Research Article

Distortion-Based Slice Level Prioritization for Real-Time Video over QoS-Enabled Wireless Networks

Ismail A. Ali, Martin Fleury, and Mohammed Ghanbari

School of Computer Science and Electronic Engineering, University of Essex, Colchester CO4 3SQ, UK

Correspondence should be addressed to Martin Fleury, fleum@essex.ac.uk

Received 29 April 2012; Revised 22 June 2012; Accepted 22 June 2012

Academic Editor: Martin Reisslein

Copyright © 2012 Ismail A. Ali et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper presents a prioritization scheme based on an analysis of the impact on objective video quality when dropping individual slices from coded video streams. It is shown that giving higher-priority classified packets preference in accessing the wireless media results in considerable quality gain (up to 3 dB in tests) over the case when no prioritization is applied. The proposed scheme is demonstrated for an IEEE 802.11e quality-of-service- (QoS-) enabled wireless LAN. Though more complex prioritization systems are possible, the proposed scheme is crafted for mobile interactive or user-to-user video services and is simply implemented within the Main or the Baseline profiles of an H.264 codec.

1. Introduction

There have recently emerged two forms of video streaming to mobile devices. The first, HTTP adaptive streaming [1], employing reliable TCP transport, has no need to protect the video stream against channel errors but is subject to delays. These delays mainly arise from the repeated transmissions that TCP imposes whenever packets are lost. Additionally, delay may occur due to the pull-based nature of the service. Therefore, though suitable for some forms of one-way commercial streaming, HTTP adaptive streaming is unsuitable for interactive services such as video conferencing. It is also unsuitable for mobile user-to-user streaming, because of the need to create multiple copies of the same video at different resolutions and set up a complex management structure to allow client access to an appropriate stream. Therefore, a second native form of streaming is necessary for delay- or storage-intolerant video streaming, and it is this form of streaming that is the subject of this paper. In this form of streaming [2], video is pushed from the server without the need for a feedback channel to make continual client requests. The Real-time Transport Protocol (RTP) with underlying Internet Protocol (IP)/User Datagram Protocol (UDP) for network routing and transport updates the client-side decoder with synchronization information. If MPEG-2

Transport Stream (TS) packets are multiplexed within each RTP packet, then audio can accompany video in a single packet stream. Adaptive bitrate adjustments (through scalable coding or transcoding) can occur, based on performance metrics carried by Real-time Transport Control Protocol (RTCP) packets, and pseudo-VCR functionality, if needed, is available through the Real-time Streaming Protocol (RTSP).

When mobile video streaming in native mode with IP/UDP/RTP packetization, there is a need to avoid periodic increased delay due to less efficient intracoded I-pictures [3] at the start of each Group of Pictures (GoP). One of the advantages of native streaming, is that an IPPP... picture structure can be adopted on wireless networks. This means that there is just one I-picture at the start of a stream, followed by a continuous stream of predictively coded P-pictures. In contrast, in HTTP adaptive streaming each video chunk (i.e., a GoP) must have a point of random access at the start of each chunk [1], for example, an I-picture. However, using a continuous sequence of predominantly intercoded P-pictures runs the risk upon packet loss of spatiotemporal propagation of errors. To counteract this problem, an H.264/AVC codec permits the inclusion of intracoded macroblocks (MBs) within the P-slices making up a compressed video frame. These MBs can be placed naturally by the encoder, if, for example, no suitable

predictive reference exists from an occluded region. However, they can also be forcibly inserted as a form of nonperiodic intrarefresh. Notice that nonperiodic intrarefresh still allows random access to take place (if needed), as discussed further in Section 2. There are various forms of nonperiodic intrarefresh including: random placement of intracoded MBs up to a given number within each picture [4]; as part of an evolving isolated region [5]; as a line of intracoded MBs that cycles in position over a sequence of pictures [6]. The issue of which of these to choose is an interesting debate but as insertion of a cyclic intracoded line certainly does result in a complete refresh despite corrupted data [6], this paper assumes that this simply implemented mechanism is used.

The introduction of a cyclic intracoded line results in unequal error sensitivity within individual slices of video pictures, as a result of the additional intracoded MBs. To exploit this, individual slice-bearing packets within each video frame can be dropped and the effect on the objective video quality (PSNR) of the whole frame measured. Packets resulting in the highest video quality penalty (when dropped) can then be given the highest priority, while the ones introducing the least penalty are given the lowest priority level. In this work, we apply this proposed prioritization scheme to quality-of-service (QoS) enabled wireless LAN delivery. Specifically, we employ IEEE 802.11e [7], a QoS amendment that adds four queuing prioritization levels to the access of standard IEEE 802.11 (WiFi) networks. As compared to our preliminary work in [8], the proposed scheme does not require any modification to the standard H.264/Advanced Video Coding (AVC) slicing. Instead, the scheme can be applied to any preencoded video stream provided there are a reasonable number of slices per frame. Like the work in [8], this paper's scheme also involves just one video frame delay, as the packets of the video frame to be analyzed should be available to perform the distortion analysis. It ought to be mentioned that the original raw video is not required for this analysis, as the decoded frame without drops can act as a reference for the PSNR calculations. Other work by the authors explored alternative ways to prioritize data in the presence of a cyclic intracoded line or examined the impact of a cyclic intracoded line. In [9], the line split the frame into three unequal regions and a scheme was presented that ensured the regions' areas were properly assigned to slices. Then in [10], regions were allowed to wrap around a frame's boundaries so that region sizes could be equalized. The latter scheme was found to be preferable to a simple geometrical division. Finally, in [11], it was shown that compared to employing periodic intrarefresh with an intracoded I-frame, insertion of a cyclic intracoded line was especially favourable for less active video sequences. However, the work in [11] made no contribution to the issue of prioritization. In fact, both the other two schemes [9, 10] differed from the present proposal because prioritization of slices was determined by the position of the cyclic intracoded line within each video frame rather than through distortion analysis.

The remainder of this paper is organized as follows. Section 2 considers the context to the experiments in this paper. Section 3 details our prioritization scheme. The

scheme is then evaluated in Section 4, which contains generic and network-specific results. Finally, Section 5 makes some concluding remarks.

2. Context

This Section explains the context to this work, in the sense that it explains what forms of intracoded intrarefresh are possible and why a popular feature or tool of H.264/AVC, Flexible Macroblock Ordering (FMO) [3] was not used in conjunction with prioritization. Furthermore, it reviews research on how best to prioritize video data, when mapping the priority classes to a wireless LAN QoS structure.

For mobile applications with limited processing power and constrained bandwidths, the omission of both bi-predictively coded B and periodic I-pictures is advantageous. Due to the risk of burst errors within wireless channels through entry of the mobile receiver into a deep fade, there is a risk of the loss of many of the I-picture packets. This can render useless the remainder of a GoP as, due to predictive coding, all subsequent pictures in the GoP employ the I-picture as a predictive coding reference anchor. As remarked in Section 1, it is still possible to provide random access to a video stream by what is known as gradual decoding (or decoder) refresh (GDR) [5], without the need for periodic I-pictures. Thus, prioritizing video packets according to their picture type (I, B, or P) is not convenient for mobile applications. Currently a prioritization scheme based on the three data partitions available under H.264/AVC [12] is also not convenient in practice. Data partitioning is only available in the H.264/AVC the Extended profile whereas mobile devices tend to rely on hardware implementations of the codec in the Baseline profile. In fact, data partitioning is not implemented in many software implementations of the codec such as QT, Nero, and LEAD, to name a few at random.

Though forced random insertion of intracoded MBs within a video stream is possible [4], this arrangement does not necessarily permit GDR, as it does not account for the direction of motion within a sequence. However, it should be noticed that in [13] the problem of duplication of MBs in random insertion was avoided and MBs to be intracoded were selected according to whether they could be error concealed or not. In GDR, in the presence of packet loss, the stream is reset gradually to a clean state, from which future predictions can be made. However, forced intrarefresh with an MB line can permit GDR. If there are N lines per picture then the worst-case GDR should take place within $2N-1$ pictures [6]. Periodic intracoded pictures do permit more flexible random access, as might be used to support pseudo-VCR functionality. However, for wireless viewing of typically short clips VCR functionality is not uppermost in the mind of the viewer. Besides, the end-to-end packet delay is also reduced by the dispersed insertion of intrarefresh MBs, as periodic intracoded frames result in an influx of packets into transmission buffers, causing the waiting time to increase. All the same, one should note that I-pictures or GDR allow viewers to join a live stream at a point other than at the start of a broadcast, as might well occur during a video

conference. Additional I-pictures might also be used (if scene cut detection is in place) to reset a stream after a change of scene.

We have utilized distortion analysis at the slice level. It is also possible [14] to undertake distortion analysis at the MB level. However, analysis at an individual MB level significantly increases the computational complexity arising from the required video content analysis. Moreover, methods exploiting “explicit” FMO also increase the bitrate and the degree of interpacket dependency due to the need to include additional packets with the updated MB maps for every picture. Other adaptive schemes such as in [15] have relied on feedback from the receiver. Once the decoder detects an error, it informs the encoder, which transmits intracoded MBs to halt any error propagation. However, this procedure is unsuitable for conversational video services such as videophone or mobile teleconferencing. In fact, though an interesting case for evolving isolated regions as a form of GDR is made in [5], the irregular nature of the regions formed in which all predictive reference is internal means that explicit FMO must be used.

Because the position of a cyclic intracoded line of MBs is easily predicted from one picture to the next, it does not require the overhead of an MB map. Consequently, the work in this paper does not use FMO explicit mode. In fact, as previously remarked, it does not use FMO at all and, hence, avoids the overhead associated with FMO [16]. This is also convenient, as many content creation tools such as QuickTime Pro do not allow the use of FMO and the H.264/AVC Constrained Baseline and Main profiles do not support FMO.

Previous experiments by the authors of [17] have involved prioritization through layered coding with the Scalable Video Coding (SVC) extension of H.264. Again in practice, this scheme currently runs into an implementation problem, as apparently hardware implementations of H.264/SVC do not exist, restricting the type of mobile device that can be used. However, cross-layer signalling is available in the H.264/SVC Network Abstraction Layer Unit (NALU) header as a 6-bit priority id field. Others have also experimented with mapping SVC layers to IEEE 802.11e priority classes. For example, the authors of [18] present a packet significance level algorithm for placing packets in an appropriate priority queue. The authors show that their algorithm is preferable to a static allocation of base layer and enhancement layers across the priority queues.

The possibility of mapping priority classes to the wireless QoS structure of IEEE 802.11e [7] has been explored by a good number of research papers over the years since IEEE 802.11e’s development from late 2005. IEEE 802.11e itself is further considered in Section 3.2. In [19], prioritization was managed at a frame level, rather than the subframe scheme in this paper. Prioritization was dynamic in the sense that it depended both on the frame type (I-, B-, or P-frame) and the queue occupation of the normal video queue. A problem with this approach is that B-frames are not present in the Baseline profile of H.264/AVC, which is intended to limit energy consumption on mobile devices. In fact, the intracoded line technique also makes it possible to dispense

with all but the first I-frame. The cross-layer signalling between frame type and IEEE 802.11e priority queue is achieved through marking the Type of Service (TOS) field in the IP header (now replaced by the 6-bit Differentiated Services Code Point (DSCP) field). As the video queue fills up a Random Early Detection (RED) algorithm allocates packets to alternative priority queues according to their frame type priority. However, if header compression is employed then packets may not be queued with their IP headers intact, impeding cross-layer signalling in a wireless network. It must also be remarked that IP headers are not available at the application layer, whereas the codec generated headers described in Section 3.2 are accessible.

In [20], packet classification is performed at the subframe- or slice-level. However, the authors employ the same method of prioritization as in [12] that is through data-partitioned video encoding. A practical problem with that approach is that data-partitioning is only present in the Extended profile of H.264/AVC. The Gilbert-Elliott model for “bursty” channels is employed to govern dynamic allocation of packets to queues. However, it is unclear how a statistical channel model can predict actual channel conditions at any one point in time, though clearly a simulation will confirm the results. For cross-layer signalling the authors use a similar method to the one described in this paper, that is through the H.264/AVC generated header. In contrast, prioritization by packet deadline is an interesting idea of [21], which has apparently not been presented before in this context. A packet scheduler tries to ensure that each packet is transmitted before its display deadline expires. An extension would be to transmit before a packet’s decode deadline expires, as this may be a longer deadline. How cross-layer signalling would be used to identify deadlines was not specified but presumably Real-time Transport Protocol (RTP) headers could be inspected.

3. Proposed Scheme

This Section outlines the prioritization scheme itself; a sample application to wireless QoS (as might be used at a hotspot or within a home network); some video configuration issues.

3.1. Prioritization Scheme. Using a horizontal (or vertical) sliding intrarefresh line, Figure 1, reduces spatiotemporal error propagation arising from packet loss. However, introducing an intracoded MB line within a temporally predicted picture represents a significant percentage of the bits devoted to compressing the whole picture. Nevertheless, a packet containing data from an intra-code MB line represents a small portion of the image area. Therefore, only a small potential quality penalty arises from the loss of a packet containing intracoded MBs due to the small image area affected. Therefore, those packets containing some or all data originating from the intracoded MB line are of lower priority than other packets, as far as the effect on the reconstruction video quality at the decoder is concerned.

In the prioritization assignment algorithm, the compressed data is broken up into fixed-size slices. A slice [18]

FIGURE 1: Cyclic intracoded MB line technique for the *Paris* test sequence, showing successive MB lines in lighter shading, with some slice boundaries also shown.

```

pktsBuffered ← 0
loop
  receive a packet
  if (new frame received = true) then
    decode the frame (to be used as a reference for PSNR calculations)
    n = 0
    while n < pktsBuffered do
      move packet n from buffer1 to buffer2
      calculate PSNR for the remaining packets in buffer1, put the n : PSNR result in list1
      move back packet n from buffer2 to buffer1
      n ← n + 1
    end while
    sort list1 in ascending order according to PSNR field
    remove ⌊pktsBuffered/3⌋ elements from list1 and assign Pri2 to corresponding packets in buffer1 (set NRI to "10")
    remove next ⌊pktsBuffered - ⌊pktsBuffered/3⌋/2⌋ elements from list1 and assign Pri2 to corresponding packets in buffer1
    (set NRI to "01")
    assign Pri0 to packets in buffer1 corresponding to the remaining elements in list1 (set NRI to "00")
    flush buffer1
    pktsBuffered ← 0
  else
    add packet to FIFO buffer1
    pktsBuffered ← pktsBuffered + 1
  end if
end loop

```

ALGORITHM 1: Algorithm to assign priorities to slices using slice distortion, $\lfloor \dots \rfloor$ denotes the floor operation.

is a self-contained decoding unit with a header containing resynchronization information. In the test implementation, slices are formed by selecting MBs in raster-scan order, as shown in Figure 1. Algorithm 1 describes the algorithms employed. (In the Algorithm, annotation concerning NRI for cross-layer signalling is explained in Section 3.2.) In an implementation, a maximum slice size can be fixed, as occurs in our evaluation (refer to Section 3.3). All slices are at the maximum except possibly for the last one to be formed. However, treatment of MBs to slice assignment is implementation dependant. For each slice-bearing packet, the impact on the reconstruction PSNR is tested by removing that packet's data from that frame's compressed bitstream and then finding the PSNR relative to the decoded frame (refer back to Section 1). This process is repeated for each slice within the frame. The resulting PSNRs are then sorted into rank order so that priority classification classes can be formed. In the test implementation, there are just three

priority classes to match suitable classes within IEEE 802.11e. Thus, once the slices are in rank order the top third of the slices are assigned to the highest priority, the middle third to the intermediate priority, and the lowest third to the lowest priority. If the number of slices was not an exact multiple of three then additional slices are assigned to the lower priorities in turn. For example, if there are two extra slices they can be individually assigned to the two lower priority classes. Other possibilities exist but these are not critical to evaluation of the scheme. The mapping to IEEE 802.11e priority classes is now described.

3.2. IEEE 802.11e EDCA and Cross-Layer Signalling. We have employed IEEE 802.11e [7] to exploit the proposed prioritization scheme. IEEE 802.11e Enhanced Distributed Channel Access (EDCA) adds QoS support to legacy IEEE 802.11 wireless networks by introducing four Access Categories

(ACs): AC0, AC1, AC2, and AC3 for Background (BK), Best-Effort (BE), Video (Vi), and Voice (Vo), respectively, in order of increasing priority. Each AC has its associated queue (set to 40 variable-sized packets in tests) with entry to the queue defined by a mapping function. Should several packets emerge simultaneously from the queues then contention is resolved by the virtual collision handler before a transmission attempt.

To better deliver priority-classified packets and exploit the unequal error sensitivity, this paper proposes mapping different priority packets across the IEEE 802.11e EDCA ACs as an effective alternative to assigning the complete stream to AC2. Priority 2 packets are mapped to AC2, the default access category for video. The least important priority 0 packets are mapped to AC0 while priority 1 packets are mapped to AC1. Each AC has different Distributed Coordination Function (DCF) parameters for the Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA) back-off mechanism. In tests (Section 4), the default IEEE 802.11e Medium Access Control (MAC) parameter values for the IEEE 802.11b radio were employed but an extension is to tune these parameters to set a desired quality/delay tradeoff.

Figure 2 shows the cross-layer signalling architecture adopted in this article to signal the priorities to the MAC layer. Briefly, H.264/AVC Network Abstraction Layer (NAL) units (virtual packets output by an H.264/AVC encoder) contain the priority information by virtue of the two Nal_Ref_Idc (NRI) bits within a NAL unit header. At the application layer (specifically, the NAL sublayer), the NRI bits of NALUs are changed according to the importance of each NALU as determined by the distortion analysis. At the MAC layer, packets are classified and mapped to the IEEE 802.11e ACs based on the NRI bits in NALU headers.

3.3. Video Configuration. The H.264/AVC Main profile encoded different Common Intermediate Format (CIF) frames (352×288 pixels/frame) test sequences at 30 fps with 4:2:0 chroma subsampling. Notice that the Main profile does not include FMO, which precludes the selection of individual MBs to form priority classes (refer back to Section 2). The same applies to the Constrained Baseline Profile, which is suitable for video conferencing applications. The Baseline Profile does include FMO and is suitable for mobile streaming applications, though no strong relationship exists between profiles and target applications [19]. However, that profile does not support more complex forms of coding such as data partitioning, which, as previously mentioned, otherwise might be employed for the purpose of packet prioritization.

The use of an IPPPP... coding structure in this study reduces the decoding complexity on mobile devices that would occur were bipredictive B-frames to be included. Recall that the cyclic intra-code line of MBs was introduced to mitigate the risk of spatiotemporal error propagation across successive P-pictures. Context Adaptive Variable Length Coding (CAVLC) entropy coding (CAVLC is applied to the quantized transform coefficients, while Universal VLC

FIGURE 2: Cross-layer priority signalling to the IEEE 802.11e access categories.

(UVLC) is applied to other syntactic elements. The alternative, Context Adaptive Binary Adaptive Coding (CABAC) results in a 10–15% gain in coding efficiency, but cannot be implemented as CAVLC can be through switchable look-up-tables. Consequently, CABAC is omitted from the Baseline profile because of its complexity [22] and single-frame reference was employed, with both settings selected to reduce computation on mobile devices. Motion-copy error concealment [23] was set at the decoder as an effective means of concealment except for sequences with very rapid motion. Streams were coded with a CBR target of 1 Mbps. Packet size payloads and, hence, slice sizes were limited to a maximum of 500 B to reduce the risk of error from long packets and network fragmentation.

The motion estimation search range was also set to eight to reduce computation. This setting has an effect on potential contamination from an unclear area to an already cleansed area of a picture sequence. Notice that the H.264/AVC Constrained Intraprediction (CIP) flag was also set, as otherwise reference to intercoded MBs is possible, which negates the ability of the insertion of a cyclic MB line to arrest the propagation of spatiotemporal errors. Some loss of coding efficiency arises from setting CIP but this is inevitable, as some form of intrarefresh cannot be avoided. However, if random placement of forced intracoded MBs took place then the need for CIP would result in a greater deterioration in video quality. This is because the MBs of an intracoded MB line are adjacent and consequently well correlated with each other. However, randomly placed MBs may be far apart and, hence, not well correlated. The result is that spatial reference will not be an effective form of prediction, even if the search range could extend far enough.

TABLE 1: Mean Y-PSNR gain (loss) when dropping different priority packets, as compared to dropping randomly over a range of test sequences.

		Δ PSNR (dB)				
Loss rate (%):		2	4	6	8	10
Akiyo	Drop Pri 0	1.73	2.62	3.3	3.94	4.21
	Drop Pri 1	0.61	0.69	0.9	1.15	1.27
	Drop Pri 2	-1.52	-2.33	-2.52	-2.37	-2.41
Mobile	Drop Pri 0	1.85	2.76	3.12	3.31	3.57
	Drop Pri 1	0.53	0.94	1.16	1.42	1.43
	Drop Pri 2	-2.1	-2.34	-2.22	-2.12	-2
Soccer	Drop Pri 0	1.77	2.63	3.03	3.25	3.38
	Drop Pri 1	0.26	0.61	0.72	0.85	0.91
	Drop Pri 2	-1.69	-2.17	-2.24	-2.17	-2.19
Foreman	Drop Pri 0	1.99	3.03	3.31	3.45	3.67
	Drop Pri 1	0.61	0.88	0.93	0.87	1.09
	Drop Pri 2	-1.68	-2.09	-2.19	-2.18	-2.02
Football	Drop Pri 0	0.69	1.13	1.47	1.42	1.42
	Drop Pri 1	0.26	0.63	0.82	0.88	0.82
	Drop Pri 2	-1.07	-1.43	-1.27	-1.27	-1.38
Highway	Drop Pri 0	2.01	3.18	3.88	4.5	4.54
	Drop Pri 1	0.72	1.18	1.37	1.49	1.37
	Drop Pri 2	-1.95	-2.44	-2.71	-2.67	-2.78

4. Evaluation

In this Section, we test the generic behavior of the scheme before considering an example IEEE 802.11e WLAN simulation.

4.1. Uniform Drop Tests. The test sequences of *Paris* and *Stefan* were employed. The former is typical of TV studio clips that can be appreciated on a mobile device [24] for the audio as well as the video and the latter has high temporal coding complexity. In Figures 3 and 4, the impact of dropping prioritized packets is compared to random drops. (Error bars represent one standard deviation in the plots herein.)

As is evident, there is a considerable gain from only dropping those packets classified into the low priority (Drop-pri-0) up to the given percentage on the horizontal axis. Dropping packets solely from priority one class (Drop-pri-1) is somewhat better than random drops but if only high-priority packets are dropped (Drop-pri-2) there is a serious deterioration in video quality. The effect of increased coding complexity on *Stefan*, Figure 4, is to decrease the mean video quality level for the given bit-budget without affecting the overall pattern. Table 1 confirms this behavior for a variety of video content type. The gain seems greatest at higher packet loss rates for relatively static sequences.

4.2. Network Simulations. To show the advantage of the proposed scheme, the application scenario in Figure 5 was simulated with the well-known ns-2 network simulator. Each plot in the following graphs is the result of around 1000

FIGURE 3: Mean Y-PSNR for the *Paris* video sequence when dropping packets from different priorities.

FIGURE 4: Mean Y-PSNR for the *Stefan* video sequence when dropping packets with different priorities.

runs after performing statistical analysis to find the mean and standard deviation at the given loss rate percentages. The scenario consists of a tablet computer receiving video streamed from a streaming server plugged-in at the wireless home router. There is also a smartphone sending Voice-over-IP (VOIP) traffic to the Internet and a laptop computer competing for bandwidth, while performing web browsing. In situations as in Figure 5, IEEE 802.11e was developed to offer prioritized access to delay-sensitive applications by prioritizing traffic over higher priority queues in order to reduce packet drops through buffer overflow. Table 2 details the traffic sources feeding into the home router's output buffer.

In Figures 6 and 7, the impact of the congesting traffic (along with self-congestion from the streaming video) is

TABLE 2: Traffic in the scenario of Figure 6.

Traffic type	Source	Destination	IEEE 802.11e AC	Protocol	Data rate (kbps)
Video	Streaming server	TV	2 (2, 0, 1 when mapping)	UDP	1000
CBR	Phone	Internet	3	UDP	16–160
CBR	Laptop	Internet	1	UDP	200
CBR	Laptop	Internet	0	TCP	200

compared with the effect of mapping the entire video stream to the IEEE 802.11e designated AC2 class. At its worst, for a packet loss rate of 10%, there is over 3 dB gain from the proposed mapping for streaming the *Paris* studio scene. Again, the impact is a little reduced for the more active *Stefan* sequence but still well-worth applying. Table 3 presents the PSNR gain when using the proposed mapping scheme over assigning the video stream’s packets to AC2 for a range of test sequences.

FIGURE 7: Mean Y-PSNR for the *Stefan* video sequence for the simulated network scenario.

4.3. *Discussion.* Others have also presented performance evaluations that have highlighted the advantages of employing prioritization mappings. The work in [25] is a comparison of mapping schemes for IEEE 802.11e. The study [25] employed Standard Definition television frames, rather than the CIF frames employed by other studies more concerned with common mobile device screen resolutions. The standard mapping to AC2 was compared to one that distinguished between I-, P-, and B-frames and another that grouped I- and P-frames into the same priority category. As has already been remarked in Section 2, this type of frame-level slice classification lacks the flexibility of subframe or slice classifications. However, if mobile devices are not used then retaining a traditional slicing structure for broadcast video may be required for compatibility reasons. In fact, if B-frames are available to be dropped then the authors found that a sudden drop in quality after packet losses could be avoided. This was not the case under the default video mapping. The authors also observed that less active sequences suffered less from impairments after packet drops. A related observation was made by the authors of this paper in [11], which, as previously remarked, indicated that a measure of temporal activity such as the number of nonzero motion vectors, could guide whether periodic I-frames were employed or not.

TABLE 3: Mean Y-PSNR gain when applying the proposed mapping scheme, as compared to assigning the video stream to AC2 for a range of test sequences.

Loss rate	PSNR gain (dB)									
	0.5	1.5	2.5	3.5	4.5	5.5	6.5	7.5	8.5	9.5
Soccer	0.29	0.85	1.20	1.96	1.59	2.21	2.68	2.83	3.13	3.25
Foreman	0.38	1.02	1.58	1.99	2.24	2.39	2.54	2.89	3.01	3.11
Highway	0.21	0.73	1.04	1.38	1.84	2.02	2.46	2.67	2.71	2.86
Football	0.00	0.10	0.08	0.09	0.31	0.29	0.44	0.50	0.69	0.45

to 20 nodes generated traffic within the IEEE 802.11 network. This work, which is contemporary with that reported in [12], confirms the advantage of this type of mapping. The authors of [27] also employed priority classification based on data partitioning but tested the results in an indoor wireless testbed with three laptops as receivers. Best-effort TCP traffic was also present. Again, the study confirmed the advantages of mapping across some of the access categories but this time in a situation with real-world access contention.

5. Conclusion

The main intent of this paper is to demonstrate a simple procedure for prioritization of video packets that can be implemented across H.264 profiles without the need for flexible macroblock ordering. Slice-based distortion analysis reduces the implementation overhead compared to individual MB-based optimizations. The paper has demonstrated considerable gain in video quality from the resulting prioritization classification when modelled in a home network. Because delay sensitive applications were targeted, the scheme tolerates a single video frame delay, during which slice-level distortion analysis is performed. It is also possible to extend the scheme to slice distortion analysis across multiple frames but this clearly will incur more delay implications. The emerging High Efficiency Video Coding (HEVC) standard considers ways to improve implementation efficiency, particularly for high-definition (HD) video. Though for testing efficiency our results are presented for CIF video, the findings can be applied to HD video over other high data rate members of the IEEE 802.11 family, such as IEEE 802.11ac. However, in that case larger slice sizes should be selected.

References

- [1] O. Oyman and S. Singh, "Quality of experience for HTTP adaptive streaming services," *IEEE Communications Magazine*, vol. 50, no. 4, pp. 20–27, 2012.
- [2] B. Bing, *3D and HD Broadband Video Networking*, Artech-House, Boston, Mass, USA, 2010.
- [3] I. E. G. Richardson, *H.264 and MPEG-4 Video Compression*, John Wiley & Sons, Chichester, UK, 2004.
- [4] P. Haskell and D. Messerschmitt, "Resynchronization of motion compensated video affected by ATM cell loss," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '92)*, pp. 545–548, March 1992.
- [5] M. M. Hannuksela, Y. K. Wang, and M. Gabbouj, "Isolated regions in video coding," *IEEE Transactions on Multimedia*, vol. 6, no. 2, pp. 259–267, 2004.
- [6] R. M. Schreier and A. Rothermel, "Motion adaptive intra refresh for low-delay video coding," in *Proceedings of the International Conference on Consumer Electronics (ICCE '06)*, pp. 453–454, Las Vegas, Nev, USA, January 2006.
- [7] "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 8: Medium Access Control (MAC) Quality of Service Enhancements," IEEE Std 802. 11e-2005 (Amendment to IEEE Std 802. 11, 1999 Edition (Reaff 2003)), 2005.
- [8] I. Ali, S. Moiron, M. Fleury, and M. Ghanbari, "Prioritized packetization for video with intra-refresh macroblock line," in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME '11)*, pp. 1–6, July 2011.
- [9] I. Ali, S. Moiron, M. Fleury, and M. Ghanbari, "Enhanced prioritization for video streaming over QoS-enabled wireless networks," in *Proceedings of the IEEE Wireless Advanced (WiAd '11)*, pp. 268–272, June 2011.
- [10] I. Ali, S. Moiron, M. Fleury, and M. Ghanbari, "Enhanced prioritization for video streaming over wireless home networks with IEEE 802. 11e," in *Proceedings of the IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB '11)*, pp. 1–6, June 2011.
- [11] I. Ali, M. Fleury, and M. Ghanbari, "Content-aware intra-refresh for video streaming over lossy links," in *Proceedings of the IEEE International Conference on Consumer Electronics (ICCE '12)*, pp. 118–119, January 2012.
- [12] A. Ksentini, M. Naimi, and A. Guéroui, "Toward an improvement of H.264 video transmission over IEEE 802.11e through a cross-layer architecture," *IEEE Communications Magazine*, vol. 44, no. 1, pp. 107–114, 2006.
- [13] G. Côté and F. Kossentini, "Optimal intra coding of blocks for robust video communication over the Internet," *Signal Processing*, vol. 15, no. 1, pp. 25–34, 1999.
- [14] X. Wang, C. Kodikara, A. H. Sadka, and A. M. Kondo, "Robust GOB intra refresh scheme for H.264/AVC video over UMTS," in *Proceedings of the 6th IEEE International Conference on 3G and Beyond*, pp. 1–4, November 2005.
- [15] J. T. Wang and P. C. Chang, "Error-propagation prevention technique for real-time video transmission over ATM networks," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, no. 3, pp. 513–523, 1999.
- [16] W. T. Tan, E. Setton, and J. Apostolopoulos, "Lossless FMO and slice structure modification for compressed H.264 video," in *Proceedings of the 14th IEEE International Conference on Image Processing (ICIP '07)*, vol. 4, pp. IV285–IV288, September 2007.
- [17] I. A. Ali, M. Fleury, and M. Ghanbari, "Congestion-resistant scalable media stream mapping for an IEEE 802.11E sensor

- network,” in *Proceedings of the 17th IEEE International Conference on Image Processing (ICIP '10)*, pp. 2901–2904, September 2010.
- [18] X. Li, T. Ren, and J. Xu, “A cross-layer design for transmission of scalable H.264 video over IEEE 802.11e networks,” in *Proceedings of the 1st International Conference on Computational Problem-Solving (ICCP '10)*, pp. 306–309, December 2010.
 - [19] R. Soni, N. Chilamkurti, G. Giambene, and S. Zeadally, “A cross-layer design for H.264 video stream over wireless local area networks,” in *Proceedings of the International Symposium on Computer Science and its Applications (CSA '08)*, pp. 387–392, October 2008.
 - [20] W. T. Chen, T. C. Lin, Y. C. Chang, and J. C. Chen, “Dynamic packet selection for H.264 video streaming over IEEE 802.11e WLANs,” in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '08)*, pp. 3133–3138, March 2008.
 - [21] Q. Liu, Z. Zou, and C. W. Chen, “A deadline-aware virtual contention free EDCA scheme for H.264 video over IEEE 802.11e wireless networks,” in *Proceedings of the IEEE International Symposium of Circuits and Systems (ISCAS '11)*, pp. 625–628, May 2011.
 - [22] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra, “Overview of the H.264/AVC video coding standard,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560–576, 2003.
 - [23] Y. K. Wang, M. M. Hannuksela, V. Varsa, A. Hourunranta, and M. Gabbouj, “The error concealment feature in the H.26L test model,” in *Proceedings of the International Conference on Image Processing (ICIP '02)*, vol. 2, pp. 729–732, September 2002.
 - [24] F. Agboma and A. Liotta, “Addressing user expectations in mobile content delivery,” *Mobile Information Systems*, vol. 3, no. 3-4, pp. 153–164, 2007.
 - [25] R. MacKenzie, D. Hands, and T. O’Farrell, “QoS of video delivered over 802.11e WLANs,” in *Proceedings of the IEEE International Conference on Communications (ICC '09)*, pp. 1246–1250, June 2009.
 - [26] C. Casetti, C. F. Chiasserini, L. Merello, and G. Olmo, “Supporting multimedia traffic in 802.11e WLANs,” in *Proceedings of the IEEE Vehicular Technology Conference (VTC '05)*, pp. 2340–2344, May 2005.
 - [27] R. Haywood, S. Mukherjee, and X. H. Peng, “Investigation of H.264 video streaming over an IEEE 802.11e EDCA wireless testbed,” in *Proceedings of the IEEE International Conference on Communications (ICC '09)*, pp. 1516–1520, June 2009.