Optimal Post-Process/In-Loop Filtering for Improved Video Compression Performance

Dong-Hwan Kim, Hwa-Yong Oh, Oğuzhan Urhan, *Member*, IEEE, Sarp Ertürk, *Member*, IEEE,
Tae-Gyu Chang, *Member*, IEEE

Abstract — *This paper presents a novel optimal filtering approach that can be integrated into video compression schemes to improve video compression performance. In the proposed approach the encoder computes the coefficients of a linear filter in an optimal way, so as to minimize the squared error between the original frame and the filtered reconstructed frame. The encoder then multiplexes the filter coefficients into the bit-stream for decoder access. Reconstructed image frames are filtered at the decoder using optimal filter coefficients to obtain improved image frames. The optimal filtering approach is evaluated in the form of a post-process filter as well as in the form of an in-loop filter. It is shown using H.263+ and H.264/AVC that the proposed approach improves video compression performance of standard compression schemes*[1].

Index Terms — Video compression, optimal filtering, in-loop filtering, post filtering.

I. INTRODUCTION

Emerging consumer electronics devices such as digital television, mobile video terminals, video telephony, digital video camcorders, and multimedia services for networking applications, have increased demand for efficient video communications [1]. Almost all state-of-the-art video consumer electronics use video compression schemes to reduce redundancies in video sequences, for storage or transmission.

Many video compression standards, including the ISO/IEC MPEG-1, MPEG-2, and MPEG-4 standards as well as the ITU-T H.261, H.263, H.264 standards have been developed for video compression. For example, MPEG-2 has been adopted for digital video broadcast and distribution of video on VCD or DVD, while H.263 has found applications in video conferencing systems as well as video streaming in broadband and wireless networks. The latest H.264 standard, also called MPEG-4 Version 10, or Advanced Video Coding (AVC), is currently the state-of-the-art video compression scheme, jointly developed by MPEG and the ITU-T in the framework of the Joint Video Team (JVT). With various new tools to enhance the ability to predict the picture content, H.264/AVC outperforms all of the previous standards, in terms of coding efficiency. H.264/AVC is suitable for video applications in all areas and intended to replace the existing standards [2]. Several applications, such as high-definition DVD and digital video broadcasting for handheld devices and high-definition television systems, have already adopted H.264 or its modified versions [3].

Research in the area of video compression is mostly concentrated on reducing the computational complexity of the system, possibly at the cost of compression efficiency [4-6], or improving the compression efficiency of the system, possibly at the cost of computational complexity [7,8].

Significant research is carried out in the area of video compression to improve coding efficiency as well as to reduce compression artifacts particularly encountered at low bit-rates. Pre- and post-processing operations as well as in-loop deblock filtering has gained attention due to their potential of improving the coding performance and visual appearance. Most of these techniques are aimed at reducing blocking artifacts that may cause major visual degradation, affecting the performance of block-based video compression schemes.

The utilization of an in-loop deblocking filter has been accepted in H.263+ (Annex J) [9] and the H.264/MPEG-4 AVC [10] video coding standards. Unlike post-filtering, predicted pictures are computed from filtered versions of the previously reconstructed frame in in-loop filtering. That is, filtered frames are used as reference frames for motion compensation of subsequent coded frames. In this case, all standard conformant decoders have to perform identical filtering in order to stay in synchronization with the encoder. The deblocking filter of H.263+ [9] is applied to a window of four edge pixels in the horizontal direction first, and then similarly in the vertical direction, with the weight of filter coefficients depending on the quantizer step size for a given macroblock, where stronger coefficients are used for a coarser quantizer. The deblocking filter included in H.264/MPEG-4 AVC [10] on the other hand is an adaptive filter that adjusts its strength according to the compression mode of a macroblock (Intra or Inter), the quantization parameter, motion vector, frame or field coding decision and the pixel values. In [11] it has been investigated if in-loop deblock filtering should be removed entirely from the coding loop or not, and it is concluded to use in-loop

---

[1] This work was supported by the Scientific and Technical Research Council of Turkey (TUBİTAK) and Korean Research Foundation (KRF) Cooperation Program project entitled "Optimal and near-optimal postfiltering for improved compressed video quality".

Dong-Hwan Kim, Hwa-Yong Oh, and Tae-Gyu Chang are with the School of Electrical and Electronics Engineering, Chung-Ang University, Seoul, Korea (e-mail: free@dmc.cau.ac.kr, proteus@dmc.cau.ac.kr, tgchang@cau.ac.kr).

Oğuzhan Urhan and Sarp Ertürk are with Kocaeli University Laboratory of Image and Signal processing (KULIS), Department of Electronics and Telecommunication Engineering, University of Kocaeli, 41040 Kocaeli, Turkey (e-mail: urhano@kou.edu.tr, sarp@ieee.org).

Contributed Paper
Manuscript received October 6, 2007

deblock filtering at low bit rates to get higher coding quality gain, with the possibility of having in-loop deblock filtering switched off when it is required to reduce the total amount of coding calculation.

An iterative projection on convex sets (POCS) based method is proposed in [12] for image deblocking mainly. In [13], a new smoothness constraint set and its projection operator in the wavelet transform domain based on POCS is presented for HDTV decoded images in order to remove unnecessary high-frequency components caused by blocking artifacts. Another POCS-based approach is presented in [14], and differs from standard POCS based approaches in that it uses pixel-wise local adaptability, with three locally adaptive constraint sets being introduced to improve performance. Local variance and edge information are used together in [15] to reduce blocking and ringing artifacts and it is shown that this method provides better results compared to the POCS based approach presented in [12]. The method proposed in [16] also takes local variance and the edge map into account but uses fuzzy decision making to determine the strength of filtering. A weighted average of pixels symmetrically positioned according to block boundaries is considered for deblocking purposes in [17]. In [18] the surrounding region is classified as smooth or complex for detected blocking effects and smooth areas are simply filtered by a non-linear 1D filter while complex regions are processed using a feed-forward neural network structure. In [19], a 2D deblocking filter is presented instead of the 1D deblocking filter used in H.264 [10] to enable faster implementation. A low complexity deblocking approach which takes local activity at block boundaries into account and applies different kinds of filtering according to the decided mode is presented in [20]. A Markov Random Field (MRF) based deblocking approach which considers the masking property of the HVS is proposed in [21]. The wavelet transform and MRF are utilized together in [22] for improved deblocking performance while preserving true edges and textural information. A non-iterative method is presented in [23] that uses overcomplete wavelet representations to reduce both, blocking and ringing, artifacts. A weighted combination of shifted transforms is used for deblocking and deringing of compressed images in [24]. This approach furthermore uses a multiplier-less transform instead of the discrete cosine transform to enable parallel hardware implementation. The method presented in [25] superimposes some random noise into the compressed blocky image as a first step of deblocking, and then uses an adaptive image denoising technique to remove blocking.

In [26] it has been proposed to utilize an optimal post-processing filter at the decoder to improve warped discrete cosine transform (WDCT) image encoding performance. The filter coefficients are determined at the encoder in least-squares sense, encoded besides the transform coefficients, and used at the decoder to improve the quality of the decoded image. It is shown in [26] that a significant gain in image quality can be achieved by the post-process filtering for image coding.

In this paper, it is proposed to use optimal filtering to improve video compression performance. The coefficients of a linear filter are determined at the encoder so as to minimize the least squares error between the original and filtered frames and these filter coefficients are multiplexed into the bit-stream. Because linear filtering is utilized, the computational complexity induced at the decoder as a result of this process is comparatively low. The optimal filtering approach is evaluated in the form of a post-process filter as well as in the form of an in-loop filter for video coding. It is shown that the proposed approach improves video compression performance.

II. OPTIMAL FILTERING OF COMPRESSED VIDEO FRAMES

It is proposed to utilize an optimal linear frame filter in the decoding stage so that the squared error between the original frame and the filtered reconstructed frame is minimized. If $\hat{I}(i,j)$ is used to denote the reconstructed frame before filtering, the filtered reconstructed frame $\hat{I}_F(i,j)$ is defined as

$$\hat{I}_F(i,j) = \hat{I}(i,j) * \mathbf{G} \tag{1}$$

where $\mathbf{G}$ represents a filter kernel of dimensions $l \times l$, and $*$ represents convolution. It is proposed to obtain the optimal filter coefficients in least squares (LS) sense, by solving

$$\min_{\mathbf{G}} \left\| I(i,j) - \hat{I}_F(i,j) \right\|_2^2 = \min_{\mathbf{G}} \left\| I(i,j) - \hat{I}(i,j) * \mathbf{G} \right\|_2^2 \tag{2}$$

for the filter coefficients, where $I(i,j)$ represents the original frame. While it is possible to use different LS approaches for this purpose, the iterative preconditioned conjugate gradients (IPCG) [27] was used in this paper because of its iterative nature.

For simplicity, assume that the vector $\mathbf{g}$ is used to represent the filter kernel coefficients in row-stacked form so that $\mathbf{g}(m+nl) = \mathbf{G}(m,n), 0 \le m,n \le l-1$, the vector $\mathbf{b}$ is used to represent the original image values in row stacked form so that $\mathbf{b}(i+jw) = I(i,j), 0 \le i \le w$ and $0 \le j \le h$ (where w and h represent the width and height of the frame image), and the matrix $\mathbf{A}$ with dimensions $wh \times l^2$ is

$$\mathbf{A} = \begin{bmatrix} \mathbf{a}_{0,0}^T \\ \mathbf{a}_{0,1}^T \\ \vdots \\ \mathbf{a}_{w,h}^T \end{bmatrix} \tag{3}$$

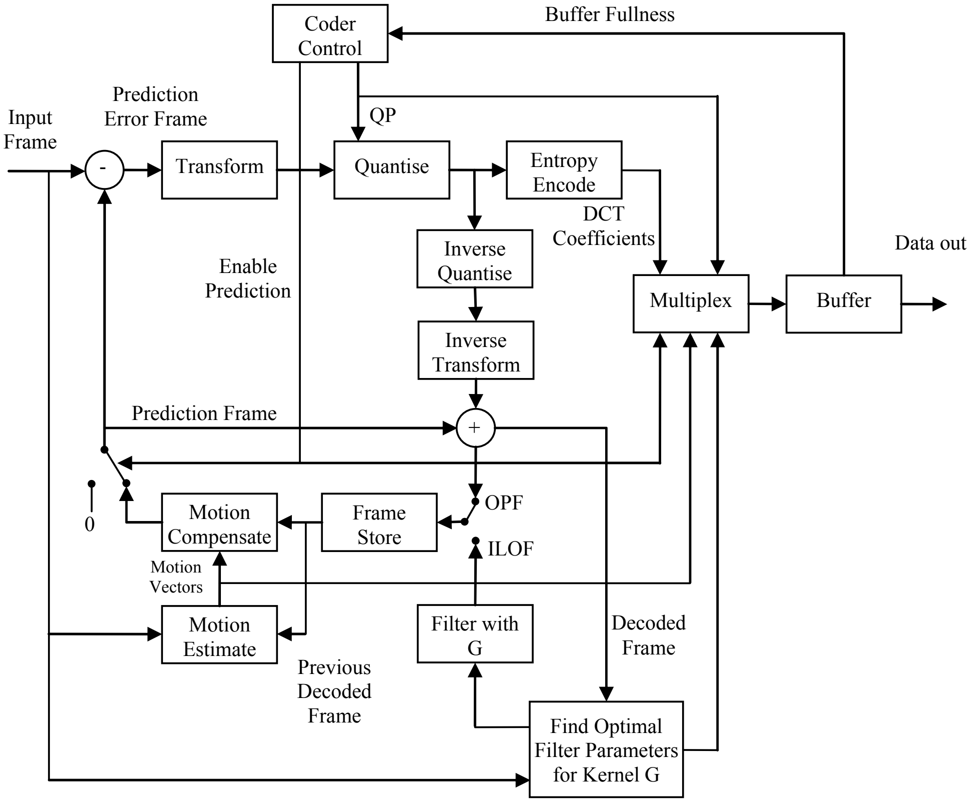

Fig. 1. Overall block diagram of the encoder in the proposed approach for a typical motion compensated predictive coding scheme.

where $\mathbf{a}_{i,j}^{T}$ is the row-stacked form of an $l \times l$ sized 2D window centered around the pixel location (i,j) of $\hat{I}(i,j)$. Then it is possible to express the system to be solved in the form of

$$\mathbf{Ag} = \mathbf{b} \tag{4}$$

In order to solve this equation system using preconditioned conjugate gradients it is necessary to obtain a square coefficient matrix. It is possible to multiply both sides of (4) with $\mathbf{A}^{\mathrm{T}}$ to make the matrix on the left hand side of the equation a square matrix. Then it is possible to solve this new equation using preconditioned conjugate gradients iteratively. If $\bar{\mathbf{A}}=\mathbf{A}^{\mathrm{T}}\mathbf{A}$ and $\bar{\mathbf{b}}=\mathbf{A}^{\mathrm{T}}\mathbf{b}$ is defined for convenience, it is possible to express the system to be solved in the form of

$$\bar{\mathbf{A}}\mathbf{g} = \bar{\mathbf{b}} \tag{5}$$

and the solution of this system will provide a least-squares solution when solved for $\mathbf{g}$. The symmetric positive pre-conditioner $\mathbf{M}$ is constructed so that $\mathbf{M}$ has the diagonal elements of $\bar{\mathbf{A}}$ (diagonal pre-conditioning). The filter kernel is initialized to be equal to the unit impulse response kernel so that $\mathbf{G}_0(m,n) = \delta\left(m-(l-1)/2, n-(l-1)/2\right), 0 \le m,n \le l-1$. The solution of (5) is then carried out in an iterative way. If $\mathbf{r}_k$ is used to denote the residual and $\mathbf{p}_k$ denotes the direction in which the next step should be taken for the k th iteration, the unrolled implementation of the iterative preconditioned conjugate gradients algorithm is implemented as follows [28]:

```
$\mathbf{r}_0=\bar{\mathbf{b}}-\bar{\mathbf{A}}\mathbf{g}_0, \mathbf{z}_0=\mathbf{M}^{-1}\mathbf{r}_0, \boldsymbol{\rho}_0=\mathbf{r}_0\mathbf{z}_0$
$\mathbf{p}_1=\mathbf{z}_0, \mathbf{q}_1=\bar{\mathbf{A}}\mathbf{p}_1, \boldsymbol{\alpha}_1=\frac{\boldsymbol{\rho}_0}{\mathbf{p}_1^{\mathrm{T}}\mathbf{q}_1}$
for i=1,2,...
    $\mathbf{g}_i=\mathbf{g}_{i-1}+\boldsymbol{\alpha}_i\mathbf{p}_i$
    $\mathbf{r}_i=\mathbf{r}_{i-1}-\boldsymbol{\alpha}_i\mathbf{q}_i$
    if residual is small enough exit
    $\mathbf{z}_i=\mathbf{M}^{-1}\mathbf{r}_i$
    $\boldsymbol{\rho}_i=\mathbf{r}_i\mathbf{z}_i$
    $\boldsymbol{\beta}_i=\frac{\boldsymbol{\rho}_i}{\boldsymbol{\rho}_{i-1}}$
    $\mathbf{p}_{i+1}=\boldsymbol{\beta}_i\mathbf{p}_i+\mathbf{z}_i$
    $\mathbf{q}_{i+1}=\bar{\mathbf{A}}\mathbf{p}_{i+1}$
    $\boldsymbol{\alpha}_i=\frac{\boldsymbol{\rho}_i}{\boldsymbol{\rho}_{i+1}^{\mathrm{T}}\mathbf{q}_{i+1}}$
end
```

(6)

The filter coefficients are multiplexed into the bit-stream to facilitate access at the decoder. Experimental results have shown that a kernel size of 5×5 gives acceptable results in terms of balancing the improvement in compression quality and computational load. In this case, a total of 25 filter coefficients need to be encoded per frame as side information. As the purpose of this paper is to basically demonstrate the improvement obtained by the optimal filtering approach; appropriate entropy encoding techniques for filter coefficients as well as faster computation methods will be researched in future work.

An overall encoder block diagram of the proposed approach for a typical motion compensated predictive coding scheme is shown in Fig. 1. Two options for optimal filtering are considered in this work. In the first approach, in-loop optimal filtering (ILOF) is evaluated, where the optimal filtered image frames are used in the coding loop. In the second case, optimal post-process filtering (OPF) is evaluated, in which case optimal filtered image frames are not used in the coding loop.

III. EXPERIMENTAL RESULTS

In order to evaluate the performance of the proposed ILOF and OPF approaches in practical video coding systems, these approaches have been integrated into the H.263+ reference software tmn3.2 and H.264 reference software JM version 11. The H.263+ reference software was used to not only assess the efficiency of the proposed approaches but also evaluate the introduced complexity and effect on speed. The H.264 reference software was used to evaluate the efficiency of the proposed approach in state-of-the-art video coding approaches.

Table I and Table II show average Y-PSNR (luminance – peak signal to noise ratio) versus bit-rate results for the QCIF sized Foreman and Coastguard test sequences, respectively, of length 300 frames, encoded at a frame rate of 10 frames/sec using H.263+. Standard H.263+ results, H.263+ with the proposed in-loop optimal filtering (denoted as H.263+ ILOF), H.263+ with its own standard in-loop deblocking filter (denoted as H.263+ J), H.263+ with the proposed in-loop optimal and its standard deblocking filter together (denoted as H.263+ J ILOF), H.263+ with the proposed optimal post-process filtering (denoted as H.263+ OPF), and H.263+ with the proposed optimal post-process filtering and its standard deblocking filter together (H.263+ J OPF) are used for performance evaluation.

Table I shows that the proposed in-loop optimal filtering improves the PSNR performance by more than 1 dB for low bit-rates and about 0.5 dB for higher bit-rates for the foreman sequence compared to the non-filtered case. Compared to the standard deblocking filter case it is seen that the in-loop optimal filter again provides an improved PSNR performance. It is furthermore seen for the Foreman sequence that if in-loop optimal filtering is used together with the standard deblocking filter the average PSNR is even further increased, but the gain is now less. In case of post-process filtering it is seen that although an increase in PSNR is achieved compared to conventional H.263+, the PSNR results are lower than in-loop filtering, and furthermore combining post-process filtering and deblocking filter does not improve the PSNR more than that achieved by post-process filtering itself. It is useful to note that optimal post-process filtering as well as optimal in-loop filtering provide higher PSNR results than the standard in-loop deblocking filter accepted in H.263+ (Annex J), with optimal in-loop filtering providing the highest PSNR results of all.

Table II shows that the proposed in-loop optimal filtering provides PSNR improvements of more than 1dB for low bit-rates and the gain reduces slightly with increasing bit-rate for the Coastguard sequence. Compared to the standard de-blocking filter, in-loop optimal filtering and optimal post-process filtering again provide higher PSNR performance, and the standard de-blocking filter can be used in addition to in-loop optimal filtering to even further improve the PSNR performance, but the gain is smaller. Therefore taking into account the computational load introduced at the decoder by the nonlinear in-loop deblocking filter, the linear optimal in-loop filter provides superior PSNR performance at lower computational decoder complexity.

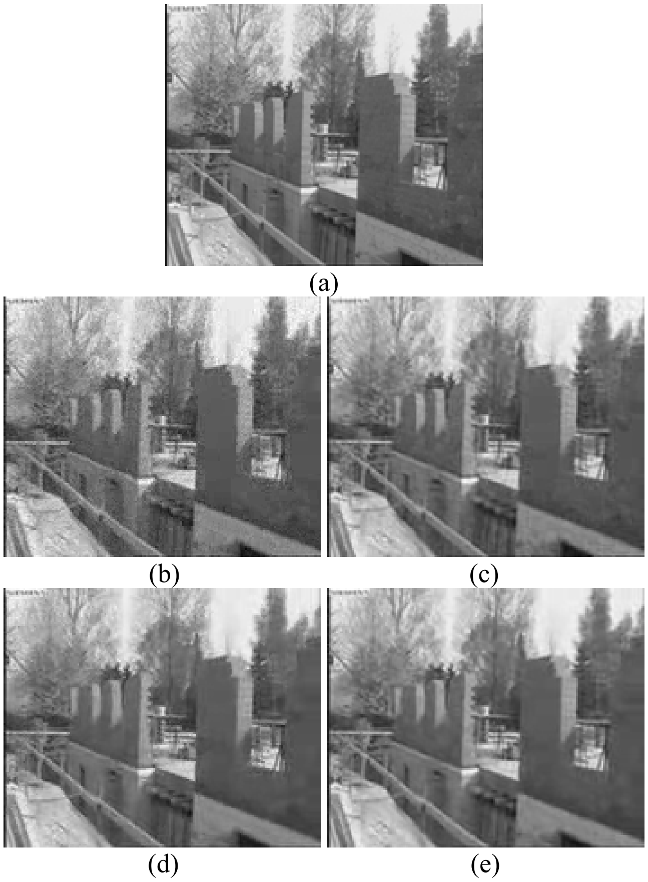
(a)
(b) (c)
(d) (e)

Fig. 2. Foreman frame # 288 (a) Original (b) H.263+ (PSNR: 28.99 dB) (c) H.263+ ILOF (PSNR: 31.62 dB) (d) H.263+ J (PSNR: 31.36 dB) (e) H.263+ J ILOF (PSNR: 31.96 dB)

TABLE I
RATE-DISTORTION RESULTS FOR THE FOREMAN SEQUENCE USING H.263+.

	PSNR (dB)											
Bit-rate (kbps)	25	50	75	100	150	200	250	300	350	400	450	500
H.263+	26.80	28.81	30.55	31.87	33.81	35.48	36.86	38.00	38.94	39.74	40.44	41.05
H.263+ J	27.80	30.21	31.80	33.06	34.80	36.32	37.55	38.57	39.45	40.17	40.85	41.44
H.263+ ILOF	27.91	30.41	32.20	33.52	35.33	36.80	37.98	38.95	39.77	40.45	41.05	41.60
H.263+ J ILOF	28.26	30.80	32.60	33.93	35.70	37.16	38.30	39.27	40.06	40.72	41.34	41.88
H.263+ OPF	27.96	30.41	32.17	33.37	35.24	36.67	37.79	38.73	39.49	40.13	40.70	41.21
H.263+ J OPF	27.98	30.42	32.18	33.37	35.24	36.67	37.79	38.73	39.50	40.14	40.70	41.22

TABLE II
RATE-DISTORTION RESULTS FOR THE COASTGUARD SEQUENCE USING H.263+.

	PSNR (dB)											
Bit-rate (kbps)	25	50	75	100	150	200	250	300	350	400	450	500
H.263+	24.91	27.14	28.65	29.80	31.46	32.93	34.16	35.27	36.29	37.24	38.06	38.77
H.263+ J	25.90	28.30	29.70	30.72	32.17	33.50	34.64	35.65	36.62	37.50	38.28	38.95
H.263+ ILOF	26.35	28.79	30.21	31.27	32.83	34.13	35.26	36.23	37.12	37.95	38.67	39.27
H.263+ J ILOF	26.38	28.91	30.41	31.48	33.01	34.30	35.42	36.37	37.28	38.10	38.79	39.40
H.263+ OPF	25.67	28.70	30.18	31.17	32.66	33.90	34.96	35.82	36.62	37.41	38.05	38.58
H.263+ J OPF	26.30	28.71	30.18	31.17	32.66	33.90	34.96	35.82	36.62	37.42	38.05	38.58

Sample frames of the Foreman sequence encoded at 50 kbps are provided in Fig. 2 to enable visual assessment. Experimental results demonstrate that in-loop optimal filtering can improve the compression efficiency of standard video compression schemes, without introducing visual artifacts. Because a simple linear filter is used for the optimal filtering purpose, the computational complexity induced at the decoder because of the proposed approach will basically be lower compared to non-linear filtering approaches used for deblocking for example.

In order to reduce the complexity of the computation process and facilitate faster execution, loop unrolling, block reordering, fast memory caching and SSE (Streaming SIMD Extension) instructions have been used in the implementation of the optimal filtering. For a QCIF sized frame, the computation of the optimal filter coefficients took on average less than 50 msec per frame on a P4 1.7 GHz Centrino mobile processor with 1GB RAM. Including all other encoding operations, an encoding time of less than 80 msec per frame is obtained including the optimal filter coefficient computation process. Hence, an encoding throughput of about 12.5 frame/sec can be achieved with the reference H.263+ software, even without dedicated hardware. Note that convergence was achieved after 47 iterations on average, but the iteration process is quite fast, while the matrix multiplication is the main reason for computation delay. The image filtering takes only about 10 msec per frame.

Tables III and IV present PSNR results for the QCIF sized Foreman and Coastguard sequences encoded at 30 frames/sec using H.264 with and without the standard deblocking filter and with the proposed ILOF and OPF approaches. For all cases, the Intra period is 30 frames and a single reference

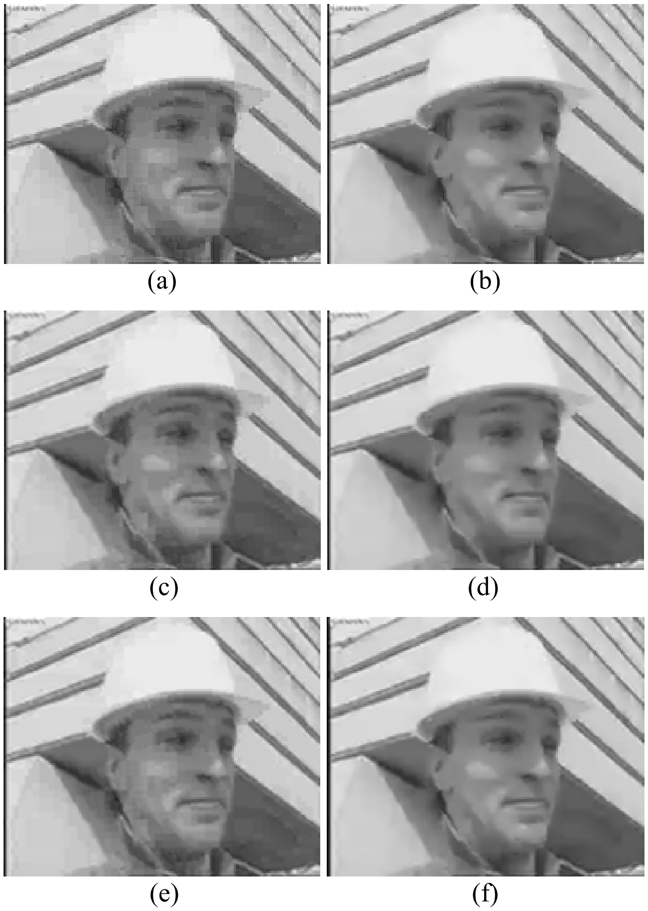
(a) (b)
(c) (d)
(e) (f)

Fig. 3. Foreman frame # 5 (a) H.264 w/o deblocking (PSNR: 30.95 dB) (b) H.264 with deblocking (PSNR: 31.44 dB) (c) H.264 ILOF (PSNR: 31.99 dB) (d) H.264 ILOF with deblocking (PSNR: 32.22 dB) (e) H.264 OPF (PSNR: 31.73 dB) (f) H.264 OPF with deblocking (PSNR: 32.07 dB)

TABLE III
RATE-DISTORTION RESULTS FOR THE FOREMAN SEQUENCE USING H.264.

	PSNR (dB)									
Bit-rate (kbps)	50	100	150	200	250	300	350	400	450	500
H.264 w/o Deblocking	29.40	32.09	33.70	34.96	36.01	36.94	37.70	38.42	38.93	39.61
H.264 with Deblocking	29.83	32.42	34.04	35.27	36.26	37.15	37.88	38.59	39.10	39.74
H.264 ILOF	30.27	32.91	34.55	35.80	36.77	37.62	38.36	39.04	38.52	40.12
H.264 ILOF with Deblocking	30.46	33.09	34.70	35.95	36.91	37.75	38.47	39.16	39.62	40.22
H.264 OPF	30.21	32.88	34.48	35.69	36.70	37.58	38.32	38.99	39.49	40.12
H.264 OPF with Deblocking	30.43	33.06	34.68	35.88	36.85	37.72	38.43	39.11	39.60	40.22

TABLE IV
RATE-DISTORTION RESULTS FOR THE COASTGUARD SEQUENCE USING H.264.

	PSNR (dB)									
Bit-rate (kbps)	50	100	150	200	250	300	350	400	450	500
H.264 w/o Deblocking	28.25	30.43	31.95	33.11	34.01	34.90	35.68	36.36	36.60	37.29
H.264 with Deblocking	28.53	30.64	32.18	33.33	34.19	35.04	35.78	36.47	36.72	37.39
H.264 ILOF	29.01	31.14	32.62	33.76	34.63	35.47	36.20	36.84	37.08	37.70
H.264 ILOF with Deblocking	29.20	31.28	32.76	33.88	34.73	35.55	36.25	36.90	37.15	37.74
H.264 OPF	28.82	30.98	32.47	33.61	34.49	35.36	36.11	36.78	37.00	37.67
H.264 OPF with Deblocking	28.98	31.09	32.61	33.75	34.61	35.45	36.18	36.85	37.09	37.74

frame is used. The proposed ILOF and OPF approaches provide superior PSNR performance for H.264 as well. The PSNR results obtained using H.264 with ILOF and OPF are significantly higher than H.264 without any deblocking filtering and even higher by about 0.5 dB for H.264 with deblocking filter, particularly at lower bit-rates. The proposed ILOF and OPF approaches still provide improvements in PSNR even at higher bit-rates. Experimental results show that the PSNR performance can even be increased if ILOF or OPF are used in conjunction with the conventional deblocking filter. It is seen that ILOF is typically superior to OPF.

Fig.3 shows sample frames of the Foreman sequence encoded at 50 kbps using H.264 with and without the proposed ILOF and OPF approaches as well as the original deblocking filter. It is seen that ILOF by itself improves the visual appearance significantly compared to original H.264 without the deblocking filter. As ILOF requires the decoder to perform simple linear filtering, the computational load induced at the decoder is much lower than that introduced by the nonlinear deblocking filter. Therefore it is possible to use ILOF without deblocking filter for consumer electronics applications that require low complexity decoding, which is particularly the case for portable video communications equipment. ILOF can be used in conjunction with the conventional deblocking filter to further improve the performance if the computational load of the deblocking filter is acceptable.

IV. CONCLUSION

Novel in-loop optimal filtering and optimal post-process filtering approaches are proposed in this paper to improve video compression performance. In the proposed approaches, the coefficients of a linear filter are determined at the encoder so as to minimize the least squared error between the original and filtered frame and these filter coefficients are multiplexed into the bit-stream for decoder access. It is shown that the proposed approaches can improve video compression performance. Unlike previous in-loop or post filtering schemes that basically aim to reduce blocking effects, the proposed approach genuinely aims to improve the overall compression quality in an optimal way by minimizing the error between the original and filtered reconstructed frames.

Results demonstrated in this paper show that in-loop optimal filtering and optimal post process filtering can successfully be integrated into compression schemes. In-loop optimal filtering shows in overall superior performance compared to optimal post-process filtering.

REFERENCES

[1] Y. Xu, and Y. Zhou, “H.264 video communication based refined error concealment schemes,” IEEE Trans. Consumer Electronics, vol. 50, no. 4, pp. 1135-1141, Nov. 2004.

[2] C-W. Ho, O.C. Au, S-H. G. Chan, S-K. Yip, and H-M. Wong, “Low-Complexity Rate Control for Efficient H.263 to H.264/AVC Video Transcoding,” IEEE International Conference on Image Processing, ICIP 2006, vol. 1, pp. 853-856, Oct. 2006.

[3] J-F. Yang, H-M. Hang, E. Steinbach, and M-T. Sun, “Advanced Video Technologies and Applications for H.264/AVC and Beyond,” EURASIP Journal on Applied Signal Processing, vol. 2006, Article ID 27579, 3 pages, doi:10.1155/ASP/2006/27579

[4] C.S. Kannangara, I.E.G. Richardson, M. Bystrom, J.R. Solera, Z. Yafan, A. MacLennan, R. Cooney, “Low-complexity skip prediction for H.264 through Lagrangian cost estimation,” IEEE Trans. Circuit Syst. Video Technol., vol. 16, no. 2, pp. 202-208, Feb.2006.

[5] A. Ertürk, and S. Ertürk, “Two-Bit Transform for Binary Block Motion Estimation,” IEEE Trans. Circuit Syst. Video Technol., vol. 15, no. 7, pp. 938- 946, Jul. 2005.

[6] S. Ertürk, “Multiplication-free one-bit transform for low-complexity block-based motion estimation,” IEEE Signal Process. Lett., vol. 14, no. 2, pp. 109-112, Feb. 2007.

[7] A.M. Tourapis, F. Wu, S. Li, “Direct macroblock coding for predictive (P) pictures in the H.264 standard,” SPIE Visual Communications and Image Processing (VCIP), vol. 5308, pp 364-371, Jan. 2004.

[8] S. Ertürk, “A New Perspective to Block Motion Estimation for Video Compression: High-Frequency Component Matching,” IEEE Signal Process. Lett., vol. 14, no.2, pp. 113-116, Feb.2007.

[9] G. Cote, B. Erol, M. Gallant, and F. Kossentini, “H.263+: Video Coding at Low Bit Rates,” IEEE Trans. Circuits Syst. Video Technol., vol. 8, no.7, pp. 849-866, Jul. 1998.

[10] P. List, A. Joch, J. Lainema, G. Bjontegaard, and M. Karczewicz, “Adaptive deblocking filter,” IEEE Trans. Circuits Syst. Video Technol., vol. 13, no. 7, pp. 614-619, Jul. 2003.

[11] K. Chono, and Y. Miyamoto, “Study on in-loop filtering issues,” Input Document JVT-B078, Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG (ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6) Meeting, Feb. 2002.

[12] A. Zakhor, “Iterative procedures for reduction of blocking effects in transform image coding,” IEEE Trans. on Circuits Syst. Video Technol., vol.2, no.1, pp. 91-95, Jan. 1992.

[13] G.-R. Kwon, H.-K. Kim, Y. Kim, and S.-J. Ko, “An Efficient POCS-based Post-processing Technique Using Wavelet Transform in HDTV,” IEEE Trans. Consumer Electron., vol. 51, no.4, pp. 1283-1290, Nov. 2005.

[14] J.J. Zou, H. and Yan, “A Deblocking Method for BDCT Compressed Images Based on Adaptive Projections,” IEEE Trans. on Circuits Syst. Video Technol., vol. 15, no. 3, pp. 430-435, Mar. 2005.

[15] H.-S. Kong, A. Vetro, and H. Sun, “Edge map guided adaptive post-filter for blocking and ringing artifacts removal,” Proc. IEEE Int. Symp. On Circuits and Systems, (ISCAS '04), Vancouver, Canada, vol. 3, pp. 929-932, May 2004.

[16] H.-S. Kong, Y. Nie, A. Vetro, S. Huifang, and K.E. Barner, “Adaptive fuzzy post-filtering for highly compressed video,” Proc. IEEE Int. Conf. on Image Processing, (ICIP '04), Singapore, vol. 3, pp. 1803-1806, Oct. 2004.

[17] A.Z.Averbuch, A. Schclar, and D.L. Donoho, D.L., “Deblocking of Block-Transform Compressed Images Using Weighted Sums of Symmetrically Aligned Pixels,” IEEE Trans. on Image Process., vol. 14, no. 2, pp. 200-212, Feb. 2005.

[18] K.-K.Kwon, S.-H. Im, and D.-S. Lim, “Deblocking algorithm in MPEG-4 video coding using block boundary characteristics and adaptive filtering,” Proc. IEEE Int. Conf. on Image Processing, (ICIP’05), Genova, Italy, vol. 3, pp. 541-544, Sept. 2005.

[19] M.N. Bojnordi, M.R. Hashemi, and O. Fatemi, “A Fast Two Dimensional Deblocking Filter for H.264/AVC Video Coding,” Proc. Canadian Conf. on Electrical and Computer Eng., Ottawa, Canada, pp. 2017 – 2020, May 2006.

[20] S.-C.Tai, Y.-R. Chen, C.-Y. Chen, and Y.-H. Chen, “Low complexity deblocking method for DCT coded video signals,” IEE Proc.-Vis. Image Signal Process., vol. 153, no. 1, pp. 46-56, Feb. 2006.

[21] S. Xie, and Z. Xu, “An adaptive de-blocking algorithm based on MRF,” Proc. 17th IEEE Int. Conf. on Tools with Artificial Intelligence (ICTAI’05), Hong Kong, pp. 14-16, Nov. 2005.

[22] M. Shi, Q. Yi and J. Gong, “A New Deblocking Algorithm Based on Wavelet Transform and MRF for DCT-Coded Images,” Proc. Canadian Conf. on Electrical and Computer Eng., Ottawa, Canada, pp. 1944-1947, May 2006.

[23] A.W.-C.Liew, and H. Yan, “Blocking Artifacts Suppression in Block-Coded Images Using Overcomplete Wavelet Representation,” IEEE Trans. on Circuits Syst. Video Technol., vol. 14, no.4, pp. 450-461, Apr. 2004..

[24] R. Samaduni, A. Sundararajan, and A. Said, “Deringing and deblocking DCT compression artifacts with efficient shifted transforms,” Proc. IEEE Int. Conf. on Image Processing, (ICIP '04), Singapore, vol. 3, pp. 1799-1802, Oct. 2004.

[25] Z. Long, and N. Younan, “Adaptive Deblocking of Images with DCT Compression,” Proc. Int. Conf. on Image Processing, Computer Vision, & Pattern Recognition (IPCV'06), Las Vegas, USA, pp. 430-435, Jun. 2006.

[26] O. Urhan, and S. Ertürk, “Parameter Embedding Mode and Optimal Post-Process Filtering For Improved WDCT Image Compression,” IEEE Trans. on Circuits Syst. Video Technol., Accepted for publication.

[27] M.R. Hestenes, and E. Stiefel, “Methods of Conjugate Gradients for Solving Linear Systems,” Journal of Research of the National Bureau of Standards, vol. 49, no. 6, pp. 409-436, 1952.

[28] S. Romero, and L.F. Romero, “An Optimized Preconditioned Conjugate Gradient Algorithm,” Technical Report No: UMA-DAC-02/11, University of Malaga, Sept. 2002. [Online]: www.ac.uma.es/inv-des/inves/reports/2002/UMA-DAC-02-11.ps.gz

Dong-Hwan Kim received his B.S. and M.S. degrees in Electrical and Electronics Engineering from the Chung-Ang University, Seoul, Korea in 2003 and 2005, respectively. He is a Ph.D. candidate in Electrical and Electronics Engineering from the Chung-Ang University. His research interests are in the area of multimedia signal processing and digital communications.

Hwa-Yong Oh. received his B.S., M.S. and Ph.D. degrees from Chung-Ang University, Seoul, Korea, in 2003, 2005 and 2007, respectively, both in electrical engineering. Now he is working at Signals and Communication Systems LAB in Chung-Ang University. His main research interests are multimedia signal processing, wireless communication systems and embedded system.

Oğuzhan Urhan (S’02-M’06) received his B.Sc., M.Sc., and PhD degrees in electronics and telecommunications engineering from the University of Kocaeli, Kocaeli, Turkey, in 2001, 2003, and 2006, respectively. Since 2001 he has been with the Department of Electronics and Telecommunications Engineering, University of Kocaeli, Turkey, where he is currently appointed as assistant professor. He worked as a visiting professor at Chung-Ang University, Korea, between September 2006 and March 2007. His research interests are in the area of digital signal and image processing, in particular, image and video restoration and coding.

Sarp Ertürk (M’99) received his B.Sc. in Electrical and Electronics Engineering from Middle East Technical University, Ankara in 1995. He received his M.Sc. in Telecommunication and Information Systems and Ph.D. in Electronic Systems Engineering in 1996 and 1999 respectively from the University of Essex, U.K. From 1999 to 2001 he carried out his compulsory service at the Army Academy, Ankara. He is currently appointed as Full Professor at Kocaeli University, where he worked as Assistant Professor between 2001 and 2002, and Associate Professor between 2002 and 2007. Between March-September 2006 he was a visiting professor at the School of Electrical and Electronics Engineering, Chung-Ang University, Seoul, Korea. His research interests are in the area of digital signal and image processing, video coding, remote sensing and digital communications.

Tae-Gyu Chang. (M’86) received the B.S. degree from the Seoul National University, Seoul, Korea, in 1979, and M.S. degree from Korea Advanced Institute of Science and Technology, Seoul, in 1981, and Ph.D. degree from University of Florida, Gainesville, in 1987, all in electrical engineering. From 1981 to 1984, he was with the Hyundai Engineering/Electronics Inc., Seoul, as a Computer Systems Design Engineer. From 1987 to 1990, he was a Faculty Member of Tennessee State University, Nashville, as an Assistant Professor of information systems engineering. In March 1990, he joined the faculty of the Chung-Ang University, Seoul, where he is currently a Professor at the Department of Electrical and Electronics Engineering. His research interests include multimedia signal processing and communications.