ADAPTIVE SPATIO-TEMPORAL FILTERING FOR VIDEO DE-NOISING

Hye-Yeon Cheong†, Alexis Michael Tourapis, Joan Llach, Jill Boyce

Department of Electrical Engineering - Columbia University†
Thomson Corporate Research, Princeton, NJ 08540, USA
hyeyeon@ieee.org†, {Alexandros.Tourapis, Joan.Llach, Jill.Boyce}@thomson.net

ABSTRACT

Video de-noising is an important feature of many modern video encoding architectures since it can considerably enhance coding efficiency while at the same time improving objective and subjective quality. In this paper, a new scheme for video de-noising based on spatio-temporal filtering is presented, which can be used as a pre-processing stage within a video encoder. Spatial filtering is performing through the adaptive selection and combination of a wavelet based and a 2-D Wiener filter, while for temporal filtering we employ bi-directional block based motion compensation using the Enhanced Predictive Zonal Search (EPZS) algorithm. Experimental results are presented which show a significant improvement in video quality when the de-noised video sequence is encoded with an H.264 encoder.

1. INTRODUCTION

Digital still or video images can contain noise due to the capturing or analog to digital conversion process, or even due to transmission reasons. Noise, nevertheless, apart from the visual displeasing impact it may have, can also have a severe adverse effect in many applications and especially video compression. Due to its random nature, noise can considerably decrease spatial and temporal correlation thus limiting the coding efficiency of such noisy video signals. Thus, it is desirable to remove noise without however removing any of the important details of the image, such as edges or texture.

Several Video De-noising architectures already exist in the literature where de-noising is performed by either considering spatial [2] or temporal filtering methods, or a combination thereof. Even the most advanced spatial methods, such as Wiener [2] or wavelet filtering [3], tend to be more appropriate for still images, while, due to their nature, temporal and spatio-temporal methods are more appropriate for video signals due to the temporal correlation that exists between adjacent pictures. A rather detailed review of such methods can be found in [1]. Such methods can be classified into motion and non-motion compensated filters [4], which may consider or not motion estimation and compensation techniques for filtering the current picture.

In [5], we presented a spatio-temporal video de-noising architecture combined with an H.264[6] video encoder [7]. Spatial filtering was performed on all pixels using a threshold based 3×3 pixel average, while the motion estimation process of H.264 was reused for performing the temporal filtering. Considering that the H.264 video coding standard allows the consideration and use of multiple references for predicting a block or macroblock, it is possible using this strategy to generate several possible temporal predictions for the current pixel. These temporal predictions were then averaged together to form the final filtered picture. It should be noted that in this approach, motion estimation and compensation were always performed on previously filtered pixels. Although this process could result in the generation of a more accurate motion field, this process could also result in some cases in the removal of some of the more refined details of a scene such as texture or edges.

In this paper, we will present an improved spatio-temporal filter that, unlike our previous approach, considers a different spatial filtering method based on a combination of wavelet decomposition and Wiener filtering. Furthermore, we consider both past and future pictures during motion estimation and compensation, thus improving the temporal filtering of the algorithm, while these are later combined using a weighted averaging process.

In Section 2 we will describe the spatial filtering method employed by our proposed system. In Section 3 we will then discuss the temporal filtering method. Finally, experimental results will be given in Section 4, followed by a conclusion in Section 5.

2. SPATIAL FILTERING USING WAVELET DECOMPOSITION AND 2D WIENER FILTERING

Wavelet based de-noising methods are amongst the highest performing methods used for image and video de-noising [1][3]. Considering that in most cases noise tends to be more prominent within the high frequencies of an image, it is possible by performing a wavelet decomposition to essentially isolate the noise and more efficiently remove it.

In general, wavelet decomposition is performed by first filtering an image $f(x,y)$, where x and y represent the columns and rows of the image, in a given dimension (i.e. rows), using a low pass filter $G(x)$ and a high pass filter $H(x)$ in parallel. These two filtered images are then decimated resulting into two new lower resolution images $f_L(x,y)$ and $f_H(x,y)$, the first containing the low and the second the high

0-7803-8554-3/04/$20.00 ©2004 IEEE.

frequencies of the image. This step is then repeated for the other image dimension (i.e. columns) for both images. This would result into four different images, or bands, specifically $f_{LL}(x,y)$, $f_{LH}(x,y)$, $f_{HL}(x,y)$, and $f_{HH}(x,y)$. These bands could be further refined into smaller bands by reapplying the same method to each one separately. Reconstruction can be performed through up-sampling and filtering with properly constructed filters that are related to $G(x)$ and $H(x)$ in a way to either achieve perfect reconstruction of $f(x,y)$ or minimize the reconstruction error. We would refer the reader to [8] for further information on Wavelets. In general, most wavelet based image de-noising algorithms consider each band separately and apply different filtering methods in each band to reduce the noise.

As we have previously mentioned, noise tends to affect mainly the higher frequencies of an image. Due to this fact, we observe that by selecting a good decomposition filter it may be possible to remove most of the noise by simply discarding the highest band, which in the above case is $f_{HH}(x,y)$. This could considerably simplify our design. This process can be seen in Figure 1, where essentially we may even avoid generating certain bands since these will not be used during the filtering process. For our proposed architecture the well-known 64-tap Johnston filter [9] was selected, which is a perfect reconstruction filter. Other smaller filters could also be used (e.g. 32- or 8-tap Johnston, JPEG2000 (5,3), (9,7) or (13,9), etc.) but have not yet been tested. In general though, it would be desirable for such a filter to be relatively short and discrete thus allowing a simpler implementation especially in hardware. Nevertheless, since we did not intend our current implementation for such architecture, a more complex filter was selected. We will now call the reconstructed image using this method as $f'_{Sp1}(x,y)$.

It is however possible in some cases for the above method to fail in removing most of the noise. For this reason we have selected to also create an additional spatial filter hypothesis $f'_{Sp2}(x,y)$ from the original image using a 2-D low pass Wiener filter [2]. This filter essentially depends on the characteristics of the current image and the existing noise. Unfortunately we have observed that this filter, although it can have great results on smooth regions and in the presence of significant noise, it tends to also remove fine details such as texture or in some cases severely blurs the image. In this case this filtered hypothesis could prove to be highly unreliable and should not be considered. For this purpose, for an image of size N×M, we also consider the mean absolute difference (*MAD*) between $f'_{Sp1}(x,y)$ and $f'_{Sp2}(x,y)$ which is defined as follows:

$$MAD = \sum_{x=0}^{N}\sum_{y=0}^{M}\left|f'_{Sp1}(x,y) - f'_{Sp2}(x,y)\right|. \qquad (1)$$

Based on this metric, the hypothesis $f'_{Sp2}(x,y)$ is only considered within our system if only $MAD < 5$. Otherwise $f'_{Sp2}(x,y) = f'_{Sp1}(x,y)$.

3. BLOCK BASED MOTION COMPENSATED TEMPORAL FILTERING

As we have previously mentioned, temporal filtering can further enhance the performance of a video de-noising system by exploiting the temporal correlation of adjacent pictures. This is achieved by performing motion estimation and compensation to construct additional temporal hypotheses that are later combined with the spatially filtered hypotheses. Similar to [5], a block based motion estimation algorithm is used for performing motion estimation, namely the well-known Enhanced Predictive Zonal Search (EPZS) [10]. This algorithm, apart from being highly efficient in terms of complexity, can estimate rather accurately the true motion of a sequence, which is rather vital for our architecture. This has already been exploited previously using similar algorithms [11] to perform temporal interpolation [12] or de-interlacing [13]. We will refer the reader to [10] for more details on this algorithm.

Unlike [5], where only past pictures were used for performing temporal filtering, in this approach future frames are also considered. This is very similar to the concept of Bi-directional pictures used in many video coding standards. More specifically, we may select *m* past and *n* future pictures, perform motion estimation on each one of them and based on the generated motion vectors construct a temporal prediction for the current picture. For complexity purposes we have selected $m=n=2$. Assuming that the current picture $f(x,y,t)$ is at time t, this implies that we will consider the pictures $f(x,y,t-2)$, $f(x,y,t-1)$, $f(x,y,t+1)$, and $f(x,y,t+2)$. Motion estimation using these four images is performed by splitting $f(x,y,t)$ into non-overlapping blocks of size $B_1 \times B_2$ and by performing a search within a small window in each one of these reference pictures in order to locate the best possible match. The matching criterion used is the SAD (Sum of

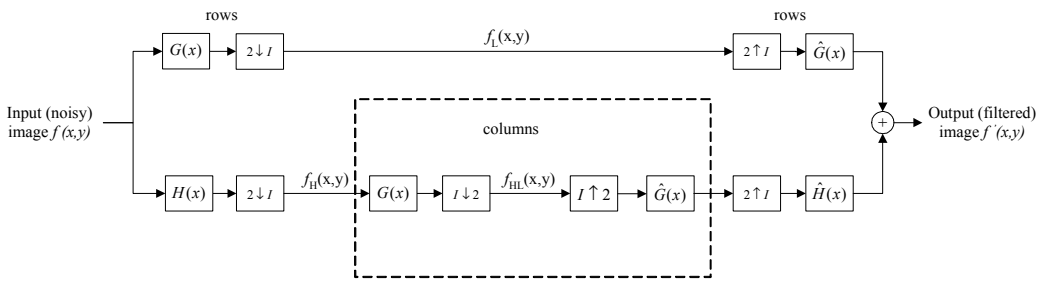


Figure 1: Proposed Wavelet Based Spatial Filtering Method

Absolute Differences), which is computed as:

$$SAD(c, r(\mathbf{m})) = \sum_{x=1,y=1}^{B_1,B_2} |c[x,y] - r[x-m_x, y-m_y]|, \quad (2)$$

with c and r being the blocks of the current and reference picture respectively. In our approach we have selected B_1=B_2=8.

To more accurately predict motion and enhance performance, subpixel motion estimation and compensation down to quarter-pixel positions were used. Subpixel values were generated by considering the same filters as in H.264, more specifically the 6-tap filter [1 -5 20 20 -5 1]/32 for half-pixel, and the bilinear filter for quarter-pixel positions. Obviously higher precision could also be considered, but was not considered due to complexity constraints.

Considering that in some cases, especially during complicated and/or high motion, motion estimation may fail, possibly resulting in visual artifacts, we also introduce a threshold mechanism within the motion estimation process. If the minimum SAD value for the current block from a given reference picture is above a threshold *T=512,* then for this reference the prediction is equal to the original block. This can be seen as rather similar to intra decision within a video encoder.

After motion estimation, motion compensation is performed to generate the temporal hypotheses that will be used for temporal filtering. Although motion estimation can be more efficient if performed on spatially filtered pictures, motion compensation is performed on the original unfiltered pictures, thus allowing us to retain the high frequencies that could be lost due to the spatial filtering process. On the other hand, considering that noise tends to be completely uncorrelated from one picture to another, this would be in general canceled during the temporal filtering. The hypotheses generated through this process are $f'_T(x,y,t-2)$, $f'_T(x,y,t-1)$, $f'_T(x,y,t+1)$, and $f'_T(x,y,t+2)$.

In a final step, these temporal hypotheses are combined with the spatial hypotheses $f'_{Sp1}(x,y)$ and $f'_{Sp2}(x,y)$ using weighted averaging, to generate the final de-noised image $\hat{f}(x,y,t)$ as follows:

$$\hat{f}(x,y,t) = w_{Sp1} f'_{Sp1}(x,y,t) + w_{Sp2} f'_{Sp2}(x,y,t) + \sum_{k}^{\{-2,-1,1,2\}} w_k f'_T(x,y,t+k)$$

Weights could be adaptive (e.g. based on SAD), or could even be fixed and depend on the temporal distance and reliability of each hypotheses. For our implementation the following equation for generating $\hat{f}(x,y,t)$ was used:

$$\hat{f}(x,y,t) = \left\lfloor \frac{18 f'_{Sp1}(x,y,t) + 4 f'_{Sp12}(x,y,t) + \sum_{k}^{\{-2,-1,1,2\}} \frac{4 f'_T(x,y,t+k)}{k^2}}{32} \right\rfloor$$

4. SIMULATION RESULTS

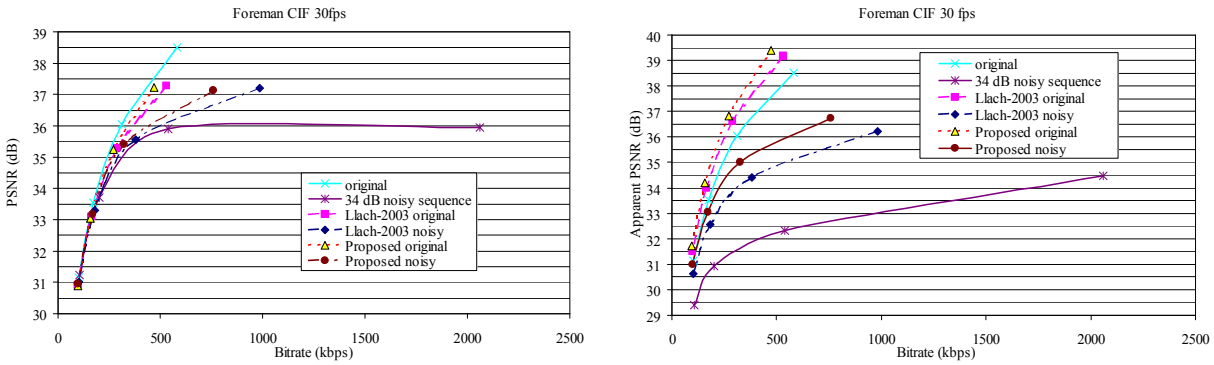
To evaluate the performance of the proposed scheme zero mean Gaussian noise with σ_N = 5 was added to CIF resolution sequences *Foreman*, *Mother & Daughter*, *News*, *Stefan*, and *Table Tennis*. All sequences had 292 frames and were de-noised using the scheme in [5], labeled as Llach-2003, and our new proposed approach. For [5], we completely separated the de-noiser from the encoder, and no H.264 encoding was performed. The PSNR in dB of the de-noised sequences compared to the original, noise-free, sequence was then computed.

Table 1: Performance Evaluation of the Proposed Scheme

Sequence	Llach-2003	Proposed	Difference
Foreman	+2.84dB	+3.58dB	*+0.74dB*
Mother & Daughter	+4.08dB	+4.39dB	*+0.31dB*
News	+3.77dB	+4.03dB	*+0.26dB*
Stefan	+1.65dB	+2.48dB	*+0.83dB*
Table Tennis	+2.44dB	+2.97dB	*+0.53dB*
Average	+2.96dB	+3.49dB	*+0.53dB*

The complete results in PSNR improvement versus the noisy sequences are shown in Table 1. We observe that the new scheme can achieve up to 0.83dB (for sequence Stefan), or 0.53dB on average, additional improvement compared to our previous scheme in [5].

In a second test, the original clean version of foreman was filtered with the proposed scheme and [5]. These sequences, and their noise added counterparts were all coded using the latest H.264 encoder [7]. Main profile was used



(a) (b)

Figure 2: RD plots for Foreman using a) *normal* PSNR and b) *apparent* PSNR.

using a single reference and QP values of 24, 28, 32, and 36. Simulation results can be seen in Figure 2. In this figure we can see the RD performance compared to the original clean sequence (Figure 2a) but also the *apparent RD performance* for each sequence (Figure 2b). The first graph presents the PSNR of each encoded sequence compared to the original, clean sequence, while the second presents the PSNR of the encoded sequence compared to the current source. As expected, the filtered sequences have a lower RD plot compared to the original sequence, considering that the filtering process has considerably altered the content, but on the other hand have a higher apparent RD performance. The apparent RD performance suggests that the filtering has made the content considerably easier to encode, especially at higher bitrates, since most of the uncorrelated noise, or other less important details have been removed from the actual content. This is more obvious when comparing the subjective performance of each scheme. The filtered sequences tend to have less blocking artifacts and are more visually pleasing than the original encoded at a higher bitrate as can also be seen from Figure 3. Furthermore temporal variability in quality is considerably less noticeable since the noise-filtered sequences are more temporally correlated. Finally, we observe that the proposed scheme has both a higher RD and an apparent RD plot than the scheme in [5], which demonstrates its superior performance. It needs to be mention that the apparent RD performance needs to be used with relative caution since very strong spatial filtering could also lead to a seemingly good curve but nevertheless considerably distort and blur the original content.

Our new strategy could be further improved through the consideration of more advanced spatial filtering techniques [3][4], and further refinement of the motion estimation process with the use of smaller block types and subpixel units. An adaptive weighting process instead of the current fixed weight method for the temporal filtering could also be used.

5. CONCLUSION

In this paper a new spatio-temporal video de-noising architecture was presented, which combines wavelet decomposition, Wiener filtering, and block based motion estimation and compensated techniques. Our proposed strategy can considerably reduce undesirable noise within a video sequence, while at the same time allows a video encoder, such as H.264, to achieve improved objective and subjective quality at a given bitrate compared to similar architectures.

6. REFERENCES

[1] J.C. Brailean, R.P. Kleihorst, S. Efstratiadis, A.K. Katsaggelos, and R.L. Lagendijk, "Noise reduction filters for dynamic image sequences: A review," *Proceedings of the IEEE*, vol. 83, pp. 1272-1292, Sept.'95.

[2] A.K. Jain, *Fundamentals of Digital Image Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1989.

[3] C.Q. Zhan and L.J. Karam, "Wavelet-Based Adaptive Image Denoising With Edge Preservation," *Proc. of the IEEE 2003 Int. Conf. on Image Process.*, MA-L2, Sept.'03.

[4] M.K. Özkan, A.T. Erdem, M.I. Sezan, and A.M. Tekalp, "Efficient Multiframe Wiener restoration of blurred and noisy image sequences," *IEEE Transactions on Image Processing*, vol. 1, pp. 453-476, Oct.'92.

[5] J. Llach, J.M. Boyce, "H.264 encoder with low complexity noise pre-filtering," *Proc. SPIE, Applications of Digital Image Processing XXVI*, vol. 5203, p. 478-489, Aug.'03.

[6] *Advanced video coding for generic audiovisual services*, http://www.itu.int/rec/recommendation.asp?type=folders&lang=e&parent=T-REC-H.264.

[7] JVT Reference Software official version 7.5b, *http://bs.hhi.de/~suehring/tml/download/jm75b.zip*.

[8] M. Vetterli, J. Kovacevic, *Wavelets and Subband Coding*. Englewood Cliffs, NJ: Prentice-Hall, 1995.

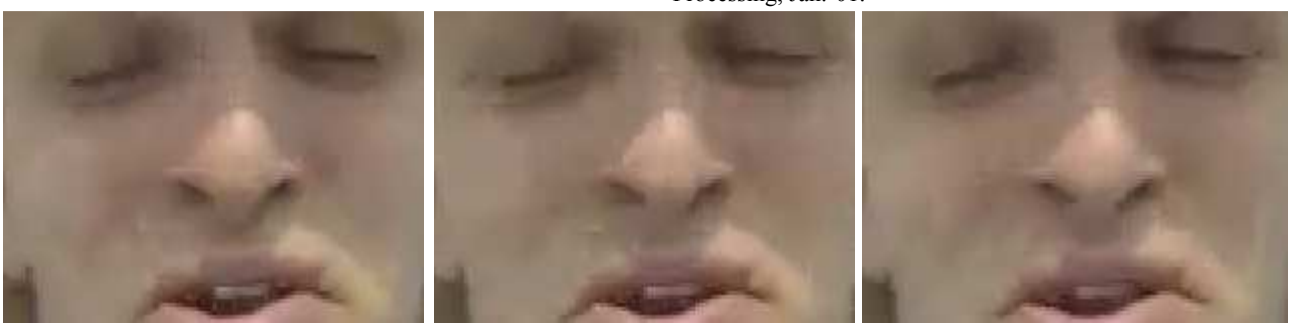
[9] J.D. Johnston, "A Filter Family Designed for Use in Quadrature Mirror Filter Banks," *Proc. ICASSP*, vol. 1, pp. 291-294, Denver, CO, Apr.'80.

[10] H.Y. Cheong, A.M. Tourapis, "Fast motion estimation within the H.264 codec," *Proc. of the Intern. Conf. on Mult. and Expo (ICME '03)*, vol. 3, pp. 517-520, July'03.

[11] A.M.Tourapis, O.C.Au, and M.L.Liou, "Highly efficient predictive zonal algorithms for fast block-matching motion estimation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, iss. 10, pp. 934- 947, Oct.'02.

[12] A.M. Tourapis, H.Y. Cheong, O.C. Au, M.L. Liou, "Temporal Interpolation of Video Sequences using Zonal-based Algorithms", Proc. Of IEEE Int. Conf. On Image Processing, Thessaloniki, Greece, Oct.'01.

[13] A.M. Tourapis, O.C. Au, M.L. Liou, "Advanced De-Interlacing Techniques with the use of Zonal Based Algorithms", Proc. Of SPIE Conf. On Visual Communication and Image Processing, Jan.'01.



(a) (b) (c)

Figure 3: Portion of frame 66 of Foreman encoded a) without filtering, b) using Llach-2003, and c) proposed using QP 36.