Phase-Shifting for Nonseparable 2-D Haar Wavelets

Mais Alnasser and Hassan Foroosh, *Senior Member, IEEE*

***Abstract*—In this paper, we present a novel and efficient solution to phase-shifting 2-D nonseparable Haar wavelet coefficients. While other methods either modify existing wavelets or introduce new ones to handle the lack of shift-invariance, we derive the explicit relationships between the coefficients of the shifted signal and those of the unshifted one. We then establish their computational complexity, and compare and demonstrate the superior performance of the proposed approach against classical interpolation tools in terms of accumulation of errors under successive shifting.**

***Index Terms*—Nonseparable, phase-shifting, 2-D Haar transform, wavelets.**

I. Introduction and Related Work

In many image processing applications involving wavelets [1]–[12], data may need to be phase shifted. Due to lack of shift-invariance of wavelets, simple operations such as translation and scaling require back and forth transformations [13]. One can achieve shift-invariance by relaxing the orthogonality condition as in [14]. However, nonorthogonality makes the interpretation of the correlation properties among the transform coefficients more difficult. Another approach is to introduce redundancy by avoiding decimation [15]–[19], which unfortunately undermines wavelets' use in compression and coding. A more recent approach achieves shift-invariance through complex wavelets [20]–[23]. Complex wavelets prove to be useful in solving the shift-invariance problem without compromising many other properties. However, they often suffer from lack of speed and poor inversion properties. A more successful attempt in this category is perhaps the dual-tree complex wavelet transform (DT-CWT) [24], [25]. Although DT-CWT provides a good tradeoff between fully decimated wavelets and the redundant complex wavelet transform, it does so by trading off the compression capabilities.

Rather than modifying classical wavelets or designing new ones, our goal is to establish a relationship between the coefficients of a transformed signal and those of its shifted version. Of course, such a relation would be wavelet-dependent and may not be straightforward. The key idea is that as long as the relation is known, one can tackle shift-variance, since all coefficients of a shifted signal can be mapped to those of the original 0-shift signal. On the other hand, shift-variance is tackled without compromising speed and compression properties. Furthermore, establishing explicit relations between the coefficients of a signal and those of its shifted version, would allow us to perform compressed domain processing without requiring a chain of forward and backward transforms. This is particularly of interest in data compression and progressive transmission. Our focus in this paper is on the nonseparable 2-D Haar wavelet transform due to its additional desirable properties, such as symmetry and conservation of image aspect ratio through different compression levels, which is important to applications that require progressive reconstruction.

Manuscript received July 26, 2007; revised December 19, 2007. This work was supported in part by a grant from Electronic Arts—Tiburon. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Pier Luigi Dragotti.

The authors are with the Computational Imaging Lab (CIL), Department of Computer Science, University of Central Florida, Orlando, FL 32816-2362 USA (e-mail: nasserm@cs.ucf.edu; foroosh@cs.ucf.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2008.919950

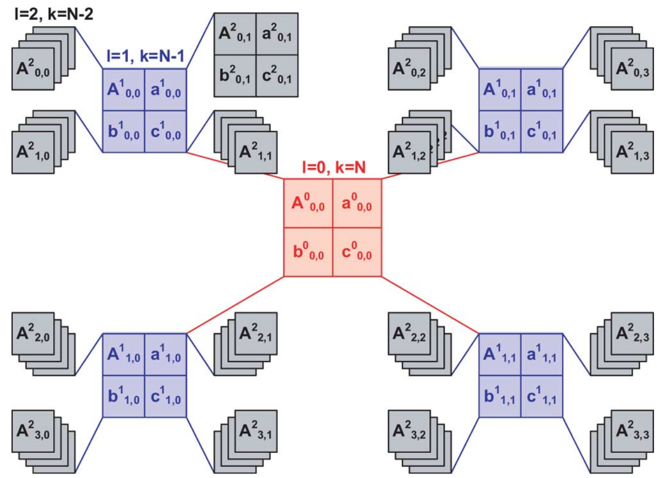


Fig. 1. Haar transform of the 2-D signal $x(m,n)$ is composed of the dc value $A^0_{0,0}$ and the detail coefficients $a^l_{i,j}$, $b^l_{i,j}$ and $c^l_{i,j}$, where $l = 0,\ldots,N-1$, $i = 0,\ldots,2^l-1$, and $j = 0,\ldots,2^l-1$. The blur coefficients $A^l_{i,j}$ at each level are used to help derive the equation for phase-shifting the signal $x(m,n)$, but are not used in the final form of the equation. Only the first three levels of the tree are shown with one l to avoid cluttering the figure.

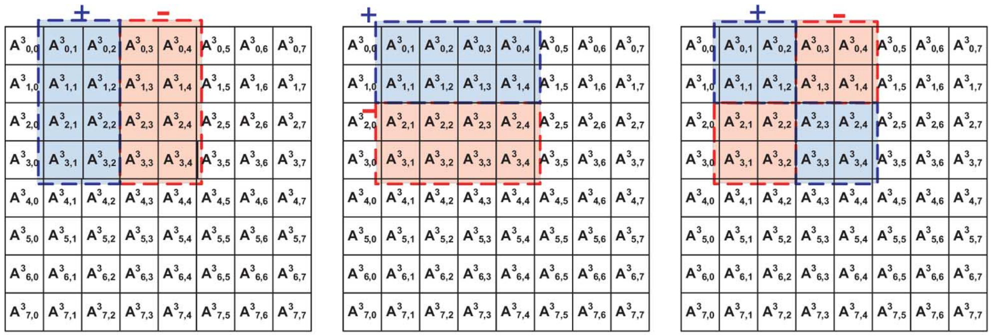
We derive explicit expressions for shifting data directly in the Haar domain using only the available coefficients of the original 0-shift signal. We also show how our solution can be expanded for noninteger phase shifts, and evaluate our approach against popular interpolation methods in terms of accumulation of errors through successive shifts.

II. Notations and Setup

Let $x(n,m)$ be a 2-D signal of size $2^N \times 2^N$, where N is a positive integer. The Haar transform of $x(n,m)$ can be expressed using a tree as in Fig. 1. The tree is constructed of N levels with $x(n,m)$ residing at the leaves, i.e., the Nth level. The ijth node at level l in the tree is made to hold the ijth blur coefficient $A^l_{i,j}$ and the detail coefficients $a^l_{i,j}$, $b^l_{i,j}$, and $c^l_{i,j}$, where

1057-7149/$25.00 © 2008 IEEE

Authorized licensed use limited to: IEEE Xplore. Downloaded on January 21, 2009 at 18:27 from IEEE Xplore. Restrictions apply.



(a) (b) (c)

Fig. 2. Blur coefficients at level $l = 3$. In (a)–(c), respectively, the rectangular windows show the coefficients included in evaluating $a^2_{0,0_{\text{new}}}$, $b^2_{0,0_{\text{new}}}$, and $c^2_{0,0_{\text{new}}}$ under a horizontal odd shift of one. These are evaluated by summing and subtracting the highlighted quadrants in the windows as shown in the diagrams.

$l = 0, \ldots, N-1$, $i = 0, \ldots, 2^l - 1$, and $j = 0, \ldots, 2^l - 1$. We denote $x(i,j)$ as $A^N_{i,j}$ and let $l = 0, \ldots, N$ with $a^N_{i,j} = 0$, $b^N_{i,j} = 0$, and $c^N_{i,j} = 0$. For brevity, we will sometimes refer to the horizontal, vertical and diagonal detail coefficients by a, b, and c, respectively.

Let

$$\begin{aligned} X^l_{i,j} &= a^{l-1}_{\lfloor i/2\rfloor,\lfloor j/2\rfloor} + b^{l-1}_{\lfloor i/2\rfloor,\lfloor j/2\rfloor} + c^{l-1}_{\lfloor i/2\rfloor,\lfloor j/2\rfloor} \\ Y^l_{i,j} &= -a^{l-1}_{\lfloor i/2\rfloor,\lfloor j/2\rfloor} + b^{l-1}_{\lfloor i/2\rfloor,\lfloor j/2\rfloor} - c^{l-1}_{\lfloor i/2\rfloor,\lfloor j/2\rfloor} \\ Z^l_{i,j} &= a^{l-1}_{\lfloor i/2\rfloor,\lfloor j/2\rfloor} - b^{l-1}_{\lfloor i/2\rfloor,\lfloor j/2\rfloor} - c^{l-1}_{\lfloor i/2\rfloor,\lfloor j/2\rfloor} \\ W^l_{i,j} &= -a^{l-1}_{\lfloor i/2\rfloor,\lfloor j/2\rfloor} - b^{l-1}_{\lfloor i/2\rfloor,\lfloor j/2\rfloor} + c^{l-1}_{\lfloor i/2\rfloor,\lfloor j/2\rfloor}. \end{aligned} \tag{1}$$

The following formula shows the relation between the blur coefficient $A^l_{i,j}$ and its parent at level $l-1$:

$$A^l_{i,j} = \begin{cases} A^{l-1}_{i/2,j/2} + X^l_{i/2,j/2}, & i \text{ is even, } j \text{ is even} \\ A^{l-1}_{i/2,\lfloor j/2\rfloor} + Y^l_{i/2,\lfloor j/2\rfloor}, & i \text{ is even, } j \text{ is odd} \\ A^{l-1}_{\lfloor i/2\rfloor,j/2} + Z^l_{\lfloor i/2\rfloor,j/2}, & i \text{ is odd, } j \text{ is even} \\ A^{l-1}_{\lfloor i/2\rfloor,\lfloor j/2\rfloor} + W^l_{\lfloor i/2\rfloor,\lfloor j/2\rfloor}, & i \text{ is odd, } j \text{ is odd.} \end{cases} \tag{2}$$

We let $D^l_{i,j}$ be the difference between $A^0_{0,0}$ and $A^l_{i,j}$, then

$$A^l_{i,j} = A^0_{0,0} + D^l_{i,j}. \tag{3}$$

By substituting (3) into (2), $D^l_{i,j}$ can be computed recursively solely in terms of the detail coefficients as follows:

$$D^l_{i,j} = \begin{cases} D^{l-1}_{i/2,j/2} + X^l_{i/2,j/2}, & i \text{ is even, } j \text{ is even} \\ D^{l-1}_{i/2,\lfloor j/2\rfloor} + Y^l_{i/2,\lfloor j/2\rfloor}, & i \text{ is even, } j \text{ is odd} \\ D^{l-1}_{\lfloor i/2\rfloor,j/2} + Z^l_{\lfloor i/2\rfloor,j/2}, & i \text{ is odd, } j \text{ is even} \\ D^{l-1}_{\lfloor i/2\rfloor,\lfloor j/2\rfloor} + W^l_{\lfloor i/2\rfloor,\lfloor j/2\rfloor}, & i \text{ is odd, } j \text{ is odd} \\ 0, & i = j = l = 0. \end{cases} \tag{4}$$

A 2-D signal can be shifted horizontally or vertically. Both types of shifts affect the $a^l_{i,j}$, $b^l_{i,j}$ and $c^l_{i,j}$ at all levels. At level $N-k$, there are $2^k \times 2^k$ nonredundant coefficient sets each of size $2^{N-k} \times 2^{N-k}$ [26], where $k = 1, \ldots, N$. A horizontal shift $s_h = 0, \ldots, 2^N - 1$ or a vertical shift $s_v = 0, \ldots, 2^N - 1$ can be one of the following possibilities:

- a shift that is divisible by 2^k;
- an odd shift;
- an even shift that is not divisible by 2^k.

We derive the formulae for evaluating the a detail coefficients under a horizontal shift s_h, for each of the three possibilities, followed by similar derivations for b and c detail coefficients. Formulae for vertical shifts can be derived in a similar manner.

III. Horizontal Coefficients for Horizontal Shift

A. *Shifting by a Multiple of* 2^k

This is the simplest case. A shift s_h in the discrete time domain that is equal to $2^k u$ is a horizontal circular shift of the 0-shift detail coefficients at level $N-k$ by u, that is

$$a^{N-k}_{i,j_{\text{new}}} = a^{N-k}_{i,(j+u)\%2^{N-k}}, \quad k = 1, \ldots, N \tag{5}$$

$$a^{N-k}_{i,j_{\text{new}}} = \sum_{m=2^k i}^{2^k(i+1)} \frac{\left(\left(A^N_{m,j_1\%2^N} + \cdots + A^N_{m,(j_2-1)\%2^N}\right) - \left(A^N_{m,j_2\%2^N} + \cdots + A^N_{m,(j_3-1)\%2^N}\right)\right)}{4^k} \tag{6}$$

$$\text{where } j_1 = 2^k j + s_h, \quad j_2 = 2^{k-1}(2j+1) + s_h, \quad j_3 = 2^k(j+1) + s_h. \tag{7}$$

Authorized licensed use limited to: IEEE Xplore. Downloaded on January 21, 2009 at 18:27 from IEEE Xplore. Restrictions apply.

where $0 \leq u \leq 2^{N-k}-1$ and% is the mod operation. Notice that for levels $N-(k-1), N-(k-2), \ldots, N-1$ a horizontal shift of $2^k u$ in the time domain is a horizontal circular shift of the coefficients at those levels by $2u, 2^2u, \ldots, 2^{k-1}u$, respectively. In other words, a horizontal shift of $2^k u$ in the time domain shifts the coefficients at level $N-k$ horizontally by u, while shifting the coefficients at level $N-(k-1)$ horizontally by twice as much, and the coefficients at level $N-(k-2)$ by four times as much, and so on.

B. Shifting by an Odd Amount

By examining the tree in Fig. 1, we notice that [see (6) and (7), shown at the bottom of the previous page]. Equations (6) and (7) evaluate $a^{N-k}_{i,j_{\text{new}}}$ by summing the first half and subtracting the second half of each row of blur coefficients at the leaves level N, which fall inside the window determined by the two corners $(2^k i, 2^k j+s_h)$ and $(2^k(i+1), 2^k(j+1)+s_h)$. Fig. 2(a) illustrates the evaluation of $a^2_{0,0}$ using the blur coefficients at level $l=3$ under a horizontal odd shift of one.

Substituting (3), (4), and then (1) into (7), we get the coefficients of the shifted signal as follows [see (8), shown at the bottom of the page].

Note that at $k=1$, j_2 is a noninteger value. When that is the case we set $a^{N-1}_{i,j_2\%2^{N-1}}$ to 0.

C. Shifting by an Even Amount That is Not Divisible by 2^k

In this case, s_h is divisible by 2^t, where $1 \leq t \leq k-1$ and 2^t is the highest power of 2 by which s_h is divisible. This allows us to let $s_h = 2^t u$, where u is an odd integer in the range $0 \leq u \leq 2^{N-t}-1$. This means that the coefficients at levels $N-1, \ldots, N-t$ are a subset of the first case. In other words, the 0-shift coefficients at levels $N-1, N-2, \ldots, N-t$ are circularly shifted in the horizontal direction by $2^{t-1}u, 2^{t-2}u, \ldots, u$, respectively. On the other hand, $a^{N-k}_{i,j_{\text{new}}}$ is verified to be an odd shift of the blur details at level $N-t$. Therefore, at level $N-k$, $a^{N-k}_{i,j_{\text{new}}}$ can be evaluated using the following modification of (7) [see (9), shown at the bottom of the page].

Following the same steps as above, we get (10), shown at the bottom of the page.

Note that the second case is the same as the third case where $t=0$. This gives rise to the following final formula for evaluating the a detail coefficients under a horizontal shift [see (11), shown at the bottom of the next page].

IV. Vertical Coefficients for Horizontal Shift

For this section and the next one, again we let $s_h = 2^t u$ be a horizontal shift, where $0 \leq s_h \leq 2^N-1$, $0 \leq t \leq N$, 2^t is the highest power of 2 by which s_h is divisible, and $0 \leq u \leq 2^{N-t}-1$ is an odd positive integer. Then for all $k \leq t$

$$b^{N-k}_{i,j_{\text{new}}} = b^{N-k}_{i,(j+s_h/2^k)\%2^{N-k}}. \tag{12}$$

$$a^{N-k}_{i,j_{\text{new}}} = \sum_{m=2^{k-1}i}^{2^{k-1}(i+1)-1} \frac{\left(\begin{aligned} &D^{N-1}_{m,j_1\%2^{N-1}} + 2\sum_{n=j_1+1}^{j_2-1} D^{N-1}_{m,n\%2^{N-1}} - 2\sum_{n=j_2+1}^{j_3-1} D^{N-1}_{m,n\%2^{N-1}} \\ &-D^{N-1}_{m,j_3\%2^{N-1}} - a^{N-1}_{m,j_1\%2^{N-1}} + 2a^{N-1}_{m,j_2\%2^{N-1}} - a^{N-1}_{m,j_3\%2^{N-1}}\end{aligned}\right)}{2^{2k-1}}$$
$$\text{where } j_1 = 2^{k-1}j + \left\lfloor \frac{s_h}{2} \right\rfloor, \quad j_2 = 2^{k-2}(2j+1) + \left\lfloor \frac{s_h}{2} \right\rfloor, \quad j_3 = 2^{k-1}(j+1) + \left\lfloor \frac{s_h}{2} \right\rfloor \tag{8}$$

$$a^{N-k}_{i,j_{\text{new}}} = \sum_{m=2^{k-t}i}^{2^{k-t}(i+1)} \frac{\left(\left(A^{N-t}_{m,j_1\%2^{N-t}} + \cdots + A^{N-t}_{m,(j_2-1)\%2^{N-t}}\right) - \left(A^{N-t}_{m,j_2\%2^{N-t}} + \cdots + A^{N-t}_{m,(j_3-1)\%2^{N-t}}\right)\right)}{4^{k-t}}$$
$$\text{where } j_1 = \frac{2^{k-t}j + s_h}{2^t}, \quad j_2 = \frac{2^{k-t-1}(2j+1) + s_h}{2^t}, \quad j_3 = \frac{2^{k-t}(j+1) + s_h}{2^t} \tag{9}$$

$$a^{N-k}_{i,j_{\text{new}}} = \sum_{m=2^{k-t-1}i}^{2^{k-t-1}(i+1)-1} \frac{\left(\begin{aligned} &D^{N-t-1}_{m,j_1\%2^{N-t-1}} + 2\sum_{n=j_1+1}^{j_2-1} D^{N-t-1}_{m,n\%2^{N-t-1}} - 2\sum_{n=j_2+1}^{j_3-1} D^{N-t-1}_{m,n\%2^{N-t-1}} \\ &-D^{N-t-1}_{m,j_3\%2^{N-t-1}} - a^{N-t-1}_{m,j_1\%2^{N-t-1}} + 2a^{N-t-1}_{m,j_2\%2^{N-t-1}} - a^{N-t-1}_{m,j_3\%2^{N-t-1}}\end{aligned}\right)}{(2\times 4^{k-t-1})}$$
$$\text{where } j_1 = 2^{k-t-1}j + \left\lfloor \frac{s_h}{2^{t+1}} \right\rfloor, \quad j_2 = 2^{k-t-2}(2j+1) + \left\lfloor \frac{s_h}{2^{t+1}} \right\rfloor, \quad j_3 = 2^{k-t-1}(j+1) + \left\lfloor \frac{s_h}{2^{t+1}} \right\rfloor \tag{10}$$

Authorized licensed use limited to: IEEE Xplore. Downloaded on January 21, 2009 at 18:27 from IEEE Xplore. Restrictions apply.

To compute the $b_{i,j_{\mathrm{new}}}^{N-k}$ coefficient for $k > t$ after a horizontal shift, we write $b_{i,j_{\mathrm{new}}}^{N-k}$ in terms of the coefficients at level $N - t$ [see (13), shown at the bottom of the page]. Equation (13) defines a window that encloses the blur coefficients at level $N-t$ that fall between the two corners $(2^{k-t}i, 2^{k-t}j + s_h/2^t)$ and $(2^{k-t}(i+1), 2^{k-t}(j+1) + s_h/2^t)$, which are the same bounds used to compute $a_{i,j_{\mathrm{new}}}^{N-k}$ in (11). $b_{i,j_{\mathrm{new}}}^{N-k}$ is found by summing each row of blur coefficients in the upper half of the window and subtracting the sum of each row in the lower half. Fig. 2(b) illustrates the computation of $b_{0,0}^2$ using the blur coefficients at level $l = 3$ under a horizontal odd shift of one.

By substituting (1), (3), and (4) in (13), and combining the result with (12) we get the coefficients of the shifted signal [see (14), shown at the bottom of the page].

$$k > t : a_{i,j_{\mathrm{new}}}^{N-k} = \sum_{m=2^{k-t-1}i}^{2^{k-t-1}(i+1)-1} \frac{\left(D_{m,j_1\%2^{N-t-1}}^{N-t-1} + 2\sum_{n=j_1+1}^{j_2-1} D_{m,n\%2^{N-t-1}}^{N-t-1} - 2\sum_{n=j_2+1}^{j_3-1} D_{m,n\%2^{N-t-1}}^{N-t-1} - D_{m,j_3\%2^{N-t-1}}^{N-t-1} - a_{m,j_1\%2^{N-t-1}}^{N-t-1} + 2a_{m,j_2\%2^{N-t-1}}^{N-t-1} - a_{m,j_3\%2^{N-t-1}}^{N-t-1} \right)}{2^{2k-2t-1}}$$

$$k \le t : a_{i,j_{\mathrm{new}}}^{N-k} = a_{i,(j+s_h/2^k)\%2^{N-k}}^{N-k}$$

$$\text{where } j_1 = 2^{k-t-1}j + \left\lfloor \frac{s_h}{2^{t+1}} \right\rfloor, \quad j_2 = 2^{k-t-2}(2j+1) + \left\lfloor \frac{s_h}{2^{t+1}} \right\rfloor, \quad j_3 = 2^{k-t-1}(j+1) + \left\lfloor \frac{s_h}{2^{t+1}} \right\rfloor \tag{11}$$

$$b_{i,j_{\mathrm{new}}}^{N-k} = \sum_{m=2^{k-t}i}^{2^{k-t-1}(2i+1)-1} \frac{\left(A_{m,j_1\%2^{N-t}}^{N-t} + \cdots + A_{m,(j_2-1)\%2^{N-t}}^{N-t} \right)}{4^{k-t}} - \sum_{m=2^{k-t-1}(2i+1)}^{2^{k-t}(i+1)-1} \frac{\left(A_{m,j_1\%2^{N-t}}^{N-t} + \cdots + A_{m,(j_2-1)\%2^{N-t}}^{N-t} \right)}{4^{k-t}}$$

$$\text{where } j_1 = \frac{2^{k-t}j + s_h}{2^t}, \quad j_2 = \frac{2^{k-t}(j+1) + s_h}{2^t} \tag{13}$$

$$k > t+1 : b_{i,j_{\mathrm{new}}}^{N-k} = \sum_{m=2^{k-t-1}i}^{2^{k-t-2}(2i+1)-1} \frac{\left(D_{m,j_1\%2^{N-t-1}}^{N-t-1} + 2\sum_{n=j_1+1}^{j_2-1} D_{m,n\%2^{N-t-1}}^{N-t-1} + D_{m,j_2\%2^{N-t-1}}^{N-t-1} - a_{m,j_1\%2^{N-t-1}}^{N-t-1} + a_{m,j_2\%2^{N-t-1}}^{N-t-1} \right)}{2^{2k-2t-1}} - \sum_{m=2^{k-t-2}(2i+1)}^{2^{k-t-1}(i+1)-1} \frac{\left(D_{m,j_1\%2^{N-t-1}}^{N-t-1} + 2\sum_{n=j_1+1}^{j_2-1} D_{m,n\%2^{N-t-1}}^{N-t-1} + D_{m,j_2\%2^{N-t-1}}^{N-t-1} - a_{m,j_1\%2^{N-t-1}}^{N-t-1} + a_{m,j_2\%2^{N-t-1}}^{N-t-1} \right)}{2^{2k-2t-1}}$$

$$k = t+1 : b_{i,j_{\mathrm{new}}}^{N-k} = \frac{\left(b_{i,(j+\lfloor s_h/2^k \rfloor)\%2^{N-k}}^{N-k} - c_{i,(j+\lfloor s_h/2^k \rfloor)\%2^{N-k}}^{N-k} + b_{i,(j+\lfloor s_h/2^k \rfloor+1)\%2^{N-k}}^{N-k} + c_{i,(j+\lfloor s_h/2^k \rfloor+1)\%2^{N-k}}^{N-k} \right)}{2}$$

$$k \le t : b_{i,j_{\mathrm{new}}}^{N-k} = b_{i,(j+s_h/2^k)2^{N-k}}^{N-k}$$

$$\text{where } j_1 = 2^{k-t-1}j + \left\lfloor \frac{s_h}{2^{t+1}} \right\rfloor, \quad j_2 = 2^{k-t-1}(j+1) + \left\lfloor \frac{s_h}{2^{t+1}} \right\rfloor \tag{14}$$

Authorized licensed use limited to: IEEE Xplore. Downloaded on January 21, 2009 at 18:27 from IEEE Xplore. Restrictions apply.

TABLE I
COMPARISON WITH OTHER METHODS: ACCUMULATED RESIDUAL ERRORS IN ROOT MEAN SQUARE (RMS) AFTER SEVERAL SUCCESSIVE SHIFTS (SEE TEXT FOR MORE DETAILS)

										
Bilinear	30.3943	14.0688	21.6153	21.2099	28.244	18.2492	26.867	6.7089	13.6063	21.4582
Bicubic	19.7466	6.9046	13.0134	12.7079	20.2513	8.6019	14.9709	4.7218	6.3287	12.1973
Cubic Spline	14.975	4.4063	8.9801	8.1769	15.0742	5.5216	10.162	3.9943	4.2309	7.788
Our Method	0.7646	0.7212	0.6236	0.7845	0.714	0.7215	0.6625	0.7961	0.743	0.7123

V. DIAGONAL COEFFICIENTS FOR HORIZONTAL SHIFT

Similar to 5 and 12, we can apply the following equation when the horizontal shift s_h is divisible by 2^k:

$$c^{N-k}_{i,j_{\text{new}}} = c^{N-k}_{i,(j+s_h/2^k)\%2^{N-k}}. \tag{15}$$

To compute the $c^{N-k}_{i,j_{\text{new}}}$ coefficient for $k > t$ after a horizontal shift, we write $c^{N-k}_{i,j_{\text{new}}}$ in terms of coefficients at level $N-t$ [see (16), shown at the bottom of the page].

Equation (16) defines a window which has the same bounds used to compute $a^{N-k}_{i,j_{\text{new}}}$ and $b^{N-k}_{i,j_{\text{new}}}$ in (11) and (14), respectively. By examining the example in Fig. 2(c), one can see that $c^{N-k}_{i,j_{\text{new}}}$ is computed by summing the first half and subtracting the second half of each row in the upper half of the window, while subtracting the first half and summing the second half of each row in the lower half of the window, hence, (16).

By substituting (1), (3), and (4) in (16), and combining the result with (15), we get (17), shown under (16).

$$c^{N-k}_{i,j_{\text{new}}} = \sum_{m=2^{k-t}i}^{2^{k-t-1}(i+1)-1} \frac{\left(\left(A^{N-t}_{m,j_1\%2^{N-t}}+\cdots+A^{N-t}_{m,(j_2-1)\%2^{N-t}}\right)-\left(A^{N-t}_{m,j_2\%2^{N-t}}+\cdots+A^{N-t}_{m,(j_3-1)\%2^{N-t}}\right)\right)}{4^{k-t}}$$
$$- \sum_{m=2^{k-t-1}(i+1)}^{2^{k-t}(i+1)-1} \frac{\left(\left(A^{N-t}_{m,j_1\%2^{N-t}}+\cdots+A^{N-t}_{m,(j_2-1)\%2^{N-t}}\right)-\left(A^{N-t}_{m,j_2\%2^{N-t}}+\cdots+A^{N-t}_{m,(j_3-1)\%2^{N-t}}\right)\right)}{4^{k-t}}$$
$$\text{where } j_1 = \frac{2^{k-t}j+s_h}{2^t}, \quad j_2 = \frac{2^{k-t-1}(2j+1)+s_h}{2^t}, \quad j_3 = \frac{2^{k-t}(j+1)+s_h}{2^t} \tag{16}$$

$$k > t+1 : c^{N-k}_{i,j_{\text{new}}} = \sum_{m=2^{k-t-1}i}^{2^{k-t-2}(2i+1)-1} \frac{\begin{pmatrix} D^{N-t-1}_{m,j_1\%2^{N-t-1}} + 2\sum_{n=j_1+1}^{j_2-1} D^{N-t-1}_{m,n\%2^{N-t-1}} - 2\sum_{n=j_2+1}^{j_3-1} D^{N-t-1}_{m,n\%2^{N-t-1}} \\ -D^{N-t-1}_{m,j_3\%2^{N-t-1}} - a^{N-t-1}_{j_1\%2^{N-t-1}} + 2a^{N-t-1}_{j_2\%2^{N-t-1}} - a^{N-t-1}_{j_3\%2^{N-t-1}} \end{pmatrix}}{2^{2k-2t-1}}$$
$$- \sum_{m=2^{k-t-2}(2i+1)}^{2^{k-t-1}(i+1)-1} \frac{\begin{pmatrix} D^{N-t-1}_{m,j_1\%2^{N-t-1}} + 2\sum_{n=j_1+1}^{j_2-1} D^{N-t-1}_{m,n\%2^{N-t-1}} - 2\sum_{n=j_2+1}^{j_3-1} D^{N-t-1}_{m,n\%2^{N-t-1}} \\ -D^{N-t-1}_{m,j_3\%2^{N-t-1}} - a^{N-t-1}_{j_1\%2^{N-t-1}} + 2a^{N-t-1}_{j_2\%2^{N-t-1}} - a^{N-t-1}_{j_3\%2^{N-t-1}} \end{pmatrix}}{2^{2k-2t-1}}$$
$$k = t+1 : c^{N-k}_{i,j_{\text{new}}} = \frac{\begin{pmatrix} b^{N-k}_{i,(j+\lfloor s_h/2^k\rfloor)\%2^{N-k}} - c^{N-k}_{i,(j+\lfloor s_h/2^k\rfloor)\%2^{N-k}} \\ -b^{N-k}_{i,(j+\lfloor s_h/2^k\rfloor+1)\%2^{N-k}} - c^{N-k}_{i,(j+\lfloor s_h/2^k\rfloor+1)\%2^{N-k}} \end{pmatrix}}{2}$$
$$k \le t : c^{N-k}_{i,j_{\text{new}}} = c^{N-k}_{i,(j+s_h/2^k)\%2^{N-k}}$$
$$\text{where } j_1 = 2^{k-t-1}j + \left\lfloor \frac{s_h}{2^{t+1}} \right\rfloor, \quad j_2 = 2^{k-t-2}(2j+1) + \left\lfloor \frac{s_h}{2^{t+1}} \right\rfloor, \quad j_3 = 2^{k-t-1}(j+1) + \left\lfloor \frac{s_h}{2^{t+1}} \right\rfloor \tag{17}$$

Authorized licensed use limited to: IEEE Xplore. Downloaded on January 21, 2009 at 18:27 from IEEE Xplore. Restrictions apply.

$$h \geq t+1:$$

$$a_{i,j_{\text{new}}}^{N'-k} = \sum_{m=2^{k-t-1}i}^{2^{k-t-1}(i+1)-1} \frac{\left(D^N_{\lfloor m/N'-t-1\rfloor,\lfloor j_1\%2^{N'-t-1}/N'-t-1\rfloor} + 2\sum_{n=j_1+1}^{j_2-1} D^N_{\lfloor m/'-t-1\rfloor,\lfloor n\%2^{N'-t-1}/N'-t-1\rfloor} - 2\sum_{n=j_2+1}^{j_3-1} D^N_{\lfloor m/N'-t-1\rfloor,\lfloor n\%2^{N'-t-1}/N'-t-1\rfloor} - D^N_{\lfloor m/N'-t-1\rfloor,\lfloor j_3\%2^{N'-t-1}/N'-t-1\rfloor}\right)}{2^{2k-2t-1}}$$

$$h < t+1, \quad k > t:$$

$$a_{i,j_{\text{new}}}^{N'-k} = \sum_{m=2^{k-t-1}i}^{2^{k-t-1}(i+1)-1} \frac{\left(D^{N'-t-1}_{m,j_1\%2^{N'-t-1}} + 2\sum_{n=j_1+1}^{j_2-1} D^{N'-t-1}_{m,n\%2^{N'-t-1}} - 2\sum_{n=j_2+1}^{j_3-1} D^{N'-t-1}_{m,n\%2^{N'-t-1}} - D^{N'-t-1}_{m,j_3\%2^{N'-t-1}} - a^{N'-t-1}_{m,j_1\%2^{N'-t-1}} + 2a^{N'-t-1}_{m,j_2\%2^{N'-t-1}} - a^{N'-t-1}_{m,j_3\%2^{N'-t-1}}\right)}{2^{2k-2t-1}}$$

$$h < t+1, \quad k \leq t: a_{i,j_{\text{new}}}^{N'-k} = a^{N'-k}_{i,(j+s_h/2^k)\%2^{N'-k}}$$

$$\text{where } j_1 = 2^{k-t-1}j + \left\lfloor \frac{s_h}{2^{t+1}} \right\rfloor, \quad j_2 = 2^{k-t-2}(2j+1) + \left\lfloor \frac{s_h}{2^{t+1}} \right\rfloor, \quad j_3 = 2^{k-t-1}(j+1) + \left\lfloor \frac{s_h}{2^{t+1}} \right\rfloor \tag{18}$$

$$h \geq t+1, \quad k > t+1: b_{i,j_{\text{new}}}^{N'-k} = \sum_{m=2^{k-t-1}i}^{2^{k-t-2}(2i+1)-1} \frac{\left(D^N_{\lfloor m/N'-t-1\rfloor,\lfloor j_1\%2^{N'-t-1}/N'-t-1\rfloor} + 2\sum_{n=j_1+1}^{j_2-1} D^N_{\lfloor m/N'-t-1\rfloor,\lfloor n\%2^{N'-t-1}/N'-t-1\rfloor} + D^N_{\lfloor m/N'-t-1\rfloor,\lfloor j_2\%2^{N'-t-1}/N'-t-1\rfloor}\right)}{2^{2k-2t-1}}$$
$$- \sum_{m=2^{k-t-2}(2i+1)}^{2^{k-t-1}(i+1)-1} \frac{\left(D^N_{\lfloor m/N'-t-1\rfloor,\lfloor j_1\%2^{N'-t-1}/N'-t-1\rfloor} + 2\sum_{n=j_1+1}^{j_2-1} D^N_{\lfloor m/N'-t-1\rfloor,\lfloor n\%2^{N'-t-1}/N'-t-1\rfloor} + D^N_{\lfloor m/N'-t-1\rfloor,\lfloor j_2\%2^{N'-t-1}/N'-t-1\rfloor}\right)}{2^{2k-2t-1}}$$

$$h < t+1, \quad k > t+1: b_{i,j_{\text{new}}}^{N'-k} = \sum_{m=2^{k-t-1}i}^{2^{k-t-2}(2i+1)-1} \frac{\left(D^{N'-t-1}_{m,j_1\%2^{N'-t-1}} + 2\sum_{n=j_1+1}^{j_2-1} D^{N'-t-1}_{m,n\%2^{N'-t-1}} + D^{N'-t-1}_{m,j_2\%2^{N'-t-1}} - a^{N'-t-1}_{m,j_1\%2^{N'-t-1}} + a^{N'-t-1}_{m,j_2\%2^{N'-t-1}}\right)}{2^{2k-2t-1}}$$
$$- \sum_{m=2^{k-t-2}(2i+1)}^{2^{k-t-1}(i+1)-1} \frac{\left(D^{N'-t-1}_{m,j_1\%2^{N'-t-1}} + 2\sum_{n=j_1+1}^{j_2-1} D^{N'-t-1}_{m,n\%2^{N'-t-1}} + D^{N'-t-1}_{m,j_2\%2^{N'-t-1}} - a^{N'-t-1}_{m,j_1\%2^{N'-t-1}} + a^{N'-t-1}_{m,j_2\%2^{N'-t-1}}\right)}{2^{2k-2t-1}}$$

$$h < t+1, \quad k = t+1: b_{i,j_{\text{new}}}^{N'-k} = \frac{\left(b^{N'-k}_{i,(j+\lfloor s_h/2^k\rfloor)\%2^{N'-k}} - c^{N'-k}_{i,(j+\lfloor s_h/2^k\rfloor)\%2^{N'-k}} + b^{N'-k}_{i,(j+\lfloor s_h/2^k\rfloor+1)\%2^{N'-k}} + c^{N'-k}_{i,(j+\lfloor s_h/2^k\rfloor+1)\%2^{N'-k}}\right)}{2}$$

$$h < t+1, \quad k \leq t: b_{i,j_{\text{new}}}^{N'-k} = b^{N'-k}_{i,(j+s_h/2^k)\%2^{N'-k}}$$

$$\text{where } j_1 = 2^{k-t-1}j + \left\lfloor \frac{s}{2^{t+1}_h} \right\rfloor, \quad j_2 = 2^{k-t-1}(j+1) + \left\lfloor \frac{s_h}{2^{t+1}} \right\rfloor \tag{19}$$

Authorized licensed use limited to: IEEE Xplore. Downloaded on January 21, 2009 at 18:27 from IEEE Xplore. Restrictions apply.

The relations (11), (14), and (17) can now be used to compute the new detail coefficients of the Haar transform at all different levels after any horizontal shift $s_h = 0, \ldots, 2^N - 1$ using only the 0-shift coefficients. The equations for the vertical shift can be derived following the same steps that we used for horizontal shifting. They turn out to look the same as the horizontal case after interchanging the a's with b's, i's with j's and m's with n's.

VI. Subpixel Shifting

Let the size of the signal be $2^N \times 2^N$, $N' = N + h$ and $k = 1 + h, \ldots, N + h$, where h is the number of added levels. Equations (11), (14), and (17) can now be modified to allow for noninteger shifting by a precision of $1/2^h$ by substituting N' for each N in the equations. On the other hand, we can verify that $D_{i,j}^{N+h_0} = D^N_{\lfloor i/2^{h_0} \rfloor, \lfloor j/2^{h_0} \rfloor}$, where $0 \le h_0 \le h$. This combined with (4) allows us to modify (18) and (19) (shown on the previous page), as well as (20) (shown at the bottom of the page), in such a way that avoids having to actually up-sample the signal for noninteger shifts, thus saving memory space in actual implementation, especially that the size increases exponentially. However, we have to split the equation into two cases. The first is when $h \ge t + 1$, which is when the coefficients at the added levels are being used to compute $a^{N'-k}_{i,j_{\mathrm{new}}}$, $b^{N'-k}_{i,j_{\mathrm{new}}}$ and $c^{N'-k}_{i,j_{\mathrm{new}}}$. The second is when t is large enough for the coefficients at the original levels of the tree to be used. This leads to the phase shifting relation for noninteger values as in (18)–(20).

VII. Experimental Results and Discussion

For integer shifting, our solution for Haar-domain phase shifting, given by (11), (14), and (17), is exact and does not incur any errors. For the subpixel case, a shift is approximated by modeling the process as upsampling by a factor of h followed by an integer shift. However, our solution does not require to actually upsample the image—i.e., the solution derived performs subpixel shifts directly from the original coefficients. For a factor of h, we can obtain a precision of $1/2^h$, i.e., subpixel shifts are approximated by the closest value in multiples of $1/2^h$.

In order to evaluate our method, we compared our results with bilinear, bicubic, and cubic spline interpolations. For this purpose, using each method, we performed successive noninteger shifts until the test image was shifted by some prescribed large integer value. We then used the residual error accumulated over these successive shifts as a measure of performance. We performed this experiment on numerous test images, some of which are shown in Table I. As shown inTable I, the accumulated residual errors in root mean square (rms) are on average an order of magnitude smaller in our method than the other three methods, i.e., our method accumulates far less error over successive shifts.

TABLE II
Complexity and Probability at Each Reduction Level k for Evaluating the 2-D Detail Coefficients $a^{N-k}_{i,j_{\mathrm{new}}}$

Reduction Level	Complexity	Prob.$=\frac{\text{Number of Coefficients at } k}{\text{Number of Coefficients}}$
$k=N$	$O((\frac{L}{2})^2)$	$\frac{3}{L^2-1}$
$k=N-1$	$O((\frac{L}{2^2})^2)$	$\frac{(2^1)^2\times 3}{L^2-1}$
$k=N-2$	$O((\frac{L}{2^3})^2)$	$\frac{(2^2)^2\times 3}{L^2-1}$
$k=N-3$	$O((\frac{L}{2^4})^2)$	$\frac{(2^3)^2\times 3}{L^2-1}$
:	:	:
$k=1$	$O((\frac{L}{2^N})^2)$	$\frac{(2^{N-1})^2\times 3}{L^2-1}$

$$
h \ge t+1, \quad k > t+1:
$$
$$
c^{N'-k}_{i,j\mathrm{new}} = \frac{\left(\displaystyle\sum_{m=2^{k-t-1}i}^{2^{k-t-2}(2i+1)-1} \left(D^N_{\lfloor m/N'-t-1 \rfloor, \lfloor j_1 \% 2^{N'-t-1}/N'-t-1 \rfloor} + 2\sum_{n=j_1+1}^{j_2-1} D^N_{\lfloor m/N'-t-1 \rfloor, \lfloor n \% 2^{N'-t-1}/N'-t-1 \rfloor} - 2\sum_{n=j_2+1}^{j_3-1} D^N_{\lfloor m/N'-t-1 \rfloor, \lfloor n \% 2^{N'-t-1}/N'-t-1 \rfloor} - D^N_{\lfloor m/N'-t-1 \rfloor, \lfloor j_3 \% 2^{N'-t-1}/N'-t-1 \rfloor} \right) - \displaystyle\sum_{m=2^{k-t-2}(2i+1)}^{2^{k-t-1}(i+1)-1} \left(D^N_{\lfloor m/N'-t-1 \rfloor, \lfloor j_1 \% 2^{N'-t-1}/N'-t-1 \rfloor} + 2\sum_{n=j_1+1}^{j_2-1} D^N_{\lfloor m/N'-t-1 \rfloor, \lfloor n \% 2^{N'-t-1}/N'-t-1 \rfloor} - 2\sum_{n=j_2+1}^{j_3-1} D^N_{\lfloor m/N'-t-1 \rfloor, \lfloor n \% 2^{N'-t-1}/N'-t-1 \rfloor} - D^N_{\lfloor m/N'-t-1 \rfloor, \lfloor j_3 \% 2^{N'-t-1}/N'-t-1 \rfloor} \right) \right)}{2^{2k-2t-1}}
$$
$$
h < t+1, \quad k = t+1: c^{N'-k}_{i,j_{\mathrm{new}}} = \frac{\left(b^{N'-k}_{i,(j+\lfloor s_h/2^k \rfloor)\%2^{N'-k}} - c^{N'-k}_{i,(j+\lfloor s_h/2^k \rfloor)\%2^{N'-k}} - b^{N'-k}_{i,(j+\lfloor s_h/2^k \rfloor+1)\%2^{N'-k}} - c^{N'-k}_{i,(j+\lfloor s_h/2^k \rfloor+1)\%2^{N'-k}} \right)}{2}
$$
$$
h < t+1, \quad k \le t: c^{N'-k}_{i,j_{\mathrm{new}}} = c^{N'-k}_{i,(j+s_h/2^k)\%2^{N'-k}}
$$
$$
\text{where } j_1 = 2^{k-t-1}j + \left\lfloor \frac{s_h}{2^{t+1}} \right\rfloor, \quad j_2 = 2^{k-t-2}(2j+1) + \left\lfloor \frac{s_h}{2^{t+1}} \right\rfloor, \quad j_3 = 2^{k-t-1}(j+1) + \left\lfloor \frac{s_h}{2^{t+1}} \right\rfloor \qquad (20)
$$

Authorized licensed use limited to: IEEE Xplore. Downloaded on January 21, 2009 at 18:27 from IEEE Xplore. Restrictions apply.

On the other hand, the low complexity of the derived solutions allow for fast processing directly in the transform domain. By examining (11), one can find that the complexity of evaluating $a_{i,j_{\text{new}}}^{N-k}$ can be expressed by the difference of the bounds of the two inner sums in the equation multiplied by the difference of the bounds of the outer sum, that is $O((j_3 - j_1) \times (2^{k-t-1}))$. Substituting the values for j_1 and j_3, the complexity is shown to be $O(2^{k-t-1} \times 2^{k-t-1})$ when $k > t$. When $k \leq t$ the complexity becomes $O(1)$. Therefore, one can determine that the worst case is when $t = 0$, that is when the shift is odd. In that case the complexity of computing $a_{i_{\text{new}}}^{N-k}$ becomes $O(2^{k-1} \times 2^{k-1})$. Let $L \times L = 2^N \times 2^N$ be the size of the 2-D signal, then the number of the detail coefficients is $L^2 - 1 = 4^N - 1$. The a detail coefficients are one third of the total number of coefficients, i.e., $4^N - 1/3$. At reduction level $k = N$, i.e., the root, the complexity of computing $a_{0,0_{\text{new}}}^{0}$ is $O(2^{N-1} \times 2^{N-1}) = O((L/2)^2)$ with a probability of $3/L^2 - 1$. At the next reduction level $k = N-1$, the complexity is $O(2^{N-2} \times 2^{N-2}) = O(2^2 \times (L/2^2)^2)$ with a probability of $2^2 \times (3/L^2 - 1)$. Table II shows the complexity and its probability at each reduction level k.

By multiplying the complexities and the probabilities in Table II and summing them up, the average performance of the worst case for evaluating $a_{i,j_{\text{new}}}^{N-k}$ is found to be $O(\log(L))$. Following the same analysis, one can find that the worst case complexities for evaluating $b_{i,j_{\text{new}}}^{N-k}$ and $c_{i,j_{\text{new}}}^{N-k}$ are found to be $O(\log(L))$, as well.

References

[1] T. Chang and C. C. J. Kuo, "Texture analysis and classification with tree-structured wavelet transform," *IEEE Trans. Image Process.*, vol. 2, no. 4, pp. 429–441, Oct. 1993.

[2] N. Saito and R. R. Coifman, "Local discriminant bases," in *Proc. Wavelet Applications in Signal and Image Processing II*, A. F. Laine and M. A. Unser, Eds., 1994, pp. 2–14.

[3] N. Saito and R. Coifman, "On local orthonormal bases for classification and regression," in *Proc. ICASSP*, 1995, vol. 3, pp. 1529–1532.

[4] M. Do and M. Vetterli, "Rotation invariant texture characterization and retrieval using steerable wavelet-domain hidden Markov models," *IEEE Trans. Multimedia*, vol. 4, no. 4, pp. 517–527, Dec. 2002.

[5] M. Do, S. Ayer, and M. Vetterli, "Invariant image retrieval using wavelet maxima moment," in *Proc. Vis. Inf. Inf. Syst.*, 1999, pp. 451–458.

[6] M. Do and M. Vetterli, "Texture similarity measurement using Kullback–Leibler distance on wavelet subbands," in *Proc. IEEE Int. Conf. Image Processing*, 2000, vol. 3, pp. 730–733.

[7] R. R. Coifman and M. V. Wickerhauser, "Adapted waveform analysis as a tool for modeling, feature extraction, and denoising," *Opt. Eng.*, vol. 33, no. 7, pp. 2170–2174, 1994.

[8] R. R. Coifman and D. L. Donoho, "Translation-invariant de-noising," Tech. Rep., Dept. Statist., 1995.

[9] D. L. Donoho, "De-noising by soft-thresholding," *IEEE Trans. Inf. Theory*, vol. 41, pp. 613–627, 1995.

[10] M. Figueiredo, J. Bioucas-Dias, and R. Nowak, "Majorization-minimization algorithms for wavelet-based image restoration," *IEEE Trans. Image Process.*, vol. 16, no. 12, pp. 2980–2991, Dec. 2007.

[11] D. D. H. Krim, S. Mallat, and A. Willsky, "Best basis algorithm for signal enhancement," in *Proc. ICASSP*, 1995, vol. 3, pp. 1561–1564.

[12] M. Vetterli and J. Kovačevic, *Wavelets and Subband Coding*. Upper Saddle River, NJ: Prentice-Hall, 1995.

[13] H.-W. Park and H.-S. Kim, "Motion estimation using low-band-shift method for wavelet-based moving-picture coding," *IEEE Trans. Image Process.*, vol. 9, no. 4, pp. 577–587, Apr. 2000.

[14] E. P. Simoncelli, W. T. Freeman, E. H. Adelson, and D. J. Heeger, "Shiftable multi-scale transforms," *IEEE Trans. Inf. Theory*, vol. 38, no. 2, pp. 587–607, Mar. 1992.

[15] M. Holschneider, R. Kronland-Martinet, and P. Tchamitchian, "A real-time algorithm for signal analysis with the help of the wavelet transform," in *Proc. Int. Conf. Wavelets. Time-Frequency Methods and Phase Space*, Marseille, France, 1989, pp. 286–292.

[16] P. Dutilleux, "An implementation of the algorithm à trous to compute the wavelet transform," in *Wavelets, Time-Frequency Methods and Phase Space*, J. Combes, A. Grossmann, and P. Tchamitchian, Eds. New York: Springer, 1989, pp. 298–304.

[17] S. Mallat, *A Wavelet Tour of Signal Processing*. New York: Academic, 1998.

[18] C. S. Burrus, R. A. Gopinath, and H. Guo, *Introduction to Wavelets and Wavelets Transforms*. Englewood Cliffs, NJ: Prentice-Hall, 1997.

[19] M. Lang, H. Guo, J. E. Odegard, C. S. Burrus, and R. O. Wells, "Noise reduction using an undecimated discrete wavelet transform," *IEEE Signal Process. Lett.*, vol. 3, 1996.

[20] B. S. Manjunath and W. Y. Ma, "Texture features for browsing and retrieval of image data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 8, pp. 837–842, Aug. 1996.

[21] D. E. Newland, "Harmonic wavelet analysis," *Proc. Roy. Soc. Lond. A*, vol. 443, pp. 203–225, 1993.

[22] D. E. Newland, "Harmonic wavelets in vibrations and acoustics," *Phil. Trans. Roy. Soc. Lond. A*, vol. 357, no. 1760, pp. 2607–2625, 1999.

[23] J. Magarey and N. Kingsbury, "Motion estimation using complex wavelets," in *Proc. IEEE ICASSP*, Washington, DC, 1996, pp. 2371–2374.

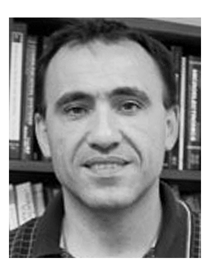
[24] P. de Rivaz and N. Kingsbury, "Complex wavelet features for fast texture image retrieval," in *Proc. Int. Conf. Image Processing*, 1999, pp. I:109–113.

[25] P. D. Rivaz and N. G. Kingsbury, "Fast segmentation using level set curves of complex wavelet surfaces," presented at the Int. Conf. Image Processing, 2000.

[26] H. Sari-Sarraf and D. Brzakovic, "A shift-invariant discrete wavelet transform," *IEEE Trans. Signal Process.*, vol. 45, no. 10, pp. 2621–2626, Oct. 1997.

Mais Alnasser, received the Ph.D. degree in computer science from the University of Central Florida, Orlando, in 2008.

She is currently a Senior Research Engineer in Advanced Micro Devices (AMD). Her research interests include global illumination, 3-D modeling, and the common grounds of computer vision and computer graphics.



Hassan Foroosh (M'02–SM'03) recieived the M.S. and Ph.D. degrees in computer science, specializing in computer vision and image processing, from INRIA-Sophia Antipolis, France, in 1993 and 1996, respectively.

He is an Assistant Professor of electrical engineering and computer science and the Director of the Computational Imaging Laboratory (CIL), University of Central Florida (UCF), Orlando. He has authored and co-authored over 70 peer-reviewed journals and conference papers.

Dr. Foroosh is an Associate Editor of the IEEE TRANSACTIONS ON IMAGE PROCESSING and has been on the organizing and the technical committees of numerous international colloquia. In 2004, he was a recipient of an academic excellence award from Sun Microsystems and the Piero Zamperoni award from the International Association for Pattern Recognition (IAPR).

Authorized licensed use limited to: IEEE Xplore. Downloaded on January 21, 2009 at 18:27 from IEEE Xplore. Restrictions apply.