Image Compression Using A Modified SPIHT Algorithm And Linear Prediction

Torsten Radtke, Volker Zerbe
Faculty of Computer Science and Automation, Technical University of Ilmenau,
P. O. Box 100565, D-98684 Ilmenau, Germany
e-mail: zerbe@theoinf.tu-ilmenau.de

Abstract

Conventional image compression algorithms treat images as 2-dimensional data and do not allow to compress an image if only parts are available. The algorithm we present here compresses successive rows by a modified SPIHT algorithm and uses linear prediction to "forecast" coefficients in low-frequency bands. Due to the low memory requirements it is suited for applications with memory limitations. Because the algorithm uses only the last two image rows instead of the entire image, the transmission delay is quite low. These advantages make the algorithm suitable for applications with memory limitations and delay constraints at the cost of a lower image quality.

Keywords: image compression, wavelet transform

1 Introduction

In many different fields, digitized images replace conventional analog images like photographs or X-ray slides. Compressed representations are required to speedup transmission and decrease storage cost. Conventional wavelet based image codecs like EZW [6] or SPIHT [5] yield excellent result and create embedded bitstreams. In these algorithms images are treated as 2-dimensional arrays leading to several disadvantages:

- Nearly any of the wavelet coefficients can be accessed at any time[1].
- It is not possible to process one part (i. e. one row) after another. Only if the entire image is located in memory the compression can start.
- Time critical applications can not afford the delay to store the whole image in memory and some systems does not have enough memory to store an entire image.

We implemented the EZW and SPIHT algorithm to transmit a series of still images at very low bandwidth (9600 baud) in realtime. At the desired frame rate of 2 to 5 frames per second the conventional algorithms lead to a delay of about 2 frames and memory requirements of more than 2 Mbit (assuming an image size of 256 x 256 pixels and 32 bit floating point numbers).

As a consequence we developed a less effective but memory saving image compression technique. After a short review on SPIHT we describe the major components of our algorithm.

2 Review: Set Partitioning In Hierarchical Trees

Typical image coders consist of three components:

- transformation (i. e. wavelet or discrete cosine transform)
- quantization
- lossless encoding (i. e. entropy or arithmetic encoding)

The wavelet transform provides a multiresolution representation of the image. Coefficients can represent long data lags corresponding to a narrow band, low frequency range or short data legs corresponding to a wide band, high frequency range. Smooth areas, sharp edges and scales in between are available. For further information we refer to [2, 7].

Because there are much more coefficients representing possible edges than smooth areas, much of the information is contained in the position of coefficients corresponding to significant edges. Said & Pearlman developed a technique which creates an efficient representation of the positions of significant coefficients.

The set of pixels is divided into partitioning subsets $\mathcal{T}_m$ and function (1) will be evaluated.

$$S_n(\mathcal{T}) = \begin{cases} 1, & \max_{(i,j)\in\mathcal{T}} \{|c_{i,j}|\} \geq 2^n \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

[1] This does not apply to the wavelet transform.

If it returns 0 the decoder knows that all coefficients in the set are insignificant. It returns 1 if the subset is significant. Then a rule is used to partition $\mathcal{T}_m$ into new subsets $\mathcal{T}_{m,l}$ and the function is used to compute the significance of the subsets. This set partitioning continues until the function is used to determine significance of single coordinate subsets.
The spatial orientation tree defines the spatial relationship in a hierarchical subband system. Each tree node corresponds to a pixel and is identified by the pixel coordinate. The pixels of the same spatial location in the next finer scale are the offspring of a node. Each node of the tree has either no offspring (pixels in the finest scales) or 4 adjacent offspring in a 2 x 2 group. Pixels in the LL-band are tree roots and also grouped in 2 x 2 adjacent pixels. Their offspring branching rule is different, in each group one of them has no descendants.
Four different sets of coordinates are used by the set partitioning rules:

- $\mathcal{O}(i,j)$: set of coordinates of all offspring of point (i,j)
- $\mathcal{D}(i,j)$: set of coordinates of all descendants of point (i,j)
- $\mathcal{H}$: set of coordinates of all spatial orientation tree roots
- $\mathcal{L}(i,j) = \mathcal{D}(i,j) - \mathcal{O}(i,j)$

The set partitioning rules are the following:

1. The initial partition is created with the sets $\{(i,j)\}$ and $\mathcal{D}(i,j)$, for all $(i,j) \in \mathcal{H}$.
2. If $\mathcal{D}(i,j)$ is significant, it is partitioned into $\mathcal{L}(i,j)$ and the four single-element sets $(k,l) \in \mathcal{O}(i,j)$.
3. If $\mathcal{L}(i,j)$ is significant, then it is partitioned into four sets $\mathcal{D}(k,l)$, with $(k,l) \in \mathcal{O}(i,j)$.

In a coarse description the compression algorithm consists of a number of alternating sorting and refinement passes.

In the sorting pass the significance of sets of coordinates is evaluated by function (1). The partitioning rules above are applied if the subset is not a single coordinate subset. In this case its coordinate is stored in a list.

The refinement pass reduces the uncertainty interval of each coefficient in the list. For a detailed explanation, see [5].

3 Compression Of Image Rows

In this algorithm a "row" is a 1-dimensional data structure the wavelet transform is applied to. Therefore it is applicable to image columns, too. In our experiments we use the 9/7-tap biorthogonal filters from [2]. These filters show better results in our 2-dimensional SPIHT implementation than the filters from [1]. The symmetry of the filters allows a simple implementation of the symmetric extension of image edges.

Similarities over several rows are handled by the prediction of wavelet coefficients. The prediction error is then encoded in the 1-dimensional SPIHT algorithm.

3.1 The Modified SPIHT Algorithm

Only minor changes were needed to adapt our implementation of the SPIHT algorithm to 1-dimensional arrays. Because we use a 1-dimensional index even in our 2-dimensional algorithm[2], the changes affect only the initial partitioning and the number of offspring: no or two offspring are possible now (figure 1).

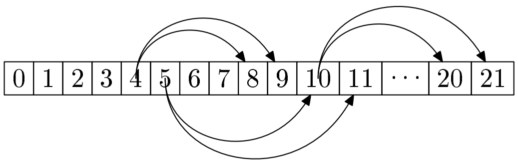

Figure 1: Parent-offspring dependencies in a one-dimensional array

Because the size of the L_n band is set to 8 the first 4 indices are not available to the sets $\mathcal{D}(i)$. The initial partitioning consists of the indices $0 \ldots 7$ for the *list of insignificant pixels* and the indices $4 \ldots 7$ for the *list of insignificant sets.*

3.2 Linear Prediction Of Coefficients

To get better compression results linear prediction of wavelet coefficients in consecutive rows is used. The coefficients of the last two rows are used to predict coefficients in the actual row. The prediction is shown in figure 2 and described in (2).

$$\hat{f}_i = f_{i-1} + (f_{i-1} - f_{i-2}) = 2f_{i-1} - f_{i-2} \qquad (2)$$

(3) describes the computation of the prediction error, assuming the enumeration of rows starts at 0.

[2] If one enumerates the image from left to right, top to bottom, starting with 0, the offspring branching rule is $\mathcal{O}(i) = \{2i, 2i+1\}$, except a small area.

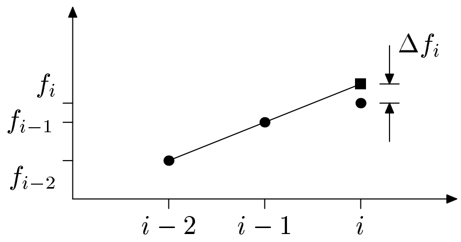

Figure 2: Linear prediction and prediction error

$$\Delta f_i = \begin{cases} f_i - f_{i-1} & ,i = 1 \\ f_i - (2f_{i-1} - f_{i-2}) & ,i \geq 2 \end{cases} \tag{3}$$

In a first approach we predicted the wavelet transform coefficients of all bands leading to the following problem: coefficients in successive rows started to oscillate. The explanation is the following: because of the small dependencies between adjacent coefficients in high frequency bands, the prediction error can become much more than the coefficient value. Coefficients with large magnitudes are treated first in the SPIHT algorithm, so few low-frequency coefficients—which determine the overall appearance of the image—remain in the compressed representation. In our experiments the best results were achieved on image "Lena" with prediction of the first 32 to 64 low-frequency coefficients at a target size of 32 bytes per row.

Figure 3 shows a magnification of a 128 x 128 region in the compressed image "Lena". The high independence between consecutive rows causes the visible "stripes" in a compressed image. In opposition to figure 5 the prediction is directed from top to bottom.

4 Results

Due to the embedded fashion of the generated bitstream the SPIHT algorithm can reach a target bit rate or file size exactly. In our case the algorithm can reach the desired size of one row exactly. Table 1 shows the results of the conventional SPIHT coder and our new algorithm on image "Lena". Image quality is measured by the peak signal-to-noise-ratio (PSNR[3]).

Note that the prediction direction is from left to right and not from top to bottom. There are differences in measured image quality between the compression directions. On image "Lena" PNSR decreased by 2.33 dB at 8192 bytes to 3.17 dB at 16384 bytes if the prediction is directed from top to bottom.

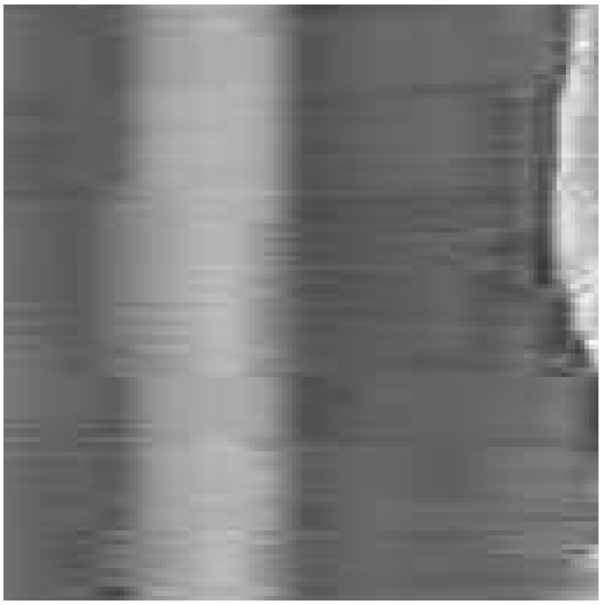

Figure 3: Magnification of a 128 x 128 pixel region in the compressed image "Lena"

comp.	#bytes	entire image	row-by-row
8:1	32768	39.90 dB	36.09 dB
16:1	16384	36.75 dB	31.49 dB
32:1	8192	33.63 dB	26.91 dB

Table 1: Comparison of a conventional SPIHT coder and the new algorithm

In Table 2 the effect of prediction of the first 32 lowest frequency coefficients is shown. One can see that the prediction becomes more important with higher compression. As a result of the prediction it is impossible to use only the first part of the bitstream to decode the image. Because coefficient values are used to predict values of the next row, a changed value would lead to a wrong prediction. The original image "Lena" and a compressed version at a target size of 32 bytes per row are shown in figures 4 and 5.

comp.	#bytes/row	w/o prediction	prediction
8:1	64	35.50 dB	36.09 dB
16:1	32	30.45 dB	31.49 dB
32:1	16	25.60 dB	26.91 dB

Table 2: Effect of prediction on measured image quality

[3] For definitions, see [5, 7].

Figure 4: Original image “Lena”

Figure 5: Compressed image “Lena” at 32 bytes per row

5 Summary

The algorithm presented here has several advantages over 2-dimensional compression systems: it supports the coding of successive rows[4] and minimizes the transmission time delay.

Small performance gains were achieved by linear prediction of coefficients in low-frequency bands. To increase the robustness on noisy channels techniques described in [3, 4] are not neccessary if no prediction is used. In this case each row is a synchronization point. Arithmetic encoding of SPIHT symbols could result in higher PSNRs at the cost of a slower compression.

The algorithm is applicable to a wide variety of problems: wherever data is collected row-by-row (e. g. scanner or telefax) and the required image quality is low, this image compression technique is suited for.

References

[1] E. H. Adelson, E. Simoncelli, and R. Hingorani, “Orthogonal Pyramid Transform for Image Coding”, Proc. SPIE, vol. 845, pp. 50–58, 1987

[2] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, “Image Coding Using Wavelet Transform”, IEEE Transactions on Image Processing, vol. 1(2), pp. 205–220, April 1992

[3] C. D. Creusere, “A New Method of Robust Image Compression Based on the Embedded Zerotree Wavelet Algorithm”, IEEE Transactions on Image Processing, vol. 6(10), pp. 1436–1442, October 1997

[4] J. K. Rogers and P. C. Cosman, “Robust Wavelet Zerotree Image Compression with Fixed-length Packetization”

[5] A. Said and W. A. 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(3), pp. 243–249, June 1996

[6] J. M. Shapiro, “Embedded Image Coding Using Zerotrees of Wavelet Coefficients”, IEEE Transactions on Signal Processing, vol. 41(12), pp. 3445–3462, December 1993

[7] M. Vetterli and J. Kovačević, “Wavelets and Subband Coding”, Prentice Hall, New Jersey, 1995

[4] Without linear prediction it supports independent coding of rows.