New Approach to Orthogonal Multiplierless Wavelet Family Synthesis

PIOTR LIPINSKI
Division of Computer Networks
Technical University of Lodz
ul. Stefanowskiego 18/22, 90-924 Lodz
POLAND

MYKHAYLO YATSYMIRSKYY
Department of Computer Science
Technical University of Lodz
ul. Wolczanska 215, 90-924 Lodz
POLAND

Abstract: In this article we introduce a new approach for orthogonal wavelet synthesis. The new approach is more general than Daubechies approach and can be used to synthesize more computationally efficient wavelets for VLSI implementations. In order to synthesize new wavelet family we impose alternative conditions on wavelet filters. The new wavelet family is more computationally efficient than equivalent Daubechies wavelets and gives better results in terms of signal compression.

Key-Words: wavelets, wavelet synthesis, Daubechies wavelets, orthogonal wavelets, orthogonal transforms

1 Introduction
Wavelet transforms are widely used in signal compression and coding and are incorporated in commercial hardware and software products offered by many vendors. Therefore, there is a strong need to develop efficient wavelet transform algorithms [1-5].

Intensive research being conducted on the topic of efficient wavelet filter design and implementation adaptive filter design [6-10]. Here we have shown the approach to synthesize computationally efficient, wavelets.

2 Problem Formulation

Every wavelet transformation of the input signal \( x \) can be found from (1,2):

\[
y' = C x, \quad y = P(y'),
\]

where:
- \( x \) – input signal of length \( N \),
- \( y' \) – wavelet transform of the input signal,
- \( C \) – wavelet transform matrix,
- \( P(x) \) – permutation of vector \( x \).

If we impose the perfect reconstruction condition and we assume that the input signal is periodical the wavelet transform matrix \( C \) takes the form given by (3):

\[
C = \begin{bmatrix}
c_0 & c_1 & \ldots & c_k \\
c_k & -c_{k-1} & \ldots & -c_0 \\
c_k & -c_{k-1} & \ldots & -c_0 \\
\vdots & \vdots & \ddots & \vdots \\
c_k & -c_{k-1} & \ldots & -c_0 \\
c_k & -c_{k-1} & \ldots & -c_0
\end{bmatrix}
\]

In the matrix \( C \) (3) odd rows correspond to low frequency filter in Mallat algorithm and \( c_0, c_1, \ldots, c_k \), are low frequency filter impulse response coefficients while even rows correspond to high frequency coefficients and \( c_k, -c_{k-1}, \ldots, -c_0 \) are high frequency filter impulse response coefficients. The multiplication by the matrix \( C \) is equivalent to filtering and decimation in Mallat’s algorithm [11]. The resultant vector must be yet permuted to obtain low and high frequency outputs in such a way that first \( N/2 \) elements of the \( y \) vector are odd elements of vector \( y' \) and elements from \( N/2+1 \) to \( N \) are even elements of vector \( y' \).

Our aim here is to develop new wavelet family by imposing new condition to wavelet matrix coefficients. The new wavelets have better properties for VLSI implementations in terms of computational complexity than well known wavelets (such as Daubechies) [4,5].
3 Four-tap filter

Let us now consider the four-tap wavelet transform. The transform matrix C takes the following form:

\[
C_{\text{4-tap}} = \begin{bmatrix}
    c_0 & c_1 & c_2 & c_3 \\
    -c_1 & -c_2 & -c_3 & -c_0 \\
    c_0 & c_1 & c_2 & c_3 \\
    -c_1 & -c_2 & -c_3 & -c_0
\end{bmatrix}
\]

(4)

The four-tap wavelet transform is orthogonal when transform matrix coefficients satisfy (4) and (5)

\[
c_0^2 + c_1^2 + c_2^2 + c_3^2 = 1,
\]

(4)

\[
c_2c_0 + c_3c_1 = 0
\]

(5)

To solve this equation we have to impose additional conditions on matrix coefficients. Here we assume that filters must satisfy (6) and (7).

\[
c_0 - c_2 + c_2 - c_3 = c_1, \quad (6)
\]

\[
c_1 = m c_0, \quad (7)
\]

where:

- \(m\) – wavelet parameter

The matrix coefficients \(c_0 - c_3\), can be found from (4-7):

\[
c_0 = \frac{m+1}{\sqrt{2(m^2-1)}}, \quad c_1 = \frac{m(m+1)}{\sqrt{2(m^2-1)}},
\]

\[
c_2 = \frac{m(m-1)}{\sqrt{2(m^2-1)}}, \quad c_3 = \frac{1-m}{\sqrt{2(m^2-1)}}.
\]

Notice, that by substituting \(m= \sqrt{3}\) we obtain Daubechies 4-tap filter, but here, unlike in Daubechies wavelets we are not confined to only one solution. We can adjust the value of the parameter \(m\), which can be used to obtain Daubechies-like rational wavelets. Such rational wavelets have similar properties as Daubechies wavelets but are more computationally efficient.

To derive Daubechies-like rational wavelets we must notice, that all four coefficients \(c_0 - c_3\) have the same denominator, which can be omitted. This leads to the following values of \(c_0 - c_3\) coefficients:

\[
c_0 = m + 1, \quad c_1 = m(m + 1),
\]

\[
c_2 = m - 1, \quad c_3 = 1 - m
\]

When we set \(m=2\) the filter coefficients are given as follows: \(c_0 = 3, c_1 = 6, c_2 = 2, c_3 = -1\). This is a Daubechies-like wavelet, which has very similar shape and properties as Daubechies 4 wavelet, but has absolute transform matrix coefficients, see fig. 1,2.

![Fig. 1 Daubechies-like 4-tap rational wavelet](image1)

![Fig. 2. Daubechies 4-tap irrational wavelet](image2)

4 Six-tap filter

Let us now consider the six-tap wavelet transform. The transform matrix C takes the following form:

\[
C_{\text{6-tap}} = \begin{bmatrix}
    c_0 & c_1 & c_2 & c_3 & c_4 & c_5 \\
    -c_1 & -c_2 & -c_3 & -c_4 & -c_5 & -c_0 \\
    c_0 & c_1 & c_2 & c_3 & c_4 & c_5 \\
    -c_1 & -c_2 & -c_3 & -c_4 & -c_5 & -c_0 \\
    c_0 & c_1 & c_2 & c_3 & c_4 & c_5 \\
    -c_1 & -c_2 & -c_3 & -c_4 & -c_5 & -c_0
\end{bmatrix}
\]

(4)

The four-tap wavelet transform is orthogonal when transform matrix coefficients satisfy (8-10):

\[
c_0^2 + c_1^2 + c_2^2 + c_3^2 + c_4^2 + c_5^2 = 1
\]

(8)

\[
c_3c_0 + c_2c_1 + c_1c_2 + c_0c_3 = 0
\]

(9)

\[
c_2c_0 + c_3c_1 + c_4c_2 + c_5c_3 = 0
\]

(10)

To solve this equation we have to impose additional conditions on matrix coefficients. Here we assume that filters must satisfy (11-13):

\[
c_2 - c_2 - c_2 + c_4 - c_5 = 0
\]

(11)
\[ c_1 = n c_0, \]  
\[ c_1 = n c_3, \]

(12)

(13)

where:

\( m, n \) – wavelet parameters

The solution of the above system of equations (8-13), is given as follows:

\[ c_2 = \frac{(n-1)(n-m)}{\sqrt{2(n^2+1)(n^2+4)}} \]

\[ c_4 = \frac{m(n-1)(n-m)}{\sqrt{2(n^2+1)(n^2+4)}} \]

\[ c_2 = \frac{(m^2+1)(1+n)}{\sqrt{2(n^2+1)(n^2+4)}} \]

\[ c_4 = \frac{m(1+mn)(n-1)}{\sqrt{2(n^2+1)(n^2+4)}} \]

\[ c_0 = \frac{(1-n)(1+mn)}{\sqrt{2(n^2+1)(n^2+4)}} \]

Notice, that by substituting \( m = \sqrt{3} \) we obtain Daubechies 4-tap filter, but here, unlike in Daubechies wavelets we are not confined to only one solution. We can adjust the value of the parameter \( m \), which can be used to obtain Daubechies-like rational wavelets. Such rational wavelets have similar properties as Daubechies wavelets but are more computationally efficient.

To derive Daubechies-like rational wavelets we must notice, that all four coefficients \( c_0 \) – \( c_5 \) have the same denominator, which can be omitted. This leads to the following values of \( c_0 \) – \( c_1 \) coefficients:

\[ c_0 = (n-1)(n-m) \]

\[ c_1 = m(n-1)(n-m) \]

\[ c_2 = (m^2+1)(1+n) \]

\[ c_3 = n(m^2+1)(1+n) \]

\[ c_4 = m(1+mn)(n-1) \]

\[ c_5 = (1-n)(1+mn) \]

When we set \( m = 5/2, n = 1/4 \) the matrix coefficients yield \( c_0 = 220, c_1 = 550, c_2 = 348, c_3 = -78, c_4 = -75, c_5 = -30 \). This is a Daubechies-like wavelet which has very similar properties as Daubechies 6 wavelet, but transform matrix coefficients are rational, see fig. 3,4.

5 Experimental results

Simulations have been performed on a set of test images to evaluate the performance of the proposed new wavelets in comparison with the well known Daubechies wavelets. We have compared the quality of compressed images terms of Signal-to-Noise Ratio (SNR), ability to concentrate energy in low frequencies and Computational Cost per pixel.

The training set consists of a eight well known, test images such that shown in Fig. 1, all of 512 × 512 gray scale with 8-bits resolution. Fig. 5. The table 1 demonstrates the SNR for eight test images at a compression ratio 4. Bold numbers indicate better performance. Analogous results are obtained for compression ratio 8 and 16.

Table 1. Signal to noise ratio for test images. Compression ratio 4.

<table>
<thead>
<tr>
<th>Test image</th>
<th>db4 wavelet</th>
<th>New 4-tap wavelet</th>
<th>Db6 wavelet</th>
<th>New 6-tap wavelet</th>
</tr>
</thead>
<tbody>
<tr>
<td>lena</td>
<td>70.2 dB</td>
<td>70.4 dB</td>
<td>66.8 dB</td>
<td>66.6 dB</td>
</tr>
<tr>
<td>peppers</td>
<td>69.1 dB</td>
<td>69.8 dB</td>
<td>66.5 dB</td>
<td>66.4 dB</td>
</tr>
<tr>
<td>moon</td>
<td>73.4 dB</td>
<td>73.3 dB</td>
<td>70.9 dB</td>
<td>70.7 dB</td>
</tr>
<tr>
<td>Sailboat</td>
<td>67.7 dB</td>
<td>68.3 dB</td>
<td>64.8 dB</td>
<td>65.0 dB</td>
</tr>
<tr>
<td>airplane</td>
<td>68.2 dB</td>
<td>68.5 dB</td>
<td>65.9 dB</td>
<td>66.0 dB</td>
</tr>
<tr>
<td>mandrill</td>
<td>66.1 dB</td>
<td>66.5 dB</td>
<td>65.0 dB</td>
<td>65.0 dB</td>
</tr>
<tr>
<td>boat</td>
<td>68.9 dB</td>
<td>69.0 dB</td>
<td>66.6 dB</td>
<td>66.7 dB</td>
</tr>
<tr>
<td>aerial</td>
<td>63.6 dB</td>
<td>64.4 dB</td>
<td>62.5 dB</td>
<td>62.5 dB</td>
</tr>
</tbody>
</table>
6 Performance analysis

Performance comparison of the conventional and proposed algorithms for calculating 4-tap and 6-tap 1-D wavelets is presented.

Table 2 lists the number of additions and nontrivial multiplications per one output sample for 4-tap, and 6-tap wavelets. The efficient algorithms presented in table 2 are described in detail in [5] and [12]. The implementation of 4-tap and 6-tap wavelet filter coefficients is given in table 3. We also take advantage of the symmetry of wavelet filter: c₁=mc₀, c₃=nc₂, c₄=-mc₅.

Table 2 Number of nontrivial multiplications, and additions

<table>
<thead>
<tr>
<th>Wavelet Type</th>
<th>Nmult</th>
<th>NADD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mallat Daub. 4-tap</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>Mallat Daub. 6-tap</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>Lift Daub. 4-tap</td>
<td>2.5</td>
<td>2</td>
</tr>
<tr>
<td>Lift Daub. 6-tap</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>Efficient Cooklev Daub. 4-tap [5]</td>
<td>2.5</td>
<td>5</td>
</tr>
<tr>
<td>Efficient Cooklev Daub. 6-tap [5]</td>
<td>3.5</td>
<td>7</td>
</tr>
<tr>
<td>Efficient Daub. 4-tap [12]</td>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td>Efficient Daub. 6-tap [12]</td>
<td>2</td>
<td>5.5</td>
</tr>
<tr>
<td>New 4-tap</td>
<td>0</td>
<td>6</td>
</tr>
<tr>
<td>New 6-tap</td>
<td>0</td>
<td>14</td>
</tr>
</tbody>
</table>

Table 3 Filter coefficients in sum of power of two representation.

<table>
<thead>
<tr>
<th>4-tap filter</th>
<th>6-tap filter</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>2+1</td>
</tr>
<tr>
<td>6</td>
<td>4+2</td>
</tr>
<tr>
<td>-1</td>
<td>-1</td>
</tr>
<tr>
<td>220</td>
<td>256-32-4</td>
</tr>
<tr>
<td>550</td>
<td>512+32+4+2</td>
</tr>
<tr>
<td>348</td>
<td>512-128-32-4</td>
</tr>
<tr>
<td>-87</td>
<td>-128+32+8+1</td>
</tr>
<tr>
<td>-75</td>
<td>-64-8-4-1</td>
</tr>
<tr>
<td>-30</td>
<td>-32+2</td>
</tr>
</tbody>
</table>

7 Conclusion

In this paper a new approach to orthogonal wavelet family synthesis has been introduced. The wavelets obtained based on this approach achieve comparable compression results as well known Daubechies wavelets and have similar shape and characteristics. The main advantage of the new wavelet family is that it does not require nontrivial multiplications and is more computationally efficient than Daubechies wavelets. In addition, the wavelet filter coefficients have absolute values, which assures perfect reconstruction of the input signal. The parameters m and n can be also used in a future to adjust the wavelet shape to the characteristics of a transformed signal.

References: