Hierarchical Lossy Bilevel Image Compression Based on Cutset Sampling

Shengxin Zha, *Member, IEEE,* Thrasyvoulos N. Pappas, *Fellow, IEEE,* and David L. Neuhoff, *Fellow, IEEE*

Abstract—We consider lossy compression of a broad class of bilevel images that satisfy the smoothness criterion, namely, images in which the black and white regions are separated by smooth or piecewise smooth boundaries, and especially lossy compression of complex bilevel images in this class. We propose a new hierarchical compression approach that extends the previously proposed fixed-grid lossy cutset coding (LCC) technique by adapting the grid size to local image detail. LCC was claimed to have the best rate-distortion performance of any lossy compression technique in the given image class, but cannot take advantage of detail variations across an image. The key advantages of the hierarchical LCC (HLCC) is that, by adapting to local detail, it provides constant quality controlled by a single parameter (distortion threshold), independent of image content, and better overall visual quality and rate-distortion performance, over a wider range of bitrates. We also introduce several other enhancements of LCC that improve reconstruction accuracy and perceptual quality. These include the use of multiple connection bits that provide structural information by specifying which black (or white) runs on the boundary of a block must be connected, a boundary presmoothing step, stricter connectivity constraints, and more elaborate probability estimation for arithmetic coding. We also propose a progressive variation that refines the image reconstruction as more bits are transmitted, with very small additional overhead. Experimental results with a wide variety of, and especially complex, bilevel images in the given class confirm that the proposed techniques provide substantially better visual quality and rate-distortion performance than existing lossy bilevel compression techniques, at bitrates lower than lossless compression with the JBIG or JBIG2 standards.

Index Terms—Lossy bilevel image coding, cutset sampling,constant quality, JBIG2, Markov random fields (MRFs).

I. INTRODUCTION

Compression of bilevel images, i.e., images whose pixels are either black or white, is an important special case of image compression, both because some images are bilevel (text, halftones, graphics, line drawings, pen and ink sketches, logos, computer icons, silhouettes, etc.) and because bilevel images can be used to describe segment boundaries (e.g., foreground/background) in a variety of applications, including object-based compression (MPEG-4 [2] and second generation image coding techniques [3]). This paper proposes *lossy* compression techniques for a broad class of bilevel images, in which the black and white regions are separated by smooth, or piecewise smooth boundaries. Most bilevel images of the types listed above fall in this category, especially when the resolution is high, except halftones.

The JBIG standard [4], [5] provides an efficient method for *lossless* compression of all types of bilevel images, including some halftones generated with periodic screens (with small period), but not stochastic halftones like error diffusion and blue-noise screening (long period). However, the resulting bit rates are still relatively high. Thus, there is a need for lossy techniques to make substantial further reductions in coding rate, provided satisfactory perceptual quality can be maintained.

The JBIG2 standard [6], [7] is an evolution of JBIG that, in addition to lossless bilevel compression, aims to accomplish lossy bilevel compression at low bitrates with minimal degradation in image quality. JBIG2 relies on partitioning the bilevel image into regions (of text, graphs, and halftones) and encoding each region with a different scheme. However, the current stage of the standard can only handle lossy compression of text and halftones. Thus, there is a need for lossy compression of bilevel images that do not fall in these two categories. There is also a substantial literature on lossy compression techniques for coding object contours (e.g., for MPEG-4) and line drawings [8]–[12]. However, the applicability of such techniques is quite limited, because they assume not only smooth but well-defined, isolated, binary objects.

As mentioned above, the techniques we consider here apply to a broad class of images in which the black and white regions are separated by smooth or piecewise smooth boundaries. We will refer to this as the *smoothness criterion*. While, with the exception of halftones, most bilevel images fall in this category, most of the benefits apply to *complex* bilevel images that satisfy the smoothness criterion. We are aware of only two techniques that work reasonably well on the broader class of bilevel images that satisfy the smoothness criterion, the lossy cutset coding (LCC) approach by Reyes *et al.* [13], [14], the extension of which is the objective of this paper, and a finite automata approach by Culik and Valenta [15], [16].

The key idea of LCC [13], [14] is to losslessly encode all the pixels on a fixed rectangular grid (which as we will see forms a *cutset*), and to rely on the decoder to reconstruct the interiors of blocks formed by the grid based on a Markov random

field (MRF) model. The MRF model provides constraints for reconstructing the smoothest image that is consistent with the samples on the cutset. Reyes *et al.* [13], [14] showed that LCC provides better visual quality and rate-distortion performance than the Culik and Valenta techniques [15], [16], at bitrates lower than lossless compression with the JBIG standard. However, the problem with the fixed-grid approach [13], [14] is that the amount of detail can vary within an image and from image to image. In the first case, there is no fixed grid size that works well over the entire image, and in the latter, there is no grid size that works well for multiple images, that is, the grid size would have to be hand-tuned for each image. Thus, the fixed-grid approach is not a complete, practical algorithm.

We propose a *hierarchical LCC (HLCC)* approach that adapts the grid size to local image detail, thus forming a quadtree (c.f. [17]). While this requires additional bits for encoding the quadtree and the additional grid contours, we show that it provides a better overall perceptual quality and rate-distortion performance. The key idea is to use a coarser grid in smooth areas of the image and a finer grid in areas with more detail. Given a desired error rate and an initial (typically large) grid size, the grid size is locally adjusted by splitting to meet the target error rate. This results in a fixed quality coder and better rate-distortion performance, the two key advantages of HLCC. The bitrate depends on image content. In contrast, in the fixed-grid approach, both the error rate and the bitrate depend on the local image detail, and are only indirectly controlled by the grid size. As we will see, the hierarchical approach preserves key structures of the image and adds details as the splitting threshold decreases.

We also introduce several extensions of LCC, which improve reconstruction accuracy as well as perceptual quality, especially for large grid sizes, thus extending the range of bitrates that the technique can handle without breaking down. The first is to add one or more *connection* bits, which provide structural information by specifying which black or white runs on the boundary of a block must be connected. A second extension is a grid presmoothing step, which is intended to align the smoothness of the block boundary with that of the interior. The idea is to flip short runs of black or white pixels on the cutset, if it does not increase the reconstruction error in the block interior. While flipping such runs increases the reconstruction error on the boundary, it improves perceptual quality by eliminating artifacts caused by isolated runs on the cutset. In fact, it may even decrease the reconstruction error in the interior. In addition, it results in a slight decrease in the bitrate. Thus, the rate-distortion performance improves both perceptually and according to error rate.

In addition, we propose a progressive encoding scheme, whereby we first encode the entire image using a coarse grid, which results in a relatively smooth reconstruction, and then progressively refine the grid to add details as more bits are transmitted. That is, the encoding process can be terminated at any block size at the cost of adding a very small amount of bitrate overhead over the hierarchical approach.

Finally, for both the hierarchical and progressive schemes, we use arithmetic coding to encode each pixel on the cutset (conditioned on a previously encoded pixel) and the side information (connection and quadtree splitting bits). The decoded grid pixels and side information are then used to reconstruct each block from its boundary based on the MRF model. We obtain accurate estimates of the encoding rate via the empirical first-order conditional entropy of the bits to be encoded.

Experimental results with simple and complex bilevel images that satisfy the smoothness criterion demonstrate that the hierarchical approach with the proposed extensions yields lower error rate and better perceptual quality than the fixed-grid approach at comparable or even lower bitrates. This is because it adapts to local detail for efficient distribution of bits around the image. While for some images and bitrates the rate-distortion performance gains compared to the fixed-grid approach are modest, it provides constant quality, controlled by the splitting threshold, within and across images. This ensures that perceptually important details are well and efficiently encoded, even when they occupy only a small fraction of the image; as such, they have little effect on the error rate but substantial effect on overall perceptual quality. In addition, as the bitrate decreases, the quality decreases gracefully, preserving the most important information. Moreover, with this added flexibility, our results bring out and contribute to a better understanding of the capabilities and limitations of cutset coding.

We also compare the performance of the proposed approach to the Culik and Valenta techniques [15], [16], and show that the former results in significantly better perceptual quality for comparable bitrates. Compared to the lossless JBIG and JBIG2 techniques, the proposed approach offers reasonable quality over a range of bitrates, including structurally lossless quality [18] (encoded images of quality comparable to the original even though there may be noticeable differences in a side-by-side comparison) at rates lower than JBIG and JBIG2. On the other hand, at high bitrates, JBIG and JBIG2 outperform the proposed technique.

The paper is organized as follows. In section II, we review the LCC approach. Section III describes two extensions of LCC: extra connection bits and presmoothing. Section IV introduces the proposed hierarchical LCC approach. Section VI presents a progressive hierarchical approach. Section V describes the lossless encoding specifics. Section VII presents the experimental results and Section VIII summarizes the conclusions.

II. REVIEW OF FIXED-GRID LCC

In this section we review the LCC approach of Reyes *et al.* [13], [14] and describe the particular decoding strategy adopted in this paper. LCC encodes the pixels on a fixed rectangular grid using a lossless compression technique, such as arithmetic or Huffman coding. The rectangular grid subdivides the image into (typically square) blocks, and provides a specification for the reconstruction by the decoder. The decoder then relies on an MRF model to reconstruct the pixels in the block interiors, as the maximum *a posteriori* probability (MAP) estimates, given the values of the pixels on the block boundary.

(a) Original   (b) Specification   (c) Reconstruction

Fig. 1: Cutset Specification and Reconstruction

(b) 2nd order neighborhood

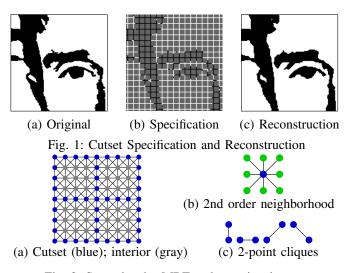(a) Cutset (blue); interior (gray)   (c) 2-point cliques

Fig. 2: Second-order MRF and associated cutset

The MRF model provides constraints for reconstructing the smoothest image that is consistent with the samples on the block boundaries. An example is shown in Figure 1. For a first- or second-order MRF [14], the grid forms a cutset, in the sense that the pixels in the block interior are independent of the pixels in the interiors of the neighboring blocks [13], [14].

If the bilevel image satisfies the smoothness criterion and if the spacing of the grid lines is not too big, it can be argued that the reconstructed image preserves the structure of the original image. One can then argue that this is an example of *structurally lossless* compression [18], [19] for bilevel images, whereby the original and reconstructed images are similar and both of high quality, even though in a side-by-side comparison there may be visible differences. We now look at the MRF-based reconstruction in more detail.

A. Detailed Markov Random Field Formulation

Let the image be sampled on an $N \times N$ grid, that is, on columns (rows) 0, N, $2N$, etc. Thus, the size of the block interiors is $(N - 1) \times (N - 1)$ pixels, and the size of the blocks including the boundaries is $(N + 1) \times (N + 1)$ pixels.

The MRF in [13], [14] is specified by a graph $G = (V, E)$ and a collection of clique potential functions $\psi_c(x)$. V is a set of nodes, which specifies the pixel locations of the image, and edges in E define a neighborhood relation between nodes. Figure 2 shows a cutset associated with a second-order MRF, in which each node (pixel) is associated with a second-order neighborhood. A clique c is a subset of nodes in V that are neighbors of each other, that is, they are connected by an edge. In this paper, we consider a second-order MRF and only the two-point cliques (horizontal, vertical, and diagonal) shown in Figure 2(c). Let C denote the collection of all two-point cliques in G. Now consider a bilevel image $\mathbf{x}$ that assigns a 0 (white) or 1 (black) to each node in V. For each $c \in C$, a clique potential function $\psi_c(\mathbf{x})$ takes a value that depends only on the nodes in c. For the two-point cliques, we define the potential functions as follows:

$$\psi_c(\mathbf{x}) = \begin{cases} -\beta, & x_s = x_q, (s, q) \in c, \beta > 0 \\ +\beta, & x_s \neq x_q, (s, q) \in c, \beta > 0 \end{cases} \quad (1)$$

When one pixel is black and the other is white, we will refer to the clique as an *odd bond*, and when the pixels are the same, we will refer to the clique as an *even bond*. The probability density function of an image realization $\mathbf{x}$ is given by the *Gibbs* density

$$p(\mathbf{x}) = \frac{1}{z} \exp\{-\sum_{c \in C} \psi_c(\mathbf{x})\} \quad (2)$$

where z is a normalizing constant. It can also be expressed in terms of the odd and even bonds as follows:

$$p(\mathbf{x}) = \frac{1}{z} \exp\{-\beta t(\mathbf{x}) + \beta(|E| - t(\mathbf{x}))\} \quad (3)$$

where E is the total number of edges in all cliques and $t(\mathbf{x})$ is the total number of odd bonds.

A key property of an MRF is that the probability of a realization over a subset of nodes, given the values of the neighbors of the subset, is independent of the values of the rest of the nodes [13], [14]. Thus, as we mentioned above, for the given second-order MRF model, given the pixels on the boundary, the pixels in the block interior are independent of the pixels in the interiors of the neighboring blocks. It is this property that makes the rectangular grid of pixels a *cutset*.

The MAP estimate of the interior pixels of a block, given the values of the boundary pixels, reduces to finding the bilevel block interior that in combination with the boundary has the fewest black-white transitions between pixels (odd bonds) [13], [14], [20]. Reyes *et al.* [13], [14] found explicit rules for optimal reconstructions for boundary specifications that contain: (a) all black (or white) pixels; (b) one run of black (and one run of white) pixels; and (c) two runs of black (and two runs of white) pixels. In all of these cases, the optimal reconstruction consists of monochrome regions bounded by minimally varying (smooth is hard to define for pixelated lines) *nonintersecting* lines connecting the endpoints of the boundary runs. In [14] they called such lines *simple paths*. A path is a sequence of edges connecting two pixels. A simple path consists of edges that are either all in the same direction (horizontal, vertical, diagonal, or antidiagonal) or all in two directions, one horizontal or vertical, and the other diagonal or antidiagonal. They also showed that, depending on the endpoint locations, there may be a number of optimal reconstructions, in which case, the decoder picks one at random.

Examples of optimal reconstructions for 1-run and 2-run blocks using simple paths are shown in Figure 3. In the 1-run case, all simple paths connect the end points of the run, resulting in two monochrome regions. In the case of two runs, there are two possibilities: simple paths connect the endpoints of each of the white runs or they connect the end points of each of the black runs, as shown in Figure 3 (b) and (c), respectively. In the former case, this results in one connected black region separating two white regions, and the converse in the latter. An optimal reconstruction can be found in each case. While only one of these choices provides the overall optimum in the MAP sense, Reyes *et al.* [13], [14] found it beneficial for the encoder to signify the one that best represents the original image via a *connection bit*. This is decided by comparing the two reconstructions to the actual block interior, and selecting

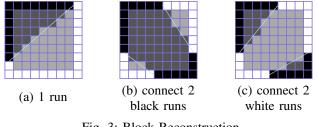(a) 1 run   (b) connect 2 black runs   (c) connect 2 white runs

Fig. 3: Block Reconstruction

the one that results in the lower error rate. While the error rate does not correlate well with image quality when comparing different compression approaches (e.g., LCC [13] with [16]), Reyes *et al.* [21] found that, compared to other criteria for the run connection decision, the error rate provides the most robust and perceptually meaningful results. Reyes *et al.* [13], [14] showed that the connection bit is critical for preserving structural information, and results in dramatic improvements in rate-distortion performance.

For boundaries with three or more runs, the reconstruction was based on the two longest runs of one color [13], [14]. They used the following ad hoc rules: (i) change all but the two longest black runs to white and reconstruct by connecting the white runs; (ii) change all but the two longest white runs to black and reconstruct by connecting the black runs; and (iii) choose between the results of (i) and (ii) based on the error, and indicate the decision by a connection bit. They argued that for relatively small grid sizes ($N = 8$), the occurrence of more than two black (white) runs is rare, and when it does happen, the ad hoc approximation does not significantly affect the resulting error rate or visual quality. In summary, all that LCC needs to transmit are the pixels on the grid and the connection bits (for blocks with two or more runs).

In this paper, we base all reconstructions on digital straight lines using Bresenham's approximation [22]. Note that such lines are simple paths (optimal) and, in addition to eliminating the ambiguity of multiple optimal solutions, they eliminate any biases towards black or white reconstructions. Examples are shown in Figure 3.

III. EXTENSIONS OF FIXED-GRID LCC

In this section we introduce a number of improvements of LCC [13], [14] that enhance its performance.

A. 4-connectivity of Reconstructed Regions

As we discussed in Section II, the MRF formulation was intended to impose constraints for reconstructing the smoothest image that is consistent with the samples on the block boundaries. However, while the optimal reconstruction rules are well defined, they can lead to some reconstructions that cannot be characterized as smooth. For example, the first row of Figure 4 shows the Bresenham line reconstructions for the boundary specifications when the connection bit indicates that the black runs should be connected. Note that the black runs are 8-connected [23, ch11]; these reconstructions correspond to the fewest odd bonds for connected black runs. Interestingly, however, the white runs are also 8-connected. To avoid such cases and to enhance the smoothness and visual continuity of the connections, we will assume that the connection bit

8-connected black runs
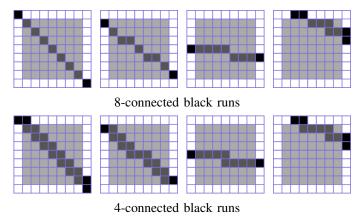

4-connected black runs

Fig. 4: Connectivity examples: connecting black runs

enforces 4-connectivity between black or white runs. The second row of Figure 4 shows examples of 4-connected black runs. In this case the white runs are not connected. Note that two one-pixel runs on opposite corners of the block cannot be 4-connected; so we added two pixels in the first example of the second row to enable a 4-connected reconstruction. Assuming 4-connectivity of the black runs, the reconstruction is optimal as it follows the simple path rule of [14]. Overall, imposing 4-connectivity enhances the visual continuity of the connections. The error rate of reconstructions stays essentially the same. To enforce the 4-connectivity, we use the 4-connected Bresenham line implementation in the *OpenCV* library [24].
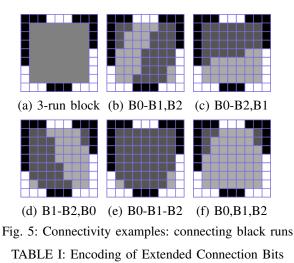
B. Extended Connection Bits

As we discussed in Section II, Reyes *et al.* [13], [14] argued that, for relatively small grid sizes, blocks with three or more runs are rare and do not affect the quality of the results. When the grid size is large (e.g., $N = 16$), however, such blocks are more common and reducing them to two runs introduces significant reconstruction errors. In this section, we introduce additional connection bits and associated reconstruction rules to handle up to four runs in a block (four black and four white runs). We will refer to the use of additional connection bits as *extended* connection bits.

The reconstruction rules for boundaries with two or fewer black (and white) runs remain unchanged. The rules for blocks with boundaries containing three or more black (and white) runs are modified as follows. In a 3-run block, there are five possibilities for connecting the runs: (1-3) three ways of connecting two black runs and *self-connecting* the other black run, i.e., connecting the two end-points of the run, as in the 1-run case; (4) connecting all black runs; and (5) self-connecting each black run. Specifically, if all the black runs on the boundary are labeled clockwise from the top right as $B0, B1, B2$, then as shown in Figure 5 the different choices are:

- B0-B1, B2 (connect B0 with B1, self-connect B2)
- B0-B2, B1 (connect B0 with B2, self-connect B1)
- B1-B2, B0 (connect B1 with B2, self-connect B0)
- B0-B1-B2 (connect B0, B1, B2)
- B0, B1, B2 (self-connect B0, B1, B2)

In a 4-run block, there are 14 possibilities for connecting the runs: (1-6) six ways of connecting two black runs and

(a) 3-run block (b) B0-B1,B2 (c) B0-B2,B1

(d) B1-B2,B0 (e) B0-B1-B2 (f) B0,B1,B2

Fig. 5: Connectivity examples: connecting black runs

TABLE I: Encoding of Extended Connection Bits

number of black (white) runs	0	1	2	3	≥ 4
number of connection choices	1	1	2	5	14
binary code length	0	0	1	3	4

self-connecting the other two black runs; (7-8) two ways of connecting pairs of adjacent black runs; (9-12) four ways of connecting three black runs and self-connecting the other black run; (13) connecting all black runs; and (14) self-connecting each black run. Specifically, if all the black runs on the boundary are labeled clockwise from the top right as $B0, B1, B2, B3$, then the different choices are:

- B0-B1, B2, B3
- B0-B2, B1, B3
- B0-B3, B1, B2
- B1-B2, B0, B3
- B1-B3, B0, B2
- B2-B3, B0, B1
- B0-B1, B2-B3
- B1-B2, B3-B0
- B0-B1-B2, B3
- B0-B1-B3, B2
- B0-B2-B3, B1
- B1-B2-B3, B0
- B0-B1-B2-B3
- B0, B1, B2, B3

Note that, as we discussed in Section II, optimal reconstruction does not allow intersecting lines in the block interior. The rules for connecting black runs are symmetric and equivalent with the rules for connecting white runs. For block boundaries with 5 or more runs, the shortest runs are ignored and considered as part of their connecting (surrounding) runs, so that the 4-run connection rule can be applied to the block. The short runs are then self-connected.

The 5 choices for 3-run blocks are signaled by 3 connection bits and the 14 choices for 4-run blocks are signaled by 4 connection bits. Table I lists the number of connection choices and the length of the binary code for each number of black runs. The fixed-length codes are uniquely decodable given the number of runs. We then use arithmetic coding to compress, which results in moderate gains (about 10%). The implementation is discussed in Section V.

C. Presmoothing the Cutset

The MRF-based reconstruction algorithm assumes that the underlying bilevel image satisfies the smoothness criterion. However, the actual images to be encoded are not always smooth. Moreover, due to the size of the grid relative to image details, there may be black blobs in the image that, after cutset sampling, are represented only as a single black run on a grid line, and then reconstructed as a single thin line segment, or

Original   Cutset   Reconstruction   Presmoothed
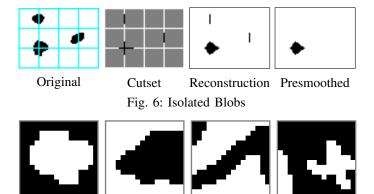
Fig. 6: Isolated Blobs

Fig. 7: Nonhomogeneous Blocks

are represented by parallel black runs on adjacent grid lines, but due to the presence of multiple runs, the connection bit cannot signal that the reconstruction should connect these runs. Examples of the former are shown in Figure 6, and of the latter in Figure 8(b) and (e), where only one connection bit was used. In both cases, the reproductions misrepresent the detail, and it is better to eliminate the detail before encoding. Although this may increase the reconstruction error rate, it improves the perceptual quality of the reconstruction, as can be seen in Figures 6 and 8(d), and has the additional benefit of reducing bitrate. Since the proposed bilevel coding algorithms are lossy, there is no reason to insist on lossless encoding of the grid samples, if we can improve bitrate and perceptual quality.

We thus use the following rule for removing a run in a block boundary. Starting from the shortest run, we flip the color of the run, i.e., eliminate the run, if the error in the interior of all the blocks adjacent to the run does not increase. This rule is not applied to 4×4 or smaller blocks, that is, when the number of interior pixels is less than the number of boundary pixels, unless the run is completely isolated, i.e., not connected with any interior pixels. Such run removals are typically associated with large blocks with multiple runs or isolated runs that are not connected to any interior pixels in the reconstruction.

IV. HIERARCHICAL LCC APPROACH

The distribution of shapes and sizes of black and white regions in a bilevel image is not necessarily homogeneous. For encoding large smooth regions, a large cutset grid size is most efficient. For encoding fine details, a small grid size is needed to preserve image structure and details. Figure 7 shows examples of blocks decoupled by a cutset, in which the block interior cannot be adequately reconstructed based on the boundary pixels and the MRF model we reviewed in Section II. Thus, adaptive block splitting is desirable.

We propose a hierarchical LCC (HLCC) coding approach that adapts the cutset to local image detail. The image is initially sampled on an $N \times N$ grid, defined in Section II. The blocks are processed in raster order. For each block, the block interior is reconstructed from its boundary specification, using connection bits when there are two or more black runs. The reconstructed block is then compared with the original block using a distortion metric. If the distortion exceeds a threshold, the block is subdivided into four subblocks. The

(a) Original   (b) Fixed   (c) Hierarchical

(d) Presmoothed   (e) Fixed Recon.   (f) Hier. Recon.
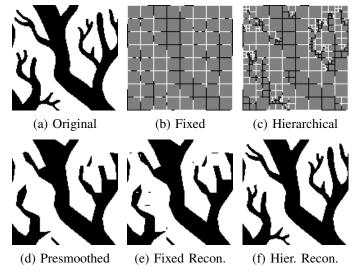
Fig. 8: Fixed vs. Hierarchical Cutset

process is repeated recursively for each subblock until all the subblocks satisfy the distortion threshold or a minimal block size is reached. The minimal block size is typically 1×1, so that the block splitting is controlled by the distortion threshold. An example of the resulting cutset sampling grid is shown in Figure 8. Note that the reconstructions assume only one connection bit.

For each block, the encoder uses a *splitting bit* to indicate whether it should be subdivided or not. The information to be encoded for an $N \times N$ block includes: (1) the boundary pixels; (2) a splitting bit; and (3) the information for encoding the subblocks if the block is split, or the connection bits (if any) if the block is not split. As in [13], we found that including the connection bits results in considerable improvements in rate-distortion performance.

For the distortion metric, we use the error rate, that is, the fraction of pixels changed by the encoding/decoding process. This is consistent with the criterion used for the connection decisions discussed by Reyes *et al.* [13], [14], [21]. We explored four variations of the splitting criterion: (1) the distortion over the entire block exceeds a fixed threshold; (2) the threshold decreases with block size; (3) the distortion of one of the subblocks exceeds a fixed threshold; and (4) taking into account the error distribution in subblocks. However, all of these strategies yielded essentially the same overall perceptual distortion at a given encoding rate, even though they each produced a different distribution of errors around the image. So, we picked the simplest strategy (1).

V. Lossless Coding

In the hierarchical LCC approach, the information to be encoded includes the pixels on the cutset, the connection bits, and the splitting bits. The connection bits are generated for blocks with two or more black runs, but are not encoded if the block is split. The splitting bits are generated and encoded for all blocks, except when the minimal block size is reached. The cutset pixels, connection bits, and splitting bits are ordered into separate streams, and losslessly encoded using an arithmetic

coder (AC). Here, as in [13], AC is based on the first-order conditional probabilities. We also tried second-order AC but did not offer any significant improvement. Overall, depending on the degree of compression, AC provides gains by factors of 3-6, almost entirely from encoding the cutset pixels; typically, the higher the compression the higher the gains.

An AC requires that for each value to be encoded there is an associated probability distribution. When encoding the cutset pixels, a first-order AC uses the conditional probability $p_i(x_i|t_i)$ of the value x_i of the i-th pixel, given t_i, the value of a specific pixel in the previously encoded bitstream, which is called the *context*. The conditional probabilities $p_i(x_i|t_i)$ are estimated by the fraction of previous occurrences of x_i with t_i as its context, and are updated each time a new pixel is encoded. The number of bits required for AC encoding of the cutset pixels is estimated by the empirical conditional entropy,

$$H_P = -\sum_{i=1}^{N_P} \log_2 p_i(x_i|t_i) \tag{4}$$

where N_P denotes the total number of cutset pixels, and $x_i, t_i \in \{0, 1\}$ denote the i-th cutset pixel and its context. The encoding of the connection and splitting bits is done in a similar fashion using a first-order AC.

The AC described above is a *one-pass* coder. The probability distributions are estimated in the same fashion by the encoder and the decoder. It is also possible to use a *two-pass* AC, whereby the probability distributions are first estimated over all the pixels (and all the connection and splitting bits, respectively) and transmitted to the decoder before the encoding begins. The estimate of the required number of bits for the cutset pixels becomes

$$H_P' = -N_P \sum_t p(t) \sum_x p(x|t) \log_2 p(x|t) \tag{5}$$

where N_P denotes the total number of cutset pixels, $x, t \in \{0, 1\}$ denote a cutset pixel and its context, and $p(t)$ denotes the empirical distribution of the context pixels.

In this paper, we use one-pass AC because it yields essentially the same bitrate as the two-pass AC. In fact, a one-pass AC coder may be more efficient than a two-pass coder because it can adapt to local image characteristics.

A. Hierarchical Probability Estimation

For first-order AC, the context for estimating the conditional probability distribution of a pixel on the cutset is a previously encoded neighboring pixel, located at either the top or the left on the cutset. If both the top and the left pixels have been previously encoded, then the one on the left is used. The context for a binarized connection bit is the previously encoded binarized connection bit, and for a splitting bit, the previously encoded splitting bit. As is conventional, the conditional probability estimates used when AC encoding a current cutset, connection, or splitting bit are relative frequencies of the past values of the same type of bit in instances with the same context as the current bit.

The hierarchical structure can be further utilized to improve bitrate efficiency by obtaining separate estimates of the conditional probabilities for different block sizes; we will refer
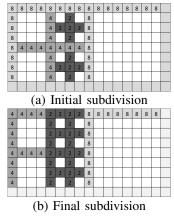
(a) Initial subdivision

(b) Final subdivision

Fig. 9: Level assignments for prob. distribution estimation

to these as *coding levels*. We have explored two schemes for assigning a pixel x_i to a coding level: based on the initial or the final subdivision of the cutset. Figure 9 illustrates an example of the two schemes. In this example, the initial, maximum block size is 8×8 pixels.

The blocks are recursively subdivided into 4×4 and 2×2 based on the splitting criterion. The initial subdivision scheme assigns pixels to the coding level that corresponds to the initial block size that they belong to before subdividing. This scheme can be directly adapted to a one-pass progressive coding scheme. The final subdivision scheme assigns pixels on the left and top boundaries of the smallest block they reside in to the corresponding coding level. This results in slightly better rate-distortion performance when subdivision occurs frequently. Note that in both schemes, the pixel and its context do not necessarily have to belong to the same coding level. However, it is the coding level of the pixel itself that determines what probability estimate it contributes to.

The decision bits (connection and splitting bits) are also encoded based on level-wise probability estimation. The connection bits, when they exist, are encoded only if there is no further splitting of the block. The splitting bits, on the other hand, are generated and encoded for every block and subblock.

B. Hierarchical Lossless Coding

The cutset pixels and extra bits are fed into an arithmetic coder following [25]. The number of bits for hierarchical LCC is estimated by total entropy H

$$H = \sum_n (H_{P_n} + H_{C_n} + H_{S_n}) \qquad (6)$$

where n denotes the level in hierarchical coding, which is determined by the block size, and H_{P_n}, H_{C_n}, and H_{S_n} are the empirical conditional entropies (defined as in (4)) for lossless encoding of the cutset pixels, the binarized connection bits, and the splitting bits, respectively. When the total entropy is divided by the number of pixels in an original image, it is the bitrate for encoding the image.

VI. PROGRESSIVE SCHEME

The hierarchical approach can be easily adapted to obtain a progressive coding scheme that allows encoding and decoding of an image as the bitstream is transmitted, progressively

adding image details to go from a coarse to a fine image reconstruction. The encoder and decoder can stop coding at any block size N_f (coding level) by adding a small overhead to the hierarchical approach.

In the hierarchical approach, the connection bits, when generated, are encoded only for non-splitting blocks. In the progressive scheme, the connection bits are always encoded, regardless of the block splitting decision, in order to provide a better reconstruction when the encoder stops before proceeding to the next coding level. The encoding of the extra connection bits only adds a very small overhead to the bitstream for a substantial improvement in the image quality of the partial reconstructions. Note also that in the progressive approach, any cutset smoothing must be done after decoding, as *post-smoothing,* because any isolated lines eliminated at one level may be connected at the next level.

The progressive coding scheme is obtained from the initial subdivision of the hierarchical coding. The bitstream at the coding level associated with block size $N/2 \times N/2$ includes the pixels on the $N/2 \times N/2$ grid, the connection bits associated with the $N/2 \times N/2$ blocks, and the split bits generated from previous coding level associated with $N \times N$ blocks, if it exists. The decoder uses the part of the bitstream from current and previous coding levels to reconstruct the image.
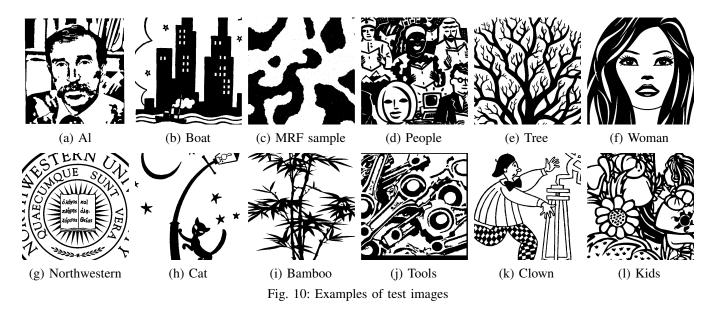
VII. EXPERIMENTAL RESULTS

The proposed approaches were tested on a variety of bilevel images. Examples of 512×512 test images are shown in Figure 10. These images satisfy the smoothness criterion, except for the "MRF sample" image, which includes isolated dots. To obtain the rate-distortion performance of the proposed techniques, we used the bitrate obtained with a one-pass first-order arithmetic coder, implemented in *C++,* based on ref. [25], and the contexts described in Section V. For the distortion we used the error rate.

A. Four-connectivity

As we mentioned in Section III-A, imposing 4-connectivity in the reconstruction does not result in any significant change in error rate. However, there is a small but noticeable improvement in perceptual quality. Figure 11 shows two examples of reconstructions without and with enforcement of 4-connectivity. Note the thicker tail in Figure 11(c). The bottom example in Figure 11 illustrates an added advantage of imposing 4-connectivity. While the reconstruction in Figure 11(e) has lower error (and could thus not be obtained via the use of a connection bit), imposing 4-connectivity results in a smoother, if not as accurate, reconstruction. Note also that in Figure 11(e) presmoothing the cutset would eliminate the line at the leftmost corner of the letter "N." In the rest of the experiments we allowed only 4-connected reconstructions.

B. Fixed-grid LCC with Extended Connection Bits

As we discussed in Section III-B, for fixed-grid LCC, the introduction of additional (extended) connection bits can improve rate-distortion performance when the block size is large. Table II shows the percentage of blocks with different

(a) Al   (b) Boat   (c) MRF sample   (d) People   (e) Tree   (f) Woman

(g) Northwestern   (h) Cat   (i) Bamboo   (j) Tools   (k) Clown   (l) Kids

Fig. 10: Examples of test images

(a) Original   (b) 8-connected   (c) 4-connected

(d) Original   (e) 8-connected   (f) 4-connected

Fig. 11: Reconstruction with 8- and 4-connectivity

TABLE II: Block % vs. number of runs ("Al" image)

%	number of black (white) runs on block boundary					
N	0	1	2	3	4	≥ 5
32	14.5	25.8	25.8	19.9	10.2	3.9
28	20.2	26.0	30.7	14.7	6.6	1.7
24	24.6	31.8	26.9	11.4	3.1	2.3
20	28.1	36.4	22.8	9.2	1.9	1.6
16	35.0	39.2	19.2	4.8	1.7	0.2
12	46.4	36.1	14.4	2.3	0.7	0.1
10	52.7	35.3	9.9	1.7	0.3	0
8	58.0	34.5	6.6	0.9	0.1	0
6	67.3	28.4	3.9	0.4	0	0
4	76.2	22.1	1.7	0.1	0	0
3	81.8	17.5	0.7	0	0	0
2	87.4	12.5	0.2	0	0	0

C. LCC with Presmoothed Cutset

Figure 14 shows an example of presmoothing the cutset, that is, eliminating isolated runs that do not connect to any pixels in the adjacent block interiors. Such isolated runs violate the smoothness criterion. In addition, presmoothing will eliminate speckles of noise. Pre-smoothing typically results in slightly increased error rate (E), decreased bitrate (R), and most importantly, improved perceptual quality. The overall rate-distortion performance also improves, as shown in Figure 15. Note that the improvement is pronounced for the "MRF sample" image, which contains a lot of noise speckles. In the remainder of the experimental results we will use presmoothing with the proposed techniques.

D. Hierarchical LCC

Figure 16 shows rate-distortion curves for the hierarchical LCC approach for initial grid sizes 32 and 16, and different splitting thresholds T, ranging from 1.0 to 0.0 (lossless). Note that the minimal block size is 1×1, so block splitting continues until the distortion threshold is met.

The figure also shows the rate-distortion curves of the fixed-grid approach for grid sizes ranging from 32 to 1. The figure shows that the hierarchical approach clearly outperforms

number of black (white) runs for the test image "Al." The block sizes range from 2 to 32. Note that the number of blocks with 3 or more runs is insignificant for $N \leq 8$, but becomes significant as N approaches 16 and higher. For $N < 32$, covering up to 4 pairs of black and white runs provides a good balance between reconstruction accuracy and bitrate overhead. Figure 12 shows that the use of extended connection bits, as opposed to the *baseline* approach that uses only up to one connection bit per block, improves the rate-distortion performance in the low bitrate region. Figure 13 shows examples of fixed-grid LCC with $N = 16$. It can be seen that the use of extended connection bits leads to significant reduction in the error rate (E) with a small increase in bitrate (R), but more importantly, there is a significant improvement in the perceptual quality of the reconstructions. In the remainder of the experimental results we will use the extended connection bits with the proposed techniques.
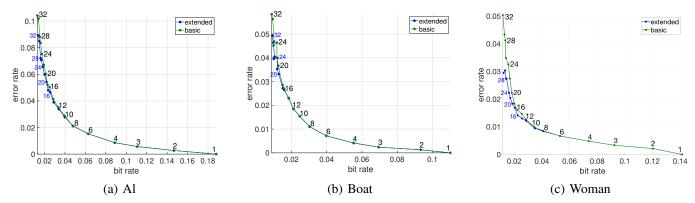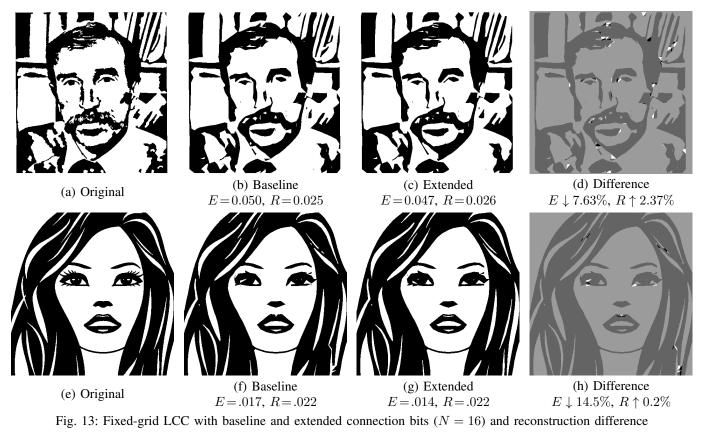
(a) Al   (b) Boat   (c) Woman

Fig. 12: Rate-distortion curves for fixed-grid LCC with baseline and extended connection bits (labeled with block size N).

(a) Original   (b) Baseline $E\!=\!0.050,\ R\!=\!0.025$ (c) Extended $E\!=\!0.047,\ R\!=\!0.026$ (d) Difference $E\!\downarrow\!7.63\%,\ R\!\uparrow\!2.37\%$

(e) Original   (f) Baseline $E\!=\!.017,\ R\!=\!.022$ (g) Extended $E\!=\!.014,\ R\!=\!.022$ (h) Difference $E\!\downarrow\!14.5\%,\ R\!\uparrow\!0.2\%$

Fig. 13: Fixed-grid LCC with baseline and extended connection bits ($N=16$) and reconstruction difference

the fixed-grid approach in the high bitrate range, because it encodes the image details more efficiently. In the low bitrate range, the advantage of the hierarchical approach is less pronounced, because there is very little splitting. The figure also shows that the rate-distortion curves of the hierarchical approach with smaller (16) initial block size may start above but end up as good as or better than those with higher initial block size (32).

Figure 17 shows the hierarchical cutsets and the corresponding decoded images for different block splitting error rate thresholds; the initial block size is 16. The figure demonstrates that the hierarchical approach preserves key structures of the image and adds details as the splitting threshold decreases. The threshold provides an upper bound for the error rate of the decoded image, and thus controls the compression quality consistently across different images.

Figure 18 compares the hierarchical approach with the fixed-grid approach. The comparison shows that the hierarchical approach yields better perceptual quality and lower error rate than the fixed-grid approach at comparable or even lower bitrates. Note the improvements in the windows of the "boat" image and the eyes of the "woman." These can be attributed to the adaptation to image details with efficient bit allocation.

E. Level-wise Arithmetic Coding

Figure 19 shows that calculating probability distributions separately for each of the hierarchical coding levels (block sizes) substantially improves the rate-distortion performance compared to calculating probability distributions for all block sizes together. The rate-distortion curves for hierarchical coding utilizing level-wise probability distributions based on the initial and the final subdivision are virtually identical, and
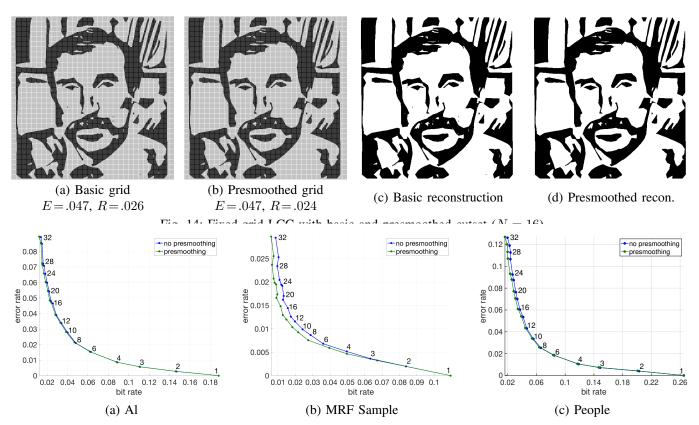
(a) Basic grid
$E = .047, R = .026$

(b) Presmoothed grid
$E = .047, R = .024$

(c) Basic reconstruction

(d) Presmoothed recon.

Fig. 14: Fixed-grid LCC with basic and presmoothed cutset ($N = 16$).

(a) Al

(b) MRF Sample

(c) People

Fig. 15: Rate-distortion curves for fixed-grid LCC: basic and presmoothed cutset (labeled with block size N). The blue curves are are the same as those in Figure 12.
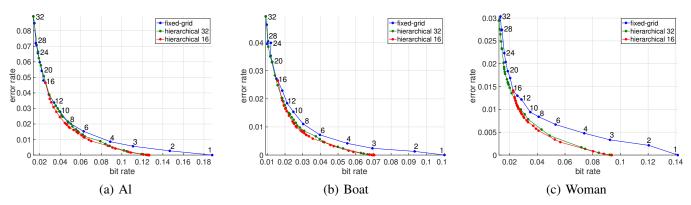

(a) Al

(b) Boat

(c) Woman

Fig. 16: Rate-distortion curves for hierarchical (initial block sizes 32 and 16) vs. fixed-grid LCC (with block size N).

only shown as a blue line in the figure. We select the initial subdivision in the remainder of this paper so that it is consistent with the progressive scheme.

Table III shows how the bits are distributed at each coding level for the test image "Al," losslessly encoded (splitting threshold is 0) at 1.2 bits per pixel. The table includes the number of blocks and cutset pixels to be encoded at each level, the connection bits (up to 4 bits per block, as explained in Section V and Table I), and the splitting bits (1 bit per block at each level, excluding the last level, as described in Section V). As the cutset grid becomes finer, the ratio of splitting bits over cutset pixels increases. The connection bits are encoded only if generated by non-splitting blocks, and thus contribute a relatively small bitrate overhead.

TABLE III: Encoded bits for lossless encoding of "Al" image

Block size	blocks	pixels	connection bits	splitting bits
16	1024	32769	18	1024
8	2272	16472	66	2272
4	4512	14664	151	4512
2	5264	6580	63	5264
1	2192	548	0	0
Overall		71033	298	13072

F. Progressive Scheme

Figure 20 shows progressive coding starting with block size 16 and ending with $N_f = 16$, 8, 4 and 2. As we discussed in Section VI, the cutset post-smoothing is performed after decoding (reconstruction), thus removing isolated lines on the cutset grid without affecting the decoding of the next level.
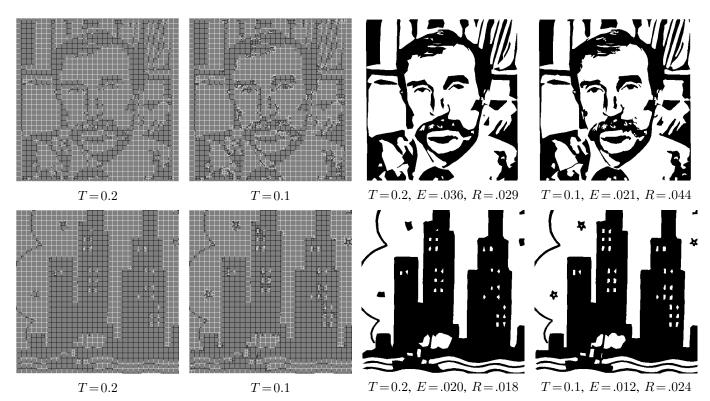
| $T=0.2$ | $T=0.1$ | $T=0.2, E=.036, R=.029$ | $T=0.1, E=.021, R=.044$ |

| $T=0.2$ | $T=0.1$ | $T=0.2, E=.020, R=.018$ | $T=0.1, E=.012, R=.024$ |

Fig. 17: Hierarchical LCC: Cutsets and decoded images with initial block size 16 and different block splitting error rate thresholds T, resulting error rates E, and bitrates R.
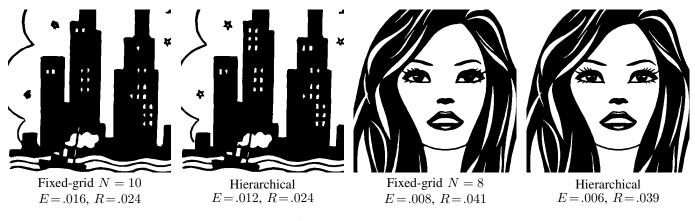

| Fixed-grid $N = 10$ $E=.016, R=.024$ | Hierarchical $E=.012, R=.024$ | Fixed-grid $N = 8$ $E=.008, R=.041$ | Hierarchical $E=.006, R=.039$ |

Fig. 18: Comparison of fixed-grid and hierarchical LCC approaches

TABLE IV: Lossless encoding bitrates

Test image	Hierarchical	Progressive	Overhead
Al	.127	.130	.003
Boat	.070	.071	.001
Tree	.170	.175	.005
Woman	.094	.096	.002

When the final block size is 1, the reconstruction is lossless. The splitting threshold is 0, that is, we always split if an error exists. Observe how the reconstructions add details as the final block size decreases. Table IV compares the progressive and non-progressive schemes, for lossless coding based on the initial subdivision. It is clear that the overhead for the additional connection bits is very small.

G. Comparison with Other Approaches

We now compare the proposed hierarchical LCC (HLCC) with other bilevel compression approaches, namely, the fixed-grid LCC of Reyes *et al.* [13], the finite automata approach by Culik and Valenta [15], [16], and the JBIG and JBIG2 standards. Figure 21 compares the rate-distortion performance of the four approaches, while Figure 22 shows decoded images at comparable bitrates in the lossy but essentially structurally lossless range. The rate-distortion curves show that both LCC and HLCC clearly outperform the finite automata approach, with HLCC offering a definite advantage over LCC. The JBIG and JBIG2 standards offer efficient lossless compression. However, HLCC provides essentially structurally lossless performance at lower bitrates than the lossless JBIG and JBIG2 standards, as can be seen in Figure 22. For the JBIG2

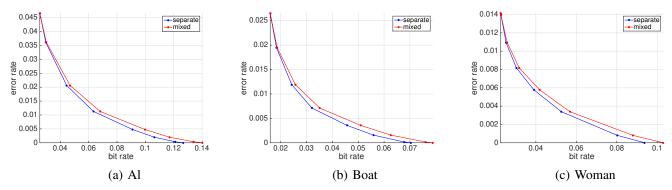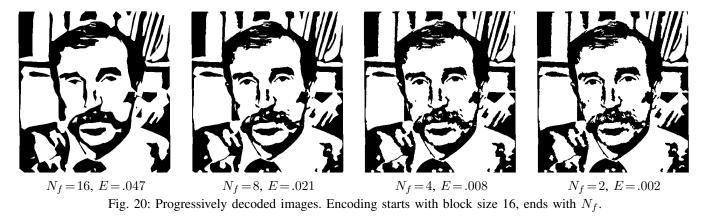(a) Al   (b) Boat   (c) Woman

Fig. 19: Rate-distortion curves: Lossless coding based on level-wise (blue) vs. mixed (red) probability distributions for all block sizes; points on curves obtained with initial block size 16 and varying splitting threshold from 1 to 0.

$N_f = 16$, $E = .047$ $N_f = 8$, $E = .021$ $N_f = 4$, $E = .008$ $N_f = 2$, $E = .002$

Fig. 20: Progressively decoded images. Encoding starts with block size 16, ends with N_f.

implementation we used the JBIG2 open source encoder at [26]; there is no lossy option for our experimental content. The figure also shows a clear advantage of HLCC over LCC in the fine details, especially for the "Woman" image.

H. Compression Quality Control

A key advantage of the proposed hierarchical coder is that it is a fixed error rate coder. Achieving essentially constant quality across an image ensures that perceptually important details are well and efficiently encoded, even when they occupy only a small fraction of the image. In HLCC the error rate is controlled on a block by block basis, namely, it cannot exceed a specified threshold. Thus, the bitrate required to achieve a given quality level depends on image content. In contrast, in the fixed-grid approach, the error rate is determined indirectly by the block size. The hierarchical approach controls the quality and the bitrate follows, while the fixed-grid approach does not directly control either.

To demonstrate this fact, we collected 128 bilevel images, ranging from simple silhouettes, to graphics, to relatively complex cartoons, to complex sketches. Some of these images, especially the complicated sketches, do not satisfy the piecewise smooth property. The size and detail of these images varies. Figure 23 shows some of the test images, sorted from easy to difficult in terms of the required bitrate. The images were encoded with the fixed-grid approach using different cutset sampling steps (16, 8, and 4) and the hierarchical approach starting with block size 64 and different splitting thresholds (0.05, 0.03, and 0.01). Note that when the splitting

threshold is 1, the hierarchical approach reduces to the fixed-grid approach with additional overhead for encoding the splitting bit; when the splitting threshold is 0, the hierarchical coding becomes lossless. The rate-distortion performance is shown in Figure 24. Note that the distribution of the points for each threshold of the hierarchical approach is more or less flat, that is, the quality is fixed. On the other hand, the fixed-grid approach yields results in varied error rate and varied bitrate for a given block size.

I. Cummulative Comparison of HLCC, LCC, JBIG, and JBIG2

We now compare the rate-distortion performance of the HLCC and LCC approaches averaged over the 129 images we mentioned in the previous subsection. The average bitrate is computed as the ratio of the total number of bits needed to encode the images to the total number of pixels in the 129 images. Similarly, the average error rate is computed as the ratio of the total error to the total number of pixels. Figure 25 shows the results. The cummulative rates for JBIG and JBIG2 are also shown.

VIII. Conclusions

We presented HLCC, a simple, efficient, and effective algorithm for hierarchical lossy bilevel image compression. The main advance over the fixed-grid LCC approach on which it is based is that it adapts the grid size to local image detail, thus eliminating the most important LCC drawback, the fact
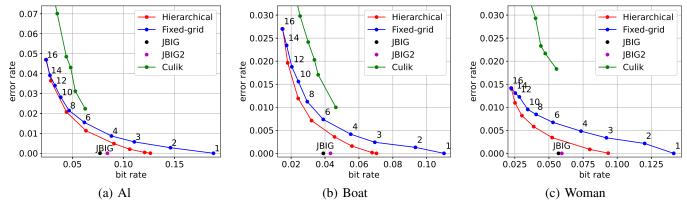
(a) Al   (b) Boat   (c) Woman

Fig. 21: Rate-distortion comparison of HLCC (initial block size 16, splitting thresholds: 1, 0.2, 0.1, 0.05, 0.03, 0.02, 0.01, 0); LCC (labeled with block size N); Culik-Valenta (error rate factors 400, 300, 200, 100, 50, 1); JBIG; and JBIG2

JBIG $R=.077$ JBIG2 $R=.084$ HLCC $E=.011$ $R=.063$ LCC $E=.015$ $R=.063$ Culik $E=.024$, $R=.062$

JBIG $R=.056$ JBIG2 $R=.059$ HLCC $E=.006$ $R=.039$ LCC $E=.009$ $R=.041$ Culik $E=.018$, $R=.056$
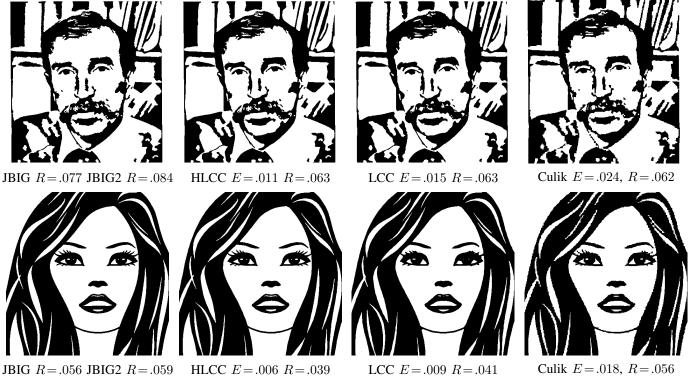
Fig. 22: Comparison: JBIG; JBIG2; HLCC (splitting threshold 0.05); LCC ($N=6$); Culik-Valenta (error rate factor 1)

that it cannot be optimized for detail variations within an image. The grid adaptation makes HLCC a fixed error rate coder that provides constant quality, controlled by a single parameter (distortion threshold), both within an image and across all images. The proposed approach also introduces several other enhancements of the LCC approach that improve reconstruction accuracy and perceptual quality. These include the use of multiple connection bits that provide structural information by specifying which black (or white) runs on the boundary of a block must be connected, a boundary presmoothing step, stricter connectivity constraints, and more elaborate probability estimation for arithmetic coding. Overall, the proposed approach provides better rate-distortion performance than the fixed-grid approach, especially at high coding rates, and better visual quality at low coding rates. While in some cases the improvement in quantitative rate-distortion per-formance is modest, constant quality ensures that perceptually important details, which occupy a small fraction of the image, are well and efficiently encoded, resulting in significantly better overall perceptual quality. In addition, we proposed a progressive scheme, which refines the image reconstruction as more bits are transmitted, with very small additional overhead.

IX. ACKNOWLEDGEMENT

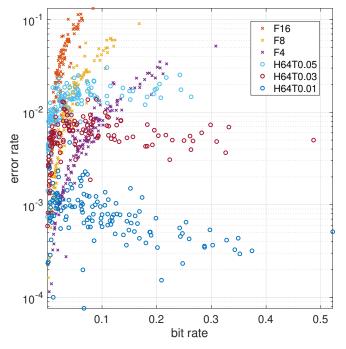| 5333×3333 | 2000×2000 | 2400×1874 | 512×512 | 984×1024 | 512×512 | 1683×1581 |

Fig. 23: Examples of easy to hard test images

Fig. 24: Hierarchical vs. fixed-grid approach. Each color represents a coder with a set of specific parameters. Each point represents the bitrate and error rate of a test image.
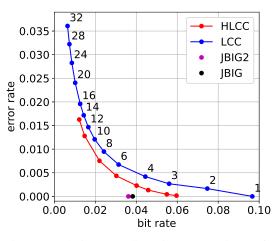

Fig. 25: Cummulative rate-distortion comparison of HLCC (initial block size 16, splitting thresholds: 1, 0.2, 0.1, 0.05, 0.03, 0.02, 0.01, 0); LCC (labeled with block size N); JBIG; and JBIG2

REFERENCES

[1] S. Zha, T. N. Pappas, and D. L. Neuhoff, "Hierarchical bilevel image compression based on cutset sampling," in *Proc. Int. Conf. Image Processing (ICIP)*, Orlando, FL, Oct. 2012, pp. 2517–2520.

[2] ISO/IEC 14496, "Information technology – generic coding of audio-visual objects," July 1999.

[3] M. Kunt, A. Ikonomopoulos, and M. Kocher, "Second-generation image-coding techniques," *Proc. IEEE*, vol. 73, no. 4, pp. 549–574, Apr. 1985.

[4] H. Hampel, R. B. Arps, C. Chamzas, D. Dellert, D. L. Duttweiler, T. Endoh, W. Equitz, F. Ono, R. Pasco, I. Sebestyen, C. J. Starkey, S. J. Urban, Y. Yamazaki, and T. Yoshida, "Technical features of the JBIG standard for progressive bi-level image compression," *Signal Processing: Image Communication*, vol. 4, pp. 103–11, 1992.

[5] K. Sayood, *Introduction to Data Compression*, 4th ed. Morgan Kaufmann, 2012.

[6] P. G. Howard, F. Kossentini, B. Martins, S. Forchhammer, and W. J. Rucklidge, "The emerging JBIG2 standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, no. 7, pp. 838–848, Nov. 1998.

[7] F. Ono, W. Rucklidge, R. Arps, and C. Constantinescu, "JBIG2-the ultimate bi-level image coding standard," in *Proc. Int. Conf. Image Proc.*, vol. 1. IEEE, 2000, pp. 140–143.

[8] A. K. Katsaggelos, L. P. Kondi, F. W. Meier, J. Ostermann, and G. M. Schuster, "MPEG-4 and rate-distortion-based shape-coding techniques," *Proc. IEEE*, vol. 86, no. 6, pp. 1126–1154, June 1998, special issue on *Multimedia Signal Processing*.

[9] A. Vetro, Y. Wang, and H. Sun, "Rate-distortion modeling for multiscale binary shape coding based on Markov random fields," *IEEE Trans. Image Process.*, vol. 12, pp. 356–364, Mar. 2003.

[10] S. M. Aghito and S. Forchhammer, "Context-based coding of bilevel images enhanced by digital straight line analysis," *IEEE Trans. Image Process.*, vol. 15, no. 8, pp. 2120–2130, 2006.

[11] K. Dhou, "An innovative design og a hybrid chain coding algorithm for bi-level image compression using an agent-based modeling approach," *Applied Soft Computing Journal*, vol. 79, pp. 94–110, 2019.

[12] P. Elia and J.-J. Ding, "Morphological residue encoding and piecewise approximation techniques for lossless binary image compression," in *IEEE Asia Pacific Conf. Circuits Systems (APCCAS)*. IEEE, 2019, pp. 353–356.

[13] M. G. Reyes, X. Zhao, D. L. Neuhoff, and T. N. Pappas, "Lossy compression of bilevel images based on Markov random fields," in *Proc. Int. Conf. Image Processing (ICIP-07)*, vol. 2, San Antonio, TX, Sept. 2007, pp. II–373–II–376.

[14] M. G. Reyes, D. L. Neuhoff, and T. N. Pappas, "Lossy cutset coding of bilevel images based on Markov random fields," *IEEE Trans. Image Process.*, vol. 23, no. 4, pp. 1652–1665, Apr. 2014.

[15] K. Culik II and V. Valenta, "Finite automata based compression of bi-level and simple color images," in *Proc. IEEE Data Compression Conf.*, Snowbird, Utah, 1996.

[16] ——, "Finite automata based compression of bi-level and simple color images," *Computer & Graphics*, vol. 21, no. 1, pp. 61–68, 1997.

[17] M. Kunt, M. Benard, and R. Leonardi, "Recent results in high-compression image coding," *IEEE Trans. Circuits Syst.*, vol. 34, no. 11, pp. 1306–1336, Nov. 1987.

[18] T. N. Pappas, D. L. Neuhoff, H. de Ridder, and J. Zujovic, "Image analysis: Focus on texture similarity," *Proc. IEEE*, vol. 101, no. 9, pp. 2044–2057, Sept. 2013.

[19] T. N. Pappas, J. Zujovic, and D. L. Neuhoff, "Image analysis and compression: Renewed focus on texture," in *Visual Information Processing and Communication*, ser. Proc. SPIE, vol. 7543, San Jose, CA, Jan. 2010.

[20] M. G. Reyes, D. L. Neuhoff, and T. N. Pappas, "MAP interpolation of an ising image block," in *Science and Information Conference: Advances in Computer Vision (CVC)*, vol. 943. Orlando, FL: Springer, Apr. 2019, pp. 237–256.

[21] M. G. Reyes, X. Zhao, D. L. Neuhoff, and T. N. Pappas, "Structure-preserving properties of bilevel image compression," in *Human Vision and Electronic Imaging XIII*, ser. Proc. SPIE, B. E. Rogowitz and T. N. Pappas, Eds., vol. 6806, San Jose, CA, Jan. 2008, pp. 680 617–1–12.

[22] J. E. Bresenham, "Algorithm for computer control of a digital plotter," *IBM Systems Journal*, vol. 4, no. 1, pp. 25–30, Mar. 1965.

[23] A. Rosenfeld and A. C. Kak, *Digital Picture Processing*. New York: Academic Press, 1976.

[24] G. Bradski, "The openCV library," *Dr. Dobb's Journal of Software Tools*, 2000.

[25] I. H. Witten, R. M. Neal, and J. G. Cleary, "Arithmetic coding for data compression," *Communications of the ACM*, vol. 30, no. 6, pp. 520–540, 1987.

[26] "JBIG2 encoder." [Online]. Available: https://github.com/agl/jbig2enc/

Shengxin Zha received the B.S. degree in electrical engineering from the Xi'an Jiaotong University in 2010, and the M.S. and Ph.D. degrees from Northwestern University in electrical and computer engineering and electrical engineering, respectively. In 2016 she joined Facebook as a research engineer, and since 2020, as a technical lead manager. Here research interests are in image processing, computer vision, AI, and machine learning.

Thrasyvoulos N. Pappas (M'87, SM'95, F'06) received the S.B., S.M., and Ph.D. degrees in electrical engineering and computer science from MIT in 1979, 1982, and 1987, respectively. From 1987 until 1999, he was a Member of the Technical Staff at Bell Laboratories, Murray Hill, NJ. In 1999, he joined the ECE Department at Northwestern University. His research interests are in human perception and electronic media, and in particular, image quality and compression, image analysis, content-based retrieval, model-based halftoning, and tactile and multimodal interfaces. Prof. Pappas is a Fellow of SPIE and IS&T. He has served as Editor-in-Chief of the IEEE Transactions on Image Processing, Vice President-Publications for the Signal Processing Society (SPS) of IEEE, and co-chair of SPIE/IS&T Human Vision and Electronic Imaging Conference. He is currently Editor-in-Chief of the IS&T Journal of Perceptual Imaging.

David L. Neuhoff (S'72, M'74, SM'83, F'94, LF'14) received the B.S.E. degree from Cornell University in 1970, Ithaca, NY, USA, and the M.S. and Ph.D. degrees in electrical engineering from Stanford University, Stanford, CA, USA in 1972 and 1974, respectively. He served as a faculty member of the University of Michigan, Ann Arbor, MI, from 1974 until his retirement in June 2019 as the Joseph E. and Anne P. Rowe Emeritus Professor of Electrical. From 1984 to 1989, he was an Associate Chair of the EECS Department, and again from 2008 to 2018. He spent two sabbaticals at Bell Laboratories, Murray Hill, NJ, USA, and one at Northwestern University, Evanston, IL, USA. His research and teaching interests include communications, information theory, and signal processing, especially data compression, quantization, image coding, image similarity metrics, source-channel coding, sensor networks, and Markov random fields. Dr. Neuhoff co-chaired the 1986 IEEE International Symposium on Information Theory, was Technical Co-Chair of the 2012 IEEE Statistical Signal Processing Workshop, and initiated the IEEE Information Theory Society effort to install statues of Claude Shannon around the U.S.. He has twice been an Associate Editor for the IEEE Transactions on Information Theory, has twice served on the Board of Governors of the IEEE Information Theory Society, and was its president in 2006.