3D-Deblocking for Error Concealment in Block-Based Video Decoding Systems

Markus Friebe and André Kaup

Chair of Multimedia Communications and Signal Processing,
University of Erlangen-Nuremberg
Cauerstr. 7, 91058 Erlangen, Germany
{friebe,kaup}@LNT.de

Abstract. In this contribution we propose a 3D-deblocking method for macroblock loss recovery in block-based video decoding systems. For a lost macroblock, the motion vector is estimated. Using the estimated motion vector a deblocking filter recovers the lost macroblock by the corresponding motion compensated image samples of the previous frame. Macroblock borders are filtered, if block artifacts are estimated by this deblocking filter. It is shown that in case of transmission errors this restoration technique can successfully be used in block-based video decoding systems.

Index Terms Error concealment, video restoration, video coding and transmission

1 Introduction

When transmitting block-based coded video over error-prone networks, packets may be lost in case of transmission errors. Due to entropy coding, image information is lost until a synchronization marker is reached. Here, visually annoying block artifacts are introduced in decoded video frames. In this contribution we propose a 3D-deblocking method which can reduce those artifacts. There are several different methods for error concealment in block-based video decoding systems. Generally, restoration techniques are either spatial to conceal Intra-coded frames, temporal or spatio-temporal to conceal Inter-coded frames. In this contribution we are investigating a spatio-temporal method for macroblock loss recovery in Intra- and Inter-coded frames.

The Boundary Matching Algorithm (BMA) [1] recovers a motion vector of the lost macroblock by using the information of surrounding errorfree received motion vectors. This is a temporal technique. The Decoder Motion-Vector Estimation Algorithm (DMVE) [2] is also a temporal technique and uses image samples which are immediately neighbored to the lost macroblock. Using these neighboring image samples, the DMVE is looking for the best match in the previous frame. H.264 Intra [3] is using surrounding errorfree or concealed image samples for spatial linear interpolation. A content-based adaptive spatio-temporal method (CABLR) [4] is using temporal image information for macroblock loss recovery, if the temporal information fits well. Otherwise correctly received or concealed spatial neighboring macroblocks are used to recover a lost macroblock. Finally a range constraint is applied on spatially recovered macroblocks.

H.264 Intra introduces blurred image areas because of spatial linear interpolation. BMA and DMVE are both temporal methods and introduce block artifacts in case of object occlusions and uncovering. CABLR conceals either temporal or spatial. Through this switching scheme, either blurred image areas or block artifacts are visible. The proposed 3D-deblocking is a joint spatio-temporal method which outperforms the reference methods both in objective and subjective video quality.

2 3D-Deblocking

The block diagram of the proposed 3D-deblocking is shown in Fig. 1. The 3D-deblocking method is estimating a motion vector for the lost macroblock. The deblocking unit uses the estimated motion vector for temporal recovery and additionally spatial neighboring macroblock samples of the current frame for deblocking.

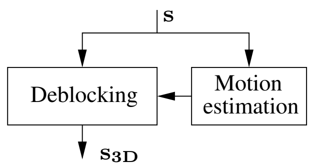

Fig. 1. Block diagram of 3D-deblocking.

2.1 Motion Estimation

The motion vector of the lost macroblock $\tilde{mv}_m$ and $\tilde{mv}_n$ is estimated similar to DMVE [2] by minimizing a weighted sum of absolute differences SAD_w. Eq. (1) and Eq. (2) show, how SAD_w is calculated and how the motion vector is estimated from the minimum SAD_w. Image samples s in the current frame k are compared to motion compensated image samples in the previous frame $k-1$. A weighting function w represents correct ($w = 1$) and incorrect ($w = 0$) received image samples in the current frame k. In Fig. 2, the area marked

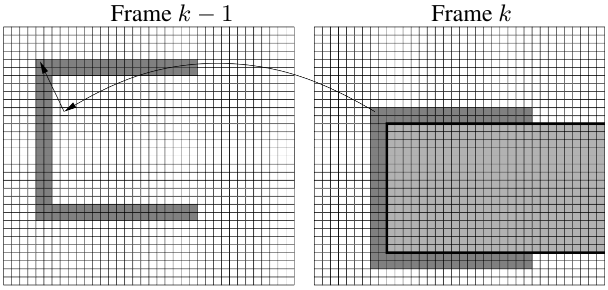

Fig. 2. Left: Frame $k-1$. Right: Frame k. The area marked in dark-gray represents the $SAD_w[mv_m, mv_n]$ evaluation area with $w[m,n] = 1$. The area marked in light-gray represents the lost image samples with $w[m,n] = 0$. Both for the left most consecutive macroblock loss.

in dark-gray represents correct and the area marked in light-gray represents incorrect received image samples for the left most consecutive macroblock loss. The small arrow in Fig. 2 shows one of the evaluated motion vectors. The size of a lost macroblock is given by M and N in row and column direction respectively. m_0 and n_0 represents the left top image sample in the lost macroblock. Hence we calculate the weighted sum of absolute differences by

$$SAD_w[mv_m, mv_n] = \sum_{m=m_0-2}^{m_0+M+1} \sum_{n=n_0-2}^{n_0+N+1} w[m,n] \cdot |s[m,n,k] - s[m+mv_m, n+mv_n, k-1]| \quad (1)$$

and the best matching motion vector by

$$[\tilde{mv}_m, \tilde{mv}_n] = \arg \min_{mv_m, mv_n} (SAD_w[mv_m, mv_n]). \quad (2)$$

The difference between DMVE and this method is the weighting function. DMVE treats concealed areas as correctly received image samples by setting $w = 1$. In this approach, concealed areas are treated as incorrectly received areas by setting $w = 0$. After estimating the lost motion vector, the missing macroblock can be restorated by the corresponding motion compensated image information of the previous frame $k-1$.

In case of object occlusions and uncovering, motion compensated image samples of frame $k-1$ may not fit very well and some kind of block artifacts may be introduced. Therefore a deblocking filter is further used for reducing block artifacts.

3 Deblocking

A deblocking method based on [5] can reduce these artifacts by using the motion compensated macroblock image information of frame $k-1$ and spatial neighboring image samples of frame k. For deblocking, the considered image samples are either neighboring spatial or temporal image information. For detection and classification of block artifacts, image samples at macroblock borders are analyzed. Based on this classification, image samples at macroblock borders are adapted such that these samples represent the statistic of the neighboring image area. In Fig. 3, the block diagram of the deblocking method is shown. First, temporal error concealment is copying the corresponding motion compensated macroblock of the previous frame into the lost macroblock area:

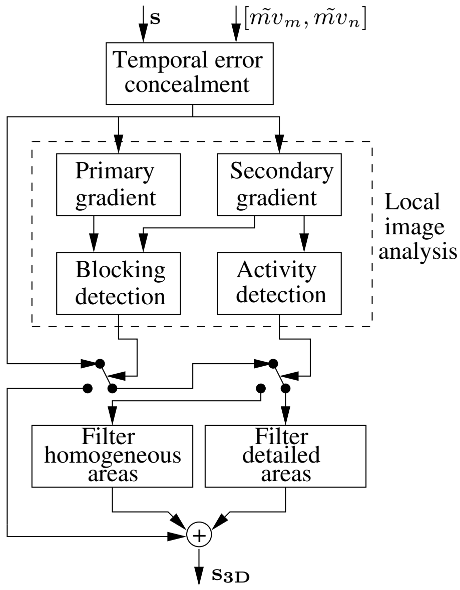

Fig. 3. Block diagram of the deblocking filter.

$$s[m,n,k] = s[m+\tilde{mv}_m, n+\tilde{mv}_n, k-1]. \quad (3)$$

$m = m_0, ..., m_0+M-1$ and $n = n_0, ..., n_0+N-1$ represents the lost macroblock area. Two different gradients, a primary gradient $\bigtriangledown_p$ and a secondary gradient $\bigtriangledown_s$, are computed for analyzing a macroblock border. $\bigtriangledown_p$ represents the sum of absolute differences along a macroblock border and $\bigtriangledown_s$ the mean value of two sum of absolute differences within neighboring macroblocks. Eq. (4) and Eq. (5) are examples for analyzing the top macroblock border.

$$\bigtriangledown_p = \sum_{n=0}^{N-1} |s[m_o-1, n_o+n, k] - s[m_o, n_o+n, k]| \quad (4)$$

$$\nabla_s = 0.5 \cdot \sum_{n=0}^{N-1} |s[m_o - 2, n_o + n, k] - s[m_o - 1, n_o + n, k]| + 0.5 \cdot \sum_{n=0}^{N-1} |s[m_o, n_o + n, k] - s[m_o + 1, n_o + n, k]| \quad (5)$$

Based on ∇_p and ∇_s, a block detection unit estimates if block artifacts appear along a macroblock border. In uncompressed images, the ratio between primary gradient ∇_p and secondary gradient ∇_s is approximately one.

$$\frac{\nabla_p}{\nabla_s} \approx 1 \quad (6)$$

In compressed images, because of the block based coding structure, this ratio is greater than one. Block artifacts are decided to be present in case this ratio exceeds a defined threshold T_{block}.

$$\frac{\nabla_p}{\nabla_s} > T_{block} \quad (7)$$

In this case, two different deblocking filters for homogeneous or detailed areas are used for filtering the investigated macroblock border.

Based on the secondary gradient ∇_s, an activity detection unit estimates what kind of deblocking filter is used for deblocking the considered macroblock border. The activity detection unit compares the secondary gradient with a threshold $T_{activity}$. In case

$$\nabla_s > T_{activity}, \quad (8)$$

the deblocking filter for detailed areas is selected. Otherwise, the deblocking filter for homogeneous areas is used.

3.1 Deblocking Filter for Homogeneous Areas

For estimated homogeneous macroblock borders, a filter for homogeneous areas with operating distance of length 4 is used. Fig. 4 illustrates how this filter works. The step function located at the macroblock boundary (samples marked in gray) is formed to a ramp function (samples marked in white). At the considered macroblock border, each boundary position is filtered. At each boundary position, the step height Δ is evaluated. A factor of $1/5$ is multiplied with the measured step height Δ and added or subtracted to or from the considered input sample. An edge at a macroblock border is estimated, if the absolute value of Δ is higher than a threshold T_{edge}. In this case, the filter for deblocking homogenous areas is switched of.

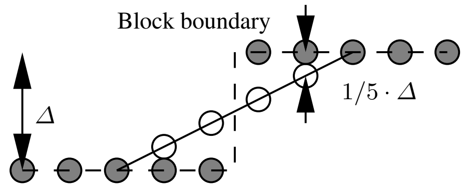

Fig. 4. Deblocking homogenous area.

3.2 Deblocking Filter for Detailed Areas

For estimated detailed macroblock borders, a filter with a smaller operating distance of two pixels is used. The operating distance is smaller than for filtering homogenous areas, because details should be preserved in detailed areas. At each macroblock border position, a 4-point 1D-DCT upright and centered to the macroblock border is done. In Fig. 5 the gray marked samples show which samples are used for the 4-point 1D-DCT in case of a vertical macroblock border. The highest coefficient

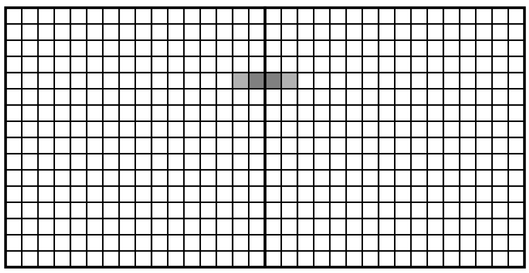

Fig. 5. Deblocking detailed area.

index of the 4-point 1D-DCT is responsible for appearing block artifacts [5]. This coefficient is decreased by a factor α. The inverse 4-point 1D-DCT is only computed for the two dark-gray samples in Fig. 5. In case the absolute difference between the two recovered samples after filtering is higher than $\Delta/2$, the deblocking filter for detailed areas is switched off. Otherwise, the samples obtained from the inverse 4-point 1D-DCT are used for the samples marked in dark-gray.

4 Simulation Results

For simulations we consider uncompressed video frames of size CIF and 25 frames per second. Consecutive macroblocks are lost in every 5th video frame. In each third macroblock row, 18 consecutive macroblocks are lost starting with the third macroblock in one row. The following parameter set for 3D-deblocking is used for this simulation: $T_{block} = 1.5$, $T_{activity} = 250$ for the luminance, $T_{activity} = 125$ for the chrominances, $T_{edge} = 100$ and $\alpha = 0.03$. For BMA the necessary

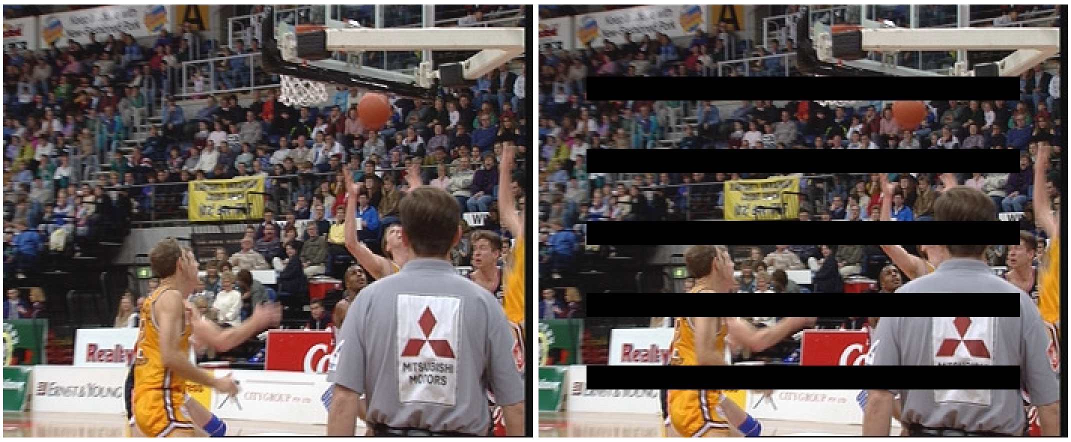

Fig. 6. Left: Original image. Right: Consecutive 16×16 macroblock loss.

Fig. 7. Left: H.264 Intra [3]. Right: DMVE [2].

Fig. 8. Left: CABLR [4]. Right: 3D-deblocking.

surrounding motion vectors has to be pre-computed and DMVE uses a two line neighborhood.

The colorspaces Y, U and V are filtered separately for H.264 Intra. For DMVE, CABLR and 3D-deblocking, the motion vector is only estimated for the luminance and further used for the chrominances. If the luminance in CABLR is recovered spatially, then also the chrominances are recovered spatially. For 3D-deblocking, the deblocking unit is used for Y, U and V separately.

In Fig. 6 a frame of the *Basketball* sequence is shown without errors at the left hand side and with consecutive macroblock loss at the right hand side. The results for H.264 Intra are shown at the left hand side and for DMVE at the right hand side in Fig. 7. H.264 Intra is introducing blurred image areas because of the linear interpolation. DMVE introduces some block artifacts in case of object occlusions and uncovering, as seen at the arm of the left basketball player. Also CABLR is introducing some block artifacts, as seen in Fig. 8 at the left hand side. For the proposed 3D-deblocking method, the recovered areas don't look blurred and appearing block artifacts are further reduced. This can be seen in Fig. 8 at the right hand side.

In Table 1 mean luminance PSNR results for different videos and error concealment methods are shown. The PSNR values are only evaluated at the lost macroblock areas. Spatial methods like H.264 Intra yields the lowest PSNR values. CABLR performs well in video sequences with less motion and homogeneous image areas like *Foreman*. DMVE yields the highest results for the *Coastguard* sequence. 3D-deblocking on average achieves the highest PSNR results over all video sequences.

In Fig. 9 luminance PSNR results for the sequence *Basketball* are shown. H.264 Intra achieves for every evaluated frame the smallest PSNR results. The results obtained from CABLR are for 4 of 20 frames better than the results from DMVE. 3D-deblocking yields for 18 of 20 frames the highest PSNR results.

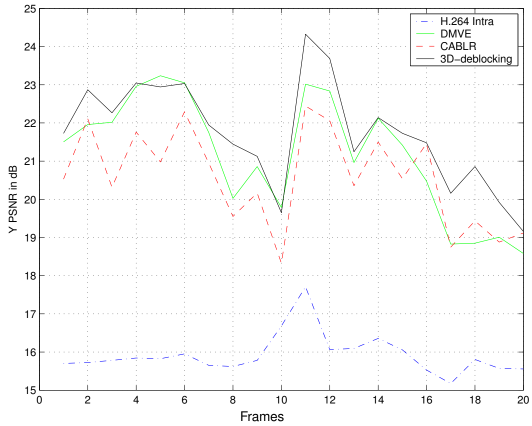

Fig. 9. PSNR results for evaluated *Basketball* frames.

Table 1. Mean Y PSNR results in dB.

Video	*Fore-man*	*Basket-ball*	*Sales-man*	*Flower garden*	*Coast-guard*
H.264 Intra [3]	21.99	15.93	22.26	15.44	17.74
Zero MV	27.02	15.60	32.91	16.48	22.59
BMA [1]	30.76	20.27	33.41	23.31	26.32
DMVE [2]	32.75	21.16	32.60	26.51	**28.46**
CABLR [4]	33.15	20.58	32.63	25.86	27.68
3D-deblocking	**33.46**	**21.73**	**33.49**	**27.23**	27.86

5 Summary

In this paper we presented a 3D-deblocking method for macroblock loss recovery in block-based video decoding systems. The simulation results show that for macroblock loss in uncompressed video frames this method outperforms reference methods like H.264 Intra, BMA, DMVE, and CABLR both in subjective and objective video quality. For compressed video data, in case of transmission errors, appearing artifacts are propagated from erroneously decoded Intra-coded frames to Inter-coded frames. 3D-deblocking is well suited for recovering lost macroblocks. Using 3D-deblocking for Intra-coded frames, error propagation to Inter-coded frames can be reduced. Additionally, using 3D-deblocking for Predictive-coded frames, error propagation to Bidirectional-coded frames can be reduced.

References

1. Lam, W.M., Reibman, A.R., Liu, B.: Recovery of Lost or Erroneously Received Motion Vectors. In: Proc. Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP), Minneapolis (1993) V417–V420
2. Zhang, J., Arnold, J.F., Frater, M.R.: A Cell-Loss Concealment Technique for MPEG-2 Coded Video. In: IEEE Trans. Circuits Syst. Video Technol. Volume 10. (2000) 659–665
3. Wang, Y.K., Hannuksela, M.M., Varsa, V.: The Error Concealment Feature in the H.26L Test Model. In: Proc. Int. Conf. on Image Processing (ICIP), Rochester (2002) 729–732
4. Park, J., D.-C. Park, R.J.M., El-Sharkawi, M.A.: Content-Based Adaptive Spatio-Temporal Methods for MPEG-Repair. In: IEEE Trans. Image Process. Volume 13. (2004) 1066–1077
5. Piastowski, P.: System for Decoder-Independent Reduction of Block Artifacts (System zur Decoder-unabhängigen Reduktion von Blockartefakten). In: ITG Fachbericht Elektronische Medien: 11. Dortmunder Fernsehseminar. Number 188, VDE-Verlag (2005) 215–218