Cascaded Differential and Wavelet Compression of Chromosome Images

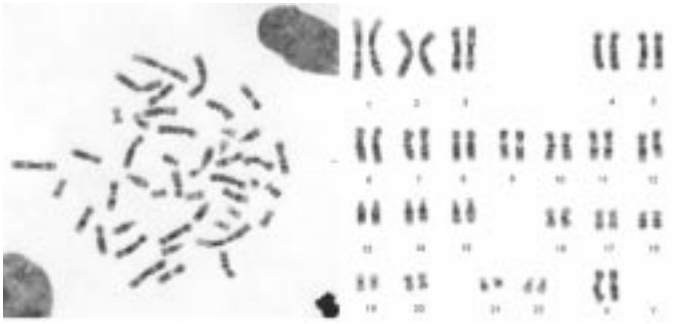
Zhongmin Liu, *Student Member, IEEE*, Zixiang Xiong, *Member, IEEE*, Qiang Wu*, *Member, IEEE*, Yu-Ping Wang, *Member, IEEE*, and Kenneth Castleman, *Member, IEEE*

***Abstract*—This paper proposes a new method for chromosome image compression based on an important characteristic of these images: the regions of interest (ROIs) to cytogeneticists for evaluation and diagnosis are well determined and segmented. Such information is utilized to advantage in our compression algorithm, which combines lossless compression of chromosome ROIs with lossy-to-lossless coding of the remaining image parts. This is accomplished by first performing a differential operation on chromosome ROIs for decorrelation, followed by critically sampled integer wavelet transforms on these regions and the remaining image parts. The well-known set partitioning in hierarchical trees (SPIHT) (Said and Perlman, 1996) [1] algorithm is modified to generate separate embedded bit streams for both chromosome ROIs and the rest of the image that allow continuous lossy-to-lossless compression of both (although lossless compression of the former is commonly used in practice). Experiments on two sets of sample chromosome spread and karyotype images indicate that the proposed approach significantly outperforms current compression techniques used in commercial karyotyping systems and JPEG-2000 compression, which does not provide the desirable support for lossless compression of arbitrary ROIs.**

***Index Terms*—Chromosome spread and karyotype images, differential operations, integer wavelet transform, JPEG-2000, lossy and lossless compression, region-of-interest coding, SPIHT.**

I. Introduction

CHROMOSOME karyotyping analysis [2] is an important screening and diagnostic procedure routinely performed in clinical and cancer cytogenetic labs. Chromosome spread images are acquired through microscope imaging and subsequently analyzed for individual chromosome segmentation, orientation, measurement, and classification. The result of this procedure is a so-called karyotype image in which all chromosomes in a cell are graphically arranged according to an international system for cytogenetic nomenclature (ISCN) [3] classification. Fig. 1 shows a typical (512×512, 8 bits/pixel) G-banding metaphase cell spread and a karyotype of all the chromosomes in that cell. Ordinarily, in the practice of clinical cytogenetics, both the chromosome spread and karyotype images are saved for additional medical opinions and for medical record keeping. With the recent development in the use of digital media for biomedical image archiving, storage, and communication, efficient compression techniques are highly desirable to accommodate the rapid growth of chromosome image data.



(a) (b)

Fig. 1. (a) A metaphase cell spread image and (b) its karyotype. In the karyotype, all chromosomes in the spread are rotated and copied onto an image with constant background and positioned according to their classes. The label annotation is drawn separately.

Image compression[1] techniques generally fall into two categories: lossy and lossless compression. Some information is irretrievably lost in lossy compression, whereas there is no loss of information in lossless compression, i.e., the coding process is reversible. Lossy compression generally achieves higher compression ratios than lossless compression. However, lossy compression has found only limited use in medical applications because image information is critical for clinical evaluation and diagnosis cannot be compromised. As such, commercial karyotyping systems currently store entire chromosome spread or karyotype images in the TIFF format and use lossless techniques such as Lempel–Ziv–Welch (LZW) coding [4], [5] for compression.

Unlike some other types of medical imagery, chromosome images (see Fig. 1) have an important common characteristic: the regions of interest (ROIs) to cytogeneticists for evaluation and diagnosis are all well determined and segmented prior to image storage. The remaining background images, which may contain cell nuclei and stain debris, are kept as well in routine cytogenetics lab procedures for specimen reference rather than for diagnostic purposes. Since the chromosome ROIs are much more important than the rest of the image for karyotyping analysis, lossless compression for the former is required while lossy

[1]We use image coding and image compression interchangeably in this paper.

Manuscript received April 24, 2001; revised November 28, 2001 This work was supported in part by the National Institutes of Health (NIH) under SBIR Grant 1R43GM/CA62724-01, the National Science Foundation (NSF) under CAREER Grant MIP-00-96070, the NSF under Grant CC-01-04834, the Army Research (ARO) YIP under Grant DAAD19-00-1-0509, and the Office of Naval Research (ONR) under YIP Grant N00014-01-1-0531. *Asterisk indicates corresponding author.*

Z. Liu and Z. Xiong are with Department of Electrical Engineering, Texas A&M University, College Station, TX 77843 USA.

*Q. Wu is with Advanced Digital Imaging Research, LLC. 2450 S. Shore Blvd., Ste. 305, League City, TX 77573 USA (e-mail: qwu@adires.com).

Y.-P. Wang and K. Castleman are with Advanced Digital Imaging Research, LLC. League City, TX 77573 USA.

Publisher Item Identifier S 0018-9294(02)02992-0.

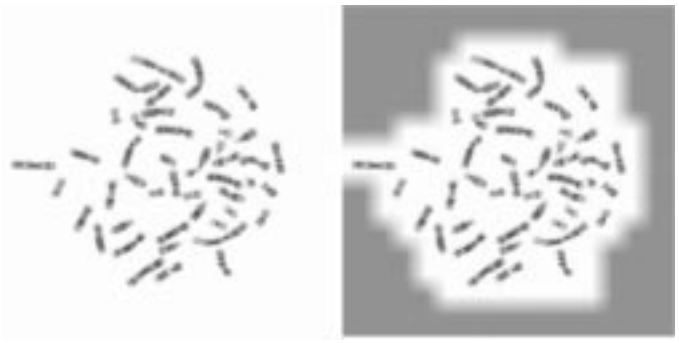
compression for the latter is acceptable. This calls for lossy and lossless ROI (ROI) coding.[2] In contrast, commercial chromosome karyotyping systems fail to utilize the ROI information by compressing entire chromosome spread or karyotype images.

The new wavelet-based JPEG-2000 standard [6] offers many features, including ROI coding which is accomplished by assigning higher priority in the coding process to wavelet coefficients *in and around* the ROIs. However, transform and coding of the ROIs and the background image are not done separately in JPEG-2000—there is no clear separation between the ROIs and the background in the wavelet domain as there is in the image domain. Reconstruction of the image domain ROIs, thus, requires wavelet coefficients from a larger region, whose size depends on the filter length and the levels of wavelet decomposition. Unless lossless compression of the whole rectangular image is achieved, there is no guarantee of lossless compression of the ROIs. In short, JPEG-2000 supports lossless compression of the whole image but not of arbitrary ROIs.

To improve the efficiency of LZW coding and rectify the above-mentioned shortcoming of JPEG-2000, we propose a new method in this paper, one which takes advantage of the ROI information and seeks to code chromosome images adaptively with respect to image content. Specifically, we aim to render lossless compression inside the chromosome ROIs, while achieving lossy-to-lossless compression for the rest of the image, based on a combination of differential and wavelet coding techniques. We accomplish our goal by first performing a differential operation on the chromosome ROIs [7], followed by separate *critically sampled* integer wavelet transforms on the chromosome ROIs and on the remaining image parts. We modify the celebrated set partitioning in hierarchical trees (SPIHT) [1] algorithm for ROI coding and generate separate *embedded* bit streams for both the chromosome ROIs and the rest of the image. An embedded bit stream has the property that each additional bit improves somewhat upon the quality of the decoded image and that the whole bit stream can be truncated at any point to provide a decoded image with quality commensurate with the bit rate. Although we typically insist upon lossless compression of the chromosome ROIs, lossy compression of these regions can also be achieved simply by decoding at lower bit rates than the encoding one. That is, both lossy and lossless compression modes for chromosome ROIs and the rest of the image are available.

We pick the SPIHT coder because it has lower complexity than the JPEG-2000 coder but achieves comparable coding performance for regular rectangular images. Experiments on two sets of sample chromosome spread and karyotype images indicate that our proposed technique significantly outperforms those (e.g., LZW coding) currently used in commercial karyotyping systems. In addition, by treating the chromosome ROIs and the remaining image parts separately with critically sampled wavelet transform and modified SPIHT coding for each part, we achieved beyond what JPEG-2000 can offer in terms of lossless ROI coding.

[2]One simple way to compress the chromosome ROIs is to extract all pixels inside the ROIs in one file and use WinZip to compress the resulting file.



(a) (b)

Fig. 2. (a) The image corresponding to the chromosome ROIs of Fig. 1(a) with a white background. Our proposed algorithm can losslessly compress these chromosome ROIs to 23 219 bytes. (b) The lossy image decoded from a JPEG-2000 bit stream of 23 219 bytes.

To illustrate our contribution in this paper, we present an example that compares different compression schemes for the image in Fig. 2(a), which corresponds to the chromosome ROIs of Fig. 1(a) with a white background. There are 39 289 pixels within the ROIs. WinZip (Version 8.0) compresses a file of 39 289 bytes consisting of these pixels into 35 184 bytes. Our proposed wavelet-based scheme can losslessly compress these chromosome ROIs into 23 219 bytes—achieving a 35% savings over LZW coding (bits needed for specifying the ROI boundaries are not counted in either case, actually 2258 bytes are used to store the ROI boundaries).

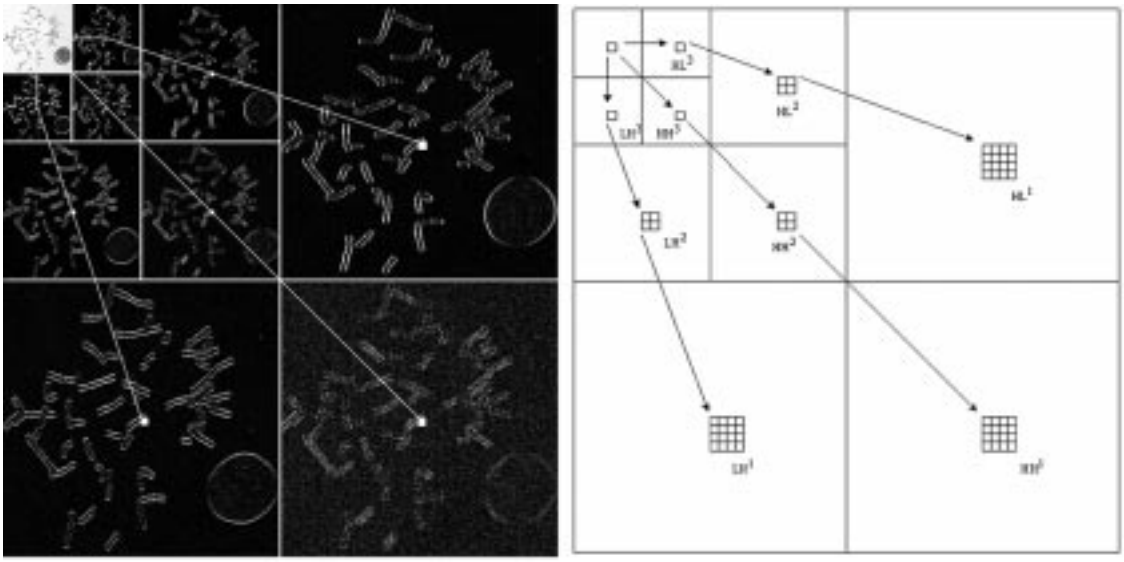
We also use JPEG-2000 with the chromosome ROI support to losslessly compress the 512×512 image with white background and the compression result is 35 514 bytes.[3] The JPEG-2000 bit stream is made up of layered, truncated versions of the lossless bit stream that can be decoded into different lossy images. We use the first 25 447 (23 219 plus 2258) bytes of the bit stream to decode a lossy version shown in Fig. 2(b). Pixels both in and around the ROIs are different from the original, with the peak signal-to-noise ratio (PSNR)[4] in the chromosome ROIs being 47.84 dB. Thus, using JPEG-2000 instead of our proposed scheme for compressing chromosome ROIs leads to either inferior lossless coding performance or quality loss at the same bit rate.

As SPIHT-based ROI coding of images and video is still an active area of research [8]–[10], we point out that, although our new algorithm is developed specifically for chromosome image compression, it can be used for general ROI image compression and extended to object-based video coding [11], [12] for MPEG-4.

The rest of this paper is organized as follows: Section II reviews wavelet image coding. In Section III, the differential operation is motivated first, followed by a detailed description of our proposed coding scheme. Section IV presents experimental results on two representative chromosome spread and karyotype image sets and compares the proposed compression method with the technique currently used in commercial karyotyping systems. Section V concludes the paper.

[3]As our proposed algorithm can be applied to any ROI image, it achieves lossless coding at 33 528 bytes for the same image, confirming that SPIHT and JPEG-2000 coding perform comparably for regular rectangular images.

[4]The PSNR is defined as $10\log_{10}(255^2/\text{MSE})$ and measured in decibels (dB).



(a) (b)

Fig. 3. Wavelet decomposition offers a tree-structured image representation. (a) A three-level wavelet decomposition of a chromosome spread image. (b) A spatial orientation tree consisting of coefficients from different bands that correspond to the same spatial region of the original image. Arrows identify the parent-children dependencies.

II. Wavelet Image Coding

Since the introduction of the wavelet transform [13], [14] as a signal processing tool in the late 1980s, a variety of wavelet-based coding algorithms [1], [15] have advanced the limits of compression performance well beyond that of the current commercial JPEG image compression standard [16]. These algorithms achieve twice as much compression as the baseline JPEG coder does at the same quality and they form the basis of the new JPEG-2000 [6] image compression standard.

The improved performance of wavelet image coding [17] over JPEG coding stems from the fact that wavelet decompositions offer *space-frequency* representations of images, i.e., low-frequency coefficients have large spatial support (good for representing large image background regions), whereas high-frequency coefficients have small spatial support (good for representing spatially local phenomena such as edges). The wavelet representation, therefore, calls for new quantization strategies that go beyond traditional subband coding [18] techniques (e.g., bit allocation and deadzone quantization) to exploit this underlying space-frequency image characterization.

Shapiro made a breakthrough in 1993 with his embedded zerotree wavelet (EZW) coding algorithm [15]. Since then a new class of algorithms have been developed that achieve significantly improved performance over the EZW coder. In particular, Said and Pearlman's work on SPIHT [1], which improves the EZW coder, has established zerotree-based techniques as the current state-of-the-art of wavelet image coding since the SPIHT algorithm proves to be very efficient for both lossy and lossless compression.

A. SPIHT Coding

A wavelet image representation can be thought of as a tree-structured spatial set of coefficients. *A spatial orientation tree* is defined as the set of coefficients from different bands that represent the same spatial region in the image. Fig. 3 shows a three-level wavelet decomposition of a chromosome spread image and a spatial orientation tree. Arrows in Fig. 3(b) identify the parent-children dependencies in a tree. The lowest frequency band of the decomposition is represented by the root nodes (top) of the tree, the highest frequency bands by the leaf nodes (bottom) of the tree, and each parent node represents a lower frequency component than its children. Except for a root node, which has only three children nodes, each parent node has four children nodes, the 2×2 region of the same spatial location in the immediately higher frequency band.

Both the EZW and SPIHT algorithms [1], [15] are based on the idea of using multipass zerotree coding to transmit the largest wavelet coefficients (in magnitude) at first. We use "zerotree coding" as a generic term for both schemes, although the SPIHT coder is more popular because of its superior performance. A set of tree coefficients is significant if the largest coefficient magnitude in the set is greater than or equal to a certain threshold (e.g., a power of two); otherwise, it is insignificant. Similarly, a coefficient is significant if its magnitude is greater than or equal to the threshold; otherwise, it is insignificant. In each pass the significance of a larger set in the tree is tested at first: if the set is insignificant, a binary "zerotree" bit is used to set all coefficients in the set to zero; otherwise, the set is partitioned into subsets (or child sets) for further significance tests. After all coefficients are tested in one pass, the threshold is halved before the next pass.

The underlying assumption of the zerotree coding framework is that most images can be modeled as having decaying power spectral densities. That is if a parent node in the wavelet coefficient tree is insignificant, it is very likely that its descendants are also insignificant. The zerotree symbol is used very efficiently in this case to signify a spatial subtree of zeros.

When the thresholds are powers of two, SPIHT coding can be thought of as a bit-plane coding scheme. It encodes one bit-plane at a time, starting from the most significant bit. With the sign bits

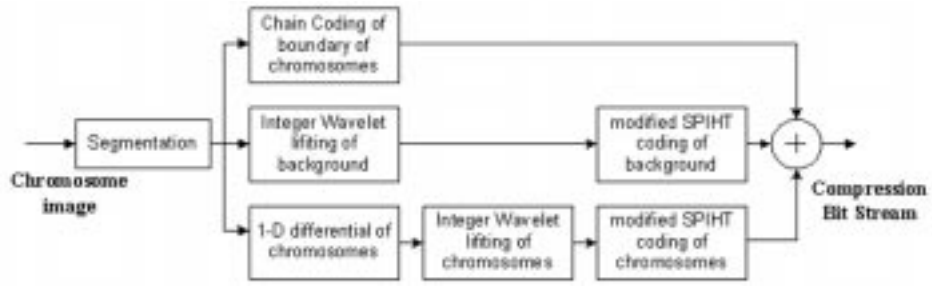


Fig. 4. The proposed compression algorithm.

and refinement bits (for coefficients that become significant earlier) being coded on the fly, SPIHT achieves embedded coding in the wavelet domain using three lists: the list of significant pixels (LSP); the list of insignificant pixels (LIP); and the list of insignificant sets (LIS). The SPIHT coder performs competitively with most other coders published in the literature [19], while possessing desirable features such as relatively low complexity and rate embeddedness.

B. JPEG-2000

In response to the rapid progress in wavelet image coding research, the International Standards Organization has adopted the wavelet transform as the workhorse in the new JPEG-2000 image coding standard. The baseline JPEG-2000 coder employs the embedded block coding with optimized truncation (EBCOT) [20] algorithm for bit-plane coding of wavelet coefficients. While the SPIHT algorithm applies arithmetic coding [21] on the significant bits only, EBCOT additionally uses arithmetic coding on the sign bits and refinement bits. Furthermore, EBCOT breaks one bit-plane into three *fractional* bit-planes and compresses them in decreasing order of rate-distortion (R-D) importance. Because of this, the complexity of JPEG-2000 coding is higher than that of SPIHT coding.

In terms of compression efficiency, JPEG-2000 performs comparably to SPIHT. The strength of the JPEG-2000 standard lies in its rich set of features such as lossy and lossless compression, scalability in rate and image resolution, ROI coding, open architecture, and robustness to bit errors, to name a few. We refer curious readers to a recently published comprehensive book on JPEG-2000 [6] for details.

Here, we emphasize the fact that JPEG-2000 implements "soft" ROI coding: not only are pixels in the ROIs decoded with much better quality; pixels around the ROIs also get favorable treatment, albeit to a lesser extent than those in the ROIs [see Fig. 2(b)]. As mentioned in Section I, this is because the wavelet transforms for the ROIs and the background image are not done separately in JPEG-2000. For applications where clear separations of ROIs and the background are not insisted upon, JPEG-2000 offers a smooth (or soft) transition of image quality across ROI boundaries.

III. Cascaded Differential and Wavelet Coding

Before detailing the proposed coding scheme, we point out some important differences between chromosome spread and karyotype images. In chromosome spread images, the background regions outside chromosomes usually contain materials such as interphase cell nuclei, stain debris and transmitted light microscope shading. Chromosomes in spread images are randomly oriented. Segmentation of these chromosomes is first done automatically, followed by user interaction to ensure that all the chromosomes are properly located and isolated. In karyotype images, however, the segmented chromosomes from a spread image are re-oriented before copying onto a constant-background image and graphically arranged according to their ISCN classification.

The coding method we propose here seeks to encode the chromosome images adaptively with respect to the region contents in the image. It is aimed to render lossless compression inside the chromosome ROIs for both the spread and karyotype images. As to the background regions, although practically there is no diagnostic information available from these regions in both types of images, our method is still slated to incorporate a lossy-to-lossless coding provision for the background regions of spread images as an option.

The overall design of the coding scheme is based on a combination of differential and wavelet coding operations. Initially, a differential operation is performed on chromosome ROIs only for decorrelating a chromosome image. After this, critically sampled integer wavelet transforms are computed for the chromosome differentials and the remaining image parts separately. A modified SPIHT algorithm is then applied to generate embedded bit streams that allow continuous lossy-to-lossless compression, depending upon whether the pixels are inside the chromosome ROIs or not. Fig. 4 shows the diagram of the proposed compression scheme. Since the background regions in karyotype images are constant-valued and trivial to code, in the remainder of this paper, we limit the discussion to chromosome spread images only unless otherwise stated.

The differential operation is motivated first in the sequel, followed by a detailed description of each part of our proposed coding scheme.

A. Differential Operations

Chromosomes in spread and karyotype images are all segmented as ROIs. Compression of these chromosome ROIs can be better approached by a combination of differential and wavelet techniques than the LZW coding. In traditional predictive image coding [22], an image is modeled as a first-order autoregressive process

$$X(n) = \rho X(n-1) + W(n)$$

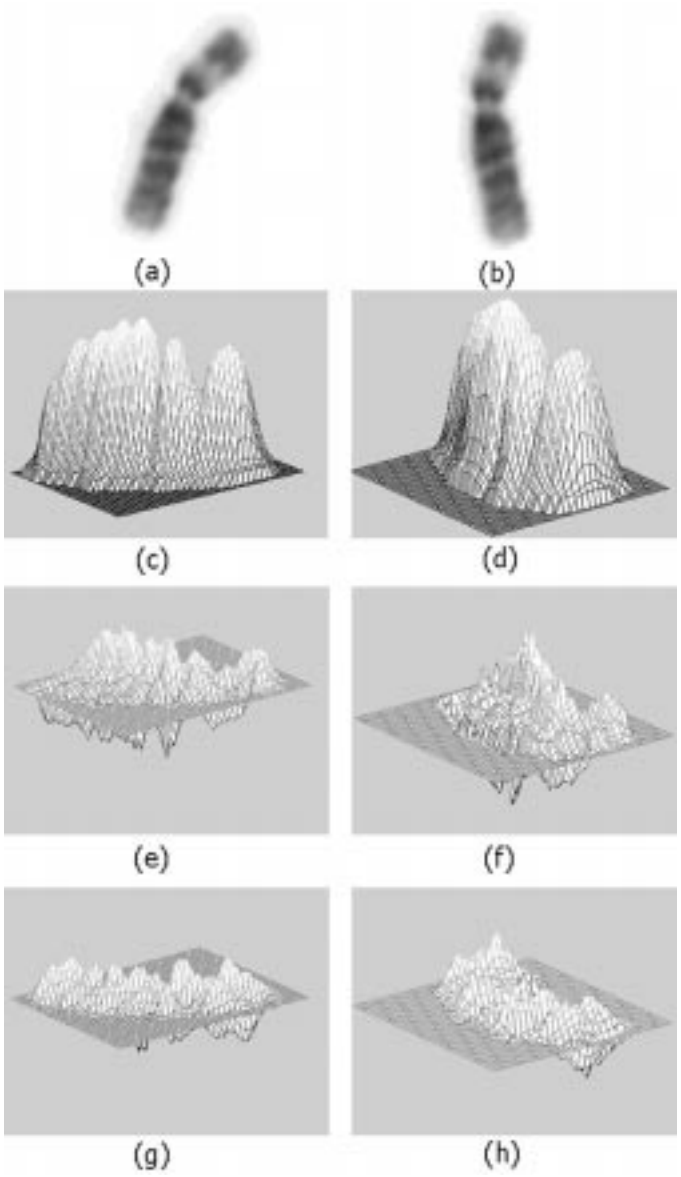


Fig. 5. (a) A chromosome in a spread image. (b) The same chromosome in the karyotype image. (c) Three-dimensional (3-D) view of the chromosome in (a). (d) Three-dimensional view of the chromosome in (b). (e) Three-dimensional view of the chromosome in (a) after 1-D horizontal differentiation. (f) Three-dimensional view of the chromosome in (b) after 1-D horizontal differentiation. (g) Three-dimensional view of the chromosome in (a) after 1-D vertical differentiation. (h) Three-dimensional view of the chromosome in (b) after 1-D vertical differentiation.

where ρ is the correlation coefficient between neighboring pixels $X(n)$ and $X(n-1)$ and $W(n)$ is the stationary zero-mean innovations process that is independent of past image pixels $X(m)$ for $m < n$. Note that dependencies among rows and columns are ignored in this simple one-dimensional (1-D) model. The best linear mean square predictor of the current pixel $X(n)$ is $\rho X(n-1)$. It is more efficient to code the innovations process

$$W(n) = X(n) - \rho X(n-1)$$

using differential pulsecode modulation coding [23] than to code the original image $X[n]$ because $\sigma_W^2 = (1-\rho^2)\sigma_X^2$, i.e., the variance of $W(n)$ is smaller than that of $X(n)$.

Image pixels in general are highly correlated. This is also true for chromosome images. For example, Fig. 5(a) and (b) shows a single chromosome extracted and enlarged from a spread image and its corresponding karyotype; Fig. 5(c) and (d) shows their 3-D representations. The neighboring pixels in these images are highly correlated (with $\rho \approx 0.90$). Because our interest lies in lossless compression of chromosome ROIs and ρ is close to one for chromosome images, we approximate ρ as one so that

$$E(n) = X(n) - X(n-1)$$

is still an integer sequence. We, thus, code the pixel-to-pixel differentials rather than the pixels themselves.

Although $X(n-1)$ is not the best linear mean square predictor of $X(n)$, we have the variance of $E(n)$ as

$$\sigma_E^2 = 2(1-\rho)\sigma_X^2 \tag{1}$$

which is less than σ_X^2 if $\rho > 1/2$. Given $\rho \approx 0.90$, we conclude that $\sigma_E^2 < \sigma_X^2$ for chromosome images. Fig. 5(e) and (f) shows the 1-D horizontal differentials and Fig. 5(g) and (h) shows the 1-D vertical differentials of the chromosome and karyotype images, respectively. It is easy to see from these figures that the variance (or energy) of horizontal/vertical differentials is smaller than that of the original chromosome image. Thus, it is easier to compress the differentials than the original chromosome images using the wavelet transform.

For karyotype images, recall that the segmented chromosomes from a spread image are all rotated to the vertical orientation [see Fig. 1(b)], pixel correlation is stronger along this direction in these images. According to (1), we choose differentials along the vertical direction (with higher correlation) so that σ_E^2 is smaller for better performance in the subsequent step of wavelet compression. This is done by computing the 1-D vertical differentials on pixels inside the boundary of each chromosome region and replacing each with values of its vertical differential.

For spread images, because the chromosomes are randomly oriented, we do not expect the choice of differential orientation to affect the coding efficiency as much. This is confirmed by our experiments in Section IV.

B. Lifting-Based Wavelet Transforms of Chromosome ROIs

Many wavelet filters have been designed and used for various applications. Different wavelet filters are compared for lossy image compression in [24] and lossless image compression in [25]. In general, the 5/3 filters [26] outperform other wavelet filters for lossless compression, while the Daubechies 9/7 filters [27] are the overall best for lossy compression. These filters are chosen as the default filters for lossless and lossy image compression, respectively, in JPEG-2000.

It was shown in [28] that every finite impulse response wavelet or filter bank can be decomposed into lifting steps and that each lifting step can be further split into two additions and one multiplication. In addition to achieving as much as a $2\times$ speed-up over filtering-based implementation, the lifting-based approach makes it very easy to have an integer-to-integer mapping. Because image pixels take integer values, integer wavelet transforms are needed for lossless image compression [26];

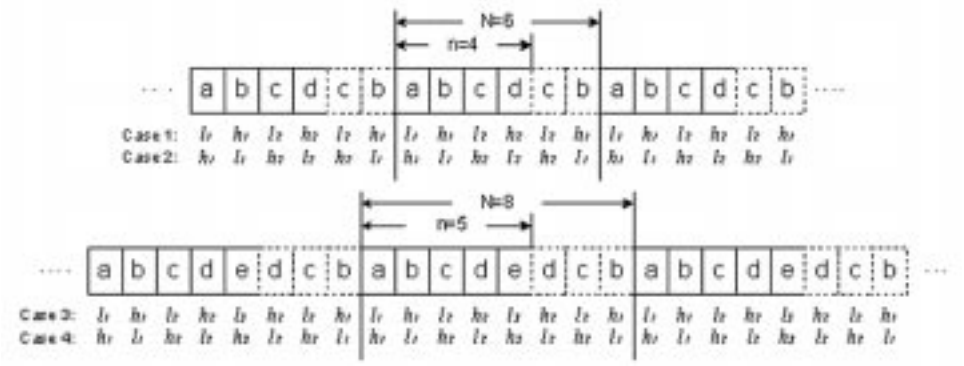


Fig. 6. Critically sampled wavelet transform for both even and odd length signals using odd-length symmetric biorthogonal filters and odd-symmetric extensions over boundaries.

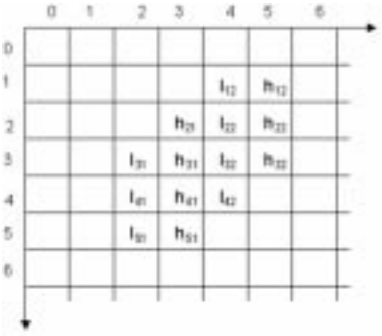
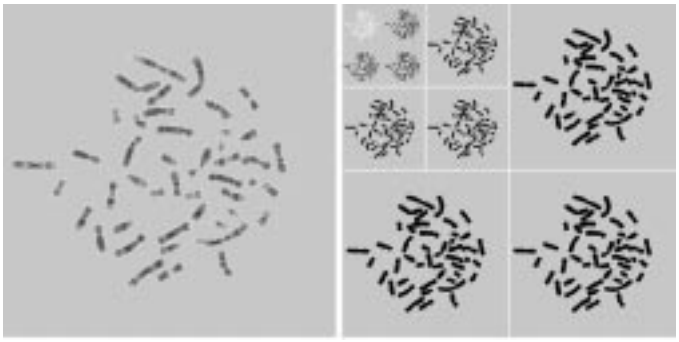


Fig. 7. Example of two-dimensional (2-D) ROI with horizontal low-pass and high-pass coefficients aligned vertically.



(a) (b)

Fig. 8. Wavelet representation of the chromosomes in one spread image. (a) The original image. (b) A three-level critically sampled integer wavelet transform of the chromosomes (ROIs).

otherwise the wavelet coefficients—hence the reconstructed image pixels—will no longer consist of integers, destroying invertibility. We used both the 5/3 and 9/7 biorthogonal wavelet filters in our experiments (although the former have lower computational complexity than the latter) and implemented integer wavelet transforms based on these filters via lifting [28], [29]. The lifting-based forward transform equations for these filters can be written explicitly as

$$\frac{5}{3}: \begin{cases} d[n] = d_0[n] - \lfloor \frac{1}{2}(s_0[n+1] + s_0[n]) \rfloor \\ s[n] = s_0[n] + \lfloor \frac{1}{4}(d[n] + d[n-1]) + \frac{1}{2} \rfloor \end{cases}$$

$$\frac{9}{7}: \begin{cases} d_1[n] = d_0[n] + \lfloor \frac{1}{128}(203(-s_0[n+1] - s_0[n])) + \frac{1}{2} \rfloor \\ s_1[n] = s_0[n] + \lfloor \frac{1}{4096}(217(-d_1[n] - d_1[n-1])) + \frac{1}{2} \rfloor \\ d[n] = d_1[n] + \lfloor \frac{1}{128}(113(s_1[n+1] + s_1[n])) + \frac{1}{2} \rfloor \\ s[n] = s_1[n] + \lfloor \frac{1}{4096}(1817(d_1[n] + d_1[n-1])) + \frac{1}{2} \rfloor \end{cases}$$

where $x[n]$, $s[n]$, and $d[n]$ are the input signal, low-pass band, and high-pass band, respectively, and $s_0[n] = x[2n]$ and $d_0[n] = x[2n+1]$. The inverse transform equations can be derived easily from the forward ones.

For chromosome images, we only perform the wavelet transform within the ROIs, which are arbitrarily shaped. Many approaches have been proposed in the literature for 2-D shape-adaptive wavelet transforms [30]–[33]. In our proposed coding algorithm, we use odd-symmetric extensions over the ROI boundaries.

Consider a 1-D signal with finite length $n > 1$. We obtain a periodic signal with period $N = 2n - 2$ by adding odd-symmetric extensions over its boundaries (see Fig. 6). Note that, regardless of the signal length n, N is always even so that there are an equal number of low-pass and high-pass coefficients in each period after a one-level wavelet transform. Fig. 6 shows the boundary extensions and filtering orders for the following cases: 1) even length signal starting at an even position; 2) even length signal starting at an odd position; 3) odd length signal starting at an even position; and 4) odd length signal starting at an odd position.

If we use lifting to compute the wavelet coefficients along the extended periodic signal (the lifting steps proceed from right to left for cases 2 and 4), then the interleaved wavelet coefficient representation is also periodic with period N. Moreover, there are only n distinct coefficients in each period, with an extra low-pass or high-pass coefficient when n is odd. This means that one can compute critically sampled wavelet transforms for both even or odd length signals. In the above boundary extensions and transform, the low-pass filtering is fixed at even positions and high-pass filtering is fixed at odd positions. This feature has the advantage that, after a transform along one dimension, both low-pass and high-pass coefficients are automatically aligned for subsequent transforms along the other dimension. Fig. 7 shows that, after filtering along the horizontal direction for 2-D ROIs, all the low-pass coefficients have even x coordinates, while all the high-pass coefficients have odd x coordinates.

Fig. 8 shows a chromosome spread image and a three-level critically sampled integer wavelet transform of the chromo-

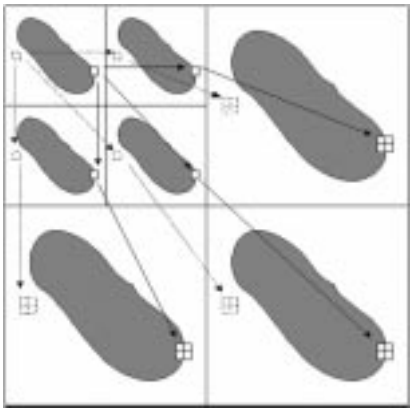


Fig. 9. Illustration of two spatial orientation trees in the wavelet-domain ROIs. All coefficients in one tree (drawn in dotted lines) are outside the ROIs. This tree is not coded in the modified SPIHT algorithm. Another tree (drawn in solid lines) has some of its coefficients outside the ROIs. Significance test of a subset in the tree is skipped if every coefficient in the subset is outside the ROIs.

somes (ROIs) via lifting. It is easy to see that an ROI in the image domain induces an ROI for each subband in the transform domain. The resulting wavelet-domain ROI will be used in the later coding stage.

C. The Modified SPIHT Algorithm

After the aforementioned cascaded differential and wavelet transform, the wavelet coefficients are encoded by a modified SPIHT algorithm. The wavelet coefficients are treated as a collection of spatial orientation trees in SPIHT, with each tree consisting of the coefficients from all subbands that correspond to the same spatial location in the image. Fig. 9 shows that the spatial orientation trees are divided into three categories: 1) All coefficients in the tree are inside the wavelet domain ROIs; 2) Some coefficients in the tree are outside the ROIs (drawn in solid lines in Fig. 9); and 3) All coefficients in the tree are outside the ROIs (drawn in dotted lines in Fig. 9).

Recall that the SPIHT algorithm maintains three lists (LSP, LIP, and LIS) in the process of bit-plane coding. It outputs three types of bits: significance bits, sign bits, and refinement bits. The modified SPIHT algorithm differs from the original one only in that the extraneous coefficients outside the ROIs are not coded. This would, in turn, require that the region boundary information (the chain code) about the ROIs is available at both the encoder and decoder. Starting with the wavelet-domain ROIs induced from that for the original image, the modified SPIHT algorithm skips coding a spatial orientation tree if all coefficients in the tree are outside the ROIs. This is simply done by not putting the coordinates of the root node (in the lowest frequency band) of the tree in the LIP and the LIS in the SPIHT initialization step.

For a spatial orientation tree with some coefficients outside the ROIs, the significance test of a coefficient in the tree is skipped if that coefficient is outside the ROIs. Likewise, the significance test of a subset in the tree is skipped if all coefficients in the subset are outside the ROIs. As sign bits and refinement bits are only associated with coefficients in the ROIs, no modification is needed in related parts of the original SPIHT algorithm for these bits.

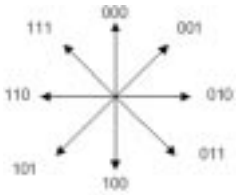


Fig. 10. The boundary direction code.

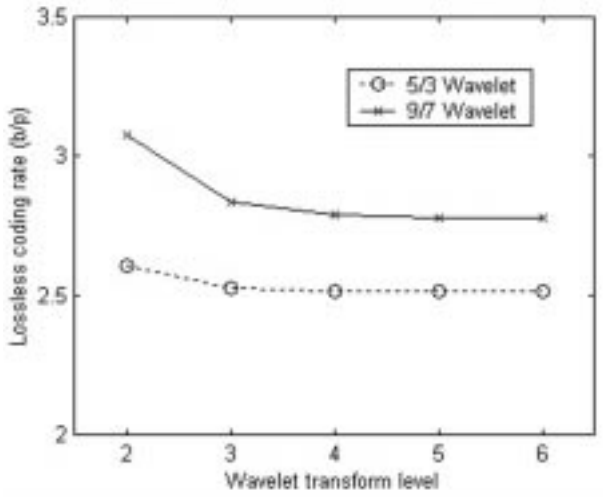
Finally, when all coefficients in a spatial orientation tree are inside the ROIs, the tree is coded in the same way as in the original SPIHT. Note that if the ROIs are the whole image, then the modified SPIHT algorithm described above will give exactly the same performance as the original SPIHT algorithm does (there is no need to use and send the region boundary chain code in this case). Thus, the modified SPIHT algorithm is more general than the original one.

With the modified SPIHT coding algorithm, lossless coding of arbitrary ROIs is achieved when all bit planes of the integer wavelet coefficient are coded. Because the lossless bit stream is embedded, it can be truncated at any point to provide a decoded image with quality commensurate with the bit rate. Thus, the algorithm provides the attractive feature of lossy-to-lossless coding.

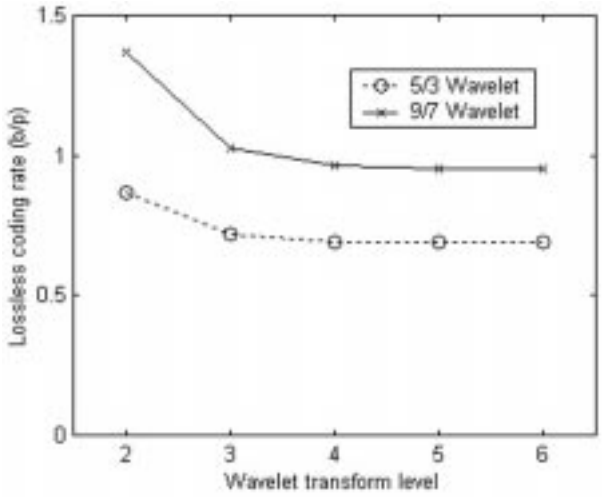
D. Chromosome Region Boundary Coding

Chromosomes in spread images are usually segmented automatically, based on a combination of image segmentation techniques [34], [35], and reviewed interactively by cytogeneticists for correction of possible machine errors. The result of the segmentation process is a set of delineated ROI region boundaries of the chromosomes. Hence, the foreground chromosome ROIs and background ROI are defined and separated.

Chromosome regions are represented compactly using the chain codes [22] along the chromosome boundaries. They contain not only the contour information of chromosome ROIs, but also the relative position of these regions in the image. A chain code starts by specifying an arbitrarily selected starting point with coordinates (x, y) located on the region boundary. The identified pixel has eight neighbors. At least one of these must also be a boundary point. The boundary chain code specifies the direction in which a step must be taken to go from the present boundary point to the next. Fig. 10 shows the eight directions of the identified pixel. Since there are eight possible directions, every direction is represented by three bits, say, from zero through seven. Thus, the boundary chain code consists of the coordinates of the starting point, followed by a sequence of direction codes that specify the path around the boundary. With the chain code, it takes only one (x, y) coordinate pair and three bits for each remaining boundary point to store the region boundary of one chromosome. We can randomly access individual chromosomes to accommodate lossless coding of the chromosome ROIs and lossy-to-lossless coding of the background. The overhead generated by the chromosome boundary chain code is usually very nominal.

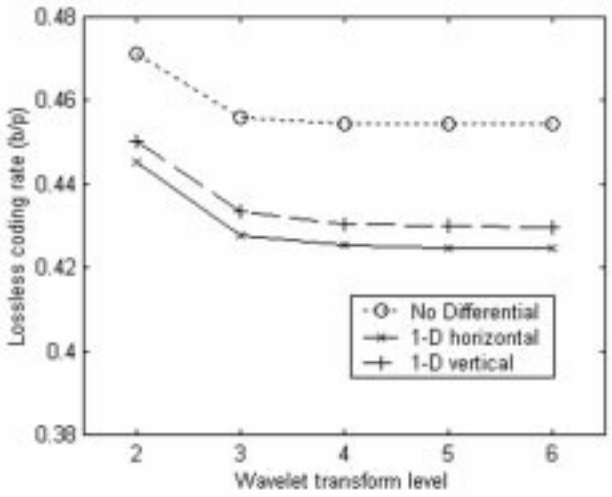


(a)

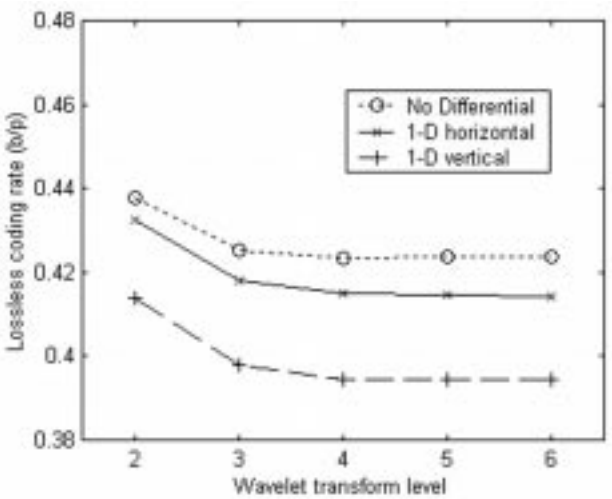


(b)

Fig. 11. Lossless compression results (in b/p) for whole spread and karyotype images by use of the modified SPIHT algorithm with the 5/3 and 9/7 wavelet filters and different wavelet decomposition levels. (a) Average lossless compression results for ten spread images. (b) Average lossless compression results for ten karyotype images.



(a)



(b)

Fig. 12. Lossless compression of chromosome ROIs in (a) ten spread and (b) ten karyotype images using cascaded differential and wavelet coding with different differential operations and levels of wavelet transform. Results are given in terms of average b/p versus wavelet transform levels.

IV. Experimental Results

Experiments are conducted to test the performance of our proposed coding scheme. We use two sets of randomly selected chromosome data: one containing ten spread images and the other one having ten karyotype images. Each image is 764×560 and each pixel has eight bits resolution. These images are fairly representative of the human chromosome spreads and karyotypes analyzed and archived in clinical cytogenetics labs.

In order to compare the compression performance objectively, we also conduct experiments based on the same data using three benchmark image compression methods: 1) storing these images in TIFF format with LZW coding[5], which is currently employed in commercial karyotyping systems; 2) LZW coding with the latest version WinZip 8.0; and 3) JPEG-2000 compression.

[5]The version of LZW coder used in TIFF is earlier than the one in WinZip 8.0

In our proposed coding scheme, lossless compression is mainly used to compress the chromosomes ROIs. During our experiments, however, we also test different methods on the whole image for comparison. In the lossy-to-lossless coding experiments, on the other hand, the background in a chromosome spread image is compressed using different bit rates, while the chromosome ROIs are compressed losslessly. To make the performance evaluation easy and consistent, all results presented in this section are described by bit rate in terms of bit per pixel (b/p) for lossless compression, and by PSNR (in decibels) for lossy compression.

A. Lossless Coding Performance

1) Performance by Use of Different Wavelet Filters and Levels of Wavelet Transform: We compare the lossless performance of SPIHT coding by use of different wavelet filters and levels of wavelet transform for the whole image. Because the chromosome image size (764×560) is not some power of two

TABLE I
LOSSLESS COMPRESSION RESULTS FOR TEN CHROMOSOME SPREAD AND TEN KARYOTYPE IMAGES. THE BIT RATE RANGE [in BITS/PIXEL (b/p)] IS REPORTED FOR EACH CODING SCENARIO, WITH THE MEAN INCLUDED IN PARENTHESES. (a) LOSSLESS COMPRESSION OF THE ORIGINAL 764×560 IMAGES. (b) LOSSLESS COMPRESSION OF ARBITRARILY SHAPED CHROMOSOME ROIS. BIT RATES USED IN CHAIN CODES FOR SPECIFYING CHROMOSOME BOUNDARIES ARE ALREADY COUNTED

	TIFF with LZW	WinZip 8.0	JPEG-2000	Modified SPIHT
Spread with original background	3.4326-4.8267 (4.1499)	3.4477-4.0405 (3.6399)	2.3708-2.5823 (2.4259)	2.4031-2.6359 (2.5167)
Spread with flat background	0.8228-1.0588 (0.9334)	0.5980-0.8567 (0.7164)	0.6642-0.9537 (0.7118)	0.6004-0.8720 (0.7214)
Karyotype (with flat background)	0.7455-1.0414 (0.8930)	0.4977-0.7910 (0.6311)	0.6089-0.8682 (0.6910)	0.6159-0.8677 (0.6897)

(a)

	WinZip 8.0	Cascaded differential and wavelet coding
Spread	0.5263-0.7393 (0.6216)	0.3268-0.4736 (0.4298)
Karyotype	0.3936-0.6596 (0.5328)	0.2840-0.4592 (0.3943)

(b)

in either dimension, we have to use the modified SPIHT algorithm even for coding the whole image. In our implementation of modified SPIHT coding, we use the 5/3 and 9/7 wavelet filters with different decomposition levels (e.g., two to six). Note that no differential operations are applied before modified SPIHT coding on the whole image.

Fig. 11 shows lossless compression results (averaged over ten images) by using the modified SPIHT on the original spread images and karyotype. The 5/3 filters give better performance than the 9/7 filters for both types of images. This is consistent with results reported in [25]. We henceforth only report results from using the 5/3 filters in lossless coding experiments. We also observe from Fig. 11 that, as the number of wavelet decomposition levels increases, the coding efficiency improves up to some optimum point and then either stays at the optimum point or even degrades slightly. Four to six levels of wavelet decomposition, thus, work the best for lossless compression of the whole image.

2) Performance by Use of Different Differential Operations: We now compare the lossless performance of cascaded differential and wavelet/SPIHT coding by use of different differential operations for chromosome ROIs. The modified SPIHT algorithm is applied on either the original chromosome ROIs (without differential operation), the 1-D horizontal differentials, or 1-D vertical differentials. The 5/3 filters are used with different levels of wavelet decomposition, and lossless compression results (averaged over ten images) are plotted in Fig. 12 for chromosome ROIs in both spread and karyotype images. Four to six levels of wavelet decomposition also work the best for lossless compression of chromosome ROIs. More importantly, Fig. 12 clearly shows the coding gain of using differential operations. In addition, compressing 1-D vertical differentials gives the best result for karyotype images. This is because the chromosomes have been rotated and vertically oriented. For spread images, however, compression of 1-D horizontal or vertical differentials gives very close results because the chromosomes are randomly oriented. Based on these observations, we conclude that it is best to use six levels of wavelet decomposition with the 5/3 filters in modified SPIHT to code 1-D vertical differentials for lossless compression of chromosome ROIs. We will use this setting in our cascaded differential and wavelet coder in the following for comparison with other lossless coders.

3) Comparison With Other Lossless Coding Techniques: We compare our proposed coding scheme with LZW coding and JPEG-2000. Because the LZW coder built in the TIFF format and the JPEG-2000 coder only handle lossless compression of regular rectangular images; whereas there exists no such constraint for the LZW coder in WinZip and in our proposed coder, two sets of experiments are conducted. The first set of experiments compares LZW (in TIFF and WinZip), JPEG-2000, and modified SPIHT coding of regular rectangular spread images (with original or flat background) and karyotype images; the second ones match LZW coding against cascaded differential and wavelet coding of chromosome ROIs. We pick the latest version WinZip 8.0 for testing LZW coding and use Taubman's Kakadu V2.2 implementation, which is fully compliant with the standard, for JPEG-2000 coding.

Results from the first set of experiments are given in Table I(a). The bit rate range (in bits per pixel) is reported for each coding scenario, with the mean included in parentheses. LZW coding in WinZip 8.0 uniformly outperforms the older version used in the TIFF format. While JPEG-2000 and modified SPIHT offer comparable compression performance, both work better than LZW coding in TIFF. Between the modified SPIHT and LZW coding in WinZip 8.0, the former gives better lossless compression performance for spread images with original background; both perform comparably for spread and karyotype images with flat background. Finally, we see that karyotype images with flat background are easier to compress than spread images with flat background.

Results from the second set of experiments are summarized in Table I(b), and plotted in Fig. 13. For fair comparisons, LZW coding is used here to compress the file consisting of pixels only in the chromosome ROIs, and the rate spent in chain codes for specifying chromosome boundaries (about 6%–10% of the total rate) is included in the final rate. Although we only code pixels

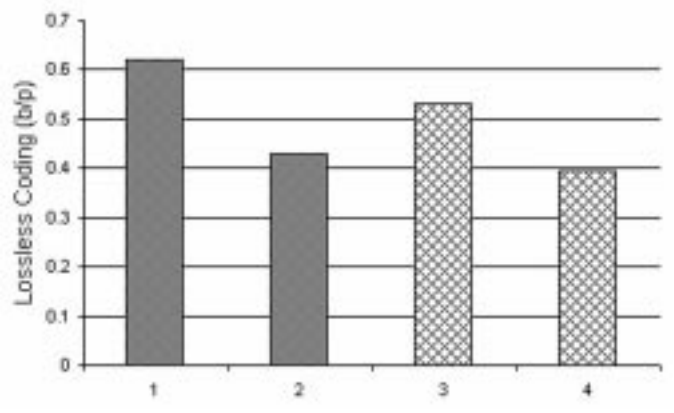


Fig. 13. Averaged results from lossless compression of the chromosome ROIs, including the chromosome boundary chain codes. The compression methods and the ROI types are: 1: WinZip 8.0 on the spread ROIs; 2: The proposed method on the spread ROIs; 3: WinZip 8.0 on the karyotype ROIs; and 4: The proposed method on the karyotype ROIs.

in ROIs in both coders, the reported bit rate in b/p is computed with respect to the original image size (764×560) for easy comparison with results in Table I(a).

Results in Table I(b) demonstrate that our cascaded differential and wavelet coder outperforms the LZW coder in WinZip 8.0 by 31% for coding chromosome ROIs in spread images and 24% for coding chromosome ROIs in karyotype images, respectively.

From results in both tables, we clearly see that our proposed coder is by far the best for lossless compression of chromosome ROIs. Compared with storing chromosome images with flat background in TIFF format with LZW coding, our proposed scheme offers 54% and 55% savings in losslessly archiving chromosome ROIs in spread and karyotype images, respectively. In other words, our new method more than doubles the amount of compression achieved in current commercial karyotyping systems. This demonstrates the power of our proposed approach, in which we can take pixels in chromosome ROIs only and compress them efficiently.

Before concluding this subsection, we point out that, although we have so far focused exclusively on lossless compression, our proposed coder also allows lossy compression of chromosome ROIs. This can be done simply by decoding a truncated version of the losslessly compressed bit stream. The lossy mode is useful in applications like progressive image transmission in cytogenetic telemedicine and fast searching and browsing of chromosome spread and karyotype images.

B. Lossy-to-Lossless Coding of Chromosome Background Images

Only chromosome spread images are used in these tests. This is because the background in a karyotype image is generally constant-valued and, therefore, requires no coding. We assume that the chromosome ROIs and boundaries are already losslessly coded by using our cascaded differential and wavelet coder. Lossy-to-lossless compression is achieved by directly applying the modified SPIHT coder on background images using the 5/3 or the 9/7 filters and six levels of integer wavelet decomposition.

TABLE II
LOSSLESS COMPRESSION RESULTS FOR TEN CHROMOSOME BACKGROUND IMAGES THE BIT RATE RANGE (b/p) IS REPORTED FOR EACH CODING SCENARIO, WITH THE MEAN INCLUDED IN PARENTHESES

	5/3 filters	9/7 filters
Chromosome background	2.1072-2.3887 (2.2013)	2.4141-2.6399 (2.4982)

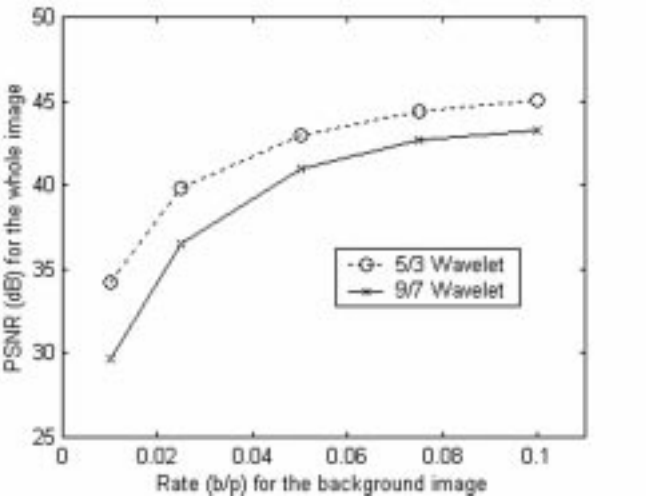


Fig. 14. PSNR performance (in dB) of the modified SPIHT coder on lossy coding of chromosome background images. The PSNR numbers are computed with respect to the whole image, in which there is no loss in chromosome ROIs. The rates are for the chromosome background images only.

Lossless compression results for ten chromosome background images are summarized in Table II. The 5/3 filters again outperform the 9/7 filters.

Lossy coding of a chromosome background image can be realized by encoding/decoding at any rate lower than the lossless rate needed for that image. Fig. 14 depicts the PSNR performance averaged over ten different background images at five different rates (0.010, 0.025, 0.050, 0.075, 0.100 b/p). These PSNR numbers are computed with respect to the whole image (764×560), in which there is no loss in chromosome ROIs. The reported rates are for the chromosome background images only. The overall bit rate will be the sum of the rate for lossy coding of the background image plus the rate [0.4298 b/p on average as seen from Table I(b)] for lossless coding of the chromosome ROIs.

Fig. 15 displays the original and reconstructed versions of "144 461ASU" at different rates for the background. Chromosome ROIs in this image are losslessly coded at a rate of 0.4166 b/p using the cascaded differential and wavelet coder. At 0.01 b/p, the background blur is noticeable. At 0.025 b/p, it looks better but is still a little blurry. At 0.05 b/p, we have to compare the reconstructed image with the original to tell the difference. Finally, at 0.075 b/p or 0.1 b/p, the difference between the two images becomes imperceptible.

Because a background image only contains scattered cell nuclei and stain debris, even at very low bit rate (0.050 to 0.100 b/p), the reconstructed image is of very high quality (with over

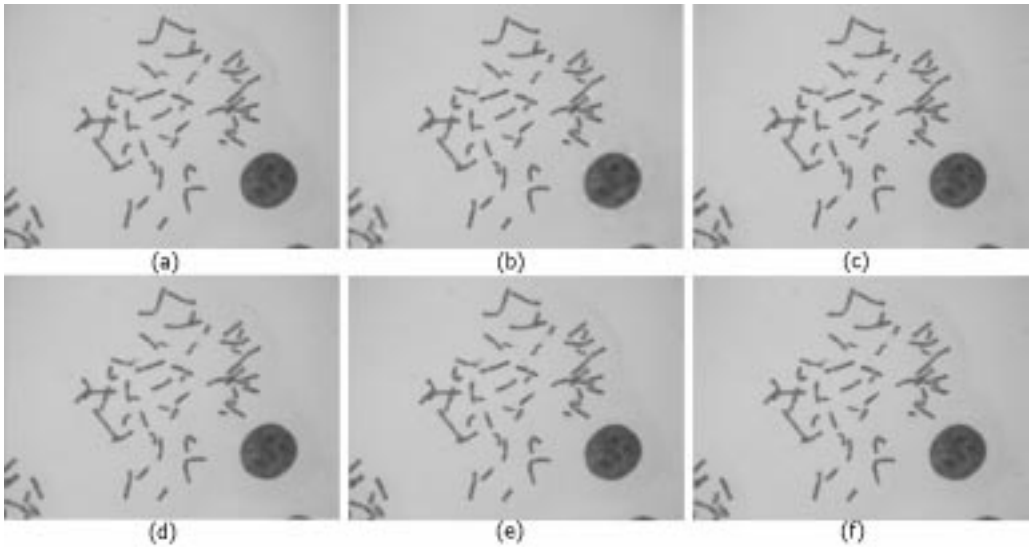


Fig. 15. Different versions of the spread image "144 461ASU". (a) The original. (b) The background reconstructed at 0.01 b/p using our proposed coding method (37.322 dB) (c) The background reconstructed at 0.025 b/p (41.544 dB). (d) The background reconstructed at 0.05 b/p (43.878 dB). (e) The background reconstructed at 0.075 b/p (44.785 dB). (f) The background reconstructed at 0.1 b/p (45.314 dB).

40 dB in PSNR). It is, thus, not necessary to insist on lossless compression of these background images, which requires a much higher rate (2.2013 b/p on average).

V. Conclusion

We have proposed a new coding scheme for efficient compression of digital chromosome images. Our approach seeks to exploit the knowledge that, for most spread and karyotype images, the chromosome ROIs, containing all necessary diagnostic information, are well defined and segmented before the images are stored. Chromosome boundaries are hence readily available to both the encoder and decoder in terms of compactly represented boundary chain codes. In contrast, image compression techniques (e.g., LZW coding) currently used in commercial karyotyping systems code the entire image as a whole, wasting valuable chromosome ROI segmentation information while retaining much of the unwanted redundancy. The new JPEG-2000 standard, unfortunately, does not provide the desirable support for lossless compression of arbitrary ROIs.

We can losslessly compress the chromosome ROIs, while lossily compressing the background and reconstructing the image with imperceptible difference, both at low bit rates. Our results are significantly better than those from LZW coding of chromosome ROIs. With the modified SPIHT coder offering state-of-the-art wavelet image coding performance at lower complexity than JPEG-2000's, we have achieved beyond what JPEG-2000 can offer in terms of lossless chromosome ROI coding.

Another advantage of the proposed approach is that our coder generates embedded bit streams that allow progressive image coding. This is very attractive for emerging applications like progressive image transmission in cytogenetic telemedicine and fast searching and browsing of chromosome and karyotype images.

References

[1] A. Said and W. Pearlman, "A new, fast, and efficient image codec based on set partitioning in hierarchical trees," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, pp. 243–250, June 1996.

[2] J. Graham and J. Piper, "Automatic karyotype analysis," *Meth. Mol. Biol.*, vol. 29, pp. 141–185, 1994.

[3] "An international system for human cytogenetic nomenclature," *Cytogenet. Cell Genetics*, 1985.

[4] J. Ziv and A. Lempel, "Coding theorems for individual sequences via variable-rate coding," *IEEE Trans. Inform. Theory*, vol. 24, pp. 530–536, May 1978.

[5] T. Welch, "A technique for high-performance data compression," *IEEE Comput. Mag.*, vol. 17, pp. 8–19, June 1984.

[6] D. Taubman and M. Marcellin, *JPEG2000: Image Compression Fundamentals, Standards, and Practice*. Norwell, MA: Kluwer, 2001.

[7] Q. Wu, Z. Xiong, Y. Wang, and K. Castleman, "Wavelet-based lossy-to-lossless coding of cytogenetic images with arbitrary regions of support," in *Proc. ISPACS'00*, Honolulu, HI, Nov. 2000, pp. 72–75.

[8] E. Atsumi and N. Farvardin, "Lossy/lossless region-of-interest image coding based on set partitioning in hierarchical trees," in *Proc. ICIP'98*, vol. 1, Chicago, IL, Oct. 1998, pp. 87–91.

[9] K. Park and H. Park, "Region-of-interest coding based on set partitioning in hierarchical trees (SPIHT)," *IEEE Trans. Circuits Syst. Video Technol.*, submitted for publication.

[10] M. Penedo, W. Pearlman, P. Tahoces, M. Souto, and J. Vidal, "Region-based wavelet coding methods for digital mammography," *IEEE Trans. Med. Imag.*, submitted for publication.

[11] *IEEE Trans. Circuits Syst. Video Technol. (Special Issue on Object-Based Video Coding and Description)*, vol. 9, Dec. 1999.

[12] G. Minami, Z. Xiong, A. Wang, and S. Mehrotra, "3-D wavelet coding of video with arbitrary regions of support," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, pp. 1063–1068, Sept. 2001.

[13] I. Daubechies, *Ten Lectures on Wavelets*. Philadelphia, PA: SIAM, 1992.

[14] S. Mallat, *A Wavelet Tour of Signal Processing*. New York: Academic, 1997.

[15] J. Shapiro, "Embedded image coding using zero trees of wavelet coefficients," *IEEE Trans. Signal Processing*, vol. 41, pp. 3445–3463, Dec. 1993.

[16] W. Pennebaker and J. Mitchell, *JPEG: Still Image Data Compression Standard*. New York: Van Nostrand Reinhold, 1992.

[17] Z. Xiong and K. Ramchandran, "Wavelet image compression," in *Handbook of Image and Video Processing*, A. Bovik, Ed. New York: Academic, 2000.

[18] J. Woods, Ed., *Subband Image Coding*. Boston, MA: Kluwer Academic, 1991.

[19] (1998) Wavelet Image Coding: PSNR Results. Univ. California Los Angeles (UCLA) Image Commun. Lab.. [Online]. Available: http://www.icsl.ucla.edu/~pl/psnr_results.html

[20] D. Taubman, "High performance scalable image compression with EBCOT," *IEEE Trans. Image Processing*, vol. 9, pp. 1158–1170, July 2000.
[21] I. Witten, R. Neal, and J. Cleary, "Arithmetic coding for data compression," *Commun. ACM*, vol. 30, pp. 520–540, June 1987.
[22] K. Castleman, *Digital Image Processing*. Englewood Cliffs, NJ: Prentice Hall, 1996.
[23] N. Jayant and P. Noll, *Digital Coding of Waveforms*. Englewood Cliffs, NJ: Prentice-Hall, 1984.
[24] J. Villasenor, B. Belzer, and J. Liao, "Wavelet filter evaluation for image compression," *IEEE Trans. Image Processing*, vol. 4, pp. 1053–1060, Aug. 1995.
[25] M. Adams and F. Kossentini, "Reversible integer-to-integer wavelet transforms for images compression: Performance evaluation and analysis," *IEEE Trans. Image Processing*, vol. 9, pp. 1010–1024, June 2000.
[26] A. Calderbank, I. Daubechies, W. Sweldens, and B. Yeo, "Wavelet transforms that map integers to integers," *J. Appl. Comput. Harmon. Anal.*, vol. 5, pp. 332–369, July 1998.
[27] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image coding using wavelet transform," *IEEE Trans. Image Processing*, vol. 1, pp. 205–221, Apr. 1992.
[28] I. Daubechies and W. Sweldens, "Factoring wavelet transforms into lifting steps," *J. Fourier Anal. Applicat.*, vol. 4, pp. 247–269, 1997.
[29] W. Sweldens and P. Schroder, "Building your own wavelets at home," *ACM SIGGRAPH Course Notes on Wavelets in Computer Graphics*, pp. 15–87, 1996.
[30] J. Apostolopoulos and J. Lim, "Critically sampled wavelet representations for multidimensional signals with arbitrary regions of support," in *Proc. ICASSP'98*, vol. 3, Seattle, WA, May 1998, pp. 1507–1512.
[31] J. Kim, J. Lee, E. Kang, and S. Ko, "Region-based wavelet transform for image compression," *IEEE Trans. Circuits Syst.*, vol. 45, pp. 1137–1140, Aug. 1998.
[32] J. Li and S. Lei, "Arbitrary shape wavelet transform with phase alignment," in *Proc. ICIP'98*, vol. 3, Chicago, IL, Oct. 1998, pp. 683–687.
[33] S. Li and W. Li, "Shape-adaptive discrete wavelet transforms for arbitrarily shaped visual object coding," *IEEE Trans. Circuits Syst. Video Technol*, vol. 10, pp. 725–743, Aug. 2000.
[34] Q. Wu, J. Snellings, L. Armory, P. Suetens, and A. Oosterlinck, "Model-based contour analysis in a chromosome segmentation system," in *Automation of Cytogenetics*, C. Lundsteen and J. Piper, Eds. Berlin, Germany: Springer-Verlag, 1989.
[35] L. Ji, "Fully automatic chromosome segmentation," *Cytometry*, vol. 17, pp. 196–208, 1994.

Zhongmin Liu (S'99) received the B.S. degree in engineering physics and M.S. degree in reactor engineering from Tsinghua University, Tsinghua, China, in 1992 and 1994, respectively. He is currently a Ph.D. degree candidate in the Department of Electrical Engineering, Texas A&M University, College Station, since 1999.

From 1994 to 1996, he was with the Precision Instrument Department, Tsinghua University, as a Research Assistant. From 1996 to 1998, he was with Hewlett-Packard China, as an R&D engineer. His research interests include digital image processing, wavelet coding, joint source-channel coding and pattern recognition.

Zixiang Xiong (S'91–M'92) received the Ph.D. degree in electrical engineering from the University of Illinois at Urbana-Champaign in 1996. From 1995 to 1997, he was with the Department of Electrical Engineering, Princeton University, Princeton, NJ, as a visiting student and then a post-doctoral fellow.

During 1995 to 1997 he was also with Sarnoff Corporation. From 1997 to 1999, he was with the University of Hawaii, Honolulu. He joined the Department of Electrical Engineering at Texas A&M University, College Station, as an Assistant Professor in 1999. He spent the summers of 1998 and 1999 at Microsoft Research, Redmond, WA, and the summers of 2000 and 2001 at Microsoft Research, China in Beijing. His research interests are data compression, Internet/wireless multimedia and image processing.

Dr. Xiong received a NSF Career Award in 1999, an ARO Young Investigator Award in 2000, and an ONR Young Investigator Award in 2001. He is currently an Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY and an Eugene Webb Faculty Fellow at Texas A&M University.

Qiang Wu (S'89–M'90) received the Ph.D. degree in electrical engineering in 1991 from the Catholic University of Leuven (Katholieke Universiteit Leuven), Leuven, Belgium.

In December 1991, he joined Perceptive Scientific Instruments, Inc., Houston, Texas, where he was a Senior Software Engineer from 1991 to 1994, and was a Senior Research Engineer from 1995 to 2000. He was a key contributor to research and development of the core technology for PSI's PowerGene cytogenetics automation products, including digital microscope imaging, image segmentation, enhancement, recognition, compression for automated chromosome karyotyping, and FISH image analysis for automated assessment of low-dose radiation damage to astronauts during and after NASA space missions. He is currently a Lead Research Engineer at Advanced Digital Imaging Research, LLC., Houston, Texas. He has served as Principal Investigator of five National Institutes of Health (NIH) SBIR grants since 1995. His research interests include pattern recognition, image processing, artificial intelligence, and their biomedical applications.

Yu-Ping Wang (M'00) received the Ph.D. degree in communication and electrical system in 1996 and the M.Sc. degree in computational mathematics in 1993 both from Xi'an Jiaotong University, China. He received the B.Sc. in applied mathematics in from Tianjin University, Tianjin, China, in 1990.

During 1996 to 2000, he did research work at the National University of Singapore, Singapore, Washington University Medical School, St. Louis, MO, and Texas A&M University, College Station. He is a Senior Research Engineer with Advanced Digital Imaging Research, LLC, League City, TX. His research interests include computer vision, signal processing, and applied mathematics as well as their particular applications to medical imaging.

Kenneth Castleman (S'69–M'69) received the B.S., M.S., and Ph.D. degrees in electrical engineering from The University of Texas at Austin in 1965, 1967, and 1969, respectively.

From 1970–1985, he was a Senior Scientist at NASA's Jet Propulsion Laboratory, Pasadena, CA, where he developed digital imaging techniques for a variety of medical applications. During that period he also served as a Lecturer at the California Institute of Technology (Caltech), Pasadena, and a Research Fellow at the University of Southern California (USC), Los Angeles, and at the University of California, Los Angeles (UCLA).

In 1984, he founded Perceptive Systems, Inc (PSI), a company that developed imaging instruments for cytogenetics, introducing many innovations into the field. He is the author of more than 60 articles in the scientific literature, two textbooks on digital imaging, and three patents. He has served as Principal Investigator on more than a dozen government-sponsored research grants. He has served on advisory boards at the National Institutes of Health, The University of Texas, Carnegie-Mellon University, and the FBI. He is now the president of Advanced Digital Imaging Research, LLC (ADIR) in Houston, Texas.

In 1994, Dr. Castleman was inducted into the Space Foundation's Space Technology Hall of Fame. He is a fellow of the American Institute for Medical and Biological Engineering. He is also a past chairman of the IEEE student branch at The University of Texas at Austin.