

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/221559049>

Robust Video Transmission Using Reversible Watermarking Techniques

Conference Paper · December 2010

DOI: 10.1109/ISM.2010.31 · Source: DBLP

CITATIONS

4

READS

50

2 authors, including:



Reuben A. Farrugia

University of Malta

82 PUBLICATIONS 295 CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:



Light field Image Restoration [View project](#)



Improving the quality of very low resolution facial images [View project](#)

All content following this page was uploaded by [Reuben A. Farrugia](#) on 19 May 2014.

The user has requested enhancement of the downloaded file.

Robust Video Transmission using Reversible Watermarking Techniques

Robert Facciol and Reuben A. Farrugia

Department of Communications and Computer Engineering
University of Malta
Malta

robertfacciol@gmail.com, reuben.farrugia@um.edu.mt

Abstract— This paper presents a novel error-resilient strategy which employs a reversible watermarking technique to protect the H.264/AVC video content. The proposed scheme adopts reversible watermarking to embed an error detection codeword within every Macroblock (MB). The watermark is then extracted at the decoder and used to detect the corrupted MBs to be concealed. The proposed scheme further manages to recover the original video content after watermark extraction, thus providing no loss in video quality. The simulation results demonstrate that the proposed approach provides a substantial gain of up to 2.6 dB in Peak Signal-to-Noise Ratio (PSNR) relative to the standard with a minimal increase in complexity.

Keywords- Error detection, error resilient coding, fragile watermarking, reversible watermarking, H.264/AVC

I. INTRODUCTION

Recent advances in video compression and wireless technologies have made location-independent access to multimedia services possible. However, one of the major challenges in wireless video delivery is the vulnerability of video coding standards to transmission errors, where even a single corrupted bit may cause significant visual impairments which compromise the perceptual quality of the recovered video content [1].

H.264/AVC [2] offers several error resilient schemes which make wireless video delivery more robust to transmission errors [3] – [5]. Nevertheless, these methods consider a packet-loss scenario, where the receiver discards the corrupted packets and conceals all the Macroblocks (MBs) contained within the packet, even the uncorrupted MBs [6]. This implies that the error resilient methods adopted by the standard operate at a lower bound since they assume a worst-case scenario.

A number of error resilient strategies can be found in literature. In [7], syntax and semantic violation rules were proposed which only manage to locate 57% of the visually impaired regions. Joint Source-Channel techniques were adopted in [8] – [10]. However, the method proposed in [8] reduces the compression efficiency of the codec while the other approaches provide limited protection. The performance of the Pixel-level artefact detection mechanisms proposed in [11] – [13] employ heuristic thresholds which are sequence dependent. Machine learning algorithms were recently introduced to detect the visually impaired MBs to be concealed [14], [15]. These approaches were found to provide a better generalization. Nonetheless,

they are still prone to false positives and thus may result in superfluous concealment of undistorted MBs.

Fragile watermarking was adopted in [16] – [18] to embed information that aids the detection and concealment of distorted regions. A low resolution version of each video frame was embedded in itself in [19] using spread-spectrum watermarking techniques and is used to aid concealment of distorted regions. However, the watermarking techniques adopted in [16] – [19] are irreversible, and therefore the embedded information reduces the quality of the transmitted video even when transmitted over noise-free channels.

This paper presents the application of reversible watermarking where a variable length error detection code is embedded within each MB. This payload is reversibly extracted by the decoder to check the correctness of the protected video content. The presented method manages to achieve Peak Signal-to-Noise Ratio (PSNR) gains of up to 2.6 dB relative to the standard. Furthermore, the presented results show that the proposed system outperforms other state of the art watermarking based error resilient methods found in literature.

This paper is organized as follows. The Reversible Data Embedding (RDE) procedure adopted in this paper is presented in Section 2 followed by the description of the proposed error resilient method. The simulation results are presented in Section 4 while the final comments and conclusion are provided in Section 5.

II. REVERSIBLE DATA EMBEDDING (RDE)

The Reversible Data Embedding (RDE) proposed by Tian [20] was adopted in this work since it has the appealing property that the decoder can recover the original undistorted video after watermark extraction. The RDE algorithm was originally implemented in the pixel domain where a pair of coefficients x and y are transformed according to

$$l = \left\lfloor \frac{x+y}{2} \right\rfloor \quad \text{and} \quad h = x - y \quad (1)$$

where l is the average and h is the difference of the coefficients. The difference h is said to be expandable if

$$|2 \times h + b| \leq \min(2(255 - l), 2l + 1) \quad (2)$$

where b is the bit to be embedded. The difference value, h , is said to be changeable if it satisfies the condition

$$\left\lfloor 2 \times \left\lfloor \frac{h}{2} \right\rfloor + b \right\rfloor \leq \min(2(255 - l), 2l + 1) \quad (3)$$

Unambiguously expandable (UEN) pairs are those pairs which are still expandable after data embedding [21]. Bits can only be embedded in expandable (EN), unambiguously expandable (UEN) and changeable (CN) pairs. If a pair does not satisfy any of these conditions it is said to be non-changeable (NC) and no data can be hidden in that pair. To embed a bit b , the coefficient pairs are modified according to

$$x' = l + \left\lfloor \frac{h' + 1}{2} \right\rfloor, \quad y' = l - \left\lfloor \frac{h'}{2} \right\rfloor \quad (4)$$

where x' and y' are the modified pixel pairs and h' is derived using

$$h' = 2 \times h + b \quad (5)$$

Given that not all pixel pairs are suitable for data embedding, the decoder must track the pixel-pairs which were modified at the encoder. Therefore, a location map is required, where a map bit value would be required for each pair of coefficients. However, [21] and [22] reduce this since UEN and NC pairs can be immediately recognized by the extracting algorithm without the need of a map bit. Furthermore, the location map can be further compressed using a lossless compression algorithm [20].

The data extracting algorithm is then reversed and the embedded bit is extracted from the least significant bit (LSB) of the difference h . Once the embedded bit is extracted, h is modified according to

$$h' = \left\lfloor \frac{h}{2} \right\rfloor \quad (6)$$

The original coefficients are then derived using (4).

III. REVERSIBLE DATA EMBEDDING USING RELATION BASED WATERMARKING

The RDE algorithm was originally proposed to be applied on pixel pair coefficients. However, the quantization process adopted by lossy video compression standards, such as H.264/AVC, corrupts the embedded information, thereby making the algorithm useless. For this purpose, it was decided to apply the RDE on the quantized transformed coefficients, as shown in Figure 1a [23]. The watermark extraction is performed in the feedback to ensure synchronization between the encoder and decoder.

The information to be embedded is inserted in the quantized transform coefficients of the 4×4 blocks. The data was embedded only within UEN and EN pairs since CN pairs require extra bits to be sent. Furthermore, since a different number of bits are embeddable in different 4×4 blocks, the Relation-Based Error Detection (RBED) [16], which is a variable length check code, was used. The payload to be embedded is derived using

$$K = \left(\sum_{i=0}^{15} c(i) \right) \text{MOD}(M) \quad (7)$$

where $M = 2^n$, n is the number of bits embeddable in the 4×4 block, $c(i)$ is the i^{th} transform coefficient and K is the resulting payload to be embedded.

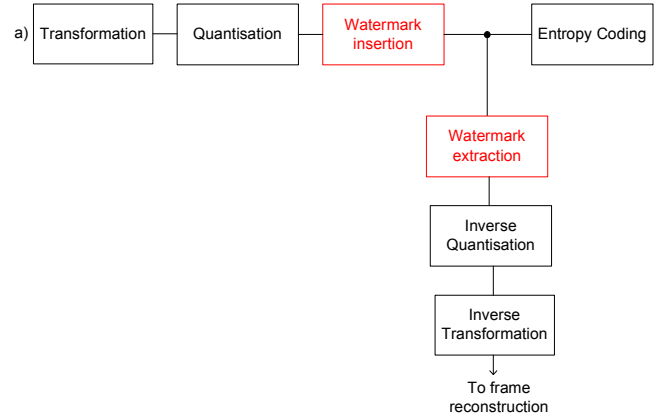


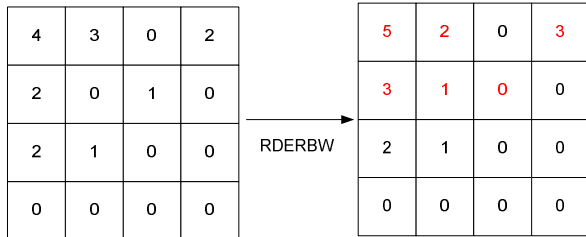
Figure 1: a) watermark insertion and extraction at the encoder, b) watermark extraction and validation at the decoder

As regards the location map, no bits were used to mark UEN and NC pairs. However, since EN pairs can become CN after data embedding, EN pairs which become CN after data embedding were marked with a 1 while CN pairs were marked with a 0. The location map was not compressed, and was transmitted as side information.

The insertion of the watermark was done within the prediction loop of the encoder. This ensures that the chosen encoded 4×4 block is the one that requires the least amount of bits to be represented. Furthermore, being a reversible watermark, the DC component of the 4×4 block can also be used for data embedding without introducing distortion in the frame. Figure 2 shows an example of the watermark insertion. The transform coefficients are considered as a sequence of numbers organised as per the zig-zag scan adopted by the encoder. They are then grouped in pairs and classified as EN, CN and NC using (1) – (3). The number of bits embeddable is then calculated. Once the number of embeddable bits is known, the payload is calculated using (7), and it is then embedded into the pairs using (4) and (5). The map value is then set according to the new categorisation of the modified pairs.

At the decoder, the watermark is extracted from the 4×4 blocks of coefficients (as show in Figure 1b) and the payload calculated again after the watermark is extracted. Transmission errors are detected when the extracted payload and the newly calculated payload after watermark extraction do not match. Figure 3 shows an example of the extraction algorithm. The extraction algorithm works exactly the same as the insertion algorithm up to the classification of the pairs. For EN pairs, the LSB of h is extracted and then h is modified according to (6). The original coefficients are then obtained using (4). For CN pairs, the location map is checked to determine if data extraction is required. If the

map value is 1, a bit is extracted from the LSB of h , otherwise data extraction does not take place. The subsequent modification of h and restoration of the original coefficients are the same as for EN pairs. For NC pairs, the decoder knows *a priori* that no data extraction is to take place. Once all of the coefficient pairs have been considered, the embedded payload has been extracted. The payload is then calculated again using the 'new' set of pairs and compared to the extracted payload. If they do not match, then an error has been detected and the MB is marked as corrupted. MBs in that slice from that MB onwards are then discarded and concealed.

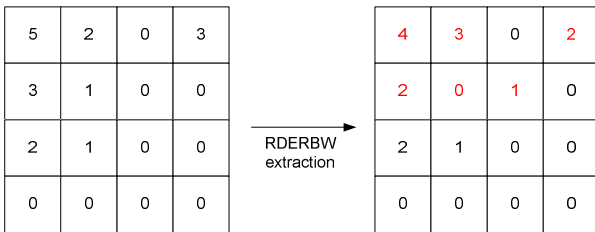


Number of bits embeddable: 7
 Payload: $(4+3+2+2+0+0+2+1+1+0+0+0+0+0+0) \text{ MOD } 2^7 = 15 \rightarrow 0001111$

Pair	Set	Embed Bit	New Pair	New Set	Map Value
(4,3)	expandable	1	(5,2)	expandable	-
(2,2)	expandable	1	(3,2)	expandable	-
(0,0)	expandable	1	(1,0)	changeable	1
(2,1)	expandable	1	(3,0)	changeable	1
(1,0)	changeable	-	(1,0)	changeable	0
(0,0)	expandable	0	(0,0)	expandable	-
(0,0)	expandable	0	(0,0)	expandable	-
(0,0)	expandable	0	(0,0)	expandable	-

Bits embedded: 7
 Map: 011

Figure 2 : Reversible Data Embedding Relation Based Watermarking (RDERBW) insertion example



Pair	Set	Map	Extract	New Pair	Extracted bit
(5,2)	expandable	-	Yes	(4,3)	1
(3,2)	expandable	-	Yes	(2,2)	1
(1,0)	changeable	1	Yes	(0,0)	1
(3,0)	changeable	1	Yes	(2,1)	1
(1,0)	changeable	0	No	(1,0)	-
(0,0)	expandable	-	Yes	(0,0)	0
(0,0)	expandable	-	Yes	(0,0)	0
(0,0)	expandable	-	Yes	(0,0)	0

Number of bits extracted: 7
 Payload: $(4+3+2+2+0+0+2+1+1+0+0+0+0+0+0) \text{ MOD } 2^7 = 15 \rightarrow 0001111$

Figure 3: Reversible Data Embedding Relation Based Watermarking (RDERBW) extraction and validation example

IV. SIMULATION RESULTS

The proposed error resilient strategy was integrated within the Baseline profile of the Joint Model (JM) software 12.2 [24] which was modified to allow the decoder to decode

partially damaged H.264/AVC slices. The decoder thus accepts corrupted packets and employs the syntax and semantic check rules presented in [7] to detect non-H.264/AVC bitstreams. The reception of corrupted slices can in practice be achieved using transport layer protocols such as UDP Lite [25]. The raw video sequences were encoded at QCIF resolution, with the format IPPPP.... The encoder applies a random Intra refresh of 5% and each slice has a maximum of 10 MBs. In compliance with the JM software, each slice was encapsulated within RTP/UDP/IP packets and transmitted over an Additive White Gaussian Noise (AWGN) channel. The standard error concealment algorithms implemented by the JM 12.2 software model were used to conceal the corrupted MBs for both the standard and proposed approach. The standard mechanism adopts Slice Level Concealment (SLC) where the entire corrupted slice is dropped and concealed. On the other hand, the proposed approach manages to detect the actual MBs affected by the transmission errors and thus can locate the MBs to be concealed, thus employing MB Level Concealment (MLC). It will be shown in latter simulation results that the proposed scheme provides better localization of the corrupted MBs and thus minimizes the area to be concealed, thus improving both objective and subjective quality. In order to get a better representation of the performance of the system, at least 10 different noise patterns for each Bit Error Rate (BER) were considered. The location map was assumed to be received uncorrupted.

The method proposed in this paper, Reversible Data Embedding Relation Based Watermarking (RDERBW), was compared to the standard video coding with no watermark (NW) and other state of the art watermarking approaches found in literature such as Forced Even Watermarking (FEW) [17], and Relation Based Watermarking (RBW) [16]. FEW was applied to all transform coefficients except the DC coefficient and RBW was implemented using 10 AC coefficients and $M=6$.

Figure 4 and Figure 5 illustrate the rate distortion curves for *Foreman* and *Mother and Daughter* sequences respectively. The watermarked sequences achieve a lower PSNR relative to the standard. This mainly occurs since the embedded information modifies the image statistics which results in a reduced compression rate. Although the PSNRs of FEW and RDERBW are quite close to each other, in reality the visual quality of FEW is slightly worse than that of RDERBW as shown in Figure 6.

Figure 7 and Figure 8 illustrate the rate distortion curves for the two sequences at a constant Quantization Parameter (QP). These results show that the quality of the video produced by RDERBW at fixed QP outperforms the other two watermarking schemes considered. These plots further show that although RDERBW is a lossless watermark, there is still a slight degradation in PSNR (which increases with decreasing QP) with respect to the non-watermarked sequence. This is mainly because after insertion of the watermark, the statistics of the block are altered. The consequence of this change is that the prediction method chosen for NW and RDERBW may be different. The encoder chooses the prediction method which minimises the

residual. Thus, changing the statistics of the 4×4 block will change the residual information, causing a different prediction method to be chosen from the standard video codec NW. The different prediction methods may produce different residuals, resulting in a slightly different reconstructed image with a different PSNR than that with no watermark. This slight difference in PSNR may in theory be zeroed by inserting the watermark module exactly before the entropy encoding of the residuals, and not within the frame prediction loop. This would, however, come at the cost of an increase in bit rate.

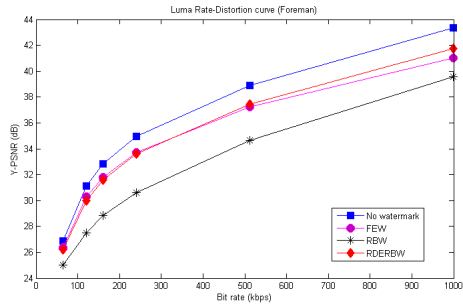


Figure 4 : Rate distortion curve for Foreman (encoder)

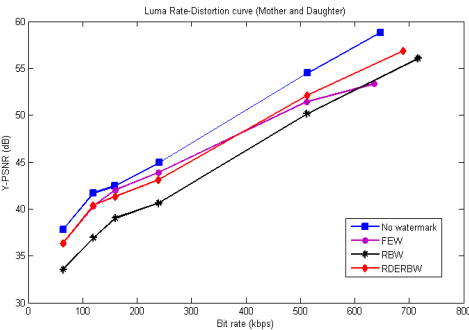


Figure 5 : Rate distortion curve for Mother and Daughter (encoder)

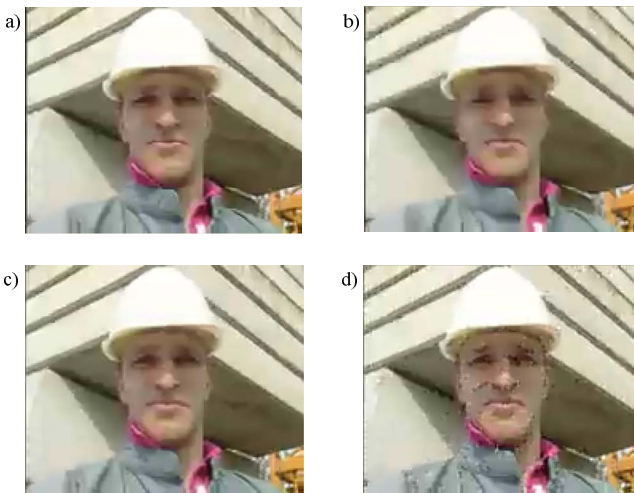


Figure 6 : Foreman frame 2 at 64kbps: a) non-watermarked, b) FEW, c) RDERBW, d) RBW

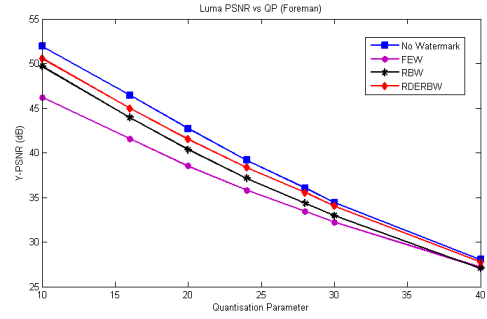


Figure 7 : Foreman PSNR curves at a constant QP

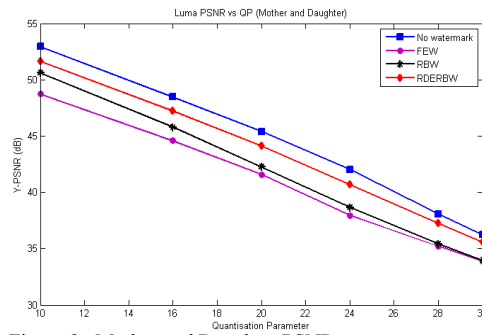


Figure 8 : Mother and Daughter PSNR curves at a constant QP

As shown in Figure 9 and Figure 10, RDERBW outperforms the standard NW scheme, where PSNR gains of up to 2.4 dB were achieved. This is achieved even though the transmitted standard codec achieves a higher PSNR from the encoder side as shown in Figure 4. It is also important to notice that RDERBW outperforms the other state of the art watermarking schemes FEW and RBW. As the BER becomes very small (in the order of 10^{-6}), the errors introduced are minimal and so, as the video sequences start to converge to their transmitted quality, the NW sequence achieves a quality as good as the watermarked sequence. Figures 11 and 12 compare RBW with the proposed reversible scheme RDERBW at 120kbps. These figures clearly show that the reversible watermarking scheme achieves PSNR gains of up to 2.6 dB relative to the standard.

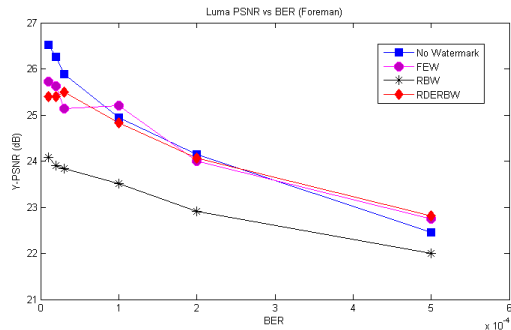


Figure 9 : PSNR curves at 64kbps (Foreman) after decoding

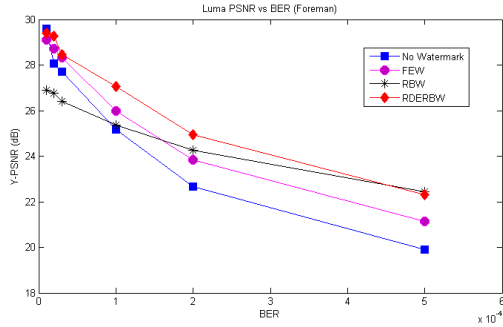


Figure 10 : PSNR curves at 120kbps (Foreman) after decoding

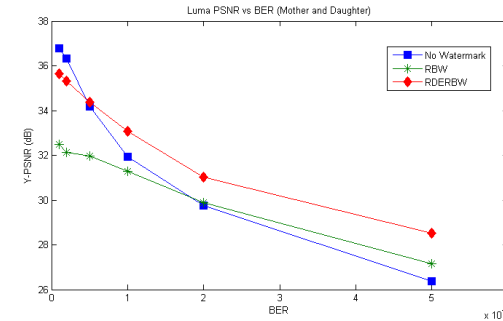


Figure 11 : PSNR curves at 64kbps (Mother and Daughter) after decoding

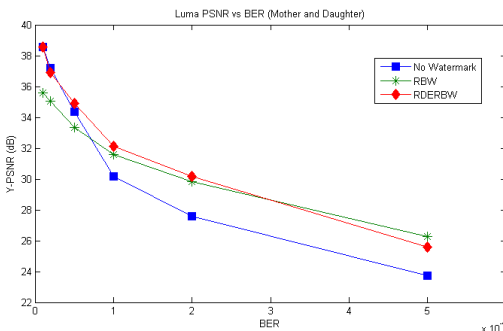


Figure 12 : PSNR curves at 120kbps (Mother and Daughter) after decoding

Figure 13 shows the results of a simulation for the *Mother and Daughter* sequence at a QP of 28, which gives a data rate of 64kbps for NW for this sequence. In this case NW started with a 0.8dB gain over RDERBW from the encoder. At the high BER of 5×10^{-4} RDERBW has a clear gain of 1.7dB over NW, showing the increase in robustness at such high error rates.

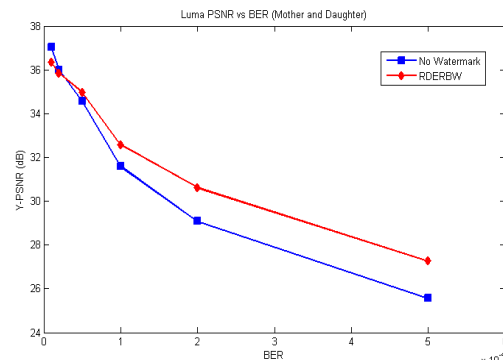


Figure 13 : PSNR curves at QP=28 (Mother and Daughter)

Comparing the Cumulative Distribution Functions (CDFs) of the standard H.264/AVC (NW) sequence, FEW and RDERBW (Figure 14) and considering 25dB as the threshold for good quality video, one observes that NW has a probability of 51% to experience unsatisfactory video quality while for FEW this probability goes down to 45.5%. On the other hand, the probability of experiencing poor video quality for the proposed RDERBW goes down to 28.5%. This further confirms that the proposed RDERBW method does indeed perform better. This superiority is further confirmed by the subjective results shown in Figure 15.

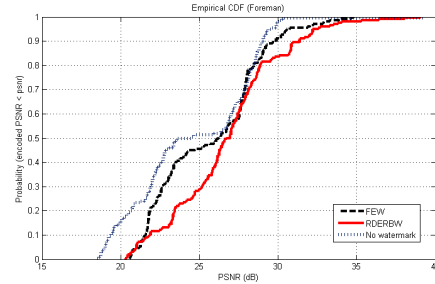


Figure 14: CDF for Foreman at 120kbps and BER of 1×10^{-4}



Figure 15: Corrupted video sequences after concealment at 120kbps (BER 1×10^{-4}) a) no watermark, b) FEW, c) RDERBW

Since watermark insertion modifies the statistics of the 4×4 blocks, it can reduce the compression efficiency resulting in an increase in bit rate. The increase in bit rate with respect to the NW sequence for the proposed scheme was calculated by varying the QP between 10 and 40. RDERBW increased the bit rate by at most 18.4%, and only by 8.1% for 64kbps (refer to Figure 16 and Figure 17). Although FEW decreases the bit rate, as expected, since it inserts more zeros near the end of the 4×4 blocks, the end quality is worse than for RDERBW as shown in the previous simulation results. As regards the uncompressed location map, the maximum bit rate increase incurred was 5.6%. This can be further reduced using lossless compression algorithms such as JBIG.

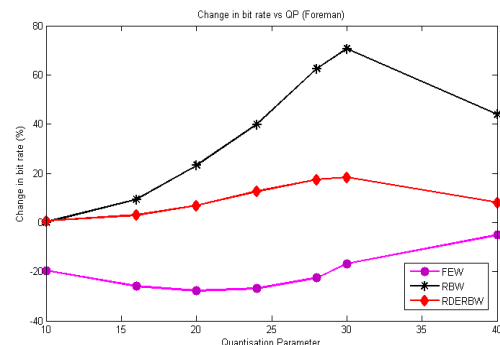


Figure 16 : Change in bit rate caused by watermarks (Foreman)

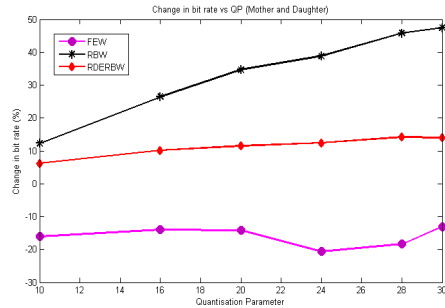


Figure 17 : Change in bit rate caused by watermarks (Mother and Daughter)

V. COMMENTS AND CONCLUSIONS

The proposed RDERBW method presented in this paper outperforms the error resilient strategies of both the standard and the other watermarking based strategies tested from literature. The proposed method achieves a significant improvement in performance at the cost of a minimal increase in complexity. The results give a clear indication that the inclusion of watermarks within the video content increases the robustness of the bit stream and in general results in a better visual quality after decoding. At constant bit rates, it was shown that RDERBW introduces distortion comparable to FEW in the sequence but manages to provide a better visual experience after decoding with transmission errors. At constant QPs, RDERBW introduces the least distortion in the video sequence. Although the lossless watermark implemented requires a location map for data recovery, the rate increase at a constant bit rate required for transmission of this map is at most 5.6%, which can be further compressed, hence justifying the use of such a scheme given the clear gains in quality observed in the simulation results. The complexity of the RDERBW algorithm is minimal and is suitable for real-time wireless applications.

REFERENCES

- [1] T. Stockhammer, and M.M. Hannuksela, "H.264/AVC Video for Wireless Transmission," *IEEE Wireless Commun.*, vol. 12, no. 4, pp. 6-13, Aug 2005.
- [2] Advanced Video Coding for Generic Audiovisual Services, ISO/IEC 14496-10 and ITU-T Rec. H.264, 2005.
- [3] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, pp. 560-576, Jul. 2003.
- [4] S. Wenger, "H.264/AVC over IP," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, pp.645-656, Jul. 2003.
- [5] T. Stockhammer, M. Hannuksela, and T. Wiegand, "H.264/AVC in wireless environments," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, pp.645-656, Jul. 2003.
- [6] M. Etoh, and T. Yoshimura, "Wireless Video Applications in 3G and Beyond," *IEEE Wireless Commun.*, vol. 12, no. 4, pp. 66-72, Aug. 2005.
- [7] L. Superiori, O. Nemethova and M. Rupp, "Performance of a H.264/AVC error Detection algorithm based on syntax check", *Journal of Mobile Multimedia*, vol. 3, no. 4, pp 314 – 330, 2007.
- [8] V. Buttigieg and R. Deguara, "Using Variable Length Error-Correcting Codes in MPEG-4 Video," in *IEEE Int. Symp. on Information Theory*, Adelaide, Australia, Sep. 2005.
- [9] H. Nguyen and P. Duhamel, "Robust Source Decoding of Variable-Length Encoded Video Data taking into account Source Constraints," *IEEE Trans. on Commun.*, vol. 53, no. 7, pp. 1077-1084, Jul. 2005.
- [10] R.A. Farrugia, C.J. Debono, "Robust Transmission of H.264/AVC Sequences using List Decoding and Source Constraints," in *Proc. of the 14th IEEE Mediterranean Electrotechnical Conference (MELECON 2008)*, Ajaccio, France, May 2008.
- [11] R.A. Farrugia, and C.J. Debono, "Enhancing the Error Detection Capabilities of the Standard Video Decoder using Pixel Domain Dissimilarity Metrics," in *IEEE Proc. Int. Conf. EUROCON*, Warsaw, Poland, Sep. 2007.
- [12] S. Ye, X. Lin and Q. Sun, "Content Based Error Detection and Concealment for Image Transmission over Wireless Channels," in *IEEE Proc. Int. Symp. Circuits and Syst.*, Bangkok, Thailand, May 2003.
- [13] E. Khan, S. Lehmann, H. Gunji and M. Ghanbari, "Iterative Error Detection and Correction of H.263 Coded Video for Wireless Networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 12, pp. 1294-1307, Dec. 2004.
- [14] R.A. Farrugia, C.J. Debono, "A Robust Error Detection Mechanism for H.264/AVC Coded Video Sequences Based on Support Vector Machines," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 12, pp. 1766 - 1770, December 2008.
- [15] R.A. Farrugia and C.J. Debono, "Decoding H.264/AVC using Prior Information and Source Constraints," in *IEEE Proc. of Int. Picture Coding Symposium (PCS)*, May 2009.
- [16] O. Nemethova, G.C. Forte and M. Rupp, "Robust Error Detection for H.264/AVC using Relation based Fragile Watermarking," in *Proc. Int. Conf. on Systems, Signals and Image Process.*, Budapest, Hungary, Sep. 2006.
- [17] M. Chen, Y. He and R.L. Lagendijk, "A Fragile Watermark Error Detection Scheme for Wireless Video Communications," *IEEE Trans. on Multimedia*, vol. 7, no. 2, pp. 201-211, Apr. 2005.
- [18] W. Park and B. Jeon, "Error Detection and Recovery by Hiding Information into Video Bitstream using Fragile Watermarking," *Proc. SPIE Visual Commun. and Image Process.*, vol. 4671, pp. 1-10, Jan. 2002.
- [19] C.B. Adsumilli, M.C.Q. Farias, S.K. Mitra and M. Carli, "A Robust Error Concealment Technique using Data Hiding for Image and Video Transmission over Lossy Channels," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 11, pp. 1394-1406, Nov. 2005.
- [20] J. Tian, "Reversible data embedding using a difference expansion," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, pp.890-896, Aug. 2003.
- [21] H.J. Kim, V. Sachnev, Y. Q. Shi, J. Nam and H. Choo, "A novel difference expansion transform for reversible data embedding," *IEEE Trans. Inf. Forensics Security*, vol.3, no.3, pp.456-465, Sep. 2008.
- [22] L. Kamstra and H.J.A.M. Heijmans, "Reversible data embedding into images using wavelet techniques and sorting," *IEEE Trans. Image Process.*, vol.14, no.12, pp.2082-2090, Dec. 2005.
- [23] S.D. Lin, H. Meng and Y. Su, "A novel error resilience using reversible data embedding in H.264/AVC," in *6th International Conference on Information, Communications & Signal Processing*, 2007, pp.1-5, 10-13 Dec. 2007.
- [24] H.264/AVC Software Coordination, "JM Software," ver. 12.2 [Online] Available: <http://iphome.hhi.de/suehring/tml>.
- [25] L-A. Larzon, M. Degermark, S. Pink, L-E. Jonsson, and G. Fairhurst, "The Lightweight User Datagram Protocol (UDP-Lite)," IETF RFC 3282, July 2004.