Multiresolution Motion Compensation in the Wavelet Domain for Scalable Video Coding

Jong-Tae Kim*, Chang-Mo Yang, Dong-Keun Lim and Yo-Sung Ho
Dept. of Information and Communications
Kwang-ju Institute of Science and Technology (K-JIST)

ABSTRACT

Wavelet transform is a popular tool for image and video coding. It has several advantages in multiresolution analysis and subband decomposition of images. Motion estimation and motion compensation methods are widely used to reduce temporal redundancy in video sequences. Although there have been several attempts to compensate for the motion in the wavelet domain, their performances are limited due to the shift-variant problem. In this paper, we propose a new motion compensation method in the wavelet domain to overcome the shift-variant problem. Experimental results show that the proposed method outperforms the previous motion compensation methods in the wavelet domain as well as the full search algorithm in the spatial domain.

Keywords: Video Compression, Motion Estimation, Multiresolution Motion Estimation, Wavelet

1. INTRODUCTION

Recent explosion of multimedia services has motivated a considerable interest in development of video coding algorithms for transmission or storage of video sequences over band-limited channels. Transmitting digital video over the Internet or wireless networks encounters two major problems: bandwidth fluctuation and packet loss. A scalable video coding scheme is a good solution for these problems.

Scalability is an important research topic in video compression that has recently received a lot of attention. Its applications include video on asynchronous transfer mode (ATM) networks, interworking of video standards, video service hierarchies with multiple spatial resolutions, HDTV with embedded digital TV, and so on. In the scalable video coding system, decoders of various complexities can decode and display appropriate reproductions of the compressed video from the same bitstream.

Generally speaking, scalability is archived at the expense of implementation complexity, i.e., a scalable video coder is likely to be more complex relative to a single layer coder. We can consider three different types of scalability: quality (SNR), spatial, and temporal scalabilities. In this work, we focus mainly on the SNR scalability, since it attracts more attention than the other two.

The MPEG-2 and MPEG-4 standards improve the previous MPEG-1 standard by providing extra features, such as interlaced video and scalable syntax. MPEG-2 and MPEG-4 archive their scalability via prediction. In other words, when they transmit higher resolution versions of the current image frame, two prediction methods are used. One is obtained by spatially interpolating the decoded lower resolution image of the current frame, and the other is obtained by temporally compensating for the higher resolution image of the predicted frame with motion information. The two prediction methods can be combined for a better prediction. Although the multiscale prediction approach used in MPEG offers good flexibility for scalable coding, it does not achieve good coding efficiency since the two resolutions of the image are coded independently.

Wavelet transform generates an inherent multiscale representation for video signals. Efficient image compression schemes based on embedded zerotree wavelet (EZW) and set partitioning in hierarchical trees (SPIHT) have recently been proposed by Shapiro1 and Pearlman,2 respectively. These algorithms provide excellent rate-distortion performance and ideal for progressive transmission because the coded bitstream is generated

* jtkim@kijst.ac.kr; Phone: +82-62-970-2258; Fax: +82-62-970-2204; http://www.kijst.ac.kr; Dept. of Information and Communications, K-JIST, 1 Oryong-dong, Puk-gu, Kwangju, 500-712, Korea.

Visual Communications and Image Processing 2002,
C.-C. Jay Kuo, Editor, Proceedings of SPIE Vol. 4671 (2002) © 2002 SPIE • 0277-786X/02/$15.00

1103
in the order of significance. By extending the SPIHT algorithm to video coding, Taubman and Zakhor\textsuperscript{3} have proposed a 3-D wavelet decomposition of video sequences by treating the temporal direction as the third dimension. Their layered zerotree coding (LZC) leads to multirate and multiscale video compression. However, since no motion estimation and motion compensation was used in their scheme, its rate-distortion performance is not much better than that of MPEG. Besides, temporal scalability is not used since the lower-scale temporal image is blurred by object movement. In addition, implementation of the 3-D wavelet coding scheme is complex since it requires a large amount of memory to store the entire set of 3-D images for forward and inverse wavelet transforms. The intrinsic long coding delay also makes the 3-D wavelet coding less attractive in real-time video applications.

In this paper, we modify the SPIHT algorithm for video coding by adding motion compensation. The coded bitstream consists of the motion field and the residual error. We have used embedded wavelet coding to encode the motion field and the residual error efficiently at each resolution scale.

This paper is organized as follows. In Section 2, we review conventional motion compensation methods both in the spatial and in the wavelet transform domains. In Section 3, we discuss the shift-variant problem and propose an algorithm to overcome the problem. After we discuss simulation results in Section 4, we make conclusions in Section 5.

2. MOTION COMPENSATION IN THE WAVELET DOMAIN

2.1. Wavelet Transform and Multiresolution Decomposition

One-dimensional (1-D) signals can be decomposed into subband signals using the analysis filterbank and the decimation operation. We can decompose each subband into smaller subbands iteratively. Fig. 1 shows two-level wavelet decomposition of images.

![Wavelet Decomposition Diagram](image)

Figure 1: Wavelet Decomposition

For analysis and synthesis, we use Daubechies' 9/7-tap wavelet filters which are defined by their filter coefficients.

\[ H_0 = \{ 0.037828, -0.023849, -0.110624, 0.377402, 0.852698, \\
0.377402, -0.110624, -0.023849, 0.037828 \} \]  

(1)

\[ G_0 = \{ -0.064538, -0.040689, 0.418092, 0.788485, \\
0.418092, -0.040689, -0.064538 \} \]  

(2)
Two-dimensional (2-D) discrete wavelet transform (DWT) of images can be implemented by 1-D filtering both in the x and the y directions using the property of separability. The M-level 2-D DWT of an image \( f(x, y) \) can be expressed as a sequence of subbands:

\[
\left\{ S_{2^M}, \left[ W_{2^j}^V \right]_{j=V,H,D}, \ldots, \left[ W_{2^j}^M \right]_{j=V,H,D} \right\}
\]  

(3)

where \( \{ S_{2^K}, K = 0, 1, 2, \ldots, M \} \) shows a set of approximation of \( f(x, y) \) in the resolution of \( \{ 2^{-1}, 2^{-2}, \ldots, 2^{-M} \} \). Obviously, \( S_0 \) is the original image, and \( S_2 \) is the approximation of \( f(x, y) \) at the next lower resolution of \( 2^{-1} \). \( W_{2^j}^V \) is the detailed image in the resolution of \( 2^{-j} \) at location \( j \), where \( V, H, \) and \( D \) indicate vertical, horizontal, and diagonal locations, respectively.

\( W_{2^M} \) can be derived from the difference of information between two approximations of an image in the resolution of \( 2^{-M} \) and \( 2^{-(M-1)} \).

\[
W_{2^M} = S_{2^M} - S_{2^{M-1}}
\]

(4)

If we apply the 2-D wavelet transform by separable filtering, \( W_{2^M} \) is decomposed into \( W_{2^M}^V \), \( W_{2^M}^H \), and \( W_{2^M}^D \).

### 2.2. Block-Matching Motion Estimation

Motion estimation and motion compensation techniques are commonly employed in video coding standards to remove temporal redundancy existing in video sequences. Motion estimation techniques for video coding can be categorized into two classes: block-matching algorithms and pel-recursive algorithms. Pel-recursive algorithms have been rarely used because they are inherently complex and sometimes run into convergence problems. In the block-matching algorithm, an image is partitioned into blocks and the same displacement vector is assigned to all pixels within the block. In this motion model, we assume that the image is composed of rigid objects with translation motion only. Although the model is clearly restrictive, it is justified by the fact that the complex motion can be approximated by a sum of simple translational motion components.

In the block-matching algorithm, motion of each block with the dimension of \((M \times N)\) pixels is estimated. The range of the motion vector is constrained by the search window. The block-matching algorithm ignores rotational motion and assumes that all pixels within the block have the same uniform displacement. Under the assumption, we can estimate a motion vector by finding the least distortion between the \((M \times N)\) block in the current frame and its corresponding block in the previous frame within a search window of size \([(M + 2m_2) \times (N + 2n_1)]\), as shown in Fig. 2. In case of fast motion or scene change, motion estimation may not be effective. Also, the pixels in the \((M \times N)\) block can conceivably be moved in different directions. A control mechanism such as the sum of motion compensation prediction errors can be introduced to investigate effectiveness of the block matching algorithm. The distortion between the block in the current image and the displaced block in the previous image within the search window can be defined in many different ways. One of the most popular matching criteria is the mean absolute difference (MAD), which is defined by

\[
MAD(i, j) = \frac{1}{MN} \sum_{m=1}^{M} \sum_{n=1}^{N} |X_{m,n} - X_{m+i,n+j}^R|
\]

(5)

### 2.3. Multiresolution Motion Estimation

Fig. 3 shows a video coder with multiresolution motion estimation and motion compensation (MRME/MRMC). The video coder consists of three parts: wavelet transform, quantization, and motion estimation. Each frame is decomposed by three stages of the wavelet transform. In the MRME scheme, Level-3 (the lowest resolution), Level-2 and Level-1 (the highest resolution) images are divided into smaller blocks nominally with the dimension of \( n \times n \), \( 2n \times 2n \), and \( 4n \times 4n \) pixels, respectively. Motion vectors at the highest subband are estimated by the conventional block matching motion estimation. Motion vectors at the next level are then predicted from the motion vector of the preceding level and they are refined at each level.

Although Zhang has proposed several techniques for motion estimation, we only consider the \( S_0/W_3 + \text{refine} \) technique for motion estimation since it provides superior performance over the others. Fig. 4 shows a schematic
Figure 2: Block-Matching Algorithm

Figure 3: Wavelet-Based Video Coder
decomposition of the multiresolution motion estimation technique. The motion vectors of $W^V_4$, $W^H_4$, and $W^D_4$ are predicted from the motion vectors of $W^V_8$, $W^H_8$, and $W^D_8$, respectively, using the following equation.

$$V^p_4(x, y) = 2V^p_8(x, y) + \Delta^p_k(x, y), \quad p \in \{V, H, D\}$$

(6)

where $V^p_4(x, y)$ represents the motion vector of the reference block centered at $(x, y)$ for the $p$-direction subband for the various level $k$. Similarly, the motion vectors of $W^V_2$, $W^H_2$, and $W^D_2$ are predicted as follows:

$$V^p_2(x, y) = 4V^p_8(x, y) + \Delta^p_2(x, y), \quad p \in \{V, H, D\}$$

(7)

The hierarchical motion prediction and refinement operations significantly reduce computational complexity relative to the full-search block-matching method, while providing good performance of motion compensation.

### 3. THE PROPOSED MRME/MRMC ALGORITHM

In this section, we describe details of the new multiresolution motion estimation and motion compensation algorithm in the wavelet domain.

#### 3.1. Multiresolution Motion Estimation (MRME)

In this work, we decompose a video frame in three levels and obtain 10 subbands. It is well known that human vision is more perceptible to errors in lower frequencies than those incurred in higher subbands. The subband $S_8$ has only $1/64$ pixels of the original size, but contains a large percentage of the total energy. In the multiresolution motion estimation algorithm, motion vectors are first estimated for the lowest frequency subband, $S_8$, $W^V_8$, $W^H_8$, and $W^D_8$. Then, motion vectors in lower frequency subbands are refined using the motion information obtained in higher frequency subbands. In this process, errors in higher subbands can be propagated and expanded to all subsequent lower subbands. We consider this point in our proposed method.

In conventional methods, they have used variable block-size MRME by considering human vision characteristics. Smaller blocks are used in lower frequency subbands and larger blocks are used in higher frequency subbands. Therefore, the number of motion vectors in each subband are identical. Similarly, in our proposed algorithm, the block sizes in $S_8$ and $W_8$ subbands are $2 \times 2$, while those in $W_4$ and $W_2$ bands are $4 \times 4$ and $8 \times 8$, respectively. The number of blocks are $(\text{width}/16) \times (\text{height}/16)$ for each subband. Therefore, one pixel shift in the $S_8$ corresponds to eight pixel shift in the original domain. In the lowest subband, motion vectors...
are represented \([-2, 2]\) pixels. Using the subpixel motion estimation method, we can increase the accuracy of motion compensation in each frequency subband. In our proposed method, we use 1/2-pixel motion estimation in subbands \(S_8, W^V_8, W^H_8\), and \(W^D_8\).

Conventional multiresolution motion estimation methods\(^6,7\) exploit motion dependency between lower and higher subbands. Inaccurate motion vectors in lower frequency bands can propagate errors to higher bands. Therefore, we focus on accurate motion estimation in \(S_8, W^V_8, W^H_8\), and \(W^D_8\) subbands.

We first decompose the current and the previous image frames into ten subbands each by three-level wavelet decomposition. For \(S_8\) and three \(W_8\) subbands, we insert average values between pixels for the horizontal and the vertical directions. Then, we can obtain \(4 \times 4\) pixels where the size of the original block is \(2 \times 2\). The proposed method interpolates the previous and the current frames, while conventional methods\(^5,7\) interpolate only the previous frame.

For two interpolated subbands, we adopt the conventional block matching algorithm where the search area is \(\pm 4\) pixels. We find motion vectors using the mean absolute difference (MAD) as a matching criterion. Motion vectors in \(S_8\) subbands are estimated by

\[
V(S_8) = NME[f_n(S_8), f_{n-1}(S_8)]
\]

where \(NME\) stands for the new motion estimation operation.

The motion vectors in \(W_8\) subbands are also estimated by

\[
V(W_8)_{j=V,H,D} = NME[f_n(W_8^j), f_{n-1}(W_8^j)]
\]

For the other subbands, we predict initial motion vectors by scaling the upper-level motion vectors. Because motion activities in higher level are highly related to lower-level motions, we exploit the parent-child relationship of wavelet coefficients. We can represent this operation by

\[
E \left[ V_{2^j}^l(W_{2^j}) \right]_{i=1,2}^{j=V,H,D} = 2^{M-1-i}V_{l}^{j} \left( W_{2^j} \right)
\]

where \(i\) indicates the decomposition level, and \(j\) indicates the position of subband.

The next step is the motion vector refining process that is illustrated in Fig. 5. For neighboring pixels of the estimated location, we can find motion vectors in the small area by

\[
\Delta V_{2^j}^l \left( W_{2^j} \right) = \left\{ (dx, dy) | \min \left[ \sum_{p=-\Omega}^{\Omega} \sum_{q=-\Omega}^{\Omega} |Prev(x + p, y + q) - Curr(x, y)| \right] \right\}_{i=1,2}^{j=V,H,D}
\]

We can also get \(V_{2^j}^l(x, y)\) by

\[
V_{2^j}^l \left( W_{2^j} \right)_{j=V,H,D} = E \left[ V_{2^j}^l \left( W_{2^j} \right) \right] + \Delta V_{2^j}^l \left( W_{2^j} \right)
\]

### 3.2. Multiresolution Motion Compensation (MRMC)

The wavelet transform provides multiresolution/multifrequency expression of a signal with localization in both time and frequency domains. However, the wavelet transform is highly dependent on the alignment of the signal and the discrete grid chosen for the analysis.\(^8\) In order to perform motion compensation in the wavelet domain, the coefficients of the transformed signal need to be predicted. The main difficulty of multiscale video coding lies in the fact that decimation and expansion operations in the wavelet transform domain are shift-variant.\(^9\)

Fig. 6 explains the shift-variant problem. In Fig. 6(a), there are two 1-D signals, the original and the one-pixel shifted signals. The front signal represents the original signal, and the backside signal is the one-pixel
Figure 5: Motion Vector Refinement

(a) Original and One-Pixel Shifted Signals  (b) Low Band Coefficients  (c) High Band Coefficients

(d) Original and Two-Pixel Shifted Signals  (e) Low Band Coefficients  (f) High Band Coefficients

Figure 6: Shift-Variant Problem
shifted signal whose motion in the spatial domain is +1. Fig. 6(b) and Fig. 6(c) show wavelet coefficients of the original signal and the one pixel shifted signals. As we can see in Fig. 6(b) and Fig. 6(c), low-frequency coefficients have different directions of motion, and high-frequency coefficients have opposite signs. Because of this problem, motion vectors of low-frequency subbands cannot represent those of high frequency subbands accurately.\(^8\) This is called as the shift-variant problem in the wavelet transform.

In order to reduce the errors due to the shift-variant problem, we propose a new motion compensation method. As we stated earlier, the motion of odd pixels in the spatial domain is quite different from the motion in the wavelet domain. However, the motion of even pixels in the spatial domain are similar to motion of low frequency and high frequency coefficients, as shown in Fig. 6(e) and Fig. 6(f).

Therefore, we can change the motion vector of odd pixels in the spatial domain to the motion vector of even pixels by the following equation.

\[
NV\left( W^j_8 \right)_{j=V,H,D} = \frac{V\left( W^j_8 \right)}{2} \quad j=V,H,D
\]  

(13)

This operation can reduce motion compensated residual errors of higher frequency subbands noticeably, as shown in Fig. 6(c) and Fig. 6(f). Then, we can motion compensate with one-pixel precision using these motion vectors.

As illustrated in Fig. 7, the overall motion compensation procedure can be described by Eq. (14).

\[
f_c (x, y) = MChalf [V(S_b)] + \left\{ MC_{one} \left[ NV\left( W^j_8 \right) \right] + MC_{one} \left[ V\left( W^j_2 \right) \right] \right\}_{j=1,2,3} \quad j=V,H,D
\]  

(14)

where \( MC_{one} \) means the motion compensation of the one-pixel resolution, and \( MChalf \) means the motion compensation of the half-pixel resolution.

![Motion Vector Compensation Diagram](image)

Figure 7: Motion Vector Compensation

4. EXPERIMENTAL RESULTS

In order to evaluate the proposed method, we compare performances of the proposed and two conventional methods, the full-search algorithm in the spatial domain and the MRME algorithm in the wavelet domain. We use the FOOTBALL sequence with dimension of 720 x 480 pixels and the COASTGUARD sequence with dimension of 352 x 288 pixels.
Motion estimation is performed by the block matching algorithm with $16 \times 16$ pixel blocks and the search area is within $\pm 16$ pixels in the spatial domain. For the wavelet decomposition of images, we use three-level Daubechies' 9/7-tap biorthogonal filter banks.

Fig. 8 displays the FOOTBALL sequence and its motion-compensated residual errors using the three motion estimation algorithms. As shown in Fig. 8, the proposed method has better subjective image quality because it does not have blocking effects. Comparing Fig. 8(d) to Fig. 8(c), we can see that the proposed method has smaller errors values in high frequency regions by reducing the shift-variant problem.

Figure 8: Motion-Compensated Residual Errors [FOOTBALL]

Fig. 9 and Fig. 11 plot MAD values of residual images for 30 frames of FOOTBALL and COASTGUARD sequences, respectively. Average MAD values of these sequences using different motion estimation algorithms are listed in Table 1, which indicates that the proposed method outperforms the conventional methods.

Table 1: Average MAD Value of Residual Errors

<table>
<thead>
<tr>
<th></th>
<th>Full-Search</th>
<th>MRME</th>
<th>Proposed</th>
</tr>
</thead>
<tbody>
<tr>
<td>FOOTBALL</td>
<td>11.18</td>
<td>9.13</td>
<td>7.81</td>
</tr>
<tr>
<td>COASTGUARD</td>
<td>6.85</td>
<td>8.39</td>
<td>6.55</td>
</tr>
</tbody>
</table>

In Fig. 10 and Fig. 12, we compare coder performances after including the actual encoding operation for motion vectors and residual errors using different motion estimation algorithms. The motion vectors are encoded using DPCM and the adaptive arithmetic coder with a fixed context. We adopt the set partitioning in hierarchical trees (SPIHT) coding algorithm\(^2\) to encode residual errors. As shown in Fig. 10 and Fig. 12, the proposed algorithm provides better performance.

We will compare computation complexity of different motion estimation algorithms. If we assume that the frame size is $N \times M$ with the search area of $\pm n$ in each direction and the desired accuracy is $1/r$-pixel, the
Figure 9: Mean Absolute Differences [FOOTBALL]

Figure 10: Rate-Distortion Curve [FOOTBALL]
Figure 11: Mean Absolute Differences [COASTGUARD]

Figure 12: Rate-Distortion Curve [COASTGUARD]
number of multiplications for each block is \((r^2 - 1)(2n + 1)^2\) for \(r > 1\). Complexity burden of different motion estimation algorithms are listed in Table 2. The complexity of the proposed scheme is lower than the full-search method, but slightly higher than the conventional MRME method.

<table>
<thead>
<tr>
<th>Table 2: Computational Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full-Search</td>
</tr>
<tr>
<td>MRME</td>
</tr>
<tr>
<td>Proposed</td>
</tr>
</tbody>
</table>

5. CONCLUSIONS

In this paper, we have proposed a new multiresolution motion estimation and motion compensation algorithm for wavelet-based video coding. The proposed algorithm can overcome the shift-variant problem in the wavelet domain and perform the motion estimation and motion compensation operations more efficiently. Besides, it has lower computational complexity than the spatial domain full-search and other MRME algorithms. Simulation results demonstrate that the proposed scheme outperforms the full-search method in the spatial domain and the conventional MRME scheme in the wavelet domain in terms of PSNR and MAD values. In addition, the proposed algorithm does not produce the blocking artifacts and is suitable for adaptive scalable video coders.

ACKNOWLEDGEMENTS

This work was supported in part by the Korea Science and Engineering Foundation (KOSEF) through the Ultra-Fast Fiber Optical Network (UFON) Research Center at Kwangju Institute of Science and Technology (K-JIST), and in part by the Ministry of Education (MOE) through the Brain Korea 21 (BK 21) project.

REFERENCES