CODING ARTIFACT REDUCTION USING NON-REFERENCE BLOCK GRID VISIBILITY MEASURE

I. O. Kirenko, R. Muijs, L. Shao
Philips Research Laboratories, Eindhoven, The Netherlands
Ihor.Kirenko@philips.com, Remco.Muijs@philips.com, L.Shao@philips.com

ABSTRACT

In this work a new method is proposed for coding artifact reduction of MPEG compressed video sequences. The method makes use of a simple cost-effective technique that allows the block grid position and its visibility to be determined without the need for access to the coding parameters. This information, combined with the results of local spatial analysis of luminance and chrominance components of a decoded image, is used to effectively suppress coding artifacts while preserving the sharpness of object edges. Results of our experiments confirm the high efficiency of the proposed approach.

1. INTRODUCTION

MPEG-2 provides lossy compression by quantizing coefficients of 8x8 DCT transform taken over the image, so that the number of bits needed to code the image is reduced. A coarse quantization of DCT coefficients can cause various types of coding artifacts in decoded video sequences. Blockiness, ringing, mosquito noise and quantized film grain are the major artifacts [1].

Subjective experiments have indicated that blockiness is the most annoying coding artifact at low to moderate bitrates and is highly correlated with the overall perceived quality of MPEG-2 encoded video [1]. Optimal suppression of blocking artifacts requires information about both the position and the visual strength of the discontinuities. However, consumer electronics devices do not always have an access to encoded bit-streams, and thus to coding parameters. The most common example of such devices is a TV set, which receives already decoded video signals from a DVD player, a set-top-box, or a cable network. Therefore, there is a need for post-processing algorithms that can detect and remove coding artifacts based only on an analysis of the decompressed video signal.

Different techniques have been studied extensively in the past few years aimed at the reduction of blockiness and ringing in decoded sequences [2], [3], [4]. The state-of-the-art methods are able to significantly reduce the visibility of coding artifacts. However, most of those methods either use a bit-stream information (e.g. grid position, quantization parameters), or require external parameters to control the strength of artifacts suppression.

The artifact reduction method, proposed here, does not use coding parameters, and therefore it can be applied in a system, where the encoded bit-stream is not available. Moreover, the algorithm adapts the filtering automatically based on a local spatial analysis and a block grid visibility. Thus, no external control parameters are required.

The proposed method consists of two main parts:

- Detection of blocking artifacts;
- Adaptive low-pass filtering of the detected artifacts.

The first part of the proposed algorithm is described in the next section. Section 3 describes the local spatial analysis and adaptive low-pass filtering during the second part of the algorithm. Section 4 is dedicated to the analysis of experimental results. Section 5 concludes the paper.

2. DETECTION OF BLOCKINESS

In order to preserve sharpness in decoded video sequences and at the same time to remove coding artifacts, we should separate pixels, which contain visible artifacts from pixels that belong to object edges or texture. This separation is achieved by detecting a block grid position in the decoded image and analyzing local visibility of block edges.

The visual strength of a block edge is predominantly affected by the magnitude of the edge gradient and the spatial activity in the direct vicinity of the block border [5]. In other words, the visibility of a block edge is determined by the contrast between the local gradient and the average gradient of the adjacent pixels. Based on the principle that block discontinuities can be spotted as edges that stand out from the spatial activity in their vicinity we propose a simple, efficient algorithm for detection of the grid position and estimation of a block edge visibility [6].

1-4244-0367-7/06/$20.00 ©2006 IEEE

In the following, we discuss the detection of vertical block edges, but identification of horizontal artifacts is accomplished in a similar fashion. Consider an image ***I*** with elements $Y_{i,j}$, where *i* and *j* denote the pixel and line position, respectively. To express the similarity between the local gradient and its spatial neighbors, we introduce the normalized horizontal gradient $\boldsymbol{D_{H,norm}}$ as the ratio of the absolute gradient and the average gradient calculated over *N* adjacent pixels to the left and to the right:

$$D_{H,norm}(i,j) = \frac{\left|Y_{i+1,j} - Y_{i,j}\right|}{\dfrac{1}{2N} \sum\limits_{n=-N \ldots N, n \neq 0} \left|Y_{i+n+1,j} - Y_{i+n,j}\right|} \quad (1)$$

Because block edges occur at regular intervals in the horizontal or vertical direction, they can be further highlighted by summing $\boldsymbol{D_{H,norm}}$ over all image lines *nl*:

$$S_H(i) = \sum_{j=1}^{nl} D_{H,norm}(i,j). \quad (2)$$

The presence of blocking artifacts will result in pronounced maxima in S_H. The above procedure is illustrated for the image *branch* displayed in Figure 1a. Although blocking artifacts are difficult to identify in the original image, the periodic structure of the encoding grid is clearly revealed in the horizontal accumulator S_H shown in Figure 1b. The size and offset of the grid can be readily extracted from this signal by means of conventional histogram analysis of the peak locations.

The visual strength of the blocking artifacts can be determined by averaging S_H over the block edge and intermediate positions. The Blocking Strength (BS) for the whole frame is then defined as:

$$BS = \frac{\bar{S}_H(block)}{\bar{S}_H(non-block)}, \quad (3)$$

where $\bar{S}_H(block)$ and $\bar{S}_H(non-block)$ denote the average value of S_H at the block edge and intermediate positions, respectively. The BS parameter is defined for horizontal and vertical directions.

The accuracy of the objective blockiness metric *BS* was assessed using the LIVE Image Quality Assessment Database [7], that consists of 169 JPEG encoded images and associated mean quality scores (MQS). Figure 1c displays the relation between the objective metric *BS* and the subjective MQS. The Pearson correlation coefficient of these data amounts to 0.92.

In spite of the simplicity, the outlined approach provides an accurate prediction of the subjective quality and robustly determines block-grid position for intra- as well as inter-coded frames.

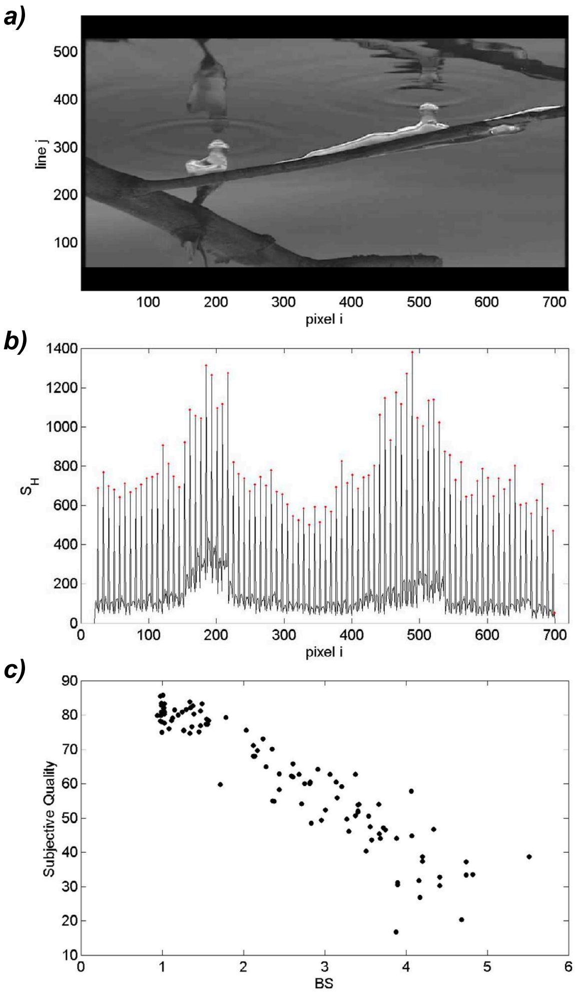

Figure 1: Detection of the position and visibility of the coding grid. Shown are (a) the image *branch* encoded at a bit-rate of 2Mb/s, (b) the horizontal accumulator S_H computed using equations (1) and (2) and (c) the the objective blockiness metric BS vs. MQS for JPEG encoded images from the LIVE database.

3. REDUCTION OF CODING ARTIFACTS

One of the most efficient and simplest way to suppress coding artifacts is by means of adaptive spatial low-pass filtering. To optimally preserve the image sharpness we have to adapt the strength of a low-pass filter to a local visibility of the block grid and to preserve object edges located close to or directly at the block grid from being blurred. The blockiness detection method described in the previous section provides us with the information about the grid position and its global visibility metrics. The local visibility of block edges is estimated by analyzing both luminance and chrominance components of decoded video frames over the detected block edges and neighboring areas.

During the analysis we make use of two observations:

- A block grid of chrominance components is located at the same position as a luminance grid.
- Difference in chrominance levels between two objects is higher than a difference between two blocks located within the same object.

Based on the results of our spatial analysis, we define three levels of local visibility of artefacts, which determine three modes of luminance filtering. One-dimensional adaptive low-pass filtering is executed first in the horizontal direction, followed by vertical filtering. The filtering is applied in raster-scan order through the whole frame.

The de-blocking process is explained for the case of a vertical block edge, when a decoded video signal has a format 4:2:2 (Figure 2). Horizontal de-blocking is accomplished in a similar fashion.

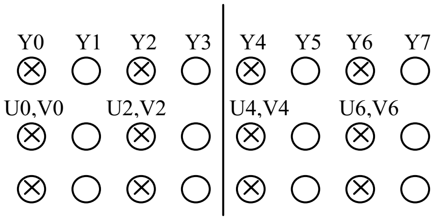

Figure 2. Spatial sampling of luminance (Y) and chrominance (U,V) according to 4:2:2 format and the analyzed edge between pixels Y3 ,Y4.

The purpose of the first two modes of our artifact reduction method is to reduce high-frequency components of a block grid.

The **first filtering mode** is used against highly visible blockiness, characterized by a local maximum amplitude of a luminance discontinuity, and nonzero discontinuities in chrominance components:

$$MAX(D1, D2, D3, D4) < |Y3 - Y4| < Tlum \qquad (4)$$
$$0 < |U2-U4| + |V2-V4| < Tchr,$$

where $D1 = |Y1-Y2|$; $D2 = |Y3-Y2|$; $D3 = |Y4-Y5|$; $D4 = |Y5-Y6|$. Usually $Tchr=6$.

Threshold *Tlum* depends on the Block Strength (BS) value for the processed frame:

$$Tlum = BS*10$$

The conditioning of chrominance gradients protects sharp vertical or horizontal object edges, which are coincided with the grid, from being destroyed.

The filtering in this mode is the strongest:

$$Y3' = (Y2 + Y3 + Y4)/3,$$
$$Y4' = (Y3 + Y4 + Y5)/3, \qquad (5)$$

where Y3’, Y4’ are output luminance pixel values.

If |Y1-Y4 |< Tlum /2, then we can filter pixels Y1, Y2, which are not adjusted to the block edge:

$$Y2' = (Y1 + Y2 + Y4)/3,$$
$$Y1' = (3*Y1 + Y4)/4, \qquad (6)$$

The similar condition is estimated for the other side of the block edge. If |Y6-Y3 |< Tlum/2, then we can filter pixels Y5, Y6:

$$Y5' = (Y6 + Y5 + Y3)/3,$$
$$Y6' = (3*Y6 + Y3)/4, \qquad (7)$$

The filtering of Y1,Y2 and Y5,Y6 pixels provides better smoothing of severe blockiness.

The **second mode** is applied to block edges with the local visibility, which is lower than in the first mode:

$$MAX(D2, D3) < |Y3 - Y4| < Tlum \qquad (8)$$
$$|U2-U4| + |V2-V4| < Tchr,$$

In this case we apply softer filtering to pixels Y3,Y4 :

$$Y3' = (Y2 + 2*Y3 + Y4)/4, \qquad (9)$$
$$Y4' = (Y3 + 2*Y4 + Y5)/4$$

If conditions |Y6-Y3 |< Tlum/3 and |Y1-Y4 |< Tlum /3 are true, then pixels Y2 and Y5 are filtered as well:

$$Y2' = (Y1 + 2*Y2 + Y4)/4, \qquad (10)$$
$$Y5' = (Y6 + 2*Y5 + Y3)/4,$$

Filtering of chrominance is performed only if one of the first two luminance filtering modes is executed. For chrominance processing, we use short filters:

$$U2' = (2*U2 + U4)/3 \quad V2' = (2*V2 + V4)/3 \qquad (11)$$
$$U4' = (U2 + 2*U4)/3 \quad V4' = (V2 + 2*V4)/3$$

Depending on the quality of compressed video sequences and requirements to the implementation complexity of the deblocking algorithm, the chrominance filtering can be omitted.

The **third filtering mode** is applied to pixels, which are not filtered in the first two modes, or in other words, which are not located on a highly visible block grid. The goal of the third mode is to reduce residual (low-pass) components of blockiness as well as mosquito noise and quantized film grain. Those artifacts exist in both smooth and textured regions; however, their visibilities are different. Although the low-pass filtering used in the third mode is less strong than in the previous modes, an image texture may be blurred, or small object edges distorted if the filtering is applied to them. To avoid smoothing of textured regions we do not filter pixels within spatially busy areas. Thus, the spatial activity around the analyzed pixel pair Y3, Y4 as well as the value of the gradient $|Y3 - Y4|$ should be low:

$$|Y3 - Y4| < Tlum/4 \qquad (12)$$

We filter pixel Y3 in the third mode if the following conditions hold:

$$D1 < Tlum /5 \ \&\& \ D2 < Tlum /4 \ \&\& \ D3 < Tlum /5 \qquad (13)$$

The pixel Y4 is filtered if the conditions below are true:

$$D2 < Tlum /5 \ \&\& \ D3 < Tlum /4 \ \&\& \ D4 < Tlum /5 \qquad (14)$$

In the third mode only the luminance pixels Y3 and Y4 are processed using the soft filters:

$$Y3' = (Y1 + Y2 + 6*Y3 + Y4 + Y5)/10 \qquad (15)$$
$$Y4' = (Y2 + Y3 + 6*Y4 + Y5 + Y6)/10$$

4. RESULTS OF EXPERIMENTS

The efficiency of the proposed artifact reduction method was evaluated using test sequences with SD and SIF resolutions compressed by an MPEG-2 coder at different bit-rates. From the various state-of-the-art algorithms, we choose two methods, which provide best results and represent two different approaches to artifact reduction. The first method is a combination of two-mode deblocking [2] (which uses a grid position) and deringing filter proposed by A. Kaup [3]. The second algorithm is an efficient but very expensive technique of A. Nostratinia [4]. PSNR results of the benchmarking are shown in Table 1.

TABLE 1
PSNR OF PROCESSED TEST SEQUENCES

bit-rate Mbit/s	algorithm [4]	algorithm [2]+[3]	proposed algorithm
"Stefan", SIF			
0.1	22.75	22.85	22.88
0.25	22.98	23.08	23.08
0.5	23.54	23.58	23.60
1.0	26.49	26.07	26.46
"Vanessa", SD			
2.0	29.21	29.46	29.62
3.0	31.00	30.95	31.40
4.0	32.45	32.00	32.78
5.0	33.67	32.78	33.93
"Porsche", SD			
2.0	30.08	30.39	30.51
3.0	31.92	31.89	32.26
4.0	33.41	32.92	33.64
5.0	34.64	33.66	34.68

TABLE 2
BIM OF TEST SEQUENCES

bit-rate Mbit/s	method	H BIM	V BIM
"Stefan", SIF			
0.50	algorithm [4]	1.66	2.79
	algor [2]+[3]	0.66	1.10
	Proposed	1.20	1.80
1.00	algorithm [4]	1.35	1.92
	algor [2]+[3]	0.52	0.74
	proposed	1.04	1.39
"Vanessa", SD			
3.0	algorithm [4]	2.71	3.11
	algor [2]+[3]	1.05	1.05
	Proposed	1.23	1.50
4.0	algorithm [4]	2.14	2.46
	algor [2]+[3]	0.96	0.98
	proposed	1.01	1.24

Besides PSNR, the Block Impairment Metric (BIM) [5] was used for evaluation, which provides an objective metric of blockiness in vertical (VBIM) and horizontal (HBIM) directions (Table 2).

Figure 3 shows a part of a decoded frame of the sequence "Foreman" (SIF, 250 kbit/s) before and after our artifacts reduction. In the picture of Figure 3 (b), blockiness is reduced significantly, while sharp object edges are preserved.

5. CONCLUSIONS

In this paper a new method of coding artifact reduction is presented. The method does not require any coding parameters and is based on the non-reference blocking artifact measure combined with the spatial analysis of luminance and chrominance components of a decoded picture. The algorithm is computationally inexpensive and requires only five lines memory. According to the results of experiments, our algorithm is able to efficiently reduce blockiness, mosquito and quantized noise in MPEG-2 decoded sequences without blurring of image texture and destruction of object edges.

(a)

(b)

Figure 3. Decoded frame before (a) and after (b) coding artifact reduction

REFERENCES

[1] C.C. Koh, S.K. Mitra, J.M. Foley, and I.E.J. Heynderickx, "Annoyance of individual artifacts in MPEG-2 compressed video and their relation to overall annoyance", Proceedings of the SPIE, San Jose, CA., January, 2005

[2] M. Yuen and H.R. Wu, "Reconstruction artifacts in digital video compression", Proc. Digital Video Compression conf. vol. 2419, pp. 455-465, February, 1995.

[3] J.Y. Tham, S. Ranganath, A.A. Kassim, "Non-iterative adaptive post-processing for ringing artifact suppression in compressed images", SCI'98/ISS'98, Vol 3, pp.195-202, 1998.

[4] A. Nostratinia, "Denoising of JPEG images by re-application of JPEG", Journal of VLSI Signal Processing, vol.27, pp. 69-79, 2001.

[5] H.R. Wu and M. Yuen, "A Generalized Block-Edge Impairment Metric for Video Coding", IEEE Signal Processing Letters, Vol.4, No. 11, November, 1997

[6] R. Muijs, I. Kirenko, "A No-reference Blocking Artifact Measure for Adaptive Video Processing", EUSIPCO 2005, 13nd Europ. Sign. Proc. Conf., Turkey, 2005.

[7] H.R. Sheikh, Z. Wang, L.Cormack, and A.C. Bovik, :LIVE Image Quality Assessment Database", http://live.ece.utexas.edu/research/quality