Quality adaptive trained filters for compression artifacts removal

QUALITY ADAPTIVE TRAINED FILTERS FOR COMPRESSION ARTIFACTS REMOVAL

Ling Shao [1], *Jingnan Wang* [2], *Ihor Kirenko* [1], *Gerard de Haan* [1 2]

[1] Philips Research Europe, High Tech Campus 36, Eindhoven 5656 AE, The Netherlands
[2] Eindhoven University of Technology, Den Dolech 2, Eindhoven 5600 MB, The Netherlands

ABSTRACT

A compression artifacts removal algorithm that is adaptive to the artifact visibility level of the input video signal is proposed. The artifact visibility is determined per frame by the ratio of the accumulated gradient on the block edges to that of the remaining area. The filtering of each video frame is optimized using a least mean square mechanism which trains on pairs of target images and decompressed images of similar quality as the input frame. Experimental results show that the proposed approach outperforms several recent methods in coding artifact reduction.

Index Terms— Quality metric, adaptive filters, compression artifacts removal, least mean square optimization

1. INTRODUCTION

In the last decade, digital has replaced analogue transmission for television, digital storage on DVDs replaced analogue storage on VCRs, while streaming over the Internet introduced video on the PC. Digitalization, without further measures, greatly expands the required storage capacity and transmission bandwidths. All this has become possible, therefore, only because of the progress made on video compression, or digital coding. Coding standards have evolved from JPEG to MPEG-2/4 to the latest H.264/AVC. These block transform based codecs divide the image or video frame into non-overlapping blocks (usually with the size of 8 x 8 pixels), and apply discrete cosine transform (DCT) on them. The DCT coefficients of neighboring blocks are thus quantized independently. At high or medium compression rates, the coarse quantization will result in various noticeable coding artifacts, such as blocking, ringing and mosquito artifacts. Among the coding artifacts, blockiness which appears as discontinuities along block boundaries is the most annoying. Therefore, an in-loop deblocking filter is specified for H.264/AVC to reduce blocking artifacts using coding parameters inside the encoder. However, JPEG compressed images and MPEG-2/4 compressed videos will remain ubiquitous, which makes post-processing aimed at the elimination of coding artifacts still a critical and indispensable solution.

Most coding artifact reduction techniques based on post-processing, e.g. [1, 2], are designed according to heuristic tuning and testing, which takes a lot of time and is not always effective. Recently, classification-based trained filters (TF) have been proposed for optimally removing digital coding artifacts [3, 4]. The local image characteristics can be classified using local structure or activity information. For each class, the optimized filter coefficients are obtained by an off-line training process, which trains on the combination of targeted images and the degraded versions thereof that act as source. The methods introduced in [3, 4] produce promising results, when the quality of the test sequence is similar to that of the source sequences used during training. It is because a fixed level of compression is adopted for degrading the target images. We propose to train the algorithm on a range of compression levels and to select the most suitable set of filter coefficients for the test sequence. To do that, a quality (or blockiness) metric is required to indicate the quality level of the test sequence.

We introduce the quality adaptive trained filters in Section 2. The quality/blockiness metric, which represents the artifact visibility of an image, is presented in Section 3. The algorithm is evaluated objectively in comparison to other artifact reduction approaches in Section 4. In Section 5, we conclude our paper.

2. QUALITY ADAPTIVE TRAINED FILTERS

In this section, trained filters using a quality metric will be described. The algorithm is composed of two parts: the off-line training process and the run-time filtering process. Fig. 1 shows the proposed training process. Uncompressed images are used as target images. The target images are degraded with different degradation levels. For coding artifacts reduction, the degradation is compression. We refer the compressed images as degraded images. For each degradation level, each pixel in the degraded images is then classified on that pixel's neighborhood using a classification method. The classification method we use here is Adaptive Dynamic Range Coding (*ADRC*), which represents the structure information of a local region, coupled with a

1-4244-1484-9/08/$25.00 ©2008 IEEE

complexity measure. The *ADRC* code of each pixel x_i in an observation aperture is defined as: $ADRC(x_i) = 0,\ if\ V(x_i) \le V_{av};\ 1,\ otherwise,$ where $V(x_i)$ is the value of pixel x_i, and V_{av} is the average of all the pixel values in the aperture. The *ADRC* code of an image kernel is the concatenation of the *ADRC* codes of all the pixels in that kernel. The complexity measure we adopt is standard deviation, which performs the best for coding artifact reduction in the performance evaluation experiments in [5]. All the pixels and their neighborhoods belonging to a specific class and their corresponding pixels in the target images are accumulated, and the optimal coefficients per class are obtained by making the Mean Square Error (MSE) minimized statistically.

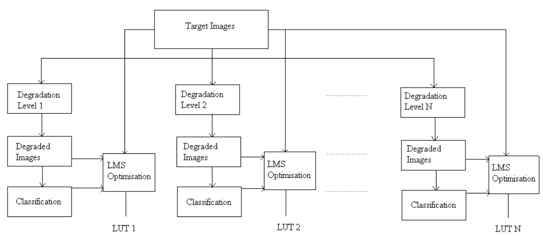

Fig. 1: The training process of the proposed method.

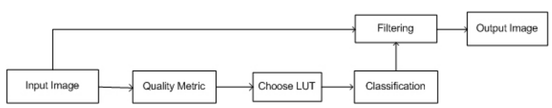

Fig. 2: The filtering process of the proposed method.

In the following, we describe the Least Mean Square (LMS) optimization for a certain degradation level. Let $F_{D,c}$, $F_{R,c}$ be the apertures of the degraded images and the target images for a particular class *c*, respectively. Then the filtered pixel $F_{F,c}$ can be obtained by the desired optimal coefficients as follows:

$$F_{F,c} = \sum_{i=1}^{n} w_c(i) F_{D,c}(i,j) \tag{2}$$

where $w_c(i), i \in [1...n]$ are the desired coefficients, and *n* is the number of pixels in the aperture.

The summed square error between the filtered pixels and the target pixels is:

$$\begin{aligned} e^2 &= \sum_{j=1}^{N_c} (F_{R,c} - F_{F,c})^2 \\ &= \sum_{j=1}^{N_c} [F_{R,c}(j) - \sum_{i=1}^{n} w_c(i) F_{D,c}(i,j)]^2 \end{aligned} \tag{3}$$

where N_c represents the number of pixels belonging to class *c*. To minimize e^2, the first derivative of e^2 to $w_c(k), k \in [1...n]$ should be equal to zero.

$$\frac{\partial e^2}{\partial w_c(k)} = \sum_{j=1}^{N_c} 2F_{D,c}(k,j)[F_{R,c}(j) - \sum_{i=1}^{n} w_c(i) F_{D,c}(i,j)] = 0 \tag{4}$$

By solving the above equation using Gaussian elimination, we will get the optimal coefficients as follows:

$$W = X^{-1}Y \tag{5}$$

where

$$W = \left[w_c(1), w_c(2), \cdots, w_c(n)\right]^T$$

$$X = \begin{bmatrix} \sum_{j=1}^{N_c} F_{D,c}(1,j)F_{D,c}(1,j) & \cdots & \sum_{j=1}^{N_c} F_{D,c}(1,j)F_{D,c}(n,j) \\ \sum_{j=1}^{N_c} F_{D,c}(2,j)F_{D,c}(1,j) & \cdots & \sum_{j=1}^{N_c} F_{D,c}(2,j)F_{D,c}(n,j) \\ \cdots & \cdots & \cdots \\ \sum_{j=1}^{N_c} F_{D,c}(n,j)F_{D,c}(1,j) & \cdots & \sum_{j=1}^{N_c} F_{D,c}(n,j)F_{D,c}(n,j) \end{bmatrix}$$

$$Y = \left[\sum_{j=1}^{N_c} F_{D,c}(1,j)F_{R,c}(j), \sum_{j=1}^{N_c} F_{D,c}(2,j)F_{R,c}(j), \cdots, \sum_{j=1}^{N_c} F_{D,c}(n,j)F_{R,c}(j)\right]^T$$

The LMS optimized coefficients for each class are then stored in a look-up table (LUT) for future use. For different degradation levels, several LUTs are obtained after training. Fig. 2 shows the filtering process of the algorithm. The quality of each input image is first evaluated using a quality metric. Then, the most suitable LUT is selected for that image according to its quality score. The optimized coefficients are retrieved from that LUT to filter the image based on pixel classification.

3. ARTIFACT VISIBILITY METRIC

For an input video frame, a quality metric is required to indicate which LUT is the most appropriate. For compression artifacts removal, an artifact strength measure can be used as the quality metric. Since blocking artifacts are the most noticeable, a blockiness metric, introduced in [2], is adopted to measure the artifact level of a video frame.

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. 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, a simple and efficient algorithm for detection of the grid position and estimation of a block edge visibility is applied.

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:

$$D_{H,norm}(i,j) = \frac{\left|Y_{i+1,j} - Y_{i,j}\right|}{\frac{1}{2N}\sum_{n=-N\ldots N, n\neq 0}\left|Y_{i+n+1,j} - Y_{i+n,j}\right|} \quad (6)$$

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:

$$S_H(i) = \sum_{j=1}^{nl} D_{H,norm}(i,j) \quad (7)$$

The presence of blocking artifacts will result in pronounced maxima in S_H. The above procedure is illustrated for the image *branch* displayed in Fig. 3(a). 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 Fig. 3(b). 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:

$$BS = \frac{\overline{S}_H(block)}{\overline{S}_H(non-block)} \quad (8)$$

where $\overline{S}_H(block)$ and $\overline{S}_H(non-block)$ 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 [6], that consists of 169 JPEG encoded images and associated mean quality scores (MQS). Fig. 3(c) 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 for an encoded frame.

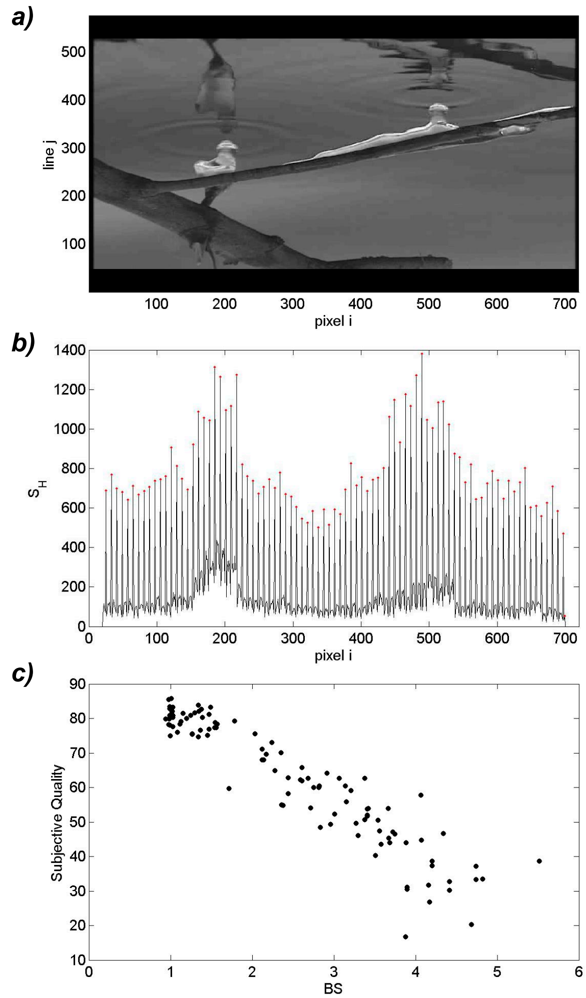

Fig. 3: Detection of the position and visibility of the coding grid. Shown are (a) the image *branch* encoded at a bit-rate of 2Mb/s, (b) the horizontal accumulator S_H computed using equations (6) and (7) and (c) the the objective blockiness metric BS vs. MQS for JPEG encoded images from the LIVE database.

4. EXPERIMENTS AND EVALUATION

We carry out experiments using the proposed quality adaptive trained filters on MPEG coding artifact reduction in this section. MPEG-2 (http://www.chiariglione.org/mpeg/standards/mpeg-2/mpeg-2.htm) is used for compression, since sufficient coding artifacts appear when using this classical compression standard, which poses a challenge for artifacts reduction algorithms. A set of sequences with varied content, as depicted in Fig. 4, is employed for experiments. The sequences are compressed with different compression rates in the range of 1-4 Mbits/s to test the scalability of the algorithm on various qualities of the test sequences.

To evaluate the performance the proposed algorithm, Mean Square Error (MSE) between the original uncompressed sequence and the outcome sequence after applying the proposed artifact reduction on the decompressed sequence is calculated. The MSE scores of different sequences with different compression rates are shown in Table 1. For benchmarking, we compare the MSE results with two state-of-the-art methods, proposed in [1, 7]. These two methods both rely on block grid locations and apply different filtering according to the complexity of the region. To show the superiority of quality metric controlled trained filters, the results of trained filters that optimize the coefficients using a mixture of compression rates during training are also shown. For the mixed trained filters, the filter coefficients selected for a particular image kernel are only dependent on pixel classification, i.e. no quality metric is used.

Table 1: MSE scores of different methods.

Sequence/Bit-rate (bits/s)	Proposed	Mixed TF	Ref [1]	Ref [7]
Bicycle/1M	102.726	109.812	135.750	121.850
Bicycle/2M	71.416	74.116	95.884	88.879
Bicycle/3M	50.343	51.964	70.143	68.191
Bicycle/4M	38.466	39.400	55.352	56.063
Girlsea/1M	79.115	81.674	86.634	82.621
Girlsea/2M	41.141	41.813	44.898	41.369
Girlsea/3M	21.824	22.365	24.194	22.583
Girlsea/4M	14.221	14.554	16.128	15.643
Soccer/1M	38.862	39.359	41.137	40.243
Soccer/2M	18.986	19.130	20.178	20.442
Soccer/3M	15.277	15.510	16.084	16.688
Soccer/4M	13.105	13.634	13.892	14.709

The comparison in Table 1 shows that the proposed quality adaptive trained filters outperform the other methods significantly for all the sequences with different qualities. For subjective evaluation, Fig. 5 shows the result of the proposed algorithm for the Bicycle sequence. It is easy to see that the proposed algorithm can suppress coding artifacts and preserve object details simultaneously.

5. CONCLUSION

A quality adaptive algorithm based on trained filters is proposed for coding artifacts reduction. During training, multiple LUTs are obtained for different degradation levels. The quality of an input video frame is designated by a blockiness visibility metric. A most suitable LUT for that quality level is then chosen for filtering the input frame.

The proposed approach is not dependent on block grid positions and removes various coding artifacts simultaneously. The training process is computationally intensive, because it has to train on a large variety of video data. Fortunately, it only needs to be done off-line and once. During the filtering process, optimized filter coefficients are retrieved from a LUT according to the quality level and pixel classification. The LUTs can be implemented on external Read Only Memory (ROM).

Fig. 4: Snapshots of the test sequences: Bicycle, Girlsea and Soccer

Fig. 5: A decoded image before and after applying the proposed algorithm.

REFERENCES

[1] L. Shao and I. Kirenko, “Coding artifact reduction based on local entropy analysis”, IEEE Trans. Consumer Electronics, Vol. 53(2), pp. 691-696, May 2007.

[2] I. Kirenko, R. Muijs and L. Shao, “Coding artifact reduction using non-reference block grid visibility measure”, Proc. IEEE International Conference on Multimedia and Expo, Toronto, Canada, July 2006.

[3] L. Shao, “Unified compression artifacts removal based on adaptive learning on activity measure”, Digital Signal Processing, doi:10.1016/j.dsp.2006.10.007 (In Press).

[4] M. Zhao, R. E. J. Kneepkens, P. M. Hofman and G. de Haan, “Content adaptive image de-blocking”, Proc. IEEE International Symposium on Consumer Electronics, Sep. 2004.

[5] L. Shao, H. Zhao and G. de Haan, “An overview and performance evaluation of classification-based least mean square trained filters” In preparation to submit to IEEE Trans. Image Processing.

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

[7] S. D. Kim, J. Yi, H. M. Kim and J. B. Ra, “A deblocking filter with two separate modes in block-based video coding”, IEEE Trans. Circuits and Systems for Video Technology, Vol. 9(1), pp. 156-160, Feb. 1999.