Aliasing reduction via frequency roll-off for scalable image/video coding

Yongjun Wu and John W. Woods

Center for Image Processing Research

Rensselaer Polytechnic Institute, Troy, NY, 12180-3590, USA

ABSTRACT

The extracted low resolution video from a motion compensated 3-D subband/wavelet scalable video coder is unnecessarily sharp and sometimes contains significant aliasing, compared to that by the MPEG4 low pass filter. In this paper, we propose a content adaptive method for aliasing reduction in subband/wavelet scalable video coding. We try to make the low resolution frame (LL subband) visually similar to that of the MPEG4 decimation filter through frequency roll-off. Scaling of the subbands is introduced to make the variances of the subbands comparable in these two cases. Thanks to the embedded properties of the EZBC coder, we can achieve the needed scaling of energies in each subband by sub-bitplane shift in the extractor and coefficient scaling in the decoder. An analysis is presented for the relationship between sub-bitplane shift and scaling, which shows that our selected sub-bitplane shift works well for high to medium bitrates. Two different energy-matching structures, i.e. a dyadic decomposition and non-dyadic decomposition, are proposed. The first dyadic method has low complexity but low accuracy for energy matching, while the second non-dyadic method is more accurate for energy matching but has higher analysis/synthesis complexity. The first scheme introduces no PSNR loss for full resolution video, while the second one introduces a slight PSNR loss for full resolution, due to our omission of interband context modeling in this case. Both methods offer substantial PSNR gain for lower spatial resolution, as well as substantial reduction in visible aliasing.

Keywords: scalable video coding, aliasing reduction, motion compensated temporal filtering, frequency roll-off

1. INTRODUCTION

Highly scalable video coding based on a motion compensated 3-D subband/wavelet decomposition has emerged as a promising area in video processing research and an important component in interactive multimedia technology. As specified in the *Call for proposals on scalable video coding technology*,[1] the main scalability requirements for video are temporal, spatial, and SNR scalability. For spatial scalability, one can directly extract low resolution video from the pre-encoded bitstream of a fully scalable video coder, such as MC-EZBC.[13] The low resolution video is actually the LL subband after one or two steps of spatial decomposition. However, as seen in Fig. 1, the CDF 9-tap filter is not an ideal choice for lowpass filtering, compared to the MPEG4 13-tap low pass filter.[2] If the original frames have significant high frequency components, such as in the MPEG test clip *City*, the subband low resolution frame will have significant spatial aliasing, as seen on left side of Fig. 2. This aliasing is more visually annoying in video, due to the fact that it moves around or "phases" with moving of the objects containing the high frequency edges. Previous work[4,5] tried to reduce the spatial aliasing using a three-lifting-step filter. Compared to the CDF 9-tap lowpass filter, the lowpass filter in the three-lifting-step filters is better for lowpass filtering. Hence the low resolution video has less spatial aliasing as shown in.[5] However, this advantage is achieved at the cost of a quality loss for full and some lower resolution video.

In this paper, we propose a content adaptive method for aliasing reduction without introducing a significant or any loss for full and lower resolution video.[14] We know that even if the original frames have significant high frequency energy, the low resolution frames from MPEG4 lowpass filtering (MPEG4 low resolution frame) will

John W.Woods.: E-mail: woods@ecse.rpi.edu, Telephone: 1 518 276 6079

Yongjun Wu: E-mail: wuy2@cipr.rpi.edu, Telephone: 1 425 705 7611

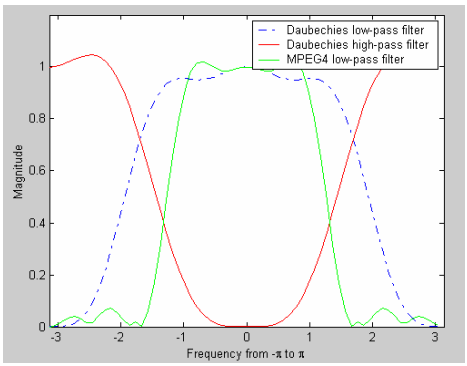

Figure 1. Comparison of frequency responses of MPEG4 and CDF (Daubechies) 9/7 lowpass filter. Also shown is the CDF 9/7 highpass filter.

not have significant aliasing, as seen on right side of Fig. 2. In Section 2 we analyze both the subband and MPEG4 low resolution frames via subband/wavelet decomposition. We model the coefficients in small subbands of the two low resolution frames as random variables, and after scaling the variance/energy in each small subband to match that of the MPEG4 low resolution frame, we can make the two low resolution frames visually similar. In Section 3.1, we explain how to implement frequency roll-off in an embedded image coder, i.e. achieve the scaling for high frequency components through sub-bitplane shift in the extractor with decoder-side weighting, with two different energy matching structures. In Section 3.2, an analysis is presented of the relationship between sub-bitplane shift and scaling. In Section 4, experimental results with frequency roll-off for still images are given. In Section 5 we extend the idea to scalable video coding. Conclusions are given in Section 6.

Figure 2. Left: LL suband of Y component after one step spatial decomposition from the 19th frame of *City*, i.e. subband CIF frame, right: Low resolution image of Y component after MPEG4 lowpass filtering from the 19th frame of *City*, i.e. MPEG4 CIF frame.

2. ENERGY ANALYSIS

We assume that the full resolution frame data is 4CIF in size. As shown in Fig. 2, the MPEG4 CIF frame is smoother than the subband CIF frame. At the same time, the subband CIF frame has significant spatial aliasing, resulting from the CDF 9-tap lowpass filtering and subsampling. The question we address is: how to make subband CIF frames similar to MPEG4 CIF frames? We propose the following energy matching procedure to this end:

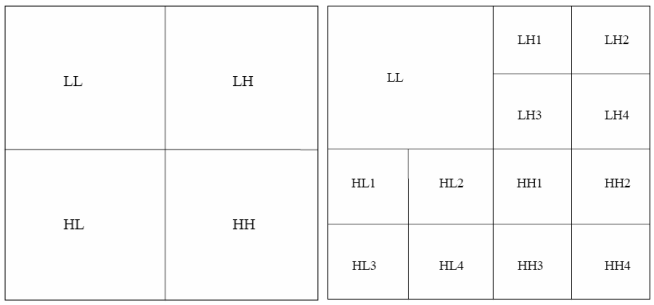

Figure 3. Left, *structure one*. Right, *structure two*.

1. Decompose the MPEG4 and subband CIF frames further into small subbands by subband/wavelet filters with decomposition structures shown in Fig. 3.
2. Estimate the variance in each small subband for the MPEG4 and subband CIF frames, and call them σ_m^2 and σ_s^2, respectively .
3. Compute a scaling factor $\alpha \triangleq \sqrt{\frac{\sigma_s^2}{\sigma_m^2}}$ for each small subband.
4. Scale down the coefficients by α in each small subband in the subband CIF frame, and reconstruct the subband CIF frame after scaling.

In step 1, if we assume the filters used in the decomposition are sufficiently ideal, then we can step-wise filter by scaling these coefficients, which we model as stationary random data,

$$x \sim N(\mu_x, \sigma_x^2), \mu_x = 0 \text{ (except } \mu_{LL} \neq 0 \text{ in LL subband)} \tag{1}$$

In step 2, we then estimate the variance and mean for each subband by,

$$\tilde{\sigma}_x^2 = \frac{1}{MN} \sum_{m=1}^{M} \sum_{n=1}^{N} c(m,n)^2, \text{ except for LL subband} \tag{2}$$

$$\tilde{\mu}_{LL} = \frac{1}{MN} \sum_{m=1}^{M} \sum_{n=1}^{N} c(m,n), \;\; \tilde{\sigma}_{LL}^2 = \frac{1}{MN} \sum_{m=1}^{M} \sum_{n=1}^{N} (c(m,n) - \mu_{LL})^2, \tag{3}$$

where the subbands are $M \times N$ in size. Hence, if we want to make the subband CIF frame similar to the MPEG4 CIF frame, the only parameter we need to match is $\tilde{\sigma}_x^2$ between the small subbands. Steps 3 and 4 accomplish the function.

In our implementation, the filters used for structure one or two decomposition are CDF 9/7 filters, which of course are not ideal bandpass filters. However in order to be compatible with the spatial decomposition in a subband/wavelet coder, such as EZBC, we adopt them for energy matching. Moreover, the assumption that the coefficients in each small subband are stationary and Gaussian only holds approximately. The smaller the subband size is, the better the assumption. For this and other reasons, energy matching works better with structure two decomposition. When structure one is employed in the following section, the set of scaling factors are not exactly the same as the results from energy comparison due to the above reasons. The energy comparison for structure one only serves as a reference.

Table 1 shows the estimated variance in each small subband with structure one. Note the energies from subband decomposition have been scaled down properly by the filter gains. We can see there are significant differences of energies in the high frequency subbands, which result in the aliasing in the subband CIF frame.

Image	$\tilde{\sigma}^2_{LL}$	$\tilde{\sigma}^2_{HL}$	$\tilde{\sigma}^2_{LH}$	$\tilde{\sigma}^2_{HH}$
(1)	2282.0	366.6	201.2	48.95
(2)	2316.6	114.7	115.7	14.36

Table 1. Energy comparison in each subband of subband CIF frame (1) and MPEG4 CIF frame (2) from 19th frame of *City* with structure one. (The $\tilde{\mu}_{LL}$ are both 229.34 and 229.34 for (1) and (2).)

With structure one, in our first attempt, we select the scaling coefficients roughly as $\alpha_{HL} = \frac{114}{366}$, $\alpha_{LH} = \frac{115}{201}$ and $\alpha_{HH} = \frac{14}{48}$. On left side of Fig. 4, we show the reconstructed image after this scaling. Compared to left figure in Fig. 2, left figure in Fig. 4 is visually more pleasing.

In order to better match the energy for the high frequency components, we can decompose the MPEG4 CIF frame and subband CIF frames by structure two. From Table 2, we can see that after using a simple quantized scaling factor $\bar{\alpha}$, we can match the energy in the subband CIF frame to that in the MPEG4 CIF frame very well. The right side of Fig. 4 shows the reconstructed subband CIF frame after scaling by the set of factors $\bar{\alpha}$, which is visually better than left side of Fig. 4.

Image	(1)	(2)	α	$\bar{\alpha}$
$\tilde{\sigma}^2_{LL}$	2282.0	2316.6	0.993	1.0
$\tilde{\sigma}^2_{HL1}$	841.18	86.75	3.11	$2\sqrt{2}$
$\tilde{\sigma}^2_{HL2}$	149.10	29.72	2.24	2.0
$\tilde{\sigma}^2_{HL3}$	297.24	260.71	1.07	1.0
$\tilde{\sigma}^2_{HL4}$	91.07	86.93	1.02	1.0
$\tilde{\sigma}^2_{LH1}$	296.15	68.44	2.08	2.0
$\tilde{\sigma}^2_{LH2}$	356.63	312.89	1.07	1.0
$\tilde{\sigma}^2_{LH3}$	66.24	16.86	1.98	2.0
$\tilde{\sigma}^2_{LH4}$	78.18	76.95	1.0	1.0
$\tilde{\sigma}^2_{HH1}$	40.82	2.19	4.31	4.0
$\tilde{\sigma}^2_{HH2}$	74.44	14.19	2.29	2.0
$\tilde{\sigma}^2_{HH3}$	36.07	7.43	2.20	2.0
$\tilde{\sigma}^2_{HH4}$	49.43	38.49	1.13	1.0

Table 2. Energy comparison in each small subband of structure two for subband CIF frame and MPEG4 CIF frame from the 19th frame of *City*. Here $\alpha = \sqrt{\frac{\tilde{\sigma}^2_{(1)}}{\tilde{\sigma}^2_{(2)}}}$ where $\tilde{\sigma}^2_{(1)}$ is the variance in column (1) and $\tilde{\sigma}^2_{(2)}$ is the variance in column (2), and $\bar{\alpha}$ is the quantized version of α, which is used finally in the EZBC coder.

The same analysis can be repeated for the MPEG and subband QCIF frames, i.e. the low resolution frames after two levels of MPEG4 lowpass filtering and two levels of CDF 9/7 subband decomposition.

3. FREQUENCY ROLL-OFF FOR EMBEDDED IMAGE CODING

From Section 2, we know that after scaling down the energy in the high frequency components, the subband CIF frame can achieve similar visual quality to the MPEG4 CIF frame. In this section, we apply frequency roll-off to embedded image coding. The question is: How to integrate the scaling factors, i.e. frequency roll-off,[9] into the embedded EZBC image coder? There are two apparent solutions. The first is to scale the high frequency components in a pre-processing stage. However, since an embedded image coder is designed for both SNR and resolution scalability, we must compensate the energy scaled down in the high frequency components when the image is decoded at full resolution. This may compromise the embedded property of our image coder. The second solution is to achieve frequency roll-off by sub-bitplane shift in the extractor and scaling backup at the

Figure 4. Left: reconstructed low resolution frame after scaling HL, LH and HH bands of Subband low resolution frame with structure one, right: reconstructed low resolution frame after scaling the 12 small bands of Subband low resolution frame using the set of $\bar{\alpha}$ in Table 2. Both figures are Y component only.

decoder. Here we first present the specific modifications in the EZBC coder[8] for frequency roll-off in Section 3.1. Section 3.2 provides some analysis of the relationship between sub-bitplane shift and scaling.

3.1. Modifications for the EZBC coder

From the EZBC coding algorithm,[7] we know that the lists and quadtrees are maintained separately for the individual subbands and quadtree levels. Moreover, for a bitplane in subband k, there are $\mathbf{QD}_k + 2$ sub-passes/sub-bitplanes, where $\mathbf{QD}_k$ is the depth of the quadtree in subband k. As shown in Section 2 for structure one dyadic decomposition, we want to scale down the HL, LH, and HH subbands of the subband CIF frame by different scaling factors. We need to maintain the arithmetic coding procedure for these three bands separately instead of integrating them together as in the current EZBC scheme. Similar separate maintenance should be done for structure two. For structure two, since it's a non-dyadic decomposition structure, currently we do not utilize the interband context modeling technique but only intraband, which fortunately, can achieve most of the coding gain. Moreover, if there are significant high frequency components, a non-dyadic decomposition is also more efficient for residual frame coding in video, compared to dyadic decomposition as recently shown for example, in.[3] Hence, compared to structure one, dyadic decomposition, there is only a slight PSNR loss or even a PSNR gain for the scheme two, non-dyadic decomposition.

In the EZBC pre-encoder, we encode a single frame as before. When we want a full resolution frame at some bitrate, the extractor works in the same way as before,[6] i.e. the subband bitplane interleaving scheme for bit allocation. However, when we want a low resolution frame, we need to scale down the high frequency components of the subband CIF frame by the corresponding factors α. Because we progressively code the image bitplane-by-bitplane, scaling down by a factor of 2 in subband k is equal to a downward shift of one bitplane for subband k. Moreover, since there are $\mathbf{QD}_k + 2$ sub-passes/sub-bitplane for coding a bitplane in subband k, we could assume that $\lfloor \frac{\mathbf{QD}_k+2}{2} \rfloor + 1$ sub-bitplane corresponds to a *fictitious* half-bitplane. Then if we wanted to scale subband k down by $2\sqrt{2}$, we could simply shift the bitstream for subband k downward by $\mathbf{QD}_k + 2 + \lfloor \frac{\mathbf{QD}_k+2}{2} \rfloor$ sub-bitplanes with $\mathbf{QD}_k + 2$ odd and $\mathbf{QD}_k + 2 + \lfloor \frac{\mathbf{QD}_k+2}{2} \rfloor - 1$ with $\mathbf{QD}_k + 2$ even. Then we allocate the bits for each subband using the current subband bitplane interleaving scheme.[6] In the EZBC decoder, we would need to divide the decoded coefficients in subband k by a factor of $2\sqrt{2}$, before reconstructing the subband CIF resolution frame.

3.2. Distortion analysis for bitplane/sub-bitplane coding in EZBC

After the subband/wavelet decomposition, the distortion from the various subbands is almost additive thanks to the near orthogonal property of CDF 9/7 filters. So to a good approximation, we have $D = \sum_k D_k$, where

$D_k = E[(\hat{s}(m,n) - s(m,n))^2]$, the mean-square distortion, where $\hat{s}(m,n)$ and $s(m,n)$ are the decoded and original subband coefficients in subband k. Also, the subband coefficient $s(m,n)$ can be written as

$$s(m,n) = \sum_i b_i(m,n) 2^i. \tag{4}$$

Hence, for bitplane coding, when we decode subband coefficient $s(m,n)$ up to bitplane B, we reconstruct the subband coefficient $\hat{s}(m,n)$ as,

$$\hat{s}(m,n) \quad = \quad \sum_{i=B}^{M} b_i(m,n) 2^i + \frac{1}{2} \cdot 2^B, \tag{5}$$

where M is most significant bitplane in subband k. The reconstructed coefficient $\hat{s}(m,n)$ in (5) is an unbiased approximation to the true coefficient $s(m,n)$, corresponding to the assumption that the possible values for $s(m,n)$ can be expressed as,

$$s(m,n) \quad = \quad \sum_{i=B}^{M} b_i(m,n) 2^i + \varepsilon_B \cdot 2^B, \tag{6}$$

where ε_B is a random variable, which has a symmetric distribution centered at $\frac{1}{2}$. As a special case, we assume it is the uniform distribution $\mathbf{U}(0,1)$. Then, when we decode all the coefficients in subband k up to bitplane B, the distortion D_{kB} can be expressed as,

$$D_{kB} = E[(\varepsilon_B - \frac{1}{2})^2] 2^{2B}. \tag{7}$$

If different ε_B, where $B \in \{0, \cdots, M-1\}$, have different distributions, we have

$$\begin{aligned} D_{kB} \quad &= \quad 4 \frac{E[(\varepsilon_B - \frac{1}{2})^2]}{E[(\varepsilon_{B-1} - \frac{1}{2})^2]} D_{k(B-1)} = 4\gamma D_{k(B-1)} = \gamma' D_{k(B-1)}, \\ \text{where } \gamma \quad &= \quad \frac{E[(\varepsilon_B - \frac{1}{2})^2]}{E[(\varepsilon_{B-1} - \frac{1}{2})^2]}, \gamma' = 4\gamma. \end{aligned} \tag{8}$$

Fictitiously if there existed a half bitplane between bitplanes B and $B-1$, i.e. $B-0.5$, and we could write,

$$s(m,n) \quad = \quad \sum_{i=B}^{M} b_i(m,n) 2^i + b_{B-0.5} \cdot 2^{B-0.5} + \varepsilon_{B-0.5} \cdot 2^{B-0.5}, \text{ and} \tag{9}$$

$$\hat{s}(m,n) \quad = \quad \sum_{i=B}^{M} b_i(m,n) 2^i + b_{B-0.5} \cdot 2^{B-0.5} + \frac{1}{2} \cdot 2^{B-0.5}, \tag{10}$$

then we would have,

$$D_{kB} \quad = \quad 2 \frac{E[(\varepsilon_B - \frac{1}{2})^2]}{E[(\varepsilon_{B-0.5} - \frac{1}{2})^2]} D_{k(B-0.5)} = 2\alpha_1 D_{k(B-0.5)} = \alpha_1' D_{k(B-0.5)}, \tag{11}$$

$$D_{k(B-0.5)} \quad = \quad 2 \frac{E[(\varepsilon_{B-0.5} - \frac{1}{2})^2]}{E[(\varepsilon_{B-1} - \frac{1}{2})^2]} D_{k(B-1)} = 2\alpha_2 D_{k(B-1)} = \alpha_2' D_{k(B-1)}, \tag{12}$$

where $\alpha_1 = \frac{E[(\varepsilon_B - \frac{1}{2})^2]}{E[(\varepsilon_{B-0.5} - \frac{1}{2})^2]}$, $\alpha_2 = \frac{E[(\varepsilon_{B-0.5} - \frac{1}{2})^2]}{E[(\varepsilon_{B-1} - \frac{1}{2})^2]}$, $\alpha_1' = 2\alpha_1$, and $\alpha_2' = 2\alpha_2$. We can assume $\alpha_1 \approx \alpha_2$, since the three random variables $\varepsilon_B, \varepsilon_{B-0.5}$ and ε_{B-1} are numerically close to each other and we already have $\alpha_1 \cdot \alpha_2 = \gamma$, hence we get

$$\alpha_1' \approx \alpha_2' \approx \sqrt{\gamma'}. \tag{13}$$

If all the ε_B, where $B \in \{0, \cdots, M-1\}$, obey the same distribution, we have

$$D_{kB} = 4D_{k(B-1)}, \text{ and } D_{kB} = 2D_{k(B-0.5)}. \tag{14}$$

As a special case, when ε_B has a uniform distribution $\mathbf{U}(0,1)$, $D_{kB} = \frac{1}{12}2^{2B}$.

In real bitplane coders, such as EZBC, SPIHT and EBCOT, the sub-bitplane coding technique is introduced as several coding passes for each bitplane. For example, EBCOT has 4 sub-passes/sub-bitplanes in coding each bitplane, and EZBC has $\mathbf{QD}_k + 2$ sub-passes/sub-bitplanes in coding each bitplane, where $\mathbf{QD}_k$ is the depth of quadtree in subband k. Specifically in the EZBC coder, the $\mathbf{QD}_k + 2$ sub-passes are done in a predefined order: the first pass is for the refinement of List of Insignificant Pixels (LIP), the second to $(\mathbf{QD}_k + 1)$th passes are for the refinement of the nodes in the quadtree of subband k, starting from leaves of the quadtree to the root of the quadtree, and the $(\mathbf{QD}_k + 2)$th pass is for the refinement of List of Significant Pixels (LSP). In that order the refined bits reduce the MSE in subband k in a decreasing order. If we can find some sub-pass p in the $\mathbf{QD}_k + 2$ sub-passes which makes distortion reduction satisfy (8), (11), (12) and (13), then we can define sub-pass p as the *fictitious* half-bitplane $B - 0.5$ between bitplane B and $B - 1$. From (8), (11), (12) and (13), we know that,

$$\Delta D_{B-0.5} = D_{B-0.5} - D_{B-1} \approx (\sqrt{\gamma'} - 1)D_{B-1}, \tag{15}$$

$$\Delta D_B = D_B - D_{B-0.5} \approx (\gamma' - \sqrt{\gamma'})D_{B-1}. \tag{16}$$

Ideally $\frac{\Delta D_B}{\Delta D_{B-0.5}} \approx 2$ according to (14). Empirically we choose sub-pass $\lfloor \frac{\mathbf{QD}_k+2}{2} \rfloor + 1$ as the half-bitplane $B - 0.5$, since we know the distortion reduction in each sub-pass is in a decreasing order.

In Tables 3–4, **B** is the bitplane number up to which the bits are discarded, and **MSE** is the resulting mean squared error. α' is the ratio for MSE's between adjacent half-bitplanes, and γ' is the ratio for MSE's between adjacent integer bitplanes. The experimental results in those tables show that our selection of sub-pass $\lfloor \frac{\mathbf{QD}_k+2}{2} \rfloor + 1$ as the half-bitplane $B - 0.5$ is a good approximation up to bitplane 4 in subband HL and LH of the second level subband decomposition for the 19th frame in *City*, and similar results are obtained for HH subband of the second level subband decomposition in the 19th frame of *City* and those subbands in the 12th frame of *Harbour*.[12] That means that at high to medium rates our selection is good. However, from the results in the tables, we can see that at low bitrates, i.e. beyond bitplane 4, either our analysis is not valid or our selection of sub-pass for the half-bitplane is not good. Since the analysis only involves the sub-bitplane coding in EZBC, it's the same for both image and video coding. Hence the analysis here is also valid for video coding.

B	1	1.5	2	2.5	3	3.5	4	4.5	5	5.5	6
MSE	0.45	1.2	2.2	4.6	10.2	16.6	44.9	57.8	163.9	178.2	458.1
α'	–	2.7	1.8	2.1	2.2	1.6	2.7	1.3	2.8	1.1	2.6
γ'	–	–	4.8	–	4.7	–	4.4	–	3.7	–	2.8
$\sqrt{\gamma'}$	–	–	2.2	–	2.2	–	2.1	–	1.9	–	1.7

Table 3. MSEs of Y component in LH subband of the second level subband decomposition for the 19th frame of *City*, after we discard the bits for some low bitpanes. The total variance in the subband is 804.8.

4. EXPERIMENTAL RESULTS FOR STILL IMAGES

We apply the schemes proposed in Section 3 for image coding. The scaling coefficients used for structure one, dyadic decomposition of subband CIF frame are, $\alpha_{LH} = \frac{1}{\sqrt{2}}$, $\alpha_{HL} = \frac{1}{2\sqrt{2}}$, $\alpha_{HH} = \frac{1}{2\sqrt{2}}$,, which corresponds to shifting LH, HL and HH subbands by 0.5, 1.5 and 1.5 bitplanes, i.e. $\lfloor \frac{\mathbf{QD}_k+2}{2} \rfloor$ and $\mathbf{QD}_k + 2 + \lfloor \frac{\mathbf{QD}_k+2}{2} \rfloor$ and $\mathbf{QD}_k + 2 + \lfloor \frac{\mathbf{QD}_k+2}{2} \rfloor$ sub-bitplanes. The scaling factors used for structure two of the subband CIF frame are $\bar{\alpha}$'s in Table 2. Fig.5 and Fig. 6 show the visual comparison for CIF resolution image of the 19th frame of *City*.

B	1	1.5	2	2.5	3	3.5	4	4.5	5	5.5	6
MSE	0.5	1.2	2.35	4.46	10.16	15.9	38.8	51.5	135.0	155.9	380.1
α'	–	2.35	2.0	1.9	2.3	1.56	2.45	1.3	2.6	1.2	2.4
γ'	–	–	4.6	–	3.7	–	3.56	–	3.3	–	3.0
$\sqrt{\gamma'}$	–	–	2.1	–	1.93	–	1.9	–	1.8	–	1.7

Table 4. MSE's of Y component in HL subband of the second level subband decomposition for the 19th frame of *City*, after we discard the bits from some low bitpanes. The total variance in the subband is 1466.6.

Figure 5. CIF resolution image of 19th frame of *City* decoded at 0.986 bpp without frequency roll-off.

Since we shift the bitstreams downward for HL, LH and HH subbands of the subband CIF frame in structure one, and those for HL1,..., HL4, LH1, ..., LH4 and HH1, ..., HH4 in structure two, the bits allocated for those subbands are significantly less, compared to the coder without frequency roll-off. Those extra bits are allocated for other subbands. Table 5 shows the effect of sub-bitplane shift in extractor on the bit allocation for each subband in structure one. We observed similar results for structure two.[12]

Coder	**level 1**	**level 2**	**level 3**	**level 4**	**level 5**	**LH**	**HL**	**HH**
(1)	96	311	803	2105	5212	2623	1093	149
(2)	95	291	670	1576	3379	2721	2795	865

Table 5. Number of bytes allocated for different subbands at bitrate 0.986 bpp. Coder (1) has structure one frequency roll-off and coder (2) has no frequency roll-off. In each level there are three subbands HL, LH, and HH, except level 1 with only the LL subband. LH, HL, and HH subbands are at level 6.

With structure one, at full resolution there is no quality loss compared to the original EZBC coder. For structure two, according to our experiments, there is also no PSNR loss at full resolution as shown in Table 6.

For the subband QCIF frame we apply another set of scaling factors with structure one as follows, $\alpha_{LH} = \frac{1}{\sqrt{2}}$, $\alpha_{HL} = \frac{1}{2}$, $\alpha_{HH} = \frac{1}{2}$, which are approximately obtained from the analysis of energy matching. For the subband QCIF frame, we also observed significant visual improvement. Since the subband QCIF frame is already quite small, it's not necessary to perform structure-two frequency roll-off. We also tested frequency roll-off for some CIF image from *Football*, with the scaling factors derived from energy matching. Significant visual improvement was also observed, where the unnecessarily sharp diagonal textures become smoother.

Coder	Rate(bpp)	***Y***	***U***	***V***
(1)	0.4110	30.89	40.07	42.73
(2)	0.4110	30.80	40.08	42.73
(1)	0.3288	29.88	39.86	42.55
(2)	0.3288	29.83	39.87	42.55
(1)	0.2055	27.97	39.23	41.77
(2)	0.2055	27.99	39.21	41.76
(1)	0.1233	26.22	38.78	41.18
(2)	0.1233	26.30	38.78	41.18

Table 6. PSNR at full resolution and different bitrate for 19th frame of *City*. Coder (1) has structure-one frequency roll-off, while coder (2) has structure-two frequency roll-off.

Figure 6. Left: CIF resolution image of 19th frame of *City* decoded at 0.986 bpp with roll-off structure one. Right: CIF resolution image of 19th frame of *City* decoded at 0.986 bpp with roll-off structure two.

5. EXTENSION TO SCALABLE MCTF-BASED VIDEO CODING

Frequency roll-off can be easily extended to 3-D scalable video coding. We obtain the scaling factors through energy matching between the average energies in each subband for structure one and structure two across the whole sequence. We find that the scaling factors for averaged energy matching in the whole sequence are almost the same as those from typical frames in Table 2 and 1.[12] This is not surprising, since the statistical information for a sequence tends to be stationary if there is no scene change.

The goal of frequency roll-off in scalable video coding is to make the low resolution video more visually pleasing and more alias-free. Now the low resolution video is created by the motion compensated temporal filter (MCTF), and extending the idea of frequency roll-off to the temporal dimension, we would logically roll-off the high temporal coefficients too. However, here we have the more modest goal of just performing roll-off on spatial frequency. Still, the coder works with the MCTF output, and so we spatially roll-off the several high temporal frames H_ts and single low temporal frame L_t output by the MCTF for each GOP. Thus the question arises of how the reconstructed low resolution frames with the roll-off applied to the MCTF output, i.e. $\mathcal{R}(\mathcal{D}(L_t))$ and $\mathcal{R}(\mathcal{D}(H_t))$, where $\mathcal{R}$ is the roll-off operator and $\mathcal{D}$ is the subband/wavelet analysis operator, relate to the low resolution frames with roll-off performed directly on the original input frames, i.e. on $\mathcal{R}(\mathcal{D}(f_n))$.

If we use one-level of MCTF with lifted Haar filters as an example, we obtain

$$H_t = \frac{1}{\sqrt{2}}(f_1 - \mathcal{M}(f_2)) \quad (17)$$

$$L_t = \mathcal{M}^{-1}(H_t) + \sqrt{2}f_2, \quad (18)$$

where $\mathcal{M}$ and $\mathcal{M}^{-1}$ are motion operator and inverse motion operator, and high temporal frame H_t and low temporal frame L_t are temporally aligned with input frames f_1 and f_2, respectively. Then we apply frequency roll-off after the subband/wavelet spatial decomposition of high and low temporal frames to obtain,

$$\mathcal{R}(\mathcal{D}(H_t)) = \frac{1}{\sqrt{2}}(\mathcal{R}(\mathcal{D}(f_1)) - \mathcal{R}(\mathcal{D}(\mathcal{M}(f_2)))) \quad (19)$$

$$\mathcal{R}(\mathcal{D}(L_t)) = \mathcal{R}(\mathcal{D}(\mathcal{M}^{-1}(H_t))) + \sqrt{2}\mathcal{R}(\mathcal{D}(f_2)). \quad (20)$$

We can approximately regard frequency roll-off operator $\mathcal{R}$ as a linear operator, since $\mathcal{R}$ is simply downward shift of the bitstreams for high frequency subbands, allocating less bits for high frequency subbands and scaling down the reconstructed subband coefficients.

In a practical scalable video coder, such as MC-EZBC, we reconstruct to obtain,

$$f_{2LL} = \frac{1}{\sqrt{2}}(\mathcal{R}(\mathcal{D}(L_t)) - \mathcal{M}^{-1}(\mathcal{R}(\mathcal{D}(H_t)))), \quad \text{and} \quad f_{1LL} = \sqrt{2}\mathcal{R}(\mathcal{D}(H_t)) + \mathcal{M}(\mathcal{R}(\mathcal{D}(f_2))),$$

as the output low resolution video.

The results with spatial frequency roll-off directly on the low resolution input frames $\mathcal{D}(f_1)$ and $\mathcal{D}(f_2)$ should be,

$$\mathcal{R}(\mathcal{D}(f_2)) = \frac{1}{\sqrt{2}}(\mathcal{R}(\mathcal{D}(L_t)) - \mathcal{R}(\mathcal{D}(\mathcal{M}^{-1}(H_t)))), \quad \text{and} \quad \mathcal{R}(\mathcal{D}(f_1)) = \sqrt{2}\mathcal{R}(\mathcal{D}(H_t)) + \mathcal{R}(\mathcal{D}(\mathcal{M}(f_2))),$$

and observation of these two sets of equations reveals that we would need the motion operators $\mathcal{M}$ and $\mathcal{M}^{-1}$ to commute with $\mathcal{RD}$ for these two sets of equation to be equivalent. Given a good motion field, with low number of unconnected pixels, this is approximately the case. Still, with our method, i.e. frequency roll-off on high and low temporal frames output from MCTF, we get the advantages of the MCTF defined video, such as reduced sensor noise and less temporal aliasing.

We tested the idea of frequency roll-off with two 4CIF sequences, *City* and *Harbour*, and three CIF sequences, *Mobile*, *Football*, and *Bus*, with scaling factors obtained from energy matching, as in Section 2.[12] While there are similar improvements in PSNR, the visual improvement is not nearly as clear as with *City*. For these other tested clips, the PSNR improvement is thought largely due to easier to code references (similarly MCTF processed but uncoded frames). Here Tables 7 and 8 list the average PSNR results at full and low resolution with and without frequency roll-off. In the tables, coder (0) is without frequency roll-off, coder (1) uses structure-one frequency roll-off, and coder (2) uses structure-two. The PSNRs are calculated with respect to their own decoder side reference frames. Please refer to[12] for the PSNR results of other sequences, and the website: ftp://ftp.cipr.rpi.edu/personal/wuy2/frequency-roll-off-MPEG-July/ for the energy matching results, scaling factors, and visual comparisons of all 5 sequences.

6. CONCLUSION

In this paper, we present a content adaptive scheme for aliasing reduction in scalable video coding without introducing any significant or no loss to full resolution video. The method is useful for the generation of low resolution video from originals. We compared the energy distribution in the subband/wavelet CIF/QCIF and MPEG4 CIF/QCIF frames, and found scaling factors to match the energy in the high spatial frequency components. We introduced two structures to match the energy in the MPEG4 CIF/QCIF and subband/wavelet CIF/QCIF frames. Structure one is compatible with dyadic decomposition and has low complexity, but has the

Coder	Testing points	***Y***	***U***	***V***	**Gain (*Y*)**
(0)	64Kbps_15fps_QCIF	21.23	27.08	25.86	
(1)	64Kbps_15fps_QCIF	23.33	26.75	25.71	+2.10
(0)	128Kbps_15fps_QCIF	26.90	30.25	29.47	
(1)	128Kbps_15fps_QCIF	28.88	30.09	29.38	+1.98
(0)	256Kbps_15fps_QCIF	33.26	37.87	37.08	
(1)	256Kbps_15fps_QCIF	35.71	37.13	36.39	+2.45
(0)	512Kbps_30fps_CIF	29.12	34.24	33.79	
(1)	512Kbps_30fps_CIF	29.12	34.24	33.79	+0.00
(0)	1024Kbps_30fps_CIF	32.94	38.58	38.00	
(1)	1024Kbps_30fps_CIF	32.94	38.58	38.00	+0.00

Table 7. PSNRs at various testing points for full sequence of *Mobile Calendar*.

Coder	Testing points	***Y***	***U***	***V***	**Gain (*Y*)**
(0)	64Kbps_15fps_QCIF	27.80	38.32	39.08	
(1)	64Kbps_15fps_QCIF	30.48	38.32	39.08	+2.68
(2)	64Kbps_15fps_QCIF	29.43	38.34	39.11	+1.63
(0)	128Kbps_15fps_QCIF	34.31	42.61	43.77	
(1)	128Kbps_15fps_QCIF	38.16	43.31	44.52	+3.85
(2)	128Kbps_15fps_QCIF	36.86	43.14	44.31	+2.55
(0)	256Kbps_30fps_CIF	30.67	41.23	42.81	
(1)	256Kbps_30fps_CIF	33.26	41.23	42.81	+2.59
(2)	256Kbps_30fps_CIF	31.65	41.24	42.87	+0.98
(0)	512Kbps_30fps_CIF	35.50	43.59	45.21	
(1)	512Kbps_30fps_CIF	38.76	44.46	46.24	+3.26
(2)	512Kbps_30fps_CIF	36.89	43.80	45.58	+1.39
(0)	2048Kbps_60fps_4CIF	35.75	44.96	46.86	
(1)	2048Kbps_60fps_4CIF	35.75	44.96	46.86	+0.00
(2)	2048Kbps_60fps_4CIF	35.55	43.40	46.14	−0.20

Table 8. PSNRs at various testing points for full sequence of *City*.

lower accuracy for energy matching, sometimes resulting in too soft a reference. Structure two uses a non-dyadic subband/wavelet decomposition, and currently does not utilize traditional interband context modeling. However, this structure two achieves energy matching more accurately and better visual quality. Moreover, as shown in [9], a non-dyadic decomposition may be better for frames with significant high spatial frequency.

While the goal here has been scalable subband/wavelet video coding, clearly the frequency roll-off method can be applied to scalable image coding. The frequency roll-off method can also been used in other embedded image coders,[12] such as SPIHT[10] and JPEG2000.[11] For different embedded coders, all the steps will be the same except the step for sub-bitplane shift, because different coders may employ different sub-bitplane coding techniques. Specifically the *fictitious half-bitplane* concept has to be extended to the other embedded coders.

REFERENCES

1. *Call for proposals on scalable video coding technology.* JTC1/SC29/WG11/N6193, Waikoloa, Hawaii, Dec. 2003.
2. *MPEG-4 Video Verification Model version 18.0.* ISO/IEC JTC1/SC29/WG11/N3908, Pisa, IT, January 2001.

3. G. C. K. Abhayaratne and E. Izquierdo. *Wavelets based residual frame coding in t+2D wavelet video coding.* ISO/IEC JTC1/SC29/WG11 MPEG2005/M11748, Hong Kong, China, January 2005.
4. V. Bottreau, C. Guillemot, R. Ansari, and E. Francois. *SVC technical contribution to CE1b: spatial transform using three lifting steps.* JTC1/SC29/WG11/MPEG2004/M10904, Redmond, WA, July. 2004.
5. V. Bottreau, C. Guillemot, R. Ansari, and E. Francois. *SVC CE5: spatial transform using three lifting steps filters.* JTC1/SC29/WG11/MPEG2004/M11328, Palma, Spain, Oct. 2004.
6. P. Chen. *Fully scalable subband/wavelet video coding.* PhD thesis, ECSE Dept. Rensselaer Polytechnic Institute, Troy, NY, May 2003.
7. S.-T. Hsiang. *Highly scalable subband/wavelet image and video coding.* PhD thesis, ECSE Dept. Rensselaer Polytechnic Institute, Troy, NY, May 2002.
8. S. T. Hsiang and J. W. Woods. Embedded image coding using zeroblocks of subbband/wavelet coefficients and context modeling. IEEE ISCS, pages 28–31, May 2000.
9. T. Naveen. *Subband compression of high definition video.* PhD thesis, ECSE Dept. Rensselaer Polytechnic Institute, Troy, NY, Jan. 1993.
10. A. Said and W. A. Pearlman. A new fast and efficient image codec based on set partitioning in hierarchical trees. *IEEE Trans. on Circuits and Systems for Video Technology*, 6:243–250, 1996.
11. D. S. Taubman and M. W. Marcellin. *JPEG 2000: Image Compression Fundamentals, Standards and Practice.* Dordrecht: Kluwer Academic Publishers, Norwell, MA, 2001.
12. Y. Wu. *Fully scalable subband/wavelet video coding system.* PhD thesis, ECSE Dept. Rensselaer Polytechnic Institute, Troy, NY, August 2005.
13. Y. Wu, A. Golwelkar, and J. W. Woods. *MC-EZBC video proposal from Rensselaer Polytechnic Institute.* JTC1/SC29/WG11/M10596/S15, Munich, Germany, Mar. 2004.
14. Y. Wu and J. W. Woods. *Aliasing reduction for scalable subband/wavelet video coding.* ISO/IEC/JTC1/SC29/WG11/M12376, Poznan, Poland, July 2005.