14th European Signal Processing Conference (EUSIPCO 2006), Florence, Italy, September 4-8, 2006, copyright by EURASIP

THE RWHT+P FOR AN IMPROVED LOSSLESS MULTIRESOLUTION CODING

Olivier Déforges [a], Marie Babel [a], Jean Motsch [b]

IETR UMR CNRS 6164, Image and Remote Sensing Group, FRANCE
[a] INSA of Rennes, 35043 RENNES. Contact: {mbabel,odeforge}@insa-rennes.fr
[b] LESTP/CREC St-Cyr, 56381 Guer. Contact: jean.motsch@st-cyr.terre.defense.gouv.fr

ABSTRACT

This paper presents a full multiresolution lossless coding method, with advanced semantic scalability. In particular, a reversible form of the usual Walsh Hadamard Transform (RWHT) is first introduced as an alternative to standard lossless transform. A pyramidal representation and decomposition schemes involving this basic transform are then proposed. Significant improvements are obtained using two additional concepts: the "locally adaptive resolution" through a quadtree representation and a prediction step. The given experimental results show that the proposed RWHT+P achieves excellent performances compared to state-of-the-art.

1. INTRODUCTION

New generations of images codecs should have of course to be efficient in terms of compression performances, but also to provide advanced functionalities such as scalability, rate control, region of interest encoding, meaningful scene description. Moreover, a single coding scheme able to compress from very low bit rates to lossless, would be a suitable solution for general purpose uses of image coding. In [1], we have introduced the LAR (Locally Adaptive Resolution) method leading to an efficient lossy image compression technique. The LAR compression method is a two-layer codec: a spatial codec and a complementary spectral one. The spatial coder provides a main image compressed at low bit rate, whereas the spectral one encodes the local texture. The quality lossy compressed images by LAR has been evaluated and recognized to be better than Jpeg-2000 [1]. The LAR method relies on a quadtree variable block-size decomposition estimated from local activity. This particular representation enabled an extension to region-based image representation and encoding from only the low bit rate images [2]. Recently, we have proposed a modified version of the codec enabling also efficient lossless coding while significantly improving its scalabity [3]: spatial and spectral layers have been subtituted by two successive multiresolution quadtree decompositions based on a modified version of the S transform. This paper presents an alternative to the previous coding scheme in terms of decomposition. This new form of LAR codec called "RWHT+P" outperforms previous version both for very low bit rates and lossless encoding, but we focus here only on the last framework. In particular, the first part of the paper introduces a new method to enable the classical Walsh-Hadamard Transform (WHT) with a 2×2 kernel to be used in a lossless transformation context. The remaining of the paper proposes a full lossless coding scheme with enhanced resolution scalability.

In lossless coding, compresssion and decompression of source data result in the exact recovery of every element of the original source data. Lossless image coding is necessary in applications where no degradation is tolerable. Examples are medical imaging, remote sensing, image/video archiving and studio applications. The state-of-the-art in lossless coding is roughly composed of two main approaches: predictive methods in the spatial domain with some popular coder such as CALIC [4], and transform-based methods generally using the wavelets theory. The main advantage of wavelets coders is that they offer scalable coding and possibly multiresolution representations of the image.

Lossless transforms have the particularity to map integers into integers in contrary to lossy ones. Most of the methods use the concept of "rounding" to enable an unambiguously retrieval of the transformed coefficients [5].

The WHT is a well known technique used for signal and image compression. Many multiresolution image coding methods use this transform on 2×2 blocks, especially in wavelet decomposition schemes. For lossless compression purposes a modifed version of the 1D WHT has been proposed by P. Lux [6], popularized by Said [5] and known as the "S" transform or "Haar integer wavelet transform" [7]. The S transform is currently known as one of the best integer wavelet basis among all existing ones for reversible compression[8]. To improve compression, a prediction stage has been associated to the S transform, leading to the popular S+P method [5], which has been later genalized by the "lifting scheme" concept [9].

Section 2 introduces the adaption of the lossy $WHT_{2\times2}$ to a lossless form called $RWHT$. Section 3 presents the pyramidal lossless compression method based on the $RWHT$. In particular, it relies on two main features successively detailed: a quadtree decomposition and a prediction/interpolation technique. Finally, we will conclude in section 4.

2. THE RWHT TRANSFORM

In order to losslessly recover the input data from the transformed vector, normalisation of the S transform has been modified:

$$WHT_{2\times2} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \quad S = \begin{bmatrix} \frac{1}{2} & 1 \\ \frac{1}{2} & -1 \end{bmatrix}. \qquad (1)$$

Then the unambiguous integers to integers mapping is possible thanks to dual rounding operations during the forward and inverse transformations. 2D $WHT_{2\times2}$ is realized by applying the previous transform along horizontal and vertical directions. However, this transformation kernel is less efficient for lossy compression, as it increases the dynamics of high frequency coefficients.

We introduce here a new technique of lossless 2D $WHT_{2\times2}$ using directly the formal transform matrix.

Let $U_{2\times2}$ be the input block with:

$$U_{2\times2} = \begin{bmatrix} u_0 & u_1 \\ u_2 & u_3 \end{bmatrix}. \quad (2)$$

Then the block transformed $Z_{2\times2}$ is defined by:

$$\begin{aligned} Z_{2\times2} &= WHT_{2\times2}(U_{2\times2}) = W_{2\times2}U_{2\times2}W_{2\times2} = \begin{bmatrix} z_0 & z_1 \\ z_2 & z_3 \end{bmatrix} \\ &= \frac{1}{2}\begin{bmatrix} u_0+u_1+u_2+u_3 & u_0+u_1-u_2-u_3 \\ u_0-u_1+u_2-u_3 & u_0-u_1-u_2+u_3 \end{bmatrix}. \end{aligned} \quad (3)$$

Let be $\hat{Z}_{2\times2}$ the rounding block of $Z_{2\times2}$ such as:

$$\begin{aligned} \hat{Z}_{2\times2} &= Round(Z_{2\times2}) = \begin{bmatrix} \hat{z}_0 & \hat{z}_1 \\ \hat{z}_2 & \hat{z}_3 \end{bmatrix} \\ &= \begin{bmatrix} Round_{z_0}[z_0] & Round_{z_1}[z_1] \\ Round_{z_2}[z_2] & Round_{z_3}[z_3] \end{bmatrix}. \end{aligned} \quad (4)$$

$Round_{z_i}[.]$ stands for the rounding operator applied on z_i which can be either downward ($\lfloor . \rfloor$) or upward ($\lceil . \rceil$) rounding.

The inverse transform is identical to the forward one. $\tilde{U}_{2\times2}$ denotes the inverse block transformed of $\hat{Z}_{2\times2}$, and $\hat{U}_{2\times2}$ the rounded block of $\tilde{U}_{2\times2}$. Defining a lossless transform implies $\hat{U}_{2\times2} = U_{2\times2}$ despite the rounding operations. To achieve that directly in the 2D space, we propose a method to control the rounding values based on the following parity function $P(.)$:

$$P(x) = \begin{cases} o \text{ if } x \text{ odd} \\ e \text{ if } x \text{ even} \end{cases}, \ x \in \mathbb{N}. \quad (5)$$

Asserting $z_0 = \lfloor z_0 \rfloor + \frac{\varepsilon}{2}, \varepsilon \in \{0,1\}$, and when substituting it in equation (3), $Z_{2\times2}$ can be expressed as follows:

$$Z_{2\times2} = \frac{1}{2}\begin{bmatrix} 2\lfloor z_0 \rfloor + \varepsilon & 2(\lfloor z_0 \rfloor - u_2 - u_3) + \varepsilon \\ 2(\lfloor z_0 \rfloor - u_1 - u_3) + \varepsilon & 2(\lfloor z_0 \rfloor - u_1 - u_2) + \varepsilon \end{bmatrix}. \quad (6)$$

As this point, there are two cases.

Even sum: If $P(\sum_{i=0}^{3} u_i) = e$, then $\varepsilon = 0$ and $\hat{Z}_{2\times2} = Z_{2\times2}$. It also leads to integer reconstructed values:

$$\tilde{u}_0 = \frac{1}{2}(4\lfloor z_0 \rfloor - 2(u_1+u_2+u_3)) = \frac{1}{2}(2u_0) = u_0 \quad (7)$$

and $\hat{u}_0 = u_0$.

Odd sum: If $P(\sum_{i=0}^{3} u_i) = o$. The problem of rounding $Z_{2\times2}$ is shifted to the problem of rounding $\varepsilon/2$ for each coefficient. Let $\Delta_i \in \{0,1\}$ be the rounding of $\varepsilon/2$ for z_i ($\Delta_i = Round_{z_i}\left[\frac{\varepsilon}{2}\right] = \frac{\varepsilon}{2} + \frac{\varepsilon_i}{2}, \varepsilon_i \in \{-1,+1\}$).

$$\begin{aligned} \hat{Z}_{2\times2} &= \begin{bmatrix} \lfloor z_0 \rfloor + \Delta_0 & \lfloor z_0 \rfloor - u_2 - u_3 + \Delta_1 \\ \lfloor z_0 \rfloor - u_1 - u_3 + \Delta_2 & \lfloor z_0 \rfloor - u_1 - u_2 + \Delta_3 \end{bmatrix} \\ &= Z_{2\times2} + \frac{1}{2}\begin{bmatrix} \varepsilon_0 & \varepsilon_1 \\ \varepsilon_2 & \varepsilon_3 \end{bmatrix}. \end{aligned} \quad (8)$$

Consequently the rebuild coefficients are expressed:

$$\tilde{U}_{2\times2} = \frac{1}{2}\begin{bmatrix} 2(u_0-\varepsilon)+(\Delta_0+\Delta_1+\Delta_2+\Delta_3) & 2u_1+(\Delta_0+\Delta_1-\Delta_2-\Delta_3) \\ 2u_2+(\Delta_0-\Delta_1+\Delta_2-\Delta_3) & 2u_3+(\Delta_0-\Delta_1-\Delta_2+\Delta_3) \end{bmatrix}. \quad (9)$$

Therefore, the correct reconstruction implies:

$$\begin{cases} \Delta_0+\Delta_1+\Delta_2+\Delta_3 = 2\varepsilon = 2 \\ \Delta_0+\Delta_2 = \Delta_1+\Delta_3 \\ \Delta_0+\Delta_1 = \Delta_2+\Delta_3 \\ \Delta_0+\Delta_3 = \Delta_1+\Delta_2 \end{cases} \quad (10)$$

Clearly the equations system on Δ_i values cannot be solved and no systematic rounding, as for the S transform, enables an exact reconstruction.

The alternative relies in controlling the rounding operations, i.e. the decoding process can distinguish from integer to non integer reconstructed value. Fixing $\{\Delta_i\}$ so that $P(\sum_{i=0}^{3}\Delta_i) = o$, produces only real values for $\tilde{u}_i$ coefficients. If we consider that $\sum_{i=0}^{3}\Delta_i = 1$, then:

$$\begin{aligned} \Delta_0+\Delta_1+\Delta_2+\Delta_3 = 1 \quad &\Rightarrow 4\frac{\varepsilon}{2} + \frac{\varepsilon_1}{2} + \frac{\varepsilon_2}{2} + \frac{\varepsilon_3}{2} + \frac{\varepsilon_3}{2} = 1 \\ &\Rightarrow \varepsilon_0+\varepsilon_1+\varepsilon_2+\varepsilon_3 = -2. \end{aligned} \quad (11)$$

For instance, the set $\{\varepsilon_0 = 1, \ \varepsilon_1 = \varepsilon_2 = \varepsilon_3 = -1\}$ is one solution for condition in (11). With a such choice, the inverse transform is finally realized in two simple steps:

1. Compute $\tilde{U}_{2\times2} = WHT(\hat{Z}_{2\times2})$.
2. If $\tilde{u}_i$ is real, then compute new $\tilde{U}_{2\times2}$ such as:

$$\tilde{U}_{2\times2} = WHT\left(\hat{Z}_{2\times2} - \frac{1}{2}\begin{bmatrix} 1 & -1 \\ -1 & -1 \end{bmatrix}\right). \quad (12)$$

It can be easily verified that $\tilde{U}_{2\times2} = U_{2\times2}$ in all cases.

3. LOSSLESS CODING WITH THE RWHT+P PYRAMID

Notations: $I(i,j)$ denotes the pixel in an image I with the coordinates (i,j), $I(\mathbf{b}^N(i,j))$ the block $\mathbf{b}^N(i,j)$ in I including the set of pixels $\{I(N.i,N.j),\dots I(N.i+N-1,N.j+N-1)\}$.

3.1 The RWHT pyramid

We introduce the pyramid $\{Y_l\}_{l=0}^{L_{max}}$ as the multiresolution representation of an image I of size $N_x \times N_y$, where L_{max} is the top of the pyramid and $l = 0$ the full resolution image. As for the classical $WHT_{2\times2}$ case, we iteratively construct the pyramid gathering four blocks to form a mean block at the upper level:

$$\left| \begin{array}{ll} l = 0, & Y_0(i,j) = I(i,j); \\ l > 0, & Y_l(i,j) = \left\lfloor \frac{1}{4}\sum_{k=0}^{1}\sum_{m=0}^{1} Y_{l-1}(2x+k, 2y+m) \right\rfloor \end{array} \right. \quad (13)$$

with $0 \le i \le N_x^l$, $0 \le j \le N_y^l$, where $N_x^l = N_x/2^l$ and $N_y^l = N_y/2^l$.

The top-down decomposition of the pyramid consists in encoding the $RWHT$ block transformed $Z_l(\mathbf{b}^2(i,j))$ of each input block $Y_l(\mathbf{b}^2(i,j))$. From (3) and (13), we get:

$$\begin{array}{l} Y_{l+1}(i,j) = \left\lfloor \frac{z_{0_l}(2i,2j)}{2} \right\rfloor \\ \Rightarrow z_{0_l}(2i,2j) = 2 \times Y_{l+1}(i,j) + \varepsilon_{z_{0_l}(2i,2j)}, \varepsilon_{z_{0_l}(2i,2j)} \in \{0,1\}. \end{array} \quad (14)$$

Then, the DC component of each block is unambiguously reconstructed from the upper level plus an additional bit. This bit is separately encoded from the other coefficients. Let $\dot{Z}_l(\mathbf{b}^2(i,j))$ be the WHT transformed block of $Y_l(\mathbf{b}^2(i,j))$ with only this bit as continuous component ($\dot{z}_{0_l} = \varepsilon_{z_{0_l}}$), then the reconstruction from the previous level and current WHT transform is given by:

$$\begin{array}{l} \tilde{Y}_l(\mathbf{b}^2(i,j)) = EXPAND(Y_{l+1}(i,j)) + \tilde{\tilde{Y}}_l(\mathbf{b}^2(i,j)) \\ \text{with } \tilde{\tilde{Y}}_l(\mathbf{b}^2(i,j)) = WHT_{2\times 2}^{-1}\left(\dot{Z}_l\left(\mathbf{b}^2(i,j)\right)\right). \end{array} \quad (15)$$

The expand function only duplicates a node value in the tree to its four sons.

At this step, we get a common pyramidal representation and encoding based on $WHT_{2\times 2}$ transforms, but with the exception of a possible lossless decomposition. Table 1 gives zero-order entropy values for lossless compression with both the S and the proposed RWHT transforms. Top level has been encoded for the two methods by a simple DPCM. Figures show that the proposed method gives compression improvements while generalising the lossy 2D WHT kernel to lossless coding.

3.2 Quadtree decomposition

In some previous works, we have investigated some coding schemes based on variable block size representations leading to efficient compression at both low and high bit rates [1]. We will demonstrate that this concept in a lossless coding context also provides significant improvements.

A quadtree partition implies that the whole image is split into squares of size $N \times N$, with $N = 2^k$ and k a positive integer. The previous pyramid representation involving a diadic decomposition is generally associated to a multilevel quatree partition $QP^{[2^{L_{max}} \ldots 2^l]}$ where the level l of the pyramid specifies also the finest resolution. More generally, we consider a global quadtree partition of the image $QP^{[N_{max} \ldots N_{min}]}$ defining allowed block sizes, and the parameter $N_l \in [N_{max} \ldots N_{min}]$ giving the upper limit of block sizes to be decomposed at level l of the pyramid. For instance, a global partition $QP^{[32 \ldots 2]}$ leads to encode only the representation from sizes 32 to 2, while $N_0 = 4$ means that blocks of sizes 4 and 2 will be decomposed at level 0.

Finally, a last decomposition paramater L_{min} specifies the last level to be encoded: for all level lower than L_{min}, value of all nodes in the pyramid are only expanded.

The image partition is constructed from the local activity, which is estimated by a morphological gradient (difference between *max* and *min* values) within blocks. Then, the first decomposition pass in the pyramid consists in refining only small blocks located on contours, according to the following expression:

$$\tilde{Y}_l(\mathbf{b}^2(i,j)) = \begin{cases} EXPAND(Y_{l+1}(i,j)) + \tilde{\tilde{Y}}_l(\mathbf{b}^2(i,j)), \\ \quad \text{if } \mathbf{b}^2(i,j) \notin QP^{[N_{max} \ldots N_l[} \text{ } and\ l \geq L_{min} \\ EXPAND(Y_{l+1}(i,j)) \text{ otherwise} \end{cases}$$
$$\text{with } l < L_{max} \quad (16)$$

$\tilde{Y}_l(\mathbf{b}^2(i,j))$ stands for the reconstructed blocks of $Y_l(\mathbf{b}^2(i,j))$.

Figure 1 gives the global coding on this model for the first pass, called C_1 coder.

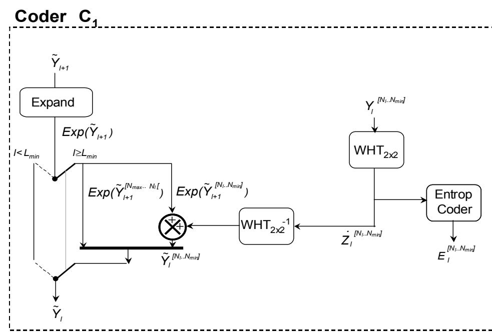


Figure 1: Simple pyramidal coder

The second pyramid pass consists in decomposing all blocks at the current level which have not be encoded during the first pass, processing local texture information.

The quadtree decomposition use has several advantages:

1. it doubles the number of decomposition levels ($2 \times L_{max}$), increasing the scalability,
2. good quality images are available at low bit rates,
3. the approach acts as an "objective context modeling", decorrelating error prediction laws between high entropy features during the first pass, and low entropy ones during the final pass.

3.3 Prediction in the RWHT pyramid

Prediction and interpolation are closed functions in the spatial domain but with different objectives: the first one tries to optimize the compression by limiting the prediction error, while the second one tends to increase image quality and resolution. A good predictor does not necessarily leads to a good interpolator and vice-versa. Both functions are useful in our coding scheme. In particular, a decomposition level during the first pass require both prediction for decomposed blocks to encode and interpolation for the other blocks to smooth homogeneous areas. Therefore, we propose an unified framework for the two functions as an unique estimation process. In the following, $\breve{Y}_l(\mathbf{b}^2(i,j))$ will denote the reconstructed blocks of $Y_l(\mathbf{b}^2(i,j))$. Then, the estimation process consists in a linear rebuilding of unknown values around their block mean, exploiting both inter and intra levels information

	Entropy (bpp)						
Image	Raw	S	RWHT	CALIC	S+P	RWHT+P	RWHT+P Qd
Barbara2	7.51	5.45	5.47	4.93	5.04	5.06	**4.89**
Hotel	7.57	5.11	5.09	**4.57**	4.97	4.83	4.60
Lena	7.44	4.77	4.75	4.33	4.33	4.30	**4.19**
Gold	7.60	5.08	5.06	4.65	4.73	4.73	**4.63**
Peppers	7.57	4.89	4.87	4.58	4.67	4.54	**4.43**
us	4.84	3.65	3.64	3.60	3.78	3.78	**3.26**
tools	7.52	5.95	5.95	5.53	5.73	5.71	**5.50**
Average	**7.15**	4.99	4.97	4.60	4.75	4.71	**4.49**

Table 1: Comparison of the proposed approaches with state-of-the-art methods. First-order entropy (bit/pixels).

in a 2D context.

$$\begin{array}{|l} \mathit{Initialization}: \\ \quad \breve{Y}_l(\mathbf{b}^2(i,j)) = \tilde{Y}_{l+1}(i,j), \forall (i,j) \in \tilde{Y}_{l+1} \\ \mathit{Estimation}: \\ \quad \breve{Y}_l(2i+k, 2j+m) = \\ \tilde{Y}_{l+1}(i,j) + \beta_m \left(\breve{Y}_l(2i+k, 2j-1+3m) - \tilde{Y}_{l+1}(i,j)\right) \\ +\beta_k \left(\breve{Y}_l(2i-1+3k, 2j+m) - \tilde{Y}_{l+1}(i,j)\right), (k,m) \in \{0,1\}^2 . \end{array} \quad (17)$$

β_m and β_k the weights applied on the local gradients.

Without any quantization step, the neighbour value differs depending on the configuration and corresponds to:

- an exactly reconstructed value (already processed position at the current level with exact coding),
- a block mean value (not yet processed position at the current level),
- an interpolated value (previous processed position but not encoded).

In this last case, an inter-dependency exists as the neighbour value has been partially computed from the current block mean. It induces a relationship between the β coefficients for two adjacent positions of two blocks. For instance, for two close positions $(2i, 2j)$ and $(2i-1, 2j)$, and looking only at horizontal relationships, expression 17 gives:

$$\begin{array}{|l} \breve{Y}_l(2i,2j) = \\ \qquad \tilde{Y}_{l+1}(i,j) + \beta_0 \left(\breve{Y}_l(2i-1,2j) - \tilde{Y}_{l+1}(i,j)\right) \\ \breve{Y}_l(2i-1,2j) = \\ \qquad \tilde{Y}_{l+1}(i-1,j) + \beta_1 \left(\breve{Y}_l(2i,2j) - \tilde{Y}_{l+1}(i-1,j)\right) \\ \Rightarrow \breve{Y}_l(2i,2j) = \\ \qquad \tilde{Y}_{l+1}(i,j) + \beta_0 \left(\tilde{Y}_{l+1}(i,j) - \tilde{Y}_{l+1}(i-1,j)\right)(\beta_1 - 1) \end{array} \quad (18)$$

Moreover, if we impose a symmetrical gradient such as:

$$\left(\breve{Y}_l(2i-1,2j) - \tilde{Y}_{l+1}(i-1,j)\right) = -\left(\breve{Y}_l(2i,2j) - \tilde{Y}_{l+1}(i,j)\right), \quad (19)$$

it leads to the relationship:

$$\beta_0 = \frac{\beta_1}{1-\beta_1}, \beta_1 \in [0, 0.5]. \quad (20)$$

The estimation effect can be calibrated by β_1 value.

- for $\beta_1 = 0$, $\breve{Y}_l(2i,2j) = \tilde{Y}_{l+1}(i,j)$: estimation has no effect (block rebuild by its average value),
- for $\beta_1 = 0.25$, $\breve{Y}_l(2i,2j) - \tilde{Y}_{l+1}(i,j) = \tilde{Y}_{l+1}(i-1,j) - \breve{Y}_l(2i-1,2j)$: the slot is regular between the two reconstructed points (smoothes the image),
- for $\beta_1 = 0.5$, $\breve{Y}_l(2i,2j) = \breve{Y}_l(2i-1,2j)$: adjacent reconstructed points are identical (accentuates the contours).

Actually, the smoothing mode ($\beta_1 = 0.25$) provides the best prediction in our experiments.

Figure 2 shows new coding schemes including the estimation stage. C_2 coder uses only inter-level relationships and can be useful for progressive reconstruction (resolution enhancement can be directly done from a previously interpolated image from the previous level). C_3 coder capitalizes also reconstruted values at the current level, and obviously leads to better compression performances.

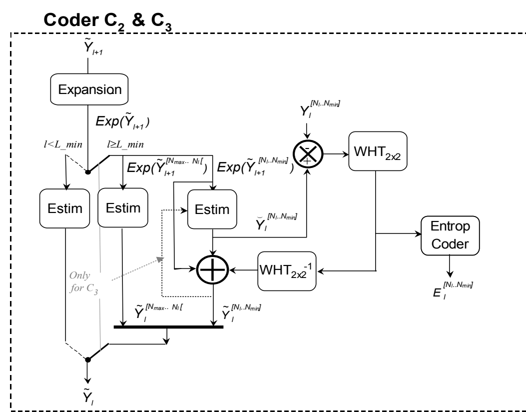


Figure 2: Pyramidal coder with prediction step

Lossless compression results are given in table 1. We have compared the proposed method with the state-of-the-art "CALIC" (non scalable) and "S+P" (scalable). The choice of "S+P" instead of another integer wavelets kernel has been motivated by the fact it remains one of the best and also because an open source coder exists and enables reliable comparisons before the entropic coding layer.

The "RWHT+P" configuration corresponds to the C_3 mode without partition (only one pass for the decomposition). "RWHT+P & Qd" involves a partition $QP^{[64...2]}$ with $N_0 = 2$ and $N_l = 2^l$ otherwise. We can notice that the laws separation for symbols to encode occuring with the quadtree decomposition, widely offsets the coding cost of its structure: this configuration outperforms both "S+P"and "CALIC".

To illustrate the "semantic" scalabity of our approach, figure 3 shows some intermediate rebuild images from initial to final steps. For six levels of decomposition, the whole lossless image encoding implies eleven sucessive streams (

a) **First pass**, $\mathbf{L_{min} = 5}$
(1 stream): 0.005 bpp

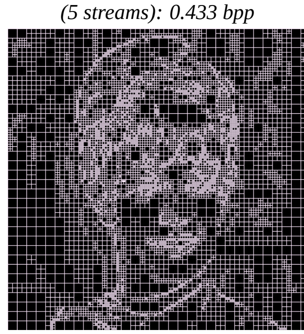
b) **First pass**, $\mathbf{L_{min} = 3}$
(3 streams): 0.073 bpp

c) **First pass**, $\mathbf{L_{min} = 1}$
(5 streams): 0.433 bpp

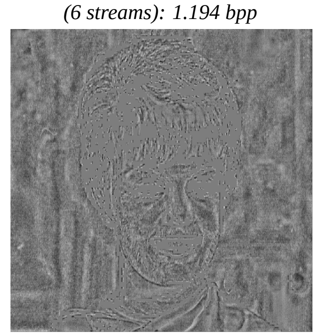
d) **First pass**, $\mathbf{L_{min} = 0}$
(6 streams): 1.194 bpp

e) **Second pass**, $\mathbf{L_{min} = 1}$
(10 streams): 2.014 bpp

f) **Second pass**, $\mathbf{L_{min} = 0}$
(11 streams): 3.87 bpp



g) **Quadtree partition**



h) **Error image x10**
Original (f) - image (d)

Figure 3: Scalable lossless coding on "Zelda" with partition $QP^{[64...2]}$

$1+2\times5$). It is noticeable that distorsion is essentially due to a blurring effect, which is less disturbing than blocks or ringing artifacts. At the end of the first pass, rebuid images present perfect accuracy upon strong contours while homogeneous areas appears as smooth regions.

4. CONCLUSION

The S transform has been primarily developped to introduce reversibility in the classic $WHT_{2\times2}$. The first part of this paper has demonstrated that the usual kernel can also offer this ability with rounding operations based on a parity criteria. The proposed *RWHT* pyramidal decomposition offer slightly better lossless compression performances than S.

Two main other originalities have been also presented leading to additional decorrelation, and further significant improvements: a content-based pyramidal decomposition, and a prediction step. The global scalable coding scheme surpasses both S+P and CALIC. It also provides locally adaptive multiresolution representations for good quality images rebuilding at low bit rates.

Actually, this coding scheme has also proved to be efficient for lossy low bit rates compression by introducing errors quantization adapted to the content: fine quantization for large blocks (human eye more sensitive to luminance differences in uniform areas) and coarse quantization for small blocks (human eye less sensitive upon contours).

One direct application of this work is definition of an archiving system for high resolution art pictures of the Louvre's museum. This digital library will provide a selective access with different quality of images (project "TSAR" supported by the French ministry of research).

REFERENCES

[1] O. Deforges and J. Ronsin, "Region of interest coding for low bit-rate image transmission," in *ICME*, july 2000, vol. 1, pp. 107–110.

[2] O. Deforges and J. Ronsin, "Supervised segmentation at low bit rates for region representation and color image compression," in *ICME*, 2002, vol. 1, pp. 665–668.

[3] M. Babel, O. Deforges, and J. Ronsin, "Interleaved s+p pyramidal decomposition with refined prediction model," in *ICIP*, 2005, vol. 2, pp. 750–753.

[4] X. Wu and N. Memon and, "A Context-based, Adaptive, Lossless/Nearly-Lossless Coding Scheme for Continuous-Tone Images (CALIC)," *International Standards Organization working document, ISO/IEC SC29/WG 1/N256*, 1995.

[5] A. Said and W. Pearlman, "Reversible image compression via multiresolution representation and predictive coding," in *Visual Communication and Image Processing*. SPIE, Novembre 1993, vol. 209, pp. 664–674.

[6] P. Lux, "A novel set of closed orthogonal functions for picture coding," *Archiv für Electronik und übertragungs Technik*, vol. 31, no. 7, pp. 267–274, 1977.

[7] K. Komatsu and K. Sezaki, "Lossless 2d discrete walsh-hadamard transform," *Proc. IEEE ICASSP*, May 2001.

[8] G. C. K. Abhayaratne, *Lossless and Nearly Lossless Digital Video Coding*, Ph.D. thesis, Univ. of Bath, 2002.

[9] Wim Sweldens, "The Lifting Scheme: A Construction of Second Generation Wavelets," *SIAM Journal on Mathematical Analysis*, vol. 29, no. 2, pp. 511–546, 1998.