

CERIAS Tech Report 2001-112
An Evaluation of Color Embedded Wavelet Image Compression Techniques
by M Saenz, P Salama, K Shen, E Delp
Center for Education and Research
Information Assurance and Security
Purdue University, West Lafayette, IN 47907-2086

An Evaluation of Color Embedded Wavelet Image Compression Techniques

M. Sáenz, P. Salama, K. Shen and E. J. Delp*
Video and Image Processing Laboratory (VIPER)
School of Electrical and Computer Engineering
Purdue University
West Lafayette, Indiana
USA

Keywords: Still Image Compression, Color Embedded Wavelet Image Compression, Spatial Orientation Trees.

ABSTRACT

Color embedded image compression is investigated by means of a set of core experiments that seek to evaluate the advantages of various color transformations, spatial orientation trees and the use of monochrome embedded coding schemes such as EZW and SPIHT. In order to take advantage of the interdependencies of the color components for a given color space, two new spatial orientation trees that relate frequency bands and color components are investigated.

1. INTRODUCTION

Date rate scalability, the capability of decoding compressed images at different data rates, is an extremely important issue in image compression. Many applications (such as document imaging, facsimile, Internet imaging, laser print rendering, digital photography, digital libraries, image databases, medical imaging and remote sensing) require that a compression technique supports different data rates for decompression.

In order to achieve scalable compression, compressed data is embedded in a single bit stream in such a way that the important information is located at the beginning of the stream. This permits the decoder to truncate the bit stream once a desired data rate or distortion has been achieved. Transform coding is commonly utilized in embedded coding, with the wavelet transform [1,2,3,4,5,6,7] being the often used transform, although other alternative transforms such as the Laplacian pyramid [8] and the discrete cosine transform [9] have been used. The greatest distortion reduction is achieved if the transform coefficients with the largest magnitudes are coded initially and with high precision. This also requires that the “locations” of the larger coefficients be encoded. At the decoder, the most significant information is decoded first and the image is refined until a satisfactory distortion or desired data rate is attained.

When using the wavelet transform it is possible to exploit the unique properties of the wavelet coefficients to efficiently encode them. The use of spatial orientation trees (SOTs), such as zerotrees, introduced first by Shapiro [2,10], resulted in a very efficient method for achieving rate scalability. A variation of this algorithm, referred to as Set Partitioning in Hierarchical Trees (SPIHT), was proposed by Said and Pearlman [1]. Zerotrees take advantage of structural similarities across bands when using wavelet decomposition of an image. A shortcoming of Shapiro's [2,10] algorithm is that it addresses the compression of monochrome images only.

To encode a color image, one can simply ignore the interdependence between the color components and encode each component as an individual grayscale image. Alternatively, one could take advantage of the interdependencies of the color components for a given color space. Various configurations for embedded color image compression that use several types of spatial orientation trees and various color transformations are described in this paper. In particular, we will describe a modification of the *Color Embedded Zerotree Wavelet (CEZW)* approach described in [11]. We will also discuss modifications to SPIHT spatial orientation tree [1].

* This work was partially supported by grants from the Rockwell Foundation, the AT&T Foundation and Lucent Technologies. Address all correspondence to E.J. Delp, ace@ecn.purdue.edu, <http://www.ece.purdue.edu/~ace>, +1 765 494 1740.

This paper is organized as follows: Section 2 describes two color transformations commonly used in color image coding. An overview of wavelet transform coding is discussed in Section 3, and the use of spatial orientation trees to achieve data rate scalability in Section 4. The basic principles of the embedded zerotree wavelet (EZW) [2] technique, SPIHT and our new approaches to embedded rate scalable color image compression are also discussed in Section 4. In Section 5, the experimental results of this study are presented.

2. COLOR SPACES

Tristimulus color spaces have been successfully used for the representation of color [12]. The color space most often used in still image compression is the red (R), green (G) and blue (B) or RGB space. Other color spaces such as luminance and chrominance spaces have been used in digital video systems. The most popular of these are the YUV and $YCbCr$ spaces [13,14]. Generally, the components of many of these spaces are highly correlated. This correlation has not been fully exploited in color image compression. The use of an appropriate color transformation can provide for significant increase in compression performance.

The transformation from RGB to YUV color space is expressed as:

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.500 & -0.419 & -0.081 \\ -0.169 & -0.331 & 0.500 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}. \quad (1)$$

One of the advantages of this transformation is that it reduces the psychovisual redundancy in an image [4]. It has been shown that the human visual system is relatively insensitive to the high frequency content of the chrominance components [14]. Thus, these components are commonly subsampled to remove redundancy as in the JPEG and MPEG standards. A problem with the transform described above is that it is not reversible[†] [3, 4]. A reversible transformation is desired for lossless compression. A reversible RGB to YUV transformation is given by:

$$\begin{aligned} Y &= \left\lfloor \frac{R + 2G + B}{4} \right\rfloor & G &= Y - \left\lfloor \frac{U + V}{4} \right\rfloor \\ U &= R - G & R &= U + G \\ V &= B - G & B &= V + G \end{aligned} \quad (2)$$

To fully exploit color dependencies, an image dependent color transformation can be used. The Karhunen-Loeve Transform (KLT) has been used in decorrelating color components [15, 16, 17]. It should be noted that in this application of the KLT we are using the transformation to decorrelate the spectral information in a color image and not the spatial information. Let the row vectors \mathbf{r} , \mathbf{g} , and \mathbf{b} , denote the lexicographic ordering of the R, G, and B color components of the image, respectively. Then the sample covariance matrix is given by:

$$C = \begin{bmatrix} \mathbf{r}\mathbf{r}^T - m_r^2 & \mathbf{r}\mathbf{g}^T - m_r m_g & \mathbf{r}\mathbf{b}^T - m_r m_b \\ \mathbf{g}\mathbf{r}^T - m_g m_r & \mathbf{g}\mathbf{g}^T - m_g^2 & \mathbf{g}\mathbf{b}^T - m_g m_b \\ \mathbf{b}\mathbf{r}^T - m_b m_r & \mathbf{b}\mathbf{g}^T - m_b m_g & \mathbf{b}\mathbf{b}^T - m_b^2 \end{bmatrix} \quad (3)$$

where the m_j 's denote the sample mean of component j , where $j \in \{R, G, B\}$. The color transformation matrix used corresponds to the eigenvector matrix of C .

Other color space representations that have been proposed for color imagery exploit specific properties that are useful in areas such as image printing and human system modeling [12]. In this paper we will represent color images using RGB, YUV and KLT spaces.

[†] The transformation described in Equation 1 is the one used in the ITU-R 601-1 digital video standard [18].

3. THE WAVELET TRANSFORM

The wavelet transform has been widely used in image and video compression since it allows localization in both the space and frequency domains [1,2,4,11,7,19]. Typically an image is decomposed into a hierarchy of frequency subbands that are processed in an independent manner. The decomposition is achieved by filtering along one spatial dimension at a time to effectively obtain four frequency bands as shown in Figure 1. The lowest subband, commonly referred to as Low-Low (LL), represents the information at all coarser scales and it is decomposed and subsampled to form another set of four subbands. This process can be continued until the number of levels of decomposition, m , is attained. The analysis filters h and g efficiently decompose the image into independent frequency spectra of different bandwidths or resolutions [5,20], producing different levels of detail.

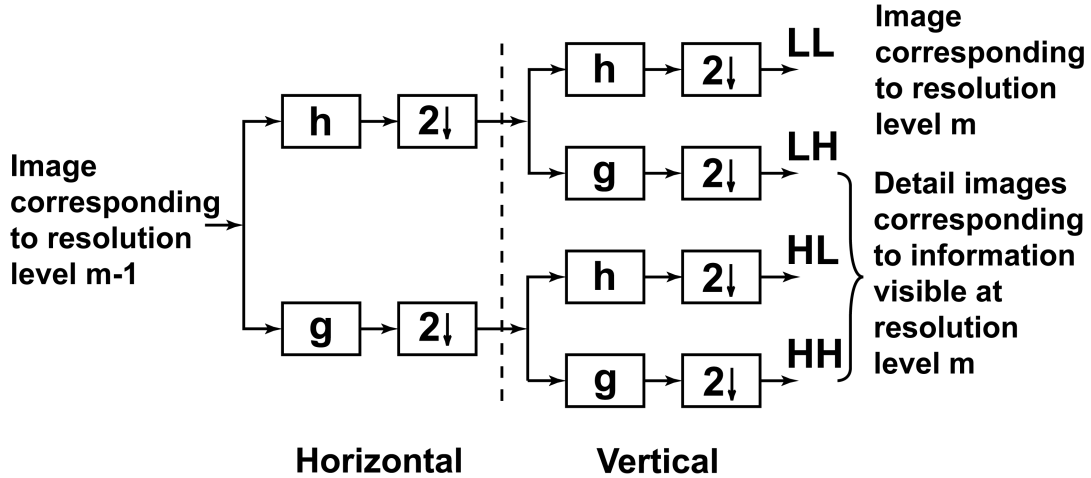


Figure 1. Block Diagram of one level of the wavelet transform decomposition. ‘ h ’ and ‘ g ’ are commonly referred to as the analysis filters. L and H stand for low and high frequency bands respectively.

The various subband signals are recombined so that the original signal is reconstructed. The reconstruction is accomplished by upsampling the lower resolution images and passing them through synthesis filters, as shown in Figure 2.

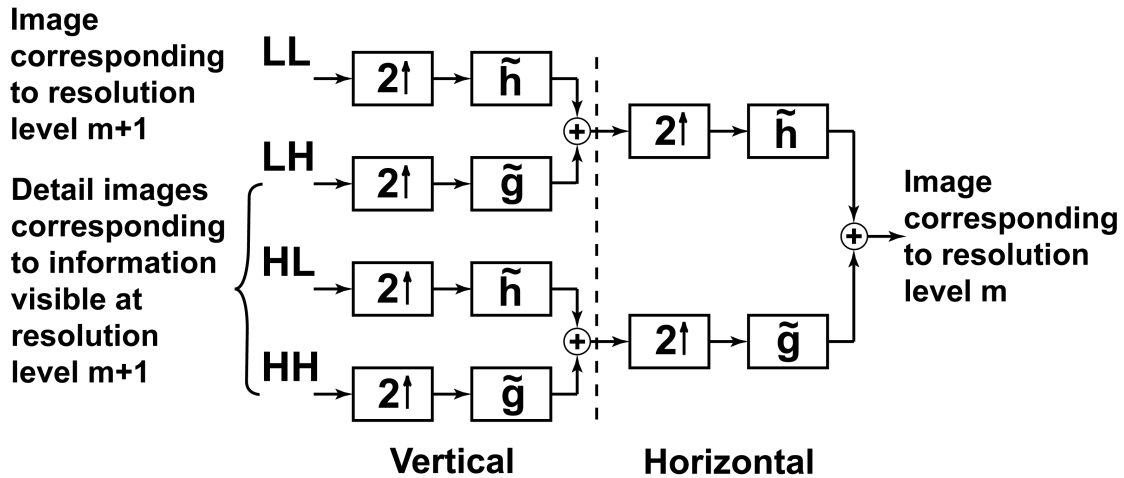


Figure 2. One level of the wavelet transform reconstruction. In this case the filters used are referred to as the synthesis filters.

Various types of analysis filters have been proposed in image and video compression. Recently there has been interest in integer to integer wavelet transforms that can be used in lossless image compression [4, 21].

To use the wavelet transform for image compression one must quantize and binary encode the wavelet coefficients. Typically wavelet coefficients with large magnitude (or high energy) are assigned more bits and hence have a higher precision. Some of the coefficients are given zero bits and therefore not included in the compressed representation of the image. As in all transform coding techniques the location of the quantized coefficients must also be known by the decoder. This is accomplished using various techniques such as the zigzag scanning used in the JPEG and MPEG standards. In the next section we shall describe other approaches to organize the wavelet coefficients into quadrees that will allow a very compact representation for image compression and rate-scalability.

4. SPATIAL ORIENTATION TREES

Shapiro observed that using quadtree representations of the wavelet coefficients provided a method for grouping the coefficients that belong to different subbands but have the same spatial location into one structure [2]. This structure, known as a spatial orientation tree (SOT), can then be represented by one symbol. SOTs have been widely used in rate scalable coding of still images and video [11].

The underlying idea behind grouping wavelet coefficients into SOTs is based on the observation that most of the energy is concentrated at the low frequency subbands in a wavelet decomposition. In addition, if the magnitudes of the wavelet coefficients in the lowest subband of a decomposition are insignificant relative to a particular threshold, then it is more likely that wavelet coefficients having the same spatial location, but residing in different subbands, will also be insignificant. Furthermore, when proceeding from the highest to the lowest levels of a wavelet pyramid the variations in the wavelet coefficients decrease. This allows for the coding of a large number of insignificant wavelet coefficients by only coding the location of the root coefficient to which the entire set of coefficients is related. Such a set is commonly referred to as a zerotree [2].

Rate scalability is achieved by organizing the coefficients in order of importance and progressively quantizing and binary encoding the wavelet coefficients. In this manner, by decoding the initial portion of the bit stream a coarse version of an encoded image is obtained at the receiving end. The image is then refined by decoding additional data from the compressed data stream. This process is terminated when a desired data rate or distortion is attained. It has been shown that if the mean squared error (MSE) is used as the distortion metric, the wavelet coefficients with the largest magnitude need to be encoded with high precision [22], and in this framework of rate scalability, they need to be encoded first.

Various schemes have been proposed to improve the performance of the traditional zerotree method [23,24]. In [24], a context modeling technique that takes advantage of zerotree structures was introduced. The resulting scheme was not rate scalable. Quantizing the coefficients using vector quantizers has also been explored. For example in JPEG-2000, a Trellis Coded Quantization (TCQ) [21] technique, which can be viewed as a time-varying scalar quantization or a computationally efficient vector quantization scheme has been proposed [6]. Other approaches include the multi-threshold wavelet coder described in [25].

Embedded coding of color images can be performed in a number of different ways. One way would be to encode each color component separately. This requires that a rate allocation scheme for assigning bits for the various components be designed. The disadvantage of such a technique is that the resulting bit stream is not fully embedded. This could be addressed by interleaving the data belonging to the three color components into one bit stream. An alternative way, which produces a fully embedded stream, would be to encode all color components at once.

The compression methods proposed by Shapiro [2], and Said and Pearlman [1] besides being developed for grayscale images, use slightly different SOTs as shown in Figure 3. The major difference between the two techniques lies in the fact that they used different strategies to scan the transformed coefficients and perform the encoding. This will be discussed below.

4.1 EZW

In the embedded zerotree wavelet (EZW) scheme, developed by Shapiro [2], the wavelet coefficients are grouped into Spatial Orientation Trees, as shown in Figure 3a. The magnitude of each wavelet coefficient in a tree, starting with the root of the tree located in the LL band of the decomposition, is then compared to a threshold T . If the magnitudes of all the wavelet coefficients in the tree are smaller than T , the entire tree structure (that is the root and all its descendant nodes) is represented by one symbol, the zerotree symbol ZTR [2]. If however, there exist significant (greater than T) wavelet coefficients in the

tree, then the tree root is represented as being significant (represented by the symbol SIG, when its magnitude is greater than T), or insignificant (the symbol IZ, isolated zero, when its magnitude is smaller than T). The descendant nodes are then each examined in turn to determine whether each is the root of a possible subzerotree structure, or not. This process is carried out such that all the nodes in all the trees are examined for possible subzerotree structures. The significant wavelet coefficients in a tree are represented by one of two symbols, POS or NEG, depending on whether their values are positive or negative, respectively. The process of classifying the coefficients as being ZTR, IZ, POS, or NEG is referred to as the dominant pass [2]. This is then followed by a subordinate pass in which the significant wavelet coefficients in the image are refined by determining whether their magnitudes lie within the intervals $[T, 3T/2)$ and $[3T/2, 2T)$. Those wavelet coefficients whose magnitudes lie in the interval $[T, 3T/2)$ are represented by the symbol 0 (LOW), whereas those with magnitudes lying in the interval $[3T/2, 2T)$ are represented by the symbol 1 (HIGH). Subsequent to the completion of both the dominant and subordinate passes, the threshold value T is reduced by a factor of 2, and the process repeated. This coding strategy, consisting of the dominant and subordinate passes followed by the reduction in the threshold value, is iterated until a target date bit rate is achieved. This threshold reduction essentially acts as a uniform quantizer of the coefficients. The sequence in which the coefficients are examined is predefined and known by the decoder. This is sometime referred to as a zerotree scanning order. The compressed bit stream therefore consists of the initial threshold and the tree symbols. This information is binary encoded using an entropy encoder. The tree symbols along with the scanning order describe where the wavelet coefficients are located in the decomposition and their respective quantization.

An EZW decoder reconstructs the image by progressively updating the values of each wavelet coefficient in a tree as it receives the data. The decoder's decisions are always synchronized to those of the encoder, making this algorithm highly sensitive to transmission errors [26].

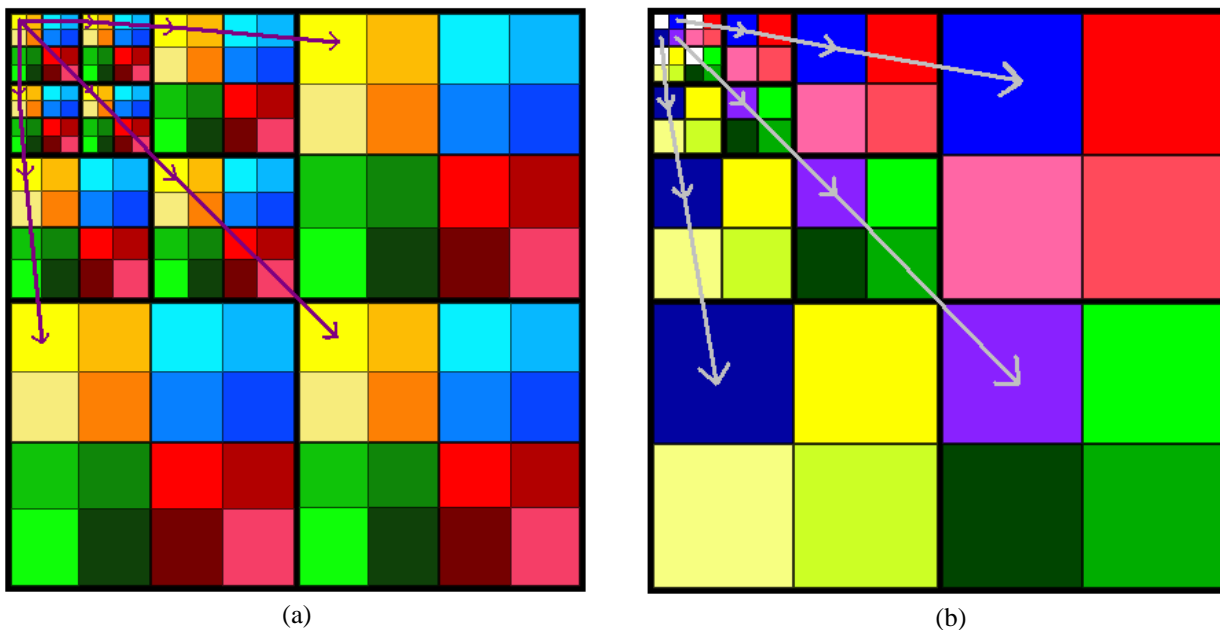


Figure 3. Diagrams of the parent (root)-descendent relations in spatial orientation trees. (a) EZW SOT: Notice that any coefficient in the LL band have 3 immediate descendents. Other coefficients, except for those in the highest frequency bands, have 4 descendents. Assume that LL is N by M in which case the number of parents in this tree would be $N \cdot M$. (b) SPIHT SOT: The LL band is divided into groups of 2 by 2 of which one coefficient has no descendents; all other coefficients, except for those in the highest frequency bands, have 4 immediate descendents. Assume that LL is N by M in which case the number of parents in this tree would be $\frac{3}{4} N \cdot M$.

4.2 SPIHT

SPIHT is similar to EZW, in that it performs a partial ordering of the coefficients using a set of octavely decreasing thresholds [1]. In this case, the initial threshold corresponds to the largest power of two that is smaller than the magnitude of

the largest coefficient. Furthermore, SPIHT uses a different SOT, which is based on the assumption that the dimension of the Low-Low band generated by the wavelet decomposition is an integer multiple of two. This allows for a two by two grouping of adjacent coefficients in the Low-Low band. In every group, one coefficient is chosen to have no descendants in the other subbands, whereas the remaining have descendants, as shown in Figure 3b.

SPIHT groups the wavelet coefficients into three lists, the list of insignificant sets (LIS), the list of insignificant pixels (LIP), and the list of significant pixels (LSP). These are initialized to the set of subtree descendants of the nodes in the highest level, the nodes in the highest level, and an empty list, respectively. A two phase process that consists of a sorting pass and a refinement pass is subsequently initiated. During the sorting pass, the algorithm traverses through the LIP testing the magnitude of its elements against the current threshold and representing their significance by 0 or 1. Whenever a coefficient is found to be significant its sign is coded and it is moved to the LSP. The algorithm then examines the list of insignificant sets and performs a magnitude check on all the coefficients in a set. If a particular set is found to be significant it is then partitioned into subsets and tested for significance, otherwise a single bit is appended to the bit stream to indicate an insignificant set. After a pass through the LIS is completed, a refinement pass through the LSP, excluding those coefficients added during the previous sorting pass, is initiated. The refinement pass is accomplished by using progressive bit plane coding of the ordered coefficients [1]. The threshold is subsequently decreased by a factor of two (resulting in uniform quantization of the coefficients) and the entire process repeated until the number of allocated bits is exhausted. Since all branching decisions made by the encoder as it searches throughout the coefficients are appended to the bit stream, the locations of the coefficients being refined or classified is never explicitly transmitted. The output of the sorting-refinement process is then entropy coded via an arithmetic encoder. The decoder recreates the decisions of the encoder to reconstruct the image.

SPIHT has been used to compress color images by first transforming the color components using the KL transform and then using the SPIHT SOT to independently compress each transformed component. A rate allocation scheme that does not pre-allocate the number of bits per color component is used [27]. Throughout the study described in Section 5, we shall refer to this color compression technique as CSPIHT.

Below, we will subsequently describe how the concepts of a spatial orientation tree and successive refinement can be used for color image compression.

4.3 CZW and SP-CZW

The spatial orientation trees described above were designed for compressing monochrome images. They do not exploit the interdependence between color components. A new SOT that exploits the correlation between the color components of the YUV color space was presented in [11]. This is known as the *Color Embedded Zerotree Wavelet (CEZW)* compression technique [28].

The underlying idea in CEZW was to expand the SOT not only across the different frequency subbands, but also across the color components. Experimental evidence has shown that at the spatial locations where chrominance components have large transitions, the luminance component has also large transitions. Thus, if a transform coefficient in a particular frequency band of the luminance component has small magnitude, the transform coefficient of the chrominance components at the corresponding spatial location and frequency band will most likely also have small magnitude [28]. This interdependence of the transform coefficients of the different color components is therefore exploited in CEZW. In the study described in the next section we will use the SOT from CEZW with various types of color transformations, and shall refer to the CEZW SOT as CZW. Below we will describe CZW in more detail and indicate how the SOT used in SPIHT can also be similarly extended to exploit correlations between color components.

In order to use a minimum number of bits to specify a large area of insignificant coefficients, that is to define zerotree nodes for color images, it is necessary that the basic EZW tree structure be enhanced. In this new approach a wavelet decomposition is performed on each color component. In the description of the SOT below, a YUV transformation is assumed. The results are applicable to any color transformation without loss of generality. The SOT is expanded by setting each chrominance node to be a descendant node of the luminance node at the same spatial location, as shown in Figure 4a. Each chrominance node has two parent nodes, one being a chrominance node having the same spatial location as the node being examined, and the other being a luminance node.

The coding strategy consists of dominant and subordinate passes. The symbols used in the dominant pass are the same as those used in EZW namely positive significant (POS), negative significant (NEG), isolated zero (IZ) and zerotree (ZT, if all the descendants coefficient are insignificant). During the dominant pass each coefficient in the Low-Low band of the luminance component is examined for potential zerotrees. A zerotree symbol is assigned when the current coefficient and all its descendents (in the luminance and chrominance components) are insignificant. The two chrominance components are alternately scanned after the luminance component has been scanned. Those coefficients in the wavelet decomposition of the chrominance components that have already been encoded as part of a zerotree while scanning the luminance component are not examined. The subordinate pass, which refines the coefficients that have been coded as significant in the previous dominant pass, is the same as that in EZW.

CZW can then be summarized as follows:

- A wavelet transform is performed on each of the three color components separately. Let T be the largest magnitude in wavelet transform coefficients.
- While bit budget is not exhausted:

$$\text{Set } T = \frac{1}{2}T$$

Dominant pass:

- The Y component is scanned first. All the descendents of each node including those in the U and V components are compared to T . The symbols POS, NEG, ZT and IZ are then assigned and entropy coded.
- The U and V components are alternately scanned. The coefficients and their descendent nodes are compared to T . Those coefficients that have been coded as a part of a zerotree in the previous step are not examined. The symbols POS, NEG, ZT and IZ are also assigned and entropy coded.

Subordinate pass:

- The coefficients that have been coded as significant in previous passes are examined for refinement as in EZW. The symbols HIGH and LOW are then assigned and entropy coded.

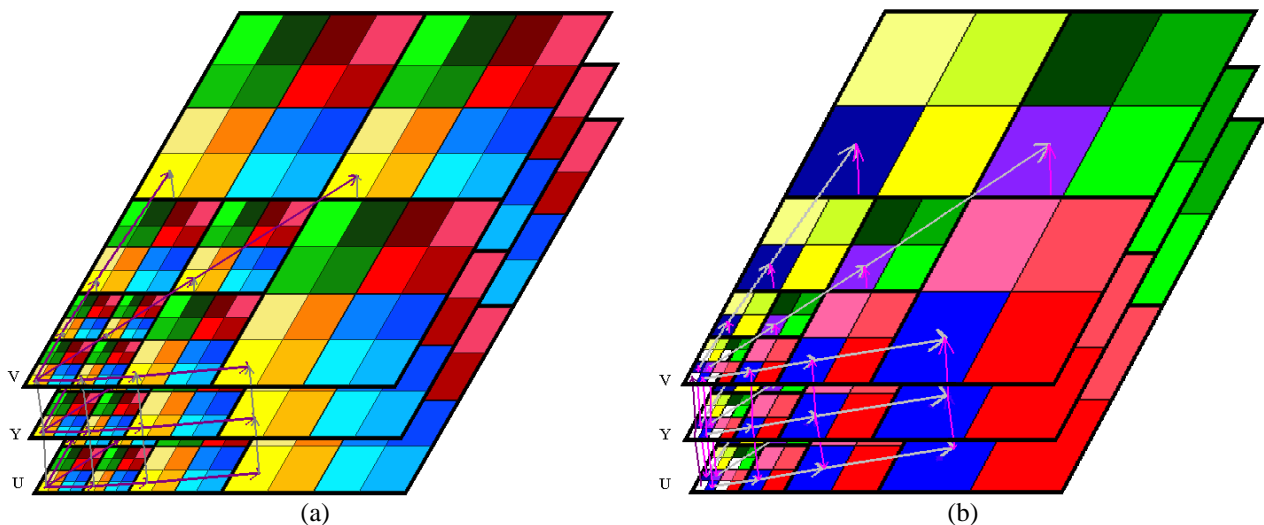


Figure 4. Diagram of the root-descendent relations in CZW and SP-CZW for a YUV image. (a) This tree is developed on the basis of the tree structure in the EZW algorithm. It was designed for a luminance-chrominance color space but could be tested for any color transformation. (b) This tree is developed on the basis of the SOT structure in the SPIHT algorithm

The SOT used in SPIHT can also be modified in a similar way to exploit correlations in the color components, as shown in Figure 4b. This can be implemented with either EZW's or SPIHT's methods for decision making and coding of the coefficients' significance. In this paper, we shall refer to this SOT used in conjunction with Shapiro's dominant and subordinate passes through the tree as SP-CZW.

5. EXPERIMENTAL RESULTS

A combination of color transformations, spatial orientation trees and embedded encoding algorithms were evaluated by utilizing them to compress a set of ten 512x512, 24 bit RGB images. The color transformations that were used were RGB to YUV and RGB to KLT. The YUV transform in Equation 1 was used in this study. The various SOTs that were examined included both the Shapiro, and Said and Pearlman SOTs defined for monochrome images, as well as their extensions to exploit interdependence between colors as described in the previous section. We shall refer to coding of color images by compressing each color component independently using EZW or SPIHT[‡] as Naive-EZW (NEZW) and Naive-SPIHT (NSPIHT), respectively. The other techniques used in our experiments were CZW, SP-CZW and CSPIHT which were described in Section 4.

A six level wavelet decomposition, based on the 9/7 biorthogonal wavelet filter described in [7] was used for all the test images. This resulted in the lowest resolution subband being an 8 by 8 image. All images were encoded at a maximum data rate of 2 bpp (bits per pixel) and decoded at six different data rates 2, 1.5, 1, 0.5, 0.25 and 0.1 bpp. Adaptive arithmetic coding was used to entropy encode the tree symbols. The size of the image, the number of levels of the wavelet decomposition, the initial threshold T , and the maximum data rate were inserted at the beginning of the bit stream as header information and included in the data rate calculation.

In the following, we will discuss the performance of all schemes, at a data rate of 0.25 bpp, for the *Barbara* image shown in Figure 5. We have included, in Figure 6, a JPEG compressed image at the same data rate. We also compared our results with CSPIHT as shown in Figure 7. The peak signal-to-noise ratio (PSNR), based on mean square error, is used as a measure of “quality”. The PSNR of a color image with color components A, B, and C is given by:

$$\text{PSNR} = 10 \cdot \log_{10} \left\{ \frac{255^2}{\frac{\text{MSE}(A) + \text{MSE}(B) + \text{MSE}(C)}{3}} \right\}, \quad (4)$$

the images were also visually compared, in all of our experiments we obtained the PSNR using the RGB color space.

When using NEZW and NSPIHT an issue that needs to be addressed is the manner in which the total number of bits allocated to the image is to be distributed among the three components. That is, are all the components allocated an equal number of bits, or should the number of bits allotted to a component be commensurate with its subjective importance. We observed that using unequal bit allocation enhances the quality of the reconstructed image more than when the bit budget is equally distributed among the three components. In particular, we observed that the decompressed image was visually more pleasing when more bits were allotted to the Y and G components of YUV and RGB images, respectively. The same was observed when more bits were assigned to one of the components of a KLT image. In our experiments the G and Y components were assigned twice as many bits as the other components for NEZW and NSPIHT. Figures 8-10 and Figures 11-13 show the performance of NEZW and NSPIHT for the three color transformations at a rate of 0.25 bpp, respectively. It was observed that the best subjective performance was attained with YUV images, although results for KLT images were comparable. The worst performance was attained with RGB images. These conclusions were also substantiated by the PSNR provided in Table 1. In all cases it was observed that NSPIHT outperformed NEZW.

Figures 14-16 and 17-19 show the performance of CZW and SP-CZW at a data rate of 0.25 bpp. It was observed that there was no noticeable difference in performance when using either Shapiro’s or Said and Pearlman’s SOT. This was substantiated by the PSNR given in Table 1. It should be noted that it was not necessary to use a predetermined bit allocation between the various color components for either technique since the allocation of bits is inherent to the refinement process. When using the extended SOTs, the G and Y components were selected to be the main component at which all roots. For KLT images the component which was allocated more bits when using the naive techniques was selected to be the main component from which the extended SOT originated.

In general, it was observed that the quality of the decompressed YUV and KLT images was better than that of RGB images for all techniques. In all cases, the KLT images were comparable in quality to the YUV images. Image edges reproduced by using the YUV transform were less noisy and sharper than those reconstructed when using the KLT transform.

[‡] The NSPIHT and CSPIHT demonstration program were obtained from Said and Pearlman’s web page at <http://ipl.rpi.edu:80/SPIHT>



Figure 5. Original Image



Figure 6. JPEG (0.25 bpp)



Figure 7. CSPIHT (0.25 bpp)



Figure 8. NEZW (RGB, 0.25bpp)



Figure 9. NEZW (KLT, 0.25bpp)



Figure 10. NEZW (YUV, 0.25bpp)



Figure 11. NSPIHT (RGB, 0.25bpp)



Figure 12. NSPIHT (KLT, 0.25bpp)



Figure 13. NSPIHT (YUV, 0.25bpp)



Figure 14. CZW (RGB, 0.25bpp)



Figure 15. CZW (KLT, 0.25bpp)



Figure 16. CZW (YUV, 0.25bpp)



Figure 17. SP-CZW (RGB, 0.25bpp)



Figure 18. SP-CZW (KLT, 0.25bpp)



Figure 19. SP-CZW (YUV, 0.25bpp)

6. CONCLUSIONS

In this paper we have investigated the use of color transformations, various spatial orientation trees and coding strategies for scalable color image compression. It was observed that the use of a SOT that exploited color dependencies performed better, both visually and with respect to PSNR, than more naive approaches. In all cases the wavelet transform techniques performed better than JPEG. At low data rates JPEG exhibited block artifacts and the wavelet techniques exhibited blur artifacts. We feel that the CZW and the modified SPIHT SOT performed similarly. It should be noted that the best performance was obtained using CSPIHT. This can be attributed to its set partitioning sorting-refinement process more than its spatial orientation tree. It was also observed that the quality of decompressed YUV and KLT images was better than that of the RGB images for all techniques. Finally, we observed that the YUV images performed slightly better visually than the KLT images despite the fact that KLT did better in PSNR. One would have to trade off the performance gain when using the more complex KLT compared with the far simpler YUV transformation.

Future work involves the implementation of the extended SOTs in conjunction with SPIHT's methodology for the refinement process. We also intend to investigate other types of color transformations that are more robust with respect to human visual system models and reversible. A copy of this paper and images used in our experiments can be obtained via anonymous ftp to skynet.ecn.purdue.edu in the directory `/pub/dist/delp/vcip99-color`.

Rate (bpp)	IMAGE	BARBARA (PSNR, dB)					
		NEZW	NSPIHT	CZW	SP-CZW	JPEG	CSPIHT
0.1	RGB	21.24	21.92	21.75	21.76	19.81	-
	KLT	21.58	22.03	22.21	22.18	-	22.99
	YUV	21.84	22.54	22.10	22.10	-	-
0.25	RGB	22.44 (Figure 8)	23.16 (Figure 11)	22.80 (Figure 14)	22.81 (Figure 17)	22.39 (Figure 6)	-
	KLT	22.54 (Figure 9)	23.20 (Figure 12)	24.09 (Figure 15)	24.10 (Figure 18)	-	25.35 (Figure 7)
	YUV	23.09 (Figure 10)	24.09 (Figure 13)	24.02 (Figure 16)	24.03 (Figure 19)	-	-
0.5	RGB	23.63	24.59	24.12	24.12	24.51	-
	KLT	24.05	24.91	26.58	26.58	-	27.54
	YUV	24.68	26.01	26.46	26.47	-	-
1.0	RGB	25.37	26.66	26.92	26.92	26.43	-
	KLT	25.83	27.24	28.68	28.69	-	29.35
	YUV	27.18	28.23	28.50	28.50	-	-
1.5	RGB	27.05	27.99	27.32	27.42	27.23	-
	KLT	27.50	28.68	29.28	29.33	-	30.25
	YUV	28.59	29.49	29.10	29.10	-	-
2.0	RGB	27.95	28.95	28.00	28.02	27.55	-
	KLT	28.39	29.87	30.00	30.01	-	31.07
	YUV	29.36	30.44	29.67	29.68	-	-

Table 1. PSNR results for the *BARBARA* image

REFERENCES

- [1] A. Said and W. Pearlman, "A new fast and efficient image codec based on set partitioning in hierarchical trees," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 6, no. 3, pp. 243-50, June 1996.
- [2] J. M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3445-3462, December 1993.
- [3] M. J. Gormish, E. L. Schwartz, A. Keith, M. Boliek and A. Zandi, "Lossless and nearly lossless compression for high quality images," *Proceedings of the SPIE/IS&T Conference on Very High Resolution and Imaging II*, vol. 3025, February 1997, San Jose, CA, pp. 62-70.
- [4] M. Boliek, M. J. Gormish, E. L. Schwartz and A. Keith, "Next generation image compression and manipulation using CREW," *Proceedings of the IEEE International Conference on Image Processing*, October 26-29, 1997, Santa Barbara, CA, pp. III-567-III-357.
- [5] S. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *IEEE Transactions on Pattern Analysis and Machine intelligence*, vol. 11, no. 7, pp. 674-693, July 1989.
- [6] J. H. Kasner, M. W. Marcellin and B. R. Hunt, "Universal trellis coded quantization," submitted to the *IEEE Transactions on Image Processing*, 1998.
- [7] M. Antonini, M. Barlaud, P. Mathie, and I. Daubechies, "Image coding using wavelet transformation," *IEEE Transactions on Image Processing*, vol. 1, no. 2, pp. 205-220, April 1992.
- [8] P. J. Burt and E. H. Adelson, "The Laplacian pyramid as a compact image code," *IEEE Transactions on Communications*, vol. 31, no. 4, pp. 532-540, April 1983.
- [9] Y. Huang, H. M. Dreizen, and H. P. Galatsanos, "Prioritized DCT for compression and progressive transmission of images," *IEEE Transactions on Image Processing*, vol. 1, no. 4, pp. 477-487, October 1992.
- [10] J. M. Shapiro, "An embedded wavelet hierarchical image coder," *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, March 23-26, 1992, San Francisco, CA, pp. V-657-V-660.

- [11] K. Shen and E. J. Delp, "Color image compression using an embedded rate scalable approach," *Proceedings of IEEE International Conference on Image Processing*, October 26-29, 1997, Santa Barbara, California, pp. III-34-III-37.
- [12] R. M. Boynton, *Human Color Vision*, Optical Society of America, 1992.
- [13] K. Jack. *Video Demystified*, Second Edition, HighText, 1996.
- [14] C. A. Poynton, *A Technical Introduction to Digital Video*. John Wiley & Sons, 1996.
- [15] S. V. Assche, W. Phillips and I. Lemahieu, "Lossless compression of pre-press images using linear color decorrelation," *Proceeding of the IEEE Data Compression Conference*, March 30- April 1, 1998, Snowbird, Utah, pp. 578.
- [16] P. Neve, W, Phillips, K. and I. Lemahieu, "Introducing a decorrelated color space in the lossy compression of pre-press applications," *Proceedings of the IS&T Fifth Color Imaging Conference: Color Science, Systems, and Applications*, Scottsdale, AZ, 1997, pp.88-91.
- [17] J. A. Saghri, T. J. Andrew and J. T. Reagan, "Practical transform coding of multispectral imagery," *IEEE Signal Processing Magazine*, vol. 12, no.1, pp. 32-43, January 1995.
- [18] CCIR Recommendation 601-3: *Encoding Parameters of digital television for studios*, 1992.
- [19] K. Shen and E. J. Delp, "Wavelet based rate scalable video compression," *IEEE Transactions Circuits and Systems for Video Technology*, vol. 8, No. 7, Nov 1998.
- [20] S. Mallat, *A Wavelet Tour of Signal Processing*, Academic Press, New York, 1998.
- [21] P. J. Sementilli, A. Bilgin, J. Kasner and M. Marcellin, "Wavelet TCQ: submission to JPEG-2000," *Proceedings of the SPIE Conference on Applications in Digital Image Processing XXI*, vol. 3460, pp. 2-12, July 1998.
- [22] R. DeVore, B. Jawerth, and B. Lucier, "Image compression through wavelet transform coding," *IEEE Transactions on Information Theory*, vol. 38 no. 2, pp. 719-746, 1992.
- [23] C. Barreto and G. Mendoncca, "Enhanced zerotree wavelet transform image coding exploiting similarities inside subbands," *Proceedings of the IEEE International Conference on Image Processing*, September 16-19, 1996, Lausanne, Switzerland, pp. II-549-II-552.
- [24] J. Chen, T. Yu and X. Wu, "Non-embedded Wavelet Image Coding Scheme," *International Conference on Image Processing*, October 26-29, 1997, Santa Barbara CA, pp. I-584-I-587.
- [25] H. Wang and C. Kuo, "A multi-threshold wavelet coder (MTWC) for high fidelity image," *Proceedings of the IEEE International Conference on Image Processing*, October 26-29, 1997, Santa Barbara, CA, pp. I-652-I-655.
- [26] P. Salama, N. Shroff, and E. J. Delp, "Error concealment in embedded zerotree wavelet codecs," *Proceedings of the International Workshop on Very Low Bit Rate Video Coding*, October 8-9, 1998, Urbana, IL, pp. 200-203.
- [27] W. A. Pearlman, "High Performance, Low Complexity Image Compression," *Proceedings of the SPIE Conference on Applications of Digital Image Processing X*, vol. 3164, July 1997, pp. 234-246.
- [28] K. Shen, *A study of real-time and rate scalable video and image compression*, Ph.D. Thesis, School of Electrical and Computer Engineering, Purdue University, December 1997.