# Block-Classified Bidirectional Motion Compensation Scheme for Wavelet-Decomposed Digital Video

**Sohail Zafar**[*], Member, IEEE **Ya-Qin Zhang**[†], Senior Member, IEEE
**Bijan Jabbari**[‡], Senior Member, IEEE

**Abstract**

In this paper we introduce a block-classified bidirectional motion compensation scheme for our previously developed wavelet-based video codec [24,25], where multiresolution motion estimation is performed in the wavelet domain. The frame classification structure described in this paper is similar to that used in the MPEG standard. Specifically, the I-frames are intraframe coded, the P-frames are interpolated from a previous I- or a P-frame, and the B-frames are bidirectional interpolated frames. We apply this frame classification structure to the wavelet domain with variable block sizes and multiresolution representation. We use a symmetric bidirectional scheme for the B-frames and classify the motion blocks as intraframe, compensated either from the preceding or the following frame, or bidirectional (i.e., compensated based on which type yields the minimum energy).

We also introduce the concept of F-frames, which are analogous to P-frames but are predicted from the following frame only. This improves the overall quality of the reconstruction in a group of pictures (GOP) but at the expense of extra buffering. We also study the effect of quantization of the I-frames on the reconstruction of a GOP, and we provide intuitive explanation for the results.

In addition, we study a variety of wavelet filter-banks to be used in a multiresolution motion-compensated hierarchical video codec. The results

[*] Sohail Zafar is with the Mathematics and Computer Science Division at Argonne National Laboratory, Argonne, IL. (E-mail: zafar@mcs.anl.gov)

[†] Ya-Qin Zhang is with David Sarnoff Research Center, Princeton, NJ. (E-mail: zhang@earth.sarnoff.com)

[‡] Bijan Jabbari is with the ECE Department at George Mason University, Fairfax, VA. (E-mail: bjabbari@gmu.edu)

show that, in general, short-length filters produce better results than do longer lengths which perform very well for still images.

# I Introduction

Several recent results have indicated that subband/wavelet-based approaches have outperformed DCT-based techniques for still images. Examples such as vector subband coder (VSC) [9], embedded zero-tree wavelets (EZW) [15], and trellis-coded wavelet compression [16] have shown superior performance to the JPEG baseline coding standard. However, when applied to full-motion video sequence, wavelet-based techniques have not shown clear advantages compared with DCT-based techniques such as MPEG 2 [11]. The reason is that block-based motion searching and compensation match well to the block-based DCT structure, while for wavelet-based coding, block-based motion structure is in fundamental conflict with the global-based wavelet decomposition. In addition, it is difficult to employ the adaptive structure of intra/inter-compensation decision present at the macro-block level for wavelet because of its global decomposition.

Several attempts have been made to solve this problem. In [24, 25], and [23] we introduced a video codec based on multiresolution motion compensation (MRMC) for a hierarchically decomposed video using wavelet decomposition, where motion estimation is performed in the wavelet domain rather than in the original image domain. The four MRMC schemes we proposed can be used for any hierarchical representation and are not particularly limited to wavelet decomposition. Since our multiresolution motion estimation scheme utilizes the correlation of motion at different levels of the hierarchy, it gives better results in terms of overall complexity and performance than do traditional approaches. In addition, intra/inter-compensation decisions can be adaptively made according to local motion activities, since motion prediction is performed after the wavelet decomposition. Based on the concept of wavelet-domain multiresolution motion prediction, several coding schemes have been developed for different applications [3, 7, 8, 12, 13]. Other original work on multiresolution coding can be found in [2, 6, 10, 18].

In most motion prediction schemes, the motion vectors of a particular frame are *predicted* from the previous frame. In MPEG terminology these are referred to as P-frames. If the progressive transmission is compromised and frames are reordered, and if more buffers are added to hold a following and preceeding frame, prediction results can be improved. These types of frame are referred to as B-frames. This paper reports our study in applying the B-frame structure to wavelet-decomposed pictures. The results show significant improvement over previous approaches in terms of both peak signal-to-noise ratio (PSNR) and subjective observation for a

set of testing sequences.

Considerable research effort has been expended in the area of filter design for wavelets. Many researchers have introduced wavelets and scaling functions having different properties, such as orientation of the filters, FIR implementation, and energy distribution. We have considered a number of different wavelet filter-banks and have tested their performance in a multiresolution video coding environment.

Section II gives a brief introduction to hierarchical representation of video, multiresolution motion compensation, and bidirectional motion compensation. Interested readers are referred to [25] and [23] for more details. Section III outlines our classification algorithm, and Section IV presents the simulation results.

# II Wavelet-Domain Motion Estimation and Compensation

## A Wavelet Decomposition and Variable Block-sized Multiresolution Motion Estimation

In the multiresolution motion estimation (MRME) scheme [25], the motion vectors are first calculated for the lowest resolution subband on the top of the pyramid. Then for all the subimages in lower layers of the pyramid, they are refined by using the information obtained for higher layers.

A variable block-sized MRME scheme significantly reduces the searching and matching time and provides a smooth motion vector field. In our study, a video frame is decomposed up to three levels, resulting in a total of ten subbands, with three subbands at each of the first two levels and four in the top level, including the subband $S_8$ , which represents the lowest frequency band (Figure 1).

$S_8$ contains a major percentage of the total energy present in the original frame, although it is only 1/64 of the original frame in size. Variable-sized blocks exploit the fact that human vision can perceive errors more easily in lower frequencies than in higher bands and tends to be selective in spatial orientation and positioning; therefore, more emphasis is given to the lower frequency bands by making the blocks at higher resolutions bigger than those at lower resolutions. In addition, errors generated by motion estimation at the lowest-resolution subbands are propagated and expanded to all subsequent lower-layer subbands.

All ten subbands have a highly correlated motion activity. Using a variable-sized block of $p2^{M-m}$ by $q2^{M-m}$ for the m-th level ensures that the motion estimator tracks

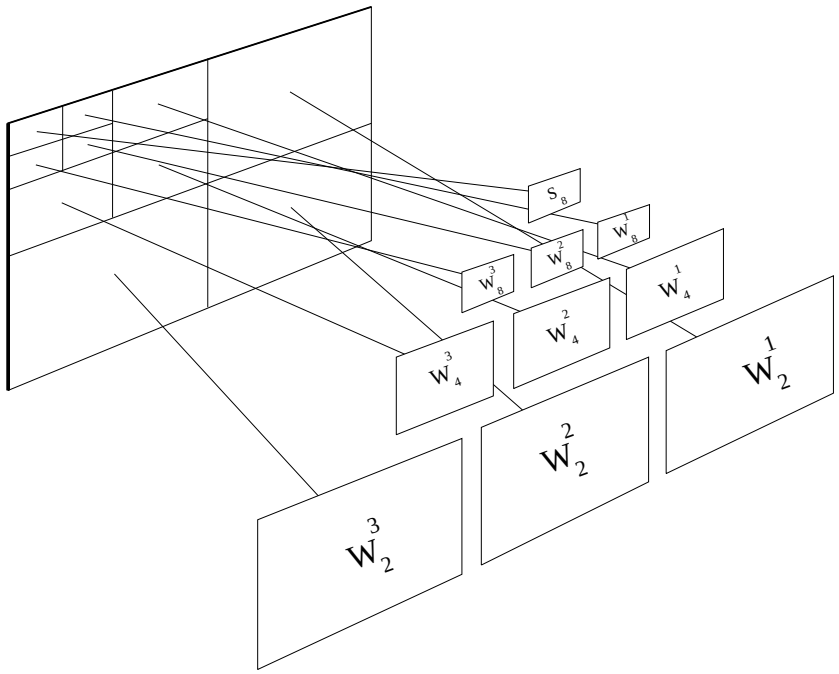



Figure 1: The Wavelet Decomposition and its Pyramid Structure

the same object/block regardless of the resolution or frequency band, where p by q is the blocksize of the highest layer at level M. With this structure, the number of motion blocks for all the subimages is constant, because a block at one resolution corresponds to the same position and the same object at all the other resolutions. In other words, all scaled subimages have the same number of motion blocks that characterize the global motion structure in different grids and frequency ranges. Variable-sized blocks thus tend to appropriately weigh the importance of different layers to match the human visual perception. This scheme can detect motions for small objects in the highest level of the pyramid. Constant block-size MRME approaches tend to ignore the motion activities for small objects in higher levels of the pyramid because a block size of $p \times q$ actually corresponds to $p2^{M-m} \times q2^{M-m}$ in the m-th layer.

Our variable-sized MRMC approach requires much fewer computations than does its fixed-size counterpart, since no interpolation is needed as the grid is refined [18]. However, in our variable block-sized MRME approach, an accurate characterization of motion information at the highest-layer subband produces very low energy in the displaced residual subbands and results in much "cleaner" copies of displaced residual wavelets (DRW) or subimages for lower-layer subbands. In contrast, interpolation is often required to obtain similar results when using schemes that have fixed-sized blocks at all resolutions.

## B Bidirectional Motion Search

In most motion prediction schemes, the motion vectors of a particular frame are *predicted* from the previous frame. This implementation requires only one frame buffer in the memory of the receiver. A frame can be reconstructed immediately as it is received because the previous frame has already been constructed.

Prediction from the past frame can track a moving object very well, but the areas uncovered as a result of this movement have no correspondence to the previous frame. On the other hand, these freshly uncovered areas do have association with the frame following it. A similar situation arises when there is a change in scene. A frame in question does not have any relation to the past frame but is related to the future frame. In these cases, prediction from the past frame does not benefit from motion compensation. But if the future frame was somehow available, the current frame could be predicted from it. It is important to note here that a future frame should not be dependent on the current frame and be available beforehand. This situation in turn means that the frames are not transmitted in progressive order, and more buffering is required at the receiver to hold more than one frame buffer. This also induces a delay in reconstruction of a frame because the reconstruction process has to wait until all the depending frames have been reconstructed, which might have their own dependencies. In MPEG terminology, such frames are termed B-frames.

Figure 2 shows the motion search procedure for a B-frame as described by MPEG specifications. The motion vector $\mathbf{V}_i(x,y)$, for a block is given by

$$\mathbf{V}_i(x,y) = \arg\min_{x,y\in\Omega}\Big\{\frac{1}{XY}\sum_{p=-X/2}^{X/2}\sum_{q=-Y/2}^{Y/2}\big|I_i(x_m+p,y_n+q) - 1/2\big(I_{i-1}(x_m+p+x,y_n+q+y)+I_{i+1}(x_m+p-x,y_n+q-y)\big)\big|\Big\}. \quad (1)$$

Equation (1) is the same as that for minimum absolute difference (MAD) except that the second term is an average of the pixel in the previous and the future frames. Note that the sign of the motion vectors (x,y) is opposite for the future frame term. This implies that if the block has moved a distance $\mathbf{V}_i(x,y)$ between the previous and current frame, it is expected to move the same distance between the current and the following frame. This symmetric search [18] has an advantage that only one set of motion vectors is needed for each block. The disadvantage is that if frames are not symmetrically located in time, the search performs rather poorly.

## C Multiresolution Motion-Compensation Algorithms

Motion compensation applied to decomposed wavelets has led to better results than applying wavelet decomposition to the frame difference after motion compensation at the original input scale [25]. Some novel techniques for hierarchical motion estimation were proposed in [23] and are reiterated here for reference.

The original video source is first decomposed into a set of ten wavelets, or subbands, $\{S_M, W_m^k(x)\ ;\ m = 1, \ldots, M\ ;\ k = 1, 2, 3\}$ with $M = 3$, by going to three levels of decomposition on the lowpass sub-image. After using variable block-sized multiresolution motion estimation and compensation, the set of displaced residual wavelets (DRW) frames $\{R_M, R_m^k(x)\ ;\ m = 1, \ldots, M\ ;\ k = 1, 2, 3\}$ are quantized, coded, and transmitted.

The four variations in the implementation of the predictive search algorithm for multiresolution motion estimation can be described as follows:

C–I Motion vectors are calculated for $S_8$ , and no further refinement is carried out. The values are scaled according to the resolution and are used for all the other subbands.

C–II The motion vectors are calculated for all four lowest-resolution fre-

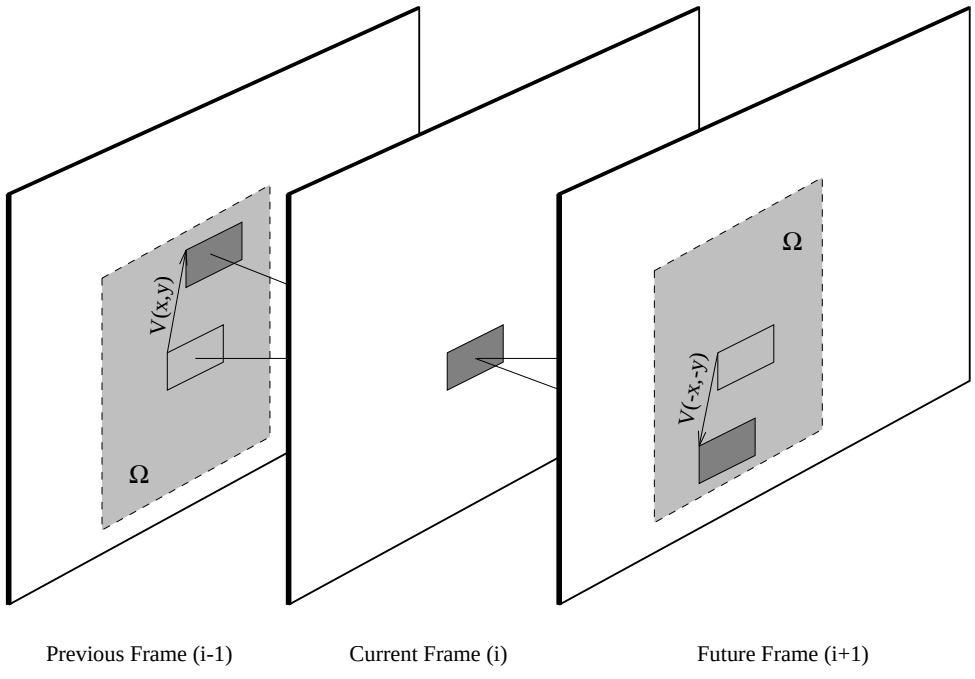



Figure 2: Bidirectional Motion Search for a B-frame

quency bands, i.e., $S_8$ and $\{W_8^i \;\; i = 1, 2, 3\}$. These values are appropriately scaled and used as the motion vectors for all the corresponding higher-resolution subbands.

C–III The motion vectors for $S_8$ are calculated and used as an initial bias for refining the motion vectors of all the other frequency bands and resolutions.

C–IV The motion vectors are calculated for all frequency bands at the highest level, i.e., $\{S_8, W_8^i \; : \; i = 1, 2, 3\}$, and these are used as an initial bias for refining the motion vectors of all lower-level corresponding bands.

Algorithms C–I and C–II use the simplest prediction model, where all the prediction coefficients are zero except one that corresponds to $S_8$ , which is set to $2^{M-m}$. Thus, the motion vectors at the resolution $m$ are given by:

$$\mathbf{V}_{i,j}^{(m)}(x, y) = \mathbf{V}_{i,0}^{(M)}(x, y)\, 2^{M-m} + \Delta^{(m)}(x, y) \tag{2}$$

for $\{j = 1, 2, 3\}$. Since no refinement is done for C–I, $\Delta^{(m)}(x, y)$ is set to *zero*. Similar equations apply to algorithms C–II through C–IV. The four algorithms are summarized in Table 1.

Table 1: Description of MRMC Algorithms.

| | |
|---|---|
| C–I | $S_8$ Only |
| C–II | $\{S_8, W_8^i \; : \; i = 1, 2, 3\}$ Only |
| C–III | $S_8$ + Refine |
| C–IV | $\{S_8, W_8^i \; : \; i = 1, 2, 3\}$ + Refine |

# III Block-Classified Bidirectional Motion Search

## A Block Classification

In a symmetric bidirectional scheme a block is compensated with the average of the previous and future frame as described by Equation (1). This process is based on the underlying assumption that the displacement of the block is the same between the

past and the current frame and between the current and the future frame, a condition that may not always be true. Therefore, in the block-classified bidirectional scheme proposed here, a block can be classified as uncompensated, motion compensated either from the previous frame or from the future frame, or symmetric bidirectionally compensated depending on which type yields the minimum energy. The matching criterion is assumed to be MAD. Therefore, a block in current frame $i$ has motion vector $\mathbf{V}_i(x,y)$ given by

$$\mathbf{V}_i(x,y) = \arg \quad \min\Big\{\mathbf{V}_i^{(1)}, \mathbf{V}_i^{(2)}(x,y), \mathbf{V}_i^{(3)}(x,y), \mathbf{V}_i^{(4)}(x,y)\Big\}, \tag{3}$$

where

$$\mathbf{V}_i^{(1)} = \sum_{p=-X/2}^{X/2} \sum_{q=-Y/2}^{Y/2} \big| I_i(x_m + p, y_n + q) \big| \tag{4}$$

$$\begin{aligned} \mathbf{V}_i^{(2)}(x,y) = \arg \min_{x,y \in \Omega} \Big\{ & \sum_{p=-X/2}^{X/2} \sum_{q=-Y/2}^{Y/2} \big| I_i(x_m + p, y_n + q) \\ & - I_{i-1}(x_m + p + x, y_n + q + y) \big| \Big\} \end{aligned} \tag{5}$$

$$\begin{aligned} \mathbf{V}_i^{(3)}(x,y) = \arg \min_{x,y \in \Omega} \Big\{ & \sum_{p=-X/2}^{X/2} \sum_{q=-Y/2}^{Y/2} \big| I_i(x_m + p, y_n + q) \\ & - I_{i+1}(x_m + p - x, y_n + q - y) \big| \Big\} \end{aligned} \tag{6}$$

$$\begin{aligned} \mathbf{V}_i^{(4)}(x,y) = \arg \min_{x,y \in \Omega} \Big\{ & \sum_{p=-X/2}^{X/2} \sum_{q=-Y/2}^{Y/2} \big| I_i(x_m + p, y_n + q) \\ & - 1/2 \big( I_{i-1}(x_m + p + x, y_n + q + y) \\ & + I_{i+1}(x_m + p - x, y_n + q - y) \big) \big| \Big\}. \end{aligned} \tag{7}$$

Equation (4) is just the energy of the block itself without any compensation, Equations (5) and (6) represent motion prediction with only the previous and future frames, respectively. Equation (7) is the same symmetric bidirectional motion prediction equation as in (1).

A block is thus classified into one of the four classes depending on which $\mathbf{V}_i^{(\cdot)}(x,y)$ is minimum. Therefore, the type of the motion block, $T_i$, is given by

$$T_i = \begin{cases} 1 & \text{if } \mathbf{V}_i^{(1)} \leq \mathbf{V}_i^{(2)}, \mathbf{V}_i^{(3)}, \mathbf{V}_i^{(4)}, \\ 2 & \text{if } \mathbf{V}_i^{(2)} < \mathbf{V}_i^{(1)}, \mathbf{V}_i^{(3)}, \mathbf{V}_i^{(4)}, \\ 3 & \text{if } \mathbf{V}_i^{(3)} < \mathbf{V}_i^{(1)}, \mathbf{V}_i^{(2)}, \mathbf{V}_i^{(4)}, \\ 4 & \text{if } \mathbf{V}_i^{(4)} < \mathbf{V}_i^{(1)}, \mathbf{V}_i^{(2)}, \mathbf{V}_i^{(3)}. \end{cases} \tag{8}$$

The block-type information, which is an added overhead, is transmitted by using a lossless coding scheme. Note that if the block is classified as type *1*, then it has no motion vectors because it is transmitted as is, without any compensation.

## B Frame Classification

The MPEG standard identifies three types of frame in its specifications, namely, I-, P-, and B-frames. I-frames are refresh frames that do not have any dependencies and can be reconstructed on their own. These frames are required for synchronization and reference purposes. P-frames are motion-compensated frames predicted from an already reconstructed I- or P-frame. B-frames are symmetric bidirectional frames predicted from the most recent I- or P-frame and the first available I- or P-frame to follow.

Simulations have shown that the *SNR* slowly deteriorates with time when interframe prediction of any kind is used as a result of error accumulation. The refresh frames (which are sent periodically or by demand) help *refresh* this error buildup, which will otherwise not only increase the bit rate but also decrease the *SNR*. The buffering requirement in a simple interframe predicted scheme is one frame buffer. With bidirectional prediction, this requirement increases to two buffers in the MPEG specifications. Further increase in buffers is an economic constraint rather than a practical or implementation issue. With the rapid decrease in cost per byte of memory, this issue may soon be alleviated. Therefore, provided sufficient buffering capacity, the *SNR* can be increased.

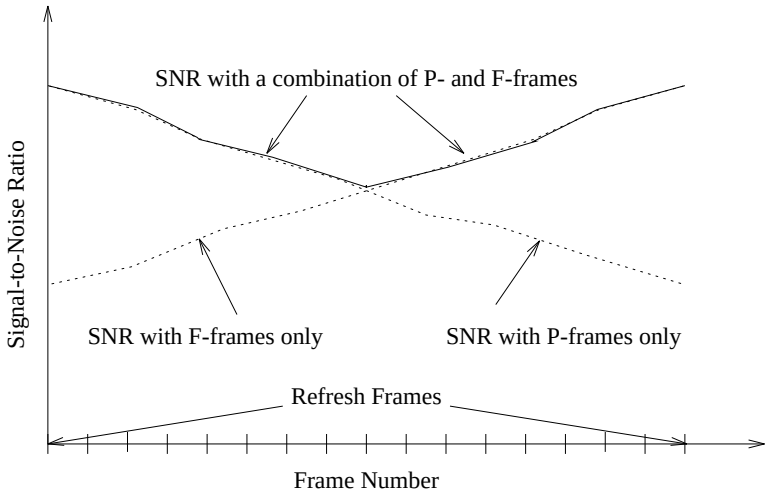



Figure 3: Improvement of SNR with F-frames

Besides the I-, P-, and B-frames, another type of frame, referred to as the F-frame, is proposed. An F-frame is a frame predicted only in the forward direction from a following frame. The receiver is assumed to have enough buffer to hold all the frames between and including the two refresh I-frames. An F-frame is thus just like a P-frame but predicted by a frame following it. If the order of the frames were reversed between the two I-frames, then an F-frame would behave exactly like a P-frame and a P-frame like an F-frame. The behavior of the *SNR* of the F-frames in the reversed case would be the same as that of the P-frames in the normal ordering, that is, decreasing with time (or with time reversal, progressively increasing). This process is illustrated in Figure 3. Therefore, by introducing F-frames during the latter half of time (between the two I-frames), as opposed to P-frames, which are transmitted during the first half, the overall *SNR* for the frames transmitted between the two I-frames is improved. B-frames can be introduced between the P-frames and between the F-frames, with their dependency on either or both.

A typical classification of frames is shown in Figure 4 for a sequence of frames between the two refresh frames. The refresh rate in this example is 10 frames. Frames 3 and 5 are P-frames, whereas 7 and 9 are F-frames. The rest of the frames, (2, 4, 6, and 8) are B-frames. The dependency of each frame is also shown in the figure. Frame 2, which is a B-frame, depends on 1 and 3. Frame 3 depends only on 1, and frame 9 depends only on 10. Therefore, the transmission sequence is 1, 10, 3, 2, 5, 4, 9, 7, 6, 8, and then 20 and so on for the next refresh cycle, or in other words, $\ldots IPBPBFBFBI\ldots$ and so on.

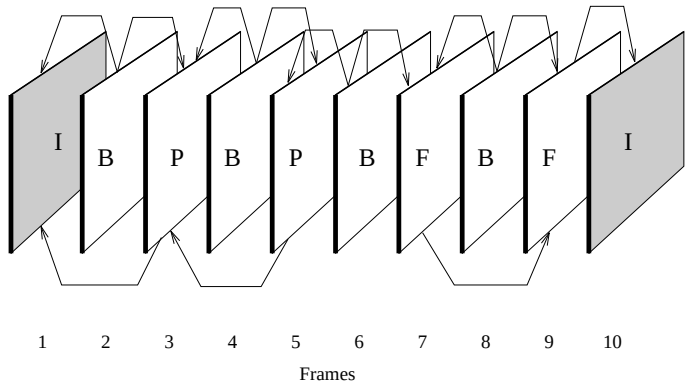



Figure 4: Frame Classification in a Typical Sequence

## C Characterization of the Motion Field

Motion compensation at higher levels of the multiresolution pyramid may not result in all-zero displaced residual wavelets, even when the displacement at full resolution of the video is a pure translation. It has been shown in [10] that for two consecutive frames $S^0_{i-1}$ and $S^0_i$ which are related by pure translation displacement of $(x, y)$, that is, $S^0_i(x_m, y_n) = S^0_{i-1}(x_m + x, y_n + y)$, the following holds:

$$S^1_i(\omega_1, \omega_2) = \frac{1}{4} H_{00}(\frac{\omega_1}{2}, \frac{\omega_2}{2}) S^0_{i-1}(\frac{\omega_1}{2}, \frac{\omega_2}{2}).e^{-jx(\omega_1/2)} e^{-jy(\omega_2/2)} + \frac{1}{4} \sum_{(k,l)\in\{(0,1),(1,0),(1,1)\}} H_{00}(\frac{\omega_1}{2} + k\pi, \frac{\omega_2}{2} + l\pi) .S^0_{i-1}(\frac{\omega_1}{2} + k\pi, \frac{\omega_2}{2} + l\pi).e^{-jx(\omega_1/2+k\pi)} e^{-jy(\omega_2/2+l\pi)},$$

where $H_{00}(\frac{\omega_1}{2}, \frac{\omega_2}{2})$ and $S^0_{i-1}(\frac{\omega_1}{2}, \frac{\omega_2}{2})$ are the Fourier transform of the 2-D lowpass filter and the original image of the previous frame, respectively. It can be seen that the direct components are related through translation, whereas the aliased components are not, unless the subsampling is Nyquist or the translation is an integral multiple of subsampling factor. Thus, in general, the images at any level are not related through simple translation even though the original full resolutions are.

Therefore, it is expected that the motion compensation on a hierarchically represented video will be less effective than that of the full-resolution image. This statement should be taken literally as to what it says. It does not mean that the "overall" performance of a multiresolution motion compensated codec will not be better than the performance of the video codec which first applies motion compensation to the original image and then performs the decomposition on the displaced frame difference. It just states that the motion compensation on a hierarchically represented video be less effective as compared to motion compensation of the full resolution image. Such a comparison is given in [22], where it has been shown that overall a multiresolution motion compensation performs better than motion compensation at the original scale, considering all the factors like the bit rate, *SNR*, and computational and search complexity.

# IV Simulation Results

The "car" test sequence was used in most of the simulations and is a full-motion interlaced color video sequence in CCIR 601 4:2:2 format with 240×720 pixels per field in the Y component, and 240×360 each in the U and V components, all with

Table 2: Energies in a typical Displaced Residual Wavelets for Algorithms C–I through C–IV.

| **Energy** | $S_8$ | $W_8^1$ | $W_8^2$ | $W_8^3$ | $W_4^1$ | $W_4^2$ | $W_4^3$ | $W_2^1$ | $W_2^2$ | $W_2^3$ |
|---|---|---|---|---|---|---|---|---|---|---|
| Original Image | 49587.23 | 7361.20 | 452.91 | 148.47 | 1391.86 | 65.89 | 18.46 | 203.53 | 7.48 | 3.31 |
| C–I | 330.89 | 843.38 | 152.59 | 193.18 | 421.56 | 56.24 | 42.68 | 117.07 | 7.86 | 4.98 |
| C–II | 330.89 | 190.41 | 42.98 | 27.98 | 430.56 | 46.99 | 25.57 | 103.97 | 7.42 | 4.26 |
| C–III | 330.89 | 181.34 | 43.54 | 28.69 | 136.24 | 15.00 | 7.15 | 46.30 | 3.51 | 2.76 |
| C–IV | 330.89 | 190.41 | 42.98 | 27.98 | 142.56 | 16.60 | 9.92 | 45.87 | 3.31 | 2.89 |

8 bits/pixel. It is a fast panning sequence and thus ideal for testing various motion compensation schemes. Experimental results are also obtained for other sequences including the "cheer leaders" and "football" used for MPEG testing. All the results and parameters follow the same pattern, although actual numbers may turn out to be different. The results obtained for these sequences will be explicitly specified, but the default is the "car" sequence. In all the results, the video signal was decomposed to three levels using the Daubechies' 4-tap filter unless specified.

Table 2 show the energy distribution in the displaced residual wavelets of the Y component of a P-frame, after applying the algorithms C–I through C–IV, in comparison with the uncompensated or original image. The motion block size for $S_8$ was 2×2. The table shows that after motion compensation, the energies in all subbands are considerably reduced. The reduction is an order of magnitude for the highest-layer subbands $\{W_8^i \;\; i = 1, 2, 3\}$ and more than two orders of magnitude for $S_8$ . This significant decrease of energy in the perceptually most significant subband is a result of the motion estimation.

A comparison of algorithms C–I through C–IV reveals that although the energy drops dramatically for $S_8$ , it may increase for other subbands if the motion vectors are not refined, as in case of Algorithms C–I and C–II. However, even when the motion vectors are refined, as in C–III and C–IV, some anomalies may still arise because of the reduced search area. In general, however, algorithms C–III and C–IV produce less energies than C–I and C–II.

A slight increase in the energy at the lower layer results in a much higher contribution in overall bit rate than a relatively large increase at the top layer, because of the difference in the number of samples at these levels. It should be emphasized here that the values for the variance (or energy) in the displaced residual wavelets with different scenarios shown in Table 2 are for *unquantized* coefficients and therefore do not necessarily correspond directly to each subband's contribution to the overall bit rate. Specifically, the table shows the relative performance of each of the four

Table 3: Entropy (in Bits/Sample) of DRWs of Y component for Algorithm C–I before and after Adaptive Truncation.

| **Entropy** | $S_8$ | $W_8^1$ | $W_8^2$ | $W_8^3$ | $W_4^1$ | $W_4^2$ | $W_4^3$ | $W_2^1$ | $W_2^2$ | $W_2^3$ | Average |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Un-quantized | 5.35 | 5.47 | 4.29 | 3.98 | 5.00 | 3.50 | 3.40 | 4.58 | 2.38 | 2.41 | 3.38 |
| $Q_M = 8.0$ | 2.55 | 2.86 | 1.73 | 1.46 | 1.53 | 0.47 | 0.36 | 0.50 | 0.05 | 0.02 | 0.42 |
| $Q_M = 12.0$ | 1.99 | 2.31 | 1.24 | 1.05 | 1.13 | 0.27 | 0.21 | 0.28 | 0.02 | 0.00 | 0.28 |

multiresolution motion estimation algorithms C–I through C–IV. The table also reflects the energy compaction of the wavelet filter used and thus gives a general idea of the contents of the original signal. Since each level is treated according to its visual importance, the final quantized figures look much different.

Quantization is the most important part of any compression algorithm because it is quantization that determines the amount of compression or the final bit rate. Scale adaptive truncation [25] was used to quantize the wavelet coefficients. The first-order entropies of the ten subbands in the luminance signal before and after quantization for algorithm C–I are shown in Table 3. The values of $Q_M$, as described in [25] are 8.0 and 12.0. The table clearly reveals the adaptive truncation process, which gives less and less importance to the subbands as we go down the hierarchy. The reduction in entropy is more significant for the lower-layer wavelets, since these are quantized with the fewest number of bits. Each column represents the entropy in bits per sample/pixel for that band, while the average, shown in the last column, is the average over all the subbands and thus represents a number for the original resolution of the input video.

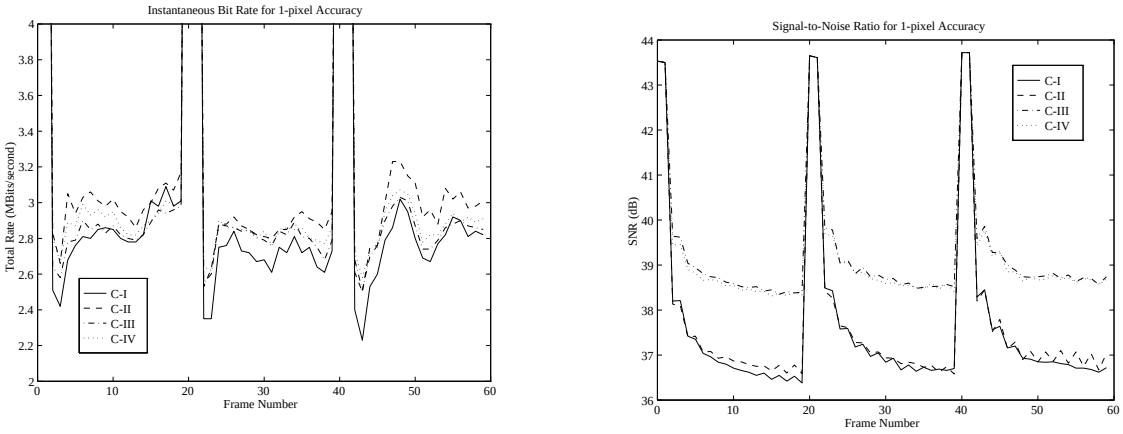
The contribution to the bit rate from the top layer is the most significant despite the smaller size as compared with the subbands in the other layers. A particular figure of entropy for a frame depends on $Q_M$ and the amount of motion in that frame. Subbands in the same layer exhibit different behavior in terms of energy contents and entropy, depending on the motion present in the direction to which the filters are sensitive. Some of the subbands show a value of zero (e.g. $W_2^3$ with $Q_M = 12.0$), which means that the coefficients after quantizing are insignificant and thus truncated to zero. This particular subband will not play any part in the reconstruction. After adaptive truncation, the energy does not drop much, but the entropy drops significantly as a direct result of quantization. In fact, in some cases, the variance might even increase.

The two most important performance measures in video compression or image compression in general are the output bit rate and the corresponding reconstructed peak *SNR*. The total instantaneous bit rate of any frame is the sum of the bits

generated by encoding of the actual frames plus the motion information, which must be sent *losslessly* in order to reconstruct the frames at the decoder. Figures 5(a) and 5(b) show the total instantaneous bit rate (Mbits per second) and the corresponding *SNR* when using P-frames only.

Notice that the I-frames are treated in a special way. Since these synchronizing refresh frames will form the base for all the following interpolated frames, they are quantized with more levels than are the interpolated frames, in order to get a good starting *SNR*. This quantization results in a sudden increase in the output bit rate, appearing as spikes in Figure 5(a). In order to reduce this sudden increase, which jumps to more than twice the average bit rate, the value of the normalizing factor $Q_M$ can be increased for these I-frames only. The resulting bit rate of algorithm C–I is shown in Figure 6(a) for values of $Q_M$ ranging from 2.0 to 10.0 while keeping the value for the P-frames at 8.0.

Figure 6, as expected, reveals that the number of bits generated by the I-frames drops monotonically with the increasing value of $Q_M$. Observe that there is a marginal increase in the bit rate of the P-frames when $Q_M$ is increased from 2.0 to 4.0. This increase results from the fact that the refresh frame used for motion compensation is not that *clean*, or, in other words, the *SNR* is comparatively poor because of coarser quantization. Thus, the displaced residual frame, based on this *poor* quality frame, will have more energy. Observe that a frame after quantization is reconstructed back at the encoder. It is this frame that is used in motion

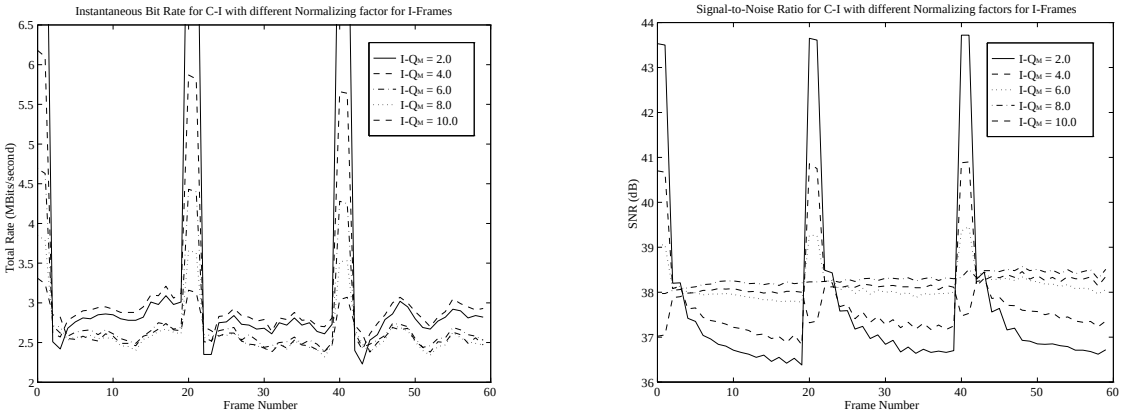



(a) Bit Rate (b) SNR

Figure 5: Bit Rate and Reconstructed Signal-to-Noise Ratio for Algorithms C–I – C–IV with P-frames Only

compensation, rather than the *clean* unquantized frame. Therefore, the resultant displaced residual difference for an incoming frame will have a higher energy if subtracted from this quantized frame buffer than if compensated from an unquantized one. The interesting result is that for all other higher values of $Q_M$, the bit rate is almost the same (even lower than that for $Q_M = 2.0$), which can be attributed to the adaptive truncation process that does a good job at keeping an almost constant bit rate. Regardless of small variations in the bit rates, the maximum difference does not exceed 500 Kbps for the P-frames.

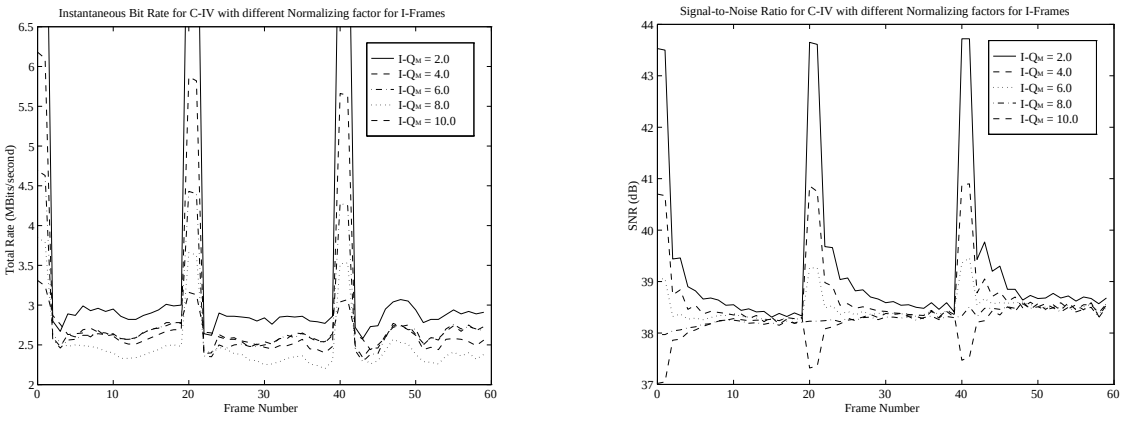
The most interesting result is observed for the *SNR* curves shown in Figure 6(b). The *SNR* for I-frames decreases as $Q_M$ is increased, as expected, but note that the values for the P-frames start to increase and then level off with those of the I-frames when the normalizing factor for both types of frames is identical (i.e., 8.0). At this point, even though the scaling factors are the same (which makes the *SNR* almost identical), the bit rate for the I-frames is still significantly higher. This anomaly is a result of the internal feedback loop. Since the quality of the I-frame is dropping, the predictor that uses the quantized frame as reference, instead of the clean unquantized version, compensates very effectively. It can be observed that a gain of almost 2.0 dB can be achieved along with a decrease in time average bit rate over an entire refresh period. If this feedback loop is eliminated and a clean frame buffer is used for motion compensation, this gain in the *SNR* is not achieved and the curves for values of $Q_M > 2.0$ are identical to those of $Q_M = 2.0$ but shifted down. Observe that for $Q_M = 10.0$, the *SNR* of the P-frames is higher than that of




(a) Bit Rate

(b) SNR

Figure 6: Bit Rate and Reconstructed Signal-to-Noise Ratio for Algorithm C–I with Variation of $Q_M$ for I-frames Only




(a) Bit Rate (b) SNR

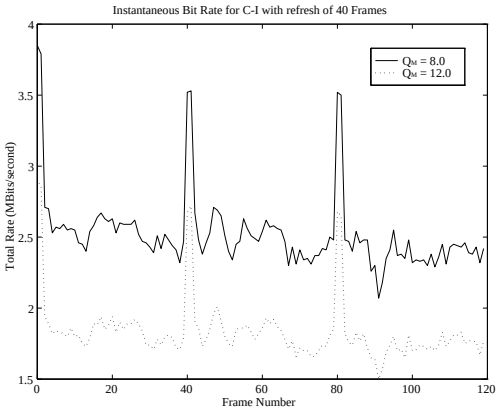
Figure 7: Bit Rate and Reconstructed Signal-to-Noise Ratio for Algorithm C–IV with Variation of $Q_M$ for I-frames Only

the I-frames.

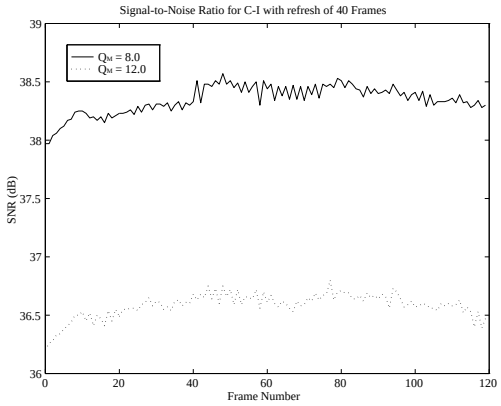
The output bit rate curves obtained by varying quantization on I-frames for Algorithm C–IV are shown in Figure 7(a). These reveal the same pattern obtained for Algorithm C–I in Figure 6(a). Notice a steady decrease in bit rate for increasing $Q_M$ until it is the same as that of the P-frames at a value of 8.0. Further increase in $Q_M$ for the I-frames slightly increases the bit rate for the P-frames because of the *less clean* frame buffer.

The *SNR* curves for varying the quantization on only the I-frames for algorithm C–IV are shown in Figure 7(b). In contrast to the results obtained for Algorithm C–I, the *SNR* for Algorithm C–IV does not show the same improvement. The reason is that in Algorithm C–IV, motion vectors for all the subbands are refined and are already generating the minimum energy in all the bands; thus, there is no improvement in the *SNR* for the P-frames. Notice that the average *SNR* for the P-frames is governed only by the quantization factor for the P-frames and not by that of the I-frames. Also notice that Algorithms C–I and C–IV behave almost identically at values of $Q_M \geq 6.0$.

As we have noticed, the *SNR* for any algorithm depends basically on the $Q_M$ values for both I-frames and P-frames. We have also observed in the previous results that if $Q_M = 8.0$ for both types of frame, we get an almost constant picture quality with varying bit rate regardless of the refresh rate. To investigate further, we selected a refresh cycle of 40 frames with values for $Q_M = 8.0$ and $Q_M = 12.0$ for




(a) Output Rate



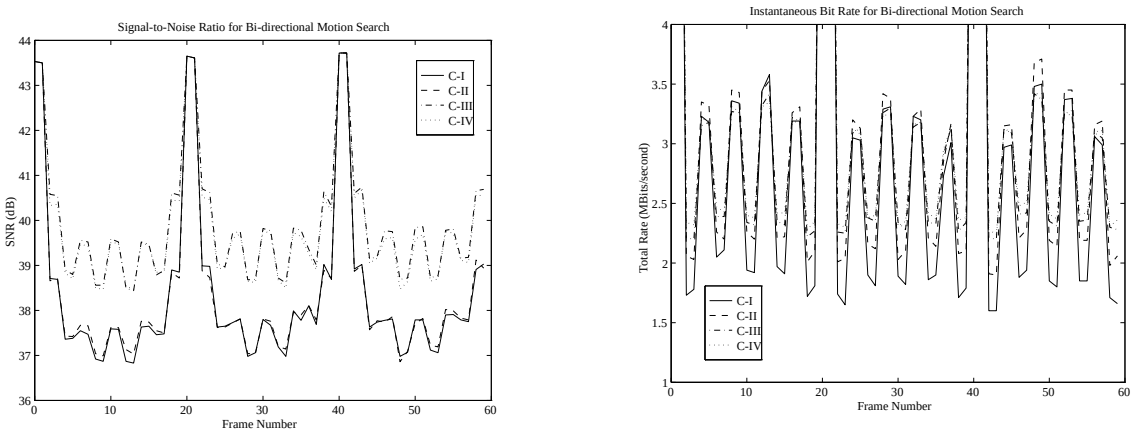

(b) Signal-to-Noise Ratio

Figure 8: Performance of C–I with a Refresh Cycle of 40 Frames

both the I-frames and the P-frames. The results are given in Figures 8(a) and 8(b), which show the output rate and *SNR*, respectively, for Algorithm C–I. The figures reveal that the instantaneous rate peaks after every 40 frames when an I-frame is transmitted. The average for the predicted frames is below 2.7 Mbps in the case of $Q_M = 8.0$ and below 2.0 Mbps for $Q_M = 12.0$. The *SNR* for the entire length of the sequence is fairly constant between 38.0 and 38.5 dB with $Q_M = 8.0$ and about 36.5 dB for $Q_M = 12.0$.

In the Bidirectional motion search algorithm described in Section III, the current, previous and forward frames were searched for the minimum energy (minimum absolute difference) to find the motion vectors for the B-frames. The concept of F-frames was also introduced in that section. Figure 9(a) shows the *SNR* for Algorithms C–I through C–IV using a frame sequence of $IBPBPBFBFBI$ for one GOP, where every other frame is a B-frame and the second half of the refresh cycle contains F-frames rather than P-frames. All the other parameters are kept the same as for nonbidirectional case (results for which are shown in Figure 5(b)). The scaling factor $Q_M$ for P-frames, F-frames, and B-frames is kept at 8.0 for comparison purposes. As expected, the curves follow the general shape of Figure 3. Note that the F-frames increase the *SNR* for the frames in the latter half of the refresh cycle because of the extra buffering.

In this case too, Algorithms C–I and C–II have almost similar *SNR* values. Moreover, the *SNR* values for Algorithms C–III and C–IV are very close to each other, as in the case of nonbidirectional motion compensation. The relative performance of the four algorithms is independent of the type of motion compensation involved.



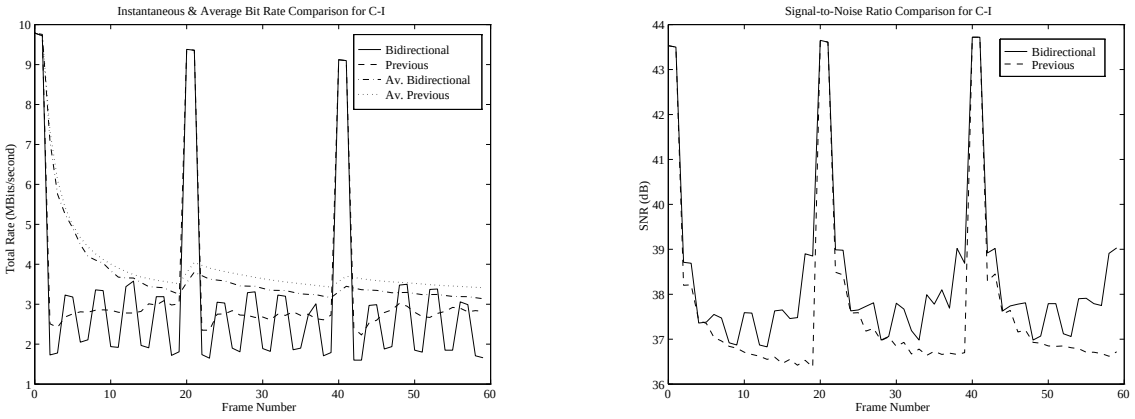

(a) Signal-to-Noise Ratio

(b) Output Rate

Figure 9: Signal-to-Noise Ratio and Output Rate of C–I through C–IV Using Bidirectional Motion Search

We again notice a 2 dB difference in the *SNR* between the two algorithms that refine the motion vectors, namely, C–III and C–IV, and the two that do not refine but use the motion vectors of the lowest resolution layer, namely, C–I and C–II.

The corresponding bit rates for the bidirectional motion search are shown in Figure 9(b). The equivalent non-bidirectional motion search curves are shown in figure 5(a). It can be seen that the bit rate for P-frames is much higher than that for the B-frames, which is to be expected. The B-frames have the least possible energy, and since the quantization factor for the two types of frame is the same, the bit rate for B-frames is lower. Notice that the output rate for the P-frames is higher than the corresponding frame in the nonbidirectional search case. The reason is that as far as the P-frames (or the F-frames) are concerned, they are separated further apart in time from their reference frame buffer (in the above case there is now one B-frame between two consecutive P-frames or F-frames). As a result of this separation, the P-frames will experience a higher motion activity, since the search area is not increased. There will now be more blocks that will not be trackable, and thus there will be an overall higher residual energy for the frame as compared with the case when motion prediction is performed using the immediately adjacent frame. Also observe from Figure 9(b) that the bits generated by P-frames for all the algorithms are considerably close to each other, while that is not true of the B-frames. The reason is due to different statistical characteristics of the coefficients in the B-frames as compared with those of the P-frames or the F-frames, in spite of the normalizing factor $Q_M = 8.0$ for all the three types.




(a) Output Rate

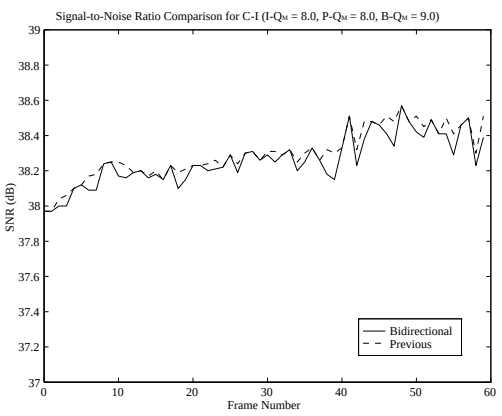
(b) Signal-to-Noise Ratio

Figure 10: Performance Comparison of C–I with and without Bidirectional Motion Search

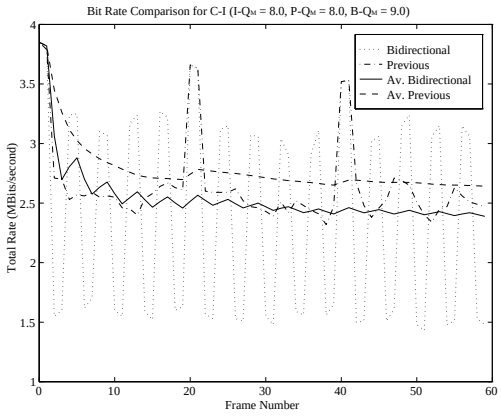
Concentrating on Algorithm C–I only and comparing the bidirectional motion search with previous frame prediction, we notice that the *SNR* in the bidirectional case is always higher. For comparison purposes, the instantaneous output rate and *SNR* for Algorithm C–I are reproduced in Figures 10(a) and 10(b), respectively. Also shown in Figure 10(a) is the average bit rate (average over time) comparison for the two scenarios, we see that, despite the higher bits for P-frames and F-frames, the overall bit rate (time average) for the bidirectional case is lower than that for the nonbidirectional case.

The first-order entropy comparison for Algorithm C–I is given in Table 4 which shows the entropy for a frame encoded as a P-frame vs the entropy encoded as a B-frame, both quantized with $Q_M = 8.0$. Though the table shows quantized entropy, the unquantized energies have the same pattern. Notice that the bidirectional search results in 43% reduction in the entropy of $S_8$ and almost 30% reduction in the average bit rate (which drops from 2.51 Mbps to 1.73 Mbps) and still increases the *SNR* by 0.5 dB.

For a better comparison between the traditional search scheme and a bidirectional search scheme, another set of simulations were run in which the quantization parameters were chosen so that the *SNR* was almost the same for the two cases (where the output rate is more difficult to control). Specifically, the quantization factor used was $Q_M = 8.0$ for I-frames and P-frames for both bidirectional search and previous frame search and $Q_M = 9.0$ for B-frames of the bidirectional search case. Note that these sets of parameters produced the best results [22] for the non-




(a) Signal-to-Noise Ratio




(b) Output Rate

Figure 11: Signal-to-Noise Ratio and Output Rate of Algorithm C–I with and without Bidirectional Motion Search Quantized with Different $Q_M$ for B-frames

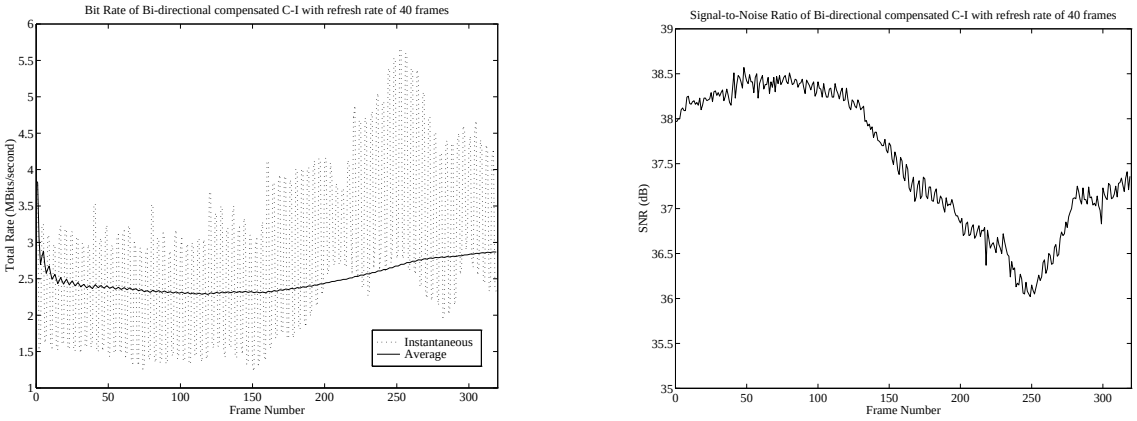
bidirectional scheme. Parameters for the bidirectional case were adjusted to match this *SNR*. The results are shown in Figure 11(a). The corresponding instantaneous and average output rates are shown in Figure 11(b).

A significant difference is apparent in the average output rates for the two cases. The bidirectional search produces an average rate at least 0.4 Mbps less than that of the traditional previous-frame search, although the instantaneous rate of P-frames is higher. B-frames have an average rate of approximately 1.6 Mbps, which is the main reason for the lower time average.

Figures 12(a) and 12(b) show the results with a group of picture sequence of $IBPBPBPBPBPBPBPBPBPBI$, that is, having a refresh rate of 40 frames for the entire "car" sequence of 320 frames. Observe that once again, because of the selection of the normalizing factors, this long refresh cycle does not degrade the

Table 4: Entropy comparison of DRWs of the Y component for Algorithm C–I with and without Bidirectional Motion Search.

| **Entropy** | $S_8$ | $W_8^1$ | $W_8^2$ | $W_8^3$ | $W_4^1$ | $W_4^2$ | $W_4^3$ | $W_2^1$ | $W_2^2$ | $W_2^3$ | Average |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Previous Frame | 2.55 | 2.86 | 1.73 | 1.46 | 1.53 | 0.47 | 0.36 | 0.50 | 0.05 | 0.02 | 0.42 |
| Bidirectional | 1.46 | 1.86 | 1.26 | 1.11 | 0.97 | 0.37 | 0.30 | 0.33 | 0.04 | 0.01 | 0.29 |




(a) Output Rate

(b) Signal-to-Noise Ratio

Figure 12: Performance of Bidirectional Motion-Compensated Algorithm C–I with Refresh Rate of 40 Frames using the "Car" Sequence

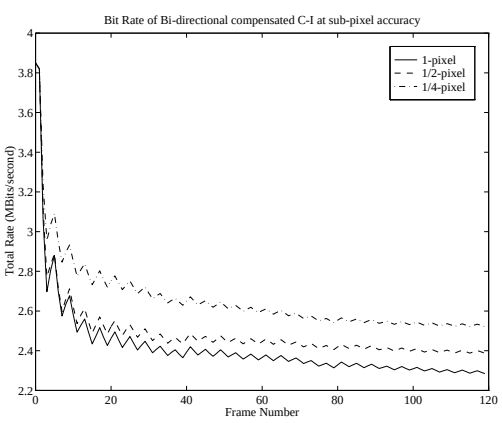
performance. We do notice an increase in bit rate starting around frame 150 and then peaking to as high as 5.5 Mbps around frame 250. At the same time the *SNR* drops from 38 dB to 36 dB. The reason is extremely high motion in the background during this period as the car, which the camera is following, comes nearer to the camera and passes by. Notice that the performance improves, both in terms of rate and *SNR*, after the car has passed by (which is around frame 280). The subjective quality at this point is still quite good because of a very high degree of motion blur, which conceals the drop in *SNR*. Even with such a high bit rate during the high motion activity, the average over the entire length of the sequence is still less than 3 Mbps.

The performance of Algorithm C–I at subpixel accuracies was compared with integer pixel results. Figure 13(a) shows the bit rate comparison at different pixel accuracies in the bidirectional case and with a refresh rate of 40 frames. The normalization factors and other parameters are the same in all three cases. The figure reveal that even for this set of parameters, as in the case of previous simulations [23], subpixel accuracy does not achieve any improvement, contrary to what was anticipated. The main reason is not that there is no reduction in the entropy of the displaced residual wavelets, especially in case of Algorithms C–III and C–IV, but that there is a significant increase in motion overhead, which leads to a higher bit rate for the entire frame. In the case of Algorithms C–I and C–II the higher bit rate is generally due to higher bit contributions from some of the subbands in the lower levels. In other words, since the spatial orientation of the filters is different, differ-

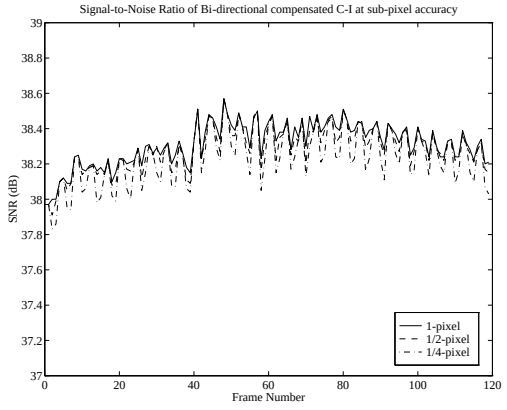
ent shifts result. Therefore, nonrefinement of motion vectors may result in higher energy as compared with the 1-pixel accuracy case. It can be seen that the average bit rate generated by integer pixel accuracy is lower than the rates generated by subpixel accuracies. The corresponding *SNR* is shown in Figure 13(b), which reveals no improvement in the *SNR*.

The same encoder parameters used to encode the "car" sequence were also used to encode the "football" sequence. The bit rates and *SNR* are shown in Figures 14(a) and 14(b), respectively. The high average bit rate of about 3.8 Mbps is because of the high motion in a football game, which this sequence shows. Also notice that the overall *SNR* is about 35.3 dB during the entire segment of the sequence. One reason for such a low quality is that the original sequence itself has a lower quality than does the "car" sequence.

The introduction of B-frames is responsible for the reduction in the average bit rate in the bidirectional motion searching scheme. In all the above simulation results showing the bidirectional case, the frame sequence used was $\ldots IBPBPB\ldots$; that is, every other frame is a B-frame. Note that in the case of a B-frame, its previous reference frame contains any reconstructed frame regardless of its type. It can be an I-frame, a P-frame, an F-frame, or even a B-frame which has already been reconstructed. This selection of already reconstructed B-frame as a candidate for reference of a future frame is different from what MPEG identifies as the frame dependencies for a B-frame. The argument for using such a scheme is that since the B-frame has already been constructed in the past, it is available as a reference.




(a) Output Rate



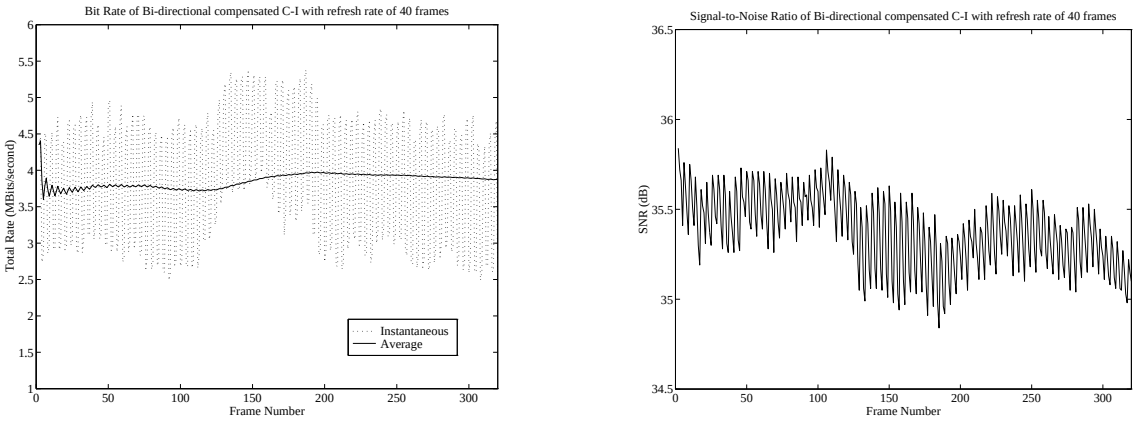

(b) Signal-to-Noise Ratio

Figure 13: Average Bit Rate and SNR of Bidirectional Motion-Compensated Algorithm C–I at 1-, $^1/_2$-pixel $^1/_4$-pixel Accuracy

Note that this scheme is still free of any deadlocks because the frame used as the forward reference in the motion estimation can only be an I-frame, a P-frame, or an F-frame and never a B-frame.

Therefore, in the proposed bidirectional motion searching scheme, if two consecutive B-frames are encoded, as for example, in a sequence $\ldots IBBPBPBFBBI\ldots$, the first B-frame (frame 2) backward dependency is the I-frame (frame 1), and the forward dependency is frame 4, which is encoded as a P-frame. But for the second B-frame (frame 3), the backward dependent frame is frame 2, a B-frame, and is forward dependent on frame 4, which is a P-frame. The first B-frame cannot be forward dependent on frame 3 because a deadlock would result at the decoder (both frames interdepending on each other). Note that in the case of the first B-frame, its two dependent frames are not symmetrically distributed in time. Frame $(i-1)$, as described in the definition of a bidirectional motion-compensated block (Equation 6), is adjacent to the current frame, while frame $(i+1)$ is two frames apart. Therefore, there will be a small number of blocks that will be classified as type *4* as compared with the case where both the dependent frames are symmetrically located. This asymmetric location of the frames degrades the performance, since the whole purpose of bidirectional search is defeated.

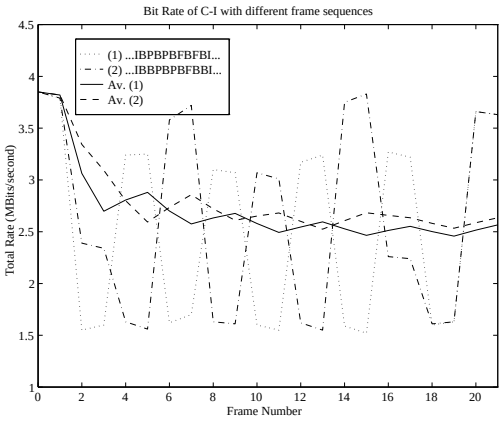
Figure 15(a) compares the bit rates generated by the framing sequence (1) $\ldots IBPBPBFBFBI\ldots$ and (2) $\ldots IBBPBPBFBBI\ldots$ for one typical refresh cycle. Note that the first B-frame (frame 2) in the case of the latter sequence (2) has a higher instantaneous bit rate than the same frame in sequence (1).



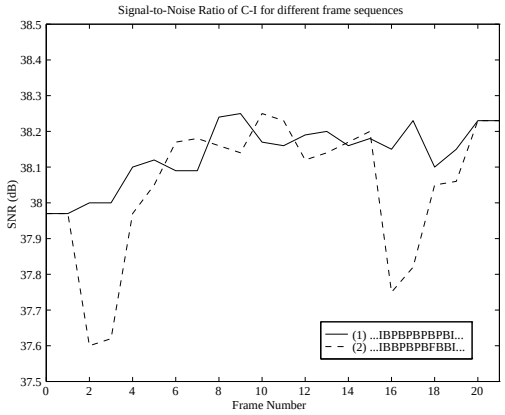

(a) Output Rate

(b) Signal-to-Noise Ratio

Figure 14: Performance of Bidirectional Motion-Compensated Algorithm C–I with Refresh Rate of 40 Frames using "Football" Sequence




(a) Output Rate




(b) Signal-to-Noise Ratio

Figure 15: Rate and Signal-to-Noise Ratio Comparison of Bidirectional Motion-Compensated Algorithm C–I with One and Two Consecutive B-frames

The reference previous and forward frames are symmetrically separated in time in the case of (1), while they are asymmetric for (2). Now note that since frame 3 in the sequence (2) has symmetric dependency when using the proposed scheme, it has a bit rate that is of the same order as any B-frame in sequence (1). Also note that the P-frames immediately following two B-frames have a higher bit rate than the ones in sequence (1) because they are separated farther apart from its reference previous frame. Not only is the energy in the displaced residual wavelets high, but the amount of occluded and freshly revealed areas resulting from moving objects is much higher. These areas do not normally have any relevant blocks in the previous frames and thus cannot be motion compensated.

The *SNR* reveals the same picture as depicted by the rate comparison. The results, shown in Figure 15(b), reveal that the quality of the asymmetrically compensated B-frame is also lower that that of the symmetrically compensated one. This degradation of quality is in addition to a slight increase in the average bit rate. Therefore, we conclude that it is always better not to have two consecutive B-frames, because such a configuration not only degrades the quality but increases the average bit rate.

# V Choice of Wavelet Filter-Bank

The choice of wavelet filter-bank in image compression has been an active research topic during the past few years. It affects not only the image quality but also the overall system design. Regularity of the wavelet has been used traditionally as the criterion for predicting the performance [14], but its relevance to signal processing has not yet been strongly established [20]. The correlation between the reconstructed *SNR* and the regularity of the filter is not very strong either. Some researchers have emphasized orthogonality as a selection criterion [4, 17], but this might conflict with other desirable characteristics. Others have characterized wavelet filter-banks in terms of their associated continuous scaling functions and wavelets derived under iteration [21].

Daubechies' 4-tap filter was the choice for all the simulation results discussed previously in this paper because of its smaller length to reduce the overall complexity. Different filter-banks used for wavelet decomposition, appearing in the literature, have different properties as regards to their spatial orientation and energy compaction, or, in other words, distribution of energy among different bands, linearity of phase, and so on. To investigate the performance of the proposed hierarchical motion estimation scheme, we have tested other wavelet basis. Some of the filters described by Daubechies in [5], for example, the 6-tap, and 8-tap, and the coiflets, were used in the simulations. Some biorthogonal filter-banks discussed in [1] were also tested.

Table 5 shows a comparison of the distribution of energy and the resulting first-order entropy, after the application of adaptive truncation, among the subbands of some of the different wavelets for the multiresolution motion scheme C–I. The first three rows show Daubechies' 4-, 6- and 8-coefficient wavelets followed by the 8-tap *least asymmetric* filter (denoted by "Daubechies' 8-tap Sym."). Only the 6-coefficient filter-bank from the family termed "Coiflets", described both by Daubechies [5] and Antonini [1] et al., was used in this work for comparison purposes. Also compared are the "Burt", 'Spline", and a variant of "Spline" denoted by "Vspline," which are discussed in more detail in [1].

The table shows the variance of the coefficients in each of the ten wavelets for an I-frame, which is not motion compensated, and the resulting entropy after quantization. The normalizing factor for all the different cases is the same for a fair comparison. The figure therefore reveals the energy compaction of the wavelet filters themselves rather than the performance of the multiresolution motion estimation. The first column for each subband shows the energy or variance, while the second column has the entropy after quantization for that particular band. The last column of the table has the average entropy figures, in bits per sample, for the entire frame. The lowest values of entropy for each subband are highlighted (boxed). It can be

Table 5: Comparison of Energy and Entropy of an I-frame with Different Wavelet Filters for Algorithm C–I.

| **Energy/Entropy** | $S_8$ | | $W_8^1$ | | $W_8^2$ | | $W_8^3$ | | $W_4^1$ | | $W_4^2$ | | $W_4^3$ | | $W_2^1$ | | $W_2^2$ | | $W_2^3$ | | Average |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Daubechies' 4-tap | 49796.00 | 6.41 | 7379.50 | 4.79 | 430.27 | 2.14 | 170.94 | 1.55 | 1379.65 | 2.59 | 63.61 | 0.43 | 16.83 | 0.24 | 219.59 | 0.79 | 5.61 | 0.05 | 0.57 | 0.01 | 0.65 |
| Daubechies' 6-tap | 51408.81 | 6.51 | 6155.21 | 4.82 | 351.36 | 2.02 | 126.12 | 1.49 | 1343.76 | 2.64 | 42.81 | 0.39 | 16.28 | 0.22 | 219.62 | 0.79 | 6.73 | 0.05 | 0.45 | 0.01 | 0.65 |
| Daubechies' 8-tap | 51205.25 | 6.45 | 7301.54 | 5.01 | 405.87 | 2.08 | 135.47 | 1.53 | 1126.68 | 2.57 | 27.42 | 0.33 | 14.74 | 0.20 | 211.96 | 0.79 | 6.82 | 0.04 | 0.36 | 0.00 | 0.64 |
| Daubechies' 8-tap Sym. | 51094.48 | 6.51 | 7273.53 | 4.84 | 316.36 | 2.01 | 126.24 | 1.43 | 1210.46 | 2.58 | 35.93 | 0.32 | 15.36 | 0.21 | 201.67 | 0.77 | 4.19 | 0.04 | 0.28 | 0.00 | 0.63 |
| Burt | 49343.71 | 6.41 | 6455.46 | 4.81 | 360.18 | 2.02 | 121.05 | 1.49 | 1298.80 | 2.56 | 43.97 | 0.37 | 17.09 | 0.23 | 225.92 | 0.79 | 6.25 | 0.05 | 0.66 | 0.01 | 0.64 |
| Haar | 49188.73 | 6.29 | 7605.26 | 4.64 | 590.44 | 2.33 | 244.67 | 1.88 | 1198.32 | 2.32 | 78.77 | 0.50 | 29.86 | 0.32 | 249.91 | 0.73 | 8.00 | 0.07 | 2.13 | 0.02 | 0.64 |
| Spline | 66077.40 | 6.70 | 9905.69 | 5.07 | 521.30 | 2.25 | 215.41 | 1.79 | 1418.51 | 2.66 | 54.27 | 0.42 | 19.98 | 0.26 | 148.93 | 0.64 | 4.12 | 0.04 | 0.07 | 0.00 | 0.63 |
| Vspline | 48696.79 | 6.41 | 5875.04 | 4.77 | 260.96 | 1.79 | 95.50 | 1.33 | 1179.19 | 2.57 | 34.26 | 0.33 | 14.41 | 0.20 | 184.90 | 0.74 | 4.10 | 0.04 | 0.21 | 0.00 | 0.61 |
| Coiflet 6-tap | 50095.65 | 6.42 | 7974.90 | 4.83 | 403.39 | 2.10 | 163.40 | 1.51 | 1206.47 | 2.53 | 43.74 | 0.35 | 16.83 | 0.23 | 206.58 | 0.76 | 4.12 | 0.04 | 0.78 | 0.01 | 0.63 |
| Vetterli | 43115.87 | 6.33 | 8675.46 | 5.02 | 461.78 | 2.39 | 209.02 | 1.79 | 1433.79 | 2.68 | 57.02 | 0.49 | 30.93 | 0.32 | 371.27 | 1.01 | 9.93 | 0.07 | 3.51 | 0.03 | 0.74 |

seen from the table that the variant of the Spline filter, which has less dissimilar-length filters, has the lowest overall entropy. Also notice that this particular filter has the lowest or very close to the lowest energy in almost all the subbands rather than just the top layer. In fact, the values at the lower layers are more important in the sense that they contribute to the bit rate more, because of the larger number of samples in these layers as compared with the top layer.

All the filter-banks have almost identical behavior, although individual values are very different from each other. The Spline filter has the highest energy in $S_8$ , while Haar has the lowest. In contrast, the Spline filter has the lowest energy in the all subbands of the bottom layer. Therefore, as far as the I-frames are concerned, the Spline filter is a good choice. The best overall performing filter-bank is the Vspline, since it gives the least average entropy. Note that Daubechies' 8-tap *least asymmetric* filter also has comparable results. The worst-performing wavelet filter-bank is that presented by Vetterli [19], because of the high variance of the coefficients in the lower bands. Since the size of the lower-level subimages is much higher, the contribution to the bit rate is higher, too.

In an interframe coding scheme, I-frames are transmitted periodically for refresh and synchronization purposes only. They constitute less than 5% of the total frames, obviously depending on the refresh rate. The majority of the frames in a sequence are the predicted frames, in this case, the P-frames and the B-frames. The performance

Table 6: Comparison of Entropy of a P-frame with Different Wavelet Filters for Algorithm C–I.

| **Entropy** | $S_8$ | $W_8^1$ | $W_8^2$ | $W_8^3$ | $W_4^1$ | $W_4^2$ | $W_4^3$ | $W_2^1$ | $W_2^2$ | $W_2^3$ | Av. |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Daub-4 | 2.84 | 3.63 | 1.97 | 1.73 | 2.18 | 0.39 | 0.31 | 0.75 | 0.04 | 0.01 | 0.54 |
| Daub-6 | 2.95 | 3.73 | 2.02 | 1.72 | 2.27 | 0.43 | 0.31 | 0.83 | 0.04 | 0.01 | 0.57 |
| Daub-8 | 2.93 | 4.08 | 2.13 | 1.86 | 2.51 | 0.42 | 0.32 | 0.91 | 0.03 | 0.01 | 0.61 |
| Daub-8S | 2.90 | 3.61 | 1.94 | 1.60 | 2.30 | 0.37 | 0.30 | 0.81 | 0.04 | 0.00 | 0.55 |
| Burt | 2.83 | 3.46 | 1.93 | 1.61 | 2.20 | 0.40 | 0.28 | 0.80 | 0.05 | 0.01 | 0.55 |
| Haar | 2.74 | 3.42 | 2.10 | 1.90 | 1.94 | 0.55 | 0.38 | 0.64 | 0.06 | 0.03 | 0.52 |
| Spline | 3.34 | 4.39 | 2.53 | 2.20 | 2.35 | 0.49 | 0.33 | 0.67 | 0.04 | 0.00 | 0.57 |
| Vspline | 2.83 | 3.61 | 1.82 | 1.49 | 2.22 | 0.34 | 0.27 | 0.76 | 0.03 | 0.00 | 0.53 |
| Coiflet-6 | 2.84 | 3.55 | 1.98 | 1.65 | 2.20 | 0.41 | 0.31 | 0.79 | 0.04 | 0.01 | 0.55 |
| Vetterli | 2.56 | 3.54 | 2.13 | 1.86 | 2.10 | 0.49 | 0.40 | 0.97 | 0.07 | 0.05 | 0.62 |

Table 7: Comparison of Entropy of a B-frame with Different Wavelet Filters for Algorithm C–I.

| **Entropy** | $S_8$ | $W_8^1$ | $W_8^2$ | $W_8^3$ | $W_4^1$ | $W_4^2$ | $W_4^3$ | $W_2^1$ | $W_2^2$ | $W_2^3$ | Av. |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Daub-4 | 1.35 | 1.82 | 1.18 | 1.01 | 0.93 | 0.31 | 0.22 | 0.29 | 0.03 | 0.01 | 0.26 |
| Daub-6 | 1.37 | 2.02 | 1.29 | 1.06 | 1.14 | 0.36 | 0.22 | 0.37 | 0.03 | 0.01 | 0.30 |
| Daub-8 | 1.42 | 2.18 | 1.37 | 1.17 | 1.25 | 0.30 | 0.22 | 0.41 | 0.02 | 0.00 | 0.31 |
| Daub-8S | 1.40 | 2.01 | 1.20 | 1.05 | 1.14 | 0.30 | 0.22 | 0.37 | 0.03 | 0.00 | 0.29 |
| Burt | 1.30 | 1.76 | 1.22 | 1.00 | 1.07 | 0.31 | 0.23 | 0.37 | 0.05 | 0.01 | 0.29 |
| Haar | 1.30 | 1.66 | 1.12 | 1.03 | 0.78 | 0.30 | 0.26 | 0.24 | 0.04 | 0.02 | 0.24 |
| Spline | 1.90 | 2.72 | 1.77 | 1.54 | 1.22 | 0.36 | 0.23 | 0.31 | 0.02 | 0.00 | 0.32 |
| Vspline | 1.33 | 2.03 | 1.14 | 0.95 | 1.12 | 0.27 | 0.21 | 0.35 | 0.02 | 0.00 | 0.28 |
| Coiflet-6 | 1.39 | 1.72 | 1.17 | 0.97 | 1.01 | 0.28 | 0.22 | 0.32 | 0.03 | 0.01 | 0.27 |
| Vetterli | 1.09 | 1.69 | 1.12 | 0.98 | 0.95 | 0.31 | 0.25 | 0.43 | 0.05 | 0.03 | 0.30 |

of the predicted frames is thus more important in terms of overall performance. Table 6 shows the average first-order entropy of the ten displaced residual wavelets of a P-frame for the different wavelet filter-banks discussed above for a frame from the "car" sequence. The results shown are for Algorithm C–I, which estimates the motion for the top layer only and uses the same values for all the other bands after proper scaling. We observe that Vetterli's filter has the lowest energy in $S_8$ , but its energy in all other subbands is significantly higher. Vspline and Haar have the lowest of the energies in almost all the bands, followed closely by Daubechies' 4-coefficient filter.

In this set of simulation results, the frame ordering is ... BPBPBPB ...; that is, every P-frame or a B-frame is followed by a B-frame or a P-frame, respectively, with a refresh cycle of 40 frames. Given such a framing sequence, there are as many B-frames as there are P-frames. It is therefore necessary to analyze the entropy of the B-frames along with that of the P-frames. Table 7 shows average entropy of DRWs for a typical B-frame. It can be seen that the same two filters Harr and Spline, with almost equal-length filters, are the winners closely followed by Daubechies' 4-tap filter. Once again notice that, in this case also, Vetterli's filter has the lowest energy in $S_8$ and $W_8^2$, but the values in other subbands, especially in the lowest layer, are too high, making it overall one of the worst-performing filters.

The results of the simulations show that the energy compaction, which is the

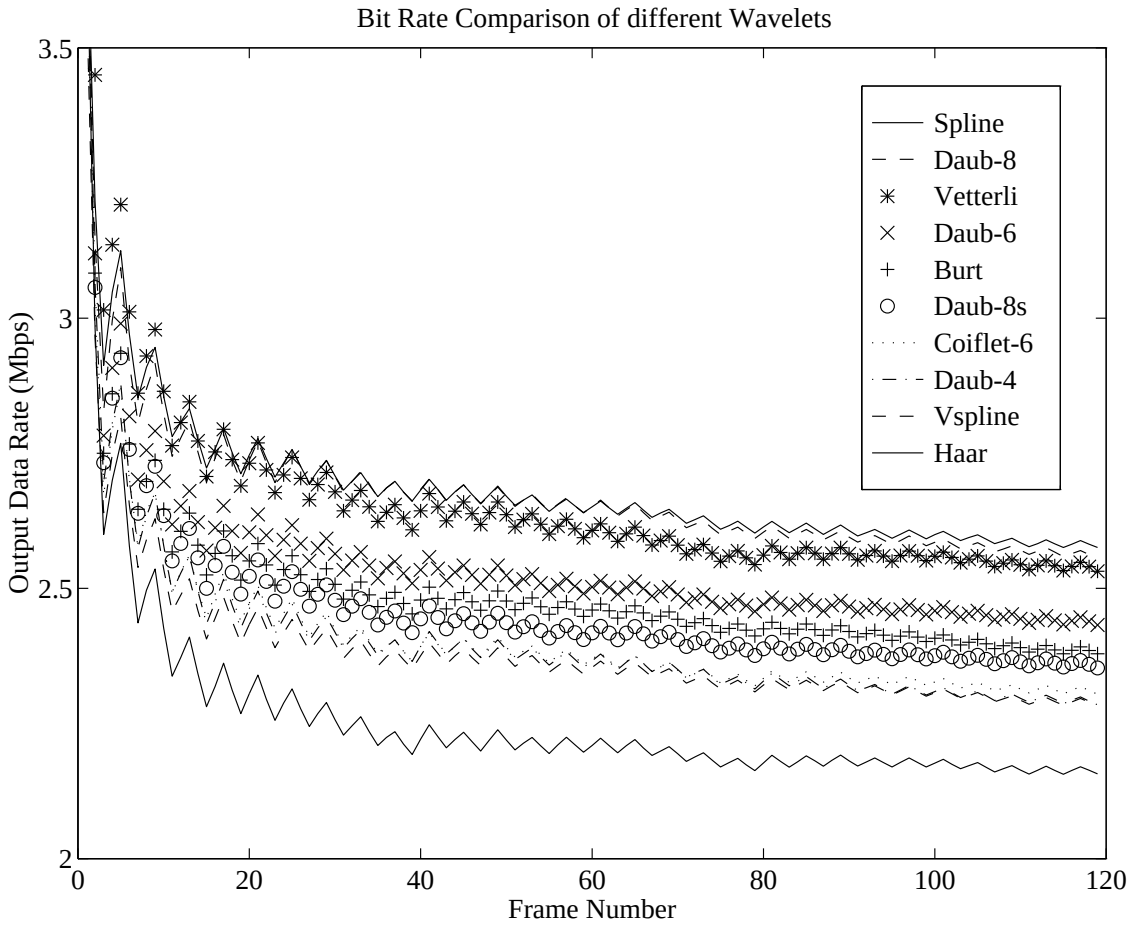



Figure 16: Rate Comparison of Different Wavelets

most desired property of a wavelet filter-bank in case of still images, is not the only attribute required for good overall performance of a codec based on multiresolution compensation. The correlation among the different subband seems to be an important aspect as well, explaining why Haar wavelets have such a performance. In order to make the multiresolution compensation perform well, there should be more correlation between $S_8$ and other bands.

Figure 16 shows the average bit rate for the "car" sequence over 120 frames. The curves show the time average of the total instantaneous bit rate from each frame, which includes the motion overhead also. The Harr wavelet has the least average bit rate of 2.16 Mbps followed by Daubechies' 4-coefficient and Vspline, both of which have almost the same rate at about 2.28 Mbps. The highest rates are generated by Spline at 2.57 Mbps, followed by Daubechies' 8-tap and Vetterli's, both generating an average rate above 2.5 Mbps. The bit rate generated by all the other wavelet

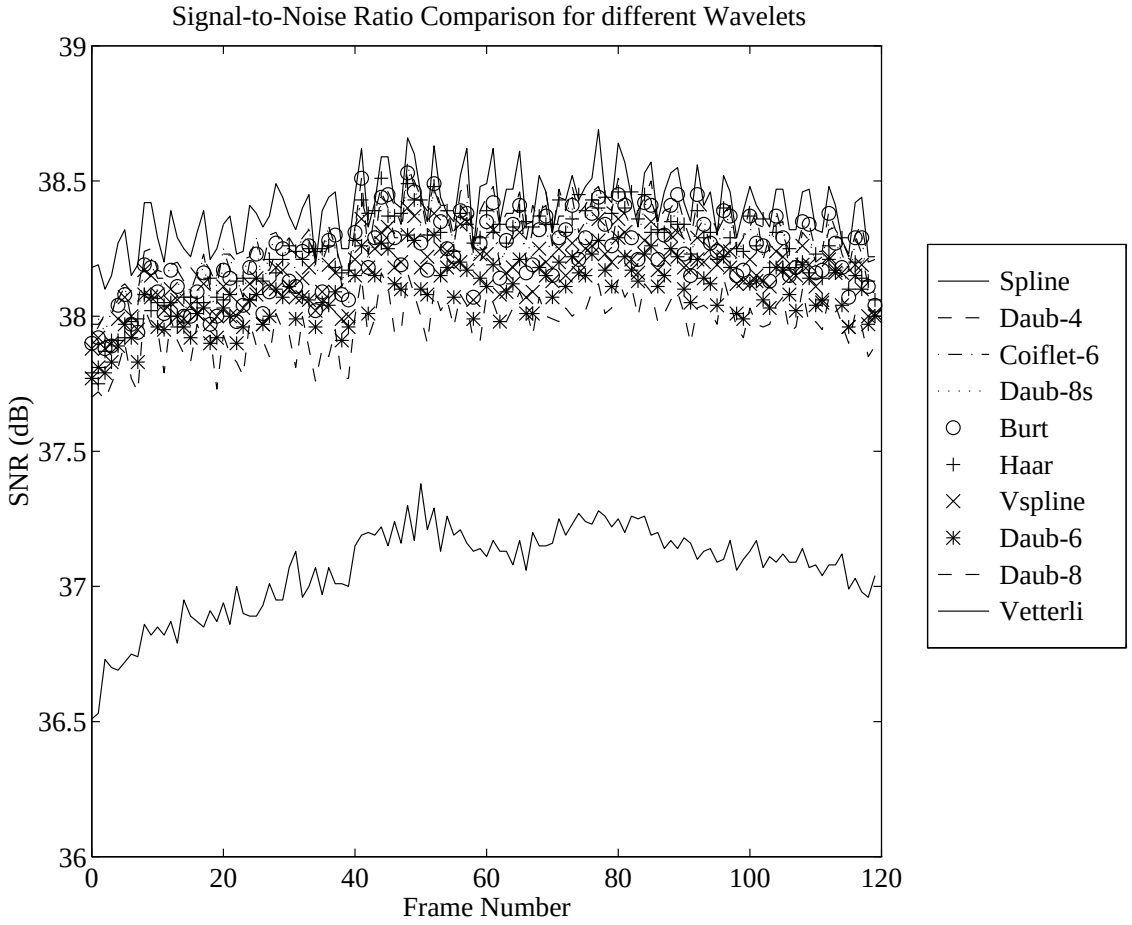



Figure 17: Signal-to-Noise Ratio Comparison of Different Wavelets

filters fall between Harr and Spline. The difference between these two extremes is almost 0.4 Mbps which is quite significant when the corresponding reconstructed *SNR* is considered.

Figure 17 shows a comparison of the reconstructed *SNR* for the different wavelets. Except for Vetterli's filter, which has significantly lower value of less than 37 dB, all the other filters have almost the same quality. The average *SNR* is approximately 38.2 dB varying within $\pm 0.2$ dB. It is impossible to set an exact figure for either the bit rate or the *SNR* over the entire length of the sequence because it depends almost completely on the contents of the scene. The higher the motion activity, the higher is the amount of information needed to transmit and thus the higher is the rate. But note that a higher rate does not necessarily mean a higher *SNR* because it depends on the amount of information that is associated with that particular frame.

Therefore, the *SNR* will be high if a small amount of information is sent out with more detail, as compared with more information and less detail.

It can therefore be inferred from the results that, in general, a filter that compacts less energy to the lowpass subband performs better (as in the case of Harr wavelet). A more important criterion, especially in case of Algorithm C–I, is the correlation of motion activity among different bands, since the lower layers, being larger in size, have more impact on the bit rate. But by far the most important criterion is the overall variance of the coefficients, not in any particular wavelet but in all the subbands. The Spline filter with least dissimilar length filter is a very clear example of the overall low variance. But Daubechies' 4-coefficient filter can be taken as a good candidate because of its small filter length (Higher length filters such as the Daubechies' 6-tap or the 8-tap tend to perform worse than the 4-coefficient filter).

# VI Concluding Remarks

Bidirectional motion compensation combined with block classification has been applied to the wavelet-based video codec we previously developed. The multiresolution motion compensation techniques introduced in [24, 25] capitalize on the correlation present among the different layers of the pyramid structure of the hierarchically represented video. Although the multiresolution motion estimation scheme doesn't necessarily provide more accurate motion vectors than motion estimation done in full resolution, it matches well to the nature of global wavelet decompostion. As a result, the overall performance of the proposed scheme provides better performance with much reduced computation and search complexity.

The frame classification and the mode decision process is similar to that described in the MPEG 2 TM5. The concept of F-frames and the introduction of the block classification algorithm both enhance the performance of the multiresolution motion-compensated video codec. The classification method is especially suited for bidirectional motion compensated frames to minimize the amount of information to be transmitted; the algorithm selects the minimum energy block from the current, previous, and future frame to reduce the bit rate and improve the *SNR* at the same time.

Different wavelet filterbanks are tested on the proposed schemes to identify the properties and relative performance of these filters. It has been shown that wavelet filterbanks which have more correlation among the different subbands perform much better than those which concentrate more of the energy to the top layer. It has also been observed that filters with dissimilar-lengths perform worse than filters with less dissimilar-length filters for the analysis and synthesis.

# References


[1] M. Antonini, M. Barloud, P. Mathieu, and I. Doubechies, "Image Coding Using Wavelet Transform," *IEEE Trans. on Image Processing*, vol. 1, pp. 205 – 220, April 1992.

[2] M. Bierling, "Displacement Estimation by Hierarchical Block Matching," *SPIE Visual Communications and Image Processing*, vol. 1001, pp. 942 – 951, November 1988.

[3] R. J. Clarke and Y. Wang, "Multiresolution Motion Estimation Scheme for Very Low Bit Rate Video Coding," *Very Low Bit Rate Video Workshop, Colchester, UK*, vol. 1, pp. 671 – 681, May 25 – 29, 1994.

[4] R. R. Coifman and M. V. Wickerhauser, "Entropy-based algorithms for best basis selection," *IEEE Tran. on Information Theory*, vol. 38, pp. 713 – 718, 1992.

[5] I. Daubechies, *Ten Lectures on Wavelets*. SIAM, 1992.

[6] H. Gharavi, "Subband Coding Algorithms for Video Applications: Videophone to HDTV Conferencing," *IEEE Trans. Circuits and Systems for Video Tech.*, vol. 1, pp. 174 – 183, June 1991.

[7] K. Goh, J. Soragham, and T. Durrani, "Multiresolution Based Algorithms for Low Bit Rate Image Coding," *IEEE Int. Conf. Image Processing, Austin, TX*, vol. 3, pp. 285 – 289, November 13 – 16, 1994.

[8] J.Joen and J.Kim, "On the Hierarchical Edge-Based Block Motion Estimation for Video Subband Coding at Low Bit Rates," *SPIE Visual Communications and Image Processing, Boston, MA*, vol. 2094, November 9 – 13, 1993.

[9] W. Li and Y.-Q. Zhang, "Vector-Based Signal Processing and Quantization for Image and Video Compression," *Proceedings of IEEE*, vol. 83, pp. 671 – 681, February 1995.

[10] T. Naveen and J. W. Woods, "Motion Compensated Multiresolution Transmission of High Defintion Video," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 4, pp. 29 – 41, February 1994.

[11] A. Netravali and B. Haskell, *Digital Pictures – Representation and Compression*. New York: Plenum Press, 1989.

[12] H. Oh, Y. Baek, G. Kim, G. Park, H. Lee, and J. Jeon, "Very Low Bitrate Video Coding Wavelet Decomposition," *ISO/IEC JTC1/SC29/WG11, MPEG94/392, Singapore*, pp. 1 – 10, November 25 – 29, 1994.

[13] S. Panchanathan, E. Chan, and X. Wang, "Fast Multiresolution Motion Estimation Scheme for a Wavelet Transform Video Coder," *SPIE Visual Communications and Image Processing,(Also IEEE Trans. Image Processing), Chicago, IL*, vol. 2308, pp. 671 – 681, September 25 – 29, 1994.

[14] O. Rioul, "Simple regularity criteria for subdivision schemes," *SIAM J. of Math. Anal.*, vol. 23, pp. 1544 – 1576, 1992.

[15] J. Shapiro, "Embedded Image Coding Using Zerotrees of Wavelet Coefficients," *IEEE Trans. Signal Processing*, vol. 41, pp. 3445 – 3463, December 1993.

[16] P. Sriram and M. Marcellin, "Image Coding using Wavelet Transforms and Entropy Constrained Trellis Coded Quantization," *to appear: IEEE Trans. on Image Processing*, vol. 4, June !995.

[17] A. H. Tewfik, D. Sinha, and P. Jorgensen, "On the Optimal Choice of a Wavelet for Signal Representation," *IEEE Tran. on Information Theory*, vol. 38, pp. 747 – 765, 1992.

[18] K. Uz, M. Vetterli, and D. LeGall, "Interpolative Multi-resolution Coding of Advanced Television and Compatible Subchannels," *IEEE Trans. on Circuit and Systems for Video Technology*, vol. 1, March 1991.

[19] M. Vetterli and C. Herley, "Wavelets and Filter Banks: Relationships and New Results," *Proc. of ICASSP'90*, April 3 – 6, 1990.

[20] M. Vetterli and C. Herley, "Wavelets and Filter banks: Theory and design," *IEEE Trans. on Signal Processing*, vol. 40, pp. 2207 – 2232, 1992.

[21] J. D. Villasenor, B. Belzer, and J. Liao, "Filter Evaluation and Selection in Wavelet Image Compression," in *Proceedings Data Compression Conference* (J. A. Storer and M. Cohn, eds.), pp. 351 – 360, IEEE Computer Society Technical Committee on Computer Communications, March 29 - 31, Snowbird Utah 1994.

[22] S. Zafar, *Motion Estimation and Encoding Algorithms for Hierarchical Representation of Digital Video.* PhD thesis, George Mason University, 1994.

[23] S. Zafar, Y. Zhang, and B. Jabbari, "Multiscale Video Representation Using Multi-Resolution Motion Compensation and Wavelet Decomposition," *IEEE Journal of Selected Areas in Communications*, January 1993.

[24] Y. Zhang and S. Zafar, "Motion-Compensated Wavelet Transform Coding for Color Video Compression," *SPIE Visual Communications and Image Processing '91: Visual Communication*, vol. 1605, pp. 301 – 316, November 1991.

[25] Y. Zhang and S. Zafar, "Motion-Compensated Wavelet Transform Coding for Color Video Compression," *IEEE Trans. on Circuit and Systems for Video Technology*, vol. 2, pp. 285 – 296, September 1992.