

Improving Wireless Multimedia Quality using Header Detection with Priors

Syed A. Khayam, Shirish Karande, Muhammad U. Ilyas, and Hayder Radha

Department of Electrical & Computer Engineering

Michigan State University

Email: {khayamsy, karandes, ilyasmuh, radha}@egr.msu.edu

Abstract—Recent wireless multimedia studies have revealed that forward error correction (FEC) on corrupted packets yields better bandwidth utilization and lesser delay than retransmissions. To facilitate FEC decoding at a wireless receiver, it is desirable to relay maximum number of (error-free and corrupted) packets to the receiver’s application layer. To that end, most cross-layer multimedia schemes perform partial checksum only on packet headers. However, even with a partial checksum, bursty wireless errors introduce frequent header corruptions, thereby causing considerable packet drops. In this paper, we extend our work in [1], which proposed *receiver-based* schemes to correct a packet’s *critical* (and corrupted) header fields. This paper: (a) poses header detection as the well-known decision theoretic problem of *detecting known parameters in noise*, (b) evaluates two detectors using bit-error traces collected over an 802.11b network, (c) provides throughput results for the detectors, and (d) uses video in conjunction with FEC as an example to highlight the efficacy of the proposed schemes. We show that header detection provides significant improvements in throughput and video quality over the conventional UDP/IP protocol stack.

I. INTRODUCTION

Cross-layer schemes are becoming increasingly popular for efficient utilization of wireless bandwidth, especially in the context of delay-constrained multimedia applications. Most cross-layer multimedia schemes relay corrupted packets to the application layer, which in turn decides to retain or drop such packets [1]–[4]. It has been shown that such a strategy results in better throughput and lesser delay than the traditional retransmission-based recovery. In many recent cross-layer studies [2]–[4], partial protocol checksums are used to cover only the headers, while ignoring payload/data errors.

Due to bursty wireless errors, many of the received packet headers are corrupted. These packets are dropped by contemporary cross-layer multimedia schemes [2]–[4]. Moreover, support for a partial checksum requires modifications to multimedia transmitters, receivers, and intermediate nodes. In a prior study [1], the authors identified that most of the received protocol headers are not required for packet processing at the receiver. We identified *critical header fields* (CHF) that uniquely identify each active multimedia session on the wireless network. The schemes proposed in [1] were truly receiver-based, requiring no changes to the transmitter and/or intermediate nodes.

In this paper we pose the header detection problem as a decision theoretic problem of *detecting known parameters in noise* [5]. As in [1], the a priori distribution of CHF is

computed using successfully received packets. Two schemes are considered here. The first scheme, referred to as *global statistics variant* (GSV), exploits the broadcast nature of wireless networks to maintain an a priori distribution of all error-free transmissions on the wireless medium. The second scheme, called *local statistics variant* (LSV), computes the a priori distribution using packets destined for the local receiver only. We also employ a likelihood function that exploits the sample space of the a priori distribution. When a corrupted packet is received, the a priori distribution and the likelihood function are used to compute the a posteriori distribution.

Performances of two detectors are evaluated, namely the *maximum a posteriori* (MAP) detector and the *rounded mean-a posteriori* (RMeAP) detector. We evaluate the efficacy of MAP and RMeAP detectors for varying number of multimedia sessions at three supported data rates (2, 5.5, and 11 Mbps) of an 802.11b LAN. Actual bit-error traces collected over an 802.11b LAN are used for all the experiments. We show that the both detectors perform remarkably well with negligible detection errors at all data rates. We use H.264/JVT video [6] in conjunction with Reed-Solomon-based forward error correction (FEC) codes to show that GSV and LSV require lesser FEC redundancy and render significantly better video quality than the traditional UDP/IP/802.11-MAC protocol stack.

II. THE HEADER DETECTION PROBLEM

In [1] we identified protocol fields in a packet’s header that are not liable to change throughout the transmission of a session and can uniquely identify a session at the receiver. The following fields were identified to satisfy the above criteria: *destination MAC address*, *source IP address*, *destination IP address*, *source port*, and *destination port*. Collectively these fields are referred to as the *critical header fields* (CHF).

On receiving a corrupt packet, the receiver attempts header detection only if either the destination MAC address or the destination IP address of the received packet matches the receiver’s own address. Thus in the present setup a packet in which both destination MAC and destination IP addresses are simultaneously corrupted represents a missed detection, as opposed to inaccurate detection where after header detection the CHF estimate is incorrect. Later in the paper we describe that missed and inaccurate detections can be respectively recovered using FEC and sequence numbers.

A. Generation of a Priori Distribution

The CHF for all packets belonging to a particular session are unique. Under the proposed approach, a multimedia receiver maintains a CHF histogram based on the last N error-free packets that it receives. (The terms *error-free* and *corrupted* respectively refer to packets that pass and fail the 802.11b MAC layer checksum.) Since the histogram is generated using error-free data, normalizing the histogram renders a robust a priori distribution of active sessions. The generation and maintenance of the CHF histogram can be based either on only the error-free packets that were destined for the receiver or, due to the broadcast nature of wireless networks, on the error-free packets destined for any node on the wireless network. We refer to the former strategy as the *local statistics variant* (LSV) and the latter as the *global statistics variant* (GSV) of the header detection methodology.

B. Generation of a Posteriori Distribution

Let W be a random variable representing the observed (received) value of the CHF in a received packet, and let X_i , for $i = 1, 2, \dots, n$, be all possible n values of the CHF based on the last N error-free packets. The list of all possible CHF can be provided by a receiver's multimedia application. Therefore the set $\bigcup_{i=1}^n X_i$, the set of all bins of the histogram, constitutes the sample space of the a priori distribution. Hence, each X_i represents the CHF of session i . The *a posteriori distribution* can now be expressed as [5]

$$P(X_i|W) = \frac{P(W|X_i)P(X_i)}{\sum_{j=1}^n P(W|X_j)P(X_j)}. \quad (1)$$

The a posteriori distribution represents the probability that, given that the received (and possibly corrupted) CHF is W , the actually transmitted CHF was X_i . The prior probabilities, $P(X_i)$'s, in (1) are taken from the normalized CHF histogram.

The *likelihood function*, $P(W|X_i)$, in (1) is computed assuming a memory-less channel with a fixed probability of bit-error p . To calculate the likelihood function $P(W|X_i)$ for a particular X_i , we compute the hamming distance between W and X_i . The hamming distance indicates the total number of bits that are different between W and X_i . Assuming that the different bits are in fact the bit-errors introduced by the memory-less channel, $P(W|X_i)$ can be written as

$$P(W|X_i) = p^{hd(W,X_i)} (1-p)^{L-hd(W,X_i)}, \quad (2)$$

where L is the length of the CHF in bits, and $hd(a, b)$ is the Hamming distance between bit sequences a and b . Equation (2) measures the probability that the observed (corrupted) CHF value is W given that the transmitted (correct) CHF was X_i . By plugging (2) in (1) we can evaluate the a posteriori distribution on which different detectors can be applied. In essence, once the a posteriori distribution has been computed, the header detection problem translates into the well-known decision theoretic problem of *detecting known parameters*

in noise [5]. The memory-less channel assumption, although somewhat unrealistic, was adopted due to the difficulty of accurate channel estimation in real-time. Furthermore, it can be shown that for any reasonable channel with a probability of error $p < 0.5$, incorrect estimates of p have no impact on CHF detection.

C. The MAP and RMeAP Detectors

The following two detectors are employed in this paper:

The Maximum a Posteriori (MAP) Detector: This detector selects the *mode* (i.e., most likely X_i) of the a posteriori distribution $P(X|W)$ as the detected CHF.

The Rounded Mean a Posterior (RMeAP) Detector: This detector selects the mean of the a posteriori distribution $P(X|W)$, rounds it to the nearest CHF value, and uses that value as the detected CHF. We map the CHF of each multimedia session to a distinct integer value. Thus the mapping yields $X_1 \rightarrow 1, X_2 \rightarrow 2, \dots, X_n \rightarrow n$. The mean is computed using the mapped values, $\{1, 2, \dots, n\}$, as the outcomes of the random variable X . Once the mean is computed and rounded, we again map the resultant integer value, say i , back to its corresponding CHF, X_i .

D. Discussion: Advantages of Header Detection over Retransmissions

At this point it is important to highlight the advantages of using header detection instead of retransmissions to recover corrupted packets. Transmitters of multimedia sessions are generally not located on the same network as the receivers; consider, for instance, a transmitter situated on the Internet. In such scenarios, TCP is generally used for retransmission-based recovery of lost packets. The delays incurred by TCP in the presence of Internet congestion can easily run in orders of seconds. Such high delay can be unacceptable for real-time multimedia applications. Furthermore, it has been shown in prior studies that the bit-error and the consequent packet losses under realistic settings of wireless networks are extremely bursty. Thus, once a corrupt packet is received, the probability that the next packet will be corrupt is very high. While it is unlikely that a retransmission-based wireless multimedia application can recover from such back-to-back packet drops in a timely manner, the bandwidth wasted in such retransmission-based recovery is also prohibitive. As a final comment, ACKs are not provided in a multicast wireless transmission. Consequently, retransmissions cannot be employed in multicast scenarios, which are quite common in many wireless multimedia applications, such as video-on-demand, video conferencing etc.

III. EXPERIMENTAL SETUP

We use the 802.11b MAC layer bit-error traces collected in [2] to simulate the wireless channel. These bit-error traces were collected at three supported data rates – 2 Mbps, 5.5 Mbps, and 11 Mbps – of an operational 802.11b wireless LAN under realistic settings. Specifically, the traces were collected by positioning the access point

and the clients in separate rooms in order to simulate a realistic business/classroom/home-network wireless setup. It was highlighted in [2] that the robustness at the physical layer decreases with an increase in data rate, and consequently more errors are propagated to the MAC layer. For the traces used in this paper, the average bit-error rates are 0.032%, 0.31%, and 5% at 2, 5.5, and 11 Mbps, respectively. Also, the burstiness of bit-errors increases significantly with an increase in data rate. We observed that if the current packet is corrupted then the probability that the next packet will also be corrupted is 0.353, 0.684, and 0.897 at 2, 5.5, and 11 Mbps, respectively.

Our experimental setup consisted of up to twenty video servers with IP addresses of well-known streaming web servers. In order to quantify the worst-case performance, we used multiple source IP addresses from the same streaming subnet so that only a few bit-errors can map one source IP address into another. All the streams were being received at only three wireless stations in the wireless LAN. The wireless receivers had IP addresses which were very close to each other so that a few bit-errors can change a destination IP address to another. Distinct source and destination ports were used for each video stream.

The video streams were assigned to the three wireless receivers in a round-robin manner. In the next section we provide performance comparison for number of streams varying from five up to twenty. (Reception of more than one multimedia streams at a single receiver is an important scenario in many multimedia applications, such as video conferencing.) Again with the motivation of measuring the worst-case performance, all streams were assumed to have exactly the same encoding bitrate, i.e., the least informative priors.

IV. DETECTION ACCURACY

In this section we evaluate the detection accuracy of the MAP and RMeAP detectors employed in conjunction with GSV and LSV schemes. Note that although actual bit-error traces are used to simulate the wireless channel, we do not assume any channel knowledge at the wireless receivers. In other words, instantaneous bit-error statistics are not available at the receivers. Therefore a fixed instantaneous bit-error probability of $p = 0.1$ is used for all the results provided in this paper.

A. Detection Accuracy at 2 Mbps

Fig. 1 (a) outlines the GSV-based performances of MAP and RMeAP detectors at 2 Mbps. The dotted line in the figure provides the total number of corrupted packets. Both MAP and RMeAP detect corrupted CHF with extremely high accuracy. (Both MAP and RMeAP detectors provide correct estimates of more than 99% of the packets.) Such high accuracy of MAP and RMeAP detectors emphasizes the effectiveness of using the prior side-information to recover corrupted packets.

Fig. 1 (b) provides the performance of the detectors at receiver-0 in an LSV scenario. The performance trends of Fig. 1 (a) include both incorrect estimates (i.e., packets whose

destination was not accurately ascertained) and missed detections (i.e., packets whose recovery was not attempted). It is easily observed from LSV performance in Fig. 1 that both MAP and RMeAP provide very accurate detection with more than 99% accurate estimates for any number of streams. Thus, although based completely on local statistics, at 2 Mbps LSV has accuracy comparable to GSV.

B. Detection Accuracy at 5.5 Mbps

The GSV-based MAP and RMeAP detector performances shown in Fig. 2 (a) are similar to 2 Mbps. Both detectors provide extremely accurate detection of corrupted packets. (More than 99% of the packets are correctly detected.) Thus although the error-rate at 5.5 Mbps is much higher than 2 Mbps [2], Fig. 2 (a) shows that even for this somewhat high error-rate 5.5 Mbps channel the MAP and RMeAP detectors render very high accuracy in a GSV scenario.

The LSV-based results of Fig. 2 (b) are also consistent with the 2 Mbps results, i.e., both MAP and RMeAP detectors provide extremely accurate detections. (More than 98% of the packets at receiver-0 are detected correctly.) Thus we conclude that since the a priori information is derived from correctly received packets, the increased error-rate at 5.5 Mbps does not degrade detection accuracy.

C. Detection Accuracy at 11 Mbps

Fig. 3 (a) demonstrates that both detectors' GSV-based detection accuracy is slightly worse than at 2 and 5.5 Mbps. This performance degradation is due to the increased error-rate and is, nevertheless, quite insignificant. (Almost 98% of the corrupted packets are detected correctly.) However, LSV-based detection performance in Fig. 3 (b) is quite different from previous results. There is a clear drop in the performances of MAP and RMeAP detectors. This performance degradation is due to the considerable increase in error-rate at 11 Mbps as opposed to 1 and 5.5 Mbps. Due to the high error-rate, and due in part to the local nature of the present variant, there are not enough error-free packets to build the much required a priori side-information. Nevertheless, keeping in view the very high error-rate at 11 Mbps, the LSV performance is still quite good with more than 85% of the packets correctly detected.

D. Coping with Incorrect Detections

In the proposed header detection methodology, a packet that gets relayed to a session that it is not intended for represents an inaccurate detection or a false positive. We observed very low false positive rates at all data rates. For instance, the GSV-based MAP detector at 2, 5.5, and Mbps had false positive rates of 0.38%, 0.18%, and 0.45%, respectively. While these false positives are almost negligible, they can desynchronize the video and/or FEC decoders. Thus it is important to detect the false positives. To that end, when using JVT/H.264 based encoding one can have a single slice per packet, with the slice sequence numbers protected with enough redundancy to ensure that these sequence numbers can always be recovered at the receiver. A receiver can then drop all

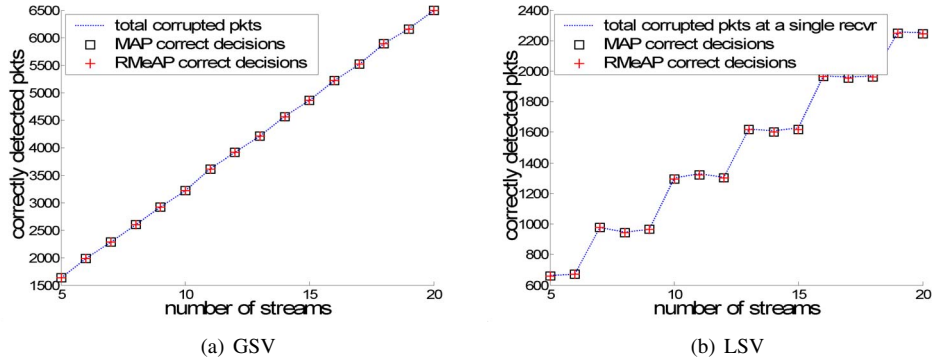


Fig. 1. Accuracy of MAP and RMeAP detectors for GSV and LSV at 2 Mbps.

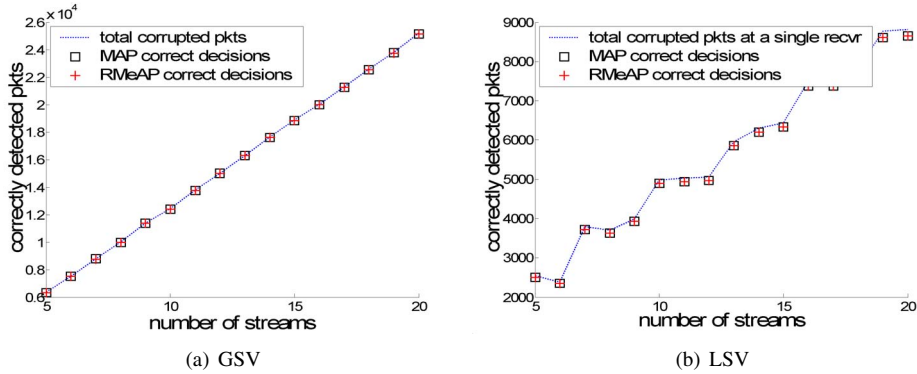


Fig. 2. Accuracy of MAP and RMeAP detectors for GSV and LSV at 5.5 Mbps.

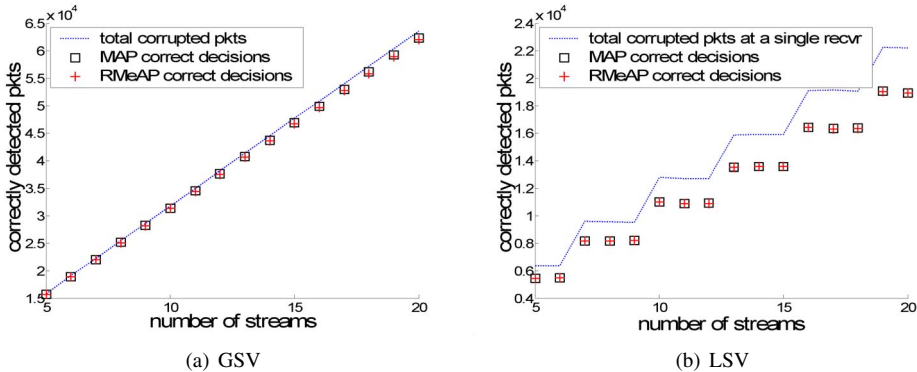


Fig. 3. Accuracy of MAP and RMeAP detectors for GSV and LSV at 11 Mbps.

packets whose slice numbers are much larger or smaller than the next/expected slice number. For video encoders that do not have a slice/packet sequence number, a small incremental packet sequence number with parity bytes can be easily inserted into each packet by the sender’s application layer. This sequence number based scheme can also provide erasure locations (i.e., dropped packets) to the FEC decoder.

V. FEC PERFORMANCE OF GSV AND LSV

Preceding sections focused on relaying maximum amount of error-free and corrupted multimedia data to the application.

This section evaluates the amount of FEC redundancy required by the application to recover the corrupted multimedia content. We compare the GSV and LSV redundancies with a traditional UDP/IP/802.11-MAC protocol stack. Since the MAP and RMeAP detectors are relaying approximately the same number of corrupted packets to the application, henceforth we only provide results for the MAP detector.

For the schemes proposed in this paper, some of the packets reaching the application layer have errors, while some other packets are lost (i.e., never reach the application due to missed or inaccurate estimations.) Thus, an FEC scheme

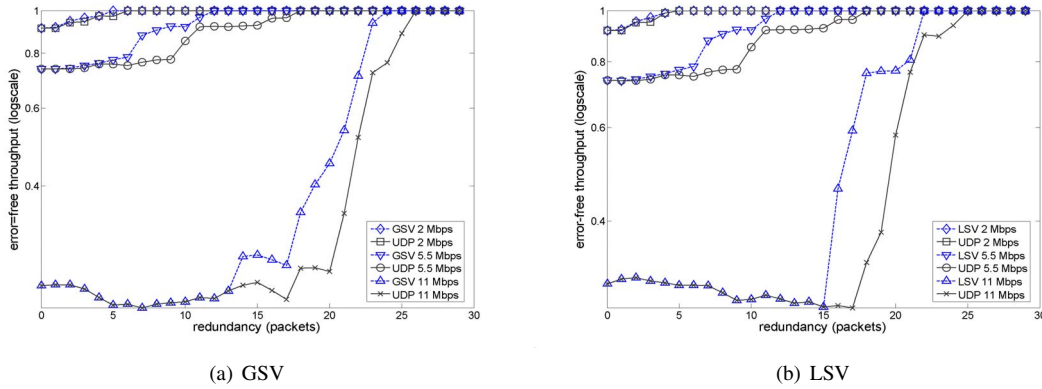


Fig. 4. FEC performances of GSV and LSV as compared to UDP/IP/802.11-MAC from the minimum up to 100% recovery.

operating in conjunction with LSV and GSV should be able to decode errors (in corrupted packets) and erasures (from missed detections) simultaneously.

A. FEC Construction

We use Reed-Solomon (RS) based FEC codes due to their popularity and simplicity. For error correction, if a codeword has $2t$ number of redundant symbols then a maximum of t transmission errors in that block can be corrected. For the same amount of redundancy, $2t$ erasures can be recovered. In the header detection schemes, for an FEC block with e_1 erasures (i.e., packet drops) and e_2 errors, if $e_1 \leq 2t$ then the FEC decoding algorithm can recover the e_1 erasures. After this erasure decoding, e_2 errors can be corrected if $e_2 \leq \lceil (2t - e_1)/2 \rceil$.

For the UDP/IP protocol stack, we employ RS erasure decoding (Berlekamp algorithm [7]) to recover dropped packets. For LSV and GSV, we use a variant of the RS decoding that is capable of joint error and erasure recovery [7]. In this section we show FEC results for a packet block length of $N = 30$. The FEC scheme is systematic and hence a packet block can be segregated into message packets and redundant packets. Each FEC codeword is composed of four bytes from a different packet, where each packet consists of 512 bytes. Each packet contributes to 128 separate FEC codewords, and each codeword spans over 30 packets. Thus, while the packet block-length is 30 packets, the code-length is 120 bytes (or 120 symbols where each symbol is drawn from a Galois field $GF(2^8)$.) Results for these FEC parameters are representative of performance trends that were observed for a wider range of packet block lengths, code lengths, and packet sizes.

B. FEC Performance of the MAP Detector

Fig. 4 outlines the FEC performances of GSV and LSV with respect to the UDP/IP/802.11-MAC protocol stack. It can be clearly seen that for all bitrates and for both GSV and LSV the conventional schemes require a higher amount of redundancy to provide 100% reliability when compared with the proposed schemes. Also observe that the FEC performances of GSV and LSV improve with increase in data rate. This is congruent

with our preceding discussions where we observed that the error-rate at 2 Mbps is quite low and therefore the errors do not require much redundancy to achieve 100% reliability. The required redundancy subsequently increases for the 5.5 and 11 Mbps data rates. Hence, the performance gap between GSV/LSV and the traditional protocol stack widens for 5.5 and 11 Mbps.

Here it is noteworthy that, due to their enhanced error-resilience features, emerging real-time applications can tolerate a certain level of losses in the multimedia content. Often in video streaming if the underlying source coding is error-resilient then the aim of the FEC scheme is to keep the number of losses under a tolerable threshold instead of attempting 100% data recovery. Thus it is important to consider the relative performance of the conventional and proposed schemes over the region where neither traditional UDP/IP/802.11-MAC nor GSV/LSV provides 100% reliability. The following section provides this performance comparison.

VI. VIDEO PERFORMANCE OF GSV AND LSV

Discussions in previous sections have concentrated on exhibiting the throughput improvements that can be achieved using GSV and LSV. For given FEC parameters (i.e., possibly less than 100% recovery), the video received by the application after header detection and FEC decoding contains error-free packets, partially corrupted packets, and completely dropped packets. It is then up to the multimedia application to recover from these channel impairments. In this section we decompress the video after FEC decoding (given certain FEC parameters) to further substantiate the advantages of using the proposed schemes over the UDP/IP/802.11-MAC protocol stack.

Our extensive experimental evaluations illustrated that at 2 Mbps the total packet drops, even for a conventional protocol stack, were very low, and the margin for throughput improvement was small. On the other hand at 11Mbps the proposed schemes easily outperformed the conventional stack. Therefore in this section only show results at 5.5 Mbps.

The sequences have a CIF frame size and are encoded at a frequency of 30 frames per second using the JVT/H.264 [6] encoder. The source encoding parameters are adjusted to

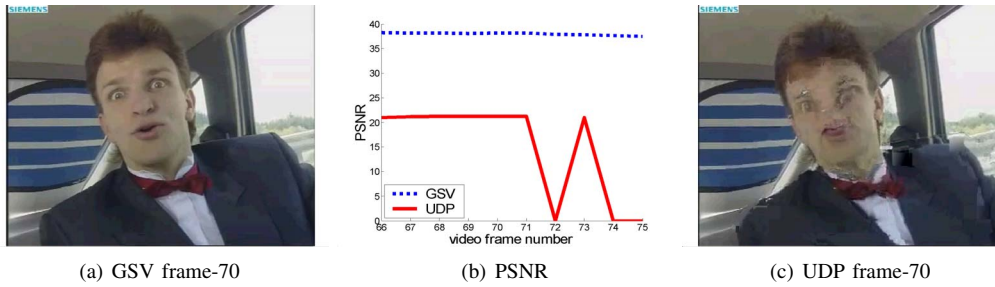


Fig. 5. GSV video quality evaluation at 5.5 Mbps for FEC with 21 message bytes in a 30 byte block.

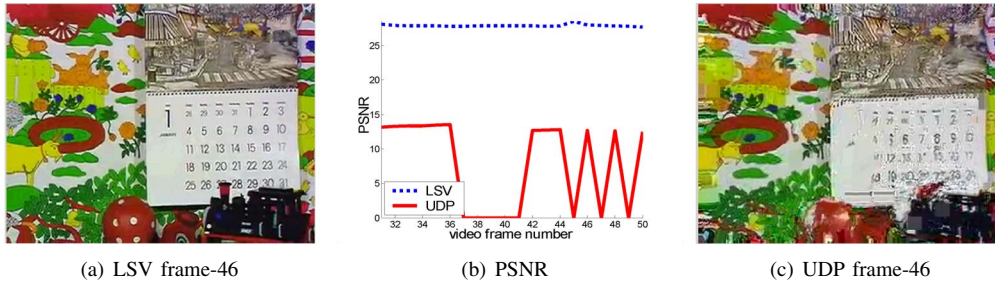


Fig. 6. LSV video quality evaluation at 5.5 Mbps for FEC with 21 message bytes in a 30 byte block.

provide a source bitrate of 700 kbps. A video packet-size of 512 bytes is used. We use the RS-based scheme described in the previous section with packet block length of 30 bytes. Each block consists of 21 message packets.

Figs. 5 and 6 provide subjective and temporal comparisons on the basis of video frames and peak signal-to-noise ratio (PSNR) for GSV and LSV, respectively. For the UDP/IP/802.11-MAC protocol stack, there were many video frames that were completely lost. These frame losses are represented by a PSNR of zero dB. In Fig. 5 we evaluate the performance of GSV for the carphone sequence. Here it should be emphasized that this video sequence has a high amount of temporal redundancy and enhanced JVT/H.264 error resilience features [6] were also enabled during source encoding of the sequence. Thus the temporal redundancy of the video sequence in conjunction with increased error resilience quantifies the best-case performance of the conventional UDP/IP/802.11-MAC protocol stack. Fig. 5 clearly establishes the advantage of using GSV over the conventional protocol stack. Since carphone is a low motion sequence, loss of entire picture frames significantly increases the jerkiness of the received video but does not necessarily result in block distortion. In the above example the losses in the conventional stack were so high and prolonged that both block distortions and motion jerkiness artifacts were observed in the multimedia content.

Video performance of the LSV scheme is outlined in Fig. 6. The PSNR comparison given in Fig. 6 (c) clearly illustrates that the LSV scheme can render significant improvements in overall video quality. It can be seen that a number of picture frames are completely lost due to the UDP/IP/802.11-MAC protocol stack and this leads to considerable block distortion and motion jerkiness. In general, our video simulation results

indicated a slightly inferior LSV performance as opposed to GSV. However, in some scenarios the performance of LSV was quite comparable or identical to GSV. Thus for channels where GSV cannot be employed, LSV provides an equally viable and effective alternative.

VII. CONCLUSIONS

In this paper, a decision theoretic approach was employed to correct packet header errors. We demonstrated that the proposed approach provides significant increase in throughput and consequently improves the video quality as compared to a traditional UDP/IP/802.11-MAC protocol stack.

REFERENCES

- [1] S. A. Khayam, M. U. Ilyas, K. Pörsch, S. Karande, and H. Radha, "A Statistical Receiver-based Approach for Improved Throughput of Multimedia Communications over Wireless LANs," *IEEE ICC*, May 2005.
- [2] S. A. Khayam, S. Karande, H. Radha, and D. Loguinov "Performance Analysis and Modeling of Errors and Losses over 802.11b LANs for High-Bitrate Multimedia Applications," *Signal Processing: Image Communications*, 18(7), pp. 575–595, August 2003.
- [3] L. Larzon, M. Degermark, and S. Pink, "UDP Lite for Real Time Multimedia Applications," *IEEE ICC*, June 1999.
- [4] E. Masala, M. Bottero, and J. C. De Martin, "Link-Level Partial Checksum for Real-Time Video Transmission over 802.11 Wireless Networks," *International Packet Video Workshop*, December 2004.
- [5] H. L. Van Trees, "Detection, Estimation, and Modulation Theory, Part I," Wiley: New York, 2001.
- [6] ISO/IEC JTC 1/SC29/WG11 and ITU-T SG16 Q.6, "Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Rec. H.264—ISO/IEC 14496-10 AVC)," Doc. JVT-G050, March 2003.
- [7] R. E. Blahut, "Theory and Practice of Error Control Codes," Addison-Wesley, May 1984.