LOSSLESS REGION OF INTEREST WITH EMBEDDED WAVELET IMAGE CODING

David Nister and Charilaos Christopoulos
Ericsson Telecom AB
TN/ETX/PN/XML, CLAB
126 25 Stockholm, Sweden
Email: {d.nister, ch.christopoulos}@clab.ericsson.se

Abstract

This paper describes a method which use the well known S+P and TT transforms to encode an especially important part an image, a Region Of Interest (ROI) in a lossless mode. Other parts of the image (the Background) are given lower quality levels allowing higher compression. The ROI coding is done in the framework of an embedded wavelet based image compression algorithm. All coding, regional and full image, is done in a naturally progressive manner all the way up to lossless.

1. Introduction

When handling digital images the main concern should of course always be the quality of the image delivered to the user. However quality should be achieved with as compact representation of the image as possible. The research efforts in the field of digital image compression have been immense over many years. One possible way to divide the techniques is into two categories, lossy and lossless algorithms.

The more common lossy algorithms take advantage of the fact that in most cases small losses can be tolerated. This gives more freedom on the choice of algorithm and higher compression efficiency depending on how much losses are inferred.

Lossless compression is used for high demanding applications where no loss of quality can be tolerated. However this requirement will make the compression performance fairly modest.

Recently lossly and lossless algorithms have partially merged. This is due to new schemes that combine the two without any substantial loss of performance for either mode [4], [9]. This is done inside the framework of the recently very popular schemes which produce an embedded bitstream [1], [3], [8], [9], [10]. An embedded bitstream has the property that the information is encoded in the order of its importance. This means that with one single bitstream the encoder performs its best independently of where the bitstream is truncated. It is therefore possible to hit a specific bitrate or to easily provide different users with different quality images depending on their demands and access link. The embedded bitstream will also make the transmission progressive so that the receiver can use the transmitted information to display an image of gradually improving quality while receiving. It should be pointed out however that which information is the most important is not always an easy choice and sometimes a highly subjective one. For example in some applications or for some users different regions of an image might have different importance. Therefore it might be desirable to have a representation which is not only compact but also selective [5].

This paper presents an image compression algorithm which achieves progressive encoding and also different quality levels on different parts of the image, specifically an arbitrary region of interest (ROI) in the image can be encoded up to lossless.
The different quality levels represent a rate distortion trade-off. For example lossless quality might be required for a small part of the image where the users real interest is located. The rest of the image which is only important in a contextual sense is then assigned a lower quality. This can maintain a fairly high compression while meeting the lossless requirement.
The algorithm is based on the completely reversible integer wavelet transforms S [6], S+P [2] and Two Ten (TT) [9]. The transform coefficients are then encoded in an embedded fashion. The idea of embedded coding originated with wavelet transform and zero-tree coding [1]. Embedded coding can also be done in a DCT framework [8]. However when lossless encoding is desired, reversible wavelet transforms is the method of choice, at least if the encoding should be naturally progressive all the way.
Otherwise another option is to use residual coding [5], where a lossy algorithm is first used to generate an approximation of the original image and then a simple or more elaborate lossless algorithm is used to encode the remaining differences. This however puts restrictions also on the lossy part of the scheme since the lossy reconstruction have to be the same on all platforms. This means that exactly the same accuracy has to be used for all decoders. In software this is equivalent to using integer implementation.
It should be noted that although it should be desirable in the same range of applications, there is a technical difference between achieving a ROI with better quality than the background and achieving a lossless ROI. In the former case the coefficients affecting the ROI the most can be improved upon to give it a better quality but when lossless encoding is the goal, all coefficients with a possibility of affecting the ROI should be reversibly encoded and should preferably be as few as possible.

The proposed scheme solves the ROI problem in such a way that:

The coding is naturally progressive and without switching algorithm (no residual coding as in [7]).

- The ROI or ROI’ s can have arbitrary shape.
- The quality of the ROI is guaranteed and then the quality of the surroundings is degrading gracefully so that there are no visually annoying edges around the ROI.
- The functionality of the solution is not dependent on the entropy coding scheme. It is also independent of the method used for coding the shape of the ROI’ s

The paper is organized as follows: Section 2 describes the method as it was implemented. In section 2.1, an algorithm overview is given. Section 2.2 is a review of the S, S+P and TT transforms. Section 2.3 describes how to derive a mask so that these transforms can be used for progressive coding of images with perfect reconstruction of selected ROI’s. Section 2.4 describes the normalization of the transform. Sections 2.5 and 2.6 give information on the progressive coding of the coefficients with modifications of the method described in [3]. Section 3 gives results and comparisons. Section 4 is a short discussion. Conclusions are drawn in section 5.

2. Method description

2.1 *Algorithm Overview*

The following steps are taken in the algorithm:

1. Calculate the S, S+P or TT transform of the image and find its normalization.
2. If a ROI is chosen, a mask is derived, indicating a set of coefficients in the transform which is sufficient for lossless ROI reconstruction.
3. Transmit the transform coefficients progressively, with the most important information first. Enough information for the ROI can be encoded, while less for the background.
4. Entropy encode the resulting symbol stream.

The decoder then reverses these steps to reconstruct the image.

2.2 *Transformation*

The transforms used in this work are the S, the S+P and the TT transforms. They are all completely reversible pyramid subband decompositions and can be done in place without memory expansion. The S transform [6] is essentially an integer Haar wavelet. The other two can be considered to consist of the S transform and of a prediction used to take out the remaining redundancies from the high frequency subbands. The prediction is done because the S transform is to simple to decorrelate the coefficients well enough and therefore leaves blocking artifacts in the reconstructed images. The S+P was first presented in [2]. The TT is described in [9]. A comparison of several integer wavelet transforms can be found in [13].
The forward transformation is done by applying a subband decomposition several times. The inverse is found by applying the corresponding compositions in reverse order. The decomposition is applied on one line or column $\{X(n): n \in [0..N-1]\}$ (vector of integers where N is even) at a time to give the outputs $\left\{L(n): n \in \left[0..\frac{N}{2}-1\right]\right\}$ and $\left\{H(n): n \in \left[0..\frac{N}{2}-1\right]\right\}$, which are the low and high frequency subbands respectively. The operations for the S transform alone are:

Decomposition (forward transform):

$$L(n) = \left\lfloor \frac{X(2n) + X(2n+1)}{2} \right\rfloor \quad (1)$$

$$H(n) = X(2n) - X(2n+1) \quad (2)$$

Where $\lfloor . \rfloor$ means downward truncation.

Composition (inverse transform):

$$X(2n) = L(n) + \left\lfloor \frac{H(n)+1}{2} \right\rfloor \quad (3)$$

$$X(2n+1) = X(2n) - H(n) \quad (4)$$

On top of this, to remove the remaining redundancies in the high frequency bands, a truncated prediction is made [2] and the high frequency coefficients replaced by the prediction error as:

$$H_d(n) = H(n) - \lfloor P(n) \rfloor \quad (5)$$

Here $H_d(n)$ is the prediction error and $P(n)$ the prediction of the high frequency coefficient $H(n)$. To include both the S+P and Two-Ten in the same framework the general prediction is:

$$P(n) = a * L(n-2) + b * L(n-1) + c * L(n) + d * L(n+1) + e * L(n+2) + f * H(n+1) + g \quad (6)$$

The parameters $a - f$ can be chosen differently, depending on what kind of image statistics are expected and what kind of filter properties are desired.

Table 1 The predictors suggested in [2] and [9]

	S	S+P (A)	S+P (B)	S+P (C)	S+P (D)	S+P (E)	TT
a	0	0	0	0	0	0	-3 / 64
b	0	1 / 4	4 / 16	2 / 8	3 / 16	3 / 16	22 / 64
c	0	0	1 / 16	1 / 8	5 / 16	6 / 16	0
d	0	-1 / 4	-5 / 16	-3 / 8	-8 / 16	-9 / 16	-22 / 64
e	0	0	0	0	0	0	3 / 64
f	0	0	2 / 16	2 / 8	6 / 16	8 / 16	0
g	0	0	0	0	0	0	-32 / 64

Table 1 shows seven different predictors. The first one corresponds to the S transform (no prediction). The next five are the predictors A-E suggested in [2] for the S+P transform. The last one corresponds to the TT transform as described in [9] with the exception that the direction of the truncation is reversed.

At the subband edges where some coefficients in the predictor might be missing, a shorter predictor has to be used. The border predictors suggested in [2] can be used or a predictor can be found by thinking in terms of symmetric extension [14].

In the proposed scheme, the predictors called S, S+P (A) and TT are used because they all have the property $f = 0$, which makes it possible to achieve a lossless reconstruction of an arbitrarily shaped region in the image as explained in section 2.3.

The decomposition including prediction, is applied successively to first all the image lines separately and then all the columns separately. This produces a four band structure corresponding to low and high horizontal and vertical frequencies (see **Figure 1**).

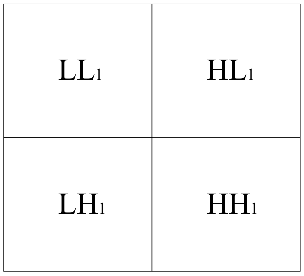

Figure 1 Four band structure of the transform

The four band decomposition is then recursively applied to the LL subband to produce a dyadic structure (**Figure 2**). Finally a constant is subtracted from the coefficients in the LL band so that their possible range is centered around 0. Alternatively this can be done to the whole image before the transformation.

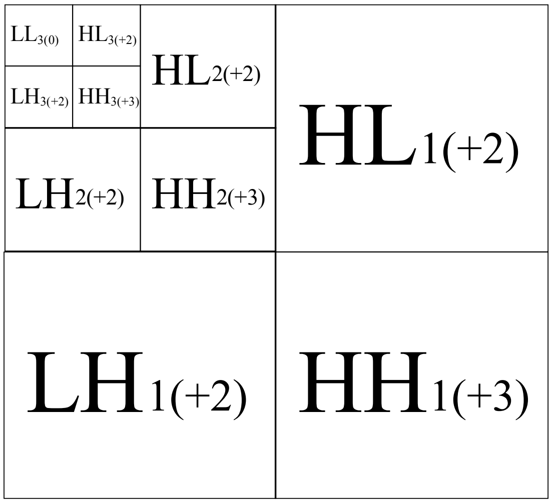

Figure 2 Subband structure of the transform. The number of extra bits (compared to the original image) necessary for the representation of the S+P and TT are shown in parenthesis

Notice that the arithmetic in the transform yields output with a few more bits needed for its representation than the original image. It is however not a problem since for a typical image the first order entropy will decrease independently of this.

2.3 Deriving the lossless mask for the transforms S, S+P(A) and TT

To achieve a perfectly reconstructed ROI, while maintaining a fair amount of compression, bits need to be saved by sending less information for the background. To do this without damaging the ROI, a lossless mask is calculated. The mask is a bit plane indicating a set of wavelet coefficients whose exact transmission is sufficient in order for the receiver to reconstruct the desired region perfectly. Thus for a specific ROI the semantics of the mask are:

$$M(x,y)=\begin{cases}1 & \textit{The wavelet coefficient } (x,y) \textit{ should be transmitted exactly}\\ 0 & \textit{Accuracy on } (x,y) \textit{ can be sacrificed without affecting ROI}\end{cases} \qquad (7)$$

In the case that there is a ROI in the image which is chosen to be lossless, one of the predictors S, S+P(A) or TT in **Table 1** should be used. This is because when these predictors are used there is no prediction of high frequencies done with the help of high frequencies. If high frequencies were used, it would produce a closed loop and an error in a coefficient outside the ROI could propagate to a coefficient inside. Thus all coefficients in the decomposition would have to be lossless (which is not our goal) or else appropriate initial prediction conditions must be supplied for the arbitrarily shaped region [5].

The mask is derived following the same steps as the forward transform (or actually tracing the inverse transform backwards). The mask thereby gets the same subband structure as the wavelet transform. To start out with, the mask is a map of the ROI in the image domain, so that it is 1 inside the ROI and 0 outside. The LL subband of the mask is then in each step updated line by line and then column by column. The mask will then indicate which coefficients are needed exactly at this step so that the inverse transform will reproduce the coefficients of the previous mask exactly. Therefore the total ROI area marked in the mask has to grow slightly for each step due to the prediction in the S+P and TT transforms. **Figure 3** shows how the lossless mask is calculated.

For example, the last step of the inverse S+P is a composition of two subbands. Then to trace this step backwards, the coefficients in the two subbands that are needed exactly are found. The step before that is a composition of four subbands into two. To trace this step backwards, the coefficients in the four subbands that are needed to give a perfect reconstruction of the coefficients included in the mask for two subbands are found.

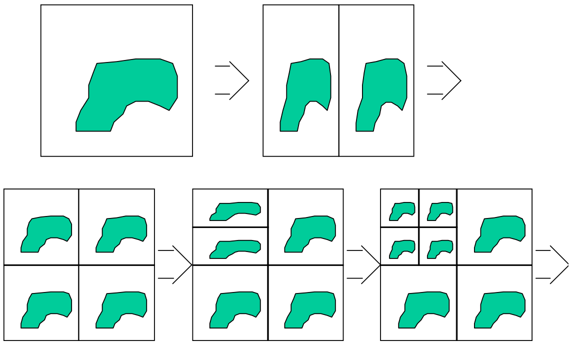

Figure 3 Calculating the lossless mask

All steps are then traced backwards to give a mask implicating that if the coefficients corresponding to the mask are transmitted and received exactly, and the inverse transform calculated on them, the desired ROI will be reconstructed perfectly.

To trace a step backwards on a separate line, let $\{X_m(n): n \in [0..N-1]\}$ be the mask before the step inversion and $\left\{L_m(n): n \in \left[0..\frac{N}{2}-1\right]\right\}$ and $\left\{H_m(n): n \in \left[0..\frac{N}{2}-1\right]\right\}$ the masks for the low and high frequency subband afterwards.

Define:

$$OR\{X_1, X_2, \ldots, X_n\} = (X_1 = 1) \vee (X_2 = 1) \vee \ldots \vee (X_n = 1) \qquad (8)$$

where $\vee$ denotes logical or.

Define also:

$$P(n) = OR\{X_m(2n), X_m(2n+1)\} \qquad (9)$$

The operations below are applied to perform the step inversion (**Figure 4-Figure 6**):

For the S transform

For all n in $\left[0..\frac{N}{2}-1\right]$:

$$H_m(n) = L_m(n) = \begin{cases} 1 & \mathit{If} \quad P(n) \\ 0 & \mathit{otherwise} \end{cases} \qquad (10)$$

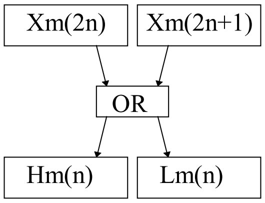

Figure 4 Lossless mask operation for S

For the S+P transform (with the A predictor)

For all n in $\left[0..\frac{N}{2}-1\right]$:

$$H_m(n) = \begin{cases} 1 & \mathit{If} \quad P(n) \\ 0 & \mathit{otherwise} \end{cases} \qquad (11)$$

$$L_m(n) = \begin{cases} 1 & \mathit{If} \quad OR\{P(n-1), P(n), P(n+1)\} \\ 0 & \mathit{otherwise} \end{cases} \qquad (12)$$

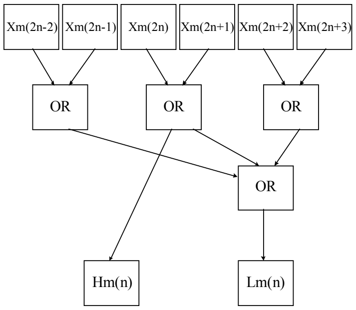

Figure 5 Lossless mask operation for S+P

For the TT transform:

For all n in $\left[0..\frac{N}{2}-1\right]$:

$$H_m(n) = \begin{cases} 1 & \mathit{If} \quad P(n) \\ 0 & \mathit{otherwise} \end{cases} \tag{13}$$

$$L_m(n) = \begin{cases} 1 & \mathit{If} \quad OR\{P(n-2), P(n-1), P(n), P(n+1), P(n+2)\} \\ 0 & \qquad\qquad\qquad \mathit{otherwise} \end{cases} \tag{14}$$

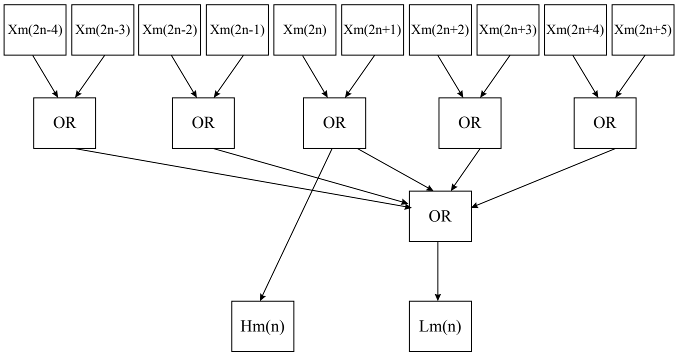

Figure 6 Lossless mask operation for TT

For synchronization, the same mask is found both in the encoder and the decoder.

If the ROI is of an a priori known and simple shape as for example a rectangle, the procedure for deriving the mask could be optimized to give the same result faster. The above method handles the general case with an arbitrary number of arbitrarily shaped ROI´s.

Notice that the ROI area of the mask will be larger for the TT tranform than for the S+P transform which in turn yields a larger mask than the S transform. A compromise therefore has to be found between the decorrelation properties of the transform and the diffusion of ROI information.

Notice also that the mask is dependent on the transform but that the transform is not altered in any way. This means that the mask could be recomputed at any time during the coding to correspond to another ROI. Therefore it is possible for the receiver to specify or change the ROI in a client-server situation.

2.4 Normalization

For the purpose of embedding it is essential that the transform is unitary or at least nearly unitary (when a MSE or SNR criterion is used) see [1]. This is since the coefficients will be coded bit plane by bit plane with the most significant bits first and the relative magnitude of the coefficients thereby suggest their relative importance. On the point of orthogonality, the fact that we expect to decorrelate the incoming images means that the basis functions of the transform are fairly orthogonal at least on the space of images expected for input. Also the fact that all the transforms are very close to the Haar wavelet, which is orthogonal, speaks for this. For a survey on the decorrelation properties of the various transforms see [13].

Consider then the normalization of the coefficients. It is undesirable to multiply by a general floating point number to achieve the normalization. This is because floating point multiplication would ruin the reversibility of the transform. We therefore restrict the normalization to arithmetic shifts (multiplication by powers of two).

The normalization for the S transform is used for all transforms. This is only approximate for the S+P and the TT, but since the scaling of the coefficients is restricted to powers of two (arithmetic shifts), this means no additional approximation. The fact that there is truncation involved in the transform is also disregarded and the transform looked upon as linear.

Consider a coefficient C of the S transform. Let $\|C\|_b$ denote the euclidian norm of the basis function of this coeffient in the image domain. If every C is replaced by $\frac{C}{\|C\|_b}$ the transform will be normalized. To find $\|C\|_b$ for all coefficients in the S transform recall equations (1-4) and disregard the truncation. Due to symmetry

$$\|X(2n+1)\|_b = \|X(2n)\|_b \neq 0 \qquad (15)$$

and using orthogonality we get:

$$\|L(n)\|_b = \sqrt{\frac{\|X(2n)\|_b^2}{2^2} + \frac{\|X(2n+1)\|_b^2}{2^2}} = \frac{\|X(2n)\|_b}{\sqrt{2}} \qquad (16)$$

and

$$\|H(n)\|_b = \sqrt{\|X(2n)\|_b^2 + \|X(2n+1)\|_b^2} = \sqrt{2}\|X(2n)\|_b \qquad (17)$$

which gives

$$\frac{\|H(n)\|_b}{\|L(n)\|_b} = 2 \qquad (18)$$

It is evident from the last equation that if the low frequency coefficients are multiplied by 2 (shifted) for every time the subband decomposition is applied, all the transform coefficients will get the same norm. This is sufficient since it is only the relative importance of a coefficient that is suggested by its magnitude. An example of the number of shifts that are used to scale the transform pyramid are shown in **Figure 7**.

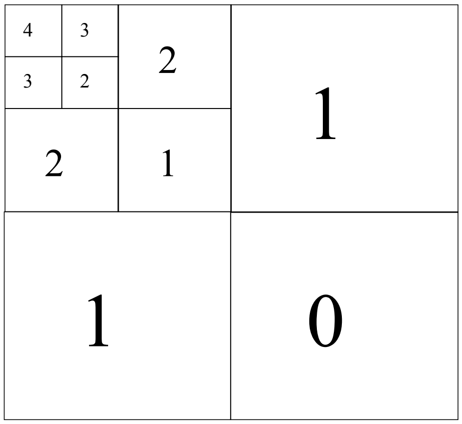

Figure 7 Number of shifts used to scale the transform pyramid

In the case that there is a ROI in the image it might be useful to manipulate the magnitude of the coefficients and thereby also their relative importance. This was first suggested in [11]. The coefficients that correspond to the ROI are then multiplied by a constant $N \geq 1$. The coefficients corresponding to the ROI can be considered to be the ones marked in the ROI mask for the S transform (see 2.3). N will be larger if we think that the ROI is much more important than the rest of the image and smaller all the way down to one when we think that all parts of the image are equally important. Again, when reversibility is desired only shifts are used. This only makes is possible to multiply by a factor of two and will give coarse scaling, but it is essential for the reversibility of the coefficients. In **Figure 8** an example of the number of shifts with an elliptic ROI and $N = 2$ is shown.

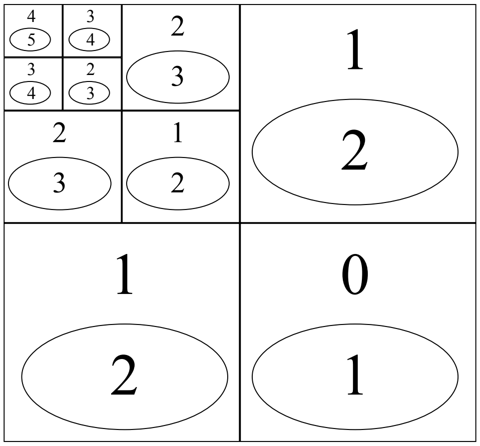

Figure 8 Example of the number of shifts when a ROI is included

2.5 Successive refinement

The progressive transmission is done by bitplanes of the transform coefficients, with the most significant bitplane first. This is motivated by the fact that the transform is orthonormal or at least nearly orthonormal which means that it preserves the energy. In other words, with respect to PSNR or RMSE an error in the transform image of a certain magnitude will produce an error of the same magnitude in the original image.

The compression takes place mainly because a typical image contains large magnitudes of low frequencies but very little of high frequencies. This means that there are very many zeroes in the more significant bitplanes of the coefficients. The task of efficient encoding therefore becomes the task of encoding these zeroes in an efficient way.

When and after the first non zero bit of a certain coefficient is found, that coefficient is said to be significant. This first non zero bit is called the first significant bit (FSB). The bits of a coefficient prior to the FSB will be referred to as the zero bits (ZBs). After the sign bit (SB) remains the fractional bits of the coefficient which will be referred to as the raw bits (RBs). These definitions are illustrated in **Figure 9** and are similar to the ones in [10].

For every zero bit sent, the length of the uncertainty interval for that coefficient seen by the decoder is divided by two. When the first significant bit is encountered, the SB of the coefficient has to be sent to maintain an embedded code. The RBs contain very little redundancy and there is very little to be gained by trying to encode these with a good prediction.

To achieve the lossless ROI the coefficients included in the lossless mask are sent up to their full bit depth while the other coefficients can be skipped after any bit depth.

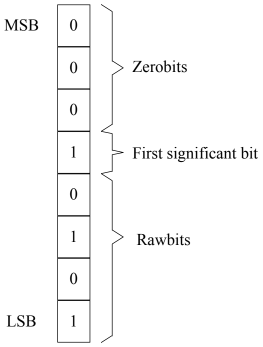

Figure 9 Example of ZBs, FSB and RBs

2.6 Coefficient ordering and coding

To order and code the coefficients the algorithm described in [3] is used. The algorithm is modified so that it is possible to skip coefficients which are not indicated by the lossless mask.

The method is using a zerotree. It keeps the transform coefficients grouped together in set partitions of the tree which are successively divided into smaller and smaller partitions. First the tree structure is described. The direct descendants in the zero tree of the transform coefficient (x, y) (where $x \in [0..image\_size\_x-1]$ and $y \in [0..image\_size\_y-1]$) are (see **Figure 10**):

If (x, y) is in the lowest frequency subband (LL):

$(x+dx, y), (x, y+dy)$ and $(x+dx, y+dy)$ where dx and dy are the lowest frequency subband horizontal and vertical sizes respectively.

Else if $\left(x \geq \frac{image\_size\_x}{2}\right)$ OR $\left(y \geq \frac{image\_size\_y}{2}\right)$:

No descendants exist. This is when the coefficient is in one of the highest frequency subbands and has got no children.

Else:

$(2x, 2y), (2x+1, 2y), (2x, 2y+1)$ and $(2x+1, 2y+1)$

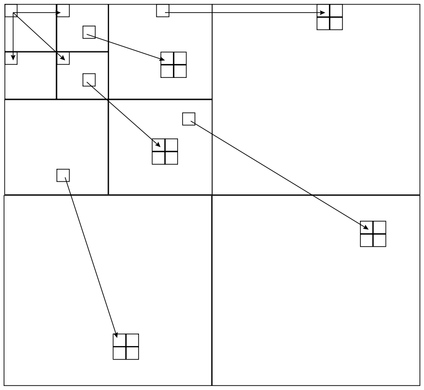

Figure 10 Parental structure of the zero tree

To hold the sets three lists are maintained. The set structure and the lists will be presented below and are similar to the ones in [3].
Some definitions are needed to describe the sets (see **Figure 11**):

$O(x, y)$ = The set of direct offspring of (x, y) (19)

$D(x, y)$ = The set of all the descendants of (x, y) (20)

$L(x, y) = D(x, y) - O(x, y)$ (21)

$OO(x, y)$ = The set of second order offspring of (x, y) (22)

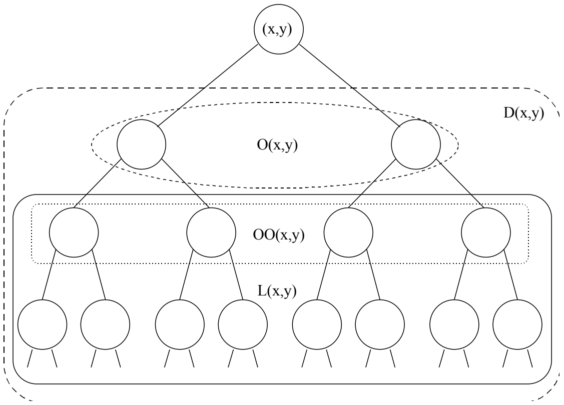

Figure 11 Illustration of the various sets

A set of coefficients is considered significant if it contains any significant coefficients.
The three lists are:

- The list of insignificant pixels (LIP). This contains single coefficients not yet found significant.

- The list of insignificant subsets (LIS). This contains not yet significant subsets of two types, which correspond to the D and L set definitions above.

- The list of significant pixels (LSP). This contains all the single coefficients for which only raw bits remain to be coded.

Define also:

$$RE(x,y) = \begin{cases} 1 & The \quad coefficient \quad (x,y) \quad has \quad remaing \quad bits \\ 0 & (x,y) \quad has \quad reached \quad its \quad full \quad bit \quad depth \quad (lossless) \end{cases} \qquad (23)$$

In the modified coding algorithm list entries are dropped if the sets they correspond to do not contain any necessary coefficients. As a criterion for this the mask $M(x,y)$ is used. In the following it is assumed that first all coefficients are considered interesting so that $M(x,y)=1$ for all (x,y). Then at some later point when enough quality has been reached for the whole image the mask is switched to one derived as above in section 2.3. The correct list entries will then be removed and the coding can be concentrated to the ROI. The algorithm is:

```
1. Empty all the lists. For all coefficients $(x,y)$ in the lowest frequency subband put $(x,y)$ in the LIP and $D(x,y)$ in the LIS. Start with the most significant bitplane.
2. For all $(x,y)$ in the LIP do:
     If $\{M(x,y)=1 \quad AND \quad RE(x,y)=1\}$ then:
         Transmit if $(x,y)$ is significant or not in this bitplane.
         If it is, move it to the LSP and transmit its sign.
     Else:
         Remove $(x,y)$ from the LIP.

3. For all $D(x,y)$ and $L(x,y)$ in the LIS (including new entries) do:
     If $D$ entry:
         If for all $(u,v) \in O(x,y)$, $\{M(u,v)=0\}$ then:
             Remove $D(x,y)$ from the LIS.
         Else:
             Transmit if $D(x,y)$ is significant. If it is:
                 Remove $D(x,y)$ from the LIS.
                 For all $(u,v) \in O(x,y)$ do:
                     Transmit if $(u,v)$ is significant. If it is, transmit its sign and add it to the LSP. Otherwise add it to the end of the LIP.

                 If $L(x,y)$ is non empty, add it to the end of the LIS.

     If $L$ entry:
         If for all $(u,v) \in OO(x,y)$, $\{M(u,v)=0\}$ then:
             Remove $L(x,y)$ from the LIS.
         Else:
             Transmit if $L(x,y)$ is significant. If it is, remove it from
             the LIS and add all the sets $D(u,v)$ such that $(u,v) \in O(x,y)$
             to the end of the LIS.

4. For all $(x,y)$ in the LSP except those included in this bitplane do:
     If $\{RE(x,y)=1\}$ then:
         Transmit the raw bit of $(x,y)$ corresponding to this bit plane.
     Else:
         Remove $(x,y)$ from the LSP.

5. Go to the next bitplane and repeat from step 2 until the desired quality is reached or the bit budget exhausted.
```

Note that to check if a set contains any coefficients in the lossless mask, only the root coefficients in the set have to be checked. This is because if a parent is not included in the mask, none of the the descendants can be either.

2.7 Arithmetic coding

Adaptive arithmetic coding is used to improve the algorithm performance. A binary arithmetic coder implemented according to the guidelines [12] in was used. Separate contexts were used respectively for the sign bits, LIP zero bits, LIS D entry zero bits, LIS L entry zero bits and LSP raw bits. One context for each category could be used, but a slight gain was observed in second order coding, that is using two contexts for each category and to let the choice of which context to use depend on the previous symbol in the category.

2.8 ROI shape

The shape of the ROI should be known by both the encoder and decoder before any list entries can be skipped. In many cases the ROI shape is only bounding, indicating a sufficient part of the image. There is in then no reason to spend too much overhead on coding the ROI shape. However if the shape constitutes useful information in itself like if marks the sharp boundaries of an actual object, it might be worthwhile.

3. Results and comparisons

To check the effectiveness of the proposed way of encoding a lossless region of interest, tests were performed on the JPEG2000 test images, with various ROI's and bitrates. Readers interested in JPEG2000 can find information in [15] and [16]. The test images can be found in [17]. The conditions for the experiments were:

- For every test image one region of interest is specified.
- For every test image only one bitstream is produced. The same bitstream is decoded to several bitrates and results are provided for these bitrates.
- For the first bitrates the algorithm should keep the quality as good as possible for the whole image. After a specific bitrate called the switch rate, the background is considered less important and the rest of the bitstream can be used to improve the quality of the ROI. Therefore at the remaining bitrates the ROI quality should be optimal. The last bitrate is when the ROI is lossless. This should be achieved with as low bitrate as possible.

Table 2 Bitrate and ROI specification

IMAGE	Switch rate (in bpp)	ROI = all (x,y)* such that:	Image dimensions W x H
Bike	0.0572	200<=x<1900 and 300<=y<2000	2048 x 2560
Cafe	0.0572	800<=x<2000 and 900<=y<2500	2048 x 2560
Aerial2	0.0715	600<=x<1900 and 200<=y<1300	2048 x 2048
Woman	0.0572	$(x-1100)^2+(y-800)^2 < 700^2$	2048 x 2560
Target	0.2289	$(x-250)^2+(y-250)^2 < 160^2$	512 x 512

* x is horizontal position and y vertical position, where the uppermost left pixel is considered to be (x=0,y=0).

The bitrate and ROI definitions are found in **Table 2**.
The ROI' s where chosen to contain the most meaningful parts of the image. This selection is of course a subjective one.
Bike: The bike (Rectangle).
Cafe: The cafe (Rectangle).
Aerial2: The cluster of buildings (Rectangle).
Woman: The face (Circle).
Target: The centre (Circle).

The methods compared are:

- S+P: The proposed scheme. The image is encoded up to the background bit rate and then the mask is calculated and the necessary coefficients are encoded further. The S+P with the A predictor is used. The procedure is progressive till the end.
- S: Same as above but with the S transform only. This method is efficient for the ROI but gives blocking artifacts in the lossy part of the image.
- TT: The proposed scheme with the TT transform. The image is encoded up to the background bit rate and then the mask is calculated and the coefficients encoded further. The procedure is progressive till the end.
- Residual: The S+P with the C predictor is used. The image is encoded up to the background bit rate and the image reconstructed. The difference from the original image is then calculated with the same procedure as in [4]. The residual with the background masked away is then encoded with JPEG-LS, the emerging lossless JPEG standard ISO/IEC JTC1/SC29/WG1. The software for JPEG-LS was found at http://www.hpl.hp.com/loco/. With this method the encoding is not progressive after the switch to JPEG-LS.

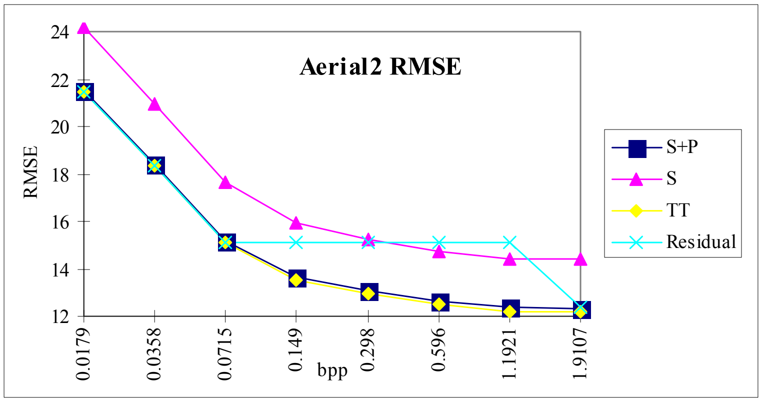

Figure 12 Aerial2 RMSE for the whole image. Switch rate 0.0715 bpp

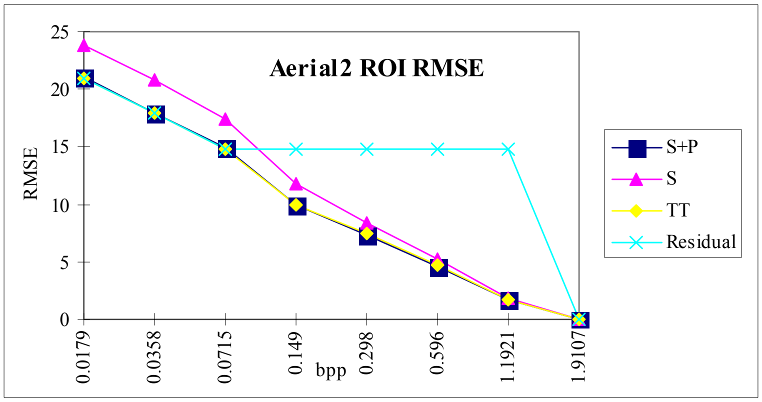

Figure 13 Aerial2 RMSE for the ROI only. Switch rate 0.0715 bpp

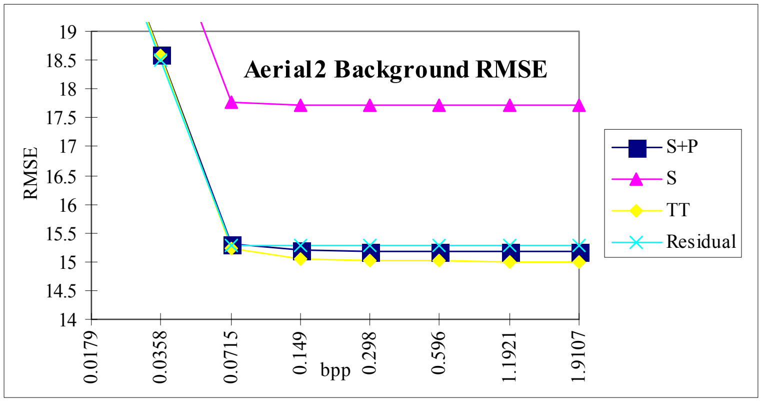

Figure 14 Aerial2 RMSE for the background only. Switch rate 0.0715 bpp

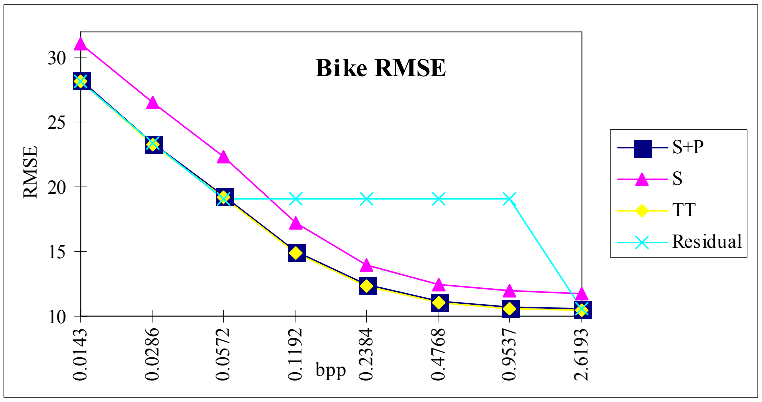

Figure 15 Bike RMSE for the whole image. Switch rate 0.0572 bpp

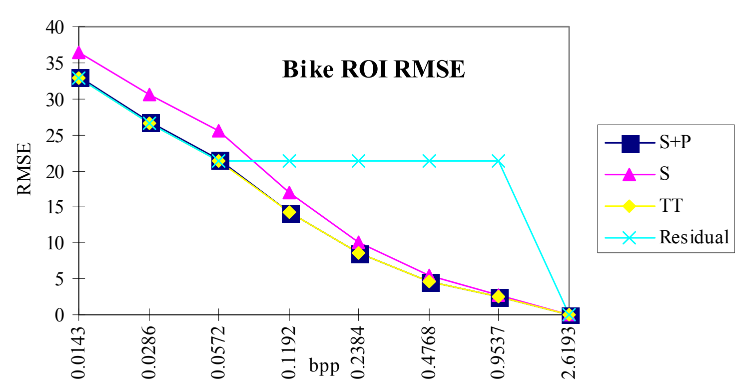

Figure 16 Bike RMSE for the ROI only. Switch rate 0.0572 bpp

Table 3 Lossless bitrate for the various images

	S+P	S	TT	Residual
Aerial2	1.91	1.96	1.92	1.90
Bike	2.62	2.69	2.62	2.63
Café	2.03	2.11	2.03	2.00
Target	1.04	0.99	1.11	1.21
Woman	1.40	1.44	1.40	1.37

In **Figure 12** and **Figure 15** rate distortion curves are given for the whole test images "Aerial2" and "Bike". The coding effort was concentrated to the ROI after the switch rate but the distortion was calculated on the whole image. In **Figure 13** and **Figure 16** results calculated separately on the ROI are given. Root mean square error (RMSE) is used as the distortion measure. The bitrate is given in bits per pixel (bpp). The S+P and the TT transform methods produce almost identical results, both better than the S transform. Notice that with the residual method it is not possible to get any intermediate images between the switch rate and the really lossless bitrate. Therefore the points on this curve only correspond to the image that was reconstructed just before the switch rate. They were included to clarify where the switch was taken.

Figure 14 gives the RMSE results on "Aerial2" for the background separately after the switch. All methods switch off the background improvement efficiently and the curves become very flat. However there is a very small improvement also after the switch with the S, S+P and TT methods. This because of the slight growth of the mask and that the algorithm is "spilling" some information outside the ROI to make the perfect reconstruction possible. One might think that spilling bits into the background is a waste of bits. This is not necessarily the case. If the encoding of the ROI also improves the quality of the background, less bits are needed to describe the background in the first place and the encoding can be terminated earlier. It is also reasonable to believe that better quality is more desirable on the parts of the image close to the ROI, than on the remote ones. In fact, the mask method produces a graceful degradation and no visually annoying edges around the ROI.

In **Table 3** the bitrate is given at which the lossless ROI was achieved for the various images and methods. The S method gives worse results for all images except "Target". This image is full of grids and patterns and therefore the short S filter does well. Notice also that the differences between the other methods are very small.

Finally it should be stressed that the real advantage of the proposed method is the flexibility and simplicity of having a lossless ROI without being forced to switch to residual coding.

4. Discussion

The method that has been described in this paper has been described as progressive in pixel fidelity. However there is no inherent problem in using it in a hierarchically progressive mode also. The choice of progression is only a choise of coded bit depth on the wavelet coefficients at different stages. The key point of the method
is to find out which coefficients that eventually should be coded to their full bit depth. This does not restrict the way to reach the full depth. However the entropy coding scheme would have to be adapted.

In a client-server situation the scheme could be used to let the client specify or change the ROI during transmission. The server is then initially transmitting the image unaware of the clients preferences. When the client gets an idea of the image the ROI is specified on a back channel to the server. Since the transform coefficients are not dependent on the ROI definition, the server can then entropy decode the stored image, decide which coefficients to skip and then entropy encode the new symbol stream.

Another way to provide this in a simple fashion is to use tiling [9]. However the tiles have to be kept quite large to avoid tiling artifacts. It might therefore be a reasonable thing to combine the methods so that the proposed method is used inside the separate tiles.

5. Conclusions

The well known S+P and TT transforms have been used to allow progressive transmission with perfect reconstruction of selected ROI's in the image. The proposed scheme calculates a mask which specifies which coefficients are needed for a certain region. This makes it possible to have progressive transmission while at the last stage of the transmission selected ROI's are reconstructed without any loss in quality. The algorithm is useful in high demanding applications, where compression is mandatory and no quality loss can be accepted in selected parts of the image, like in medical applications.

6. References

[1] J. M. Shapiro, "Embedded Image Coding using zerotrees of wavelet coefficients", *IEEE Trans. on Signal Processing*, Vol. 41, No. 12, pp. 3445-3462, Dec. 1993.

[2] A. Said and W. A. Pearlman, "Reversible image compression via multiresolution representation and predictive coding", in *Proc. SPIE Conf. Visual Communications and Image Processing* '93, Cambridge, MA, Nov. 1993, Proc. SPIE 2094, pp. 664-674.

[3] A. Said and W. A. Pearlman, "A new, fast and efficient image codec based on set partitioning in hierarchical trees", *IEEE Trans. on Circuits and Systems for Video Technology,* Vol. 6, No. 3, pp. 243-250, June 1996.

[4] A. Said and W. A. Pearlman, “An Image Multiresolution Representation for Lossless and Lossy Compression”, *IEEE Trans. on Image Processing,* Vol. 5, No. 9, pp. 1303-1310, September 1996.

[5] J. Ström, P. C. Cosman, “Medical image compression with lossless regions of interest”, *Signal Processing* 59, No. 2, Jun. (1997) 155-171

[6] V. K Heer and H-E. Reinfelder, “A comparison of reversible methods for data compression”, *Proc. SPIE*, vol. 1233 Med. Imag. IV, pp 354-365, 1990.

[7] M. Rabbani, P.W. Jones, *Digital Image Compression Techniques,* SPIE Optical Engineering Press, Vol. TT7, 1995.

[8] D. Nister and C. Christopoulos, “An embedded DCT-based still image coding algorithm”, *ISO/IEC JTC1/SC29/WG1 N610*, November 10-14, Sydney, Australia, 1997.

[9] Ricoh California Research Center, *RICOH CREW Image compression standard*, 24 October, 1997 DRAFT 11.

[10] K. Nguyen-Phi and H. Weinrichter, “DWT image compression using Contextual bitplane coding of Wavelet coefficients”, *Proc. ICASSP 97*, pp. 2969-2971, 1997.

[11] J. M. Shapiro, “Apparatus and Method for Emphasizing a Selected Region in the Compressed Representation of an Image”, *United States Patent* 5.563.960, Oct. 8,1996.

[12] I. H. Witten, R. M. Neal, and J. G. Cleary, “Arithmetic coding for data compression”, *Communications of the ACM*, Vol. 30, No. 6, pp. 520-540, June 1987.

[13] A. R. Calderbank, I. Daubechies, W. Sweldens and B. Yeo, “Lossless Image Compression using Integer to Integer Wavelet Transforms”, *Proc. ICIP 97*, pp 596-599, 1997.

[14] C. Brislawn, “Classification of Nonexpansive Symmetric Extension Transforms for Multirate Filter Banks”, *Applied and Computational Harmonic Analysis*, vol 3, pp. 337-357, 1996.

[15] ISO/IEC JTC1/SC29/WG1 N505, “Call for contributions for JPEG 2000 (JTC 1.29.14, 15444): Image Coding System”, March 21, 1997.

[16] ISO/IEC JTC1/SC29/WG1 N390R, “New work item: JPEG 2000 image coding system”, March 21, 1997.

[17] ISO/IEC JTC1/SC29/WG1, “JPEG 2000 Algorithm Test and Submission Package”, ISO/IEC JTC1/SC29/WG1 N461, 15 November 1996.