AN EFFICIENT CODING SCHEME FOR BINARY IMAGES BASED ON THE MORPHOLOGICAL SKELETON REPRESENTATION

R. Kresch and D. Malah

Department of Electrical Engineering
Technion - Israel Institute of Technology
Haifa 32000, Israel. E-mail: kresch@dekel.technion.ac.il

ABSTRACT

This work presents a new efficient scheme for *lossless* compression of binary images, by coding their Morphological Skeleton Representation. Computer simulations indicate that, typically, the proposed coding scheme significantly improves the coding rates obtained by the best previously known schemes for Skeleton coding, and is substantially more efficient than coding the original binary image by Chain Code, Quadtree and Run-length/Huffman methods. Comparison to existing coding standards for scanned bilevel documents places the proposed algorithm between the Group 3 and Group 4 algorithms, in terms of compression efficiency. The proposed algorithm is fast, when properly implemented.

The proposed scheme is based on new theoretical results obtained by the authors concerning properties of the Morphological Skeleton Representation.

1. INTRODUCTION

Mathematical Morphology [1] is a rapidly-growing non-linear theory for Image Processing and Analysis, based on Set-Theory, which deals mainly with the *geometrical* characteristics of images.

One important tool in Mathematical Morphology is the Morphological *Skeleton* Decomposition, which provides a meaningful, compact representation for binary images. It consists of a *thin simplification* of the image, conveying relevant *geometrical* information, such as size and connectivity, of the objects contained in it.

The Morphological Skeleton Representation is defined as the collection of *centers* and *sizes* of all the *Maximal Elements* inscribable in the given shape. In the above definition, *Elements* are shapes picked from a pre-defined family, such as squares or discs of increasing sizes, and *Maximal Elements* are elements from that family contained in the shape and not included in any other bigger element contained in the shape). The *centers* of Maximal Elements are called *Skeleton points*,

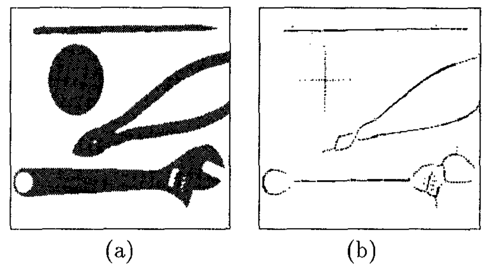


Figure 1: (a) 256 × 256-pixel binary image "Tools", and (b) its Skeleton.

and their sizes are called *quench values.* The map relating each Skeleton point to its quench value is called *Quench Function.* Fig. 1 presents a binary image, and its skeleton, computed by using a family of squares of sizes $(2n+1) \times (2n+1)$ pixels, $n = 0, 1, \ldots$.

The Morphological Skeleton Representation of discrete binary images is very *sparse.* This encouraged researchers to examine its use in Image Coding [2, 3, 4, 5]. However, the compression rates achieved until now by lossless-coding of the Skeleton were only comparable to (and sometimes even worse than) other simpler methods (such as Chain Coding, Quadtree Decomposition and Run-length Coding) applied directly to the original image. This made many researchers skeptical about Skeleton-based Coding. However, little was proposed so far concerning the improvement of the Skeleton *coding scheme* [2, 3].

2. BASIC CONCEPTS FROM BINARY MATHEMATICAL MORPHOLOGY

A discrete binary (bilevel) image is considered in Mathematical Morphology as a *set* in $\mathcal{Z}^2$, containing the foreground-pixels of the given image. In order to process and transform this set, one defines another set in $\mathcal{Z}^2$, usually simple and small, such as a 3×3-pixel square, which is called a *structuring element.* The morphological operations are based on interactions of a given image with a structuring element [1].

2.1. Morphological Operations

Let X be a a given set, and B a structuring element, both in $\mathcal{Z}^2$. One defines the *dilation* and the *erosion* of X by B, respectively $X \oplus B$ and $X \ominus B$, in the following manner [1, 2]:

$$X \oplus B \triangleq \bigcup_{x \in X} B_x, \tag{1}$$

$$X \ominus B \triangleq \bigcap_{x \in X} B_{-x}, \tag{2}$$

where B_x denotes the "*translation* of B to the point x", i.e., $B_x \triangleq \{b + x \mid b \in B\}$. If B is a 3×3 square, centered at the origin, then the dilation attaches to X those background pixels (points belonging to the complement of X) having a pixel of X within their 8-pixel neighborhood, and the erosion removes from X the pixels that have a background pixel within its 8-pixel neighborhood.

Binary *filtering* is obtained by compositions of the above operations. The *opening* and the *closing* of X by B, respectively $X \circ B$ and $X \bullet B$, are defined by:

$$X \circ B \triangleq (X \ominus B) \oplus B, \tag{3}$$

$$X \bullet B \triangleq (X \oplus B) \ominus B, \tag{4}$$

Opening usually removes from X isolated points and thin segments, and closing closes small holes in X.

2.2. Discrete Morphological Skeleton Representation

In order to calculate the Skeleton, a family of elements of different sizes must be defined. An element of integer "size" n is defined as $(n-1)$ times the dilation of an arbitrary structuring element B by itself, and is denoted by nB. An element of "radius" zero is the set containing only the origin, and the dilation, erosion, opening and closing by it is the identity operation.

The discrete Morphological Skeleton Representation is calculated with Lantuéjoul's formula [1, 2]:

$$S_n = X \ominus nB - (X \ominus nB) \circ B, \quad n = 0, 1, \ldots \tag{5}$$

where S_n, called Skeleton Subset of order n, is the set of Skeleton points related to (center of) a Maximal Element of size n. The Morphological Skeleton Representation consists of the collection of Skeleton Subsets, for $n = 0, 1, \ldots$. The Skeleton Subsets can be efficiently calculated since the sets $X \ominus nB$, $n = 0, 1, \ldots$, satisfy the recursion: $X \ominus nB = [X \ominus (n-1)B] \ominus B$. The Skeleton in Fig 1(b) was obtained using (5) with the previously mentioned 3×3 squared structuring element B.

The original shape X can be *fully* or *partially* reconstructed from the collection of Skeleton Subsets $\{S_n\}$ by the formula [1, 2]:

$$X \circ kB = \bigcup_{n \geq k} S_n \oplus nB. \tag{6}$$

Full reconstruction is obtained for $k = 0$ in (6). Setting $k > 0$ provides a partial reconstruction; it is equivalent to discarding the Skeleton Subsets of order smaller than k, and the result is a "simplified" version (opening) of X. The greater is k, the more simplified is the result.

2.3. Connectivity and Boundaries

In Mathematical Morphology, *Connectivity* is a concept related to a *symmetric* structuring element B according to the following definition: Two points x and y in $\mathcal{Z}^2$ are connected iff $x \in B_y$ (or, equivalently, $y \in B_x$). In words, two points are connected if one belongs to the neighborhood (defined by B) of the other. A 3×3 squared B defines, in this case, the well-known 8-pixel Connectivity.

Based on the above definition, one defines for any discrete binary shape X *two* boundaries: internal and external. The internal boundary is the set of points in X connected to some point in the background (the complement of X), whereas the external boundary is the set of points in the background connected to some point in X.

The above concepts are important for understanding the proposed algorithm.

2.4. Ultimate Erosions

Also important in this work is the concept of *Ultimate Erosion* [1], which can be defined as following.

Definition 1 *The* Ultimate Erosion *of order n, denoted as U_n, is the set of points in S_n which is not linked to the set $(X \ominus nB) \circ B$ by a connected path of Skeleton points.*

Intuitively, the Ultimate Erosions are those skeleton points with maximal quench value within each region of the original shape. (See the example in Fig. 2(a)). They are usually a very small percentage of the Skeleton.

3. PREVIOUS APPROACHES FOR SKELETON CODING

There are variations of two main Skeleton coding schemes in the literature: (1) Chain Coding of the Skeleton lines [3], and (2) Run-length Coding of the Skeleton Subsets [2].

The motivation for the first of the above schemes is that, in the continuous case, the Skeleton lines of connected shapes are almost always connected. However, in the discrete case, as opposed to the continuous case, the skeleton lines may have many gaps, and this considerably reduces the efficiency of the Chain Coding.

The second method considers each Skeleton Subset as a very sparse binary image, and therefore suitable for very low bit-rate coding. The Skeleton Subsets are coded in decreasing order of n, providing a *progressive transmition* scheme, since according to reconstruction formula (6), if the decoding is halted at a certain point, a simplified version of the original image is obtained. However this coding method is inefficient because coding each skeleton subset independently does not take into account the strong correlation existing between them (which is a consequence of the above mentioned partial connectivity of the skeleton lines).

4. PROPOSED CODING SCHEME

In this section we propose an efficient coding scheme of the Skeleton Representation. In comparison to the two previous schemes described in section 3, it is an hybrid method, since it takes into account the *Skeleton connectivity*, as the first scheme, and is suitable to *progressive transmition*, as the second one. Moreover, it is based on *new theoretical properties* of the Skeleton Representation, obtained recently by the authors [6], which are not considered in the previous schemes. As a consequence of all of the above, the proposed scheme typically provides better compression of binary images than the previous schemes (see section 5 bellow).

4.1. New Skeleton Properties

The new Skeleton properties, presented in [6], are summarized here in two theorems.

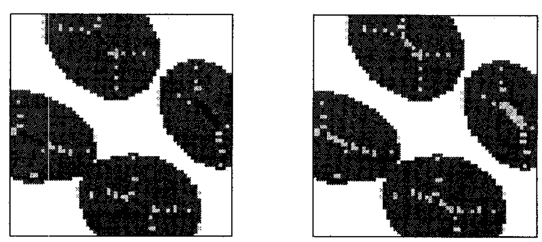
The first one strongly relates to the concept of *Ultimate Erosion* (see section 2.4 above). The theorem assumes that the same symmetric structuring element is used in the skeletonization process and in the Connectivity definition (see section 2.3 above). Throughout this paper, we consider the 3×3-squared structuring element for both purposes.

Theorem 1 *One can discard from a Discrete Skeleton Representation the* quench values *of the Skeleton points which* are not *Ultimate Erosions, and still the original image can be* fully *reconstructed.*

In [6], the above property is proved in a constructive way, providing a reconstruction algorithm to the above Skeleton with sampled Quench Function. The reconstruction algorithm is iterative, finding at each step n (n varies from its maximal value down to zero) which of the Skeleton points that are not Ultimate Erosions have quench value n; these are the Skeleton connected components that "touch" the set $(X \ominus nB) \circ B$, available (recursively computed by the algorithm) at each iterative step. Our proposed coding scheme, presented bellow, is based on this algorithm.

Actually, not all the ultimate-erosion points need to have their quench value stored. This is due to the following property stated and proved in [3] in a slightly different and more restrictive context: The quench values of all the skeleton points in any *connected component* of the skeleton are the same. Therefore, for every connected component in the set of ultimate erosions, one needs to store only the quench value of one point. Note that the set of ultimate erosions is usually a very small subset of the skeleton points, and, due to the above considerations, only a small percentage of them need to have their quench values stored! (See Fig. 2(b)).

The second theorem on which the proposed scheme is based is presented bellow. It permits *deterministic prediction* of information about S_n from the knowledge about the previously coded points. Assume that the above reconstruction algorithm related to Theorem 1 is presently at step n, and that the Skeleton Subset of order n, S_n, is to be coded now. As before, the set $(X \ominus nB) \circ B$ is available, and we denote it as Y_{n+1}. According to the theorem, the coder and the decoder



(a) (b)

Figure 2: (a) Skeleton and Ultimate Erosions of a portion of the image "Coffee Grains". The Ultimate Erosions are the black skeleton points. (b) A subset of the Ultimate Erosions (the four black points). Their quench values, in addition to the position of all the skeleton points, are sufficient for perfect reconstruction.

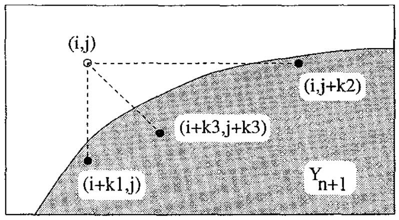


Figure 3: A point (i, j) predicted not to belong to S_n according to Theorem 2.

can predict that a certain portion of the space does not contain skeleton points from S_n. The Coding algorithm then tests points in the space according to this theorem.

In [6] we present the theorem in its generic form, but since it yields a computationally demanding test, we also present there a simplified, much faster test for the 3×3 squared structuring element, which we present here in Theorem 2.

Theorem 2 *Let* $(i, j) \in \mathcal{Z}^2$*. If any of the triplets*

$$\begin{array}{l}\{(i+k_1, j), (i, j+k_2), (i+k_3, j+k_3)\},\\ \{(i-k_1, j), (i, j-k_2), (i-k_3, j-k_3)\},\\ \{(i+k_1, j), (i, j-k_2), (i+k_3, j-k_3)\},\\ \{(i-k_1, j), (i, j+k_2), (i-k_3, j+k_3)\},\end{array}$$

for any integers k_1*,* k_2 *and* k_3 *in the interval* $[2, 2n+1]$*, is contained in* Y_{n+1}*, then the point* (i, j) *does not belong to* S_n*.*

The points in space *not* satisfying the condition of Theorem 2 can either be or not be a S_n point. Fig. 3 shows an example of a point which is predicted not to belong to S_n.

4.2. The Algorithm

In general lines, the proposed algorithm is as follows. After the Skeleton Representation is calculated, the coding is performed in the same way as the decoding, i.e., by reconstructing the original image. Let N be the maximum quench value. Initially, for each of the Ultimate Erosions U_n, $0 \leq n \leq N$, a set $\tilde{U}_n$ is formed, containing one point of each connected component of U_n. Then, the points in the above sets have their position and quench value coded. At this point, the main loop starts. At each step n, which varies from its maximum value, N, down to 0, a scanning procedure is performed on the *external boundary* of Y_{n+1} and of $\tilde{U}_n$. Only the external boundary have to be searched for points in S_n, since the Skeleton points in S_n are necessarily linked either to Y_{n+1}, if it is not an Ultimate Erosion point, or to $\tilde{U}_n$, otherwise. Some points in the above scan can be predicted not to belong to S_n by the test in Theorem 2; these points are skipped. The Skeleton points found in the above scan *must* belong to S_n (according to the reconstruction algorithm related to Theorem 1), and their position are coded by an arithmetic coder. When a skeleton point is found, its boundary is searched for other connected skeleton points in a recursive way, before the main scanning procedure goes on.

This procedure is detailed in the following algorithm:

1. Calculate the Skeleton Subsets S_n, $0 \leq n \leq N$, with a 3×3-squared structuring element. Form the sets $\tilde{U}_n$ as specified above.
2. $n \leftarrow N$. $Y_{N+1} \leftarrow \emptyset$.
3. $Z \leftarrow (Y_{n+1} \cup \tilde{U}_n)$.
4. $p \leftarrow$ (an external boundary point of Z). If there are no more external boundary points to scan, go to step 9.
5. Check (by means of Theorem 2) if p can belong to S_n or not. If it cannot, go to step 4.
6. Send to the Arithmetic Coder a "0" if p is not a skeleton point or a "1" otherwise. Use an adaptive probability model.
7. If a "1" was sent, $Z \leftarrow (Z \cup \{p\})$. Otherwise, go to step 4.
8. Recursively, scan the neighborhood of p for other connected Skeleton points. Code non-predictable points with "0" or "1" accordingly, but use a different adaptive probability model than the one in step 6. After the whole connected component is scanned and coded, go to step 4.
9. If $n = 0$, then STOP.
10. $n \leftarrow (n-1)$. $Y_{n+1} \leftarrow (Z \oplus B)$. Go to step 3.

5. SIMULATION RESULTS

Two sets of simulation tests are presented in this section.

The first one compares, in terms of *lossless* compression efficiency, the proposed algorithm with some simple, well-known coding schemes for binary images. The test image is the 256×256-pixel "Tools" (Fig. 1), and the results, in bits-per-pixel, are presented in Table 1. According to it, the proposed Skeleton coder provides the best compression.

The second set of simulation tests examines the efficiency of the proposed Skeleton coder in coding

Coder	Bit-rate
Ziv-Lempel ("Compress" in Unix)	0.171
Run-length + Huffman	0.152
Quad-tree	0.131
Chain-Code	0.091
Skeleton (proposed)	**0.071**

Table 1: Lossless compression rates, in bpp, of the proposed Skeleton coder and other known schemes, for the image "Tools".

CCITT Images	G3D1	d_8 Skeleton	G3D2	**Proposed Skeleton**	G4	Progressive JBIG
#1	37423	28261	25967	**20405**	18103	16771
#2	34367	19058	19656	**12681**	10803	8933
#3	65034	49018	40797	**37535**	28706	23710
#4	108075	102848	81815	**82194**	69275	58656
#5	68317	52476	44157	**40259**	32222	28086
#6	51171	30658	28245	**24615**	16651	13455
#7	106420	112301	81465	**83398**	69282	60770
#8	62806	35965	33025	**24815**	19114	15227

Table 2: File sizes of compressed facsimile standard CCITT documents, obtained by the proposed Skeleton algorithm, compared to previous Skeleton-Based coder and existing standards.

scanned documents (fax), and compares it to existing standard coders. The previous Skeleton-Based scheme proposed in [3] (denoted d_8 Skeleton) is also compared. The eight CCITT facsimile standard test 2376×1728-pixel images, of documents scanned at 200 dpi, are *lossless* coded by the proposed algorithm. Table 2 compares the size of the obtained coded files with the results given in [7] and [3]. Comparison of our results to the d_8 Skeleton shows a substantial improvement in Skeleton-based Coding. At this point, it is still weaker than the most advanced Standards (G4 and JBIG), but it is comparable to the 2-dimensional Group 3 Standard (G3D2, with $k = 4$), being usually more efficient than it (with exception of the "hardest" images, #4 and #7).

Since the scanning in the algorithm is performed *on the boundaries* of the expanding set Z only, the coder and the decoder procedures are fast. On a Digital DECStation 5000, programmed in Standard C, coding of the 256×256-pixel image "tools" takes about 4 seconds, and its decoding about 2 seconds.

6. CONCLUSION

A new binary image coding scheme, based on the Morphological Skeleton Representation, is presented and compared to other well-known schemes and Standards.

The coding scheme is suitable for "progressive transmition", and takes into consideration new skeleton properties not used by previous Skeleton coding schemes.

The algorithm is shown to outperform a previous Skeleton coder presented in [3], and the Group 3 Standard coders, for facsimile scanned documents (with exception of dense images, where G3D2 gives slightly better results). The proposed algorithm does not achieve the compression rates obtained by Group 4 and JBIG algorithms, but improvements in the scheme and the effect of different structuring elements are still to be tested.

Since scanned documents are composed of thin graphic lines and text, the thinning effect of the Skeleton Representation is strongly reduced. For binary images having foreground objects with larger width, the proposed Skeleton coder is found to be superior to the well-known Ziv-Lempel, Run-length + Huffman, Quad-tree and Chain coders, and its complexity is on par with their complexity.

7. REFERENCES

[1] J. Serra, *Image Analysis and Mathematical Morphology, Vol.2: Theoretical Advances.* New York: Academic Press, 1988.

[2] P. Maragos and R.W. Schafer, "Morphological Skeleton Representation and Coding of Binary Images", *IEEE Trans. ASSP*, Vol. 34, No. 5, pp.1228-1244, October 1986.

[3] J.W. Brandt, A.K. Jain and V.R. Algazi, "Medial Axis Representation and Encoding of Scanned Documents", *JVCIR*, Vol. 2, No. 2, pp.151-165, June 1991.

[4] D. Schonfeld and J. Goutsias, "Morphological Representation of Discrete and Binary Images", *Trans. Signal Processing*, Vol. 39, No. 6, June 1991.

[5] R. Kresch and D. Malah, "Morphological Reduction of Skeleton Redundancy", *Signal Processing*, Vol. 38, pp. 143-151, August 1994.

[6] R. Kresch and D. Malah, "New Morphological Skeleton Properties Leading to its Efficient Coding", submitted to *IEEE Workshop on Non-Linear Signal and Image Processing 1995*, Halkidiki, Greece, June 1995.

[7] R. Aravind, G.L. Cash, D.L. Duttweiler, H.M. Hang, B.G. Haskell and A. Puri, "Image and Video Coding Standards", *AT&T Technical Journal*, pp.67-88, January/February 1993.