Volume 8, Issue 1
February 2019

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

Fast Mode Decision with Flexible Macroblock Ordering in SVC

Muhammad Shoaib

College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia

E-mail: muhshoaib@ksu.edu.sa

ABSTRACT

Recent developments viable high quality videos at the cost of significantly increased in the computational complexities for multi-layer video encoding. Especially for wireless devices with limited processing capability such as smartphones that resulted hard to visualize high-quality content. These advances lead challenges towards low latency applications such as video conferences and streaming where retransmission approach for missing content is not feasible. So, there is critical demand for research to have an efficient coding technique to achieve the high quality video along with low complexity and error resilience features. Proposed scheme exploits the local characteristics of encoded videos to reduce computation complexity with flexible macroblock ordering tool as an error resilience scheme. Performance of proposed scheme reveals that algorithm can achieves 31% on average in terms of reducing the computing complexity, compared to the conventional approach with the same bit rate distortion.

Keywords: *Computation complexity, content feature, error resilience, flexible macroblock ordering, scalable extension of H.264*

1. INTRODUCTION

Latest advances in video coding supporting technologies are evident in the video content distributing industry. The availability of heterogeneous devices places growing demands with high quality videos especially when transmitted over unreliable networks for real time applications. These heterogeneous devices such as mobile phones with different functionalities, inspired towards practical technologies which provides videos with different formats. These devices have limited resources to computational complexity and they operate under low bit-rates.

Scalable video coding (SVC) is the scalable extension of the highest developed video coding standard MPEC-4 H.264/AVC [9]. It allows flexible transmission and supports suitable partial bit-streams extraction. The extracted partial bit-stream is required for target bit rates, with reduced temporal or spatial resolutions or quality scalabilities. In addition, this extracted procedure retains a reconstruction quality related to the rate of the partial bit streams [22]. However, rigid real-time constrains with unpredictable communication channel can cause packet losses, resulted in adequate error protection, so error concealment techniques are necessary. SVC coding scheme contains hierarchical prediction structure that imposes penalty to the transmitted video streams with possible error propagation. Lesser packet loss could transform into higher frame-error [7]. Hence, an efficient compression scheme is highly demanded that augment the low complexity video transmission for heterogeneous devices. These requirements have not yet received much attention especially with error prone environment. SVC supports several error resilience tools such as slice coding and flexible macroblock ordering (FMO) to improve error robustness in the bit stream. Slices and FMO increase the stability against error as for any packet loss, only a portion of a frame will be degraded and the error cannot propagate to other slice in the frame due to slices independency [13, 17]. FMO allows to subdivide the picture into a number of slice groups and mapped each slice group with number of macroblocks. Then, different slice groups are assigned with prioritized and region of interest. Significant error resilience tools adopted by the H.264 video coding includes reference picture identification and selection, intra MB or picture refresh, data partitioning in a frame, redundant slices or pictures, picture parameter sets, gradual decoding refresh, SP or SI pictures and constrained intra prediction techniques.

In the literature, many researches are available in coding efficiency with error resilience where FMO are used. But similar studies ignored computation complexity along with slice coding in H.264 and in SVC. In this paper, we reduce the complexity for streaming video in unreliable networks using FMO for the same coding efficiency. This paper extends our previous paper [14] towards propose scheme for low complexity with hierarchal prediction structure and extend special scalability in which the resolution ratios between enhancement layer and base layer is non-dyadic. In this research, we make the following contribution:

- We use FMO as an error resilience for bit stream encoding and applied error concealment at the decoder for output with proposed low complexity scheme without compromising rate distortion efficiency.

The rest of the paper is organized as follows. In Section 2, we present an overview of the related work. In Section 3, we discuss the essentials for FMO with slice coding and Section 4 describes the proposed scheme. In Section 5, we present the simulation results for video streams in an error-prone environment. Finally, Section 6 presents our concluding remarks.

2. RELATED WORK

The research community for video coding made substantial efforts to develop and enhance low complexity error resilient coding. Cui et al. [22] proposed a fast mode decision algorithm for high definition video sequences to speed up the

SVC encoding by exploiting the relationship between rate-distortion with statistical mode decision of enhancement layer. In the proposed mode decision scheme, first they evaluate the complexity of video content, then set threshold for rate distortion cost for inter coding mode. However, they neglect the losses due to transmission errors. El-Shafai et al. [20] proposed an adaptive hybrid error resilience and error concealment algorithm applied on 3D video that are transmitted over error susceptible wireless channels. In the proposed scheme, an adaptive preprocessing error resilience scheme were proposed using slice structured coding modes with FMO mapping that will be used for error concealment. However, they ignore the impact of increase in the computation complexity especially for limited resources devices. Asif et al. [12] proposed a generalized multilayer architecture which provides an optimized hierarchical scheme to select macroblock prediction parameter. The proposed scheme shortlisted the candidate prediction parameter prior to rate distortion optimization process. However, their main drawback is ignorance of the inherent transmission error. Huong et al. [19] generated a frame-by-frame FMO map for considering the importance of macroblock to the other macroblock in a slice group, and in the macroblock of the next frame. The main drawback is significantly increase in the encoding complexity. Yang et al. [16] proposed error concealment technique exploiting visual object as concealment units. FMO was incorporated at the encoder to utilize the special correlation. However, they ignore the increase in the computation complexity.

In [11], authors investigate video streaming limitations for multi-user, transmitted over wireless networks with quality of service (QoS). Their research resolves video streaming issues transmitted over multi-channel with multi-radio and multi-hop wireless networks. They developed distributed scheduling schemes with the minimized video distortion objectives to achieve a certain fairness level. However, the research ignored quality degradation in the videos during transmission. Ziviani et al. [3] estimate the various joint adoption possibilities to examine numerous QoS schemes to enhance the quality of an MPEG video stream that was transmitted via a lousy network. The research consists an error-control technique include a channel coder, a packetizer exercised at transport layer with unequal error-protection. The approach enhances the tolerate to the losses. In this scheme main flaw includes re-encoding or transcoding is essential for heterogeneous devices to decode video.

Paluri et al. [17] propose a generalized linear model with lesser delay and complexity for predicting CMSE, contributed by the loss of coded video slices. The approach uses unequal slice protection. The main drawback is the approach is unable to exploit local characteristics of video content. Buhari et al. [2] proposed human visual system based on watermarking algorithm with less complexity by exploiting texture features in the frame. The main drawback is the computation complexity increases. Koziri et al. [13] explore on the problem of proper size of the slice so the load imbalances among threads based on the parallel slice can be reduced. The main drawback is increase in the complexity. Ali et al. [1] introduces a sub-partition beside three partitions for coding prioritized information. The main drawback is increasing the complexity. Dan et al. [5] proposed region of interest coding, adaptively selected from the pre-encoded scalable video bit stream and enabled the extraction with adaptive setting of desirable region of interest location, size and resolution. Their approach meets the requirement for heterogeneous end-user devices. The main drawback is complexity increases with inherent condition of erroneous environment.

All of these methods are efficient with acceptable quality distortion. However, the influence of transmission errors and simultaneous provision of low complexity is not examined with error-resilient context such as slice coding scheme. The focus of the present work is on the multi-layer coding for heterogeneous devices in error prone environment with low complexity using FMO as resilience. Target applications include low delay multimedia services such as video conferences and streaming where retransmission approach is not feasible. In the following section, we briefly discuss FMO and slice coding relevant to proposed scheme.

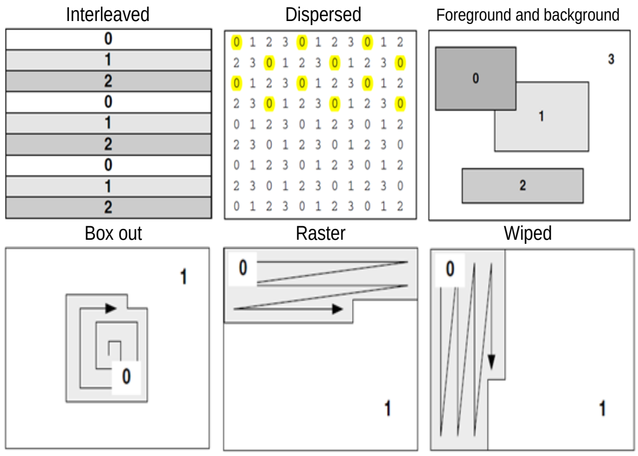


Fig. 1: FMO types (interleaved as type 0, dispersed as type 1, foreground as type 2, box out as type 3, raster scan as type 4, and wipe out as type 5), while 0,1, 2, 3 represents slice groups.

3. FLEXIBLE MACROBLOCK ORDERING

To enhance error robustness in the stream, H.264 video coding standard supports several tools. In this paper, we will emphasis on slice coding efficiency with error concealment. FMO is a very efficient supporting error-resilience tool in SVC. In FMO, the main advantage includes, it is not limited to slices with neighboring macroblocks. By exploiting this characteristic, each macroblock can freely be assigned to any slice group. The advantage of using a scattered macroblock order is recognized in the reconstruction of missing blocks which is easier in slice coding using the surrounding macroblocks information. Consequently, the errors are equally scattered in the entire picture rather than limited to one certain region. FMO has a huge prospect especially in error resilience scenarios. The subjective and objective qualities of videos are enhanced especially in the severe packet losses environment [13, 17, 19]. Figure 1 illustrate different types of FMOs.

Volume 8, Issue 1
February 2019

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

In FMO encoding, macroblocks can be assigned freely to any slice groups, so the decoder need to aware which macroblock is assigned to which slice group. The awareness is transmitted by the macroblock allocation map along with the encoded macroblocks. The macroblock allocation map is included in the picture parameter set. H.264 standard includes seven choices to store the macroblock allocation map information in the picture parameter set. As FMO is core scheme applied in this research, a brief overview of the FMO types 0 to 5 is discussed. In FMO type 0, maximum number of macroblocks are sequentially assigned in each slice group which follow raster scan order. When all slice groups have been utilized and still some are macroblocks left, the whole process is repeated initiated from slice group 0. FMO type 1 has a scattered or dispersed pattern and adopted with a predefined function. The layout of the macroblock allocation map depends on the number of slice groups in the frame. Here, the idea is to avoid laying two macroblocks of the same slice group next to each other. FMO type 2 are commonly used with region of interest scenarios in which high quantization and resilience will be applied in the foreground as compared to background. FMO types 3, 4 and 5 are terms as evolving types as it divides the macroblocks over two different slice groups. Figure 1 illustrate different types from FMO type 0 to FMO type 5, while 0, 1, 2, 3, ... represents slice groups in each FMO type.

4. PROPOSED SCHEME

In H.264, the features in the video frames are usually examined via a mapping in macroblock mode with motion vectors. In this research, we enhance our previous research [14] to analyze the motion activities in a frame, to determine a background region in the video content. We extend our scheme with the flexible macroblock ordering in slice coding and check for performance. Figure 2 shows mode and motion activities in the frame of bus sequence. The image in ‘a’ is created from the

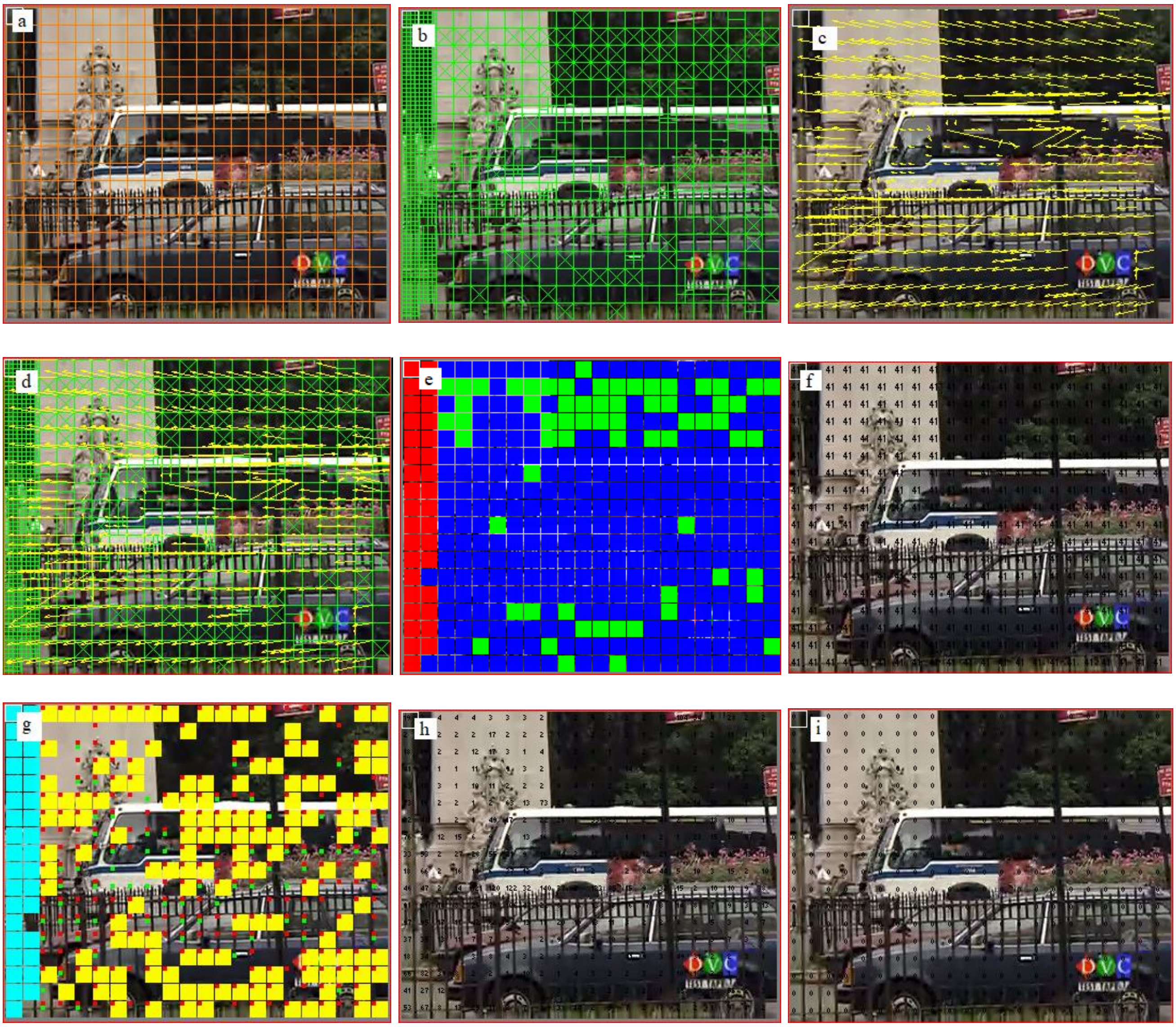


Fig. 2: Illustration for a frame in the bus sequence represents: (a) macroblock grid size, (b)macroblock partition and sub-partitions, (c) motion vector, (d) macroblock partition grid with motion vector, (e) macroblock type overlay, (f) macroblock quantization parameter overlay, (g) macroblock inter layer prediction overlay, (h) weight overlay, (i) slice overlay.

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

simulated bit stream that depicts the macroblock grid consist of 16 pixels in the picture. In picture 'b' shows the type of macroblock overlay with different grid sizes. These grid size represents macroblock with 16x16, 16x8 and 8x16, 8x8, 4x8 and 4x4 different partitions and sub partitions. Picture in 'c' yellow pointed arrows represent motion vectors for inter-predictions from backward and forward reference frames. In 'd', it is realized that motion vectors are mostly restricted to the active regions in the frame. Whereas, these arrow with different size of partitions on the right shows relation between macroblock partition and sub-partitions with motion vectors in a frame. In 'e' shows different macroblock types overlay; red shows intra types, blue shows inter, green grid shown skip types. In 'f' the number shows macroblock quantization parameter overlay. In 'g' represents inter layer prediction overlay in which yellow grip show base mode, light blue shows intra base layer. In 'h', number in the frame represents weighted prediction overlay. Finally, in 'i' number in the frame represents macroblock partition index to slice number.

Inspired by mode and motion vector relationship for various sizes, we identify the background regions in video content. This region is always containing large-size partitions with 16x16, 16x8 and 8x16. On the other hand, the MB partition types are diverse in foreground with active motion or with rich textures. It consists 8x8, 4x8 and 4x4 macroblock partitions. Same macroblock mode distribution also exists in the encoded frames in the base layer of scalable video coding [4, 6].

a. Macroblock Mode and Motion

To measure the macroblock mode and its motion vector, we analyze the motion activity computed by the following equation:

$$L = \|V_1 - V_2\|_2^2 \quad (1)$$

where L denotes the motion activity of the base layer MB. $(\|.\|_2)^2$ shows the square of $l^2 - norm$ of the motion vector difference. Motion vectors of the two neighboring blocks are represented with V_1 and V_2 respectively. We calculate the similarity between V_1 and V_2 with the square of their Euclidean distance.

b. Macroblock Mode Parameter

There exists distribution correlations between macroblock mode in base layer and its corresponding enhancement layer by [6]. For spatial scalability, the same trends exit in each MB mode partitions at the base layer and its up-sampled MBs at the enhancement layers. Moreover, the mode partition of the MBs in the current frame is similar to its the reference frames in the temporal scalability. This correlation significantly correlates for background in enhancement layers frames. These findings inspired us to derive MB mode parameter (α) from the mode context of local MBs in the base layer. Then, the parameter α is used to estimate the motion and texture features of the collocated MBs in the enhancement layers.

In enhanced special scalability (ESS), one, two or four base layer MBs are required to computer an enhancement layer MB [21]. Figure 3 shows two successive spatial layers: the base layer and the enhancement layer. Moreover, extended up-sampled base layer MB 'C' (blue grids) is also shown in Fig. 3. W_{base} and H_{base} represent the width and height of the base layer picture; W_{enh} and H_{enh} represent the width and height of enhancement layer picture respectively. The base layer picture is a sub-sampled version in the region with $W_{extract}$ and $H_{extract}$ totally or partially enclosed in enhancement layer picture, positioned at X_{orig}, Y_{orig} coordinates. $W_{extract}$ / W_{base} and $H_{extract}$ / H_{base} are up-sampling factors between the base layer picture and the extracted region in the enhancement layer picture.

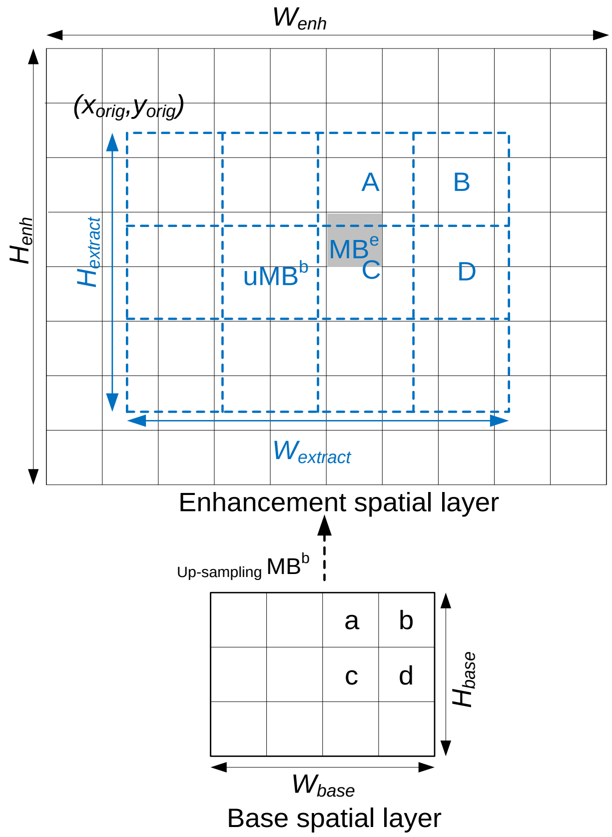


Fig. 3: Base layer MBs are shown with black grids. Enhancement layer MBs are shown with black grids and up-sampled MBs are shown with blue grids.

Figure 3 shows MB at the enhancement layer (MB^e), and up-sampled MB at the base layer (uMB^b) (i.e. block C with a blue grid), in which the projection of the center point of MB^e is coordinated. In extended special scalability, one, two or four up-sampled base layer MBs are required an enhancement layer MB. Let 'A', 'B', 'C' and 'D' (show with a blue grid) be the up-sampled blocks of the base layer blocks 'a', 'b', 'c' and 'd' (show with a black grid) respectively. The up-sampled base layer blocks 'A', 'B' and 'D' are the neighboring blocks of the current up-sampled base layer block uMB^b 'C'.

Table 1: Mode factors weight

Mb mode	Skip_	16x16	16x8	8x16	8x8	8x8ref 0	Intra_4x4
M	0	1	2	3	4	5	6
Block mode	8x8	8x4	4x8	4x4	Skip_		
M	8	9	10	11	0		

Figure 3 shows uMB_n^b (e.g. 'A') is the neighboring up-sampled MBs of uMB^b overlap with the current MB^e. The assign weight of MB mode factor (*M*) of uMB_n^b is shown in Table 1. Area factor (A) is the ratio of the overlapped area of the up-sampled base layer uMB^b with MB^e to the area of MB^e alone. In [8], high layer MBs are classified into four classes: 'corner', 'horizontal', 'vertical' and 'center'. Based on the characteristics of MB^e, MB-mode parameter (*α*) of MB^e is derived.

- For MB^e fits to 'corner', uMB^b has no neighboring MBs. The n of uMB_n^b is assigned as zero; so *α* is derived as:

$$\alpha = \left(A_0 * M_0 \right) \text{ for } MB^e \in corner \quad (2)$$

- For MB^e fits to 'horizontal' or 'vertical', uMB^b has one neighbor. The *n* of uMB_n^b is assigned as 1; α is derived as:

$$\alpha = \left(\frac{A_0 * M_0 + A_n * M_n}{1+n} \right) \text{ for } MB^e \in horizontal / vertical \quad (3)$$

- For MB^e belongs to 'center', uMB_n^b has three neighbors. The *n* of uMB_n^b is assigned as 3; and α is derived as:

$$\alpha = \left(A_0 * M_0 + \sum_3 \frac{A_n * M_n}{1+n} \right) \text{ for } MB^e \in center \quad (4)$$

c. Mode Decision in Background

Different developing techniques are applied for enhancement layer MBs in background and foreground regions based on the MB mode and its motion characteristics. In Table 1, a larger mode factor *M* is assigned for smaller MB mode partitions, and vice versa. In general, the larger the mode factor *M*, the more complex the MB is. For $n = 0$, the mode factor *M* is assigned as '0' as shown in Table 1, and the area factor *A* is assigned as '1'. To decide whether MB^e belongs to background or foreground region, a threshold T_H is set on *α*. The criterion is given as follows:

$$\begin{array}{lll} MB^e \in background & \text{if} & \alpha \le T_H \\ MB^e \in active & \text{if} & \alpha > T_H \end{array} \quad (5)$$

where $T_H = 0.36$, is experimentally determined. For MB^e exists in a background region, MB^e partition type is directly set at 16x16. Consequently, the derivation of MB type is early terminated.

d. Mode Decision in Active/Texture Region

We use a different approach to encode the MB mode in an active region. In [21] authors propose a technique in which two smaller blocks are combined together as long as they share similar motion vectors and the same reference index. In [4] authors disclose, motion vectors of neighboring blocks are similar as long as they have the same feature contents. Exploiting these conclusions, we can combine two neighboring smaller blocks into one large block, even if their MV absolute difference is larger than the MV threshold [21]. Thus, we propose a modified criterion for block combining in active regions:

$$(\alpha * L) \le T_{active} \quad (6)$$

where *L* denotes the activity calculated by (1), *α* is the MB-mode parameter calculated by (2), (3) or (4) based on the content features of MB^e. $T_{active} = 4.0$ is set as an empirical threshold. Since T_{active} is a constant, the similarity between V_1 and V_2 is adapted to *α*, which reflects the motion and texture features of MB^e. For background regions in the frames, *α* is low, and for active motion or rich texture regions, *α* is high.

5. SIMULATION RESULTS

To validate the performance of the proposed method, we use video sequences with various quantization parameters as mentioned in Table 2. The test platform used in this study is an Intel® Core™ i7, 2.5 GHz with 8 GB RAM with Window 10 operating system. The goal of the research is to analyze the complexity and visual quality of proposed methods compared to the similar existing approach. The analysis includes rate distorted due to compression as well as videos distortion from the simulated transmission of packetized streams through error-prone communication channels. We compute the difference in the number of macroblock for each partition and sub partitions between JSVM and proposed scheme for both with and without FMO to evaluate the influence of slice partitions to error concealment.

a. Generate Bit stream

In this work, four video sequences in YUV format are used as mentioned in Table 2. The performance of the proposed scheme is measured in terms PSNR-Y (dB) verses bit rate saving (kbps). In addition, it is also evaluated by the well-accepted [9, 21] measurement i.e. % change in the MB partition and sub-partitions (ΔMB) calculated as follows:

$$\Delta MB = \frac{MB_{pro} - MB_{orig}}{MB_{pro}} \times 100\% \quad (7)$$

where MB_{pro} represents the total number of MB partition and sub-partitions for proposed scheme and MB_{orig} denotes the total number of MB partition and sub-partitions for JSVM.

b. Testing Conditions

To test the proposed technique, we use the error patterns provided by the Video Coding Experts Group, with average packet loss rates (PLRs) of 3%, 5%, 10% and 20% [18]. The resulting SVC bit stream is compatible with H.264 base layer bit stream, decoded using JSVM. The packet losses are concealed using the error concealment mentioned in JSVM [9]. Other simulation conditions are as follows:

- A pixel resolution of 224x192 for the base layer and 336x288 for the enhancement layer (aspect ratio of 2:3) are used.
- It is observed that the base layer experiences a lower PLRs than the enhancement layer.
- The generated bit stream is decoded after passing through a packet loss simulator with 3% PLR for the base layer and 20% PLR for the enhancement layer; a condition that is applicable for simulation over error-prone wireless networks [10].
- Other relevant main configuration parameters are listed in Table 3. The simulation testing consists of FMO used as an error resilience to measure performance with the low complexity.

c. Objective Comparison

The objective performance of the proposed scheme is evaluated with peak signal-to-noise ratio (PSNR) with corresponding bit rate. Moreover, decrease in the number of smaller partition against larger partitions also provides an assessment for the reconstruction quality of codecs between the reference and test videos. We compare the performances for JSVM standard [9, 21], referred to as 'JSVM', with the 'Proposed' algorithm for concealed video sequences at the decoder in an error-prone environment with a PSNR-Y gain (dB) versus the bit rate (Kbps). The objection comparisons are comprising of two parts. Part one evaluate the performance between JSVM and proposed approach without implementing FMO. Whereas, second part evaluate the effect of slice partitioning to error concealment efficiency.

Table 2: Test sequences

Frame size for EL	Sequence	Frame rate	Number of frames
336×288	Bus	15	100
336×288	Foreman	15	100
336×288	Mobile	15	100
336×288	Soccer	15	100

Table 3: Main configuration parameters

Slice Mode: 1
Slice Group Map Type: 2
Base layer QP values: 22, 26, 30, 34, 38
Enhancement layer QP values: 24, 28, 32, 36, 40
Use ESS for enhancement layer: 1
Instantaneous Data Refresh: -1
Base Layer Mode: 1
Group of Picture size: 8

Figure 4 shows rate-distortion (RD) curves for JSVM and proposed scheme, without (a) FMO and (b) with FMO. To validate the performance, we use error prone wireless network testing conditions for the enhancement layer for (a) Bus sequence, (b) Foreman sequence, (c) Mobile sequence and (d) Soccer sequence. It is clearly demonstrated that the curve of the proposed algorithm is almost coincides with the curves of the original JSVM which shows that our approach is comparable or slightly increase the frame quality for PSNR gain and its coding efficiency.

In Table 4, percentage change in the total numbers for each MB mode between JSVM and proposed algorithm without implementing FMO are listed. It is noted that positive values in % indicate increase in number of large MB partition types 16x16, 16x8, 8x16. On the other hand, negative values in % shows decrease in number of MB sub-partition types 8x8, 8x4, 4x8, and 4x4. The decrease in the number of smaller blocks in the proposed scheme with respect to JSVM reflect decrease in computational complexity. From Table 4, we can reveal that proposed scheme is able to increase the number of MBs in large partition type 31.0% on average as compared to JSVM, and reduce the number of MBs in sub-partition types tested on various video sequences.

In Table 5, ΔMB between JSVM and proposed algorithm with FMO implementation are listed. Table 5 shows our proposed scheme is able to increase the number of MBs in 16x16 partition type 31.8% on average as compared to the JSVM, and reduce the number of MBs in sub-partition types on various video sequences. Tables 4 and 5 conclude that the proposed scheme considerably reduce the encoding computation complexity without any degradation for PSNR verses rate distortion.

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

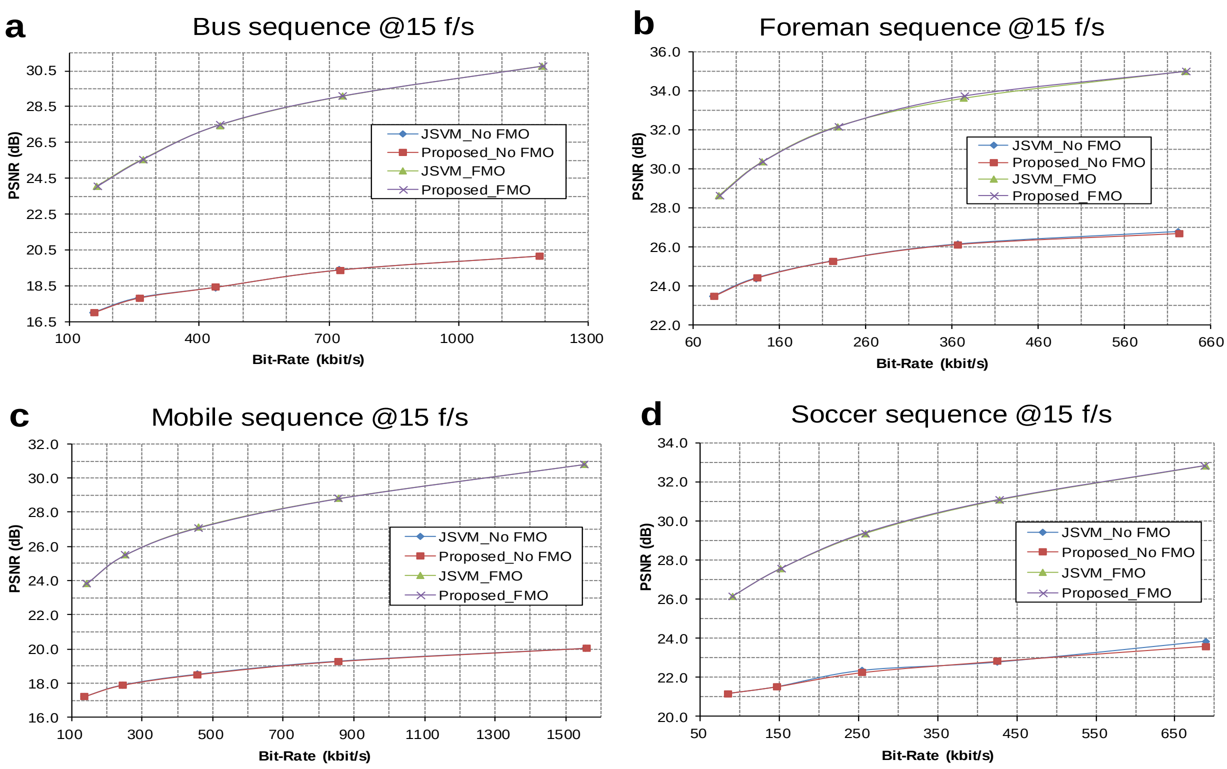


Fig. 4: RD curves for JSVM verses proposed schemes simulated over error-prone wireless networks for (a) bus, (b) foreman, (c) mobile and (d) soccer sequences.

Table 4: Percentage change in number of partition and sub-partition in macroblocks between JSVM and proposed schemes at the decoder for different sequences without FMO.

Sequence	QP_{BL}-QP_{EL}	MB Mode Partitions (%)						
		16x16	*16x8*	*8x16*	*8x8*	*8x4*	*4x8*	*4x4*
Bus	38-40	38.7	-46.3	-49.1	-47.0	-19.1	-10.1	-3.0
Bus	34-36	35.0	-36.9	-42.3	-33.5	-18.8	-10.7	-3.4
Bus	30-32	32.4	-32.4	-34.9	-21.3	-17.4	-12.5	-2.6
Bus	26-28	26.1	-23.6	-26.8	-8.4	-17.5	-14.9	-4.1
Bus	22-24	21.4	-17.6	-21.3	-0.1	-18.7	-16.5	-4.0
Foreman	38-40	42.7	-53.0	-57.2	-64.2	-5.1	-4.0	-1.4
Foreman	34-36	42.8	-44.8	-49.1	-51.3	-6.8	59.3	-1.9
Foreman	30-32	39.3	-37.8	-41.7	-37.3	-7.9	-5.4	-0.9
Foreman	26-28	35.5	-31.1	-35.0	-25.2	-9.8	-6.7	-1.2
Foreman	22-24	28.1	-23.6	-24.7	-10.0	-12.1	-8.0	-1.7
Mobile	38-40	35.4	-54.7	-63.5	-43.9	-29.3	-22.1	-5.9
Mobile	34-36	30.5	-46.2	-54.7	-29.3	-31.1	-22.7	-7.2
Mobile	30-32	26.8	-36.8	-46.7	-16.2	-32.1	-24.1	-6.8
Mobile	26-28	22.3	-27.3	-39.0	-4.2	-33.5	-25.7	-6.6
Mobile	22-24	16.2	-16.2	-30.6	8.2	-35.9	-29.1	-6.2
Soccer	38-40	31.1	-63.2	-57.2	-63.7	-7.4	-4.7	-0.5
Soccer	34-36	29.0	-54.6	-54.5	-53.3	-9.6	-4.7	-1.5
Soccer	30-32	28.6	-52.8	-52.7	-45.4	-10.9	-5.5	-1.4
Soccer	26-28	30.3	-52.6	-48.7	-38.1	-11.4	-6.8	-1.9
Soccer	22-24	27.2	-42.7	-39.5	-27.3	-14.7	-8.6	-2.1
Average		**31.0**	-39.7	-43.5	-30.6	-17.4	-9.2	-3.2

Table 5: Percentage change in number of partition and sub-partition in macroblocks between JSVM and proposed schemes at the decoder for different sequences with FMO.

Sequence	QP_{BL}-QP_{EL}	MB Mode Partitions (%)						
		16x16	*16x8*	*8x16*	*8x8*	*8x4*	*4x8*	*4x4*
Bus	38-40	39.3	-45.2	-48.9	-46.9	-16.9	-12.3	-3.2
Bus	34-36	35.7	-38.2	-41.6	-32.8	-17.3	-10.7	-3.3
Bus	30-32	31.1	-29.6	-35.8	-19.3	-16.5	-12.4	-3.8
Bus	26-28	26.8	-24.5	-27.0	-8.6	-18.2	-13.9	-3.4
Bus	22-24	21.9	-18.6	-21.6	0.8	-18.6	-16.0	-4.1
Foreman	38-40	44.4	-52.7	-56.2	-64.9	-5.1	-3.6	-1.1
Foreman	34-36	45.4	-45.9	-48.9	-51.6	-7.0	-4.9	-1.3
Foreman	30-32	41.1	-37.9	-42.1	-39.1	-7.1	-6.3	-1.7
Foreman	26-28	35.8	-31.8	-34.1	-25.2	-9.9	-6.0	-1.5
Foreman	22-24	29.5	-23.1	-25.3	-11.8	-11.6	-7.3	-1.7
Mobile	38-40	35.6	-52.6	-61.0	-40.8	-29.9	-21.1	-7.2
Mobile	34-36	31.7	-43.4	-54.4	-26.6	-28.9	-24.0	-8.2
Mobile	30-32	27.3	-35.4	-46.6	-13.3	-31.0	-26.0	-6.8
Mobile	26-28	21.6	-25.3	-36.2	-1.2	-30.8	-27.3	-7.7
Mobile	22-24	15.8	-14.6	-29.8	9.5	-34.7	-29.4	-6.9
Soccer	38-40	32.3	-61.3	-56.7	-65.3	-7.6	-4.3	-0.8
Soccer	34-36	31.1	-55.4	-55.1	-56.3	-10.5	-4.9	-1.0
Soccer	30-32	30.6	-53.9	-51.9	-46.2	4.6	-6.5	-1.6
Soccer	26-28	31.3	-51.4	-47.8	-39.6	-13.2	-99.1	-2.1
Soccer	22-24	28.1	-42.1	-38.4	-29.1	-14.6	-8.8	-2.3
Average		**31.8**	-39.2	-43.0	-30.4	-16.2	-17.3	-3.5

d. Subjective Comparison

We also evaluate subjective quality of the decoded videos. Selected frames are decoded with the pYUV viewer tool taken from different video sequences, compare for both JSVM, and proposed algorithms [15]. Figure 5 depict subjective comparisons of frames decoded with (QP_{BL}, QP_{EL}) = (22, 24). These frames are decoded without FMO. Fig. 5(a,b,c) represent 27th frame of Bus sequence. Fig. 5(d,e,f) represent 26th frame of Foreman sequence. Fig. 5(g,h,i) represent 34th frame of Mobile sequence. Fig. 5(j,k,l) represent 26th frame of Soccer sequence. The frames in Fig. 5(a,d,g,j) show the picture decoded with the JSVM algorithm. The frames in Fig. 5(b,e,h,k) show the picture decoded with the proposed algorithm. The frames in Fig. 5(c,f,i,l) shows the difference between the JSVM and proposed algorithms.

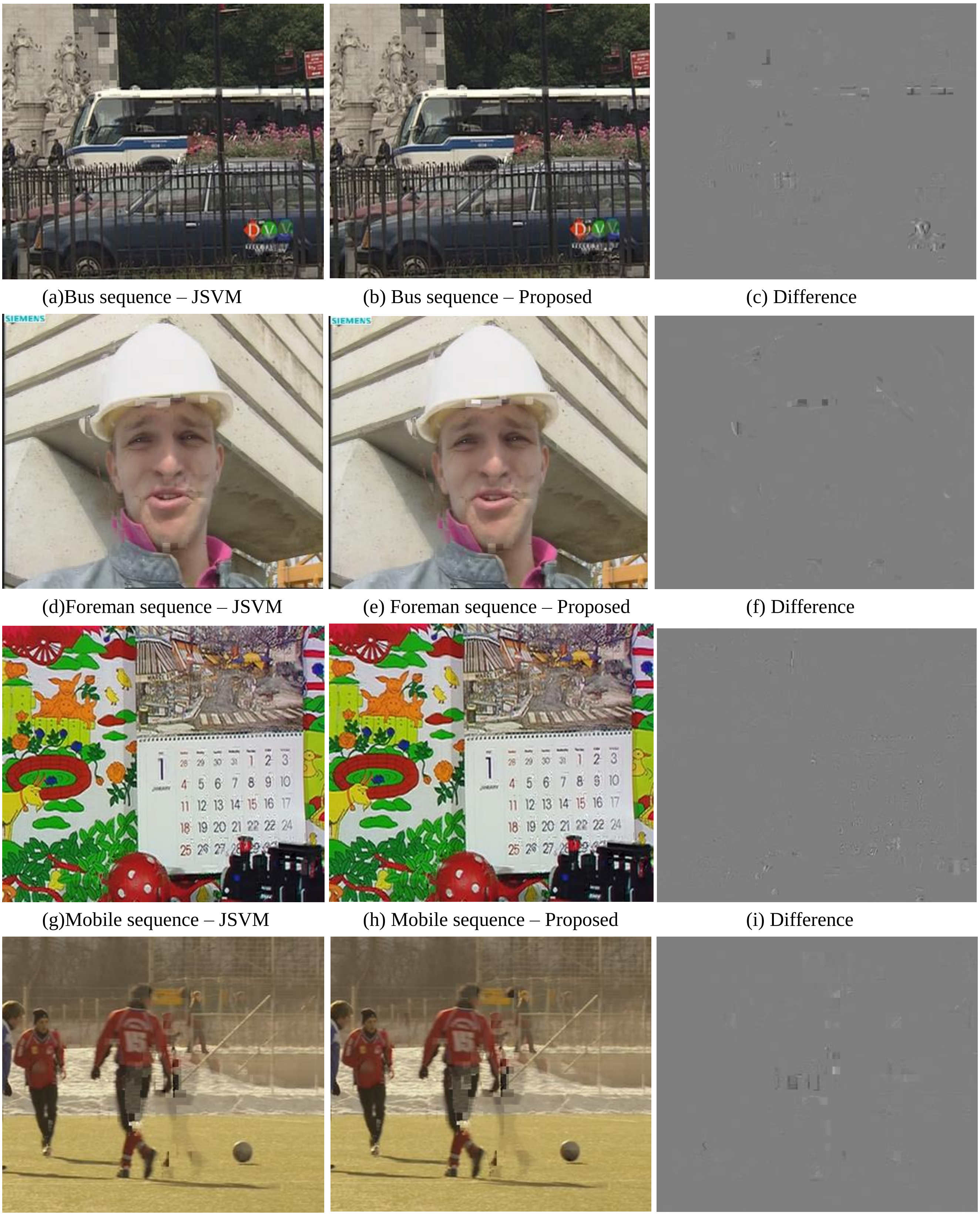
Figure 6 depict subjective comparisons of frames decoded with (QP_{BL}, QP_{EL}) = (22, 24). These frames are decoded with FMO. Fig. 6(a,b,c) represent 24th frame of Bus sequence. Fig. 6(d,e,f) represent 50th frame of Foreman sequence. Fig. 6(g,h,i) represent 52th frame of Mobile sequence. Fig. 6(j,k,l) represent 50th frame of Soccer sequence. The frames in Fig. 6(a,d,g,j) show the picture decoded with the JSVM algorithm. The frames in Fig. 6(b,e,h,k) show the picture decoded with the proposed algorithm. The frames in at Fig. 6(c,f,i,l) shows the difference between the JSVM and proposed algorithms. It is noted that in Figures 5 and 6, some visual contents of decoded 336x288 resolutions, frames are not very clear due to limitation in the fitting size of the manuscript page, especially for subjective quality difference between the JSVM and proposed.

6. CONCLUTIONS

To reduce computation complexity of macroblock mode decision in encoding a frame, we propose an innovative approach for optimal mode decision. Base on the relationship between macroblock mode and motion vector, optimal mode decision is early terminated. We exploit the content features of video for the SVC bit stream to reduce complexity and use FMO as error resilience. Extending our scheme with FMO offers a remarkable performance to reduce the error impact. Experimental results show that our scheme achieves a good performance in reducing complexity with average 30.8% for the same subjective feature in a packet loss environment.

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X



(a)Bus sequence – JSVM (b) Bus sequence – Proposed (c) Difference

(d)Foreman sequence – JSVM (e) Foreman sequence – Proposed (f) Difference

(g)Mobile sequence – JSVM (h) Mobile sequence – Proposed (i) Difference

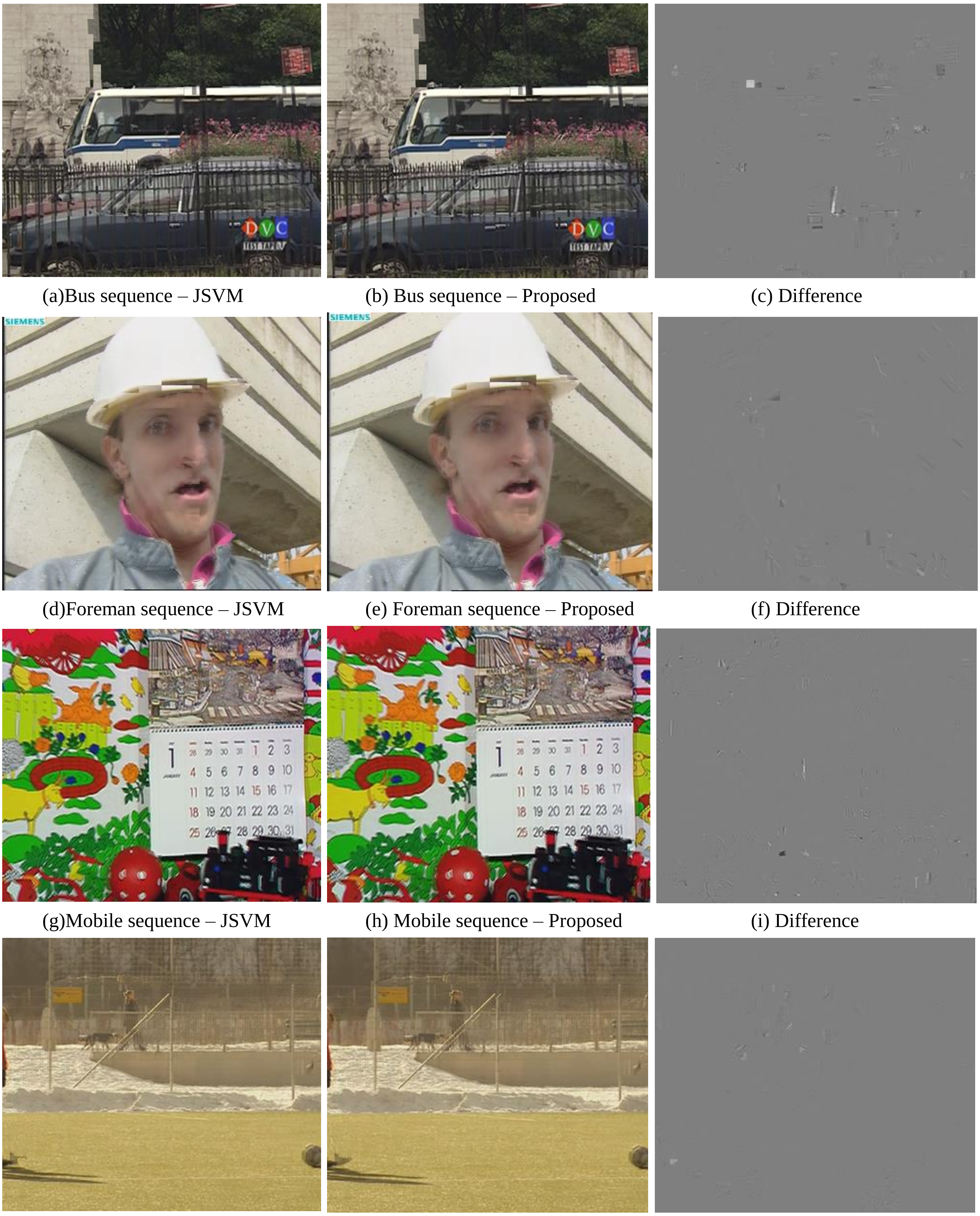
(j)Soccer sequence – JSVM (k) Soccer sequence – Proposed (l) Difference

Fig. 5: Subjective comparisons between JSVM and proposed scheme without FMO for various.

Volume 8, Issue 1
February 2019

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X



(a)Bus sequence – JSVM (b) Bus sequence – Proposed (c) Difference

(d)Foreman sequence – JSVM (e) Foreman sequence – Proposed (f) Difference

(g)Mobile sequence – JSVM (h) Mobile sequence – Proposed (i) Difference

(j)Soccer sequence – JSVM (k) Soccer sequence – Proposed (l) Difference

Fig. 6: Subjective comparisons between JSVM and proposed scheme with FMO for various.

REFERENCES

[1] A.I. Ali, S. Moirn, M. Fleury, M. Ganbari, “Data Partitioning Technique for Improved Video Prioritization,” MDPI Computer Journal, 6 (3), Article 23, 2017.

[2] A.M. Buhari, H.C. Ling, V.M. Baskaran, K.S. Wong, “Low Complexity Watermarking Scheme for Scalable Video Coding,” ICCE-TW, Taiwan, pp 5-6, 2016.

[3] A. Ziviani, B.W. Wolfinger, J.F. Rezende, O. Duarte, S. Fdida, “Joint adoption of QoS schemes for MPEG streams,” Multimedia Tools Appl 26(1):59–80, 2005.

[4] D. Wu, F. Pan, K.P. Lim, S. Wu, Z.G. Li, X. Lin, S. Rahardja, C.C. Ko, “Fast intermode decision in H.264/AVC video coding,” IEEE Trans Circ Syst Video Technol, 15(6):953–958, 2005.

[5] G. Dan, H. Ofer H, “Efficient Region-of-Interest Scalable Video Coding with Adaptive Bit-Rate Control,” Advances in Multimedia - Hindawi, vol 2013, Article 281593, 2013.

[6] H. Li, Z.G. Li, C. Wen, “Fast mode decision algorithm for inter frame coding in fully scalable video coding,” IEEE Trans Circ Syst Video Technol 16(7):889–895, 2006.

[7] J.M. Boyce, R.D. Gaglianello, “Packet loss effects on MPEG video sent over the public internet,” Proc. of the ACM Multimedia, Bristol, England, 1998.

[8] J. Reichel, “Joint scalable video model JSVM-8,” Proc. ISO/IEC JTC1/SC29/WG11/JVT-U202, Hangzhou, China, 2006.

[9] JSVM Software and Manual Available (Online): JVT.H.264/SVC reference software (JSVM) and manual. Available: https://www.hhi.fraunhofer.de/en/departments/vca/research-groups/image-video-coding/research-topics/svc-extension-of-h264avc/jsvm-reference-software.html

[10] K. Seshadrinathan, R. Soundararajan, A.C. Bovik, L.K. Cormack, “Study of subjective and objective quality assessment of video,” IEEE Trans Image Process 19(6):1427–1441, 2010.

[11] L. Zhou, X. Wang, W. Tu, G. Mutean, B. Geller, “Distributed scheduling scheme for video streaming over multi-channel multi-radio multi-hop wireless networks,” IEEE J Sel Areas Commun JSAC 28(3):409–419, 2010.

[12] M. Asif, M. Bin Ahmed, I.A. Taj, M. Tahir, “A Generalized Multi-Layer Framework for Video Coding to Select Prediction Parameters,” IEEE Access, vol (6): 25277-25291, 2018.

[13] M. Koziri, P. Papadopoulos, N. Tziritas, A. Dadaliaris et al., “Slice-Based Parallelization in HEVC Encoding: Realizing the Potential through Efficient Load Balancing,” Intl Workshop on MMSP, Canada, 2016.

[14] M. Shoaib, U. Ahmed, A. Alamri, "Multimedia Framework to Support eHealth Application," Multimedia Tools and Applications, Springer, vol (73), pp: 2081-2101, Dec. 2014

[15] PYUV: raw video sequence player. [Online]. Available: http://dsplab.diei.unipg.it/pyuv_raw_video_sequence_player_original_one

[16] S.H. Yang, C.W. Chang, C.C. Chan, “An object-based error concealment technique for H.264 coded video,” Springer Multimedia Tool and Application, 74(23): 10785-10800, 2015.

[17] S. Paluri, P. Cosmon, J. D. Matyjas, S. Kumar, “A low complexity model for predicting slice loss distortion for prioritizing H.264/AVC video,” MTAP, 75(2): 961-985, 2016.

[18] S. Wenger, “Error patterns for Internet experiments,” TU Berlin, Doc. VCEG-Q15-I-16r1, New Jersey, 1999.

[19] T.N.T. Huong, T.V. Hu, “An Improved Technique using FMO for Error Propagation Resilience in H.264 Video Coding,” Proc. in NICS, pp 225-228, Vietnam, 2015.

[20] W. El-Shafai, S. El-Rabaie, M.M. El-Halawany, F.E. Abd El-Samie, “Proposed dynamic error control techniques for QoS improvement of wireless 3D video transmission,” Intl J of comm. System, 31(10), Jul 2018.

[21] X. Wang, J. Ridge, “CE2 report: Improvement of macroblock mode prediction in ESS,” Proc. ISO/IEC JTC1/SC29/WG11/JVT-W030, California, USA, 2007.

[22] Y. Cui, W. Ren, Z. Deng, “Fast Mode Decision for HD Scalable Video Coding via Statistical Content Analysis,” Intl. J Pattern Recognition and AI, 32(8), 2018.

AUTHOR PROFILES

Muhammad Shoaib received his Ph.D. degree (2010) in Communication and Information System from Beijing University of Posts and Telecommunications, China. He received his M.Eng. (2005) and B.Eng. (1995) from NED University of Engineering and Technology, Karachi. His areas of research include video compression techniques, multi-layer video coding and IP packet based network, infrastructure and security. He worked as a Senior Manager (IP Operations) in Pakistan Telecommunication Company Limited, Pakistan from 2001-2011. He also worked as a Maintenance Engineer in R. M. International from 1996-2001. Currently, he is working as an Assistant Professor in the College of Computer and Information Sciences, King Saud University.