and the MB type is 3, INTRA, or 4, INTRA+Q, depending on the usage of difference quantization. In P-VOP, when an MB has large motion compensation error, the MB is coded as an intra block and its MB type is 3 or 4.

The difference quantization is used to increase or decrease the quantization parameter for accurate rate control. The name of the variable is *DQUANT*. If the same value of *DQUANT* were used in all MBs, the quality of reconstructed image would be very poor due to large quantization error. For our conformance test, the absolute value of *DQUANT* is set to 1 and the sign is altered whenever the MB changes. The test bitstream is generated with P-VOP in which the MB type is set to 3 or 4. Fig. 5(c) shows the result in case of MB type=4, which contains the reconstructed image of the reference decoder and the output of the bitstream analyzer.

6 Conclusion

In this paper, we present design methodologies of MPEG-4 texture decoding tests and experimental results on inverse scan, DC and AC prediction, inverse quantization, and MB types. We show test methods 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 directly show the desired information in the bitstream. This test can be easily used during the R&D and mass production of the decoders. Future work includes the dynamic test which is related to rate control, buffer verification, and memory bandwidth, and automatic detection of the fault.

Acknowledgements

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

References

[1] T. Ebrahimi and C. Horne, "MPEG-4 natural video coding– An overview," *Signal Processing: Image Communication*, vol. 15, nos. 4-5, pp. 365-385, Jan. 2000.

[2] ISO/IEC JTC1/SC29/WG11 N3156, *Overview of the MPEG-4 Standard*, Dec. 1999.

[3] ISO/IEC 14496-4, *Information Technology- Very Low Bitrate Audio-Visual Coding: Conformance (Committee Draft)*, Dec. 1998.

[4] C. M. Kim, B. U. Lee, and R.-H. Park, "Design of MPEG-2 video test bitstreams," *IEEE Trans. Consumer Electronics*, vol. CE-45, no. 4, pp. 1213-1220, Nov. 1999.

[5] T. F. Settle and L. Lu, "Digital bit stream generator for testing MPEG video decoder," *IEEE Trans. Consumer Electronics*, vol. CE-42, no. 3, pp. 422-430, Aug. 1996.

[6] ISO/IEC 14496-2, *Information Technology- Generic Coding of Audio-Visual Objects, Part 2: Visual (Final Draft of International Standard)*, Dec. 1998.

[7] ISO/IEC JTC1/SC29/WG11 MPEG99/N2687, *MPEG-4 Video Verification Model Version 13.0*, Mar. 1999.