

ADAPTIVE DE-BLOCKING DE-RINGING POST FILTER

Andreas Rossholm

Department of Signal Processing
School of Engineering
Blekinge Institute of Technology
Ronneby, Sweden

Kenneth Andersson

Multimedia Technologies
Ericsson Research
Ericsson AB
Kista, Sweden

ABSTRACT

In this paper an adaptive filter for reducing blocking and ringing artifacts is presented. The solution is designed with consideration of Mobile Equipment with limited computational power and memory. Also, the solution is computationally scalable if there is limited CPU resources in different user cases.

1. INTRODUCTION

In the Mobile Equipment (ME) today the use of video becomes more and more common. To make it possible to view a video clip or streaming video, or to make a video telephony call, it is important to compress the data as much as possible. Most codecs, video encoder and decoder, used today are designed as a block-based motion-compensated hybrid transform coder, as MPEG-4, H.263, where the transformation is done by a Discrete Cosine Transforms (DCT) on blocks of 8x8 pixels. The DCT coefficients are quantized with a quantization parameter (QP).

Two of the main artifacts from the quantization of the DCT are blocking and ringing. Blocking artifacts are also due to motion compensation. The blocking artifact is seen as an unnatural discontinuity between pixel values of neighboring blocks. The ringing artifact is seen as high frequency irregularities around the image edges. In brief; the blocking artifacts are generated due to the blocks being processed independently, and the ringing artifacts due to the coarse quantization of the high frequency components [1].

To reduce blocking artifacts, two-dimensional (2D) low-pass filtering of pixels on block boundaries of the decoded image(s) was suggested in [2]. The 2D space-invariant static filtering described in that paper reduces blocking artifacts but can also introduce blurring artifacts when true edges in the image are low-pass filtered.

To avoid blurring of true edges in the image and also to

be computationally efficient, the amount of low-pass filtering may be controlled by table-lookup as described in [3]. Large differences between initial pixel values and filtered pixel values are seen as natural image structure, and thus filtering is weak so that the image is not blurred. Small pixel differences are seen as coding artifacts, and thus stronger filtering is allowed to remove the artifacts. Based on data from other equipment, the amount of filtering can be controlled by using additional filter tables. The algorithm modifies the output of a low-pass-filtered signal with the output of a table-lookup using the difference between a delayed input signal and the filtered signal as an index into the table, and different degrees of filtering are achieved only by providing additional tables.

A combined de-blocking and de-ringing filter was proposed in [4]. The proposed filter used filter strengths on block boundaries that were different from filter strengths inside blocks, allowing for stronger filtering at block boundaries than inside blocks. This was achieved by using a metric that used different constants when computing the output values of block boundary pixels versus the output values of pixels inside the block boundary. The metric also included the QP value.

These and most other current algorithms handle de-blocking and de-ringing artifacts sequentially. Such double filtering can have a negative impact on computational complexity and memory consumption, which are parameters of particular importance in many devices, such as mobile communication devices.

Moreover, removal of blocking and ringing artifacts can add visually annoying blurring artifacts as described above. It is thus important to be careful with strong image features that likely are natural image features and not coding artifacts.

2. THE ADAPTIVE FILTER

The proposed filter is developed with two main considerations; limiting the computational complexity, and limiting the amount of working memory. The idea is to filter rows

Special thanks to Andreas Rossholms supervisor Benny Lövsström for editing and reviewing the manuscript.

of pixels of an image in a vertical direction, store the results in row vectors, and then filters the row vectors in the horizontal direction, and display the results. In the following part the adaptive filter is described in one of the above directions. Coefficients of a reference filter are modified based on the output from the reference filter passed through a table-lookup process that accesses a table of modifying weight coefficients. The output of the modified filter is added to a delayed version of the input to provide the adaptive filter output. A block diagram of the adaptive filter is shown in Fig. 1.

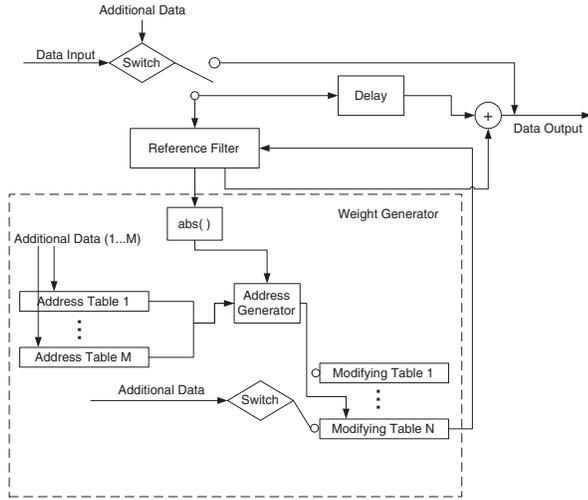


Fig. 1. A block diagram of the adaptive filter.

2.1. Overview of the filter

In Fig. 1 an input data stream of pixel data is provided to a switch that directs the input pixels to either the output of the filter or to a delay element and a reference filter. The operation of the switch is responsive to additional data, in particular, whether the input pixels belong to an error-concealed block or if the amount of filtering is limited based on location in the frame, as described in more detail below. The reference filter has coefficients that determine the filtering function, and these coefficients are selectively modified. The output of the reference filter is provided to an adder that combines the output with the delayed input produced by the delay element, thereby generating the output of the adaptive filter.

The modification of the output of the reference filter is performed by a weight generator that produces weights that selectively modify the coefficients of the filter based on the filter output to the weight generator. A signal corresponding to the absolute value of the reference filter output is produced, and this signal is provided to an address generator. The absolute value together with additional data provided by M

suitable address tables, generates addresses into N tables of modifying weight coefficients, as described in more detail below. As a set of modifying weight coefficients is retrieved from the selected table, it is provided by the weight generator to the filter, and the transfer function of the reference filter is modified accordingly. Through this modification, the filter adapts to the input stream of pixels.

2.2. Reference Filter

In the adaptive filter a 5-tap reference filter is used, $[1 \ 1 \ -4 \ 1 \ 1]$. The number of filter taps chosen is the result of a trade-off between the amount of low-pass filtering that can be performed, locality in filtering, and computational complexity. The filter coefficients are chosen to detect variations in pixel value in the filter neighbourhood with as low complexity as possible. The same filter is used for filtering luminance, denoted Y , and chrominance blocks, denoted U and V , although luminance blocks are more important to filter than chrominance blocks. The modification of the reference filter is performed with a set of modifying weights and the resulting adaptive filter response is illustrated in Fig. 2. It

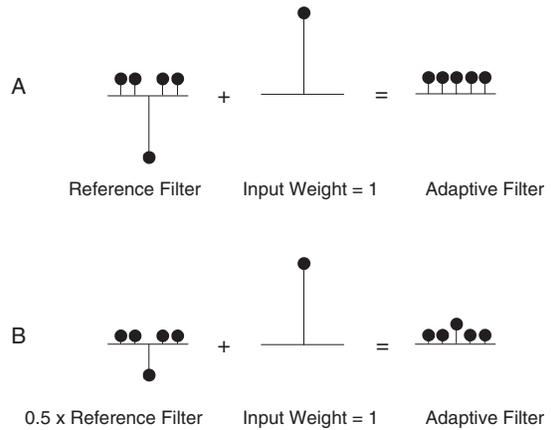


Fig. 2. Depiction of reference filter modification and adaptive filter response.

can be seen that the same modification is made to each coefficient. In the figure, the sign and magnitude of a filter coefficient or a weight are indicated by the length of the respective vertical line segment and its position above or below the horizontal reference line. The "+" sign indicates the operation of the adder. In Fig. 2B the modifying weight is shown as 0.5 and the other coefficients are fixed. Comparing Fig. 2A and Fig. 2B, it will be seen that a "weaker" adaptive filter is achieved when the reference filter coefficients are scaled by a factor of 0.5, i.e., neighboring pixels have less influence on the modified-filter output for a pixel. If the modifying weights are such that all filter coefficients

are modified in the same way (see, e.g., FIG. 2.B), also used in this implementation, the output of the modified reference filter is simply a scaling of the output of the unmodified reference filter. Otherwise, the output of the modified reference filter is calculated using the input pixels and the modified reference filter transfer function.

2.3. Weight Generator

The weight generator handles the adaptive part of the filter. It is divided into three main parts:

- 1 The address tables with additional data.
- 2 The address generator.
- 3 Modifying tables with switch and additional data.

The first part, address table, uses QP for the block as additional data and the address table length correspond to the range of the QP data. The output from the address table are positive for low QP values and negative for high QP values, resulting in potentially weaker and stronger filtering, respectively, depending on the magnitude of the reference filter output. Several address tables can be used if different sessions needs different strength of filtering.

The second part, address generator, produces a signal corresponding to the absolute value of the filter output, together with the output from the address tables, to generate addresses into one of the modifying tables with weight coefficients.

The third part, modifying tables, provides sets of weight coefficients to modify the transfer function of the reference filter, resulting in a modified, or adapted, transfer function for the adaptive filter as described in subsection 2.2. The length (i.e., the address range) of a modifying tables corresponds to the range of the reference filter output. In this implementation small address values give weights close to 1/5 and large address values give weights close to 1/260. The result is thus variation from flat low-pass filtering to very small low-pass filtering over the filter output range.

The additional data that is input to the switch, selecting modification table, is based on the position of a pixel in its block. As indicated by FIG. 3, which depicts a block of pixels, outer boundary pixels (indicated by + in the figure) select a table that corresponds to stronger filtering than the table selected for inner block pixels. Furthermore, the weights of the selected table for inner pixels (indicated by # in FIG. 3) decreases more quickly with increasing index than the weights in the boundary pixels table. This results in reduction of blocking and ringing artifacts without blurring the image too much. x/y in FIG. 3 describes filtering in horizontal/vertical direction.

Fig. 3. Depiction of a block of pixels.

2.4. Further considerations

The first switch in Fig. 1 makes it possible to limit the amount of filtering for different combinations of applications for a given device. The priority of filtering is given from low to high priority as, all luminance and chrominance blocks may be filtered, only luminance blocks may be filtered, outer boundary pixels may be filtered, and only block border pixels may be filtered.

3. RESULTS

The performance of the adaptive filter is evaluated against using no post filtering and filtering as recommended in H.263 App. III [4]. The algorithms are processed on decoded H.263 profile 0 bit streams for two different sequences each presented at four different bit rates at 15 frames per second (fps) and of size 176×144 (QCIF). The size, bitrates and framerate are chosen to correspond with the use in today's 2G and 3G networks. The peak signal-to-noise ratio (PSNR) is calculated for the post processed images and an average for the complete sequence. The PSNR of an $M \times N$ image is given by

$$PSNR = 10 \log \frac{MN \times 255^2}{\sum_{m,n} \|f(m,n) - f_{org}(m,n)\|^2}$$

The sequences used are "Foreman" and "Mother and Daughter" presented in Table 1 and Table 2. In the tables it is shown that the adaptive filter always keeps or increases the PSNR compared to the original decoded sequences. The adaptive filter gives significantly better visual quality as can be seen in Fig. 4 and Fig. 5. As shown in the tables H.263 App. III gives slightly higher PSNR than the adaptive filter but also gives somewhat blurred results compared to the adaptive filter, see Fig. 4 and Fig. 5. It shall also be noted that the complexity of the adaptive filter is about 18 cycles per filtered pixel including 2 multiplications, 10 additions, 4 shifts and 2 abs, which is significantly lower than for the H.263 App. III filter. H263 App. III will require at least 34

Foreman			
Bitrate [kbit/s]	Filter	Average PSNR [dB] for YUV	Average PSNR [dB] for Y
32	No Post Filter	30.0038	28.6846
32	H.263 App. III	30.0960	28.7631
32	Adaptive Filter	30.0482	28.7219
48	No Post Filter	31.0114	29.7355
48	H.263 App. III	31.1552	29.8716
48	Adaptive Filter	31.0611	29.7799
64	No Post Filter	31.8652	30.6329
64	H.263 App. III	32.0412	30.8038
64	Adaptive Filter	31.9159	30.6797
128	No Post Filter	34.4384	33.3140
128	H.263 App. III	34.6722	33.5539
128	Adaptive Filter	34.4716	33.3475

Table 1. Results from de-blocking and de-ringing on Foreman. All sequences have a QCIF resolution and 15 fps.

Mother and Daughter			
Bitrate [kbit/s]	Filter	Average PSNR [dB] for YUV	Average PSNR [dB] for Y
32	No Post Filter	34.3828	33.2452
32	H.263 App. III	34.5634	33.4313
32	Adaptive Filter	34.4442	33.2975
48	No Post Filter	35.6795	34.5960
48	H.263 App. III	35.8562	34.7795
48	Adaptive Filter	35.7138	34.6301
64	No Post Filter	36.6913	35.6550
64	H.263 App. III	36.8635	35.8284
64	Adaptive Filter	36.6943	35.6641
128	No Post Filter	39.4050	38.5265
128	H.263 App. III	39.5368	38.6457
128	Adaptive Filter	39.4314	38.5492

Table 2. Results from de-blocking and de-ringing on Mother and Daughter. All sequences have a QCIF resolution and 15 fps.

cycles per filtered pixel including 4 divisions, 14 multiplications and 16 additions.

4. CONCLUSION

This paper has described an adaptive filter that can improve visual quality by combat both de-blocking and de-ringing artifacts as generated by standard block based coders. The filter has further low complexity and can be used in MEs with limited computational power and memory.



Fig. 4. Luminance output from Foreman in QCIF format, coded at 32 kbps and 15 fps. From left, No Post Filter PSNR 28.30 dB, H.263 App. III PSNR 28.51 dB, Adaptive Filter PSNR 28.40 dB.



Fig. 5. Luminance output from Mother and Daughter in QCIF format, coded at 32 kbps and 15 fps. From left, No Post Filter PSNR 34.20 dB, H.263 App. III PSNR 34.35 dB, Adaptive Filter PSNR 34.22 dB.

5. REFERENCES

- [1] Michael Yuen, H.R. Wu, "A survey of hybrid MC/DPCM/DCT video coding distortions," *Signal Processing*, vol. 70, pp. 247-278, July 1998.
- [2] H. C. Reeve III, Jae S. Lim, "Reduction of Blocking Effect in Image Coding," *Proc. ICASSP*, pp. 1212- 1215, Boston, Mass. 1983.
- [3] US patent No. 5,488,420 to G. Bjontegaard for "Cosmetic filter for smoothing regenerated pictures, , e.g. after Signal Compression for Transmission in a Narrow-band Network".
- [4] ITU-T Recommendation H.263 Appendix III: "Examples for H.263 Encoder/Decoder Implementations," June 2000.