

Robust Transmission of H.264/AVC sequences using List Decoding and Source Constraints

R. A. Farrugia, *Student Member, IEEE*, and C. J. Debono, *Senior Member, IEEE*

Abstract—Video compression standards commonly adopted in wireless multimedia transmission systems generally apply variable length codes (VLCs) to achieve the required data-rates. The resulting bitstreams are sensitive to transmission errors which cause visual artifacts that propagate in both spatial and temporal domains, thus reducing significantly the quality of the reconstructed video sequences.

This paper proposes the application of a List Decoder to improve the quality of H.264/AVC encoded sequences. This method relies on sequence estimation and residual source redundancies to recover feasible images. The recovered images may still contain errors which may cause visually distorted macroblocks (MBs). Therefore, the Average Intersample Difference across Boundaries (AIDB) metric was used to detect these visually impaired MBs at pixel level which are then adequately concealed. Simulation results show that the proposed algorithm manages to achieve Peak Signal-to-Noise (PSNR) gains of up to 2.17 dB when compared to the conventional decoding method. The proposed solution is fully compatible with the H.264/AVC standard and no additional bandwidth is required.

Index Terms—Error correction coding, error resilient coding, H.264/AVC, video coding, wireless video transmission.

I. INTRODUCTION

MULTIMEDIA communication services necessitate efficient video compression standards in order to deliver high quality content within limited bandwidths. H.264/AVC seems to be an attractive candidate for all wireless video applications mainly because of its enhanced compression efficiency and network friendly design [1]. However, it is sensitive to transmission errors, which are common in wireless environments, where a single corrupted bit may cause visual distortions which will significantly reduce the perceptual quality of the reconstructed video sequence.

A feedback channel, commonly used for reliable data transmission, cannot be adopted in typical conversational and multicast/broadcast applications because of delay constraints and the unavailability of the feedback channel respectively. Moreover, low bit-rate and low delay applications generally adopt the baseline profile which has limited error-resilient mechanisms [2], [3].

In order to solve these problems, a number of error-resilient mechanisms were proposed in literature. In [4], the

authors have applied syntax analysis in order to detect corrupted macroblocks (MBs). However, this method only manages to detect 57% of the corrupted MBs. Different error detection and concealment methods have been proposed which exploit the inherent redundancies within and outside an MB [5] – [7] at pixel level. However, these features achieve low detection rates when applied individually. To alleviate this problem, the authors in [8] have combined a set of pixel level features and applied heuristic thresholds. They have also applied an iterative error correction algorithm which has achieved a significant gain in Peak Signal-to-Noise Ratio (PSNR) at the expense of a significant increase in complexity, which makes it unpractical for real-time applications. Probabilistic Neural Networks (PNNs) have been applied in [9], [10] to detect visually distorted MBs at pixel level.

Although error detection and concealment can alleviate the propagation of distorted artifacts in the spatio-temporal domains they do not manage to recover the original bitstream. The authors in [11] have replaced the variable length code (VLC) tables with variable length error correction (VLEC) codes. However, this method reduces the compression efficiency of the codec since VLEC codes have longer average codeword lengths. In [12] it was shown that additional redundancy is available in compressed video data. This work was followed by the proposal of a Maximum Likelihood decoder for H.263 encoded sequences in [13]. Sequential decoding and source constraints were adopted in [14], [15] to recover feasible bitstreams. However, the sequential algorithm introduces a variable decoding delay [16] which could be a problem in real-time applications.

This paper presents the application of a modified List decoding algorithm [17], which recovers the maximum-likelihood feasible images from corrupted H.264/AVC sequences. The proposed algorithm applies source constraints to eliminate candidate solutions which provide semantically invalid VLC sequences. The resulting sequence may still include errors which may cause visually distorted MBs. For this reason, the Average Intersample Difference across Boundaries (AIDB) dissimilarity metric [10] was adopted to detect the residual distorted MBs which are then concealed. The proposed solution achieves a significant gain in quality when compared to the standard H.264/AVC decoder.

This paper is organized as follows. A short presentation of the H.264/AVC is given in section II. In section III the modified List Decoder is described in some details while an overview of the AIDB dissimilarity metric is given in the following section. Simulation results are presented in section V followed by the final comments and conclusion.

R.A. Farrugia is with the Department of Communications and Computer Engineering, University of Malta, MSD 2080, Malta. (phone: (+356) 23403054; e-mail: rrfarr@eng.um.edu.mt).

C.J. Debono is with the Department of Communications and Computer Engineering, University of Malta, MSD 2080, Malta. (phone: (+356) 23402076; fax (+356) 21343577; e-mail: cjdebo@eng.um.edu.mt)

II. H.264/AVC VIDEO CODING

The H.264/AVC standard [18] partitions a picture into fixed-sized MBs each covering a rectangular picture area (16x16 samples of the luminance and 8x8 samples of each chrominance component). The Baseline Profile, which is generally used in wireless applications, adopts two code families, namely Exponential Golomb Codes and Context-Adaptive VLC.

The VLC tables and the code family used for various syntax elements are switched depending on the values of previously encoded syntax elements. In fact, the H.264/AVC decoder manages more than 30 different VLC tables. The decoding of a slice follows the different steps presented in Fig. 1(a) for the I-slices and Fig. 1(b) for the P-slices. The decoding of an MB is done from the initial state depending on the slice type and proceeds step by step according to the H.264/AVC standard syntax. The decoder keeps track of the processing order of the slice it is decoding and derives the next VLC table to be used in the following step.

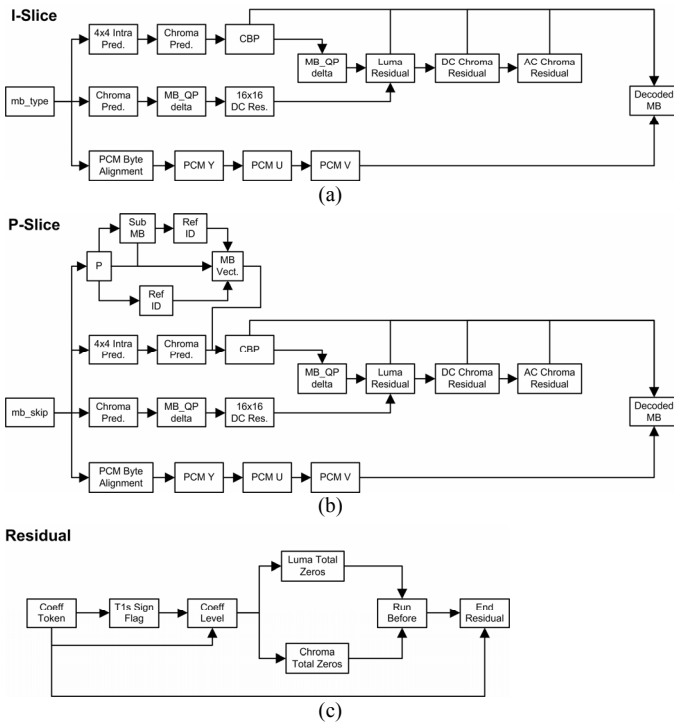


Fig. 1. An MB decoding process for (a) I-Slices, (b) P-Slices, and (c) Residual

The encoding of a residual block data, shown in Fig. 1(c), is the most complex part in the H.264/AVC encoding process and is done through the *Luma Residual*, *DC Chroma Residual*, *AC Chroma Residual*, and *16x16 DC Residual* modules. The Residual block is used to derive the number of non-zero coefficients (*Total_coeff*), the levels of the remaining non-zero coefficients (*Coeff_Level*), the total number of zeros before the last non-zero coefficient (*Total_zeros*) and the number of zeros preceding each non-zero coefficient (*run_before*). These parameters are then used for the VLC table prediction. More information is provided in [19].

The proposed error control mechanisms need to know each time which VLC table needs to be used. Since this is dependent on previously decoded information, a *Contextual Module* was developed. This module contains information on various constraints that can help the List Decoder to eliminate invalid candidate solutions and can be used to predict which table to be considered at each decoding stage.

III. PROPOSED ERROR CONTROL METHOD

The H.264/AVC codec adopts variable length codes in order to achieve high compression ratios. Each slice is generally encapsulated within RTP/UDP/IP packets, thus ensuring synchronization at slice level. When a packet is corrupted it is generally dropped and the contained slice is concealed. However, this generally results in over-concealment thus reducing the quality of the reconstructed video sequence.

The RTP payload extracted at the receiver consists of N bits which correspond to an H.264/AVC slice or frame. The number of VLC codewords and the corresponding sequence of VLC codewords have to be estimated. In order to reduce the complexity required to recover a corrupted slice or frame, the trellis architecture described in [20] and illustrated in Fig. 2 is adopted. Each state within the trellis is represented by an (n, k) double where n represent the accumulated number of bits and k denotes the symbol time. Assuming that the VLC table used at each encoding stage is known by the decoder and that the length of each codeword is denoted by $\{\lambda_1, \lambda_2, \dots, \lambda_M\}$, then each state $S_{n,k}$ at symbol time k creates M states $\{S_{n+\lambda_1, k+1}, S_{n+\lambda_2, k+1}, \dots, S_{n+\lambda_M, k+1}\}$. This decoding procedure can be terminated when $n = N$ and the survivor path with minimum Hamming distance is selected.

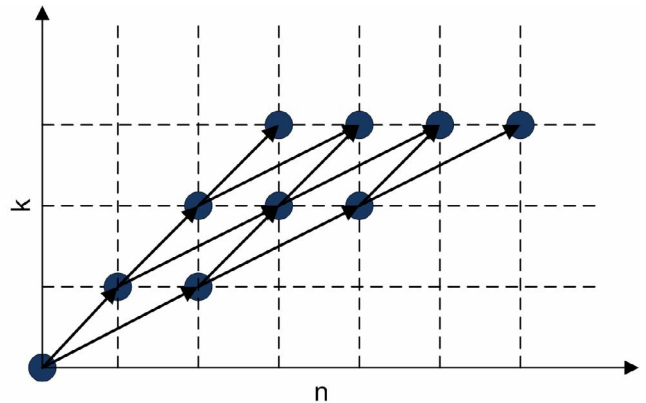


Fig. 2. Trellis diagram used by the proposed List Decoder

Furthermore, not all the valid VLC codeword sequences of N bits correspond to valid slices or frames. The sequences which do not meet a set of source constraints, inherent within the H.264/AVC syntax, are pruned to minimize complexity. Moreover, this ensures that only feasible slices or frames are decoded. The source constraints considered in this work are:

1. The decoded bitstream is of length N bits and is known by the decoder.

2. Only the codewords within a range specified by the standard were considered.
3. The decoded slice or frame contains exactly X complete MBs whose number is known a priori by the decoder.

The List decoder was modified in order to determine the maximum-likelihood VLC sequence of N bits which meets the constraints mentioned above. For this purpose, two lists are used; (1) L_k which corresponds to the list of incomplete feasible VLC sequences, each sequence being represented by a state $S_{n,k}$ and (2) F which contains complete feasible VLC sequences. At each recursion, L_{k-1} is available, and L_k is initialized as an empty list. At the end of the recursion the list L_k should contain incomplete VLC sequences of k symbols.

For each state $S_{n,k-1}$ in list L_{k-1} the following procedure is followed, assuming that the VLC table to be used is known a priori:

1. For each VLC codeword in the considered VLC table derive the length n using:

$$n = n_{before} + \lambda_i \quad (1)$$

where n_{before} is the length of the incomplete sequence being considered in state $S_{n_{before}^{k-1}}$ listed in L_k , and λ_i is the length in bits of the i^{th} codeword in the VLC table.

2. Compute the partial hamming distance Δh_i which is the Hamming distance between the i^{th} codeword and the inputted codeword of length λ_i .
3. If $\Delta h_i \leq 1$
 - 3.1. Compute the Hamming distance h_i of the sequence resulting from the concatenation of the sequence in state $S_{n_{before}^{k-1}}$ and the i^{th} codeword in the VLC table.
 - 3.2. If the concatenated sequence meets the requested source constraints the state $S_{n,k}$ is created and the resulting sequence together with the Hamming distance h_i are stored. Otherwise it is discarded.
 - 3.2.1. If $n = N$, the derived state is listed in list F which includes all complete feasible states. Otherwise if $n < N$ the state is stored in list L_k .

The recursion is terminated once the list L_k is empty given that $k > 1$. The most-likelihood feasible VLC sequence is the sequence listed in F with the smallest Hamming distance.

IV. PIXEL DOMAIN ERROR DETECTION AND CONCEALMENT

The modified List Decoder derives the most-likelihood sequence from a set of feasible slices or frames. However, errors may still be present in the recovered VLC sequence which will cause visual artifacts that propagate in the spatio-temporal domain. In order to reduce this affect, the *AIDB* dissimilarity metric [10] was used. As illustrated in Fig. 3, the dissimilarity across each boundary $AIDB(M : X)$, is computed using:

$$AIDB(M : X) = \begin{cases} \frac{1}{K} \|p^{in} - p^{out}\|_2 & \text{if } X \text{ is available} \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

where $X \in \{N, E, S, W\}$, $K = 16$, and $\|\cdot\|_2$ is the L^2 norm computed in the YUV color space. The *AIDB* dissimilarity metric is then derived by averaging the $AIDB(M : X)$ over the available neighboring MBs.

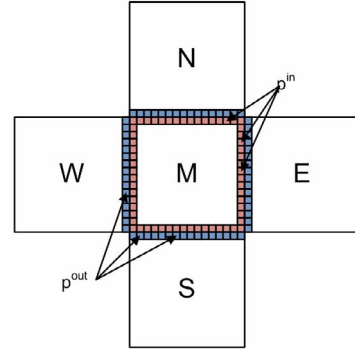


Fig. 3. The *AIDB* dissimilarity metric for an MB M

All the MBs whose *AIDB* dissimilarity metric is above a heuristic threshold are considered to be corrupted and are concealed.

V. SIMULATION RESULTS

The proposed error resilient mechanism was integrated within the Jointed Model (JM) software [21], together with the syntax check rules described in [4]. The raw video was encoded at QCIF resolution at 30 frames per second using Intra slices. The codec adopts dispersed FMO with five slices per picture. In compliance with JM software, each slice was encapsulated within RTP/UDP/IP packets. These packets were Binary Phase Shift Keying (BPSK) modulated and transmitted over an additive white Gaussian noise (AWGN) channel.

The received corrupted H.264/AVC bitstream was decoded using two different methods: 1) MB Level Concealment (MLC) where whenever a syntax rule is violated the current MB and the following ones are concealed until the end of slice, and 2) the maximum-likelihood H.264/AVC sequence derived using the modified List Decoder with the MLC being applied to the recovered feasible sequence. The performance of these methods at different bit error rates (BERs) is illustrated in Fig. 4, where the *Miss-America* sequence was considered.

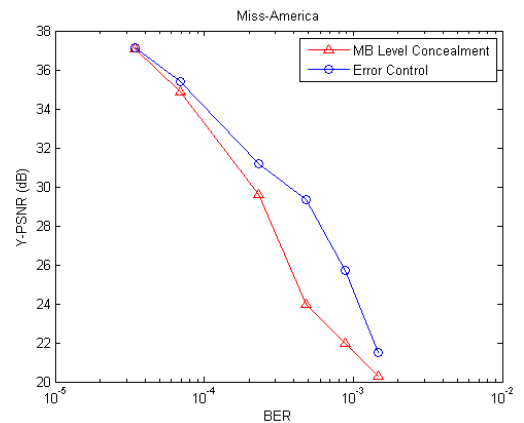


Fig. 4. PSNR for the *Miss-America* sequence at different BERs of the modified List Decoder algorithm

From Fig. 4 it is clear that deriving the maximum-likelihood feasible slice or frame provides significant gains in PSNR. However, some residual transmission errors are still present in the recovered H.264/AVC sequence which may provide significant visual distortions. For this reason, the dissimilarity metric described in section IV was used to detect corrupted MBs at pixel level. Those MBs whose *AIDB* dissimilarity metric was above 10 were considered to provide major artifacts and were thus concealed. Fig. 5 illustrates the gain in quality when the *AIDB* metric detection method is included.

The subjective gain in quality of the proposed method can be further illustrated in Fig. 6. The MLC method, Fig. 6(a), tries to decode the received H.264/AVC sequence and uses the syntax and semantic violations to detect corrupted VLC codewords. When an error is encountered the VLC codewords generally do not correspond to feasible slices or frames. The proposed List Decoder tries to derive the most-likelihood feasible image and in fact it provides a significant gain in perceptual quality as shown in Fig. 6(b). However, a number of artifacts are still present which can be detected using the *AIDB* dissimilarity metrics and then concealed, Fig. 6(c).

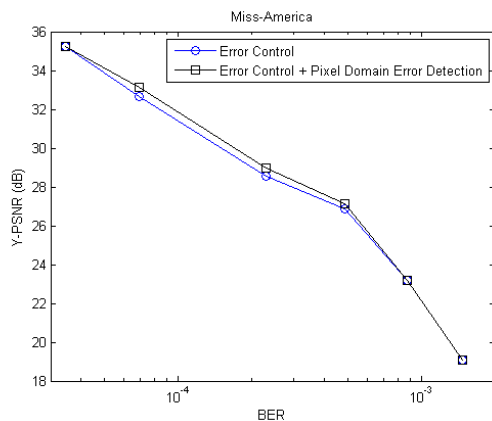


Fig. 5. PSNR for the Miss-America sequence at different BER when using the proposed algorithm with and without the *AIDB* dissimilarity metric.

VI. COMMENTS AND CONCLUSION

This paper has presented a novel error resilient method which derives the maximum-likelihood feasible slices or frames. This technique does not always result in reducing the number of errors within the corrupted bitstream. However, it recovers feasible slices and frames which most of the time result in re-synchronizing the bitstream. This generally results in an increase in the perceptual quality. Furthermore, the residual constraints reduce the complexity of the decoder and it is thought that the performance of this method will increase if more source constraints are adopted [12].

The maximum-likelihood H.264/AVC still contains errors which may cause distorted MBs. In order to alleviate this problem pixel error detection algorithms can be adopted. In this case we have adopted a simple *AIDB* dissimilarity metric in order to proof the concept. However,

better results are expected when applying more sophisticated pixel-level error detection algorithms such as those in [10].



Fig. 6. Frame 5 for the Miss-America sequence (a) MB Level concealment, (b) proposed List Decoder, and (c) proposed decoder with *AIDB* detection

VII. REFERENCES

- [1] T. Stockhammer, M.N. Hannuksela, and T. Wiegand, "H.264/AVC in Wireless Environments," in *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 657 – 673, Jul. 2003.
- [2] S. Kumar, L. Xu, M.K. Mandal, and S. Panchanathan, "Error Resiliency Schemes in H.264/AVC Standard," in *Elsevier J. of Visual Communications & Image Representation*, vol. 17 (2), pp. 425 – 450, Apr. 2006.
- [3] Jörn Ostermann, J. Bormans, P. List, D. Marpe, M. Narroschke, F. Pereira, T. Stockhammer, and T. Wedi, "H.264/AVC: Tools, Performance, and Complexity" in *IEEE Circuits Syst. Mag.*, vol. 2, no. 1, pp. 7 – 28, 2004.
- [4] L. Superiori, O. Nemethova, and M. Rupp, "Performance of a H.264/AVC Error Detection Algorithm Based on syntax Analysis," in *Proc. 2006 Int. Conf. on Advances in Mobile Computing and Multimedia*, pp. 1 – 10.
- [5] R.A. Farrugia, and C.J. Debono, "Enhancing the Error Detection Capabilities of the Standard Video Decoding using Pixel Domain Dissimilarity Metrics," in *Proc. 2007 IEEE EUROCON Conf.*, pp. 1085 – 1090.
- [6] L. Superiori, O. Nemethova, and M. Rupp, "Detection of Visual Impairments in the Pixel Domain of Corrupted H.264/AVC Packets," in *Proc. 2007 IEEE Int. Picture Coding Symposium*.

- [7] O. Lehtoranta, T.D. Hämäläinen, and V. Lappalainen, "Detecting Corrupted Intra Macroblocks in H.263 Video," in *Proc. 2002 Multimedia Signal Processing Workshop*, pp. 33 – 36.
- [8] E. Khan, S. Lehmann, J. Gunji, and M. Ghanbari, "Iterative Error Detection and Correction of H.263 Coded Video for Wireless Networks," in *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 12, pp. 1294 – 1307, Dec. 2004.
- [9] R.A. Farrugia, and C.J. Debono, "Enhancing Error Resilience in Wireless Transmitted Compressed Video Sequences through a Probabilistic Neural Network Core," in *Proc. 2007 IEEE Int. Picture Coding Symposium*.
- [10] R.A. Farrugia, and C.J. Debono, "Resilient Transmission of H.264/AVC Video Sequences using Probabilistic Neural Networks," accepted for publication in *Int. Symp. on Communication, Control and Signal Processing 2008*.
- [11] V. Buttigieg, and R. Deguara, "Using Variable-Length Error-Correcting Codes in MPEG-4 Video," in *Proc. 2005 Int. Symp. on Information Theory*, pp. 2379 – 2383.
- [12] H. Nguyen, and P. Duhamel, "Estimation of Redundancy in Compressed Image and Video Data for Joint Source Channel Decoding," in *Proc. 2003 Global Telecommunications Conf.*, pp. 2198 – 2202.
- [13] H. Nguyen, P. Duhamel, J. Brouet, and D. Rouffet, "Optimal VLC Sequence Decoding Exploiting Additional Video Stream Properties," in *Proc. 2004 IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, pp. 621 – 624.
- [14] C. Weidmann, P. Kadlec, O. Nemethova and A. Al Moghrabi, "Combined Sequential Decoding and Error Concealment of H.264 Video," in *Proc. 2004 Multimedia Signal Processing Workshop*, pp. 299 – 302.
- [15] C. Bergeron, and C. Lamy-Bergot, "Soft-Input Decoding of Variable-Length Codes applied to the H.264 Standard," in *Proc. 2004 Multimedia Signal Processing Workshop*, pp. 87 – 90.
- [16] S. Lin and D.J. Costello, *Error Control Coding: Fundamentals and Application*, Englewood Cliffs, Prentice-Hall, 1983.
- [17] T. Hashimoto, "A List-Type Reduced-Constraint Generalization of the Viterbi Algorithm," in *IEEE Trans. on Information Theory*, vol. IT-33, no. 6, Nov. 1987.
- [18] *ITU-T Rec. H.264, ISO/IEC 14496-10 AVC*, document n. JVT-G050r1, Geneva, Switzerland, May. 2003.
- [19] G. Bjontegaard and A. Luthra, "Context-adaptive VLC (CVLC) coding of coefficients" document in JVT-C028r1, Fairfax, VA, May 6-10, 2002.
- [20] R. Bauer, and J. Hagenauer, "Symbol-by-Symbol MAP Decoding of Variable Length Codes," in *Proc. 2000 3rd ITG Conf. on Source and Channel Coding*, pp. 111 – 116.
- [21] H.264/AVC Software Coordination, "JM Software," ver. 12.2 [Online], Available: <http://iphone.hhi.de/suehring/tml>.

the University of Malta. In 2000 he was also engaged as a Research Associate with Texas A&M University, Texas. In 2001 he was appointed Lecturer with the Department of Communications and Computer Engineering at the University of Malta and is now a Senior Lecturer. He is a senior member of the IEEE and is currently chair of the IEEE Malta Section. His research interests are in RF and microwave systems development and applications, resilient multimedia transmission and modeling of communication systems.

VIII. BIOGRAPHIES



Reuben A. Farrugia (S' 04) has received the first degree in electronic engineering from the University of Malta. He is currently reading a Ph.D. degree in Communications Engineering with the Department of Communications and Computer Engineering at the University of Malta, Malta.

Between 2004 and 2007 he was employed as a Research Engineer with the Department of Communications and Computer Engineering of the University of Malta. Since January 2008, he has been appointed Assistant Lecturer with the Department of Communications and Computer Engineering at the University of Malta. His research interests are in robust multimedia transmission, multimedia processing, forward error correction codes and machine learning.



Carl James Debono (S'97, M'01, SM'07) received his B.Eng.(Hons.) degree in Electrical Engineering from the University of Malta, Malta, in 1997 and the Ph.D. degree in Electronics and Computer Engineering from the University of Pavia, Italy, in 2000.

Between 1997 and 2001 he was employed as a Research Engineer in the area of Integrated Circuit Design with the Department of Microelectronics at