Australian Journal of Basic and Applied Sciences, 5(12): 3209-3216, 2011
ISSN 1991-8178

# Multilayers Fast Mode Decision Algorithms for Scalable Video Coding

Teddy Surya Gunawan, Haris Al Qodri Maarif, and OthmanO. Khalifa

Electrical and Computer Engineering Department, International Islamic University Malaysia, Jalan Gombak, 53100 Kuala Lumpur, Malaysia.

**Abstract:** Scalable video coding (SVC) is the extension of H.264/AVC standard. The features in SVC are also developed from the H.264/AVC standard, so that SVC has more features compared to H.264/AVC standard. This provides higher coding complexity in SVC encoder which causes higher encoding time for SVC. SVC is gaining great interest because of its ability and scalability to adapt in various network conditions. SVC allows partial transmission and decoding of a bitstream. This research deals with multilayers fast mode decision algorithm for decreasing encoding time or fastening the mode decision process of the SVC encoder. The proposed fast mode decision scheme has been implemented and is successfully decrease encoding time with negligible loss of quality and bitrate requirement. The simulation result shows the proposed fast mode decision algorithm provides time saving up to 45 % while maintaining video quality with negligible PSNR loss.

**Key words:** *Fast mode decision algorithm, scalable video coding, subjective evaluation, objective evaluation, base layer, enhancement layer.*

## INTRODUCTION

The advancement of multimedia and mobile communication has increased and reached the massive growth and commercial success. With the increasing reliance on the availability of multimedia information and the increasing mobility of individuals, there is a great need of providing multimedia information on the move (Al-Mualla, Nishan, & David, 2002). By this current situation, providing good multimedia quality especially in video coding become first priority. Development in video coding technology, e.g. MPEG-1 (MPEG, 1993), MPEG-2 (MPEG, 1994), and MPEG-4 (MPEG, 2004) standards, along with the developments and improvements of storage capacity, network infrastructures, and computing power are enabling an increasing number of video applications . The video applications areas today are ranging from multimedia messaging, video telephony, and video conferencing over mobile TV, wireless and wired Internet video streaming, standard and high-definition TV broadcasting to DVD, Blu-ray Disc, and HD DVD optical storage media. For these applications, a variety of video transmission and storage systems may be employed.

Scalable video coding (SVC) is gaining great interest because of its ability and scalability to adapt in various conditions of network. The term of scalability is referring to the removal of parts of the video bitstream in order to adapt it to the various needs or preferences of end users as well as to varying terminal capabilities or network conditions. SVC allows partial transmission and decoding of a bitstream (Kim, Xiong, & Pearlman, 2000). It contains the base layer and the enhancement layers. The base layer should be transmitted with very high reliability. On the other hand, the enhancement layers might be dropped or only transmitted partially according to the available network bitrate (McCanne, Vetterli, & Jacobson, 1997; Heiko Schwarz, Marpe, & Wiegand, 2007). This allows very fast and accurate network adaptation to variable bit rate channels.

A number of embedded 3-D video coding algorithms have been proposed by combining 3-D subband coding with motion compensation (Kim *et al*., 2000). McCanne *et al*. (McCanne *et al*., 1997) introduced a simple progressive video coding algorithm to cope with the restrictive delay requirements. SVC compression and adaptation technology has been developed for a variety of usage scenarios (Heiko Schwarz *et al*., 2007), including video broadcast/unicast, video conferencing, video streaming, and video surveillance. Jianhong and Jilin (Jianhong & Jilin, 2007) introduced an improved algorithm for low bit rate scalable video coding. Recently, a combination of scalable video coding with unequal erasure protection (UXP) has been proposed to overcome the problem when network is congested or in a poor condition (H. Schwarz, Marpe, Schierl, & Wiegand, 2005).

Recently, researches on fast mode decision have provided a good result to achieve efficient video encoding. It is found that the implementation of fast mode decision algorithm is not only speeding up the encoding time, but also preserves the good quality of encoded video. It is mentioned in (Goh, Kang, Cho, & Chung, 2009; H. Li, Z. G. Li, & C. Wen, 2006; Li, Li, Wen, & Chau, 2006) that the computational complexity and encoding time was more than two times faster with negligible reduced quality by implementing fast mode decision algorithm. Since the scalable video coding is an on-going standard, the scalable video coding has not been applied widely. Therefore, the objective of this paper is to develop efficient fast mode decision algorithms and to evaluate its

---

**Corresponding Author:** Teddy Surya Gunawan, Electrical and Computer Engineering Department, International Islamic University Malaysia, Jalan Gombak, 53100 Kuala Lumpur, Malaysia.
E-mail: tsgunawan@iium.edu.my

performance using subjective and objective evaluations. The rest of the paper is organized as follows. Section 2 discusses H.264 scalable video coding and fast mode decision. Section 3 describes the proposed multilayers fast mode algorithm while Section 4 discusses the implementation of the proposed algorithm. Experimental results and analyses are discussed in Section 5. Finally, Section 6 concludes this paper.

## ***Scalable Video Coding And Fast Mode Decision:***

SVC was standardized as an extension of H.264/AVC. It reuses some functions that have already been provided at H.264/AVC. Conceptually, the design of SVC covers a *Video Coding Layer* (VCL) and a *Network Abstraction Layer* (NAL), same as H.264/AVC was designed, as described in Fig. 1. VCL represents the code of the source content (input video), the NAL forms the VCL data in simple form and effective so that the VCL data can be utilized by many systems.

Data of the encoded video are gathered and organized into Network Abstraction Layer Unit (NALU) (Wiegand, Sullivan, G.Bjontegaard, & Luthra, 2003). NALUs are the packets of data which contain the integer number of bytes that represent the encoded video. NALU are classified into VCL NALU and non VCL-NALU. VCL-NALU is the units which contain encoded slice data partitions, and non VCL-NALU is the units which contain the additional information of the encoded video. The non-VCL NALU provides additional information which can assist the decoding process in the encoder side and also some related process like bitstream manipulation or display. They are parameter sets, which contain the infrequently changing information for a video sequence, and Supplemental Enhancement Information (SEI).

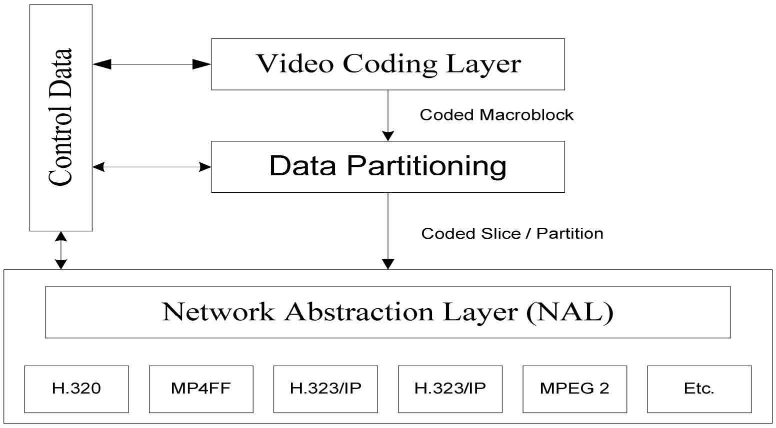

**Fig. 1:** Structure of H.264/AVC encoder.

In the H.264/AVC, the video frames are partitioned into smaller coding units which is called as macroblocks and slices (Wiegand *et al*., 2003). A video frame is partitioned into macroblocks which covers 16x16 luma samples and 8x8 samples of each of the two chroma components (for YUV420). The samples of a macroblock are predicted in terms of spatial or temporal, and the predicted residual signal is represented by using transform coding. The macroblock are partitioned into the slices which each of the slice can be parsed independently. The supported basic slices for the H.264/AVC are I-slice, P-slice, and B-slice (Wiegand *et al*., 2003). I-slice is *intra-picture* predictive coding using spatial prediction from neighbouring regions, P-slice is intra-picture predictive coding and inter-picture *predictive* coding with one prediction signal for each predicted region, and B-slice is intra-picture predictive coding, inter-picture predictive coding, and inter-picture *bipredictive* coding with two prediction signals that are combined with a weighted average to form the region prediction.

The most important parts of Scalable Extension of H.264/AVC are coding efficiency and complexity, and other parts are all common types in the H.264/AVC. Since SVC was developed as an extension of H.264/AVC with all of it well-designed core coding tools being inherited, one of the design principles of SVC was that new tools should only be added if necessary for efficiently supporting the required types of scalability. Three types of scalability have been implemented, such as temporal scalability, spatial scalability, and quality scalability. In this paper, we focused on combined scalability which combines spatial, quality, and temporal scalability to achieve maximum scalability as possible. Note that, JSVM software (JSVM, 2011) will be used for our reference software as it is open source and freely available and modifiable.

The supported mode decisions for scalable video coding are available for inter-prediction mode and intra-prediction mode. Those intra-prediction and inter-prediction are for base layer. For inter-prediction mode, there are seven features of macroblock, such as MODE_16x16, MODE_16x8, MODE_8x16, MODE_8x8, MODE_8x4, MODE_4x8, and MODE_4x4. For intra prediction, there are nine prediction modes for

INTRA_4x4, and four prediction modes for INTRA_16x16 and MODE_SKIP.For enhancement layers, two more modes are added, base_layer_mode and qpel_refinement_mode. These two modes indicate motion and prediction information including the partitioning of the corresponding macroblock of the base layer is used (Reichel, Schwarz, & Wien, 2006).

The new added features in scalable video coding provide the higher complexity for encoder than the prior video coding standard (Huang, Peng, Chiang, & Hang, 2007). The complexity makes the encoding process become longer and time consuming. In order to overcome the complexity problem, the fast mode decision algorithm will be an answer to speed up the encoding time of the encoder. In fast mode decision, the mode that will be used in encoding process is defined by Lagrangian parameter and rate distortion parameter. The two parameters to speed up the encoding process, rate distortion parameter and Lagrangian parameter are the parameters in Eq. (1) which are used as the value to decide the mode decision as well as the motion estimation.

$$J(MODE \mid QP, \lambda_{SSD}) = D(MODE \mid QP) + \lambda_{SSD} R(MODE \mid QP) \tag{1}$$

$$SSD = \sum_{ij} \left(C_{ij} - S_{ij}\right)^2 \tag{2}$$

The equation shows the relationship between rate distortion parameter $J$ and lagrangian parameter $\lambda$. $D$ is the average of the forward and backward sum of square difference (SSD) as described on Eq. (2), between the current macroblock (MB) and the motion-compensated matching blocks, and $\lambda$ is a weight parameter to control the contribution of the motion bits in total cost function. $C_{ij}$ and $S_{ij}$ are pixel of the current macroblock and pixel of the reference candidate macroblock, respectively.

The fast mode decision algorithm provided by Anselmo and Alfoso in (T.Anselmo & D.Alfonso, 2006) refers the class of predictive-recursive block matching methods. As the algorithm exploits the function of motion estimation, the motion vector is also one of the key points in the proposed fast mode decision algorithm. The fast mode decision algorithm runs in two steps, i.e. Course Search and Fine Search. The difference between the Fine Search and the Coarse Search is that Coarse Search is performed following the frames display order, while Fast Search follows the frames coding order. Therefore temporal predictors of the Coarse Search have to be scaled by an opportune coefficient before being used for the Fine Search.

Fast mode algorithm proposed by Lin (Lin, Yu, & Pan, 2006) is started by ranking the coding mode on its probability, and sorting on its priority queue. If the sequence is on low motion, the priority queue is updated by placing the current mode of macroblcoks in the first priority. Take the highest priority mode used for encoding process and compute the RD Cost of the chosen mode. If RD Cost is on the minimum level, then the current mode is selected as best mode. If it not, update again the priority mode until the RD Cost in on the minimum level, unless the current mode is the mode for the last sequence.

The algorithm presented by Wu (Wu & Tang, 2008) incorporates the ideas of motion attention model and mode decision algorithm in (He Li, Z. G. Li, & Changyun Wen, 2006). As described in Wu’s paper, the proposed fast mode decision algorithm decides the normal mode decision as implemented in H.264/SVC. After the normal mode decision of the SVC, the mode decision is taken by the possible mode on the current frame. There are four decision modes applied on this mode decision, once get the decided mode, the best mode is gained. The result presented in (Wu & Tang, 2008) shows that the algorithm provides good time saving with the negligible difference video quality.

## *Multilayers Fast Mode Decision Algorithms:*

The proposed algorithm is based on the traditional fast mode decision algorithm developed by Alfonso (T.Anselmo & D.Alfonso, 2006). This traditional fast mode decision algorithm applies the fast mode decision in base layer. On the other hand, our proposed algorithm expands the fast mode decision algorithms both into base and enhancement layers. As illustrated in the flowchart of Fig. 2, the algorithm is started by defining the layer that will be examined. As we can see from the diagram, value $i$=0 represents the base layer evaluation at the beginning, then continued to enhancement layer for higher ‘$i$’ value.

The evaluation of temporal predictor and spatial predictor are implemented. These processes are part of coarse search and fine search. In this algorithm, the evaluation of spatial and temporal predictor involves three closest neighbouring pixel of the current macroblock. The temporal predictor evaluates the closest neighbouring macroblock. For spatial predictor, location for $S1$ is $(x-1,y;N)$, $S2$ is $(x-1,y-1;N)$, and $S3$ is $(x,y-1;N)$, and for temporal predictor location for $T1$ is $(x+1,y;N-1)$, $T2$ is $(x,y;N-1)$, and $T3$ is $(x,y+1;N-1)$. The next process is updating the temporal and spatial predictor by adding 12 points grid in order to find the minimum SAD value that will be the chosen mode of the MB Mode Decision. The fine search is repetition of evaluating temporal and spatial predictor, but it runs under frames coding order while the first process (coarse search) follows frames display order. After the fine search process, the MB mode is obtained and the same process is repeated for the next layers.

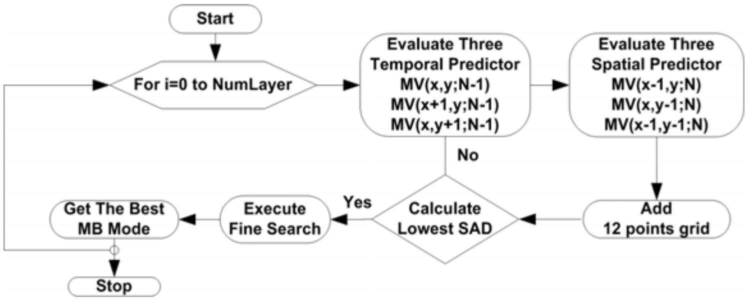

**Fig. 2:** Proposed Fast Mode Decision Algorithm.

Table 1 shows implementation of the proposed fast mode decision algorithm is implemented into two different profiles on triple layers encoding process. First profile is enhancement layer only which means that the algorithm is started from layer 1, this profile is namely as Proposed Fast Mode 1 (enhancement layer only). Second profile is base and enhancement layer which means that the algorithm is started from layer 0 (base layer), this profile is namely as Proposed Fast Mode 2 (base layer and enhancement layer). As a comparison, the original JSVM scheme without fast mode decision algorithm is also presented and named as high complexity (without fast mode decision). The original fast mode decision which has been implemented into JSVM is evaluated and namely as original low complexity (base layer only).

**Table 1:** Encoding Profile on Base Layers and Two Enhancement Layers.

| Profile | Description | Remarks |
|---|---|---|
| Profile 1 (000) | High Complexity | Without Fast Mode Decision |
| Profile 2 (100) | Original Low Complexity | Fast Mode in Base Layer Only |
| Profile 3 (011) | Proposed Fast Mode 1 | Fast Mode in Enhancement Layers Only |
| Profile 4 (111) | Proposed Fast Mode 2 | Fast Mode in Base layer and Enhancement Layer |

### *Implementation:*

The proposed algorithm has been implemented using JSVM reference software (JSVM, 2011) which is an open source code and written in C++ code. JSVM version 9.16 was used in our simulation. The original JSVM was then modified to include our proposed algorithm and then executed on a PC with Intel Core 2 Duo 2.4 GHz with 2 GB RAM and Ubuntu 10.10 operating system.

Two video sequences were used to evaluate our proposed algorithms, i.e. City and Crew with 4CIF resolution, as shown in Fig. 3(a) and Fig. 3(b), respectively. Five quantization parameter (QP) values were used in our simulation, i.e. 17, 24, 31, 38, and 45. Furthermore, four GOP sizes were used, i.e. 1, 2, 4, and 8.


(a) (b)

**Fig. 3:** Video Sequences, (a) City and (b) Crew.

## RESULTS AND DISCUSSIONS

In this section, the performance evaluation metrics will be discussed. The simulation results, discussion, and comparison for SVC without fast mode decision, original fast mode decision, and proposed fast mode decision will be presented.

***Performance Metrics:***

The performance analysis of this thesis is based on the metric proposed by G. Bjontegaard which has been presented in ITU-T VCEG 13th meeting. BDBR, BDPSNR, and Time Saving are used as the evaluation metric for fast mode decision analysis, as described on Eq. (3), (4), and (5), respectively. These equations are used as a standard performance evaluation for scalable video coding, as referred by ITU-T (Bjontegaard, 2001).

$$BDBR = \frac{Bits_{proposed} - Bits_{jsvm}}{Bits_{jsvm}} \times 100\% \qquad (3)$$

$$BDPSNR = PSNR_{proposed} - PSNR_{jsvm} \qquad (4)$$

$$TimeSaving = \frac{Time_{proposed} - Time_{jsvm}}{Time_{jsvm}} \times 100\% \qquad (5)$$

BDBR (Bjonteggard Delta Bitrate), is value of different bitrate, BDPSNR (Bjonteggard Delta Peak Signal to Noise Ration - PSNR) is the different value of PSNR and the Time Saving shows the computation time between the high complexity and fast mode decision algorithm (Bjontegaard, 2001).

***Results Analysis:***

In this section, the analysis of the complexity of SVC was presented in terms of encoding time. The time comparison between encoder with the high complexity and low complexity are showed. Note that, the high complexity and low complexity refers to original JSVM encoder without fast mode decision and with fast mode decision, respectively. In summary, the encoder with high complexity showed longer encoding time than the encoder with low complexity. Not only the encoding time, but also the quality itself will be compared between high complexity and low complexity.

Three encoding schemes, i.e. high complexity, original low complexity (original fast mode), and proposed low complexity algorithms (proposed fast mode 1 and 2) were implemented and evaluated (refer also Table 1 for the encoder profile). The high complexity scheme performs the full mode decision which examines all the possible modes in the encoder used to do encoding process (ITU-T & ISO/IEC-JTC, 2009), the original low complexity scheme performs the fast mode decision by selecting the best mode in base layer which is used for encoding process without have to evaluate all available modes (T.Anselmo & D.Alfonso, 2006), and the proposed low complexity performs the fast mode decision algorithm in both base layer and enhancement layers.

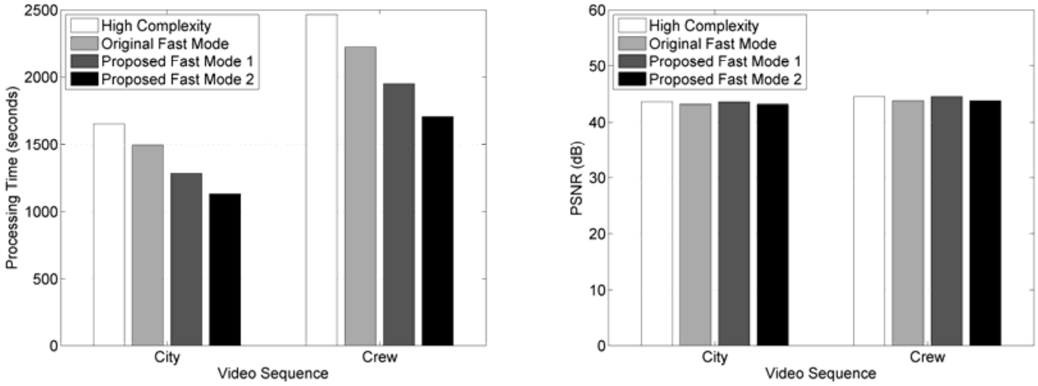

**Fig. 4:** Processing Time and PSNR for Various Video Sequences.

Fig. 4, Fig. 5, and Fig. 6 show the comparison of processing time and PSNR for various video sequences, QP and GOP values. The complexity analysis is mainly measured in terms of processing time, in which the faster the better. The video quality is measured using PSNR. Fig. 4 shows the processing time and PSNR for City and Crew sequences using fixed QP and GOP, i.e. 17 and 8. Fig. 5 shows the processing time and PSNR for various QP values, i.e. 17, 24, 31, 28, and 45, using fixed GOP of 8 and Crew video sequence. The higher the QP value, the better the quality and the longer the processing time. Finally, Fig. 6 shows the processing time and PSNR for various GOP values, i.e. 1, 2, 4, and 8, using fixed QP of 17 and Crew video sequence. The higher the GOP value, the longer it takes to encode the video sequence, while the quality is almost the same. The results showed that our proposed algorithms outperformed the original fast mode algorithm in terms of processing time while still maintaining the video quality, i.e. about the same PSNR value.

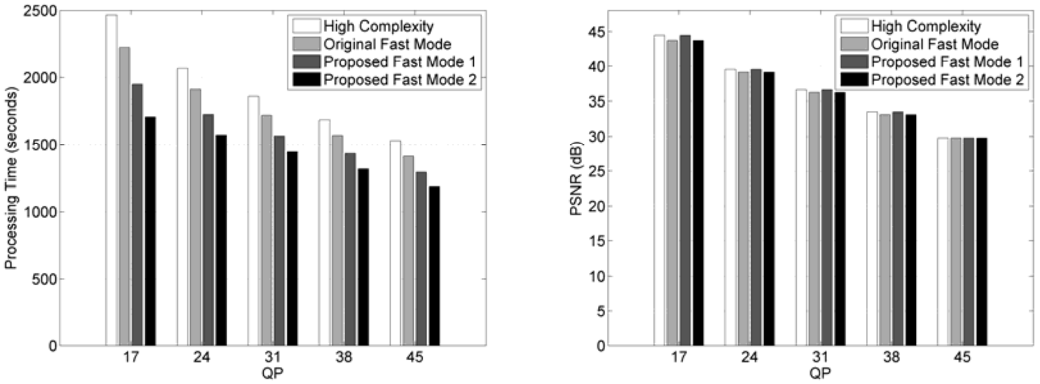

**Fig. 5:** Processing Time and PSNR for Various QP Values.

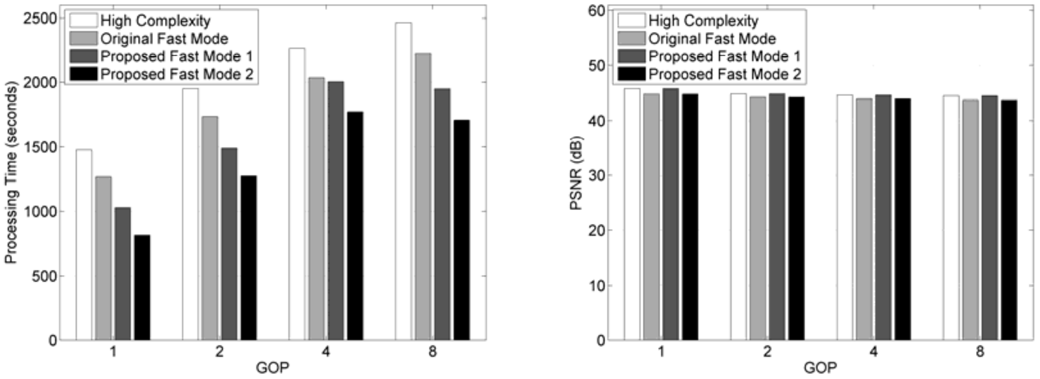

**Fig. 6:** Processing Time and PSNR for Various GOP Values.

Table 2 shows the time saving comparison of the SVC with high complexity (without fast mode), original fast mode algorithm, and proposed fast mode algorithm 1 (enhancement layers only) and 2 (base layer and enhancement layers). The proposed fast mode 1 can achieve the time saving from about 9% up to 30%, while the proposed fast mode 2 can achieve the time saving from about 16% up to 45%, compare to the original fast mode which can vary between 6% to 14%. It can be concluded that our proposed algorithms can achieve the highest time saving when used with QP of 17 and GOP of 1.

**Table 2:** Time Saving for City Sequence with Various QP and GOP Values.

| GOP | QP | Processing Time (seconds) | | | | Time Saving | | |
|---|---|---|---|---|---|---|---|---|
| | | High Complexity | Original Fast Mode | Proposed Fast Mode 1 | Proposed Fast Mode 2 | Original Fast Mode | Proposed Fast Mode 1 | Proposed Fast Mode 2 |
| 1 | 17 | 1086 | 936 | 754 | 601 | 13.81% | 30.57% | 44.66% |
| | 24 | 985 | 858 | 702 | 574 | 12.89% | 28.73% | 41.73% |
| | 31 | 924 | 809 | 669 | 551 | 12.45% | 27.60% | 40.37% |
| | 38 | 885 | 776 | 643 | 534 | 12.32% | 27.34% | 39.66% |
| | 45 | 868 | 759 | 630 | 520 | 12.56% | 27.42% | 40.09% |
| 2 | 17 | 1379 | 1226 | 1038 | 883 | 11.09% | 24.73% | 35.97% |
| | 24 | 1255 | 1129 | 975 | 848 | 10.04% | 22.31% | 32.43% |
| | 31 | 1229 | 1112 | 972 | 856 | 9.52% | 20.91% | 30.35% |
| | 38 | 1175 | 1069 | 953 | 846 | 9.02% | 18.89% | 28.00% |
| | 45 | 1143 | 1044 | 921 | 821 | 8.66% | 19.42% | 28.17% |
| 4 | 17 | 1547 | 1389 | 1357 | 1198 | 10.21% | 12.28% | 22.56% |
| | 24 | 1437 | 1303 | 1274 | 1138 | 9.32% | 11.34% | 20.81% |
| | 31 | 1393 | 1274 | 1252 | 1133 | 8.54% | 10.12% | 18.66% |
| | 38 | 1348 | 1241 | 1220 | 1114 | 7.94% | 9.50% | 17.36% |
| | 45 | 1283 | 1187 | 1162 | 1069 | 7.48% | 9.43% | 16.68% |
| 8 | 17 | 1654 | 1493 | 1283 | 1132 | 9.73% | 22.43% | 31.56% |
| | 24 | 1550 | 1405 | 1231 | 1096 | 9.35% | 20.58% | 29.29% |
| | 31 | 1496 | 1369 | 1223 | 1108 | 8.49% | 18.25% | 25.94% |
| | 38 | 1447 | 1334 | 1204 | 1098 | 7.81% | 16.79% | 24.12% |
| | 45 | 1361 | 1271 | 1139 | 1047 | 6.61% | 16.31% | 23.07% |

Table 3 shows the performance evaluation in terms of BDBR and BDPSNR, as explained in Eq. (3) and (4), respectively. On average, it can be deducted that our proposed fast mode algorithms requires minor additional bits while preserving the video quality. It is interesting to show here that the proposed fast mode 1 (enhancement layers only) provided significant time saving (as shown in Table 2) while obtained the same bitrate requirements and video quality as the high complexity. Therefore, applying fast mode algorithm on enhancement layers only provide significant reduction in processing time while preserve the video quality and bit rate requirement. However, the proposed fast mode 2 provided the highest time saving with negligible loss in video quality as shown in Table 2 and Table 3. Moreover, informal subjective evaluation of video quality conform the above objective evaluation (PSNR).

**Table 3:** BDBR and BDPSNR Performance Measures for Various GOP and QP.

| G OP | QP | High Complexity | | Original Fast Mode | | Proposed Fast Mode 1 | | Proposed Fast Mode 2 | |
|---|---|---|---|---|---|---|---|---|---|
| | | Bitrate (Kbps) | Y-PSNR (dB) | BDBR | BDPSNR | BDBR | BDPSNR | BDBR | BDPSNR |
| 1 | 17 | 21002.16 | 45.12 | 19.32% | 3.79 | 0.00% | 0.00 | 19.32% | 3.79 |
| | 24 | 6758.57 | 38.92 | 24.73% | 1.59 | 0.00% | 0.00 | 24.73% | 1.59 |
| | 31 | 1360.85 | 33.74 | -3.08% | 0.04 | 0.00% | 0.00 | -3.08% | 0.04 |
| | 38 | 382.75 | 29.35 | -9.71% | -0.11 | 0.00% | 0.00 | -9.71% | -0.11 |
| | 45 | 151.20 | 25.51 | -13.27% | -0.39 | 0.00% | 0.00 | -13.27% | -0.39 |
| | *Average* | | | 3.60% | 0.99 | 0.00% | 0.00 | 3.60% | 0.99 |
| 2 | 17 | 17845.41 | 44.19 | 6.88% | 1.77 | 0.00% | 0.00 | 6.88% | 1.77 |
| | 24 | 5507.60 | 38.72 | 17.38% | 1.19 | 0.00% | 0.00 | 17.38% | 1.19 |
| | 31 | 1284.67 | 34.33 | 1.82% | 0.34 | 0.00% | 0.00 | 1.82% | 0.34 |
| | 38 | 427.91 | 30.17 | -4.19% | 0.07 | 0.00% | 0.00 | -4.19% | 0.07 |
| | 45 | 188.14 | 26.01 | -9.83% | -0.05 | 0.00% | 0.00 | -9.83% | -0.05 |
| | *Average* | | | 2.41% | 0.66 | 0.00% | 0.00 | 2.41% | 0.66 |
| 4 | 17 | 16476.67 | 43.79 | 1.67% | 0.87 | 0.00% | 0.00 | 1.67% | 0.87 |
| | 24 | 5017.79 | 38.66 | 8.41% | 0.79 | 0.00% | 0.00 | 8.41% | 0.79 |
| | 31 | 1386.54 | 35.10 | 1.92% | 0.43 | 0.00% | 0.00 | 1.92% | 0.43 |
| | 38 | 475.35 | 31.08 | -3.79% | 0.20 | 0.00% | 0.00 | -3.79% | 0.20 |
| | 45 | 217.92 | 26.85 | -7.16% | 0.08 | 0.00% | 0.00 | -7.16% | 0.08 |
| | *Average* | | | 0.21% | 0.47 | 0.00% | 0.00 | 0.21% | 0.47 |
| 8 | 17 | 15659.14 | 43.55 | -0.36% | 0.41 | 0.00% | 0.00 | -0.36% | 0.41 |
| | 24 | 4374.17 | 38.43 | 1.36% | 0.41 | 0.00% | 0.00 | 1.36% | 0.41 |
| | 31 | 1322.96 | 35.35 | 2.55% | 0.54 | 0.00% | 0.00 | 2.55% | 0.54 |
| | 38 | 497.56 | 31.69 | -1.47% | 0.37 | 0.00% | 0.00 | -1.47% | 0.37 |
| | 45 | 230.57 | 27.39 | -3.79% | 0.26 | 0.00% | 0.00 | -3.79% | 0.26 |
| | *Average* | | | -0.34% | 0.40 | 0.00% | 0.00 | -0.34% | 0.40 |

### *Conclusions:*

Multilayers fast mode algorithms have been proposed in this paper, one which apply the fast mode algorithm in the enhancement layers only (proposed fast mode 1) and one in both layers (proposed fast mode 2). The proposed fast mode 1 provides time saving up to 31 %, while the proposed fast mode 2 provides time saving up to 45%. In terms of bit rate requirement and video quality, the proposed fast mode 1 has the same performance as the original high complexity algorithm. On the other hand, the proposed fast mode 2 has negligible bitrate and video quality differences compare to the original high complexity algorithm. Therefore, it can be concluded that our proposed algorithms has the potential to be included in the H.264 scalable video coding standard.

## REFERENCES

Al-Mualla, E., M.C. Nishan and R.B. David, 2002. *Video Coding for Mobile Communications. Efficiency, Complexity, and Resilience*: Academic Press.

Bjontegaard, G., 2001. *Calculation of average PSNR differences between RD-curves.* Paper presented at the ITU-T VCEG 13th Meeting.

Goh, G., J. Kang, M. Cho and K. Chung, 2009. *Fast Mode Decision for Scalable Video Coding Based on Neighboring Macroblock Analysis.* Paper presented at the Proceedings of the 2009 ACM Symposium on Applied Computing.

Huang, H.C., W.H. Peng, T. Chiang and H.M. Hang, 2007. Advances in the Scalable Amendment of H.264/AVC. IEEE Communication Magazine, 45(1).

ITU-T, & ISO/IEC-JTC., 2009. *JSVM 9.16 Overview*.

ITU., 1993. ITU-T Rec. H.261: Video Codec for Audiovisual Services at p x 64 kbit/s: International Telecommunication Union.

ITU., 2000. ITU-T Rec. H.263: Video Coding for Low Bit Rate communication: International Telecommunication Union.

ITU, 2007. Advanced Video Coding for Generic Audiovisual Services: ITU-T Rec. H.264 and ISO/IEC 14496-10 (MPEG-4 AVC), ITU-T and ISO/IEC JTC 1.

Jianhong, Y. and L. Jilin, 2007, December. *An Improved Method for Scalable Video Coding at Low Bit Rates.* Paper presented at the IEEE International Symposium on Intelligent Signal Processing and Communication Systems.

JSVM, 2011. Joint Scalable Video Model (JSVM) software: reference software for Scalable Video Coding (SVC): ISO/IEC MPEG and the ITU-T Video Coding Experts Group.

Kim, B.J., Z. Xiong and W.A. Pearlman, 2000. Low bit-rate scalable video coding with 3D set partitioning in hierarchical trees (3D SPIHT). IEEE Transactions on Circuits and Systems for Video Technology, 10(8).

Li, H., Z.G. Li and C. Wen, 2006. Fast Mode Decision Algorithm for Inter-Frame Coding in Fully Scalable Video Coding. IEEE Transactions on Circuits and Systems for Video Technology, 16(7).

Li, H., Z.G. Li and C. Wen, 2006, May. *Fast Mode Decision for Coarse Grain SNR Scalable Video Coding.* Paper presented at the IEEE International Conference on Acoustics.

Li, H., Z.G. Li, C. Wen and L. Chau, 2006, September. *Fast Mode Decision for Spatial Scalable Video Coding.* Paper presented at the IEEE International Symposium on Circuits and Systems.

Lin, Z., H. Yu and F. Pan, 2006. *A scalable fast mode decision algorithm for H.264.* Paper presented at the IEEE International Symposium on Circuits and Systems.

McCanne, S., M. Vetterli and V. Jacobson, 1997. Low-complexity video coding for receiver driven layered multicast. IEEE Journal on Selected Areas in Communication, 15.

MPEG, 1993. *Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to About 1.5 Mbit/s-Part 2: Video*: ISO/IEC 11172-2 (MPEG-1 Video), ISO/IEC JTC 1.

MPEG., 1994. *Generic Coding of Moving Pictures and Associated Audio Information-Part 2: Video*: ITU-T Rec. H.262 and ISO/IEC 13818-2 (MPEG-2 Video), ITU-T and ISO/IEC JTC 1.

MPEG, 2004. *Coding of audio-visual objects-Part 2: Visual*: ISO/IEC 14492-2 (MPEG-4 Visual), ISO/IEC JTC.

Reichel, J., H. Schwarz and M. Wien, 2006. *Joint Scalable Video Model (JSVM) 2.0 Reference Encoding Algorithm Description*. Buzan, Korea: ISO/IEC JTC1/SC29/WG1 1, Doc. N7084.

Schwarz, H., D. Marpe, T. Schierl and T. Wiegand, 2005. *Combined scalability support for the scalable extensions of H.264/AVC.* Paper presented at the International Conference on Multimedia and Expo.

Schwarz, H., D. Marpe and T. Wiegand, 2007. Overview of the Scalable Video Coding Extension of the H.264/AVC Standard. IEEE Transactions Circuits System Video Technology, 17(9).

Anselmo, T., and D. Alfonso, 2006, September. *Fast Rate-Distortion Optimization in the H.264/AVC standard.* Paper presented at the 2nd International Conference on Mobile Multimedia (MobiMedia), Alghero, Italy.

Wiegand, T., G.J. Sullivan, G. Bjontegaard and A. Luthra, 2003. Overview of the H.264/AVC standard. IEEE Transactions Circuits System Video Technology, 13(7).

Wu, Y.D. and C.W. Tang, 2008, April. *The Motion Attention Directed Fast Mode Decision for Spatial and CGS Scalable Video Coding.* Paper presented at the IEEE International Symposium on Broadband Multimedia Systems and Broadcasting.