ENHANCEMENT OF COMPRESSED VIDEO SIGNALS USING A LOCAL BLOCKINESS METRIC

Ihor O. Kirenko[1], *Ling Shao*[1], *Remco Muijs*[1], *Adrian Nakonechny*[2]
[1]Philips Research Laboratories, Eindhoven, The Netherlands
[2] Lviv Polytechnic National University, Lviv, Ukraine

ABSTRACT

This paper presents a new method for enhancement of the visual quality of compressed video sequences. The algorithm detects the location of a block grid, reduces coding artifacts and enhances sharpness of image details using integrated filtering that ensures low implementation cost and high quality. The results of our experiments show the high efficiency of the proposed approach.

Index Terms— *image enhancement, video coding, video signal processing, image analysis*

1. INTRODUCTION

Standard "lossy" video compression techniques such as JPEG or MPEG reduce the amount of image data that must be stored or transmitted. Due to a block-based quantisation during compression, image artefacts may appear in the decompressed images. Blocking, apparent as a high-frequency grid in luminance and chrominance and blurring of object edges are among the most visible artifacts. Due to a growing variety of digitally video coding techniques there is a need for flexible and adaptive algorithms for picture quality enhancement of compressed video signals.

The straightforward approach to improve the quality of decompressed video signals would be to cascade several image enhancement functions, including at least one coding artifact reduction method, analogue noise reduction, contrast and sharpness enhancement algorithms. Such cascades do not always work efficiently due to mutually antithetic processing of its components [1]. For example, a sharpness enhancement algorithm might enhance image details, which were reduced by an artifact reduction algorithm earlier in a video processing chain. Therefore, it is necessary to design an algorithm that is able to address several image enhancement aspects within one integrated process [1].

2. VISUAL QUALITY RECONSTRUCTION

The proposed algorithm consists of several components: detection of the block grid, estimation of the local grid visibility, pre-filtering of the detected block edges, and combined de-ringing, analogue noise reduction and sharpness enhancement based on local analysis of entropy and dynamic range. The block-scheme of the algorithm is shown in Fig. 1.

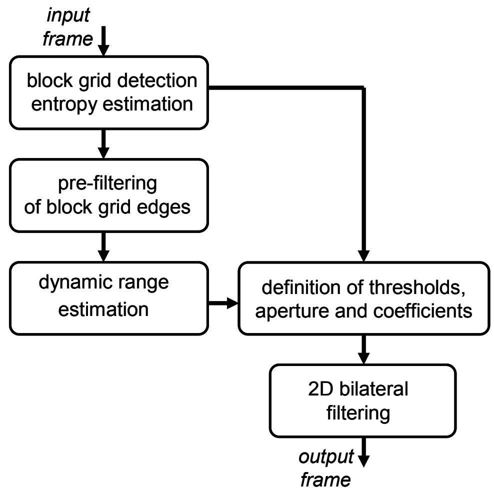


Fig 1. Block-scheme of the proposed algorithm.

2.1. DETECTION OF BLOCKINESS

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

Based on the principle that block discontinuities can be spotted as edges that stand out from the spatial activity in their vicinity [2] we propose a simple, efficient algorithm for detection of the grid position [3]. 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 with elements $V_{i,j}$, 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 D_i as the ratio of the absolute gradient and the average gradient calculated over *N* adjacent pixels to the left and to the right:

$$D_i = \frac{\left|V_{i+1,j} - V_{i,j}\right|}{\frac{1}{2N}\sum_{n=-N...N}\left|V_{i+n+1,j} - V_{i+n,j}\right|} \quad (1)$$

In order to reduce the influence of a texture on the accuracy of block gird detection, step function originated from the normalized gradients (1) is generated and analyzed further:

$$S_i = \begin{cases} 1, & D_i > t \\ 0, & D_i \le t \end{cases} \qquad (2)$$

The resulted values of step function S_i are shown in Fig.2(b). Because block edges occur at regular intervals in the horizontal or vertical direction, they can be further highlighted by summing D_i over all image lines *nl*:

$$Ah_i = \sum_{j=1}^{nl} S_j \qquad (3)$$

The presence of blocking artifacts will result in pronounced maxima in Ah. Although blocking artifacts are difficult to identify in the original image at Figure 2a, the periodic structure of the encoding grid is clearly revealed in the horizontal accumulator Ah, shown in Figure 2c. The size and offset of the grid is extracted from this signal by means of conventional histogram analysis of the peak locations.

Despite the simplicity, the outlined approach provides an accurate and robust detection of block-grid position for intra- as well as inter-coded frames without access to coding parameters. According to the experiments [3], the algorithm is reliable even if video was spatially scaled after decoding.

The full 2D grid position can be established by applying the above analysis in both the horizontal and vertical directions. After detecting the position of block artifacts, their visibility can be estimated.

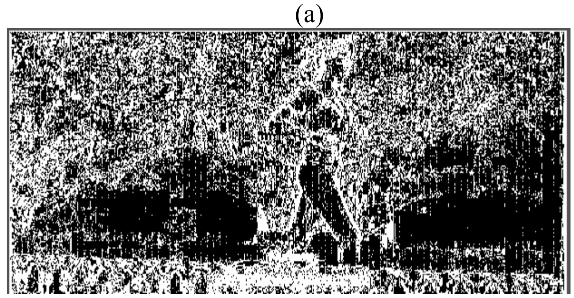
Block artifacts generally originate from:

- high gradient of pixels across the grid relatively to gradients of pixels inside blocks;
- abrupt changes of pixel intensity trends between adjacent blocks.

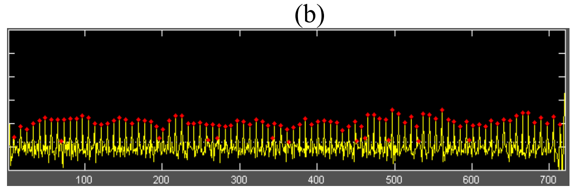
Conventional blockiness metric [2] estimates a block visibility based on an approach similar to (1) used in our algorithm for grid detection. However, blockiness is annoying not always due to a high frequency grid. The blockiness depicted in Figure 3 would be estimated by conventional blockiness metrics as non-visible, because the difference in pixel intensity across the block edge is zero. However, the artifact is still visible due to a change in a pixel intensity trend. We propose a novel local block visibility metric, which takes into account both factors of block grid visibility. Using the example of Figure 3, blockiness is locally estimated in the following way:

$$Bl = \frac{F_{grid}(V)}{F_{non-grid}(V)} = \left| \frac{(v3-v4) - \dfrac{v2-v3+v4-v5}{2}}{\dfrac{\dfrac{v1-v2+v3-v2}{2} + \dfrac{v4-v5+v6-v5}{2}}{2}} \right|, \qquad (4)$$

(a)



(b)



(c)

Fig 2: Detection of the block-grid position. (a) – an original coded frame, (b) – step-functions of horizontal gradients, (c) – the horizontal accumulator.

$F_{grid}(v)$ corresponds to a pixels gradient across a block edge, compensated by the value of an intensity mismatch at both sides of the block edge.

If the intensity of pixels v2-v5 is changing gradually (e.g. from dark to bright with equal steps), then $F_{grid}(v)=0$, even if |v3-v4|>0. At another hand, if there is a change of an intensity trend, then $F_{grid}(v)>0$, even if |v3-v4|=0.

The behaviour of $F_{non-grid}(v)$ is following:

$F_{non-grid}(v)$ is small if pixels v1,v2,v3 and/or v4, v5, v6 construct one intensity trend (e.g. gradual change of luminance from dark to bright or vice versa). Then the blockiness is more visible and metric Bl has higher value.

$F_{non-grid}(v)$ is large if pixels v1,v2,v3 and/or v4, v5, v6 do not construct one intensity trend (e.g. fluctuations of luminance from dark to bright and vice versa). Then the blockiness is visually masked by fluctuations of intensities within blocks. In this case, the blockiness metric Bl will have smaller values.

In case absolute values of all pixel pair gradients are equal, the state-of-the-art metric would have the same value for both cases mentioned above, but subjectively, the difference is significant. The proposed metric of equation (4) is able to reflect this difference.

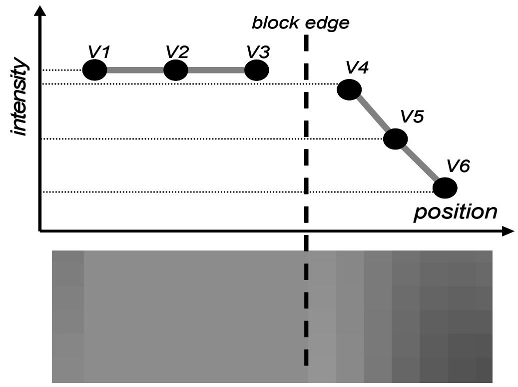


Fig. 3. Example of pixel intensity discontinuity due to the blocking artifact

2.2. COMBINED ARTIFACT REDUCTION AND SHARPNESS ENHANCEMENT

In our algorithm, the artifact reduction process consists of two parts: the pre-filtering and the entropy based filtering, where the second part includes also sharpness enhancement.

During the pre-filtering stage, adaptive blurring is applied to the detected block grid. The goal of the pre-filtering step is to attenuate steep pixel gradients at block grid edges, while preserving sharp object edges and texture in the neighborhood. During the pre-filtering, three-tap filters are applied to pairs of pixels on each side of block edges horizontally and vertically with filter coefficients that are controlled by a local blockiness metric. One of two filtering modes is chosen depending on the value of local block visibility.

The **first filtering mode** is used in the presence of highly visible luminance blockiness, characterized by a large value of Bl. The filtering is applied if

$$Thr1 < Bl < Tedge$$

The filtering in the first mode is the strongest:

$$\begin{aligned} V3' &= (V2 + V3 + V4)/3, \\ V4' &= (V3 + V4 + V5)/3, \\ V2' &= (2*V2 + V3')/3, \\ V5' &= (2*V5 + V4')/3, \end{aligned} \quad (5)$$

where V2'- V5' are output luminance pixel values. We only update the pixels when the Bl is smaller than *Tedge*, because when the Bl is large it might be a real edge.

The **second mode** is applied when the visibility of a luminance discontinuity is lower than in the first mode:

$$Thr2 < Bl <= Thr1$$

The local blockiness has a lower value either due to smaller pixel discontinuity over the block edge $F_{grid}(v)$, or due to large pixel activity within blocks $F_{non-grid}(v)$. Both cases require less strong filtering than in the first mode. The second mode filter is defined as:

$$\begin{aligned} V3' &= (V2 + 2*V3 + V4)/4, \\ V4' &= (V3 + 2*V4 + V5)/4 \end{aligned} \quad (6)$$

The thresholds Thr1, Thr2, Tedge, can be varied depending on resolution, bit-rate and sharpness of the decoded image.

Pre-filtering removes the steepest block-edges, but lower-frequency components of blockiness as well as mosquito noise and ringing are still present in the decoded video. In order to remove those artefacts, 2D bilateral low-pass filtering is applied to decoded pictures, which is integrated with sharpness enhancement to reconstruct steepness of object edges without peaking a residual noise.

The kernel size and the parameters of 2D bilateral filter are controlled using estimates of entropy and dynamic range (DR) within an analyzed window. Local entropy reflects the randomness of pixel values and describes the information content of a local region, while DR shows the maximum difference in pixel values. A low level of the entropy indicates that the region has low information content, and thus stronger filtering could be applied. The local entropy H is defined as:

$$E = -\sum_{i=0}^{255} P(v)\log_2 P(v), \quad (7)$$

where *v* – is the pixel intensity, *P(v)*- is the probability density function of the pixel intensity distribution inside the region. The aperture of the filter or its radius *R* is determined depending on the entropy value:

$$R = 1 + 4 * \frac{E_{max} - E}{E_{max} - E_{min}}, \quad (8)$$

where E_{max} and E_{min} are the pre-defined maximum and minimum values of entropy, which depend on Bl. The kernel size of the filter is small when the entropy is high and the kernel size is large, when the entropy is low.

High values of DR reflect the presence of a high contrast object edge in the image window, which means that coefficients of the bilateral filter should be adjusted in order to remove ringing and enhance sharpness of this object edge.

The values of all pixels from the neighborhood, defined by radius R, are compared to the value of the current pixels. Only the pixels whose values are within a certain range T compared to a current pixel value, take part in the low-pass filtering. If an absolute deviation of the rest of the pixels from the centre pixel is much large than the pre-defined range T, then these pixels are included in the high-frequency part of the combined filter. The range T is dependent on the value of DR. The maximum aperture of the high-frequency part of 2D bilateral filtering is limited to 5x5 window.

The low-frequency filtering coefficients C1 for a current pixel and C2 for the neighboring pixels are:

$$C_1 = \frac{1}{n} + \left(1 - \frac{1}{n}\right) * \frac{E - E_{\min}}{E_{\max} - E_{\min}} \quad (9)$$

$$C_2 = \frac{1}{n} - \frac{1}{n} * C_1 \quad (10)$$

Coefficients for the high-frequency part of 2D filtering are defined in the following way:

$$C_h = \frac{1}{nh} * \frac{E}{2 * E_{\max}} \quad (11)$$

where nh – is the number of pixels participating in the high-frequency part of the combined 2D filtering.

Thus, the low-frequency part of the combined filtering:

$$\text{LFP} = C_1 * V_c + \sum C2 * V_i \quad (12)$$

where V_i - pixels, chosen for low-pass filtering. The final result of the combined 2D filter is obtained in the following way:

$$V_c' = (C_1 V_c + \sum C_2 V_i\ ) * (1 + \frac{E}{E_{\max}}) - \sum V_h C_h \quad (13)$$

3. EVALUATION

The efficiency of the proposed artifact reduction method was evaluated subjectively, using PSNR and BIM [2] metrics, on test sequences with SD and SIF resolutions compressed by an MPEG-2 coder at different bit-rates. Methods [4,5,6] were chosen for benchmarking. During objective evaluations, the sharpening factor was set to a minimum value in order to provide fair comparison of PSNR. Table 1 shows values obtained as an average of BIM values in horizontal and vertical directions. Table 2 contains PSNR values averaged for entire sequences. The proposed algorithm outperforms the benchmarked methods in terms of blockiness reduction and provides similar or higher PSNR values. Figure 4 displays a part of the zoomed frame of the "foreman" sequence with SIF resolution compressed at 500 kbit/s, prior (Fig. 4a) and after (Fig. 4b) post-processing with the proposed algorithm. The processed picture contains much less coding artifacts and has sharper edges.

4. CONCLUSIONS

In this paper, a new flexible algorithm for enhancement of compressed video signals is presented. The algorithm is able to detect and adaptively remove coding artifacts, while preserving and even sharpening image details. No bit-stream coding parameters are required to control the algorithm Artifact reduction and de-blurring is implemented using one integrated filtering process, which guarantees that the artifacts reduced during de-blocking will not be enhanced by sharpness enhancement, as may occur in cascaded solutions.

(a) (b)

Fig.4. Compressed frame prior (a) and after (b) processing

TABLE 1. BIM OF PROCESSED TEST SEQUENCES

bit-rate Mbit/s	decoded	method [4]	method [5]	method [6]	proposed method
"Stefan" SIF					
0,25	2,57	1,93	1,94	1,81	1,47
0,50	2,26	1,69	2,22	1,59	1,38
1,00	1,64	1,38	1,62	1,34	1,11
"Vanessa" SD					
2,00	5,14	2,24	3,98	2,15	2,00
3,00	2,96	1,60	2,78	1,57	1,50
4,00	2,31	1,38	2,20	1,32	1,26

TABLE 2. PSNR OF PROCESSED TEST SEQUENCES

bit-rate Mbit/s	method [4]	method [5]	method [6]	proposed method
"Stefan" SIF				
0,10	22,88	22,75	22,89	22,91
0,25	23,08	22,98	23,10	23,23
0,50	23,60	23,54	23,61	23,55
1,00	26,46	26,49	26,51	26,48
"Vanessa" SD				
2,00	29,62	29,21	29,64	28,96
3,00	31,40	31,00	31,42	31,47
4,00	32,78	32,45	32,87	33,06
5,00	33,93	33,67	33,98	34,35

REFERENCES

[1] O.A. Ojo, and T.G. Kwaaitaal-Spassova, "An algorithm for integrated noise reduction and sharpness enhancement", *IEEE Trans. On Consumer Electronics,* Vol. 46, Issue 3, Aug.2000, pp. 474-480.

[2] H.R. Wu and M. Yuen, "A Generalized Block-Edge Impairment Metric for Video Coding", *IEEE Signal Processing Letters*, Vol.4, Issue 11, Nov. 1997, pp.317-320.

[3] R. Muijs, and I. Kirenko, "A non-reference blocking artifact measure for adaptive video processing ", *Proc. of the 13th European Signal Processing Conference (EUSIPCO)*, Turkey, 2005.

[4] I. Kirenko, "Reduction of coding artifacts using chrominance and luminance spatial analysis", *Proc. of International Conference on Consumer Electronics (ICCE)*, USA, January 2006.

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

[6] I. Kirenko, and L.Shao, "Adaptive repair of compressed video signals using local objective metrics of blocking artifacts", submitted to *International Conference on Image Processing (ICIP)*, USA, 2007.