Morphological Image Representation for Coding Applications

Renato Kresch

Morphological Image Representation for Coding Applications

Research Thesis

Submitted in Partial Fulfillment of the Requirements
for the Degree of Doctor of Science

by

Renato Kresch

Submitted to the Senate of the Technion – Israel Institute of Technology
Sivan 5755 Haifa June 1995

The work described herein was supervised by Professor David Malah, and was carried out in the Faculty of Electrical Engineering, Signal and Image Processing lab.

I would like to thank *Prof. David Malah*, not only for his dedicated and thorough supervision, but also for his wise advice at moments of decision, his patient encouragement at moments of uncertainty and pressure, his positive involvement at moments of success, and his kind help whenever it was possible. I will not forget the warm and friendly support I received from him during the last five years. Thanks very much.

Thanks very much also for the staff of the Signal Processing Lab. for the efficient work and the friendly environment. Thanks for *Nimrod Peleg*, for the excellent organization and the warm relation, *Ziva Avni* for the technical support and the pleasant atmosphere, and to *Avi Rosen* for the "art lessons" and the sociability. Thanks very much to *Uri Mozer* for the continuous help and the friendship. Thanks also to *Yoram Or-Chen* for trusting and supporting me from the beginning.

Thanks to my colleagues *Amichai Amitai*, *Zohar Sivan*, and *Amir Shatz*, who accompanied me during my first years in the Technion. Thanks to *Zachi Baharav* for his cooperation, in science and sports. Thanks to my friends *Yair Karelic*, *Hagai Krupnik*, *Gilad Cohen*, *Ido Lev*, and *Ofer Reshef* for their friendship, support, and gastronomic cooperation.

I also would like to thank the *Liron family*, which "adopted" me as a son, and provided me with a family environment in Israel. Specially to *Shiri*, for her emotional involvement and support, and for all she means to me.

Finally, I thank my own family. Even though far away, they accompany me in all my steps.

The generous financial help of the Gutwirth family fund is gratefully acknowledged.

Contents

CONTENTS (Cont'd)

CONTENTS (Cont'd)

CONTENTS (Cont'd)

Abstract

This research addresses the representation of binary and grayscale images by means of Mathematical Morphology.

Mathematical Morphology is a relatively new non-linear theory for Image Processing, based on Set Theory. It considers images as *sets* (instead of *vectors*, as in the classical Linear Image Processing), which permits *geometry-oriented* transformations of the images. This approach seems very appropriate for dealing with objects in images, and it has gained increasing attention in recent years. It was first developed for binary images, then extended for grayscale images, and finally, generalized for sets in a generic mathematical space, called Complete Lattices.

One of the main image representations in Mathematical Morphology is the *Skeleton Representation*. Its original purpose was to provide a *symmetry axis* of planar shapes, for Pattern Recognition and Shape Analysis applications, but it can also be considered as a *shape decomposition*, useful for Image Compression, and Pattern Recognition.

In recent years, the Skeleton representation was generalized a number of times, to extend the scope of its algebraic characteristics as much as possible. With these generalizations, the Skeleton's role as a symmetry axis lost its importance, while its ability to serve as an efficient image decomposition tool was extended to discrete images, grayscale images, and sets in Complete Lattices.

This work follows the above line, and further develops it. First, a new evolutionary branch is added to the Skeleton's development, by the introduction of a *Multi-Structuring-Element Skeleton* (MSES), which permits a Skeleton-type decomposition of images into "multi-parameter" families of elements. Previously, Skeleton representations were based only on "1-parameter" families of elements for image decomposition. In addition, a *Generalized Skeleton* framework is proposed, unifying the new branch, as well as all the previous Skeleton generalizations. It deals generically with sets in Lattices, and families of decomposition elements indexed by indices in a generic set I, which is totally or partially ordered. Extension of the unified generalized framework to include *grayscale images* is also considered, and two approaches are proposed and compared.

As particular cases of the proposed Generalized Skeleton Representation, one can obtain new relevant representations, as well as well-known representations, like the Quadtree and the Bit-Plane decompositions, which were not previously recognized as morphological ones. Applications and properties of the Generalized Skeleton Representation are presented, and illustrated by computer simulations.

Another Morphological representation, which is not part of the above generalized framework, called *Two-Sided Skeleton*, is also proposed in this work. As opposed to the above

generalized framework, the Two-Sided Skeleton is a self-dual (almost) representation, since inverting the gray levels of the pixels in the image almost does not affect it. Motivation and applications are also presented.

The second main topic addressed in this research is the Redundancy in the Skeleton representation. It is well known that the Skeleton representation contains redundant points, which, if removed, do not affect the perfect reconstruction property of the Skeleton. Previously, all the redundant points were considered as belonging to a single group. In this work, a study on the types of redundancy is performed, and *redundancy categories* are proposed. Each redundant point is mathematically classified into one or more of the proposed categories. Furthermore, a *generic approach* for obtaining *Redundancy-Reduced Skeletons* is developed. By this approach, Reduced Skeletons which are free of redundant points from all but one of the proposed categories can be obtained by means of *morphological closed-form formulæ*.

Still concerning the redundancy in the Skeleton, a second approach is proposed for removing most of the redundant points from that category which the first approach is not able to deal with. This second approach is based on a generalization of the concept of "Convexity", which we call B-Convexity, proposed and developed in this thesis. B-Convexity is defined, keeping an analogy with the original concept of Convexity, and is studied by means of several theorems, forming a theory. This theory is applied to Redundancy Removal, as mentioned above.

Another main topic addressed in this research is *Skeleton-based Coding of binary and grayscale images*. In the last years this area has lost much of its interest, because unsatisfactory results, when compared to other classical coders. However, as demonstrated here, much of the poor coding performance by the previous coders is because they have neglected to take in consideration several correlation characteristics of the Skeleton. In this work, new theoretical properties of the Skeleton Representation, related to the above mentioned correlation, are presented. Furthermore, a Skeleton-based coding algorithm for binary and grayscale images is developed, which efficiently takes into consideration the above properties.

Computer simulations, also presented, show that, for binary images, the proposed coding scheme substantially improves the results obtained by previous Skeleton-based Coders, and performs better than classical coders. For facsilime images, it usually performs better than the Group 3 standard, but, at this point, it is weaker than the most advanced standards: Group 4 and JBIG.

For grayscale images, the proposed algorithm performs well for images containing large flat areas and abrupt edges, like multi-layer maps and images obtained by a segmentation process.

Finally, this work performs a *comparison* between *Linear* and *Morphological* Representations. The profound algebraic similarity, and the qualitative differences between the approaches are presented and analyzed. Hybrid representations, which combine both approaches, and their applications in Coding, are also considered.

List of Symbols and Abbreviations

$\mathcal{C}$	Set of Complex Numbers
$\mathcal{F}, \mathcal{G}$	Families of shapes
$\mathcal{L}$	Linear Space
$\mathcal{N}$	Set of Naturals
$\mathcal{O}$	The origin in an Euclidean space
$\mathcal{P}$	Complete Lattice
$\mathcal{P}(\cdot)$	Set of all subsets of $(\cdot)$
$\mathcal{P}(E)$	Boolean Lattice
$\mathcal{R}$	Set of Reals
$\mathcal{S}(\cdot)$	Region of Support of $(\cdot)$
$\mathcal{Z}$	Set of Integers
$\mathcal{E}^B(\cdot)$	Extreme Points of $(\cdot)$ w.r.t. the structuring element B
$\mathcal{I}^B(\cdot)$	Internal Points of $(\cdot)$ w.r.t. the structuring element B
$X, Y, A, B, \ldots$	Sets in an Euclidean Space or in a Complete Lattice
$A(n), B(n)$	Series of sets, indexed by n
X	Original image or set
X^c	The Complement of X
B	Structuring Element
B^s	The Symmetric of B
X_b	Translation of the set X to the point b
E	Euclidean Space or an arbitrary set
I	Arbitrary set of indices
S	Set of Skeleton points
S_n, S_i	Skeleton Subsets of order n, i
RS	Reduced Skeleton
$x, y, a, b, \ldots$	Points in an Euclidean Space or in a Boolean Lattice
i, j	Indices or integer numbers
n, m	Non-negative Integer numbers
λ, μ	Non-negative Real numbers
r	Radius of a disc (non-negative real)
$f, g, f(x), g(x)$	functions from an Euclidean space to the reals
$\oplus$	Translation invariant Dilation in an Euclidean space
$\ominus$	Translation invariant Erosion in an Euclidean space
$\circ$	Translation invariant Opening in an Euclidean space
$\bullet$	Translation invariant Closing in an Euclidean space

δ	Dilation in a Complete Lattice or Structuring Function in a Boolean Lattice
ε	Erosion in a Complete Lattice
γ	Opening in a Complete Lattice
ϕ	Closing in a Complete Lattice
ψ	Arbitrary operator
$\subseteq$	Contained or equal to
$\subset$	Contained
$\cup$	Union
$\cap$	Intersection
$\vee$	Supremum in a Complete Lattice
$\wedge$	Infimum in a Complete Lattice
$\leq$	Usual less or equal to, or order in a Complete Lattice
$CH(\cdot)$	Convex-Hull
$CH^B(\cdot)$	B-Convex-Hull
$U(\cdot)$	Umbra operator
$W(\cdot)$	Envelope operator
MSES	Multi-Structuring-Element Skeleton
JBIG	Standard Coding algorithm for binary still images
JPEG	Standard Coding algorithm for grayscale still images
s.e.	Structuring Element

Chapter 1

Introduction

1.1 Background and Motivation

Image Representation is a key component in many tasks in Computer Vision and Image Processing. It consists generally of presenting an image in a form, different from the original one, in which desired characteristics of the image are emphasized and more easily accessed.

For grayscale images, most of the known representation are based on *linear* methods: Unitary Transforms (Fourier, Wavelets, etc.), Multi-resolution pyramids, Linear Prediction, and so on. Fractal approaches has also been studied. For binary images, the classical methods are not derived from a unifying theory, as opposed to the grayscale methods. The best-known methods are: Contour representation, Quadtree Decomposition, Skeleton representation and Run Lengths.

In this thesis, we consider *morphological methods* for both binary and grayscale image representation. They are based on *Mathematical Morphology*, which is a relatively new, and rapidly growing, nonlinear theory for Image Processing.

Mathematical Morphology [51, 52, 12, 10, 38] is part of Set Theory, and it has a strong *geometric* orientation. Its theory was developed by Matheron and Serra, in the middle 60's, with the purpose of describing the structure of materials by image analysis of their sections. Being originally developed for binary images, it was later (during the 70's) generalized for grayscale images as well [57].

For binary images, Mathematical Morphology provides a well founded theory for analysis and processing. Therefore, Binary Morphological Representations can be developed and analyzed. For grayscale images, Mathematical Morphology yields a nonlinear method for geometrical processing of images. Grayscale Morphological Representations are a generalization of the binary representations, and they emphasize geometrical characteristics of the image, which are not easily accessed in a linear representation.

The main morphological representation for *binary images* is the *Skeleton* [27, 29, 52]. The Skeleton (defined in section 1.2 below) was originally proposed and developed independently of Mathematical Morphology, and several works concerning it are still performed today with no aid from the morphological theory (see [54]). On the other hand, it was proved that the Skeleton can be calculated entirely by the basic operations of Mathematical Morphology [27], which makes the Skeleton a morphological representation, suitable for analysis by mor-

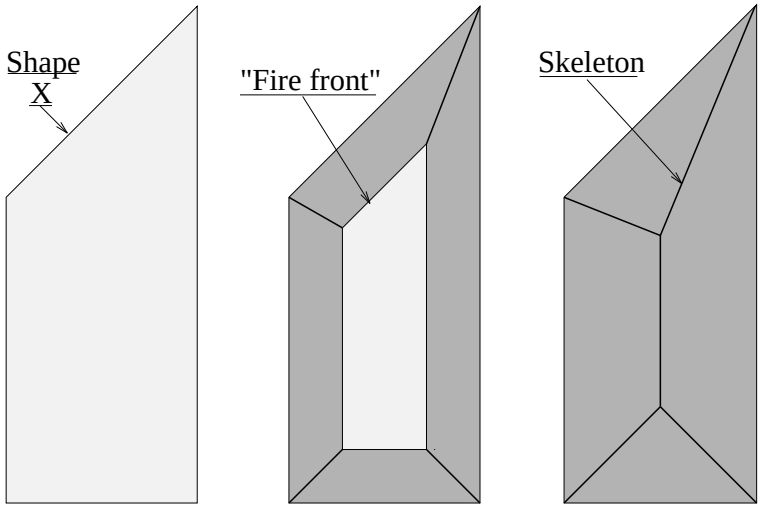


Figure 1.1: The "Grass-fire" model for obtaining the Skeleton.

phological tools. Moreover, following the generalization of the whole Morphological theory, from binary to grayscale images, the Skeleton Representation could be generalized as well to grayscale images [38].

The motivation of this work is to investigate the use of Morphological Skeletons for binary and grayscale Image Representation, and their applications, with special interest in Coding.

1.2 The Skeleton

1.2.1 Definitions

Blum [2] introduced the notion of Skeleton by means of the following intuitive model: Suppose a given shape to be a grass field, and suppose that at time $t = 0$ its whole boundary is set on fire. The fire then propagates inwards at a constant speed. The set of points at which the fire extinguishes is the Skeleton of the shape. Fig. 1.1 illustrates the above "Grass-fire" model.

Since its intuitive introduction, the Skeleton was defined mathematically in a number of ways. The various definitions are different characterizations of the "Grass-fire" model, and they provide (almost) equivalent results for continuous planar shapes. Two of the main definitions of Skeleton are the following [51]:

Definition 1 *Let a* maximal disc *inscribable in a given shape $X \subset \mathcal{R}^2$ be a disc included in X, and not contained in any other disc included in X.*

The Skeleton of X is the set of centers of all its maximal discs.

Definition 2 *Let the* distance function *$r(x)$ for a given shape $X \subset \mathcal{R}^2$ be the map relating to each point x inside X its distance to the boundary of X. Let $d(x, y)$ be the Euclidean metric in $\mathcal{R}^2$.*

The Skeleton of X is the set of points $\{s \in \mathcal{R}^2\}$ satisfying $r(y) - r(s) < d(s, y)$, for all $y \in \mathcal{R}^2$.

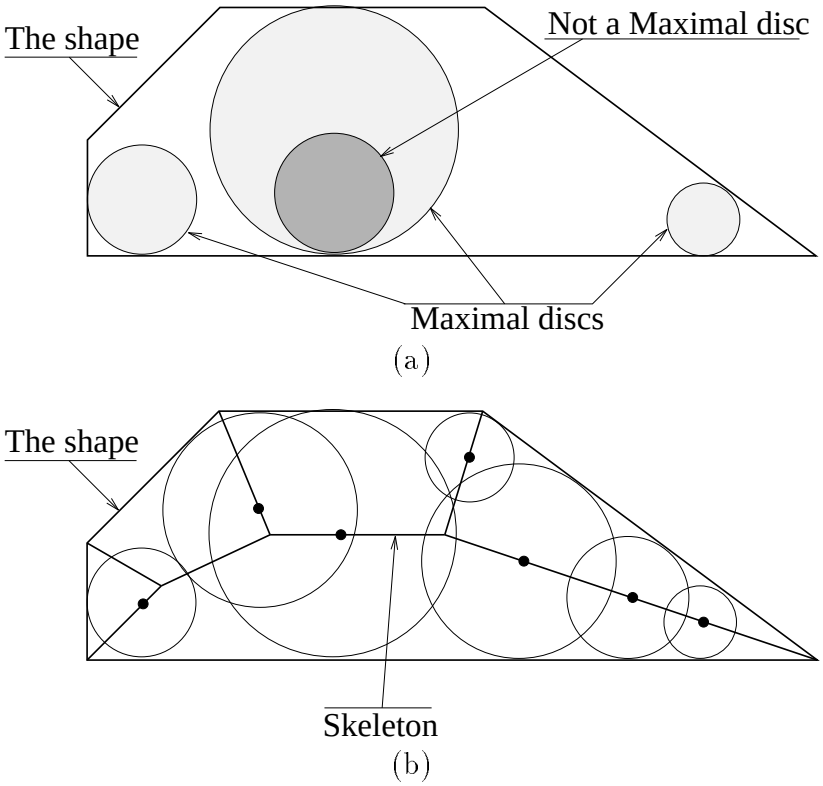


Figure 1.2: The definition of Skeleton in terms of *maximal discs*. (a) Maximal discs in a shape, (b) the Skeleton as the centers of all the maximal discs.

Fig. 1.2 and Fig. 1.3 illustrate the above definitions, respectively.

1.2.2 Algebraic Versus Topological Points of View

According to Serra [53], the study of Morphological Skeleton was historically split into two branches: *algebraic* and *topological*.

Topological Approach

The *topological* branch considers the Skeleton mainly as a *shape descriptor*. In this case, the Skeleton is supposed to provide a simplified version of the original shape, and to summarize important geometrical information about it. Therefore, the Skeleton's *shape* and *connectivity* (among other geometrical and topological properties) can be considered as relevant features for Image Analysis and Pattern Recognition.

From the topological and geometrical points of view, the most important properties of the Skeleton are:

1. It is *thin*, composed of lines and/or points.

2. It represents a *symmetry axis* (also called *medial axis*) of the original shape.

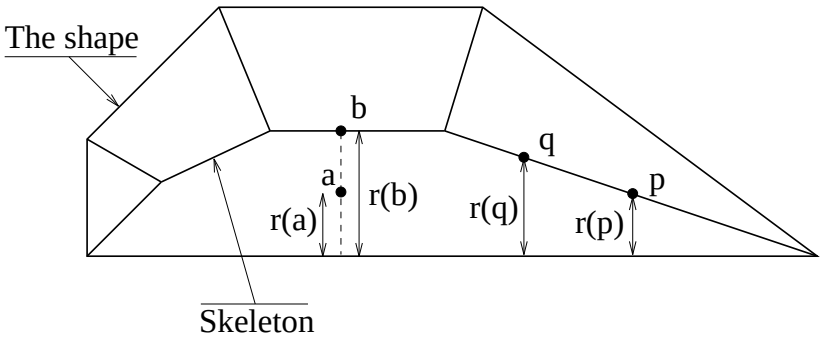


Figure 1.3: The definition of Skeleton in terms of *distance function*. The point a is not a Skeleton point since $r(b) - r(a) = d(a,b)$. On the other hand the points b, p, and q are Skeleton points. Notice that $r(q) - r(p) < d(p,q)$ as well as $r(p) - r(q) < 0 < d(p,q)$.

3. It *usually* preserves homotopy, that is, the number of connected components and the number of "holes".

4. The Skeleton points are disjoint, i.e., the same point in space cannot be the center of more than one maximal disc.

5. There are efficient algorithms for calculating the Skeleton.

On the other hand, the Skeleton presents also some negative topological characteristics:

- Small perturbations on the boundary of the original shape X can produce large branches in the Skeleton, and a very small hole in the shape can considerably alter the Skeleton. In this sense, skeletonization is not a continuous operation.
- Connectivity preservation is not always assured.

Among the main issues addressed by researchers in the topological branch are:

1. How to produce more robust Skeletons, less influenced by small perturbations?

2. How to produce Skeletons where connectivity preservation is assured?

3. How to produce *discrete* Skeletons in grids, so that homotopy, axial symmetry and the thin aspect are preserved?

Algebraic Approach

The *algebraic* branch relates to the Skeleton in a quite different way. From the algebraic point of view, the Skeleton is the result of the *decomposition* of a given set into the superposition of simpler elements, selected from a pre-defined family of elements (discs of increasing sizes). The above decomposition provides an image representation which can be used in Coding

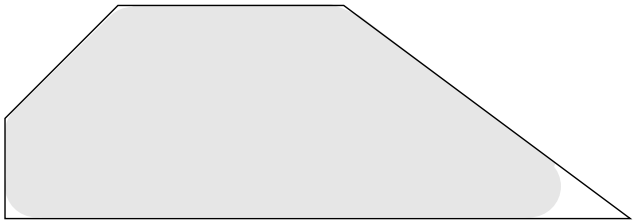


Figure 1.4: Partial Reconstruction of the Skeleton Representation. Simplification of the shape is obtained by removing Skeleton points related to maximal discs with value smaller than a threshold.

for data compression purposes. The algebraic approach also yields another tool for Pattern Recognition.

The important Skeleton characteristics in the algebraic approach, as opposed to those in 1.2.2, are:

1. The original image is fully represented by the collection of Skeleton points, together with the radius of the related maximal discs (or equivalently, the value of the distance function at each Skeleton Point). The shape reconstruction is obtained by the union of all the maximal discs. The above collection of Skeleton points and radii is called the *Skeleton Representation*.

2. The Skeleton Representation provides simplified versions of the original shape when Skeleton points with radius smaller than a threshold are discarded in the reconstruction process (see Fig. 1.4).

3. The Skeleton provides a decomposition of the original shape into features (discs) of different sizes, which can be seen as components in different "scales". The smallest maximal discs can often be considered as details, whereas the largest ones can often be considered as the main structure. This provides a *hierarchical* or *pyramidal* interpretation to the Skeleton Representation.

4. The Skeleton Decomposition can be calculated by means of an algebraic closed-form formula (see section 2.2).

The main negative characteristics of the Skeleton, in this branch, are:

- It usually contains *redundant points*, that is, many Skeleton Points can be discarded and still the original shape can be fully reconstructed. (See Fig. 1.5).

- It is not a self-dual representation (like the Chain-code or Quadtree, for instance), since the Skeleton of X^c (the complement of X) is totally different from the Skeleton of X[1].

Some of the main issues addressed by researchers in the algebraic branch are:

[1]See [15] for background on self-dual operators.

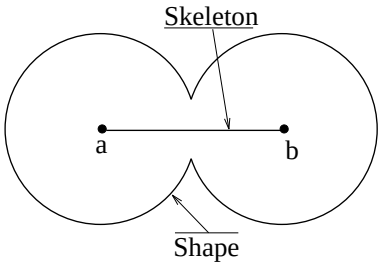


Figure 1.5: Skeleton Redundancy. Only the points a and b are not redundant in this Skeleton Representation.

1. How to reduce Skeleton's redundancy?
2. How to decompose the shape into elements other than discs?
3. How to produce *discrete* Skeletons on grids, containing as few as possible points?
4. How to efficiently code the Skeleton Representation of shapes?

1.3 A Generic Approach to Skeleton-Based Coding

1.3.1 Coding as an Optimization Problem

Consider the following optimization problem, illustrated in Fig. 1.6:

Problem 1 *Let $\{\mathcal{F}_i\}$ be a given family of "simple" shapes (e.g., squares of different sizes, discs of different radii), with indices i belonging to a set I (e.g., $I = \{0, 1, 2, \ldots\}$).*

For any given shape X, what is the smallest subset of $\{\mathcal{F}_i\}$ which exactly covers X?

By solving Problem 1 (if there is a solution), one is usually calculating an *efficient lossless representation* of X in terms of a set of indices $I(X)$ contained in I. Coding $I(X)$ typically leads to a *compression* of X. Decoding, on the other hand, can be performed by superposing those elements of $\{\mathcal{F}_i\}$ which are indicated in $I(X)$.

Unfortunately, a simple, closed-form solution for Problem 1, supposing an arbitrary family $\{\mathcal{F}_i\}$, is not expected to exist. In this case, high-complexity optimization algorithms are required.

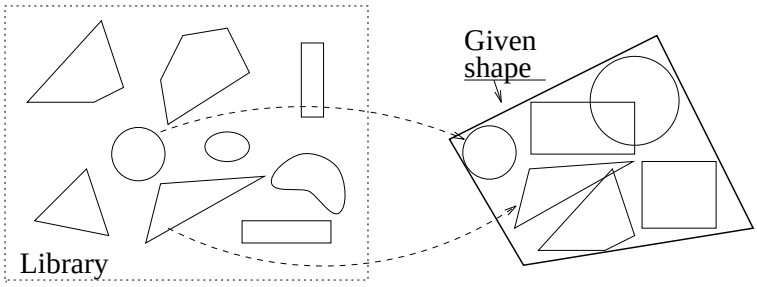


Figure 1.6: Shape representation by the union of elements from a given family of shapes.

1.3.2 The Algebraic Skeleton as a Sub-Optimal Solution

In this context, the Skeleton Decomposition, seen from the algebraic point of view, can be considered as a *"low-complexity"*, *sub-optimal* solution to Problem 1, in the particular case of $\{\mathcal{F}_i\}$ being equal to the family of all discs. The "low-complexity" is due to the existence of a closed-form formula for the Skeleton calculation. And it is a *sub-optimal* solution because it contains redundancy (see previous discussion, and chapter 6).

The algebraic approach serves as a framework to Skeleton-Based Coding of binary images. In this framework, topological issues or negative characteristics concerning the Skeleton (like lack of connectivity preservation, discontinuity of the operation, etc.) are of little importance (or not at all important), as long as coding efficiency is not affected.

1.4 Previous Works in Coding

In recent years, the algebraic branch of the Skeleton development has brought the Skeleton closer to be the optimal solution of Problem 1. This development can be observed in two main fronts:

1. Generalizations of the Skeleton framework, so that families $\{\mathcal{F}_i\}$ other than the family of discs can be used [51, 29, 48, 50, 52, 53, 21]. This generalization development is detailed in chapter 3.

2. Approaches to remove the redundant elements in the Skeleton representation, so that it becomes an actual (locally) *optimal* solution to Problem 1 (for the specific families $\{\mathcal{F}_i\}$ mentioned in item 1 above), [29, 30, 20]. This development is described in chapter 6.

Although the *structure* of the Skeleton has been extensively studied, only a small number of works address to *coding schemes* for the Skeleton. The only articles known to us, which seriously propose a scheme to code the Skeleton Representation are [29, 3]. The schemes proposed by them are described in general lines in chapter 8.

On the other hand, the above coding schemes are related to *binary Skeletons* only, i.e., Skeletons of binary images. Although *grayscale Skeleton Representations* are known for about 15 years [36, 38], we have not seen any work proposing and analyzing a coding scheme for them. Skeleton-based coding of *grayscale* images has been performed by first decomposing the image into binary images (bit-plane decomposition [48], or segmentation [39, 44]), and then coding their binary Skeletons.

Other Morphological Approaches

Another morphological representation method, for binary images, is called *Morphological Shape Decomposition* [37, 38, 41]. It consists of first calculating the discs (or other predefined convex shapes) with the greatest size contained inside the shape, then taking the residue (set-difference) between the original shape to the above greatest discs, and finally reiterating the above procedure on the residue until the whole shape is decomposed. The resulting decomposition elements are, therefore, disjoint.

The Morphological Shape Decomposition shares with the Morphological Skeleton Representation the attention of the researchers. The question of which of the representations is preferable for coding are yet to be answered. In [40], a comparison between the two methods is performed, but it does not present real coding results.

For grayscale images, the most popular image representation for coding is the *Morphological Pyramid* [59, 58, 11, 35]. The Morphological Pyramid decomposes a grayscale image into different "resolution" levels, where "resolution" in this case is related to "size", as opposed to linear pyramids where "resolution" relates to "frequency" or "scale". The Morphological Pyramid can be obtained with the same algorithm as used for calculating the linear Laplacian Pyramid [5]; the only difference is that the low pass filters in the latter are replaced by morphological filters in the former.

1.5 Original Contributions and Organization of the Thesis

The following are the main contributions of this thesis:

1. Concerning the Skeleton's algebraic framework:
   (a) A new branch is added to the evolutionary tree with the proposal of a *Multi-Structuring-Element Skeleton* (MSES).
   (b) A Generalized Skeleton's framework is proposed, unifying the new branch and the previous general framework.
   (c) The Quadtree and the Bit-Plane decompositions are shown to be particular cases of the proposed Generalized Skeleton Representation.
   (d) New particular cases of the Generalized Skeleton representation of binary and grayscale images are proposed, including multi-parameter generalizations of the Quadtree and the Bit-Plane decompositions.
   (e) An "almost self-dual" Skeleton representation (for which an image and its *inverse*[2] are similarly represented) is developed.
2. Concerning the Skeleton's redundancy:
   (a) A classification of the redundant points in the Skeleton into categories is proposed.
   (b) A generic approach for obtaining Redundancy-Reduced Skeletons is proposed, and *Morphological* closed formulæ for obtaining Skeletons with no redundant points in most of the above categories are derived.
   (c) The concept of B-Convexity, generalizing the concept of Convexity, is proposed and its theory developed. B-Convexity is then applied to Skeleton redundancy reduction.

[2] The inverse of a grayscale image $f(x, y)$ is considered here to be the image $g(x, y) = 255 - f(x, y)$.

(d) Morphological closed-formulæ are developed for the calculation of the *Essential Points* of the Skeleton [46, 48], which are those points in the Skeleton which cannot be discarded if a perfect reconstruction is desired.

3. Concerning Skeleton-based Coding:

   (a) New theoretical properties of the Skeleton, applicable to Coding, are proved. According to the first property, the radius of most of the Skeleton points *can be removed* from a Discrete Skeleton Representation, and still the original image can be completely recovered. The second property permits *deterministic prediction* of information about the Skeleton points of radius r, from the information about the Skeleton points with radius greater than r.

   (b) A Skeleton-based coding scheme for binary and grayscale images, using the above theoretical properties, is proposed and compared to other standard coders. The use of the scheme in segmentation-based coding [26, 44] is also considered and demonstrated.

The thesis is organized as follows:

Chapter 2 provides a theoretical background on Mathematical Morphology, and the Morphological Skeleton. It also describes the generalizations of Morphology, from planar shapes, through functions, up to elements in a mathematical generic framework called Lattice.

Chapters 3,4, and 5, as well as Appendix D, concern the Skeleton's framework and its generalization: Chapter 3 describes the evolution of the algebraic framework representation, and our contributions to this evolution. Chapter 4 considers Grayscale Skeletons, as further generalizations of the framework. Chapter 5 presents special particular cases of the Generalized Skeleton Representation, and its applications. In Appendix D, the definition, applications, and simulation results for the "almost self-dual" Skeleton, called *Two-Sided Skeleton*, is presented.

Chapter 6, 7, and Appendix B, concern Skeleton's redundancy reduction: Chapter 6 presents the classification of the skeleton points into categories, and one of two approaches for redundancy reduction. Chapter 7 summarizes the B-Convexity Theory, and its application as a second approach for redundancy reduction. The details of B-Convexity Theory is presented in Appendix B.

Chapter 8 concerns Skeleton-based Coding of binary and grayscale images, and our contributions to the field. Simulation results are also presented, and are compared to standard approaches. In Appendix C, mathematical generalizations of the above results are described.

Chapter 9 compares linear methods with morphological methods for grayscale image representation. Moreover, hybrid methods, combining the morphological with the linear approaches, are considered. In this context, the application of the proposed coding scheme (presented in chapter 8) in segmentation-based coding of grayscale images is demonstrated.

Finally, chapter 10 provides conclusions and proposes future research lines. Detailed proofs of some of the theorems presented in the work can be found in Appendix A.

Chapter 2

Theoretical Background on Mathematical Morphology

2.1 Binary Morphology

Mathematical Morphology is a nonlinear Image Processing theory, which was initially developed for *binary images* (see [51, 12, 38, 10]). In Morphology, a binary image is interpreted as a *set* X in an Euclidean space E. The elements of X are the foreground points of the image, whereas the background points form the set's complement X^c. The Euclidean space, E, is equal to either $\mathcal{R}^d$, if the image is continuous, or equal to $\mathcal{Z}^d$, if the image is discrete. Usually the binary images are bi-dimensional, and therefore $d = 2$, but multi-dimensional volumes can also be considered as images.

This original framework of Mathematical Morphology is called here Binary Morphology, to distinguish it from its generalizations, which were developed later, and are described in the sequel.

2.1.1 Basic Morphological Operations

In Binary Morphology, a binary image is processed by interacting with it via a pre-defined "simple" shape B, also considered as a set, called *structuring element*. For instance, B can be an open disc in $\mathcal{R}^2$. The basic morphological operations concern, usually, the interaction between a given image X, and a structuring element B.

Translation

Before the basic morphological operations are presented, the concept of *translation*, which is fundamental in binary morphology (and in general Morphology as well), must be properly defined.

Let B be a set contained in E, and let x be a point in E. The translation of the set B by the point x, denoted B_x, is defined as follows:

$$B_x \triangleq \{b + x \mid \ b \in B\} \tag{2.1}$$

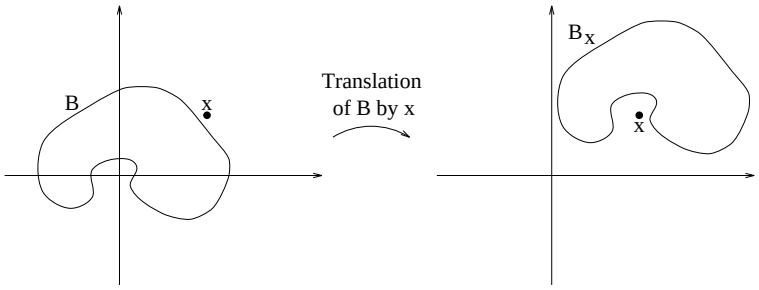


Figure 2.1: Translation of a set B by the point x. If we considered B "centered" at the origin, then after the translation, B_x is centered at x.

If B is a disc, or a square, centered at the origin, then B_x is centered at x. From this point on, we denote the origin of E as the "center" of the set B, even if B has no geometrical center, and even if the origin is not contained in B. Therefore, in the same way, B_x is said to be "centered" at x (see Fig. 2.1).

Dilation

The *Dilation* of the image X by the structuring element B, $X \oplus B$, is defined by:

$$X \oplus B \triangleq \bigcup_{x \in X} B_x \tag{2.2}$$

In words, the Dilation is obtained by centering the structuring element at each point x in X, and then taking the union.

If B is a connected shape, containing the origin, then the Dilation adds to the image X a "layer" around it (see Fig. 2.2(a)). The width of the "layer", and its shape, are determined by the structuring element's characteristics.

The most important properties of the Dilation operation are:

1. Dilation is *distributive with the union*, that is, for any sets A, B, and C in the Euclidean space E:
$$(A \cup B) \oplus C = (A \oplus C) \cup (B \oplus C). \tag{2.3}$$

2. Dilation is an *increasing* operation:
$$A \subseteq B \Rightarrow (A \oplus C) \subseteq (B \oplus C), \quad \forall A, B, C \tag{2.4}$$

Moreover, Dilation is *commutative* and *associative*, i.e.:

$$A \oplus B = \{a + b \mid \ a \in A, b \in B\} = B \oplus A \tag{2.5}$$
$$(A \oplus B) \oplus C = A \oplus (B \oplus C) \tag{2.6}$$

And, if the origin belongs to B, then Dilation is *extensive*:

$$X \oplus B \supseteq X \tag{2.7}$$

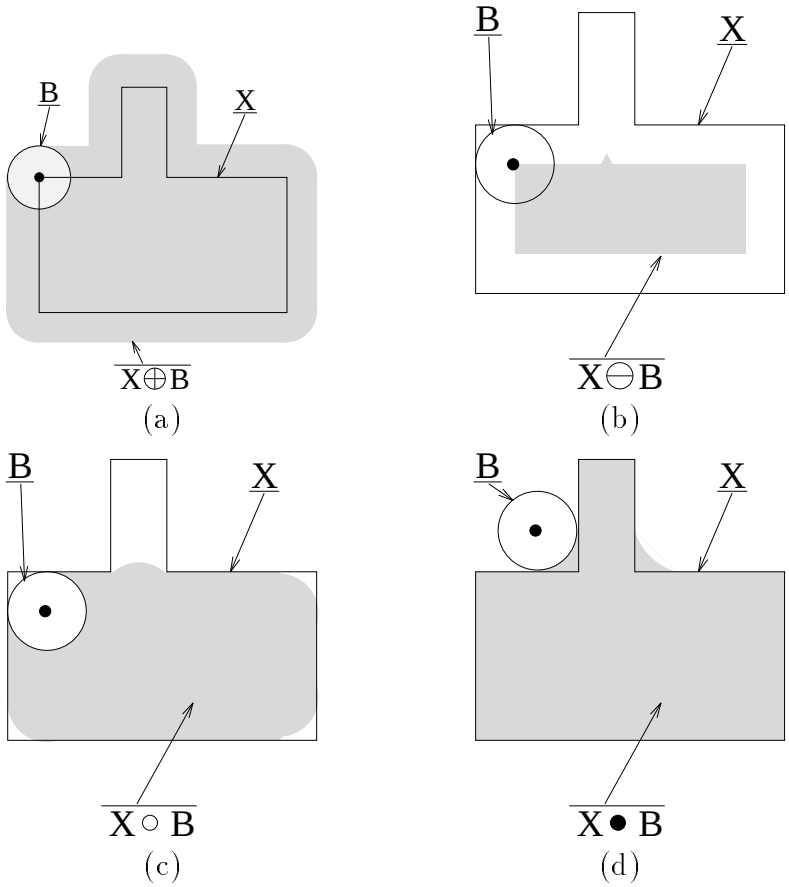


Figure 2.2: The basic operations of Binary Morphology. (a) Dilation, (b) Erosion, (c) Opening, and (d) Closing.

Erosion

The *Erosion* of X by B, denoted $X \ominus B$, is defined in the following way:

$$X \ominus B \stackrel{\triangle}{=} \bigcap_{b \in B} X_{-b} \tag{2.8}$$

Erosion is the dual operation to Dilation, because applying an Erosion to a set is the same as applying a Dilation to its complement, in the following way:

$$X \ominus B^s = (X^c \oplus B)^c \tag{2.9}$$

where B^s is the *symmetric* of B, defined by $B^s \stackrel{\triangle}{=} \{-b \mid b \in B\}$.

Erosion can be also characterized in the following way:

$$X \ominus B = \{x \in E \mid B_x \subseteq X\} \tag{2.10}$$

This means that the Erosion of X by B is the set of points at which B can be "centered", and be totally contained in X.

If B is a connected set, containing the origin, then Erosion *removes* a "layer" from X (see Fig. 2.2(b)). Again, the *width* and the *form* of the layer are determined by B.

The most important properties of the Erosion operation are:

1. It is *distributive with the intersection*, that is:

$$(A \cap B) \ominus C = (A \ominus C) \cap (B \ominus C), \forall A, B, C. \tag{2.11}$$

2. It is an *increasing* operator, that is:

$$A \subseteq B \Rightarrow (A \ominus C) \subseteq (B \ominus C), \forall A, B, C. \tag{2.12}$$

As opposed to Dilation, Erosion is neither commutative, nor associative. It satisfies instead:

$$(A \ominus B) \ominus C = A \ominus (B \oplus C) \tag{2.13}$$

If B contains the origin, then Erosion is *anti-extensive*:

$$X \ominus B \subseteq X \tag{2.14}$$

Opening and Closing

Although being dual, Dilation and Erosion are not *inverse* operation of each other, that is, generally:

$$(X \ominus B) \oplus B \neq X, \tag{2.15}$$

$$(X \oplus B) \ominus B \neq X. \tag{2.16}$$

Actually, neither Dilation, nor Erosion, have an inverse. Both are operations which usually remove part of the information in the image, which cannot be restored.

On the other hand, the above compositions of Dilations and Erosions lead to two other morphological basic operations, which are called *opening* and *closing*. The *opening* of X by B, denoted $X \circ B$, and the *closing* of X by B, denoted $X \bullet B$, are defined, respectively, by:

$$X \circ B \triangleq (X \ominus B) \oplus B, \tag{2.17}$$

$$X \bullet B \triangleq (X \oplus B) \ominus B. \tag{2.18}$$

Opening can be characterized also by the following relation:

$$X \circ B = \bigcup \{B_y \mid \ y \in E : B_y \subseteq X\} \tag{2.19}$$

This means that the opening of X by B is the set of points in X which are contained in some translation of B, totally included in X, (see Fig. 2.2). It usually removes details of the foreground of the image, which do not fit inside the structuring element. This makes the Opening a non-linear binary *filter*, which removes, from the *foreground*, features "smaller" than a certain "size", where "size" is determined by the structuring element.

The most important properties of Opening are:

1. It is *idempotent*, i.e.,
$$(X \circ B) \circ B = X \circ B \tag{2.20}$$
2. It is always *anti-extensive*, regardless to whether the origin is or is not contained in B.
3. It is *increasing*.

Closing is dual to Opening, in the same sense as Dilation is dual to Erosion, i.e.:

$$X \bullet B^s = (X^c \circ B)^c \tag{2.21}$$

Therefore, Closing can be seen as the set of points which are not contained in a translation of B^s, totally included in X^c (see Fig. 2.2(c)). It usually closes "holes" or thin background features, which do not fit inside the symmetric of the structuring element. This makes the Closing also a non-linear binary *filter*, which removes, from the *background*, features "smaller" than a certain "size", where "size" is determined by the structuring element.

The most important properties of Closing are:

1. It is *idempotent*.
2. It is always *extensive*, regardless to whether the origin is or is not contained in B.
3. It is *increasing*.

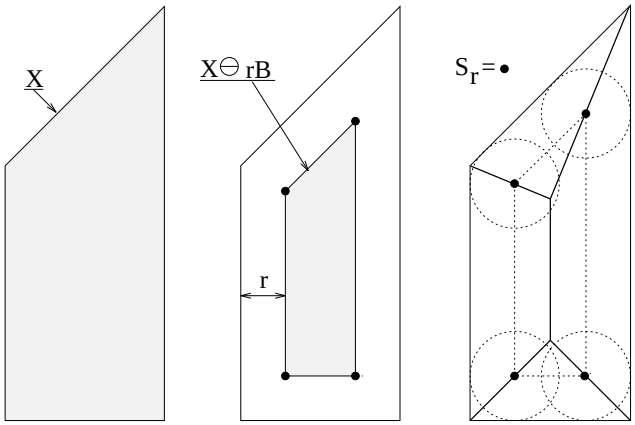


Figure 2.3: Skeleton calculation by morphological operations. The Skeleton points are the "vertices" of the regions $X \ominus rB$, for $r > 0$.

2.2 Skeleton Computation Via Morphological Operations

2.2.1 Lantuéjoul's Formula

In [27], Lantuéjoul proved that the Skeleton $S(X)$ of a *topologically open* shape X in $\mathcal{R}^2$ can be calculated by means of binary morphological operations, in the following way:

$$S(X) = \bigcup_{r>0} S_r(X) = \bigcup_{r>0} \left\{ X \ominus rB - \bigcup_{\Delta r>0} [(X \ominus rB) \circ \Delta rB] \right\} \tag{2.22}$$

where $S_r(X)$, for $r > 0$, is the set of maximal discs of radius r, and rB and ΔrB are, respectively, the topologically open discs with radii r, and the topological closed disc with radius Δr, centered at the origin.

In Lantuéjoul's Formula (2.22), the set $X \ominus rB$ represents the portion of the "grass field" not yet burned by the fire, at time $t = r$, in the "Grass-fire" model for the Skeleton (see section 1.2.1). By increasing r with positive values, one simulates the "fire propagation". The set $\bigcup_{\Delta r>0}[(X \ominus rB) \circ \Delta rB]$ represents the points at which the fire does *not* extinguish at time $t = r$, and therefore, the difference between the above sets provide the Skeleton points at $t = r$.

Lantuéjoul's Formula can be easier understood with the following simplification: Noting that the union $\bigcup_{\Delta r>0}$ in (2.22) acts here actually as a $\lim_{\Delta r \to 0}$, one can write the (informal) equation:

$$S_r(X) = X \ominus rB - [(X \ominus rB) \circ drB] \tag{2.23}$$

where drB denotes an open disc with infinitesimal radius dr. The opening by a disc with infinitesimal radius excludes from a shape its boundary points with infinite curvature (the "vertices"). Therefore, the Skeleton points, with radius r, of a shape X are the "vertices" of its eroded version $X \ominus rB$ (see Fig. 2.3).

The sets $\{S_r\}$ are called the *Skeleton Subsets*, and the function $q(s)$ relating to each Skeleton point s the radius of the respective maximal disc is called *Quench Function*.

2.2.2 Reconstruction and Representation

From the collection of Skeleton subsets $\{S_r(X)\}_{r>0}$ one can *perfectly* reconstruct the original shape X in the following way:

$$X = \bigcup_{r>0} S_r(X) \oplus rB \tag{2.24}$$

In other words, the union of all the maximal discs (centered at the points in $S_r(X)$, for each $r > 0$) equals the original image.

Equation (2.24) means that the collection of Skeleton subsets can be considered as a *shape representation*.

The Skeleton Representation permits also *partial reconstructions*, yielding simplified versions of the original shape. This is obtained by:

$$X \circ kB = \bigcup_{r>k} S_r(X) \oplus rB \tag{2.25}$$

Note that $X \circ kB$ is a smooth version of X, and that it was obtained by discarding the Skeleton subsets with radii smaller and equal to k. Moreover, (2.24) is obtained from (2.25) by setting $k = 0$.

2.3 Grayscale Morphology

In the late 70's, a number of approaches were proposed to generalize the binary morphological operations for grayscale images (considered as functions from E to $\mathcal{R}$) [51, 57, 42].

The most primitive of these approaches, proposed by Serra in 1975, considered the thresholded binary versions of the given image, for all possible threshold values, and the application of binary morphology to each one of those binary images separately. Since the binary morphological operations are increasing, the result of the operation on the thresholded images can be piled back to form a function. This approach, described in [51, pages 429-434], is not considered here.

The two other approaches were proposed by Sternberg [57], and we call them in this thesis the *umbra approach*, and the *sup-inf approach*, respectively. The latter one can be seen also as a morphological formalization of the approach independently developed by Rosenfeld [42].

2.3.1 The Umbra Approach

Let f be a function from E to $\mathcal{R}$. The *umbra* of f, denoted $U(f)$, is defined as:

$$U(f) \stackrel{\triangle}{=} \{(x,t) \in E \times \mathcal{R} \mid \ f(x) \leq t\}. \tag{2.26}$$

If f is a surface, representing a 2-D image, then its umbra is the volume below the surface (see Fig. 2.4).

The umbra is actually a *binary shape* in the Euclidean space $E \times \mathcal{R}$. Therefore, it can be operated upon by Binary Morphology, as described in section 2.1. The result of a binary

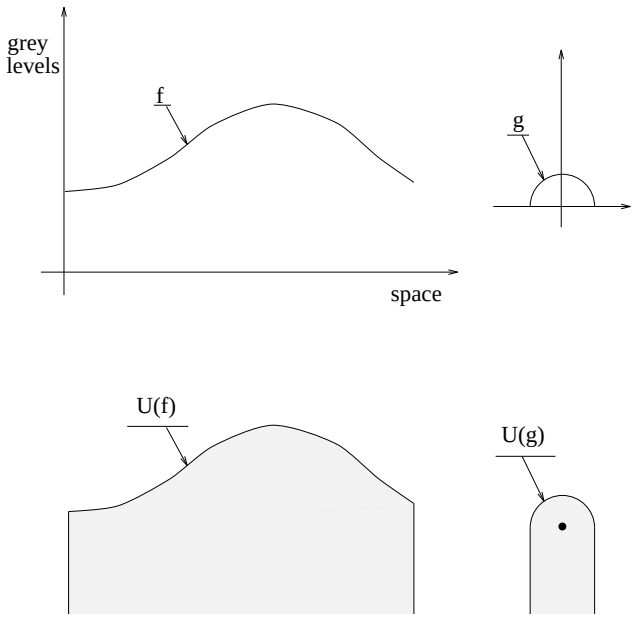


Figure 2.4: The umbra of two functions, f and g.

morphological operation between umbræ is also an umbra, and this is transformed back into a function by the following operator:

$$[W(U)](x) \stackrel{\triangle}{=} \bigvee\{t \mid \ (x,t) \in U\} \tag{2.27}$$

where U is an umbra, and $\vee$ denotes *set-supremum*. The above operator W returns a function, which is the *upper envelope* of the umbra.

For example, the Dilation of a function f by a structuring element g (which is also a function) is given by:

$$f \oplus g = W[U(f) \oplus U(g)] \tag{2.28}$$

This is illustrated in Fig. 2.5.

Similarly, Erosion of functions is obtained by:

$$f \ominus g = W[U(f) \ominus U(g)] \tag{2.29}$$

As in Binary Morphology, opening and closing are given by:

$$f \circ g = (f \ominus g) \oplus g \tag{2.30}$$

$$f \bullet g = (f \oplus g) \ominus g \tag{2.31}$$

2.3.2 The Sup-Inf Approach

The Sup-Inf approach consists of a direct transposition of (2.28) and (2.29) into an algebraic form.

Equations (2.28) and (2.29) can be written in the following form:

$$[f \oplus g](x) = \bigvee_{y \in E}[f(y) + g(x-y)] \tag{2.32}$$

$$[f \ominus g](x) = \bigwedge_{y \in E}[f(y) - g(y-x)] \tag{2.33}$$

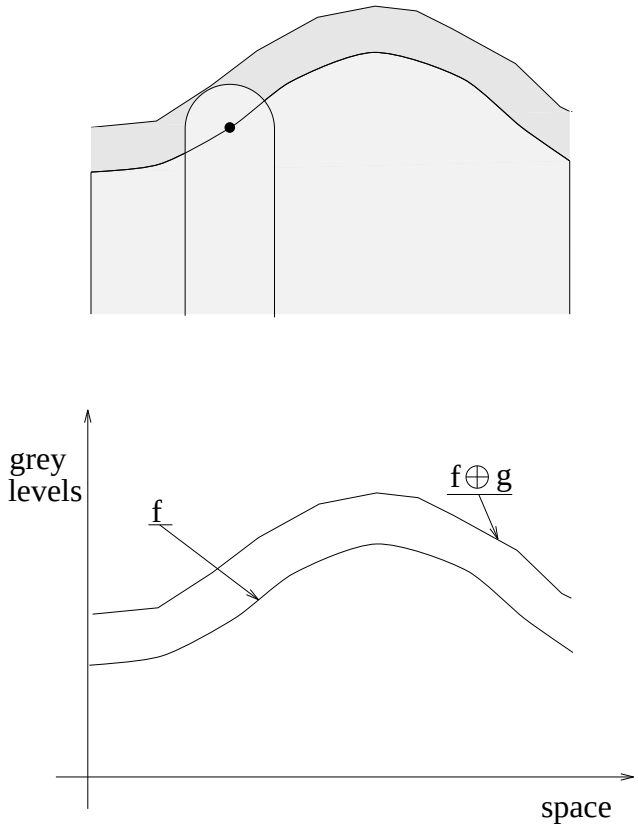


Figure 2.5: Grayscale Dilation by binary Dilation of umbræ. The functions to be dilated are those presented in Fig. 2.4.

where $\wedge$ denote *set-infimum*. These equations permit the implementation of Grayscale Morphology directly on functions, without the need to work in the higher-dimension umbra-domain.

Usually, the structuring element g relates to a shape of finite support (a square or a disc, for instance). In this case, we set $g(x) = -\infty$ for the points x outside the support. Similar consideration is taken for f, if it has a finite support.

If indeed, g has a finite support, then only the points of f that are inside the translated version of the window, $g(x-y)$, $y \in E$, are considered in the computation of the Dilation of f by g at the point x. Therefore, the region of support of g can be considered as a moving "window", inside which the operation is performed for each point x in space. A similar conclusion can be achieved for the Erosion; but this time the *symmetric* of the region of support of g is the moving "window".

Very often, the structuring element g is a *flat function*, i.e., it has a constant null value inside the region of support. In this case, (2.32) and (2.33) assume the following simplified form:

$$[f \oplus g](x) = \bigvee_{y \in \mathrm{S}[g(x-y)]} f(y) \tag{2.34}$$

$$[f \ominus g](x) = \bigwedge_{y \in \mathrm{S}[g(y-x)]} f(y) \tag{2.35}$$

where $\mathcal{S}[\cdot]$ returns the region of support.

Grayscale Morphology with flat structuring elements is very popular, not only due to its simple implementation, but also because it usually preserves edge contrast.

Opening and Closing, as before, are obtained by the appropriate compositions of Dilation and Erosion.

Fig. 2.6 show an example of applying the basic grayscale morphological operations on the 256×256-pixel image "Lena". The structuring element used here is a flat 7×7-pixel square.

2.4 Morphology on Complete Lattices

In late 80's, Serra generalized the whole framework of Mathematical Morphology, so that the generalized framework includes both Binary and Grayscale Morphology, and other new particular cases. This generalization is extensively described in [52]. The material in this section is a summary of chapter 1 of [52].

Instead of being restricted to Euclidean spaces, or functions from Euclidean spaces to the real axis, the generalized framework is based on generic mathematical spaces called *Complete Lattices*.

2.4.1 Complete Lattices

A Lattice is, by definition, a set $\mathcal{P}$, in which are defined a *supremum* operation (denoted generically by $\vee$), and an *infimum* operation (denoted generically by $\wedge$), satisfying for any elements $X, Y, Z \in \mathcal{P}$:

(a)

(b)

(c)

(d)

(e)

Figure 2.6: Example of Grayscale morphological operations. (a) Original image. (b) Dilation, (c) Erosion, (d) Closing, and (e) Opening.

1. Commutativity:

$$X \vee Y = Y \vee X, \quad X \wedge Y = Y \wedge X. \tag{2.36}$$

2. Associativity:

$$(X \vee Y) \vee Z = X \vee (Y \vee Z), \quad (X \wedge Y) \wedge Z = X \wedge (Y \wedge Z). \tag{2.37}$$

3. The law of absorption:

$$X \wedge (X \vee Y) = X, \quad X \vee (X \wedge Y) = X. \tag{2.38}$$

Every Lattice has a relation of *order* (generically denoted by $\leq$), defined in the following way:

$$X \leq Y \Leftrightarrow X \wedge Y = X, \quad X \geq Y \Leftrightarrow X \vee Y = X. \tag{2.39}$$

The Lattice $\mathcal{P}$ is called a *Complete Lattice*, if for any family of elements $\{X_i\}_{i \in I}$ in $\mathcal{P}$, I being a finite or infinite set of indices, the supremum and the infimum are in $\mathcal{P}$, i.e.:

$$\bigvee_{i \in I} X_i \in \mathcal{P} \tag{2.40}$$

$$\bigwedge_{i \in I} X_i \in \mathcal{P} \tag{2.41}$$

In a Complete Lattice, there exist two elements, $\emptyset$ and U (called respectively the "null" element and the "universe"), such that, for any $X \in \mathcal{P}$:

$$X \wedge \emptyset = \emptyset, \quad X \vee U = U \tag{2.42}$$

Notice that there is no direct relation between the above defined "Lattice", with the notion of lattice, used in Signal Processing as a grid of points. Actually, as mentioned below, a *continuous space* (e.g., $\mathcal{R}$) can be a Lattice.

2.4.2 Examples of Complete Lattices

Complete Lattice of sets in an Euclidean space

Let $\mathcal{P}(\cdot)$ denote the operator which returns the set of all subsets of a given set. For example:

$$\mathcal{P}(\{a, b, c\}) = \{\emptyset, \{a\}, \{b\}, \{c\}, \{a, b\}, \{b, c\}, \{a, c\}, \{a, b, c\}\} \tag{2.43}$$

Let us consider the set of all subsets of an Euclidean space E, i.e., $\mathcal{P}(E)$. This set is a Complete Lattice if we choose its supremum and infimum operations to be, respectively, the *union* and the *intersection*. The induced ordering in this case is the *inclusion*. In this case, the null element and the universe are, respectively, the empty set ($\emptyset$), and the Euclidean space itself (E).

The above Complete Lattice is actually the basic framework of Binary Morphology.

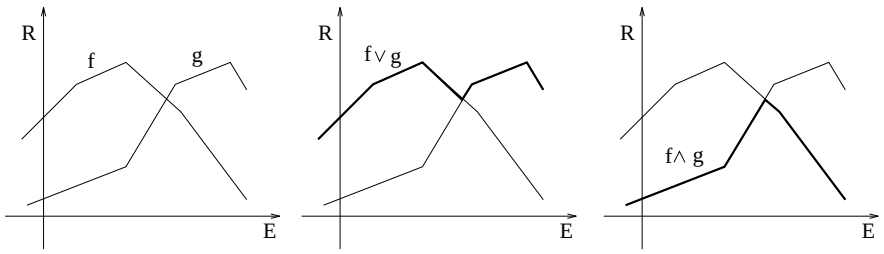


Figure 2.7: The supremum and infimum operations in the Lattice of functions.

Complete Lattice of Real Numbers

The set of real numbers, $\mathcal{R}$, is a Lattice, with the common supremum and infimum operations for real numbers. Here, the order relation is the usual order for real numbers. This Lattice, however, is not a Complete Lattice.

If we attach to $\mathcal{R}$ the infinity and minus infinity, i.e. $\mathcal{P} \stackrel{\triangle}{=} \mathcal{R} \cup \{\infty, -\infty\}$, and remain with the same supremum and infimum as before, then we obtain a Complete Lattice, where $-\infty$ and ∞ are now, respectively, the null element and the universe.

Complete Lattice of Functions

The basic framework for Grayscale Morphology is obtained as follows. Let $\mathcal{P}$ be the set of all functions from E to $\mathcal{R}$, with the following supremum and infimum operations, respectively:

$$[f \vee g](x) \stackrel{\triangle}{=} f(x) \vee g(x), \quad \forall x \in E, \tag{2.44}$$

$$[f \wedge g](x) \stackrel{\triangle}{=} f(x) \wedge g(x), \quad \forall x \in E, \tag{2.45}$$

where the supremum and infimum operations in the right side of the equations are those of the Lattice of real numbers (see Fig. 2.7). In this Lattice (which is not complete) the order relation is a partial one, and it is characterized by:

$$f \leq g \Leftrightarrow f(x) \leq g(x), \forall x \in E. \tag{2.46}$$

In order to turn the above Lattice into a Complete Lattice, the set of functions from E to $\mathcal{R} \cup \{\infty, -\infty\}$ should be considered instead. The null element and the universe are now the functions returning, respectively, $-\infty$ and ∞ to every point $x \in E$.

2.4.3 Dilations and Erosions in Complete Lattices

A *Dilation* is defined in a Complete Lattice to be any operator which commutes with the supremum of the Lattice, and preserves its null element, i.e., the operator $\delta(\cdot)$ is a Dilation iff:

$$\forall \{X_i\}, X_i \in \mathcal{P}, i \in I \quad \delta(\bigvee_{i \in I} X_i) = \bigvee_{i \in I} \delta(X_i) \tag{2.47}$$

$$\delta(\emptyset) = \emptyset \tag{2.48}$$

Similarly, an operator $\varepsilon(\cdot)$ in $\mathcal{P}$ is called an *Erosion* if it commutes with the infimum of the Lattice, and preserves the universe, i.e.:

$$\forall\{X_i\}, X_i \in \mathcal{P}, i \in I \quad \varepsilon(\bigwedge_{i\in I} X_i) = \bigwedge_{i\in I} \varepsilon(X_i) \tag{2.49}$$

$$\varepsilon(U) = U \tag{2.50}$$

Both Dilation and Erosion are increasing operations in the Complete Lattice, that is:

$$\forall X, Y \in \mathcal{P}, \quad X \leq Y \Rightarrow \begin{cases} \delta(X) \leq \delta(Y) \\ \varepsilon(X) \leq \varepsilon(Y) \end{cases} \tag{2.51}$$

For each Dilation δ in a Complete Lattice there is one and only one associated Erosion ε, satisfying:

$$\forall X, Y \in \mathcal{P}, \quad \delta(X) \leq Y \Leftrightarrow X \leq \varepsilon(Y) \tag{2.52}$$

Similarly, for each Erosion there is one and only one Dilation, such that (2.52) is satisfied. The pairs (δ, ε) satisfying the above duality are called *adjoint*.

Given a Dilation δ, its adjoint Erosion is given, for all $X \in \mathcal{P}$, by:

$$\varepsilon(X) = \bigvee\{Y \in \mathcal{P} \mid \ \delta(Y) \leq X\} \tag{2.53}$$

Conversely, the Dilation adjoint δ of a given Erosion ε can be calculated by:

$$\delta(X) = \bigwedge\{Y \in \mathcal{P} \mid \ \varepsilon(Y) \geq X\} \tag{2.54}$$

Notice that the Dilations and Erosions defined in Binary Morphology and in Grayscale Morphology are particular cases of the above defined Dilations and Erosions in Complete Lattices.

Adjoint Erosions and Dilations satisfy the following property, for all $X \in \mathcal{P}$:

$$\delta\varepsilon\delta(X) = \delta(X), \quad \varepsilon\delta\varepsilon(X) = \varepsilon(X). \tag{2.55}$$

2.4.4 Openings and Closings in Complete Lattices

An *algebraic opening* (or, simply, opening) γ in a Complete Lattice is an operator satisfying the following requirements:

1. It is idempotent, i.e., $\gamma\gamma(X) = \gamma(X)$, $\forall X \in \mathcal{P}$.
2. It is increasing, i.e., $X \leq Y \Leftrightarrow \gamma(X) \leq \gamma(Y)$, $\forall X, Y \in \mathcal{P}$.
3. It is anti-extensive, i.e., $\gamma(X) \leq X$, $\forall X \in \mathcal{P}$.

Similarly, an *algebraic closing* (or just closing) ϕ is an operator in $\mathcal{P}$ which satisfies:

1. It is idempotent, i.e., $\phi\phi(X) = \phi(X)$, $\forall X \in \mathcal{P}$.
2. It is increasing, i.e., $X \leq Y \Leftrightarrow \phi(X) \leq \phi(Y)$, $\forall X, Y \in \mathcal{P}$.

3. It is extensive, i.e., $\phi(X) \geq X$, $\forall X \in \mathcal{P}$.

Important particular cases of openings and closings are, respectively, the operators γ_δ and ϕ_δ, defined for $X \in \mathcal{P}$ by:

$$\gamma_\delta(X) \triangleq \delta\varepsilon(X), \quad \phi_\delta(X) \triangleq \varepsilon\delta(X), \tag{2.56}$$

where (δ, ε) is an adjoint pair. These operators are called, respectively, the *morphological opening* and the *morphological closing*, associated with the Dilation δ.

Given a Dilation δ, the morphological opening and closing associated with it are given, for all $X \in \mathcal{P}$, by:

$$\gamma_\delta(X) = \bigvee\{\delta(Y) \mid \ Y \in \mathcal{P}, \delta(Y) \leq X\} \tag{2.57}$$

$$\phi_\delta(X) = \bigvee\{Y \in \mathcal{P} \mid \ \delta(Y) \leq \delta(X)\} \tag{2.58}$$

When there is no ambiguity, the index δ is removed from the notation of morphological openings and closings. All the openings and closings considered in this thesis are morphological, and therefore, the index is omitted, i.e., γ and ϕ denote morphological opening and closing, respectively.

2.5 Boolean Lattices

As described in chapter 3, Skeletons are defined in a particular type of Complete Lattices, called *Boolean Lattices*. Details about Boolean Lattices and their use in Mathematical Morphology can be found in chapter 2 of [52]. This section provides a summary of that material.

2.5.1 Definitions of Boolean Lattices

Original Definition

A Complete Lattice $\mathcal{P}$ is called *complemented*, if, for each $X \in \mathcal{P}$, there is an element $X^c \in \mathcal{P}$, called the *complement* of X, such that:

$$X \wedge X^c = \emptyset, \quad X \vee X^c = U. \tag{2.59}$$

If $\mathcal{P}$ is a complemented Complete Lattice, for which the complement of each element is *unique*, it is called a *Boolean Lattice*.

In a Boolean Lattice, one can define the operation of *set-difference* (denoted $-$), in the following way:

$$X - Y = X \wedge Y^c, \quad X, Y \in \mathcal{P} \tag{2.60}$$

The set-difference is fundamental for the Skeleton calculation.

The Lattice of subsets of an Euclidean space, $\mathcal{P}(E)$, as defined in section 2.4.2, is an example of a Boolean Lattice. On the other hand, the Lattice of functions is *not* a Boolean Lattice, since one cannot define the complement of a function in the sense of (2.59). This leads to theoretical difficulties in defining a Grayscale Skeleton (see chapter 4).

Usual Definition

Let us take a generic set E (not necessarily an Euclidean space), and consider the set of its subsets $\mathcal{P}(E)$, with the supremum and infimum operations being equal, respectively, to union and intersection. The above is a Boolean Lattice, regardless to the nature and contents of the set E.

It turns out that for every Boolean Lattice $\mathcal{P}$, there exists a set E, for which the associated Boolean Lattice $\mathcal{P}(E)$ (or part of it) is isomorphic to $\mathcal{P}$.

For this reason, it is usual [52] to *redefine* a Boolean Lattice as being a set of the form $\mathcal{P}(E)$, for some set E, associated with the union and the intersection, as supremum and infimum, respectively.

2.5.2 Structuring Functions and the Basic Operations

There are two "levels" in a Boolean Lattice $\mathcal{P}(E)$:

1. The "level" of "*points*", consisting of the elements of E. It is usual to denote "points" by lower-case letters, like x, y, etc.

2. The "level" of "*sets*", consisting of the elements of $\mathcal{P}(E)$, i.e., subsets of E. The "sets" are denoted by capital letters, like X, Y, etc.

A *structuring function* $\delta(x)$ in a Boolean Lattice $\mathcal{P}(E)$ is defined as being *any* function from "points" to "sets", i.e., $\delta : E \to \mathcal{P}(E)$. Although the generic notation for structuring functions (δ) is the same as the one used above for Dilations, it is rare to confuse between them, since the latter one maps "sets" to "sets", i.e., $\delta : \mathcal{P}(E) \to \mathcal{P}(E)$. Structuring functions are, in Morphology of Boolean Lattices, the analogous of structuring elements in Binary and Grayscale Morphology.

The reason for using the same notation (which is introduced in [52]) for both operators, is that there is a one-to-one relationship between structuring functions and Dilations in Boolean Lattices. Every structuring function $\delta(x)$ determines an unique Dilation $\delta(X)$ in the following way:

$$\delta(X) = \bigcup_{x \in X} \delta(x) \tag{2.61}$$

Conversely, every Dilation $\delta(X)$ is related to an unique structuring function $\delta(x)$ by:

$$\delta(x) = \delta(\{x\}) \tag{2.62}$$

where $\{x\}$ denote the set in $\mathcal{P}(E)$ containing only the point x.

Since every structuring function automatically defines an unique Dilation, and since every Dilation uniquely determines an Erosion, a morphological opening and a morphological closing, it follows that the definition of a structuring function in a Boolean Lattice automatically defines the four basic morphological operations.

Chapter 3

Generalization of the Skeleton Framework

According to the discussion in section 1.2.2, a *generalization* of the Skeleton framework should be sought, based either on a *topological* or an *algebraic* approach. In the *topological* approach, such a generalization should aspire to solve problems like robustness, connectivity and precision as a shape descriptor; whereas, in the *algebraic* one, framework flexibility, self-duality, and representation efficiency are the main issues.

The algebraic approach is the one adopted throughout this thesis, as justified in section 1.3.2. Therefore, framework flexibility is extensively analyzed and generalized in this chapter, a quasi self-duality is proposed in Appendix D, and representation efficiency is studied in chapters 6 and 7.

3.1 Historical Background

Serra suggested in [53] that, in order to obtain an appropriate representation from the algebraic point of view, the following requirements concerning a skeleton decomposition should be satisfied:

1. *Existence and uniqueness* of the Skeleton of a set, for a given family of decomposition elements.

2. *Perfect reconstruction* of the original set from the Skeleton representation.

3. An *explicit formula* for computing the Skeleton.

The work of Lantuéjoul (see section 2.2) showed that the original Skeleton satisfies the above requirements, where requirements 2 and 3 can be satisfied using morphological operations.

During the last few years, the algebraic framework of the Skeleton was extended several times. The purpose was always to obtain decompositions according to richer families of elements (other than just discs), not failing to satisfy the above algebraic requirements. In this evolutionary development, the "Grass-fire" model and the definition in terms of *distance function* (Definition 2 in page 6) were abandoned; the Generalized Morphological Skeleton,

at every stage of its evolution, was defined only as the collection of "centers" and "radii" of maximal "elements" (as in Definition 1, in page 6), with the notions of "center", "radius", and "element" being extended.

In this section, the above evolution is described. Then, in the following sections, a new evolutionary branch is added to it. And finally, a general framework is proposed, unifying all the Generalized Skeletons in the evolution, and permitting us to obtain new representations, as particular cases of it.

Throughout its theoretical development, the Morphological Skeleton Decomposition was mainly related to *sets* in Euclidean spaces or Lattices; little was done concerning Skeletons of *functions*. The historical development presented here is also related to *sets*. Most of the generalizations can nevertheless be directly extended to functions; this is considered in chapter 4.

The historical evolution presented here has its mathematical aspects summarized in Tables 3.1 to 3.5. Tables 3.1, 3.4, 3.5 present the evolution of the conditions on the decomposition family and on the original set to be decomposed, whereas Tables 3.2 and 3.3 show the evolution of the computation (Lantuéjoul's) and reconstruction formulæ, respectively. The historical evolution is presented in the sequel, with emphasis on its main ideas.

3.1.1 Discrete Skeleton

The original Skeleton decomposes a shape into maximal discs (see Fig. 3.1(a)). However, on a rectangular grid one can not define an Euclidean disc. Therefore, skeletonization of discrete shapes is a difficult task. It turns out to be an impossible task if one seeks to maintain all the topological *and* algebraic properties presented by the continuous Skeleton.

A Skeleton on hexagonal grids, keeping the algebraic properties of the continuous Skeleton, was proposed by Serra in [51, pp. 387]. It consists of a direct adaptation of the definition in terms of maximal elements (Definition 1, page 6), where instead of maximal *discs*, the given set is decomposed into maximal *digital hexagons*. The digital hexagons are symmetric around their centers, and have sides $(n+1)$ pixels, $n = 0, 1, \ldots$. Maragos and Schafer [29] adapted this idea also for rectangular grids, where digital squares are used instead of hexagons. The digital squares are of sizes $(2n+1) \times (2n+1)$ pixels, $n = 0, 1, \ldots$, so that they also are symmetric around their centers.

The decomposition family of elements, into which a given set X is to be decomposed, is the above family of digital hexagons or squares. (See Fig. 3.1(b)). They are denoted by nB, to keep the analogy with the family of discs in the continuous case, and they are all centered at the origin.

The second column of Table 3.1 describe the conditions for the Discrete Skeleton, as compared to those of the original Skeleton (column 1 of the same table). The first 3 lines characterize the structure of the decomposition family. Lines 4 to 7 compare conditions on the family and on the given set X, so that the computation and reconstruction formulæ can be applied. Lines 8 and 9 relate to conditions added later to the historical evolution. The computation and reconstruction formulæ for the Discrete Skeleton are shown, respectively, in the second line of Tables 3.2 and 3.3, where a comparison to those of the original Skeleton (line 1 in both tables) can be seen. Note that perfect reconstruction is obtained by $k = 0$.

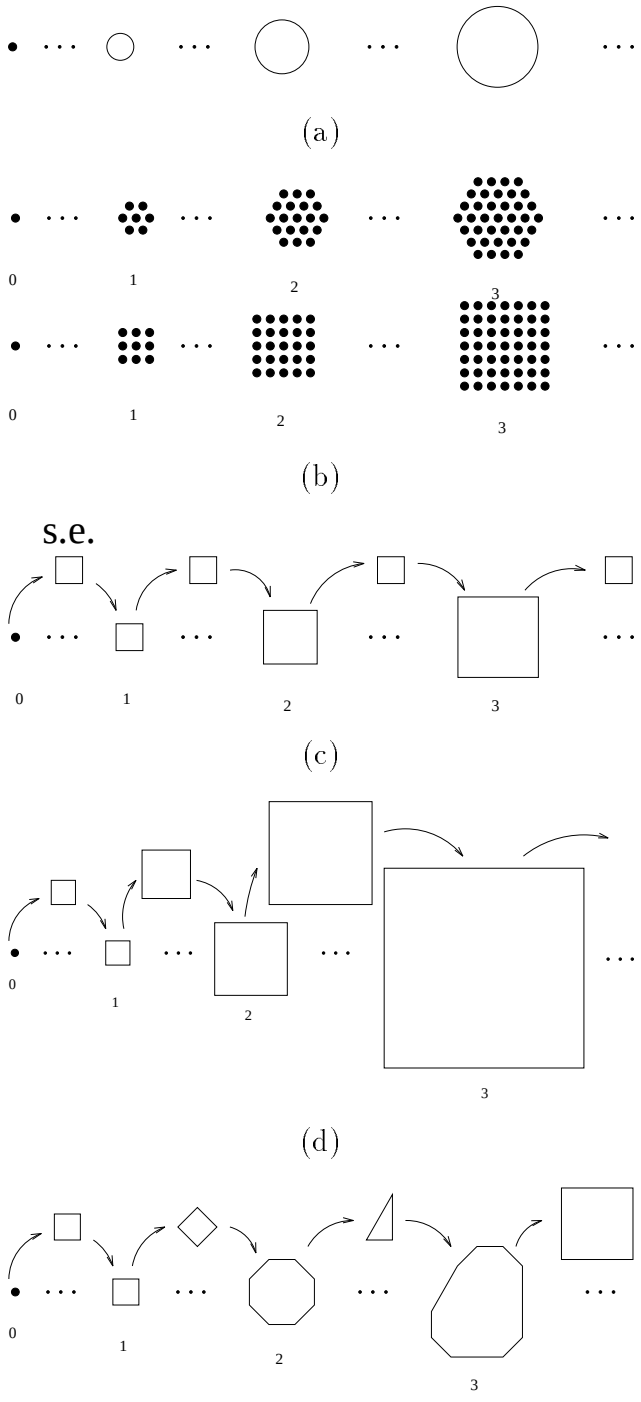


Figure 3.1: Decomposition family types for the first branch of evolutionary development of the Skeleton. (a) increasing discs (Original Skeleton), (b) discrete disc-like shapes (Discrete Skeleton), (c) family recursively generated by a fixed structuring element (Morphological Skeleton), (d) by a size-varying structuring element (Modified Skeleton), and (e) by a size and shape-varying structuring element (Generalized-Step Skeleton).

	Original Skeleton	Discrete Skeleton	Morphological Skeleton
1	Decomposition family indexed by $r \in \mathcal{R}^+ - \{0\}$.	Decomposition family indexed by $n \in \mathcal{N}$.	Decomposition family indexed by $n \in \mathcal{N}$.
2	Decomposition family is the set of discs off radii r.	Decomposition family is the set of discrete squares (or hexagons) of sizes n.	Family is recursively generated: $nB = \underbrace{B \oplus \cdots \oplus B}_{n \text{ times}}$ where B is a structuring element.
3	-	The single pixel is part of the family - it is the discrete square (hexagon) of "size 0".	The "single-point" element belongs to the family: $0B \stackrel{\triangle}{=} \{(0,0)\}$
4	X is topologically open.	-	-
5	All the discs in the family are topologically open.	-	B topologically open.
6	X is bounded.	X is bounded.	X is bounded.
7	All the discs are centered at the origin.	All the squares (hexagons) are centered at the origin.	B contains the origin.
8	-	-	B is "convex".
9	-	-	-

Table 3.1: Historical evolution of the conditions on the Skeleton decomposition - part I.

	Skeleton type	Lantuéjoul's Formula
1	Original Skeleton	$S_r = X \ominus rB - (X \ominus rB) \circ (dr)B$ where rB is an open disc of radius r, and $(dr)B$ is a closed disc with an infinitesimally small radius.
2	Discrete Skeleton	$S_n = X \ominus nB - (X \ominus nB) \circ B$ where nB is a $(2n+1) \times (2n+1)$-pixel square (hexagon with side $n-1$).
3	Morphological Skeleton	$S_n = X \ominus nB - (X \ominus nB) \circ B$ where B is the structuring element.
4	Modified Skeleton	$S_n = X \ominus 2^{n-1}B - (X \ominus 2^{n-1}B) \circ 2^{n-1}B$ for $n \geq 1$, and $S_0 = X - X \circ B$.
5	Generalized-Step Skel.	$S_n = X \ominus A(n) - [X \ominus A(n)] \circ B(n)$
6	MSES	$S_{\vec{n}} = X \ominus A(\vec{n}) - \bigcup_{\ell=1}^{d} [X \ominus A(\vec{n})] \circ B_\ell$
7	Skeleton on Lattices	$S_\lambda = \varepsilon_\lambda(X) - \bigcup_{\varepsilon>0} \gamma_\varepsilon \varepsilon_\lambda(X)$
8	Gen. Skel on Lattices	$S_\lambda = \varepsilon_\lambda(X) - \bigcup_{\mu>\lambda} \gamma_{[\mu,\lambda]} \varepsilon_\lambda(X)$ where $\lambda, \mu \in \mathcal{R}^+$.
9	PROPOSED Generaliz.	$S_i = \varepsilon_i(X) - \bigcup_{j>i} \gamma_{[j,i]} \varepsilon_i(X)$ where $i, j \in I$, generic.

Table 3.2: Historical evolution of Lantuéjoul's Formula.

We notice that conditions 4 and 5, satisfied by the original Skeleton, are not needed for the Discrete Skeleton, since every discrete set is both topologically open and closed. Actually, condition 4, which is needed in the original Skeleton for perfect reconstruction, has its role replaced here by condition 3.

3.1.2 Morphological Skeleton

Also in [29], Maragos and Schafer proposed the following further evolution step: Instead of restricting the Skeleton decomposition to elements of a family of disc-like elements on grids, let it be a decomposition into increasing versions of *any convex shape*, like a rhombus, a triangle, a line, etc. This includes also the previous decomposition into squares and hexagons.

Although basically meant to shapes on grids, the Morphological Skeleton can relate also to a *discrete family* of *continuous shapes*. At this point we wish to differentiate between a *discrete family* and *discrete shapes*. A *discrete family* of shapes is a family indexed by a discrete index, like $n \in \mathcal{N}$. *Discrete shapes*, on the other hand, are shapes in a discrete Euclidean space, like $\mathcal{Z}^L$. This can be a source of confusion, since a *Discrete Skeleton* can be found in the literature relating to both cases. Here, we denote by *discrete Skeleton* the Skeleton on discrete *spaces*, whereas by *discrete-family Skeleton* the Skeleton based on a discrete *family* of (continuous or discrete) shapes. Therefore, the Morphological Skeleton defined by Maragos and Schafer is of the latter type.

The theoretical characterization of the Morphological Skeleton, in terms of Mathematical

	Skeleton type	Reconstruction Formula
1	Original Skeleton	$X \circ sB = \bigcup_{r>s} S_r \oplus rB$ where rB is an open disc of radius r, and sB is a closed disc with radius s.
2	Discrete Skeleton	$X \circ kB = \bigcup_{n \geq k} S_n \oplus nB$ where nB is a $(2n+1) \times (2n+1)$-pixel square (hexagon with side $n-1$).
3	Morphological Skeleton	$X \circ kB = \bigcup_{n \geq k} S_n \oplus nB$ where B is the structuring element.
4	Modified Skeleton	$X \circ 2^{k-1}B = \bigcup_{n \geq k} S_n \oplus 2^{n-1}B$ for $k \geq 1$, and $X = (X \circ B) \cup S_0$.
5	Generalized-Step Skel.	$X \circ A(k) = \bigcup_{n \geq k} S_n \oplus A(n)$
6	MSES	$X \circ A(\vec{k}) = \bigcup_{\vec{n} \geq \vec{k}} S_{\vec{n}} \oplus A(\vec{n})$
7	Skeleton on Lattices	$\gamma_\mu(X) = \bigcup_{\lambda \geq \mu} \delta_\lambda(S_\lambda)$
8	Gen. Skel on Lattices	$\gamma_\mu(X) = \bigcup_{\lambda \geq \mu} \delta_\lambda(S_\lambda)$
9	PROPOSED Generaliz.	$\gamma_J(X) = \bigcup_{i \in J} \delta_i(S_i)$ where J is an *anti-umbra* in I.

Table 3.3: Historical evolution of the Formula for reconstruction from the Skeleton representation.

Morphology, is as follows. Let B be a structuring element in an Euclidean space E (continuous or discrete). We generate from B a *discrete* family of elements $\{nB\}$, each with "size" n, in the following manner:

$$\begin{aligned} nB &= \underbrace{B \oplus \cdots \oplus B}_{n \text{ times}}, \quad n \geq 1 \\ 0B &= (0,0). \end{aligned} \tag{3.1}$$

Where $(0,0)$ is the origin.

The structuring element B is required to contain the origin (considered as the "center" of B), to be topologically open, and to satisfy:

$$B \bullet B = B. \tag{3.2}$$

Condition (3.2) makes B "convex" in a general sense (see section 7.1 where the concept of B-Convexity is introduced). In the continuous case, any convex shape satisfies (3.2). In the discrete case, there is no strict definition of convexity, and (3.2) serves as an appropriate generalized definition.

Because B is convex, the elements nB have roughly the same shape as B, and have sizes roughly proportional to n times the size of B. If B is, for instance, a *continuous* square with side a, then nB is a continuous square with side na. If B is a discrete square on a rectangular grid with side equal to 3 pixels, then nB is a discrete square with side $2n+1$. The variable n is often called the "radius" of the element, to keep an analogy to the classical family of discs.

The family $\{nB\}$ can be generated *recursively* in the following way, as illustrated in Fig. 3.1(c):

$$nB = (n-1)B \oplus B, \quad n \geq 1 \tag{3.3}$$

If the conditions shown in column 3 of Table 3.1 are satisfied, then the collection of subsets $\{S_n(X)\}_{n\geq 0}$ obtained by the adapted Lantuéjoul's Formula shown in line 3 of Table 3.2 is indeed a Skeleton Decomposition. That is, an element with "radius" n and "centered" at a point s, denoted $nB_s \stackrel{\triangle}{=} \{b+s \mid b \in nB\}$, is a *maximal element* in X iff $s \in S_n(X)$ [29]. It is also an error-free representation since the original image X can be reconstructed from $\{S_n(X)\}$ by the reconstruction formula given in line 3 of Table 3.3, with $k=0$.

From the topological point of view, the Morphological Skeleton has great disadvantages; e.g., for B equal to a square, shape topology is not expected to be preserved, therefore non-connected skeletons can be obtained for connected shapes, both in continuous and discrete cases. But from the algebraic point of view, which is the one in which we are interested in, the Morphological Skeleton represents an advance towards the solution of the general optimization problem defined in section 1.3.1.

3.1.3 Modified Skeleton

The family of elements $\{nB\}_{n\in\mathcal{N}}$ used in the *discrete* Morphological Skeleton decomposition is generated by recursively dilating the structuring element B by itself. B serves here as a generator, being constant at every step of the family generation.

Sapiro and Malah [48] showed that the family generator can have a *variable size*. The *Modified Morphological Skeleton* proposed by them has its subsets $\{S_n(X)\}$ defined as in line

4 of Table 3.2. It was proved [48] that the Modified Skeleton decomposes X into maximal elements from the family $\{0B, B, 2B, 4B, 8B, 16B, \ldots\}$ (Fig. 3.1(d)). This family can be generated, as before, by a series of dilations, but not with a constant generator. The sizes of the generator, at the various steps, are the differences $\{1, 1, 2, 4, 8, \ldots\}$ between the sizes of the elements of the family.

We can observe, by comparing the conditions for the Modified Skeleton with those for the Morphological Skeleton (first column of Table 3.4 and last column of Table 3.1, respectively), that the only significant difference is in the family generation. Moreover, the Modified Skeleton also fully represents the original image X (See line 4 of Table 3.3). In the Coding simulations presented in [48], the Modified Morphological Skeleton showed better results than the previous Morphological Skeleton.

	Modified Skeleton	Generalized-Step Skel.	PROPOSED MSES
1	Decomposition family indexed by $n \in \mathcal{N}$.	Decomposition family indexed by $n \in \mathcal{N}$.	Decomposition family indexed by $\vec{n} \in \mathcal{N}^d$.
2	Family is exponentially generated: $A(n) = 2^{n-1}B$, for $n \geq 1$.	A series $\{B(n)\}_{n\geq 0}$ generates the family by: $A(n+1) = A(n) \oplus B(n)$, for $n \geq 0$.	The multi-dimension family generation: $A(\vec{n}) = n_1 B_1 \oplus \cdots \oplus n_d B_d$ where B_ℓ, $\ell = 1, 2, \ldots, d$, are structuring elements.
3	The "single-point" element belongs to the family: $A(0) = \{(0,0)\}$	The "single-point" element belongs to the family: $A(0) = \{(0,0)\}$	The "single-point" element belongs to the family: $A(\vec{0}) = \{(0,0)\}$
4	-	-	-
5	B is topologically open.	Each $B(n)$ is topologically open.	Each B_ℓ, $\ell = 1, \ldots, L$, is topologically open.
6	X is bounded.	X is bounded.	X is bounded.
7	B contains the origin.	Each $B(n)$ contains the origin.	Each B_ℓ, $\ell = 1, \ldots, L$, contains the origin.
8	B is "convex".	Each $B(n)$ is "convex".	Each B_ℓ is "convex", in the following sense: $B_\ell = B_\ell \bullet A_i$, $\forall i \in I$.
9	-	-	$A(\vec{n})_x \subseteq A(\vec{m})_y \Rightarrow \begin{cases} \vec{n} \leq \vec{m}, & x \neq y, \\ \vec{n} < \vec{m}, & x = y. \end{cases}$

Table 3.4: Historical evolution of the conditions on the Skeleton decomposition - part II.

3.1.4 Generalized-Step Skeleton

Goutsias and Schonfeld [50] showed that not only the *size* of the generator can vary at each step of the family generation, but also the generator's *shape*.

Let $\{B(n)\}_{n\in\mathcal{N}}$ be a series of structuring elements, all of them containing the origin and satisfying (3.2). Let this series generate a family of elements $\{A(n)\}_{n\in\mathcal{N}}$ in the following way (see Fig. 3.1(e)):

$$\begin{aligned} &A(n+1) = A(n) \oplus B(n), \quad n = 0, 1, 2, \ldots \\ &A(0) = (0,0). \end{aligned} \tag{3.4}$$

Goutsias and Schonfeld showed that the maximal elements from the above family $\{A(n)\}$, inside a given shape X, have their positions ("centers") given by the subsets $\{S_n\}$ as shown in line 5 of Table 3.2. Therefore, the collection $\{S_n(X)\}_{n\in\mathcal{N}}$ is called in this case *Generalized-Step Morphological Skeleton.*

The conditions for this Skeleton are shown in column 2 of Table 3.4. They show no significant difference as compared to the previous Skeletons, apart of the family structure. X is recovered from the above skeleton as shown in line 5 of Table 3.3.

Note that both the original Morphological Skeleton and the Modified Morphological Skeleton are particular cases of the Generalized-Step Morphological Skeleton. The first is obtained by choosing $B(n)$ constant and equal to B, whereas the second is obtained with the choice $B(n) = \{B, B, 2B, 4B, \ldots\}$.

3.1.5 Skeleton on Boolean Lattices

The Modified Morphological Skeleton and the Generalized-Step Morphological Skeleton can be seen as a first evolutionary branch in the development of the algebraic Skeleton (Fig. 3.2, center line).

A second branch (on the right hand side of Fig. 3.2), also evolved from the Morphological Skeleton, was proposed by Serra [52, chapter 2] as part of the generalization of Mathematical Morphology from Euclidean spaces into Lattices. This branch is reviewed here.

While the first branch makes possible Skeleton decompositions with any *totally ordered* family of elements, generated through recursive *Euclidean* dilations, the second branch, described below, permits:

1. Decompositions of sets in any Boolean Lattice. [1]

2. Translation-*variant* decompositions. As opposed to the family generator in the first branch (the structuring element B), which is translation-invariant (Fig. 3.3(a)), the generator in the second branch can vary according to its position (structuring function $\delta(x)$. See Fig. 3.3(b)).

The above ideas are mathematically characterized in the following way (see [52, chapter 2]): Let $\{\delta_\lambda(x)\}_{\lambda\geq 0}$ be a family of structuring functions in a Boolean Lattice $\mathcal{P}(E)$. I.e., each $\delta_\lambda(x)$ maps "points" in E into "sets" in $\mathcal{P}(E)$. Let $X \in \mathcal{P}(E)$ be the set to be decomposed. If E is a continuous set, such as $\mathcal{R}^2$, then X is a continuous set. Otherwise, if for instance $E = \mathcal{Z}^2$, X is discrete.

In [52], the family index is supposed to be continuous, i.e., λ is real. However, the transposition to a discrete-index case, where λ is integer, is straight-forward.

[1] Although many of the morphological tools proposed to Euclidean spaces (like the basic operators, filters, etc.) were generalized for generic Lattices, the Skeleton requires *Boolean* Lattices as framework.

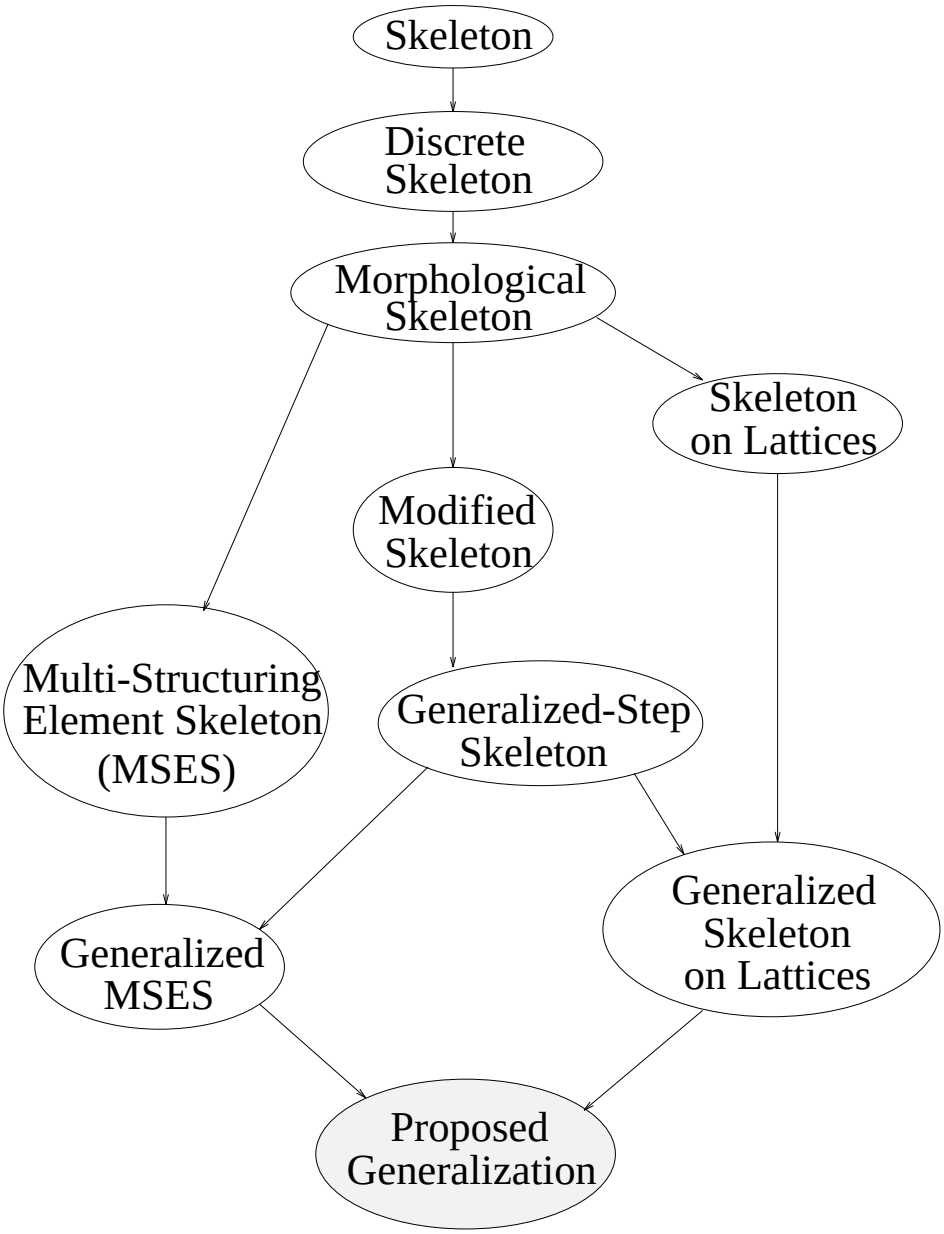
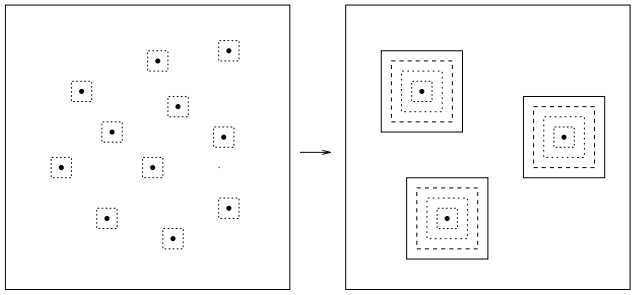
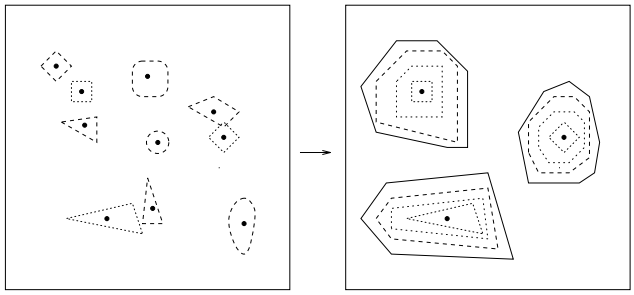


Figure 3.2: Evolutionary development of the Skeleton.



(a)



(6)

Figure 3.3: (a) Family generation by a structuring element (translation-invariant structuring function), (b) family generation by a translation-variant structuring function.

As any structuring function (see section 2.5.2), each element of the family $\{\delta_\lambda(x)\}$ automatically defines a dilation $\delta_\lambda(X)$, an erosion $\varepsilon_\lambda(X)$, an opening $\gamma_\lambda(X)$, and a closing $\phi_\lambda(X)$.

The family $\{\delta_\lambda(x)\}$ is said to be valid for skeletonization if it satisfies the requirements shown in column 1 of Table 3.5. These conditions are a generalization of those of the previous Skeletons, each having the same line number at the respective tables. Condition 5 can be disconsidered if the Boolean Lattice is discrete, since in this case this condition is naturally satisfied (this is true for all the Skeletons in the historical evolution).

An "element" here denotes any set of the form $\delta_\lambda(x)$, $\lambda \geq 0$, $x \in E$, where λ is denoted the "radius" of the element, and x is denoted its "center". An element $\delta_\lambda(x)$ is a "maximal element" in X if $\delta_\lambda(x) \subseteq X$, and for all $\mu \neq \lambda$, and for all $y \in E$, if $\delta_\lambda(x) \subset \delta_\mu(y)$, then $\delta_\mu(y) \not\subseteq X$.

The Skeleton of X is then defined as the collection of the "centers" of all the "maximal elements".

The Skeleton subsets $\{S_\lambda(X)\}$, each defined as the collection of skeleton points related to a "maximal element" with "radius" equal to λ, can be calculated [52] by the generalization of Lantuéjoul's Formula shown in line 7 of Table 3.2. The reconstruction formula assumes here the form shown in line 7 of Table 3.3.

	Skeleton on Lattices	Generalized Skel. Lattices	PROPOSED Gen. Skeleton
1	Decomposition family is indexed by $\lambda \in \mathcal{R}^+$.	Decomposition family is indexed by $\lambda \in \mathcal{R}^+ - \{0\}$.	Decomposition family is indexed by $i \in I$, where I is *any* totally or partially ordered set.
2	The family has a semi-group structure: $\delta_{\lambda+\mu} = \delta_\lambda \delta_\mu$	Granulometry generating family: $\lambda \geq \mu \Rightarrow$ $\gamma_\mu \delta_\lambda(Y) = \delta_\lambda(Y)$, $\forall Y \in \mathcal{P}(E)$	Generalized granulometry generating family: $j \geq i \Rightarrow$ $\gamma_i \delta_j(Y) = \delta_j(Y)$, $\forall Y \in \mathcal{P}(E)$
3	The "identity" belongs to the family, i.e.,: $\delta_0(x) \triangleq \{x\}$	-	-
4	-	X is morphologically open by the family, i.e.,: $X = \bigcup_{\lambda>0} \gamma_\lambda(X)$	X is morphologically open by the family, i.e.,: $X = \bigcup_{i \in I} \gamma_i(X)$
5	For any chain $\{\delta_{\lambda_k}(x_k)\}$ satisfying $(\ell > k \Rightarrow \delta_{\lambda_\ell}(x_\ell) \supset \delta_{\lambda_i}(x_i))$: $\bigcup_{k\geq 0} \delta_{\lambda_k}(x_k) = \delta_\lambda(x)$ for some $x \in E$, and $\lambda \geq 0$.	For any chain $\{\delta_{\lambda_k}(x_k)\}$ satisfying $(\ell > k \Rightarrow \delta_{\lambda_\ell}(x_\ell) \supset \delta_{\lambda_k}(x_k))$ and $(\forall k, \delta_{\lambda_k}(x_k) \subseteq X)$: $\bigcup_{k\geq 0} \delta_{\lambda_k}(x_k) = \delta_\lambda(x) \subseteq X$ for some $x \in E$, and $\lambda \geq 0$.	For any chain $\{\delta_{i_k}(x_k)\}$ satisfying $(\ell > k \Rightarrow \delta_{i_\ell}(x_\ell) \supset \delta_{i_k}(x_k))$ and $(\forall k, \delta_{i_k}(x_k) \subseteq X)$: $\bigcup_{k\geq 0} \delta_{i_k}(x_k) = \delta_i(x) \subseteq X$ for some $x \in E$, and $i \in I$.
6	X has a global ultimate erosion: $\exists \lambda_{i_0} \mid \varepsilon_\lambda(X) = \emptyset$, $\forall \lambda > \lambda_{i_0}$.	-	-
7	The family is increasing: $\lambda > \mu \Rightarrow \delta_\lambda(x) \supset \delta_\mu(x)$, $\forall x \in E$.	The family is increasing: $\lambda > \mu \Rightarrow \delta_\lambda(x) \supset \delta_\mu(x)$, $\forall x \in E$.	-
8	"Generalized Convexity": $\forall x \in E, \forall \lambda, \mu \geq 0$ $\delta_\lambda(x) = \varepsilon_\mu \delta_{\lambda+\mu}(x)$	-	-
9	-	$\delta_\lambda(x) \subseteq \delta_\mu(y) \Rightarrow$ $\begin{cases} \lambda \leq \mu, & x \neq y, \\ \lambda < \mu, & x = y. \end{cases}$	$\delta_i(x) \subseteq \delta_j(y) \Rightarrow$ $\begin{cases} i \leq j, & x \neq y, \\ i < j, & x = y. \end{cases}$

Table 3.5: Historical evolution of the conditions on the Skeleton decomposition - part III.

3.1.6 Generalized Skeleton on Boolean Lattices

In [53] Serra unified the two evolutionary branches, described above, into a Generalized Skeleton on Boolean Lattices (see Fig. 3.2). As seen above, the first branch is mainly characterized by a "step"-variant space-invariant family generator, whereas the second branch is mainly characterized by a "step"-invariant space-variant family generator. Thus, the main advance in [53] is in permitting a "step" *and* space variant family generator for the Skeleton decomposition.

The requirements on the family of structuring functions $\{\delta_\lambda(x)\}_{\lambda>0}$ for Serra's Generalized Skeleton, are presented in column 2 of Table 3.5. By comparing them to the conditions for the original Skeleton on Lattices (column 1 of same table), one can see several significant improvements:

- The structure of the family (condition 2) is much more flexible in the generalized Skeleton on Lattices. In particular, it permits "step"-variant family generation, whereas the original Skeleton on Lattices does not.
- Condition 8 is *dropped*! It turns out that this condition, which was historically included in order to give a *self-similarity* characteristic to the decomposition family, is not actually required.
- Condition 9 is added. It turns out that this condition was overlooked in the previous Skeletons; without it, in some non-conventional cases, the result of Lantuéjoul's Formula can strictly *contain* the actual Skeleton (see [53]).
- Conditions 5 and 6 are joined together.
- Condition 3 is replaced by the more general condition 4.

One can also notice that in condition 1 the value 0 was excluded from the set of indices. This is a minor difference, though, since Serra could have chosen to define the generalized Skeleton on Lattice *with* $\lambda = 0$ also. This was probably not done to keep an analogy with the original Skeleton (Table 3.1, column 1).

Another major difference is seen in the generalized Lantuéjoul's Formula (line 8 in Table 3.2). Instead of using γ_ε, $\varepsilon > 0$, the new Skeleton uses the opening $\gamma_{[\mu,\lambda]}$, which is defined as the opening related to the structuring function $\delta_{[\mu,\lambda]}(x) \stackrel{\triangle}{=} \varepsilon_\lambda \delta_\mu(x)$. This is needed for a "step"-variant family generation, since now the opening in Lantuéjoul's Formula is a function of the "step" λ. The reconstruction formula (line 8 in Table 3.3), on the other hand, remains identical to that of the previous Skeleton.

3.2 Multi-Structuring-Element Skeleton (MSES)

During the above historical development, the family of decomposition elements was always indexed by a non-negative *scalar* parameter (λ or r in the continuous case, n in the discrete case).

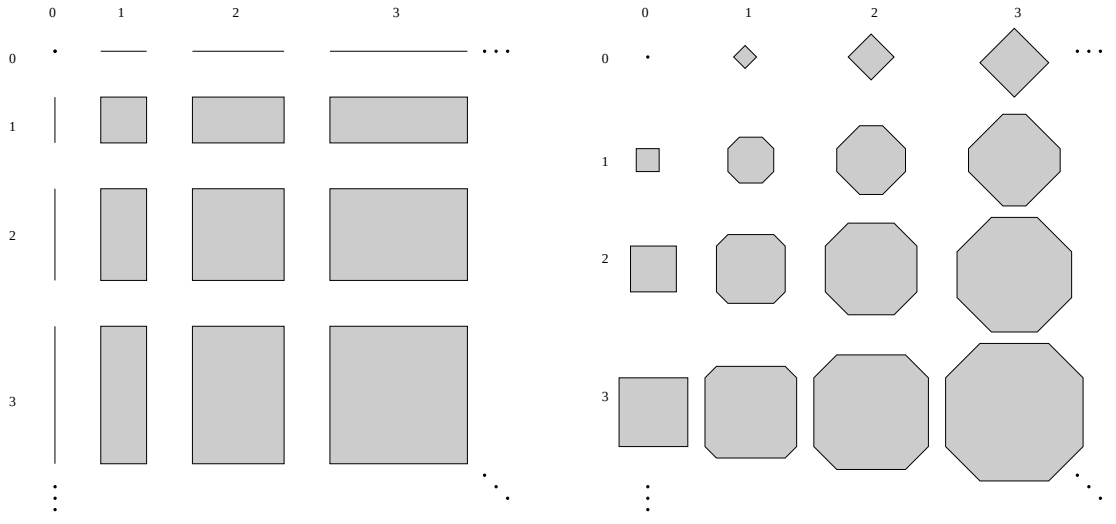


Figure 3.4: Two-parameter families of elements. (a) Structuring elements are an horizontal and a vertical unit lines, and (b) a unit square and a unit rhombus.

Here we add a new branch to the evolutionary tree described above (Fig. 3.2), by replacing the scalar parameter by a *vector* one The fact that no attempt was made previously to perform this replacement can be explained, perhaps, by the interest in obtaining a meaningful Skeleton from the *topological* and *geometrical* points of view. However, the Multi-Structuring-Element Skeleton (MSES) proposed by us in [17], and presented below, *cannot* be considered as a particular case in Serra's framework (it demands a partial ordering of the indices of the decomposition subsets), and, nevertheless, it satisfies the three *algebraic* requirements. Moreover, it *is* a collection of centers of maximal elements, as it will be shown.

The main contribution of the MSES is in providing a decomposition of shapes into maximal elements from a *multi-parameter* family of elements, $\{A(\vec{n})\}_{\vec{n}}$, where $\vec{n} = (n_1, \ldots, n_d) \in \mathcal{N}^d$. Given d structuring-elements, $B_1, B_2, \ldots, B_d$, they generate $\{A(\vec{n})\}$ in the following way:

$$A(\vec{n}) = n_1 B_1 \oplus n_2 B_2 \oplus \ldots n_d B_d. \tag{3.5}$$

For example, Fig. 3.4 shows two 2-parameter families of planar shapes; in Fig. 3.4(a), $d = 2$ and B_1 is a *vertical unit line*, whereas B_2 is an *horizontal unit line*. The family $\{A(n_1, n_2)\}$ is the set of all the discrete rectangles. Notice that the 1-parameter family of squares is contained in the 2-parameter family just defined. In Fig. 3.4(b), $d = 2$ also, and B_1 is a *unit square*, whereas B_2 is a *unit rhombus*.

A maximal element in a set X from the family $\{A(\vec{n})\}$ is defined as an element $A(\vec{n})_x$ with "radius" $\vec{n}$ and "center" x, such that $A(\vec{n})_x \subseteq X$, and $\forall \vec{m} \in \mathcal{N}^d$, $\forall y \in E$, if $A(\vec{n})_x \subseteq A(\vec{m})_y$, then $A(\vec{m})_y \not\subseteq X$.

We attach to $\mathcal{N}^d$ its strong *order relation* ($\leq$):

$$\begin{array}{c} \vec{n} \leq \vec{m} \\ \Updownarrow \\ n_\ell \leq m_\ell, \ell = 1, 2, \ldots, d. \end{array} \tag{3.6}$$

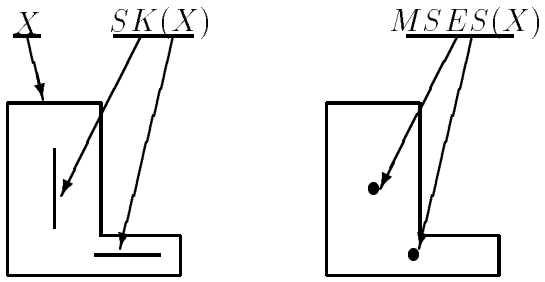


Figure 3.5: (a) A shape X and its Morphological Skeleton $SK(X)$, related to a family of squares. (b) Same shape X and its MSES, related to the family of rectangles shown in Fig. 3.4(a).

and we define the *Morphological Multi-Structuring-Element Skeleton*, for the above family, as the collections of subsets $\{S_{\vec{n}}\}$, $\vec{n} \in \mathcal{N}^d$, given by the generalized Lantuéjoul's Formula presented in line 6 of Table 3.2.

Fig. 3.5 compares a standard Morphological Skeleton and a MSES of a simple shape. In this case, the MSES contains only *two points*, because X is composed of only 2 rectangles.

If the conditions in column 3 of Table 3.4 are satisfied by the multi-parameter family, then the MSES is the set of center points of the maximal elements in X from $\{A(\vec{n})\}$ [2]. The proof of this assertion, together with the proof of the reconstruction formula (line 6 of Table 3.3), is postponed to section 3.3.3. It is shown there that the MSES is a particular case of the proposed generalized Skeleton, for which we prove the respective theorems.

3.2.1 Generalized-Step MSES

In [20], we presented a Generalized-Step MSES, which can be seen as a unification of the ideas from the first and third evolutionary branches of the Skeleton development (respectively, the center and the left branches in Fig. 3.2). The elements into which the Generalized-Step MSES decomposes a set X are *multi-parameter* combinations of elements from one-parameter sub-families, which are generated each in a *generalized-step* way (see example in Fig. 3.6).

In mathematical terms, the decomposition family $\{A(\vec{n})\}$, $\vec{n} \stackrel{\triangle}{=} (n_1, n_2, \ldots, n_d) \in \mathcal{N}^d$, is generated in the following way:

$$A(\vec{n}) = A_1(n_1) \oplus A_2(n_2) \oplus \ldots \oplus A_d(n_d) \tag{3.7}$$

where each sub-family $A_\ell(n)$, $\ell = 1, 2, \ldots, d$, is generated from a pre-defined series of shapes $B_\ell(n)$, in the same way as in the Generalized-Step Skeleton (see equation (3.4)). The

[2] Notice that the two-parameter family shown in Fig. 3.4(b) does not satisfy condition 9; therefore the MSES of a shape based on this family is expected the contain a superset of the maximal elements. For practical applications, this may not represent a problem, especially if one is interested in performing a reduction of the MSES's redundancy (Chapter 6), because this typically removes all the non-maximal elements from the representation.

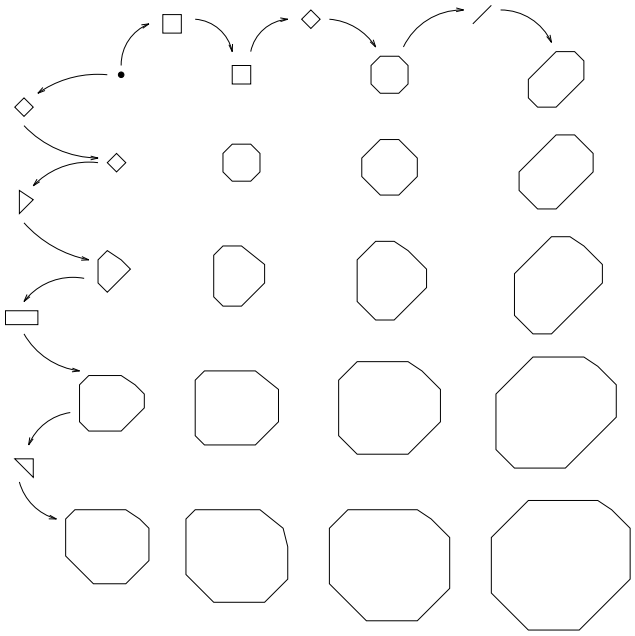


Figure 3.6: A generalized-step two-parameter family of elements, suitable for Generalized-Step MSES decomposition.

Generalized-Step MSES Subsets are given by the following version of Lantuéjoul's Formula:

$$S_{\vec{n}} = X \ominus A(\vec{n}) - \bigcup_{\ell=1}^{d} [X \ominus A(\vec{n})] \circ B_\ell(n_\ell) \tag{3.8}$$

The reconstruction formula is identical to that of the original MSES.

3.3 Proposed General Framework

Our purpose in this section is to propose a generalized definition of the Skeleton, having as particular cases both Serra's general Skeleton on Lattices (unification of the center and right branches in Fig. 3.2), and the Generalized-Step MSES (unification of the center and the left branches), thus obtaining a unification of all the three evolutionary branches. More than a theoretical unification of known decompositions, the proposed General Skeleton provides new decompositions, presented in the sequel, as particular cases of it.

The generalization is obtained mainly by replacing the previous family indices ($r \in \mathcal{R}, n \in \mathcal{N}, \lambda \in \mathcal{R}, \vec{n} \in \mathcal{N}^d$) by a *generic* index i, from *any* totally *or partially* ordered set of indices I. The computation and reconstruction formulæ and the family conditions are then adapted to assure that the algebraic requirements are still satisfied.

3.3.1 Generalized Skeleton Definition

Let us consider a set E and the Boolean Lattice defined by $\mathcal{P}(E)$ (the set of subsets of E), *inclusion* order, and *union* and *intersection* as the supremum and the infimum, respectively.

Let I be a set of indices i, totally *or partially* ordered by an order $\geq$. The set I can be, for instance, a d-dimensional space, so that the indices i are vectors.

We choose an arbitrary family of structuring functions from E into $\mathcal{P}(E)$, indexed by I, $\{\delta_i(x)\}_{i\in I}$. This choice uniquely determines the families of dilations $\{\delta_i(X)\}$, adjoint erosions $\{\varepsilon_i(X)\}$ and morphological openings $\{\gamma_i(X)\}$, where $X \in \mathcal{P}(E)$.

We define an *element* with "radius" $i \in I$ and "center" $x_0 \in E$, $\delta_i(x_0)$, as the image of x_0 by the structuring-function $\delta_i(x)$, and a *maximal element* in a set $X \in \mathcal{P}(E)$ as an element contained in X but not contained in any other element which is contained in X.

As usual, we define the Skeleton of a set $X \in \mathcal{P}(E)$ as the collection of the centers of all the maximal elements contained in it. It is unique for the chosen family of structuring-functions and the given set X. The sets $\{S_i(X)\}_{i\in I}$, each containing the centers of the maximal elements of "radius" i, are called *Skeleton Subsets*.

The above definitions (and also the theorems in next section, and their proofs) are a direct generalization of those of Serra's Generalized Skeleton on Lattices (presented in [53], and described above in the section 3.1.6).

3.3.2 Skeleton Computation and Set-Reconstruction Formulæ

The following theorem provides an explicit formula for computing the skeleton subsets $\{S_i\}_{i\in I}$:

Theorem 1 *If the family of structuring functions $\{\delta_i(x)\}_{i\in I}$ satisfies the following two conditions:*

1. $i \leq j \Rightarrow \gamma_i\delta_j(x) = \delta_j(x)$, $\forall x \in E$

   (i.e., the family is granulometry-generating *[51, 52, 53], in a generalized sense),*

2. *For all $i, j \in I$ and $x, y \in E$:*

$$\delta_i(x) \supseteq \delta_j(y) \Rightarrow \begin{cases} i > j, & \text{if } x \neq y \\ i \geq j, & x = y \end{cases}$$

then, for any $X \subset \mathcal{P}(E)$, and for all $i \in I$:

$$S_i(X) = \varepsilon_i(X) - \bigcup_{j>i\in I} \gamma_{[i,j]}\varepsilon_i(X) \tag{3.9}$$

where $\gamma_{[i,j]}$ is the morphological opening associated with the structuring function $\delta_{[i,j]}(x) \stackrel{\triangle}{=} \varepsilon_i\delta_j(x)$.

The proof is given in Appendix A.

Perfect reconstruction can be obtained from the proposed generalized Skeleton Subsets, if certain conditions are satisfied. In order to characterize these conditions, by means of a theorem, we consider the following definitions:

- Let L_X be the set of all the elements contained in X, i.e., $L_X \stackrel{\triangle}{=} \{\delta_i(x) \mid i \in I, x \in \varepsilon_i(X)\}$.

- A subset $\{\delta_{i_k}(x_k)\}_{k\in K}$ of L_X is called an *increasing chain* in L_X, if K is a totally ordered set of indices, and if $k < k'$ implies $\delta_{i_k}(x_k) \subseteq \delta_{i_{k'}}(x_{k'})$.

- We say that an increasing chain $\{\delta_{i_k}(x_k)\}$ *converges* to an element $\delta_{i_0}(x_0)$, if $\bigcup_{k\in K} \delta_{i_k}(x_k) = \delta_{i_0}(x_0)$.

- L_X is said to be *inductive for inclusion* [53], if *every* increasing chain in L_X converges to a unique element in L_X.

- Let J be any subset of the set of indices I. We say that J is an *anti-umbra* in I if the conditions $j \in J$ and $i \geq j$ imply $i \in J$.

- For an anti-umbra J in I, we define the following opening: $\forall X \in \mathcal{P}(E)$, $\gamma_J(X) \triangleq \bigcup_{j\in J} \gamma_j(X)$

Theorem 2 *Let $\{S_i(X)\}_{i\in I}$ be the skeleton decomposition of X, according to the family of structuring functions $\{\delta_i(x)\}_{i\in I}$. If L_X is inductive for inclusion, and J is an anti-umbra in I, then*

$$\bigcup_{i\in J} \delta_i(S_i(X)) = \gamma_J(X) \tag{3.10}$$

The perfect reconstruction is assured whenever $X = \gamma_J(X)$. Equation (3.10) also provides a *partial reconstruction* formula for the skeleton, in case that $\gamma_J(X) \neq X$. The proof of theorem 2 is also given in Appendix A.

The conditions in the above theorems are summarized in column 3 of Table 3.5. By comparing them with those of the Generalized Skeleton on Lattices (column 2 in same table), and by comparing the computation and reconstruction formulæ (see Tables 3.2 and 3.3), one can see that:

1. The proposed Generalized Skeleton is a direct adaptation of Serra's Generalized Skeleton on Lattices, having the scalar index λ replaced by a *generic* index i.

2. Condition 7 was dropped! This condition was historically required in order to obtain a Skeleton with a *quench function*, i.e., for which every skeleton point is related to one and *only one* maximal element. In other words, the Skeleton was previously required to have its Subsets mutually disjoint. However, we regard this as an important requirement from the *topological* point of view, but not from the *algebraic* one, since the algebraic conditions can be met also without it.

3.3.3 Multi-Parameter Skeleton and the MSES

Choosing I to be a totally ordered set of indices, such as $\mathcal{R}_+$ or $\mathcal{Z}_+$, brings us to the Serra's Generalized Skeleton on Lattices, as defined in [53]. In this section, we consider more specifically decompositions which are *not* based on totally ordered sets.

Let us consider the following particular case of the proposed Generalized Skeleton:

- I is a set of d-dimensional vectors, i.e., every index i is in the form $i \triangleq (i_1, i_2, \ldots, i_d)$. More specifically, $I = \mathcal{R}_+^d$ in the continuous case, and $I = \mathcal{Z}_+^d$ in the discrete case.

- The order of I is chosen to be its *strong order*, i.e., for any two indices $i^{(1)}$ and $i^{(2)}$ in I, $i^{(1)} \geq i^{(2)}$ iff $i_n^{(1)} \geq i_n^{(2)}$ for all $n = 1, 2, \ldots, d$.
- The conditions of theorems 1 and 2 hold.

We denote a such a Skeleton by *Multi-Parameter Skeleton*. If we denote as k_n the d-dimensional vector having its n-th component equal to 1 and all the others equal to 0, then the *continuous-case* Multi-Parameter Skeleton subsets are given by:

$$S_i(X) = \varepsilon_i(X) - \bigcup_{n=1}^{d} \left[\bigcup_{\lambda_n > 0} \gamma_{[i,i+\lambda_n k_n]} \varepsilon_i(X) \right], \qquad i \in \mathcal{R}_+^d \tag{3.11}$$

where λ_n, $n = 1, \ldots, d$, are *scalar* variables. Notice that the union in (3.9), which is performed over the *vector* variable j, is replaced in (3.11) by unions performed over *scalar* variables, n and λ_n, $n = 1, \ldots, d$.

The unions performed over λ_n in (3.11) express, actually, convergence towards a supremum, when $\lambda_n \to 0$. In the *discrete case*, these unions converge to maxima, which are reached when $\lambda_n = 1$, $n = 1, \ldots, d$. Therefore, in the discrete case, (3.11) becomes:

$$S_i(X) = \varepsilon_i(X) - \bigcup_{n=1}^{d} \gamma_{[i,i+k_n]} \varepsilon_i(X), \qquad i \in \mathcal{Z}_+^d \tag{3.12}$$

The MSES, defined in section 3.2, is actually a particular case of Multi-Parameter Skeleton, and this is shown in the remainder of this section. Let $I = \mathcal{Z}_+^d$, and its order be the strong partial order defined above. Let us select d convex structuring-elements B_n, $n = 1, \ldots, d$, and define for each i in I:

$$A_i \triangleq i_1 B_1 \oplus \ldots \oplus i_d B_d. \tag{3.13}$$

Moreover, let the family of structuring-functions (to be used in the decomposition) be in the form:

$$\delta_i(x) = A_i \oplus \{x\}, \quad i \in I \tag{3.14}$$

In this case, it holds:

$$\begin{aligned} \gamma_{[i,i+k_n]}(X) &= X \circ (A_{i+k_n} \ominus A_i) \\ &= X \circ [(A_{k_n} \oplus A_i) \ominus A_i] \\ &= X \circ (B_n \bullet A_i). \end{aligned} \tag{3.15}$$

When $B_n = B_n \bullet A_i$, $\forall i \in I$, $\forall n = 1, \ldots, d$, (condition 8 of the MSES, see Table 3.4), then the morphological openings $\gamma_{[i,i+k_n]}(X)$ are independent of i, and equal to $X \circ B_n$.

If we replace the above data in (3.12), then the associated Skeleton Subsets $S_i(X)$ are given by

$$S_i(X) = X \ominus A_i - \bigcup_{n=1}^{d} [X \ominus A_i] \circ B_n, \qquad i \in \mathcal{Z}_+^d. \tag{3.16}$$

Equation (3.16) is identical to the MSES's version of the Lantuéjoul's Formula presented in Table 3.2, when i is replaced by $\vec{n}$, and n by ℓ.

Chapter 4

Generalized Grayscale Skeletons

The proposed General Skeleton Representation presented in chapter 3, and therefore all its particular cases, are *binary*, since they are defined in *Boolean* Lattices. This chapter considers expanding the scope of the above Skeleton to *functions*, which leads to a *Generalized Grayscale Skeleton Representation*.

4.1 Background on Grayscale Skeletons

Among the first schemes for calculating a grayscale Skeleton of grayscale images is the work of Peleg and Rosenfeld [36]. In that work, the "min-max" approach for non-linear image transformations, developed by Rosenfeld [42], is applied to obtain the grayscale Skeleton, and different ways of representing it are considered. The purpose of the grayscale skeletonization proposed in [36] is essentially to provide a fuzzy image analysis tool.

In [12, 38], the above skeletonization method is presented in terms of the Grayscale Morphology operations (see section 2.3, on page 20). The resulting grayscale Skeleton is essentially the same as in [36], but, in addition, reconstruction of the original function from the Skeleton is considered.

As a direct generalization of the discrete-family morphological Skeleton (section 3.1.2, on page 34), each grayscale skeleton "subset" S_n (which is, in this case, a grayscale image) is defined in [38] by:

$$S_n \triangleq f \ominus ng - (f \ominus ng) \circ g \tag{4.1}$$

where f is the original grayscale image, g is a structuring element (also a grayscale image), $-$ denotes the usual difference between functions, and ng is given by:

$$ng \triangleq \underbrace{g \oplus g \oplus \cdots \oplus g}_{n \text{ times}} \tag{4.2}$$

Fig. 4.1 presents an example of the first steps in a skeletonization process, and the associated first "subsets" of the Grayscale Skeleton of the image "House".

Although structurally identical, the calculation of the binary Morphological Skeleton (line 3 in Table 3.2, on page 34) and the calculation of the grayscale Skeleton (4.1) have between them a major difference: The minus sign $(-)$ in (4.1) denotes the *usual difference operation*

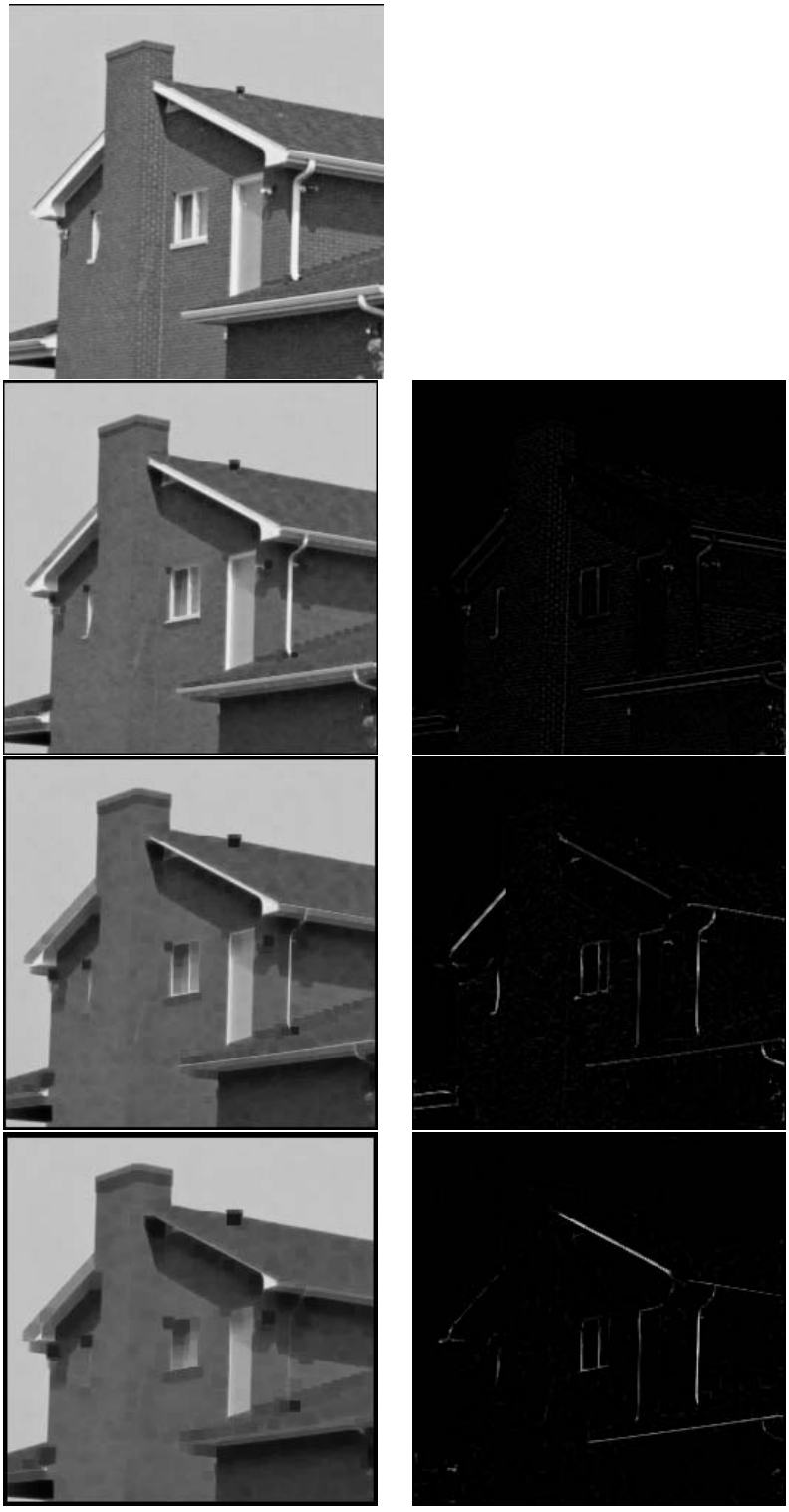


Figure 4.1: Grayscale skeletonization. Left side, up-down: Original image f, and its erosions $f \ominus ng$, $n = 1, 2, 3$, where g is a flat 3×3-squared structuring element. Right side, up-down: The first three Grayscale Skeleton "subsets", S_n, $n = 0, 1, 2$.

between functions, whereas in Table 3.2 it denotes *set-difference*. The former operation is not a generalization of the latter; actually, following the discussion in section 2.5.1 (page 28), there cannot be a direct generalization of the set-difference operation for functions.

As a consequence of the above, a direct generalization of the reconstruction formula (Table 3.3) cannot be obtained, i.e.:

$$f \circ mg \neq \bigvee_{n \geq m} S_n \oplus ng \tag{4.3}$$

where $\vee$ stands for the supremum in the Lattice of functions (see section 2.4.2, on page 26).

Instead, the following is used for perfect reconstruction:

$$f = \ldots [S_{N-2} + (S_{N-1} + S_N \oplus g) \oplus g] \oplus g \ldots \tag{4.4}$$

where N is the order of the highest non-null Skeleton "subset" S_n.

Another "imperfection" in the generalization of the Morphological Skeleton for functions concerns the basic concepts of *points* and centers, *elements*, and *maximal elements*, used in the Skeleton's definition. These concepts cannot be properly defined for functions, and therefore the Grayscale Skeleton cannot be seen as the collection of centers of maximal elements "inscribable in the image".

4.1.1 Grayscale Skeletons as Pyramids

Although theoretically "imperfect", in the sense discussed in the last section, the Grayscale Skeleton does preserve most of the intuitive properties of the binary Skeleton, especially from the algebraic point of view.

Notice that the images in Fig. 4.1 do *not* present a Grayscale Skeleton condensed in a single image, like a fuzzy *medial axis representation* of the original image (which is the way presented in [36]). Instead, the Skeleton "subsets", or "parts", are shown. This is because, from the algebraic point of view, we are not interested in *seeing a medial axis*, but rather in considering the *decomposition* obtained by the skeletonization. Similar to the binary Skeleton, each Grayscale Skeleton "subset" S_n contains those grayscale features of the original image with "size" (or "width) equal to n, where the concept of "size" is associated with the structuring element g.

Therefore, the Grayscale Skeleton (and also the Binary Skeleton) is a *hierarchical decomposition*, or in other words, a pyramid. Each "level" of the pyramid contains details of a certain "size". If we, in a generic way, associate the concept of *resolution* with *size*, meaning that the *bigger* a feature is, the *higher* is its resolution, then the Skeleton Representation can be seen as a *multi-resolution pyramid*.

In Chapter 9, a comparison between the Skeleton decomposition and the *Laplacian Pyramid* [5] is presented.

4.2 Generalized Grayscale Skeletons

One finds the same difficulties in generalizing the Proposed Skeleton framework (chapter 3) from a Boolean Lattice to the (non-boolean) Lattice of functions. We provide here two

approaches for obtaining a partial generalization, which are adapted from the two approaches presented in chapter 2, for generalizing Binary Morphology to Grayscale Morphology. The first is by means of the concept of *umbra*, and the second is by mean of $inf - sup$ operations. The latter leads to an adaptation of the grayscale Skeletons presented in the last section, whereas the former provides a Skeleton with less theoretical "imperfections".

4.3 The umbra approach

Suppose we are interested in defining a Skeleton decomposition for elements in the Lattice $\tilde{\mathcal{P}}_f$ of the functions $f : E \to \mathcal{R}$, where E is *any* set. The *supremum* and *infimum* of the above Lattice are the usual ones for functions.

We consider, then, another Lattice $\tilde{\mathcal{P}}$, this time Boolean, of all the subsets of $E \times \mathcal{R}$, i.e., $\tilde{\mathcal{P}} \stackrel{\triangle}{=} \mathcal{P}(E \times \mathcal{R})$. Here the supremum and infimum are union and intersection respectively, as in any Boolean Lattice in the form $\mathcal{P}(E)$, for some E. Each point in the above Boolean Lattice is in the form (x, z), with $x \in E$, $z \in \mathcal{R}$.

Let us define the *umbra transformation* $U(\cdot)$ from the Lattice of functions to the related Boolean Lattice as:

$$\begin{array}{l} U : \tilde{\mathcal{P}}_f \to \tilde{\mathcal{P}} \\ U(f) = \{(x, z) \in E \times \mathcal{R} \mid \ z \leq f(x)\} \end{array} \tag{4.5}$$

Similarly, we define the *envelope transformation* $W(\cdot)$ from the Boolean Lattice to the Lattice of functions as:

$$\begin{array}{l} W : \tilde{\mathcal{P}} \to \tilde{\mathcal{P}}_f \\ [W(Y)](t) = \bigvee\{z \in \mathcal{R} \mid \ (t, z) \in Y\} \end{array} \tag{4.6}$$

Notice that:

$$W[U(f)] = f, \quad U[W(Y)] \supseteq Y \tag{4.7}$$

Therefore, we say that the above Boolean Lattice *contains* the Lattice of functions.

In the above Boolean Lattice, a Skeleton decomposition of a set X is then defined with respect to a family of structuring functions which are umbræ, i.e., each $\delta_i : E \times \mathcal{R} \to \tilde{\mathcal{P}}$ has to satisfy for all $(x, z) \in E \times \mathcal{R}$: if $(x_0, z_0) \in \delta_i(x, z) \Rightarrow \forall z_1 < z_0, (x_0, z_1) \in \delta_i(x, z)$. Moreover, translation invariance in z is also required, i.e.: $\delta_i(x, z) = \delta_i(x, 0)_z$.

The Skeleton decomposition in the Lattice of functions of a function $f = W(X)$ is then defined by "projecting" the Skeleton decomposition defined in Chapter 3 in the Lattice of functions by means of $W(\cdot)$. In other words,

$$[\tilde{S}_i](t) \stackrel{\triangle}{=} [W(\tilde{S}_i)](t), \forall i \in I \tag{4.8}$$

where $\{[\tilde{S}_i](t)\}$ are the Grayscale Skeleton Subsets, and $\{\tilde{S}_i\}$ are the related Skeleton Subsets in $\tilde{\mathcal{P}}$.

Is $\{[\tilde{S}_i](t)\}$ a *Skeleton*, i.e., does it consist of "centers" of "maximal elements"? The answer is yes. Let us consider the family of functions $\{W[\delta_i(x, z)](t)\}_{i \in I}$ in $\tilde{\mathcal{P}}_f$. It is the "projection" of the family of structuring elements from the Boolean Lattice into the Lattice of functions. Each function $W[\delta_i(x, z)](t)$ is denoted an "element" with "radius" i and "center" (x, z). Notice that, since $\delta_i(x, z)$ is translation invariant in z, $W[\delta_i(x, z)](t)$ is equal to

$W[\delta_i(x,0)](t)+z$, i.e., offset-invariant. We say that an element is a *maximal element* under a given function f if:

- $W[\delta_i(x,z)](t) \leq f(t), \forall t \in E$.
- $\nexists y \in E, w \in \mathcal{R}$ and $j \in I$ such that:

$$W[\delta_i(x,z)](t) \leq W[\delta_j(y,w)](t) \leq f(t), \forall t \in E$$

Each subset $[\tilde{S}_i](t)$ consists of the "center" of all the "maximal elements" with "radius" i under the given function $f = W(X)$, therefore the resulting decomposition is indeed a Skeleton.

A version of the reconstruction formula (the evolution of which is described in Table 3.3) is also obtained for the above Skeleton of functions:

$$f(t) = \bigvee_{i \in I} W[\delta_i(\tilde{S}_i)](t). \tag{4.9}$$

where $\delta_i(\cdot)$ in (4.9) is the *dilation* associated with the structuring element $\delta_i(x,z)$. The proof is simple: In the Boolean Lattice, it holds: $X = \bigcup_{i \in I} \delta_i(\tilde{S}_i)$. Therefore, $W(X) = W[\bigcup_{i \in I} \delta_i(\tilde{S}_i)]$. Equation (4.9) is then obtained since $W(\cdot)$ is actually a dilation between the two Lattices, and since $W(X) = f$.

In summary, in order to decompose a function $f(t)$, its umbra is calculated and decomposed in the associated Boolean Lattice w.r.t. a family of structuring functions which are also umbræ (and translation-invariant in z). The resulting Skeleton Subsets are then projected back to the Lattice of functions.

4.4 The "sup-inf" Approach

In this approach, as opposed to the previous one, the calculation is performed entirely in the Lattice of functions.

Let $f \in \tilde{\mathcal{P}}_f$, and define the family of structuring functions $\{[g_i(x)](t)\}$, $i \in I$, $x \in E$. We suppose that the umbræ of f and the family of elements satisfy the conditions required by the Generalized Skeleton in the associated Boolean Lattice.

We derive for each structuring element $[g_i(x)](t)$, $i \in I$, the following operations in the Lattices of functions:

- Dilation:

$$\delta_i^g[f(t)] \stackrel{\triangle}{=} \bigvee_{x \in E} \{[g_i(x)](t) + f(x)\}$$

- Erosion:

$$\varepsilon_i^g[f(t)] \stackrel{\triangle}{=} \bigvee \{z \in \mathcal{R} \mid \ [g_i(x)](t) + z \leq f(t), \forall t \in E\}$$

- Opening:

$$\gamma_i^g[f(t)] \stackrel{\triangle}{=} \delta_i^g \varepsilon_i^g[f(t)]$$

- Closing:

$$\phi_i^g[f(t)] \stackrel{\triangle}{=} \varepsilon_i^g \delta_i^g[f(t)]$$

The Skeleton decomposition is then defined by its subsets $\{S_i^g(t)\}$, $i \in I$:

$$S_i^g(t) \stackrel{\triangle}{=} \varepsilon_i^g[f(t)] - \bigvee_{j>i} \gamma_{[i,j]}^g \varepsilon_i^g[f(t)] \tag{4.10}$$

where the *minus sign* relates to the usual minus operation between functions, and $\gamma_{[i,j]}^g[\cdot]$ is the opening associated to the structuring function $[\delta_{[i,j]}(x)](t) \stackrel{\triangle}{=} \{\varepsilon_i^g[g_j(x)]\}(t)$.

4.5 Comparison between the approaches

The morphological operations defined above for the Lattice of functions are actually the direct transposition of the morphological operations in the associated Boolean Lattice. Therefore, equation (4.10) can be written as:

$$S_i^g(t) = [W(\varepsilon_i(X)](t) - \left[W\left(\bigcup_{j>i} \gamma_{[i,j]} \varepsilon_i(X)\right)\right](t) \tag{4.11}$$

where $X = U(f)$, and the morphological operations are those of the associated Boolean Lattice. On the other hand, notice that, by definition (equation (4.8)), the umbra approach gives:

$$[\tilde{S}_i](t) = \left[W\left(\varepsilon_i(X) - \bigcup_{j>i} \gamma_{[i,j]} \varepsilon_i(X)\right)\right](t) \tag{4.12}$$

A comparison between (4.11) and (4.12) shows that the only fundamental difference between the approaches is in the nature of the subtraction operation.

Similarly, the umbra approach can be realized entirely in the Lattice of functions, by means of a formula similar to (4.10). Actually, one can use the same equation (4.10), just replacing the meaning of the subtraction operation by the following one:

$$\forall t \in E, f(t) - g(t) \stackrel{\triangle}{=} \begin{cases} f(t), & g(t) < f(t) \\ 0, & g(t) \geq f(t) \end{cases} \tag{4.13}$$

The above redefinition of the subtraction operation is the transposition into the Lattice of functions of the set-difference operation in the Boolean Lattice, followed by $W(\cdot)$.

Fig. 4.2 compares the result of grayscale skeletonization by each of the two approaches, for a simple 1-D discrete signal. For both Skeletons, a constant translation invariant generator g is used. The generator (structuring element) g is the flat function with region of support $\{-1, 0, 1\}$. In the umbra approach (Fig. 4.2(b),(c),(d)), the Skeleton "points" ("impulse" functions) have the same graylevels as the decomposition elements they represent. In the sup-inf approach (Fig. 4.2(e),(f),(g)), the graylevels of the Skeleton "points" are the numerical difference between the element they represent and their local background.

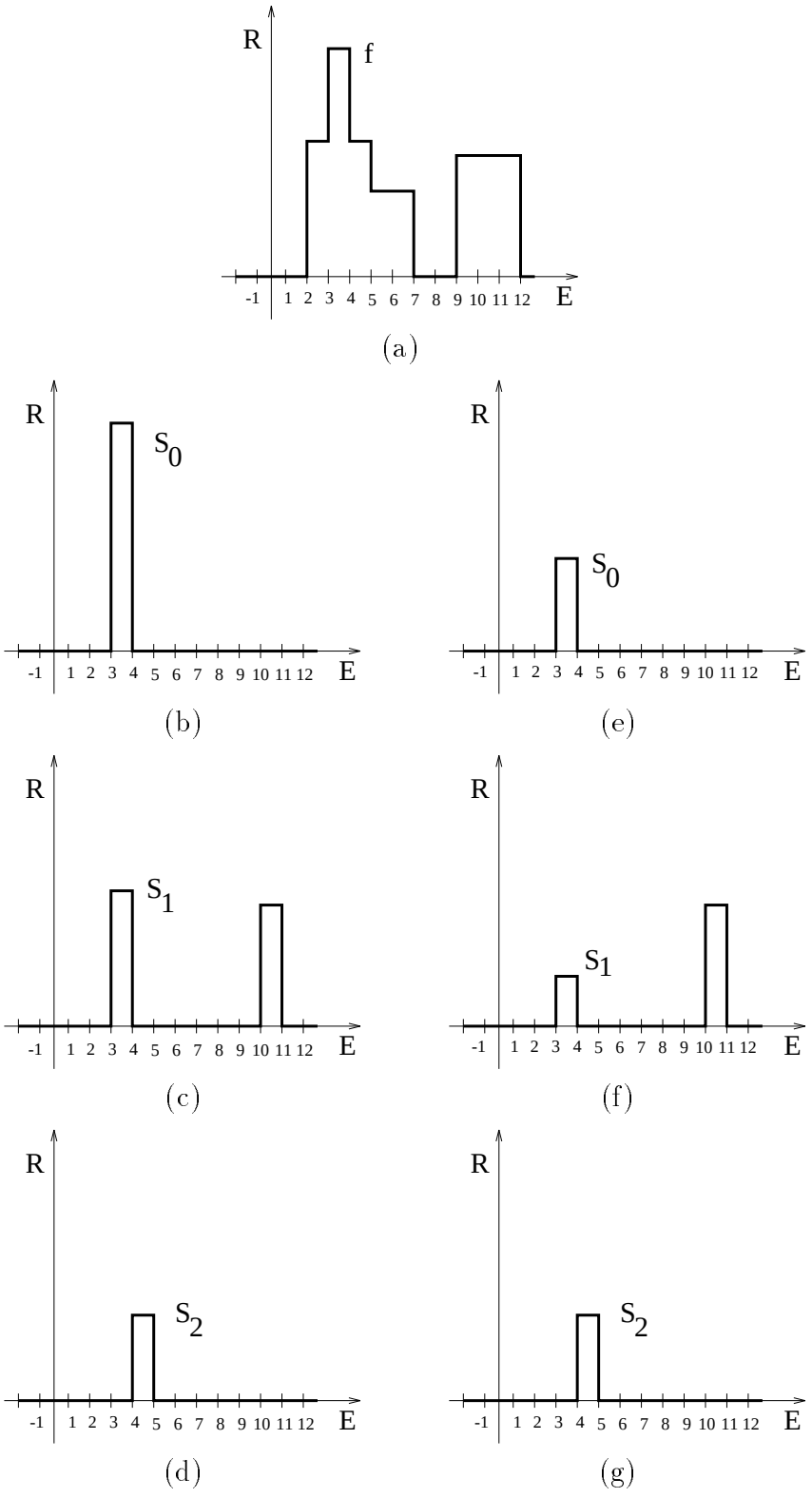


Figure 4.2: A comparison between the two Grayscale skeletonization approaches. (a) An 1-D discrete signal. (b), (c), and (d) The three "subsets" S_n, $n = 0, 1, 2$, according to the *umbra* approach. (e), (f), and (g) The three "subsets" according to the "sup-inf" approach.

Both approaches are imperfect generalizations of the Skeleton decomposition into the Lattice of functions. In the second approach (subtraction as the usual minus sign), the concept of "maximal element" is not properly defined, and the usual reconstruction formula is not valid. In the umbra approach, there is much redundancy in the resulting representation, because the sum of the energy of the decomposition elements is much higher than that of the original image, and of the Skeleton obtained by the sup-inf approach.

Chapter 5

Applications and Particular Cases of the Proposed Generalized Skeleton

5.1 Applications of the Proposed Generalized Skeleton

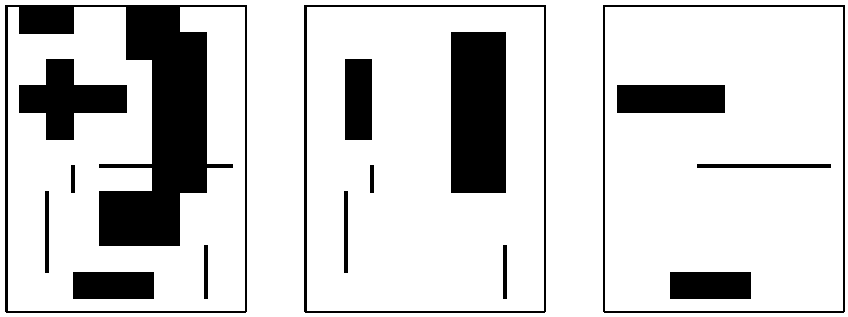
The applications of the one-parameter skeleton (I being a totally ordered 1-dimensional space), are well known. In this section, we present some applications of skeletons based on partially ordered sets, such as Multi-Parameter Skeletons.

From the *topological* and *geometrical* points of view, a Multi-Parameter Skeleton may have little interest. First, it is far less connectivity-preserving than the conventional skeleton. Also, it fails to exactly provide a *Medial Axis* of the shapes under study. And, finally, a Multi-Parameter Skeleton usually can *not* be characterized by a *quench function*, because a point $x \in E$ may be the "center" of two (or more) different maximal elements. The latter is consequence of I being *partially* ordered.

On the other hand, from the *algebraic* point of view, a Multi-Parameter Skeleton can be very useful. First, it can decompose an image into an assortment of shapes richer than the one a conventional skeleton is able to provide. Moreover, if we consider the element indices i to be a degree of "importance", or as a "category classifier", as is often done regarding the conventional skeleton, then a Multi-Parameter Skeleton can provide finer distributions and classifications. In addition, there is greater diversity of possible partial reconstructions, where their proximity to the original image are controlled by the choice of the anti-umbra J in (3.10). Finally, each of the scalar parameters of the multi-dimensional index i may have a different physical interpretation, such as size, time duration, gray-level, etc., in contrast to the conventional skeleton decomposition, where different physical characteristics of the image can not be treated independently.

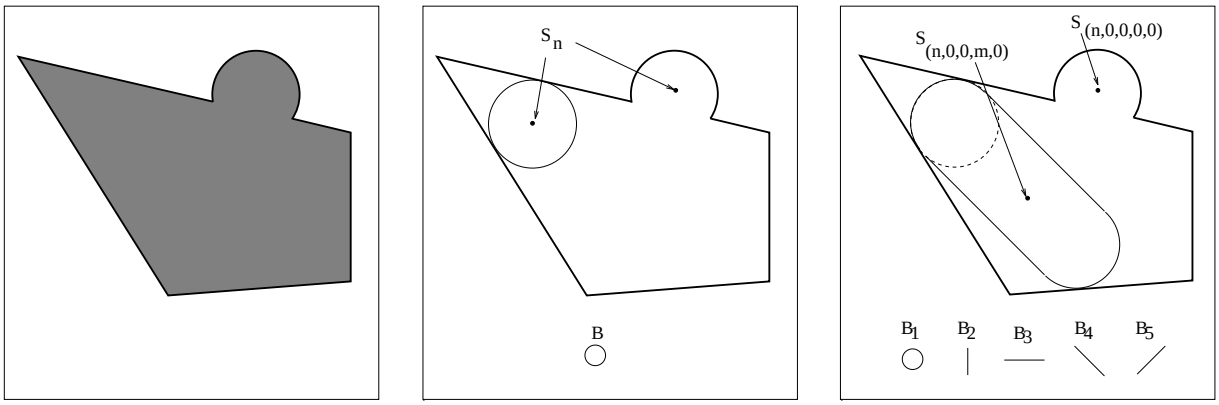
5.1.1 Shape Classification

Let us consider a *MSES* representation of a binary image composed by several objects. By considering the relation between the projections n_ℓ, $\ell = 1, \ldots, d$, of the index $\vec{n}$ associated to a MSES point, one can decide whether the related maximal element is similar to either of the family generators B_ℓ, or none of them.



(a) (b) (c)

Figure 5.1: Shape Classification via an MSES: (a) An image composed of rectangular features. (b) Features for which $C_1 \geq 75\%$. (c) Features for which $C_2 \geq 75\%$.



(a) (b) (c)

Figure 5.2: (a) A binary image containing a partially occluded disc, (b) elements detected by a conventional skeleton, (c) elements detected by a 5-parameter skeleton.

Let the *concentration* C_ℓ of the generator B_ℓ for an MSES point with "radius" $\vec{n} = (n_1, \ldots, n_d)$ be defined as $C_\ell \triangleq n_\ell / (\sum_{m=1}^{d} n_m)$. It can be seen as a measure of the similarity of a maximal element to the structuring element B_ℓ.

For example, Fig. 5.1(a) shows an image X composed of several rectangular features. A two-parameter MSES representation of X, with the rectangular family shown in Fig. 3.4(a), was calculated. Fig. 5.1(b) shows the maximal elements corresponding to MSES points for which $C_1 \geq 0.75$. Those features are similar to B_1 (vertical line). Fig. 3(c) shows maximal elements corresponding to Minimal MSES points for which $C_2 \geq 0.75$. Those features are similar to B_2 (horizontal line).

5.1.2 Pattern Recognition

Suppose we are interested in finding a pattern in a binary image, and that this pattern is not corrupted with holes but may be partially occluded. E.g., the disc in Figure 5.2(a).

In order to detect the shape, let us consider a family of translation invariant structuring-functions, with the structure shown in (3.14). If the pattern we are interested in is one of the

shapes of the family $\{A_i\}$, then it should be easy to locate such a pattern using the associated skeleton decomposition, since it provides the centers of maximal elements from that family in the given image. In other words, we consider patterns to be located as maximal elements, and define a proper family of shapes $\{A_i\}$ for decomposition. Thus, in order to detect the disc in Fig. 5.2, we may calculate the skeleton w.r.t. a family of increasing discs.

However, for a conventional skeleton, the above idea does not work well. In Fig. 5.2(b) we see that we may find, in the subsets S_n of the skeleton, maximal elements other than the pattern we are looking for.

On the other hand, a MSES could give better results. For the example in Fig. 5.2, we choose a 5-parameter skeleton, based on 5 structuring-elements: a unit disc (which we want to detect), and 4 unit lines in four different directions. In this case, we are interested in the subsets of the form $S_{(n,0,0,0,0)}$ only. As seen in Fig. 5.2(c), most of the "false alarms" obtained by the conventional skeleton are now in different subsets than $S_{(n,0,0,0,0)}$.

5.1.3 Coding Using a Hybrid Skeleton

Let us compare a Multi-Parameter Skeleton, w.r.t. a partially ordered family of structuring-functions $\mathcal{F}$, to a conventional one-parameter skeleton w.r.t. a totally ordered sub-family $\tilde{\mathcal{F}}$ *contained* in $\mathcal{F}$. For example, if we consider the family of open rectangles shown in Fig. 3.4(a) to be $\mathcal{F}$, then $\tilde{\mathcal{F}}$ could be the family of open squares. Because of the partial ordering, the number of skeleton points in the multi-parameter skeleton is expected to be larger than the number of skeleton points in the one-parameter skeleton. But after removing redundant points in both skeletons (see chapter 6), the situation is inverted; the multi-parameter skeleton is expected to contain considerably fewer points than the one-parameter skeleton, which could be of great advantage for Coding purposes.

However, since the number of subsets in the multi-parameter skeleton is usually much bigger than the number of subsets in the one-parameter skeleton (about N^d in comparison to N), this turns out to be too costly in terms of coding efficiency. Moreover, its computational burden, usually of order $\mathcal{O}(d)$, is very high if compared to $\mathcal{O}(1)$ of the one-parameter skeleton.

However, the General Skeleton Decomposition presented in section 3.3 does not restrict us to either a one-parameter or a multi-parameter skeleton; combinations of them are also possible. For example, instead of considering a decomposition w.r.t. the family of *all* the rectangles, as presented in section 3.2, or w.r.t. the family of squares only, We can *arbitrarily* select *any* sub-family of rectangles for the skeletonization. This combines, at some extent, the advantages of both the multi-parameter and the one-parameter skeletons. We call such decomposition a Hybrid Skeleton.

As opposed to a full multi-parameter decomposition, which in the general case is not substantially advantageous when compared to the one-parameter skeleton, Hybrid Skeletons showed promising results in preliminary simulations. For the facsimile image in Fig. 5.3, for example, the standard Run-length+Huffman coder [29] gives a compression of 0.34 bits per pixel for the MSES w.r.t. the hybrid Skeleton, compared to 0.42 bits per pixel for the original Skeleton w.r.t. the family of squares.[1].

[1]Both Skeletons had their redundancy removed, by the algorithm proposed in [29], previous to coding. See chapter 6 for details.

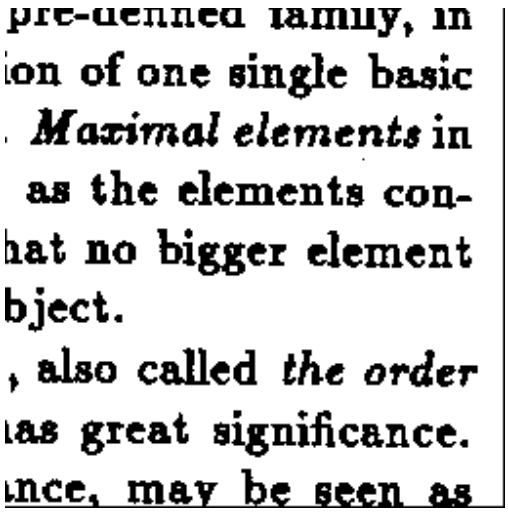


Figure 5.3: A facsimile image. Its Hybrid Skeleton decomposition is coded with higher compression rate than its original Skeleton decomposition, for the Run-Length/Huffman coder.

5.1.4 Filtering by Partial Reconstruction

Discarding lower-order Skeleton Subsets in the reconstruction process produces simplified (morphologically opened) versions of the original image, as seen in Table 3.3. Usually, a Multi-Parameter Skeleton has a greater variety of options for filtering the original image by partial reconstruction than a one-parameter Skeleton.

One can see that discarding $S_{0,0}(X)$ alone from a two-parameter MSES produces the same result as the union of the openings by the 2 structuring elements B_1 and B_2. This is a more selective filtering than the opening by a single structuring element. Generally, for an MSES with d structuring elements, the removal of the lowest MSES subset gives the union of the openings by each of the d structuring elements, which was shown to have good noise cleaning properties in [56].

Many other morphological filters can be obtained by removing different combinations of lower-order subsets. Fig. 5.4 shows the result of removing the MSES subsets with $n_1+n_2 < 2$, in contrast to the result of removing the original Skeleton subsets with $n < 1$, when the MSES is calculated with the family of discrete rectangles (Fig. 3.4(a)) and the original Skeleton with the family of discrete squares.

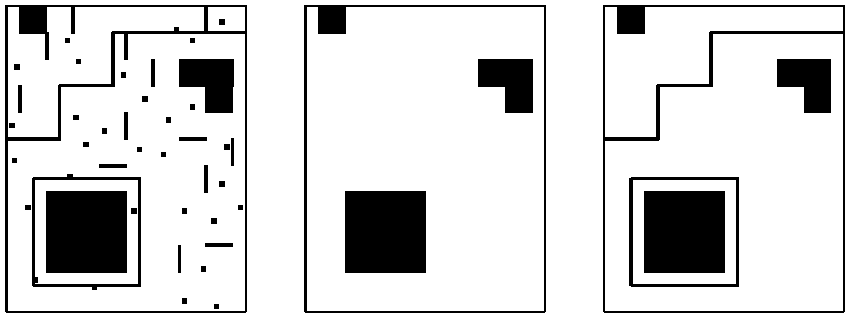
5.1.5 Image Analysis

The Regular Skeleton is closely related to a *morphological pattern spectrum* [30]. The pattern spectrum conveys geometrical information which can be further analyzed and processed. The discrete morphological pattern spectrum is defined in [30] as:

$$PS_n^B(X) = \#[X \circ nB - X \circ (n+1)B].$$

where $\#(\cdot)$ denotes finite set cardinality.

The MSES, because of its multi-parameter structure, may be seen as closely related to a *multi-dimensional pattern spectrum*, which contains the 1-dimensional pattern spectrum and



(a) (b) (c)

Figure 5.4: Image Filtering by partial reconstruction: (a) Noisy image. (b) Filtered reconstruction from the original Skeleton ($n \geq 1$). (c) Filtered reconstruction from the MSES ($n_1 + n_2 \geq 2$).

conveys more and finer details for analysis. The same type of generalization used to generate the *2-element MSES* may be used to define a *2-dimensional pattern spectrum*:

$$PS^{B_1,B_2}_{n_1,n_2}(X) = \#\{X \circ (n_1 B_1 \oplus n_2 B_2) - (X \circ [(n_1+1)B_1 \oplus n_2 B_2]) \cup (X \circ [n_1 B_1 \oplus (n_2+1)B_2])\}$$

5.2 Special Cases of the Generalized Skeleton Decomposition

This section shows how the Quadtree Decomposition of binary images and the Bit-Plane decomposition of Grayscale pictures can both be seen as particular cases of the Generalized Skeleton Decomposition.

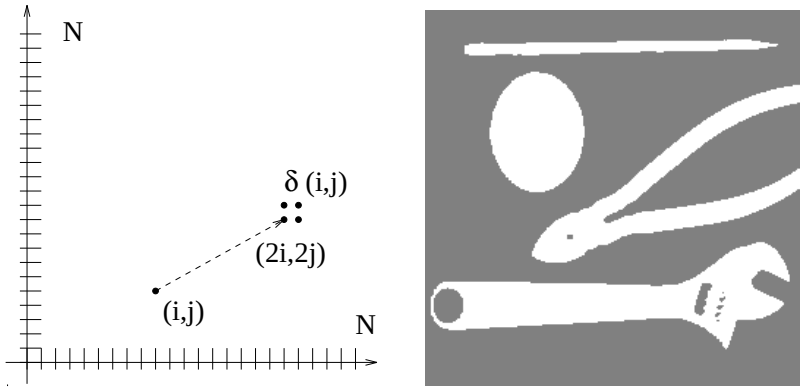
Apart from its theoretical relevance, in relating image representations thought previously to be unrelated, this result makes it possible to obtain *multi-parameter* generalizations of the Quadtree and the Bit-Plane decompositions, supported by the generalized framework of Skeleton Decomposition. These generalizations are also presented in this section.

5.2.1 Quadtree Decomposition

Let us select the Boolean Lattice as the set $\mathcal{P}(\mathcal{N}^2)$ of all the subsets of $\mathcal{N}^2$, where $\mathcal{N}$ is the set of natural numbers, and let the structuring-function $\delta : \mathcal{N}^2 \rightarrow \mathcal{P}(\mathcal{N}^2)$ be (as depicted in Figure 5.5(a)) defined by:

$$\begin{aligned} \delta(i,j) \;=\; & \{(2i,2j),(2i+1,2j), \\ & (2i,2j+1),(2i+1,2j+1)\}, \\ & (i,j) \in \mathcal{N}^2 \end{aligned} \tag{5.1}$$

The *dilation* $\delta(X)$ derived from the above structuring function is equivalent to an *interpolation* process; it first upsamples the input binary image X, and then fills the gaps between



(a) (b)

Figure 5.5: (a) The structuring-function for the Quadtree Decomposition. (b) Original binary 256×256-pixel image.

samples by dilating it (in the translation-invariant sense, i.e., performing $\oplus$) by a 2×2-pixel squared structuring element. Therefore, the adjoint *erosion* $\varepsilon(X)$ is equivalent to a *decimation* process, where X is first eroded (in the translation-invariant sense - $\ominus$) by the same 2×2 pixel structuring element as above, and then downsampled. The related *opening* $\gamma(X)$ is the result of "decimation" followed by "interpolation".

A constant generator family of structuring functions $\{\delta_n(x)\}$ is obtained by the n-fold composition of $\delta(x)$ with itself. I.e.,

$$\delta_n(x) = \begin{cases} (\delta)^n(x), & n > 0 \\ \{x\}, & n = 0 \end{cases} \tag{5.2}$$

The decomposition of a binary image, using the Generalized Skeleton (3.9) with the above family, is the Quadtree decomposition *of the foreground* of the image. For example, Figure 5.5(b) shows a binary image, and Figure 5.6(a) shows its Skeleton subsets S_n, $n = 0, 1, \ldots$. The black pixels are the Skeleton Points; each representing a maximal element. In this case, maximal elements are squares of sizes $2^n \times 2^n$ pixels. Figure 5.6(b) shows the maximal elements for the above image.

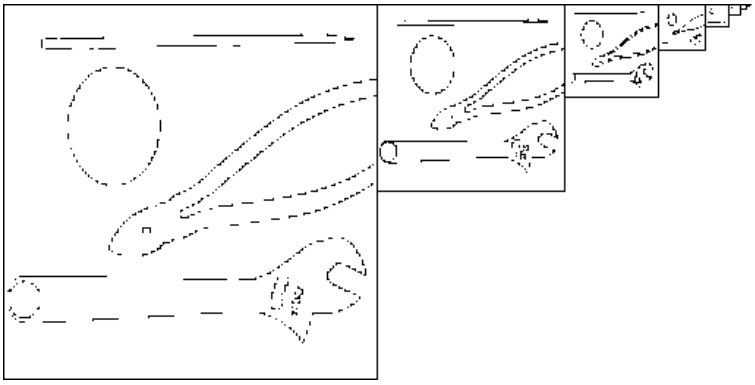
By inverting each pixel of the image (interchanging foreground and background) and applying the above decomposition again, one obtains the Quadtree decomposition *of the background*. The corresponding maximal elements are shown in Figure 5.6(c). Full Quadtree representation consists of both foreground and background decompositions.

5.2.2 Bit-Plane Decomposition

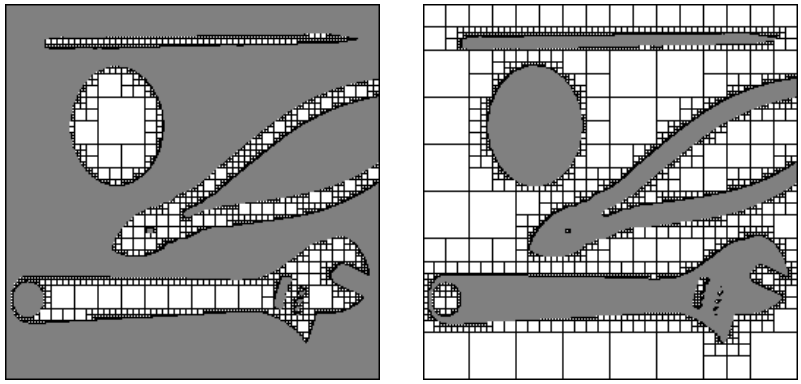
For a Bit-Plane decomposition, we select the Lattice of *grayscale functions* complemented with ∞, i.e., functions with values in the range $\{0, 1, \ldots, 255\} \cup \{\infty\}$.

We select the following dilation $\delta(f)$ in the above Lattice (f being a function):

$$\delta(f(x)) = \begin{cases} 2f(x), & 0 \le f(x) < 128 \\ \infty, & f(x) \ge 128 \end{cases} \tag{5.3}$$

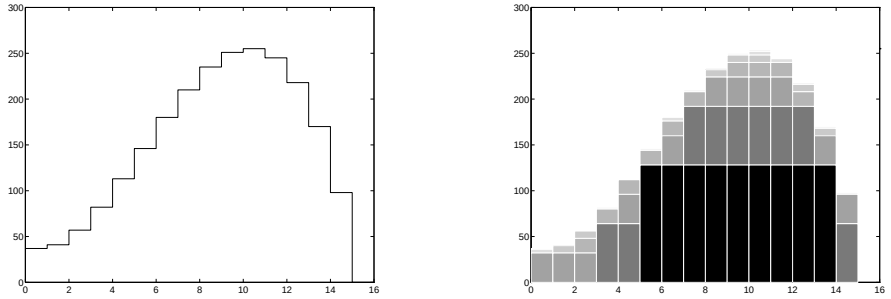


(a)



(b) (c)

Figure 5.6: Quadtree decomposition as a Skeleton. (a) Skeleton Subsets S_n, $n = 0, 1, \ldots$, of the foreground decomposition, (b) Maximal Elements (foreground decomposition), (c) Maximal Elements (background decomposition).



(a) (b)

Figure 5.7: Bit-Plane Decomposition as a grayscale Skeleton. (a) Original discrete function, (b) decomposition into "Maximal Elements" (the rectangles); each "Maximal Element" represents a "1" in the appropriate Bit-Plane (rectangles of the same color correspond to same Bit-Plane).

If we consider an 8-bit-plane image f (i.e., a function with values in the range $\{0, \ldots, 255\}$), then the corresponding erosion is $\varepsilon(f) = \lfloor f/2 \rfloor$, and the opening is $\gamma(f) = 2\lfloor f/2 \rfloor$. The sets S_n of the Grayscale Skeleton in this case, are the Bit Planes of f.

For example, Figure 5.7(a) shows an one-dimensional discrete function. Figure 5.7(b) shows the result of the decomposition. Each decomposition "rectangle" has a height 2^m, $m = 0, 1, \ldots, 7$ (note that rectangles with same m have same color), where each "rectangle" represents a "1" (Skeleton Point) in the appropriate Bit-Plane m (Skeleton subset). Although maximal elements can not be defined in this case (see section 4.1), the above "rectangles" offer a suitable approximation.

5.2.3 Generalized Quadtree Decomposition

Let us define the following structuring functions:

$$\delta_1(i,j) = \{(i,2j), (i,2j+1)\} \tag{5.4}$$

$$\delta_2(i,j) = \{(2i,j), (2i+1,j)\}. \tag{5.5}$$

The above structuring functions are depicted in Figure 5.8.

We define a 2-parameter Quadtree decomposition, by using equation (3.9) with the following family:

$$\delta_i(x) = (\delta_1)^{i_1}(\delta_2)^{i_2}(x) \tag{5.6}$$

where $i = (i_1, i_2) \in \mathcal{N}^2$.

The above Generalized Quadtree decomposes a binary image into *rectangles* rather than *squares* (see Figure 5.9).

As opposed to the original Quadtree decomposition, where the decomposition squares are disjoint, the decomposition rectangles of the Generalized Quadtree may overlap (see the light grey areas in Figures 5.9(b) and 5.9(c)). On the other hand, as opposed to standard Skeleton decompositions, where some Skeleton points are redundant and may be removed, there are no redundant rectangles. The number of decomposition rectangles is usually much smaller

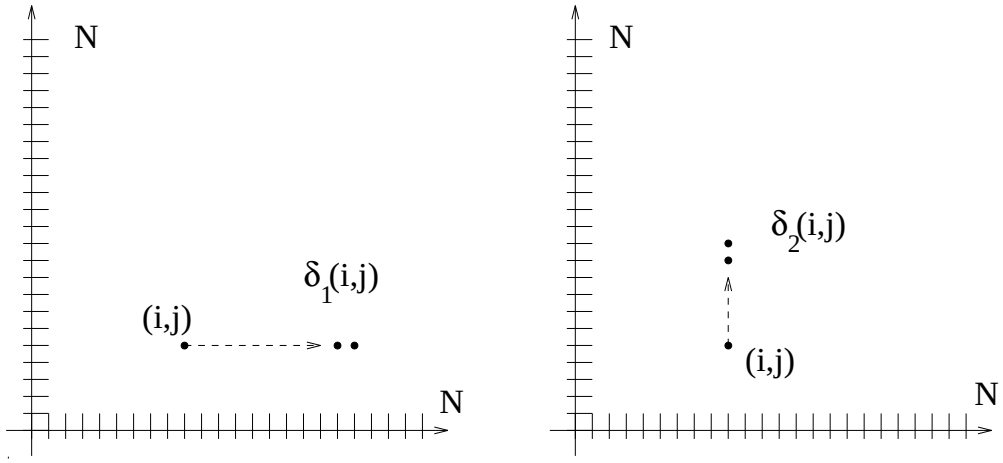


Figure 5.8: Structuring functions for the proposed Generalized Quadtree Decomposition.

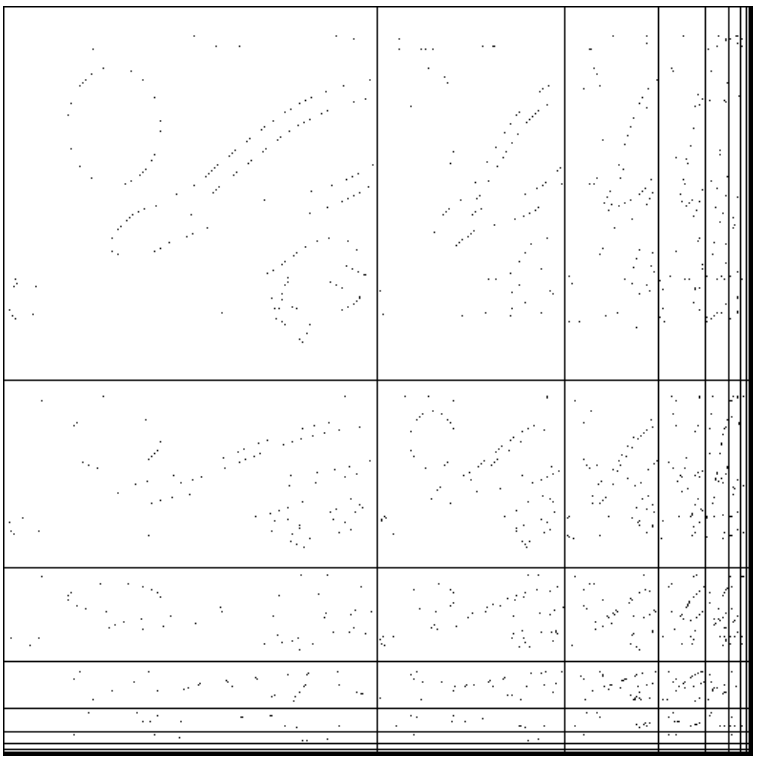
than the number of decomposition squares, as can be seen when comparing Figure 5.6(a) with Figure 5.9(a). This has potential use in coding.

5.2.4 Bit-Plane–Quadtree Decomposition

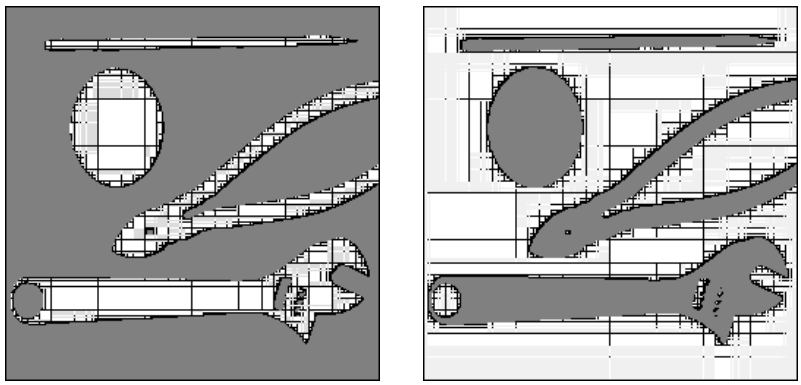
Consider the dilation $\delta(X)$ which we used for performing the standard Quadtree Decomposition in section 5.2.1. As mentioned in section 5.2.1, this $\delta(X)$ is an "interpolation", obtained by upsampling the image, followed by a translation invariant dilation ($\oplus$) with a 2×2-squared structuring element. Let us adapt the above "Quadtree" dilation $\delta(X)$ (hence also the erosion, opening and closing) so that it is applied to *grayscale functions*. We do this simply by replacing the above *binary* translation invariant dilation ($\oplus$) with a *grayscale* one, with a 2×2-pixel flat structuring element.

A *Bit-Plane–Quadtree Decomposition* can be defined, by using Lantuéjoul's generalized equation (3.9) with the family $\{\delta_i(x)\}$, $i = (i_1, i_2)$, as defined in (5.6), where δ_1 is selected to be the "Bit-Plane dilation" (5.3), and δ_2 is selected to be the above "Grayscale Quadtree dilation" ("interpolation").

This provides a decomposition of a *grayscale* image into squares of size 2^{i_2} and graylevel 2^{i_1} each, where $i_1, i_2 \in \mathcal{N}$. The number of such squares is usually much smaller that the number of "1"s in the Bit-Plane decomposition. This also could be of potential use in coding.



(a)



(b) (c)

Figure 5.9: Proposed Generalized Quadtree decomposition. (a) Generalized Skeleton Subsets $S_{n,m}$, $(n,m) \in \mathcal{N}^2$, for the foreground decomposition, (c) Maximal Elements (foreground decomposition). The light grey areas represent overlapping of the decomposition rectangles, (d) Maximal Elements (background decomposition).

Chapter 6

Generic Approach for Morphological Reduction of Skeleton Redundancy

In this chapter we study *morphological* methods to reduce the amount of redundant points in the Skeleton representation of images. The advantage of removing redundant points using *morphological operations only*, lies in the computational efficiency of these operations, when implemented on parallel machines.

6.1 Background

As pointed out in chapter 1, the Skeleton is a *redundant representation*, i.e., some of its points may be discarded without affecting its error-free representation characteristic. From the algebraic point of view, it is of interest to remove redundant Skeleton points, so that the representation contains as few as possible points.

For this purpose, Maragos and Schafer defined in [29] a *Minimal Skeleton* as being any set of points from the Skeleton which *fully represents* the *original* image and does not so if any of its points is removed. A Minimal Skeleton always exists since in the worst case it is the Skeleton itself. On the other hand, there can be more than one Minimal Skeleton for a given image. Fig. 6.1(a) shows a binary picture and its Morphological Skeleton computed with a 3x3 square structuring-element. Fig. 6.1(b)-(c) show two of the Minimal Skeletons of Fig. 6.1(a).

Maragos and Schafer propose in [29] an algorithm for finding a Minimal Skeleton from the Skeleton representation of a binary image. However, this algorithm is not fully morphological and therefore cannot be efficiently implemented on a parallel machine, in contrast to the Morphological Skeleton itself which is amenable to a parallel implementation. A fully morphological algorithm for finding Minimal Skeletons could take advantage of the parallel properties of the morphological operations and perform the computation in a more efficient way.

Sapiro and Malah defined in [46, 48] (see also [47]) an *Essential Point* of the Skeleton as any Skeleton point that cannot be removed from the original Skeleton without affecting its error-free property. The Essential Points are contained in any Minimal Skeleton, although usually are not sufficient for exact reconstruction. The set of Essential Points is unique and

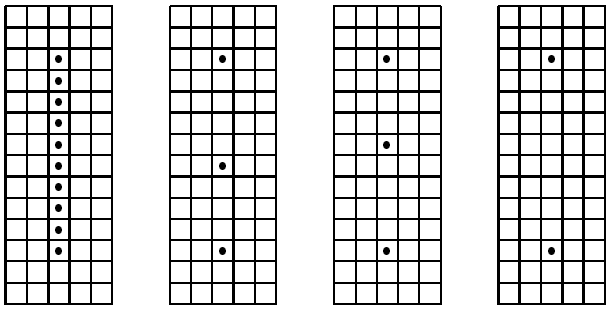


Figure 6.1: (a) A shape and its Skeleton (computed with a 3x3 square), (b)-(c) two of its Minimal Skeletons, (d) the Essential Points.

it is typically the major part of Minimal Skeletons (90% and more) [48]. Because of the above properties, Sapiro and Malah suggested in [48] that the Essential Points of the Skeleton should be found first, and then the remaining Minimal Skeleton points could be searched for in a more efficient way. The Essential Points of the shape in Fig. 6.1(a) are shown in Fig. 6.1(d). Notice that they are present in the two Minimal Skeletons shown in the figure.

Another important related topic is the "Reduced Skeleton" defined by Maragos in [30]. The Reduced Skeleton has fewer points than the regular Skeleton and it is also error-free. It is not a Minimal Skeleton but it is obtained by morphological operations only. (The mathematical definition is reviewed in section 6.4.1 below).

In this chapter, first a classification of the redundant points of a generic morphological representation into categories is proposed in section 6.2, and its specific relation to the Skeleton is considered. Then, section 6.3 considers the Essential Points: Their mathematical definition, a theorem stating that the Essential Points are the intersection of all the Minimal Skeletons, and the definition of a new type of Essential Points: *Local Essential Points*. Moreover, it is proposed an approach which takes into the account the above classification of the redundant points, and leads to a Reduced Skeleton which has less points than Maragos's Reduced Skeleton and is also error-free. And finally, morphological formulæ for calculating the Essential Points are also presented. The approach is suitable for any particular case of the Generalized Skeleton Representation presented in chapter 3.

6.2 Redundancy Classification

The concepts discussed in this chapter are suitable for both binary and grayscale images. However, we consider in this thesis only the binary case. The images can be sets in any Boolean Lattice $\mathcal{P}(E)$; in particular they may be continuous sets in $E = \mathcal{R}^2$, or discrete sets in E=$\mathcal{Z}^2$.

6.2.1 Types of Redundant Points

Let I be any totally or partially ordered set of indices, and let us consider a collection of subsets $\{T_i\}$, $i \in I$, which represents a given set $X \in \mathcal{P}(E)$ in the following way:

$$X = \bigcup_{i \in I} \delta_i(T_i). \tag{6.1}$$

where $\{\delta_i(X)\}$, $i \in I$, is a given family of dilations in $\mathcal{P}(E)$.

A point t belonging to the subset of index i, $t \in T_i$, represents the element $\delta_i(t) \subseteq X$.

If $t \in T_i$ is **redundant**, then the element it represents $(\delta_i(t))$ is contained in a region represented by some or all the other representation points, i.e.,

$$\delta_i(t) \subseteq \left(\bigcup_{j \neq i} \delta_j(T_j) \right) \cup [\delta_i(T_i - \{t\})] \tag{6.2}$$

We propose to classify each of the redundant points into one or more of the following **redundancy categories**:

Single-Element Redundancy: if there exists at least *one* element with index **greater** than i that covers $\delta_i(t)$, i.e.:

$$\exists j > i, \quad \exists y \in T_j \mid \delta_i(t) \subset \delta_j(y). \tag{6.3}$$

Higher-level Redundancy: if there exists a *union* of elements with index **greater** than i that covers $\delta_i(t)$, i.e.:

$$\delta_i(t) \subseteq \bigcup_{j > i} \delta_j(T_j). \tag{6.4}$$

Note that every "Single-Element" redundant point is also a "Higher-level" redundant point.

Lower-level Redundancy: if there exists a *union* of elements with index **smaller** than i that covers $\delta_i(t)$, i.e.:

$$\delta_i(t) \subseteq \bigcup_{j < i} \delta_j(T_j). \tag{6.5}$$

Higher/Lower-level Redundancy: if there exists a union of elements with indices **greater** or **smaller** than i that covers $\delta_i(t)$, i.e.:

$$\delta_i(t) \subseteq \bigcup_{j<i \text{ or } j>i} \delta_j(T_j). \tag{6.6}$$

Hence, every higher-level or lower-level redundant point is also a higher/lower-level redundant point.

Interlevel Redundancy: if there exists a union of elements with indices **different** from i that covers $\delta_i(t)$, i.e.:

$$\delta_i(t) \subseteq \bigcup_{j \neq i} \delta_j(T_j). \tag{6.7}$$

Hence, every higher/lower-level redundant point is also an interlevel redundant point.

Intralevel Redundancy: if the redundant point is not Interlevel redundant, i.e.,

$$\delta_i(t) \not\subseteq \bigcup_{j \neq i} \delta_j(T_j). \tag{6.8}$$

In this case, *every* set of elements (excluding $\delta_i(t)$) that covers $\delta_i(t)$ contains at least one element with index i.

Total Ordering

An important particular case is obtained when I is a *totally* ordered family of indices. In this case, we associate this totally ordering to the ordering in which the elements are calculated, i.e., we suppose that the lower-order elements are calculated *before* the higher-order elements, sequentially. This gives a "temporal" meaning to the index i.

Therefore, if I is totally ordered, we denote the Higher-level Redundancy also by *Future-level Redundancy*, and the Lower-level Redundancy by *Past-level Redundancy*. In this case, the Higher/Lower-level Redundancy and the Interlevel Redundancy are identical, and we use the name of the latter.

6.2.2 The Morphological Skeleton and Its Redundancy

The Generalized Morphological Skeleton representation of a binary image X (see section 3.3), being the set of points which are centers of maximal elements, does not contain redundant points from the "Single-Element" category, i.e., it does not contain "Single-Element Redundancy". On the other hand, it may contain redundant points from all the other categories.

For demonstration, Fig. 6.2 shows a continuous binary image composed by the union of 2 disks, P and Q, which are centered at the points p and q, respectively. The Skeleton of the shape, computed with $\delta_i(X) \stackrel{\triangle}{=} X \oplus iB$, where iB is a disc with radius $i \in I \stackrel{\triangle}{=} \mathcal{R}^+$ (totally ordered), is the segment $[p, q]$. In this case, all the skeleton points are redundant, except for p and q. The point a in Fig. 6.2(a) is a "Future-level" redundant point, because the element it represents (the dotted disk) is contained in the union of 2 bigger maximal disks (P and Q). The point b in Fig. 6.2(b) is "Interlevel" redundant, because it represents a disk (the dotted one) which in this example is contained in the union of a bigger maximal disk (Q) and a smaller maximal disk (P). The point c in Fig. 6.2(c) is "Intralevel" redundant, because the dotted disk, which it represents, is contained in the union of a larger maximal disk (Q) and a maximal disk with the same size (P), and it is not contained in any union of only larger and smaller maximal disks.

Similarly, Multi-Parameter Skeletons can contain Higher-level, Lower-level, Higher/Lower-level, Interlevel and Intralevel Redundancy.

6.3 Essential Points

An *Essential point* of a Skeleton Representation is defined to be a point of the Skeleton which, if it is removed from the original Skeleton, makes the exact reconstruction impossible

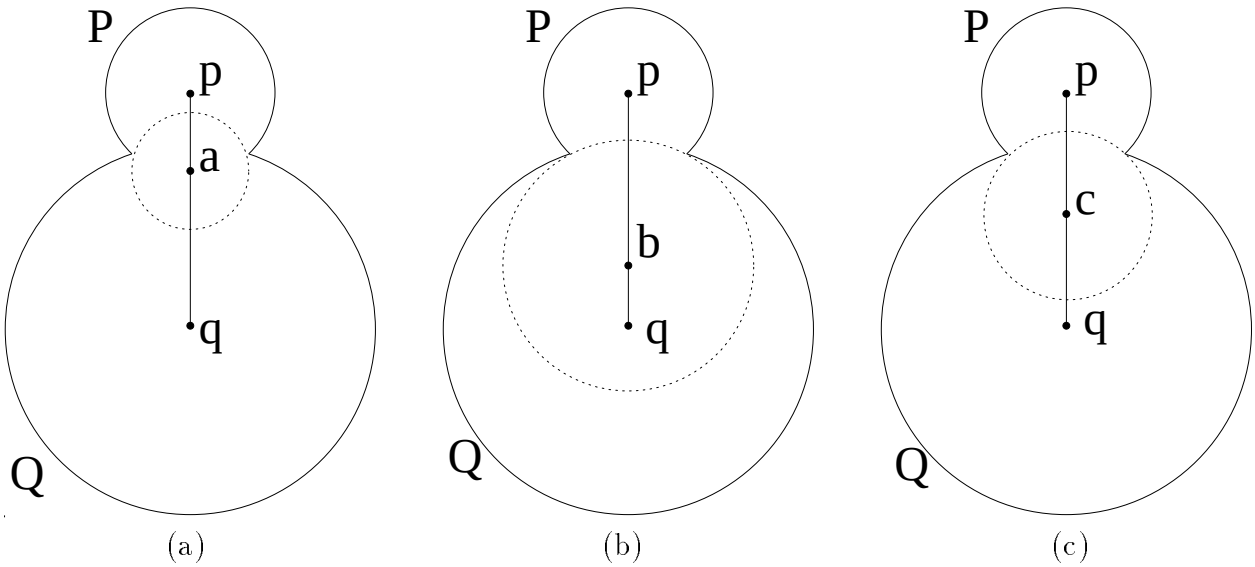


Figure 6.2: Types of redundant points in the Skeleton. (a) A binary image composed by two disks (P and Q), and its Skeleton (the segment $[p, q]$). The point a is a "Future-level" redundant point, (b) the point b is a "Interlevel" redundant point, and (c) the point c is a "Intralevel" redundant point.

[46, 48]. More specifically, a point t belonging to the Skeleton subset S_i is a *Essential Point of order i* iff[1]:

$$\left(\bigcup_{j \neq i} \delta_j(S_j)\right) \cup \delta_i(S_i - \{t\}) \neq X \tag{6.9}$$

As an example, Fig. 6.3(a) shows a binary image, Fig 6.3(b) shows its Morphological Skeleton (computed with a 3x3 square as structuring-element), Fig. 6.3(c) shows a Minimal Skeleton and Fig. 6.3(d) shows its Essential Points (which is a subset of the Minimal Skeleton).

Theorem 3 *The Essential Points of a* discrete *Skeleton are the intersection of all its Minimal Skeletons.*

Proof *Since, obviously, the Essential Points are a subset of any Minimal Skeleton, it is sufficient to prove that there does not exist a point p, belonging to all the Minimal Skeletons, which is not an Essential Point.*

Suppose, by contradiction, that there is such a point p. Since it is not an Essential Point, the Skeleton representation without it is still error-free. If we go on removing other Skeleton points, at some point we will obtain a Minimal Skeleton, since the original Skeleton is discrete and therefore it has a finite number of points. The obtained Minimal Skeleton does not contain the point p, and therefore p does not belong to the intersections of all the Minimal Skeletons. □

[1]The definition presented here is a generalization of that in [48].

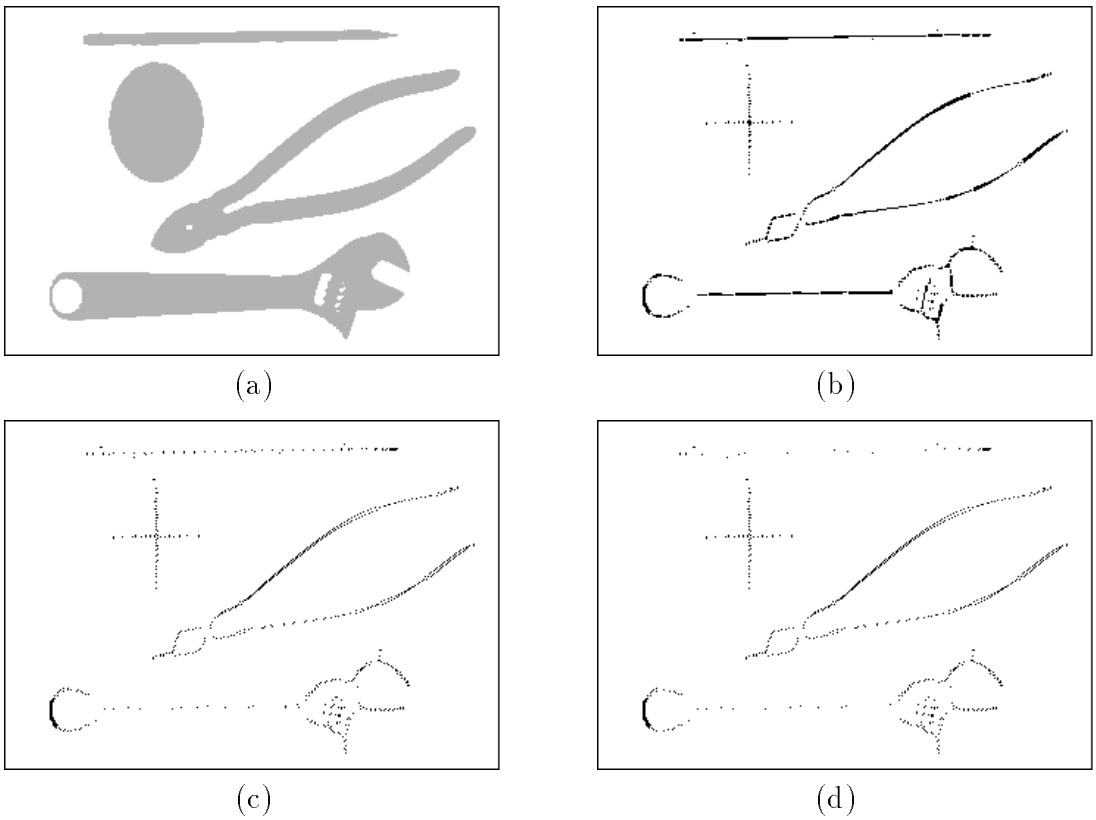


Figure 6.3: (a) The binary picture "Tools", (b) its Skeleton computed with a 3×3 square, (c) a Minimal Skeleton subset of (b), (d) the Essential Points of (b).

Global and Local Essential Points

The Essential Points, as defined above, is the set of points necessary (and not sufficient) for *perfect* reconstruction of the image X, by means of the reconstruction formula (3.10) with $J = I$. Here we define *Local Essential Points* as the points necessary (and not sufficient) for the *partial* reconstruction $\gamma_i(X)$, obtained by removing all Skeleton points with radius which are not greater or equal to i, $\forall i \in I$:

Definition 3 *A point $t \in X$ is a* **Local** *Essential Point of order i of the Skeleton of X, iff for the anti-umbra*[2] *$J = \{j \in I \mid j > i\}$:*

$$\left(\bigcup_{j \in J} \delta_j(S_j) \right) \cup (\delta_i[S_i - \{t\}]) \neq \gamma_i(X) \tag{6.10}$$

In order to avoid confusion, we denote, from this point on, by *Global Essential Points* the Essential Points previously defined. The set of Local Essential Points *contains* (or is equal to) the set of Global Essential Points.

[2]See definition in Theorem 2.

In the particular case of the original Morphological Skeleton, for which, $I = \mathcal{N}$, $i = n$, and $\delta_i(X) = X \oplus nB$, for some structuring element B, the definitions of Local and Global Essential Points are written in the following way:

Corollary 1 *A point $t \in X$ is a* **Local** *Essential Point of order n of the Skeleton of X, based on the structuring-element B, iff*

$$\left(\bigcup_{m>n} S_m \oplus mB \right) \cup ([S_n - \{t\}] \oplus nB) \neq X \circ nB \tag{6.11}$$

Corollary 2 *A point $t \in X$ is a* **Global** *Essential Point of order n of the Skeleton of X, based on the structuring-element B, iff*

$$\left(\bigcup_{m \neq n} S_m \oplus mB \right) \cup ([S_n - \{t\}] \oplus nB) \neq X \tag{6.12}$$

where S_m is the Skeleton subset of order m.

In section 6.4.2 below, efficient formulæ for calculating the Global Essential Points of a Generalized Skeleton are presented. In section 7.4 below, efficient formulæ for calculating the Local and the Global Essential Points for the original Morphological Skeleton are presented.

6.4 Redundancy Reduction

The approach presented in this section is developed for the Generalized Skeleton Representation presented in chapter 3, and therefore for any of its particular cases.

6.4.1 Reduced Skeletons

In [29] and [48], the approach used to remove redundant points from the Skeleton was *first* to calculate the Skeleton and *then* to apply a reduction algorithm to remove the redundant points.

However, we note that the skeletonization itself is a *partial reduction process*, as we demonstrate below. If the Skeleton subsets S_i would have been defined as $S_i = \varepsilon_i(X)$, $\forall n$, (where $\varepsilon_i(X)$ is the adjoint erosion of $\delta_i(X)$) then the exact reconstruction property (6.1), for $T_i = S_i$, would still be satisfied, but this "Skeleton" would contain too many points. In fact, if $\delta_0(X) \stackrel{\triangle}{=} X$, then S_0 itself would then be equal to X. Instead, the sets $\bigcup_{j>i} \gamma_{[i,j]}\varepsilon_i(X)$ of redundant points are removed from $\varepsilon_i(X)$ for all i in the definition of the Generalized Skeleton (3.9), (page 46), so that a compact representation is obtained. However, as mentioned before, only the "Single-Element Redundancy" is removed this way.

We propose to remove as many redundant points as possible *during* the skeletonization process, which is fully morphological, so that a more efficient error-free decomposition than the ordinary Skeleton is obtained by morphological operations only.

The proposed approach is based on the following relation:

$$RS_i = \begin{pmatrix}\text{representation}\\ \text{points of order}\\ i\end{pmatrix} - \begin{pmatrix}\text{redundant}\\ \text{points of}\\ \text{order } i\end{pmatrix} \tag{6.13}$$

where $\{RS_i\}$ are the *Reduced Skeleton subsets.*

The representation points are the centers of elements with index i, therefore the above relation can be written as follows:

$$RS_i = \varepsilon_i \begin{pmatrix}\text{representation}\\ \text{region of order}\\ i\end{pmatrix} - \varepsilon_i \begin{pmatrix}\text{redundant}\\ \text{region of}\\ \text{order } i\end{pmatrix} \tag{6.14}$$

The "representation region of order i" is actually the opening $\gamma_i(X)$. By replacing the field "redundant region of order i" in (6.14) by appropriate sets, one can obtain different Reduced Skeletons.

Higher-Level Redundancy-Free Skeletons

A Skeleton with *no "Higher-level Redundancy"* is obtained if we choose "redundant region" to be the *union* of all the *maximal elements* with index **greater** than i, which we denote F_i.

A simple formula for obtaining F_i *before* the calculation of the elements with index greater than i is obtained by setting $J = \{j \in I \mid\ j > i\}$ in (3.10), (page 47):

$$F_i \stackrel{\triangle}{=} \bigcup_{j>i} \delta_j(S_j) = \gamma_J(X) = \bigcup_{j>i} \gamma_j(X). \tag{6.15}$$

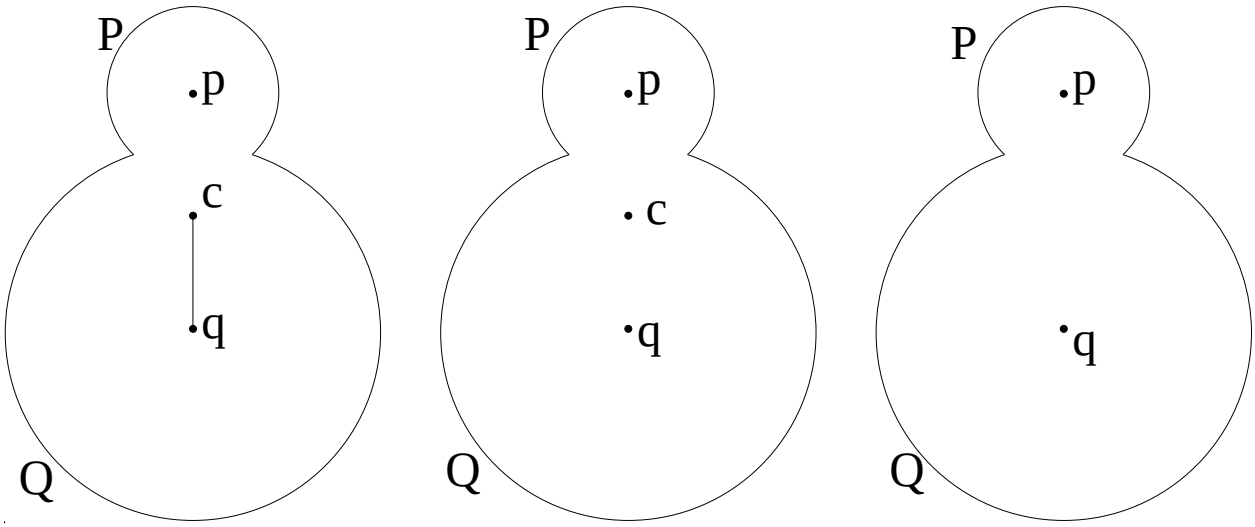
The subsets $RS_i^{(1)}$ of the resulting Reduced Skeleton with no "Higher-level" redundancy are therefore given by:

$$\begin{aligned} RS_i^{(1)} &\stackrel{\triangle}{=} \varepsilon_i[\gamma_i(X)] - \varepsilon_i\left[\bigcup_{j>i} \gamma_j(X)\right] \\ &= \varepsilon_i(X) - \varepsilon_i\left[\bigcup_{j>i} \delta_j\varepsilon_j(X)\right] \\ &= \varepsilon_i(X) - \varepsilon_i\left[\bigcup_{j>i} \delta_i\delta_{[i,j]}\varepsilon_{[i,j]}\varepsilon_i(X)\right] \\ &= \varepsilon_i(X) - \phi_i\left[\bigcup_{j>i} \gamma_{[i,j]}\varepsilon_i(X)\right] \end{aligned} \tag{6.16}$$

where $\phi_i(\cdot)$ is the closing related to the index i, and $\delta_{[i,j]}(\cdot)$, $\varepsilon_{[i,j]}(\cdot)$, and $\gamma_{[i,j]}(\cdot)$ are respectively the dilation, erosion and opening associated with the structuring function $\delta_{[i,j]}(x) \stackrel{\triangle}{=} \varepsilon_i\delta_j(x)$.

From (6.16), one concludes that $RS^{(1)}$ is the generalization of the Reduced Skeleton proposed by Maragos in [30] (see also [50]):

$$RS_n = X \ominus nB - [(X \ominus nB) \circ B] \bullet nB. \tag{6.17}$$



(a) (b) (c)

Figure 6.4: (a) The same binary image as shown in Fig.2, and its Reduced Skeleton $RS^{(1)}$, (b) its Reduced Skeleton $RS^{(2)}$, (c) its only Minimal Skeleton.

In [30], $I = \mathcal{N}$, the index n is used instead of i, and $\delta_n(X) \triangleq X \oplus nB$, where nB is the n-fold dilation of a structuring element B by itself (as in the Morphological Skeleton, see section 3.1.2). Notice also that (6.16) is identical to the calculation formula for the Generalized Skeleton (3.9) with the addition of the closing $\phi_i(\cdot)$.

Fig. 6.4(a) shows the result of the calculation of $RS^{(1)}$ associated with the Skeleton of the binary image shown in Fig. 6.2. It contains the point p and the segment $[c, q]$, where c is the same "Intralevel" redundant point shown in Fig. 6.2(c). The points from the segment (p, c), which are "Future-level" redundant in the Skeleton, are not present in $RS^{(1)}$.

Higher/Lower-Level Redundancy-Free Skeletons

If we include in "redundant region" of equation (6.14) the union of all the representation elements with order **smaller** than i as well, we obtain an error-free Reduced Skeleton, which we denote as $RS^{(2)}$, with *no Higher/Lower-level redundancy.*

The union of the representation elements with order smaller than i, which we denote as P_i, is defined by:

$$P_i \triangleq \bigcup_{j<i} \delta_j(RS_j^{(2)}). \tag{6.18}$$

Notice that P_i depend only on the sets RS_j with indices strictly smaller than i.

The subsets of $RS^{(2)}$ are given by:

$$RS_i^{(2)} \triangleq \varepsilon_i(X) - \varepsilon_i(P_i \cup F_i) \tag{6.19}$$

Fig. 6.4(b) shows the result of the calculation of $RS^{(2)}$ for the same binary image as before. It contains only the points p, q and c. The points from the segment (c, q), which

are "Interlevel" redundant in the Skeleton, are not present in $RS^{(2)}$. The point c, which is "Intralevel" redundant, is still present.

If $I = \mathcal{N}$ (totally ordered and discrete), than P_i can be computed recursively in the following way:

$$\begin{cases} P_i = P_{i-1} \cup \delta_{i-1}(RS_{i-1}^{(2)}), & i \geq 1 \\ P_0 = \emptyset. \end{cases} \tag{6.20}$$

This permits an efficient calculation of P_i and, thus, of $RS^{(2)}$.

Interlevel Reduction for Multi-Parameter Skeletons

When I is totally ordered, the interlevel redundancy and the Higher/Lower-level redundancy are identical, as stressed above, and therefore calculating a Higher/Lower-level Redundancy-Free Skeleton provides also an Interlevel Redundancy-Free Skeleton.

In the case of Multi-Parameter Skeletons (like the MSES), an Interlevel Redundancy-Free Skeleton can be obtained in the following way. Let us disregard the strong partial ordering of $\mathcal{N}^d$, used for the Skeleton calculation, and instead let us consider its *lexicographic order*:

$$\begin{array}{c} (n_1, \ldots, n_d) < (m_1, \ldots, m_d) \\ \Updownarrow \\ \left(\begin{array}{c} n_1 < m_1 \text{ or} \\ (n_1 = m_1 \text{ and } n_2 < m_2) \text{ or} \\ \vdots \\ (n_1 = m_1 \text{ and } \cdots \text{ and } n_{d-1} = m_{d-1} \text{ and } n_d < m_d) \end{array} \right) \end{array} \tag{6.21}$$

Notice that this is a total ordering.

In this case, the union of all the maximal elements with index *greater* than i, F_i, is given by:

$$F_i = \bigcup_{\lambda > 0} \left[\gamma_{(n_1+\lambda,\ldots,0,0)}(X) \cup \gamma_{(n_1,\ldots,n_{d-1}+\lambda,0)}(X) \cup \cdots \cup \gamma_{(n_1,\ldots,n_{d-1},n_d+\lambda)}(X) \right] \tag{6.22}$$

The union of the elements with index *smaller* than i, P_i, is still obtained by (6.18), but using the lexicographic order this time.

With the above redefinitions of F_i and P_i, an Interlevel Redundancy-Free Skeleton for the Multi-Parameter Skeletons is obtained by the same relation (6.19) as above. Notice that the recursive calculation of P_i, (6.20), is valid for the Multi-Parameter Skeleton, if we interpret $i-1$ as the *previous value of* i in the lexicographic order.

Intralevel Reduction

To obtain a Minimal Skeleton, the intralevel redundancy should also be removed. Unfortunately, it seems not possible to define a "redundant region" that would remove this kind of redundancy without affecting the property of exact reconstruction of the Reduced Skeletons. In the example of Fig. 6.4, the Minimal Skeleton (which is unique in this example) is shown in Fig. 6.4(c).

Example

Fig. 6.5 shows the result of calculating the Reduced Skeletons defined above, for a "real" binary picture, the 256×256 "Coffee-Grains". Its Generalized-Step Skeleton (see section 3.1.4), Fig. 6.5(a), was calculated using the shapes shown in Fig. 6.6 as the first 6 elements of $\{B(n)\}$, so that $\{A(n)\}$ is *approximately* a family of increasing *disks*, and the Skeleton is as thin as possible. The other elements of $\{B(n)\}$ were derived from them cyclically by the formula: $B(n) = B(n \bmod 6)$. This Skeleton contains 1360 representation points. Fig. 6.5(b) shows the Reduced Skeleton $RS^{(1)}$, containing 1145 points (15.8% less than the original Skeleton), and Fig. 6.5(c) shows the Reduced Skeleton $RS^{(2)}$, containing 972 points (28.5% less than the original Skeleton). For comparison, Fig. 6.5(d) shows a Minimal Skeleton, obtained with the non-morphological algorithm presented in [29]. It contains 877 points (35.5% less than the original Skeleton). According to the above numbers, the Reduced Skeleton $RS^{(2)}$ was able to remove in this example 80% of the redundant points, using morphological operations only.

Minimal MSES

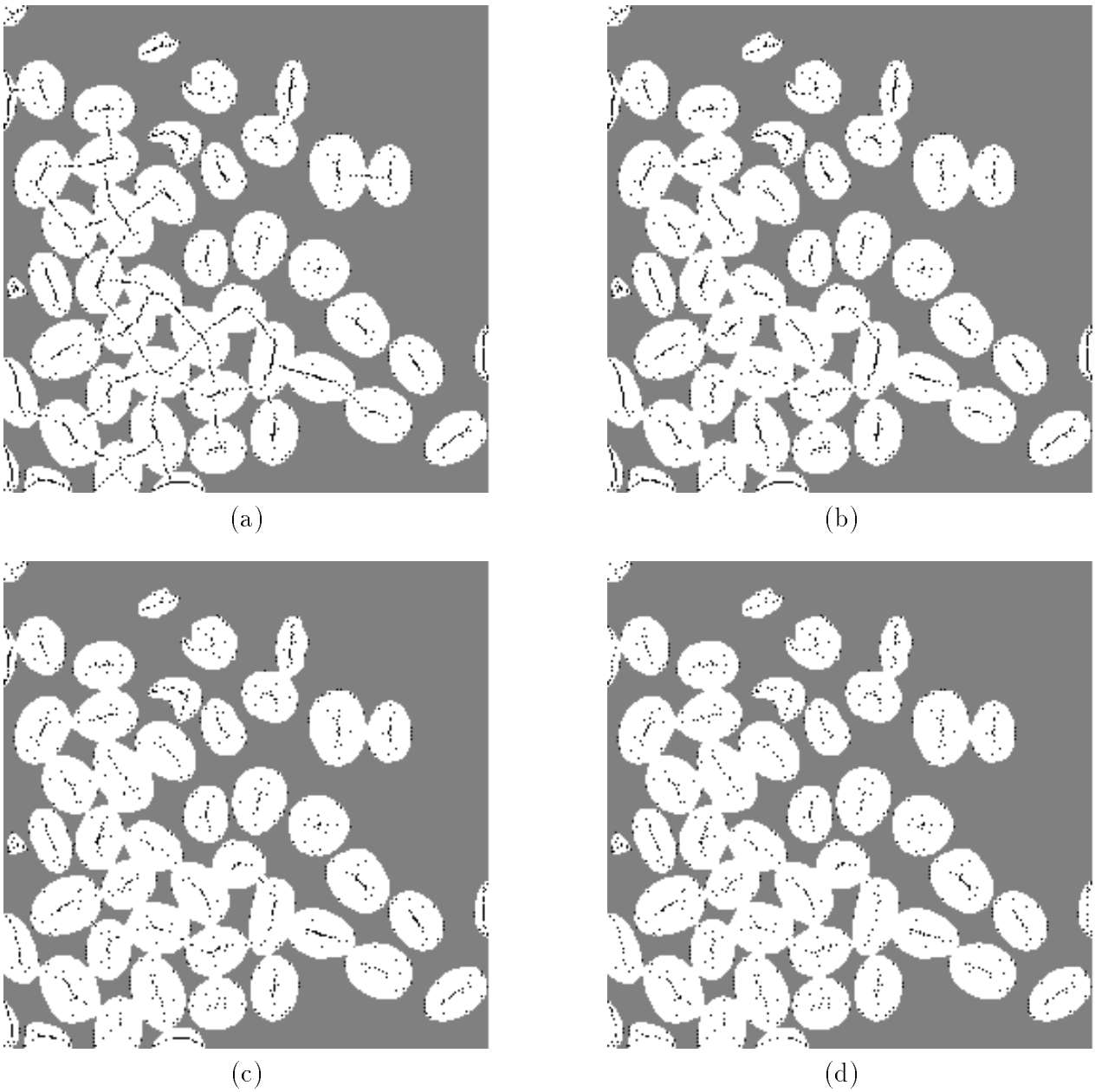
For a particular but important choice of the families of shapes for the MSES with 2 structuring-elements, $RS^{(2)}$ yields a representation with *no redundant points*, i.e., a Minimal MSES. The families of shapes which provide this result are in the form $A(n_1, n_2) = n_1 B_1 \oplus n_2 B_2$, where B_1 and B_2 are elements containing exactly 2 points, which we call *discrete elementary directional structuring-elements*. The shapes in Fig. 6.6 are examples of discrete elementary directional structuring-elements.

It is well known [29] that the ordinary Skeleton, computed with any directional structuring-element, contains no redundancy. As an extension to this property, the Reduced MSES $RS^{(2)}$, computed with any pair of structuring-elements from Fig. 6.6 (or any other pair of elementary directional elements), contains no *intralevel* redundancy.

Since $RS^{(2)}$ has no *interlevel* redundancy, the conclusion is that it contains no redundant points at all. It is therefore a Minimal MSES.

In contrast to 1-parameter families of directional shapes, in which there is little interest as kernels, the families of shapes generated by pairs of elementary directional structuring-elements are important ones. E.g., in the case of the *horizontal* and the *vertical* elementary structuring-elements (the first two elements shown in Fig. 6.6), the family $A(n, m)$ obtained is composed of all discrete rectangles (see Fig. 3.4(b)).

We compared the Minimal MSES representation, obtained by the proposed approach, of the binary image "Coffee-Grains" (Fig. 6.5) calculated with the horizontal and vertical elementary structuring-elements mentioned above, with a Minimal Skeleton representation of the same image calculated with a 3×3 square as structuring-element. Note that the 2-parameter family of rectangles, used for the MSES calculation, *contains* the 1-parameter family of squares, used for the Skeleton calculation. In this case, the number of points in the Minimal MSES is expected to be much smaller than the number of points in the Minimal Skeleton. Indeed, in the above simulation the Minimal MSES obtained contains 708 points, whereas the Minimal Skeleton contains 990 points (a difference of 28.5%). The Minimal Skeleton calculation was performed using the non-morphological algorithm presented in [29].



(a) (b) (c) (d)

Figure 6.5: Reduced Skeletons. (a) Binary image and its Skeleton, (b) its Reduced Skeleton $RS^{(1)}$, (c) its Reduced Skeleton $RS^{(2)}$, and (d) a Minimal Skeleton.

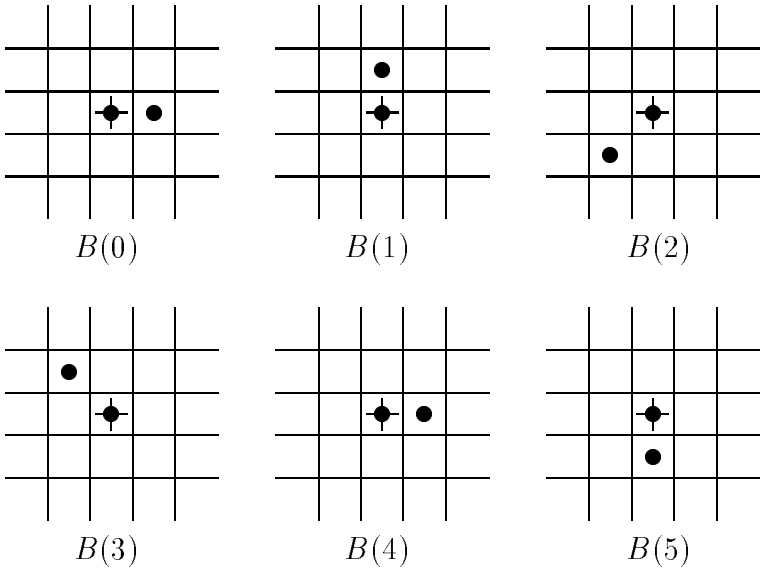


Figure 6.6: Series of shapes $\{B(n)\}$ used in the calculations of the Skeletons in Fig.3 (The symbol "+" represents the origin).

6.4.2 Extraction of Essential Points

The same approach that yields the Reduced Skeletons of the last section, also permits us to obtain the Global Essential Points of the Skeleton using morphological operations only. The calculation is performed at each step of the skeletonization process, so that the Essential Points of order i are obtained *before* the calculation of the Skeleton subsets of orders *greater* than i. Since we do not consider Local Essential Points here, throughout this section the expression "Essential Points" denotes the Global Essential Points.

To extract the Essential Points, equation (6.13) is written in the following way:

$$EP_i = S_i - \begin{pmatrix}\text{Non-Essential}\\ \text{Points of order}\\ i\end{pmatrix} \tag{6.23}$$

where EP_i is the set of Essential Points of order i, and "Non-Essential Points of order i" are those Skeleton points of order i which are not Essential Points.

Equation (6.23) may also be written as follows:

$$EP_i = S_i - \varepsilon_i(\tilde{P}_i \cup F_i \cup R_i) \tag{6.24}$$

The union $\tilde{P}_i \cup F_i \cup R_i$ ($\tilde{P}_i$ and R_i are defined below) refers to the region represented by all the Non-Essential Points of order i, i.e., a subregion from the representation region of order i, which is *represented more than once.*

As before, $\tilde{P}_i$ and F_i are related, respectively, to elements with indices "smaller" and "bigger" than i. F_i is the same as computed in (6.15) or (6.22), respectively to the cases where I is totally ordered or $I = \mathcal{N}$. $\tilde{P}_i$ is computed recursively as was done for P_i in (6.20), "accumulating" the regions covered by Skeleton points of order smaller than i:

$$\begin{cases} \tilde{P}_i = \tilde{P}_{i-1} \cup \delta_{i-1}(S_{i-1}), & i \geq 1 \\ \tilde{P}_0 = \emptyset. \end{cases} \tag{6.25}$$

The sets $\{R_i\}$ are those regions which are covered *more than once by elements of size i only*. An exact expression for computing R_i is:

$$R_i = \bigcap_{s \in S_i} [\delta_i(S_i - \{s\})] \tag{6.26}$$

The proof is presented in Appendix A.

Formula (6.26) is an efficient way to calculate R_i only for large values of i, because in that case S_i contains only few points. For small values of i, though, there are many points in the corresponding Skeleton subsets, in which case this formula looses its efficiency.

When $\delta_i(\cdot)$ is translation invariant, i.e., $\delta_i(Y) = Y \oplus B(i), \forall Y \in \mathcal{P}(E)$, for some structuring element $B(i)$, then (6.26) is equivalent to (see Appendix A for proof):

$$R_i = \bigcap_{b \in B(i)} [S_i \oplus (B(i) - \{b\})] \tag{6.27}$$

For small values of i, equation (6.27) is preferable to (6.26) because $B(i)$ in this case usually contains a small number of points.

Once R_i, F_i and $\tilde{P}_i$ are found, the Essential Points of order n can be obtained by (6.24). Since the above sets can be obtained with morphological operations only, as shown in (6.15) or (6.22), (6.18), and (6.26) or (6.27), and since (6.24) is also morphological, the conclusion is that the extraction of Essential Points can be implemented by a morphological machine.

Chapter 7

Morphological Reduction of Skeleton Redundancy Based on The B-Convexity Theory

In this chapter we propose a redundancy reduction approach, based on a generalization of the concept of Convexity.

Many properties and relations concerning Convex Sets have been extensively studied and analyzed, and a number of generalizations of Convexity were proposed before (see [60] for example), in order to extend some of these properties and relations to sets which are not strictly convex.

In Appendix B, we present a generalization of the concept of Convex Sets, developed in this work, based on the Morphological Closing operation, and study some of its properties. We also define Extreme Points of such Generalized Convex Sets, which generalize the notion of Extreme Points of Convex Sets. In this section we summarize the main results.

We then apply the above notions to Skeleton redundancy removal, and present an algorithm for obtaining an Error-Free Skeleton representation with reduced amount of *Intra-level* Redundancy, using morphological operations only, as well as morphological formulæ for calculating Local and Global Essential Points.

Finally, it is given a qualitative comparison between the redundancy approach proposed in Chapter 6 and the one proposed in the chapter.

All the results in the remainder of this chapter relate to the original Morphological Skeleton Representation, i.e., a Skeleton for which the decomposition family is in the form $\{nB\}$, $n \in \mathcal{N}$, where B is a structuring element. See section 3.1.2.

7.1 Convex Sets and Proposed Generalization

The material in this section (7.1) and in 7.2 is a summary of the B-Convexity theory, developed and presented in Appendix B, which the reader should address for further details, and proofs of the theorems.

There are several acceptable definitions for Convex-Hull and Convex Sets. They are all equivalent, up to topological differences concerning the points on the boundary of the shapes.

We can also define first the Convex-Hull and then use this definition for defining Convex Sets, or we can do the opposite.

The definitions of the Convex-Hull and Convex sets we choose to work with are the following:

- **Convex-Hull:** $CH(X)$ is the Convex-Hull of a set X iff it is the intersection of all the half-planes that contain X.
- **Convex set:** A set X is Convex iff it is identical to its Convex-Hull, i.e., $X = CH(X)$.

The generalization we propose is obtained by replacing the *half-plane* used in the above definition of the Convex-Hull by a generic set $(B^s)^c$, which is the complement of the symmetric of any structuring-element B. We denote the generalized Convex sets as B-Convex sets and the generalized Convex-Hull as B-Convex-Hull because of the dependence on the structuring-element B:

- **B-Convex-Hull:** $CH^B(X)$ is the B-Convex-Hull of X iff it is the intersection of all the translations of $(B^s)^c$ that contain X.
- **B-Convex set:** A set X is B-Convex iff it is identical to its B-Convex-Hull ,i.e., $X = CH^B(X)$.

Actually, the B-Convex-Hull, as defined above, is not a new operation; it is known in Mathematical Morphology as the Morphological **Closing**. In other words:

$$CH^B(X) = X \bullet B. \tag{7.1}$$

If we choose B to be a disc, and make its radius go to infinity, then the above Closing converges to the conventional Convex-Hull (as pointed out in [51, p. 100]), meaning that the conventional Convex-Hull is indeed a particular case of the generalized Convex-Hull.

Table 7.1 shows that some of the basic properties of the Convex-Hull and of Convex sets are naturally extended to the B-Convex-Hull operation and to B-Convex sets.

7.2 Extreme and Internal Points

7.2.1 Definition and Calculation

Like the Convex-Hull and Convex Sets, there are many ways to define *Extreme Points* of a Convex Set. Table 7.2 shows one of the classical definitions of Extreme Points for conventional Convex sets, and presents its generalization for B-Convex sets. We denote the set of Extreme Points of a given Convex Set Y by $\mathcal{E}(Y)$ and the set of Extreme Points of a given B-Convex Set X by $\mathcal{E}^B(X)$.

The following *Morphological closed-form formulæ* provide two ways of calculating the set of Extreme Points of a given B-Convex Set X:

$$\mathcal{E}^B(X) = X - \left[\bigcap_{x \in X} (X - \{x\}) \oplus B \right] \ominus B^s \tag{7.2}$$

Known Property (in traditional Convexity)	Property of the Proposed Generalization (B-Convexity)
$CH(\cdot)$ is idempotent.	$CH^B(\cdot)$ is idempotent.
$CH(X)$ is the "smallest" convex set that contains X.	$CH^B(X)$ is the "smallest" B-Convex set that contains X.
X is convex iff any two points x and y belonging to X are connected by a segment contained in X. in other words: X is convex iff $\forall\{x,y\} \subseteq X$, $CH(\{x,y\}) \subseteq X$.	If X is B-Convex, then $\forall\{x,y\} \subseteq X$, $CH^B(\{x,y\}) \subseteq X$.
The intersection of convex sets is a convex set.	The intersection of B-Convex sets is a B-Convex set.
X is convex iff every point outside X can be separated from X by a half-plane, i.e., $x \notin X \Rightarrow \exists$ a half-plane that contains x and does not intersect X.	X is B-Convex iff every point outside X can be separated from X by a translation of B^s, i.e., $x \notin X \Rightarrow \exists z \in \mathcal{R}^2$ such that $(B^s)_z$ contains x and does not intersect X.

Table 7.1: Properties of Convex-Hull and Convex sets.

Extreme Points	
Convex sets	B-Convex sets
A point t is an Extreme Point of a Convex set X iff the set $(X - \{t\})$ is also convex.	A point t is an Extreme Point of a B-Convex set X iff the set $(X - \{t\})$ is also B-Convex.

Table 7.2: Extreme Points

$$\mathcal{E}^B(X) = X - \left[\bigcap_{b\in B} X \oplus (B - \{b\})\right] \ominus B^s \tag{7.3}$$

If we consider the computational efficiency of the above equations, when implemented on a computer, then (7.2) is preferable over (7.3) if X contains fewer elements than B, and (7.3) is preferable over (7.2) otherwise.

We also define *Internal Points* of B-Convex sets. They are all the points in the set that are not Extreme. Denoting the set of Internal Points of a B-Convex set X, by $\mathcal{I}^B(X)$, we may write $\mathcal{I}^B(X) = X - \mathcal{E}^B(X)$. And, since $\mathcal{E}^B(X) \subseteq X$, it is also true that

$$\mathcal{E}^B(X) = X - \mathcal{I}^B(X). \tag{7.4}$$

Note, therefore that:

$$\mathcal{I}^B(X) = \left[\bigcap_{x\in X} (X - \{x\}) \oplus B\right] \ominus B^s \tag{7.5}$$

$$= \left[\bigcap_{b\in B} X \oplus (B - \{b\})\right] \ominus B^s \tag{7.6}$$

7.2.2 Reconstruction from Extreme Points

If a conventional Convex set Y is bounded, then it can be reconstructed back from its Extreme Points by performing the Convex-Hull operation, i.e., $CH(\mathcal{E}(Y)) = Y$. The set of Extreme Points can be seen as a compact representation of a Convex set.

For a B-Convex Set X, a *necessary* condition for perfect reconstruction from its set of Extreme Points $\mathcal{E}^B(X)$ is: $[X - \mathcal{E}^B(X)] \ominus B = \emptyset$. This suggests that X should be "smaller" (in a certain way) than B. Notice that the erosion of any bounded shape by a half-plane is always empty.

The above considerations motivate the definition of a *Reconstruction Window* for a given structuring-element B, inside which every B-Convex Set can be reconstructed from its Extreme Points. A B-Convex Set W is called a Reconstruction Window for B iff $\forall X$ B-Convex, $CH^B[\mathcal{E}^B(X \cap W)] = X \cap W$.

For example, if B is a rectangle, then B itself is a Reconstruction Window for B. If B is a *discrete* rectangle of integer sides n and m, then any discrete rectangle of sides i and j, such that $0 \leq i \leq (n+1)$ and $0 \leq j \leq (m+1)$, is a Reconstruction Window for B.

7.3 Reduced Skeleton

In the sequel we present an algorithm for *morphologically* obtaining a redundancy-reduced skeleton, based on the B-Convexity theory discussed above. The algorithm is presented below, together with an example. Figure 7.1 shows the steps of the algorithm for the example.

1. Let X be a given binary image. Choose a structuring-element B, and a family of Reconstruction Windows $\{W^{(nB)}\}$ for all the dilations nB of B. (In the example, X is the digital binary shape shown in Fig. 7.1(a) (described by the black dots), B is a 3×3 square, and $W^{(nB)}$ are $(2n+2) \times (2n+2)$ squares). Set $n = 0$.

2. Calculate the skeleton subset $S_n \triangleq X \ominus nB - (X \ominus nB) \circ B$, and the set $Z_n \triangleq X \ominus nB$. If Z_n is empty then stop. (In the example, for $n = 1$, S_1 is shown in Fig. 7.1(b) and Z_1 is seen in Fig. 7.1(c)).

3. Obtain a partition of Z_n into blocks Y_p^n such that: Y_p^n is the contents of Z_n inside the Reconstruction Window $W^{(nB)}$ centered at p, i.e., $Y_p^n = [W^{(nB)}]_p \cap Z_n$, and the blocks cover the whole set Z_n, i.e., $\bigcup_p Y_p^n = Z_n$. (In the example, the blocks Y_p^n were obtained by translating the Reconstruction Window horizontally and vertically by steps of $p = 2n + 1$ pixels, so that there is a 1-pixel-wise overlap between the blocks. The overlap by one pixel contributes to the redundancy reduction. Fig. 7.1(c) shows the first block Y_p^1 in grey, and the thin solid lines indicate the position of the other blocks.)

4. Calculate the Extreme Points of every block Y_p^n, according to nB, $\mathcal{E}^{nB}(Y_p^n)$. Note that Y_p^n is a (nB)-Convex set, since it is the intersection of two (nB)-Convex sets. (Fig. 7.1(d) shows the result of this operation in the example).

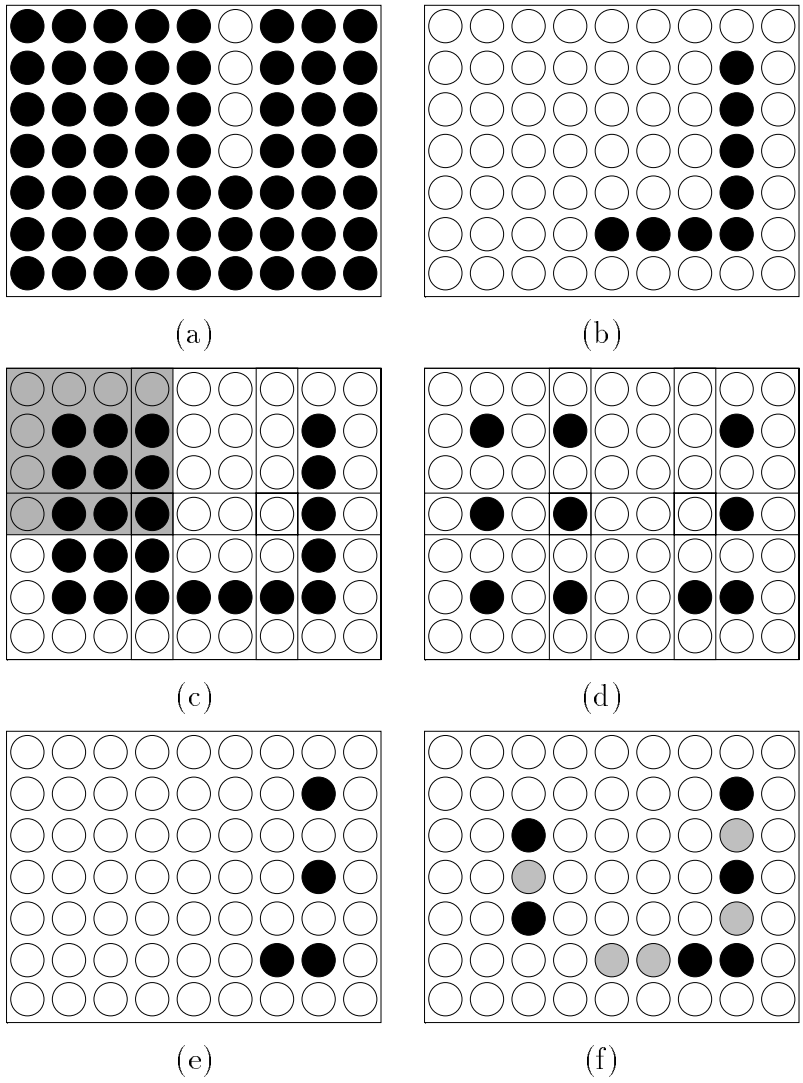
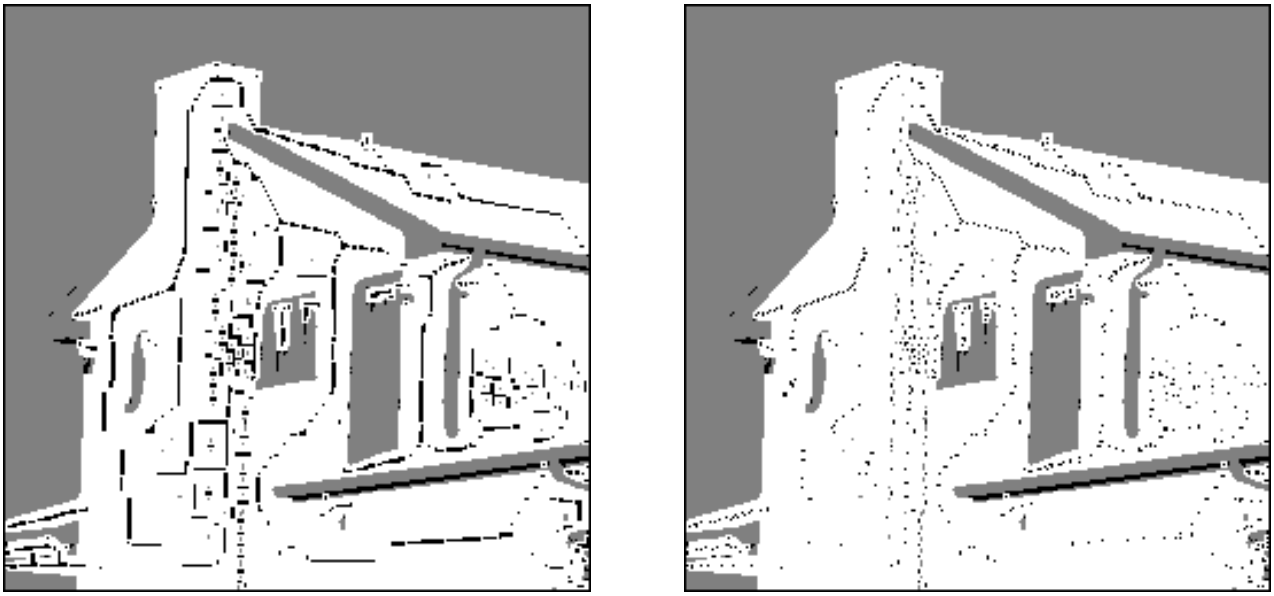


Figure 7.1: Proposed algorithm. (a) A discrete binary shape (black dots: foreground, white dots: background), (b) S_1, (c) Z_1, and the partition blocks, (d) Extreme Points of the blocks, (e) $\tilde{S}_1$, and (f) resulting reduced skeleton (black points) compared to the original skeleton (black and grey points).



(a) (b)

Figure 7.2: (a) A binary image and its skeleton, using a 3×3 squared structuring-element, (b) a reduced skeleton obtained by the proposed algorithm.

5. Define $C_n \stackrel{\triangle}{=} \bigcup_p \mathcal{E}^{nB}(Y_p^n)$ to be the set of the resulting Extreme Points of all the blocks, and intersect it with the skeleton subset S_n, obtaining $\tilde{S}_n = C_n \cap S_n$. This equivalent to:

$$\tilde{S}_n = C_n - (X \ominus nB) \circ B \tag{7.7}$$

(Fig. 7.1(e) shows $\tilde{S}_1$).

6. Increment n, and go to 2.

The collection of sets $\{\tilde{S}_n\}$ is the Redundancy-Reduced Skeleton. For a comparison between $\{\tilde{S}_n\}$ and the original skeleton $\{S_n\}$, in the scope of the above example, Fig. 7.1(f) shows the reduced skeleton composed of black dots, and the original skeleton, composed by both the black and grey dots. The grey dots are redundant points removed by the above algorithm.

Exactly as for the conventional Skeleton, the following relation holds:

$$\bigcup_{n \geq k} \tilde{S}_n \oplus nB = X \circ kB \tag{7.8}$$

which guarantees partial ($k > 0$) and perfect ($k = 0$) reconstruction of the original image.

7.3.1 Simulation

Figure 7.2(a) shows a binary image (Most-significant bit-plane of 256×256-pixel "House"), and its morphological skeleton, calculated with a 3×3 squared structuring-element. The skeleton contains 3173 points.

Fig. 7.2(b) shows the result of applying the above algorithm to the same binary image. The structuring-element and the Reconstruction Windows are the same as in the example of Fig. 7.1. The resulting skeleton fully represents the original binary image, and contains 1533 points, i.e., only 48% of the points in the original skeleton.

For comparison, a Minimal Skeleton of the above image, using the non-morphological algorithm given in [29], was calculated. It contains 1362 points, i.e., 43% of the points in the original skeleton, and 89% of the number of points in the proposed reduced skeleton. According the above numbers, the proposed skeleton was able to remove 91% of the redundant points in the original skeleton.

7.3.2 Proof of the validity of the method

We prove here that (7.8) holds. We do it using induction.

Let us assume that (7.8) holds for $k+1$ and prove that, in that case, it holds also for k.

$$
\begin{gathered}
\bigcup_{n\geq k} \tilde{S}_n \oplus nB = \\
(\tilde{S}_k \oplus kB) \cup \left[\bigcup_{n\geq k+1} \tilde{S}_n \oplus nB\right] = \\
(\tilde{S}_k \oplus kB) \cup [X \circ (k+1)B] = \\
[\tilde{S}_k \cup (X \ominus kB) \circ B] \oplus kB = \\
C_n \oplus kB = \qquad (7.9) \\
[\bigcup_z \mathcal{E}^{nB}(Y_z^k)] \oplus kB = \\
\bigcup_z [\mathcal{E}^{nB}(Y_z^k) \oplus kB] = \\
\bigcup_z Y_z^k \oplus kB = (X \ominus kB) \oplus kB \\
X \circ kB \qquad (7.10)
\end{gathered}
$$

Now it remains to prove that (7.8) holds for $k = N$, the highest order of the Skeleton. Indeed, $\tilde{S}_N = C_N$, and going back to the above proof, starting at (7.9), we get the proof for $k = N$.

7.4 Extraction of Essential Points

B-Convexity can be used also for calculating the Essential Points in a Skeleton Representation. This is demonstrated in this section.

7.4.1 Essential Points of Ribbons

Let's start with a particular case: the Essential Points of simple shapes called *ribbons*. Ribbons are considered in [37, 38] and here we give an extended definition.

Definition 4 *A ribbon is a shape for which its Skeleton, calculated with a given structuring-element B, has all its points concentrated in one single subset.*

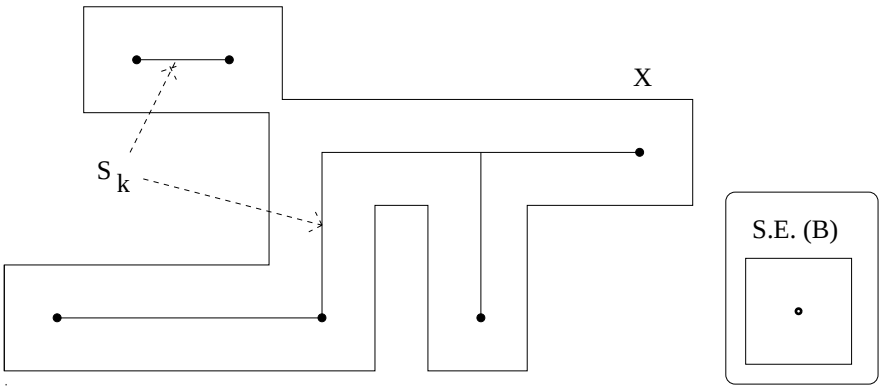


Figure 7.3: A constant-width image (generalized ribbon), based on a squared structuring-element, its Skeleton, and the Extreme Points of the Skeleton (which coincide to its Essential Points).

More precisely, if the Skeleton representation of a ribbon, using B as the structuring-element, is the collection of subsets $\{S_n\}$, then there is a number k for which $S_k \neq \emptyset$, and $S_m = \emptyset$ for every $m \neq k$.

Proposition 1 *The Global Essential Points and the Local Essential Points of the Skeleton of a ribbon are the same and they* coincide with the Extreme Points *of S_k (in the case of a ribbon, S_k is a (kB)-Convex set). I.e., the set of Essential Points of a ribbon, $E^{(\text{ribbon})}$, is given by:*

$$E^{(\text{ribbon})} = \mathcal{E}^{kB}(S_k) \tag{7.11}$$

Proof *First, let's prove that S_k is (kB)-Convex. $S_k = X \ominus kB - (X \ominus kB) \circ B$. But since every Skeleton subset of order higher than k is empty, $(X \ominus kB) \circ B = \emptyset$ and therefore $S_k = X \ominus kB$. And since $X \ominus kB = (X \ominus kB) \bullet kB$, S_k is (kB)-Convex.*

Now, let's consider the definitions of Essential Points. First, note that since, in the case of a ribbon, $\bigcup_{m \neq k} S_m \oplus mB = \bigcup_{m<k} S_m \oplus mB = \emptyset$, and $X \circ kB = X$, the definitions (6.12) and (6.11) are identical. For a ribbon shape, (6.12) is simplified, and we can say that a point t is an Essential Point of a ribbon shape iff:

$$[S_k - \{t\}] \oplus kB \neq X \tag{7.12}$$

On the other hand, t is an Extreme Point iff $[S_k - \{t\}] \oplus kB \neq S_k \oplus kB$. Since, in the case of a ribbon, $S_k \oplus kB = X$, the sufficient and necessary condition to t to be extreme is (7.12). □

Fig. 7.3 shows a ribbon and its Skeleton, using a square as the structuring-element. The enhanced points are the Essential Points of the Skeleton, which are also its Extreme Points.

Does this direct relation between Extreme Points and Essential Points hold for a generic image X? The answer is no, but we show below that there *is a strong relation* between them, even though not so direct as above.

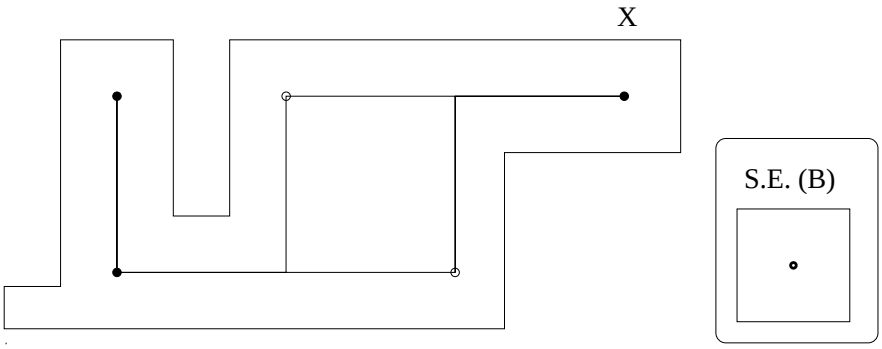


Figure 7.4: Local Essential Points of a given shape. The black points and the white circles are the Extreme Points of $X \ominus nB$, for $n = 1$. The black points are the Local Essential Points of order $n = 1$.

7.4.2 Local Essential Points of Any Image

Now, let's leave the particular case of ribbons, and consider any image. The next proposition indicates how the **Local** Essential Points of the Skeleton of any image can be obtained using B-Convexity.

Proposition 2 *The Local Essential Points of order n, $LE_n(X)$, of the Skeleton representation of a given image X is given by:*

$$LE_n(X) = \mathcal{E}^{nB}(X \ominus nB) \cap S_n \tag{7.13}$$

Proof *Note that:*

$$\begin{gathered}(\bigcup_{m>n} S_m \oplus mB) \cup ([S_n - \{t\}] \oplus nB) = \\ [(\bigcup_{m>n} S_m \oplus (m-n)B) \cup [S_n - \{t\}]] \oplus nB = \\ [(X \ominus nB) - \{t\}] \oplus nB \end{gathered} \tag{7.14}$$

Therefore, definition 1 can be rewritten in the following way: $t \in S_n$ is a Local Essential Point of order n iff

$$[(X \ominus nB) - \{t\}] \bullet nB \neq X \ominus nB \tag{7.15}$$

Which means that t is an Extreme Point of $X \ominus nB$. By calculating the set of Extreme Points of $X \ominus nB$ and keeping only those ones that belong to S_n, we obtain the Local Essential Points of the Skeleton. □

Fig. 7.4 shows a shape X, the set $X \ominus nB$ (for $n = 1$, and its Extreme Points (the black points and the white circles). The black points belong to S_n and therefore are the Local Essential Points of order n.

7.4.3 Global Essential Points of Any Image

A greater amount of computation time is required for obtaining the Global Essential Points, if compared to that required for the Local Essential Points. The next proposition gives a morphological formula for the calculation:

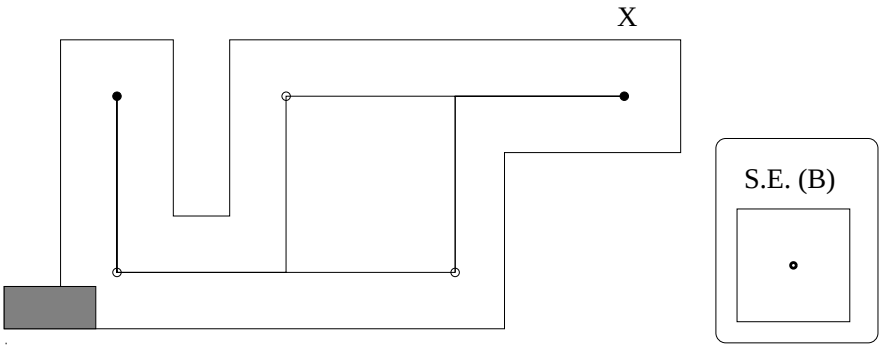


Figure 7.5: Global Essential Points of a given shape. The black points and the white circles are the Extreme Points of $X \ominus nB$, for $n = 1$. The black points are the Global Essential Points of order $n = 1$. The marked region is $\tilde{P}_n$

Proposition 3 *The set of Global Essential Points of order n, $GE_n(X)$, of the Skeleton representation of a given image X is given by:*

$$GE_n(X) = S_n - \left(\tilde{P}_n \cup [\mathcal{I}^{nB}(X \ominus nB) \oplus nB]\right) \ominus nB \tag{7.16}$$

where

$$\tilde{P}_n \stackrel{\triangle}{=} \bigcup_{m<n} S_m \oplus mB \tag{7.17}$$

and $\mathcal{I}^{nB}$ is the set of Internal Points w.r.t. the structuring element nB (see section 7.2).

The proof is shown in Appendix A. Fig. 7.5 shows a shape X, the set $X \ominus nB$ (for $n = 1$), and its Extreme Points (the black points and white circles). The black points are the Global Essential Points of order $n = 1$. Note that one of the Local Essential Points (shown in Fig. 7.4) is *not* Global Essential, because of the effect of the set $\tilde{P}_n$, represented in the figure by the marked region.

7.5 Comparison Between the Two Approaches

In this section we compare the two approaches developed above for calculating Reduced Skeletons and extracting Essential Points.

7.5.1 Reduced Skeletons

Let us consider first the Reduced Skeleton obtained by means of the B-Convexity. Most of the *Future-Level* and *Intralevel* redundant points are removed using the related algorithm. This is because the sets $(X \ominus nB)$, for each value of n, contain all the information about the present level and the future levels. By removing Internal Points of the blocks of $(X \ominus nB)$, we are actually removing most of the redundant points of these two categories. However, some of the Future-Level and Intralevel redundant points are not removed due to the artificial configuration of the window-blocks $[W^{(nB)}]_z$; some of the Extreme Points of the blocks are

not removed, but are redundant points. Moreover, the general *Interlevel* redundancy, is not removed.

We can combine the above Reduced Skeleton with the Reduced Skeletons $RS^{(1)}$ and $RS^{(2)}$ obtained by the first approach. This combination is rather simple and leads to a Reduced Skeleton with no *Interlevel* redundancy at all (characteristic of the $RS(2)$) and without most of the *Intralevel* redundancy (characteristic of the Reduced Skeleton obtained by B-Convexity). A Minimal Skeleton, however, is still not obtained.

The combination, as said above, is simple; it is obtained by substituting the step 5 of the algorithm (section 7.3) by the following:

$$\tilde{S}_n = C_n \cap RS_n^{(2)} \tag{7.18}$$

where $\{RS_n^{(2)}\}$ are the subsets of the Reduced Skeleton with no Interlevel Redundancy presented in Chapter 6. Perfect reconstruction is still assured, but equation (7.8) (partial reconstruction) no longer holds for $k > 0$ (just like it doesn't for $RS_n^{(2)}$).

7.5.2 Essential Points Extraction

Let us compare now equation (7.16) for obtaining Global Essential Points of the Skeleton, with the formula (6.24) presented in 6.4.2 for obtaining (Global) Essential Points. For the case of the one-parameter, translation-invariant, and discrete-family Morphological Skeleton we are considering here, (6.24) assumes the form:

$$GE_n = S_n - (\tilde{P}_n \cup R_n \cup F_n) \ominus nB \tag{7.19}$$

where

$$F_n \triangleq \bigcup_{m>n} S_m \oplus mB = X \circ (n+1)B \tag{7.20}$$

$$R_n = \bigcap_{t \in S_n} [(S_n - \{t\}) \oplus nB] \tag{7.21}$$

and $\tilde{P}_n$ is identical to the one defined in (7.17) above.

By comparing equations (7.16) and (7.19), we see that the only difference is that union $R_n \cup F_n$ from (7.19) was replaced by $\mathcal{I}^{nB}(X \ominus nB) \oplus nB$ in (7.16).

As stressed before, F_n is the "Redundant Region" covered by elements bigger than nB and R_n is the "Redundant Region" covered by more than one element with the same size as nB. Therefore, the conclusion is that the union of these two regions can be obtained by dilating the *Internal Points* of $X \ominus nB$ by nB.

Furthermore, by looking at equation (7.21) and at formula (7.5), we note that R_n is actually the dilation of the Internal Points of S_n by nB, i.e.:

$$R_n = \mathcal{I}^{nB}(S_n) \oplus nB \tag{7.22}$$

Therefore, another way of obtaining GE_n using the B-Convexity is:

$$GE_n = S_n - (\tilde{P}_n \cup [\mathcal{I}^{nB}(S_n) \oplus nB] \cup F_n) \ominus nB \tag{7.23}$$

Chapter 8

Skeleton Representation Coding

As discussed in section 1.3, the Skeleton decomposition of images is suitable for Compression [29]. However, the compression rates reported 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/Huffman Coding) applied directly to the original image. This made many researchers skeptical about Skeleton-based Coding.

In this chapter, we present a number of theorems concerning properties of the Skeleton Representation. These properties are not used by conventional Skeleton-coders, and this is reflected in their unsatisfactory performance. By taking these properties into account, one can either considerably improve the previous schemes, or design new efficient ones.

An example of such a scheme is also presented. Computer simulations indicate that, typically, the proposed coding scheme substantially improves the coding rates obtained by the best previous schemes for Skeleton coding, and is more efficient than coding the original binary image by Chain Code, Ziv-Lempel, Quadtree and Run-length/Huffman methods.

The theorems and the coding scheme presented in this chapter relate *only* to *Generalized-Step Skeletons* (see section 3.1.4, on page 37; column 2 of Table 3.4). Appendix C presents a partial generalization of the theorems, for the proposed Generalized Skeletons (column 3 of Table 3.5), and their proofs. An adaptation of the results and the coding scheme for grayscale functions is presented in Appendix C, section C.2.

8.1 Previous Coding Schemes

In recent years, several authors have proposed simplifications and generalizations of the Morphological Skeleton decomposition, in order to try to increase the compression ratio [50, 19, 20, 48]. Nevertheless, the improvement was generally small, and in many cases accompanied by a large increase in computation time.

On the other hand, little was proposed concerning the improvement of the *coding scheme* itself! [29, 3]

For *binary* images, There are variations of two main Skeleton coding schemes in the literature:

1. Chain Coding of the Skeleton lines [3].

2. Run-length Coding of the Skeleton Subsets [29].

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. Therefore, to take advantage of this, it is proposed in [3] to code the skeleton lines by an extended Chain Code, with symbols indicating at each point if the related radius increases, decreases or is unchanged, in addition to the direction of the next point, and with a header for each line indicating the position and the radius of its first point. 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. In [29], each Skeleton Subset has its *run-lengths* coded by a *Huffman* code or an *Elias* (ternary) code. The Skeleton Subsets S_n are coded in decreasing order of n, providing a *progressive transmission* scheme, since according to the reconstruction formula (Table 3.3), 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).

A redundancy-reduction algorithm is usually performed in order to remove most or all the redundant points in the skeleton (see chapter 6). This improves considerably the efficiency of the second scheme, but the correlation is still not taken into account. Moreover, the removal of redundant points breaks even more the continuity of digital skeleton lines, and this reduces by a great deal the performance of the first scheme. On the other hand, some authors propose to artificially connect the broken lines of the digital skeleton using dummy skeleton points, so that the first scheme is improved, but this increases the amount of redundant points in the skeleton. A trade-off between *connectivity* and *redundancy removal* is therefore created and the preferred approach is not clear.

For grayscale images, no Skeleton-based coding scheme was found by us in the literature.

8.2 Basic Definitions and Notation

In this section, we review some basic morphological concepts, which are fundamental for the understanding of the sequel.

8.2.1 Generalized-Step Skeleton

As mentioned above, the theorems in this chapter are related to a Generalized-Step Skeleton Representation of a given image X. Recall from chapter 3 that this is a *discrete-family* Skeleton, which means that the family of elements used in the Skeleton decomposition is indexed by *natural* numbers $(0, 1, \ldots)$. On the other hand, notice that X and the shapes in the above decomposition family are not restricted to be discrete. They can be discrete (sets in $\mathcal{Z}^2$), continuous (sets in $\mathcal{R}^2$) or grayscale images (functions over $\mathcal{Z}^2$ or $\mathcal{R}^2$).

For simplification, we adopt from this point on the following notation:

$$X_n \triangleq X \ominus A(n) \tag{8.1}$$

$$Y_{n+1} \triangleq X_{n+1} \oplus B(n) = [X \ominus A(n)] \circ B(n) \tag{8.2}$$

where $\{A(n)\}$ and $\{B(n)\}$ are, respectively, the decomposition family and the generator family, as defined in section 3.1.4 (page 37).

With the above notation, we can write the related Lantuéjoul's Formula (line 5 in Table 3.2) in the following way:

$$S_n = X_n - Y_{n+1} \tag{8.3}$$

Moreover, in this chapter, we denote the Euclidean space where the decomposition is defined ($\mathcal{R}^2$ in the continuous case, or $\mathcal{Z}^2$ in the discrete case) by E.

8.2.2 Descendance and Connectivity

The definitions of Descendance and Connectivity presented here are adapted from [52, pages 77 and 78].

Definition 5 (Direct Descendance) *Let B be a structuring element in $\mathcal{P}(E)$. A point $y \in E$ is a* direct descendant *of a point $x_0 \in E$, under the given structuring element, iff:*

$$y \in \{x_0\} \oplus B \tag{8.4}$$

Definition 6 (Descendance) *A point $y \in E$ is a* descendant *of a point $x_0 \in E$, under the given structuring element, iff there is a chain of points, each being a direct descendant of a previous one, starting with x_0 and ending with y.*

Definition 7 (Connectivity) *Two points x_0 and y are connected (under a pre-defined structuring element B) iff each one is a direct descendant of the other, under B, i.e.:*

$$x_0 \in \{y\} \oplus B \text{ and } y \in \{x_0\} \oplus B \tag{8.5}$$

Intuitively, B defines a *neighborhood* for which Descendance and Connectivity are considered. A point directly descends from another if the former belongs to the neighborhood of the latter. Similarly for connectivity. If, moreover, B is *symmetric* (i.e., $b \in B \Rightarrow -b \in B$), then y is a direct descendant of x_0 iff x_0 is a direct descendant of y. Therefore, in this case, Direct Descendance and Connectivity are equivalent.

8.2.3 Reconstruction Operator

Reconstruction is a very useful morphological operator, which finds several applications, such as extraction of connected components and filtering.

Definition 8 (Reconstruction) *Let A, D be two sets in a Boolean Lattice, such that $D \subseteq A$, and B be a pre-defined structuring element. The* Reconstruction *of A from D under B, $Rec\{A, D\}_B$ is given by the following recursive formula:*

$$Rec\{A, D\}_B \triangleq \{[(D \oplus B) \cap A] \oplus B\} \cap A \ldots \tag{8.6}$$

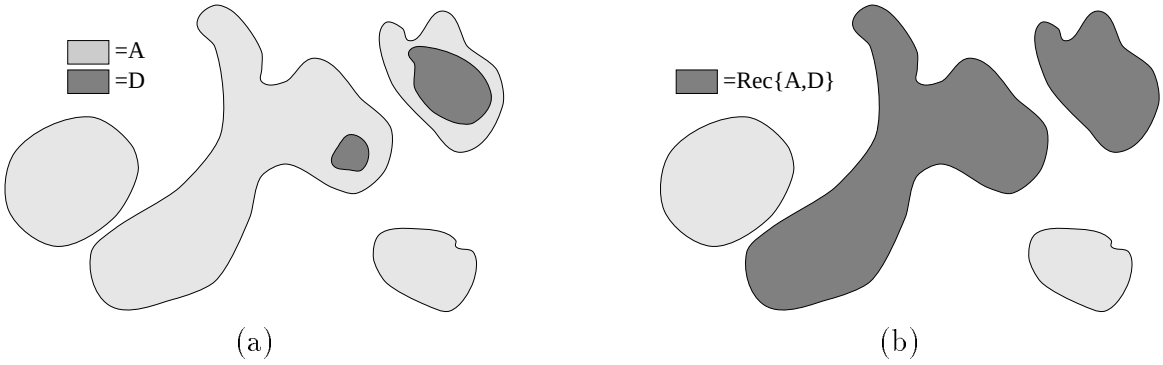


Figure 8.1: The Reconstruction operator. (a) Two sets A and D, such that $D \subset A$. (b) The result of Reconstruction of A from D, under an infinitesimal circular structuring element B.

The Reconstruction operator returns those points in A which descend from D, under B and restricted to A. When B is a *symmetric* structuring element, the Reconstruction of A from D is the collection of *connected components* of A which contain points of D (see Fig. 8.1). Notice that the notion of connected component depends on the structuring element B chosen in the Connectivity definition. If it is too "big", then relatively distant points can eventually be considered connected. Therefore, B is usually selected to be as small as possible (an 8-pixel or 4-pixel neighborhood, in the discrete case, or a infinitesimally small disc in the continuous case).

It is important not to confuse the above operation of Reconstruction with the *reconstruction of the original image X from its Skeleton Representation*, considered in the previous chapters.

8.2.4 Ultimate Erosions

In [51], the *Ultimate Erosions* are defined in terms of a decreasing family of erosions $\{X \ominus nB\}$, where n is a natural number. For each n, the Ultimate Erosions of order n, denoted U_n, of a given set $X \in \mathcal{P}(E)$, is defined by:

$$U_n \triangleq X \ominus nB - \mathrm{Rec}\left\{X \ominus nB, [X \ominus nB] \circ B\right\}_B \tag{8.7}$$

In words, the Ultimate Erosions of order n are the points of $X \ominus nB$ which do not descend, under B, from the opening of $X \ominus nB$ by B.

Intuitively, the Ultimate Erosions, at each erosion step, mark the "convex regions" of X which are going to disappear after a further erosion. Fig. 8.2 shows an example, with B being a disc. Notice that, although the original set X is composed of two connected components, the Ultimate Erosions consist of three connected components, because one of the components of X is a union of two "convex regions".

Here, we adapt the definition of Ultimate Erosions for generalized-step families as well:

Definition 9 *We define the Ultimate Erosions U_n as:*

$$U_n \triangleq X_n - Rec\left\{X_n, Y_{n+1}\right\}_{C(n)} \tag{8.8}$$

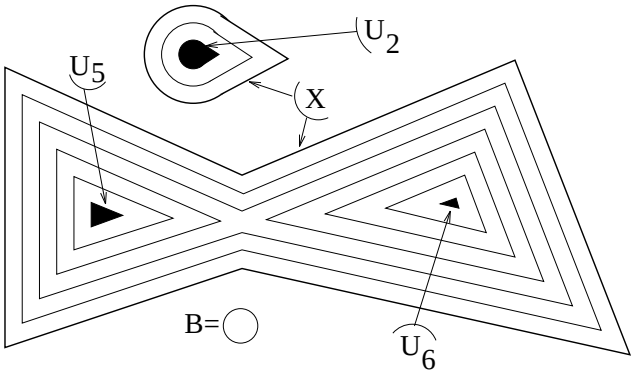


Figure 8.2: Discrete Ultimate Erosions U_n of a set X composed of two connected components.

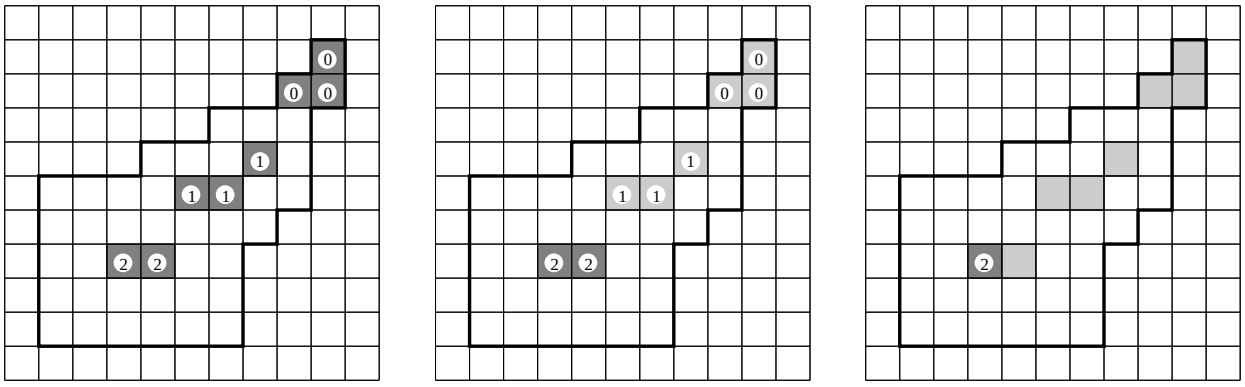


Figure 8.3: (a) A binary image (indicated by the thick line), and its Skeleton points, (b) the Ultimate Erosions (the darker points). (c) The only point in the Skeleton which needs to have its radius coded, according to Corollary 4

where X_n and Y_{n+1} are as defined in (8.1) and (8.2), respectively, and:

$$C(n) = \begin{cases} B(n-1), & n \geq 1 \\ \textit{Any structuring element } B, & n = 0 \end{cases} \tag{8.9}$$

When defined by Definition 9, *the Ultimate Erosions are contained in the Generalized-step Skeleton of X*, when it is calculated by Lantuéjoul's Formula (8.3), with the same family $\{A(n)\}$. This is because the result of the Reconstruction operation in (8.8) contains the set Y_{n+1}, which is subtracted from X_n in Lantuéjoul's Formula.

In practice, the Ultimate Erosions are those Skeleton points with maximal "radius" within each "convex region" of the original shape. They are usually a small percentage of the Skeleton. For example, consider the image in Fig. 8.3(a), and its Skeleton, calculated with a constant generator $B(n) = B$, equal to a 3×3-squared structuring element. Fig. 8.3(b) shows its Ultimate Erosions, which belong in this case to U_2 only. Fig. 8.4(a), shows another example, where the Skeleton is calculated in the same way as in the first example.

(a) (b)

Figure 8.4: (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 radius, in addition to the position of all the Skeleton points, are sufficient for perfect reconstruction.

8.3 New Skeleton Properties

Our main theoretical results concerning Coding are presented in this section. They are related *only* to **discrete-family** Generalized-step Skeleton Representations. See Appendix C for a generalization of part of the results, for the Generalized Skeleton Representation proposed in chapter 3.

The theoretical results are new Skeleton properties, divided into two categories: Quench-Function Sampling, and Deterministic Prediction.

8.3.1 Quench-Function Sampling

According to Chapter 3, the position and the "radius" of each Skeleton point is needed for perfect reconstruction of the original set X. In this section, we show that, for a *discrete-family Generalized-step Skeleton*, one can *discard* the "radius" of *most* of the Skeleton points from the representation, and still a perfect reconstruction is possible.

The following lemma helps us formulate the above assertion in the form of a theorem.

Lemma 1 *Let $\{S_n\}_{n\in\mathcal{N}}$ be the Skeleton subsets of a Generalized-step Skeleton, satisfying the conditions in Table 3.4. Let $S \triangleq \bigcup_{n\geq 0} S_n$.*

The following holds:

$$Rec\{S, Y_{n+1}\}_{C(n)} \cup U_n = X_n \tag{8.10}$$

The above result is used in the proof of the next theorem, and leads to the corollaries presented below. The proof of Lemma 1 is given in Appendix A.

The following theorem is the main result in this section.

Theorem 4 *Let $\{S_n\}_{n\in\mathcal{N}}$ be the Skeleton subsets of a Generalized-step Skeleton, satisfying the conditions in Table 3.4, for a given image X, and $S \stackrel{\triangle}{=} \bigcup_{n\geq 0} S_n$. Let Ultimate Erosions be defined as in (8.8).*

X is perfectly represented by the sets $\{U_n\}_{n\in\mathcal{N}}$ and S.

In other words, the radius of the Skeleton points which are Ultimate Erosions, together with the position of all the Skeleton points, are sufficient for perfectly representing the original set X.

Proof *We use induction in the following way:*

1. *If N is the maximal radius in the Skeleton, then $X_N = U_N$.*
2. *Once X_{n+1} is known, each set X_n, $N > n \geq 0$, can be calculated (see below), and*
3. *the original image X is equal to X_0.*

In order to obtain the second part of the above induction, suppose that X_{n+1} is available. Therefore Y_{n+1} is also available. From the hypothesis, the Skeleton S and the Ultimate Erosions $\{U_n\}$ are provided. Then X_n is obtained from the above by Lemma 1. □

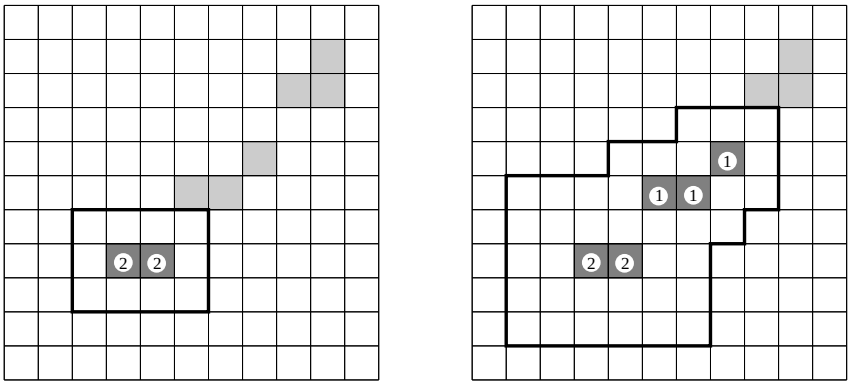
The above proof is constructive; it provides a reconstruction algorithm for the original image from the resulting "sampled" Skeleton. It consists of calculating at each step n, which varies from N down to 0, the set X_n according to (8.10). This can be implemented in the following way (illustrated by Fig. 8.5):

1. An intermediate image, which we call Z, is created and initially set to the highest Ultimate Erosions, i.e., $Z = U_N$.
2. $n \leftarrow N - 1$. (We assume $n \geq 1$, otherwise, we trivially obtain $X = U_N$).
3. $Z \leftarrow Z \oplus B(n)$. At this point Z is equal to Y_{n+1}.
4. The points of S that descend from Z, under C_n, in addition to U_n, are equal to S_n (this is a consequence of Lemma 1). If C_n is symmetric, then the above means that S_n are those connected components of S which "touch" Z, in addition to the Ultimate Erosions U_n (see Fig. 8.5).
5. $Z \leftarrow Z \cup S_n$. At this point Z is equal to X_n.
6. If $n = 0$, then stop, and $X = Z$. Otherwise, $n \leftarrow n - 1$.
7. Go to step 3.

The above algorithm is also the heart of the coding scheme proposed in section 8.4.

The following corollaries are direct consequences of (8.10).

Corollary 3 *If s is a Skeleton point with radius n, then all the Skeleton points which descend from it, under $C(n)$, have also radius n.*



(a) (b)

Figure 8.5: Reconstruction algorithm from a Skeleton with sampled Quench Function. The Skeleton is the same as in Fig. 8.3. $N = 2$, and U_2 is indicated by the dark gray pixels in (a). (a) Step n=1 of the algorithm, (b) Step n=0 of the algorithm. In (a) and (b), Z is indicated by the thick line, and the points in S_n are those connected components of S (the light gray pixels), touching Z (in this case, U_n, $n = 0, 1$, are empty).

Corollary 4 *Suppose that $C(n)$ is* symmetric*, i.e., $c \in C(n) \Rightarrow -c \in C(n)$.*

In this case, if s is a Skeleton point with radius n, then all the Skeleton points in the connected component to which it belongs (with connectivity being under $C(n)$) have also radius n.

According to Corollary 4, if the decomposition elements are symmetric, then not all the Ultimate-Erosion points need to have their radius stored! For every connected component in the set of ultimate erosions, one needs to store only the radius 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 consideration, only a small percentage of them need to have their radius stored. This provides a sampling scheme of the Quench Function[1]. (See Fig. 8.4(b)). Similar results can be deduced for non-symmetric decomposition elements, by means of Corollary 3.

Corollary 5 *A Skeleton point s has radius n* if and only if *s belongs to U_n,* **or** *$s \notin Y_{n+1}$ but s descends from Y_{n+1}, under $C(n)$.*

The above corollaries are used in the Coding scheme proposed in section 8.4.

8.3.2 Deterministic Prediction

The second theorem on which the proposed scheme is based is presented below. It permits *deterministic prediction* of information about S_n from the knowledge about the previously coded points.

Suppose a Coding procedure where, at a certain step, the Skeleton Subset of order n, S_n, is to be coded, and that Y_{n+1} is known to be available both to the coder and the decoder. Since

[1]Quench Function is defined in Chapter 2 as the function relating to each Skeleton point the radius of the related maximal element.

$S_n = X_n - Y_{n+1}$, it follows that there are no points of S_n inside the region Y_{n+1}. Therefore the coder does not need to code the *status* (whether belonging, or not, to S_n) of the pixels inside Y_{n+1}, and the decoder does not need to "look for" Skeleton points in that region at that moment. Particular versions of this result was used in the coding schemes proposed in [29].

It turns out that there is also a region *outside* Y_{n+1} that can be predicted not to contain Skeleton points from S_n. This region can be characterized by the following theorem:

Theorem 5 *Let $p \in E$. If the following holds:*

$$[(Y_{n+1} \cup \{p\}) \bullet A(n)] \circ B(n) \supset \{p\} \tag{8.11}$$

then p cannot belong to S_n.

Proof *The proof is by contradiction. Suppose that p is in S_n, and let us define the following operator:*

$$\rho(Z) \stackrel{\triangle}{=} [Z \ominus A(n)] \circ B(n). \tag{8.12}$$

By definition of Y_{n+1}, $\rho(X) = Y_{n+1}$. Also the set $Y_{n+1} \oplus A(n)$ gives Y_{n+1} when operated upon by $\rho(\cdot)$.

Therefore, since $\rho(\cdot)$ is an increasing operation (it is composed of basic morphological operations, which are increasing), then any set Z_0, such that $[Y_{n+1} \oplus A(n)] \subseteq Z_0 \subseteq X$, satisfies $\rho(Z_0) = Y_{n+1}$. In particular, $Z_0 = (Y_{n+1} \cup \{p\}) \oplus A(n)$, $p \in S_n$, satisfies it.

However, according to (8.11), $\rho(Z_0) \supset \{p\}$, and, therefore, $p \in Y_{n+1}$, which contradicts that $p \in S_n$. □

Theorem 5 provides a test for each point in E: If it passes it, i.e. (8.11) holds, then its status as a Skeleton point need not to be coded because it is known to both coder and decoder to be negative. On the other hand, if the test fails ((8.11) does not hold), nothing can be said about the point's status, and it must be coded.

The above test is however practically inviable, because it is extremely computation-demanding. Luckily, a simplified, much faster test is possible in many cases by the following corollary:

Corollary 6 *Let F be a structuring element,* not containing the origin *(which we denote o), and satisfying:*

$$[(F \cup \{o\}) \bullet A(n)] \circ B(n) \supset \{o\} \tag{8.13}$$

and let $p \in E$.

If $\{p\} \oplus F \subseteq Y_{n+1}$, then p cannot belong to S_n.

In other words, one can pre-select a template F, excluding the origin, and usually containing *few points*, such that it satisfies (8.13). Since it is independent of the input image X, the above selection is done "off-line", and only once for a given decomposition family $\{A(n)\}$. During an "on-line" Coding algorithm, the "prediction test" is performed, for each point p, by placing F "on" p, and examining the status of the points indicated by the template.

The points found in the above test are only a *subset* of the "predictable points" found in the test of Theorem 8.11. In order to find *all* the predictable points, a family $\{F_i\}$ of *all* the

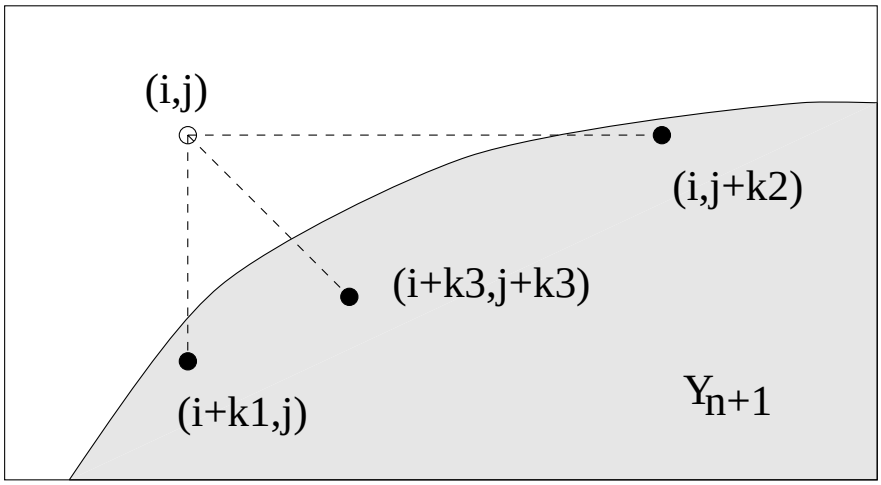


Figure 8.6: A point (i, j) predicted not to belong to S_n according to Corollary 7.

templates satisfying (8.13) should be defined, and the test in Corollary 6 must be repeated for each F_i. This could also be very computation-demanding. Often, however, a small subset of $\{F_i\}$ is enough for finding *most* of the desired points. As an example, let $E = \mathcal{Z}^2$, and consider a Skeleton decomposition of X, where B is a 3×3-*pixel squared structuring element.* In this case, Corollary 6 above can assume the following specific format:

Corollary 7 *Let $(i, j) \in \mathcal{Z}^2$, and consider a Morphological Skeleton with a 3×3-pixel* squared *structuring element. If any of the triplets*

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

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 above triplets represent a subset of the family $\{F_i\}$ related to the given squared structuring function. Fig. 8.6 shows an example of a point (i, j) which is predicted not to belong to S_n in this specific case. Fig. 8.7 shows another example; in this case, $n = 0$, and Y_{n+1} is indicated by the thick line. The dark points are those which can be predicted not to belong to S_0, according to Corollary 7.

8.4 Proposed Coding Scheme

In this section we propose an efficient coding scheme of the Skeleton Representation of binary images.

In comparison to the two previous schemes described in section 8.1, it is an hybrid method, since it takes into account the *Skeleton connectivity*, as the first scheme, and is

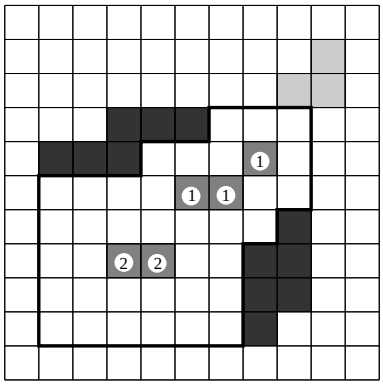


Figure 8.7: An example of deterministic prediction. The thick line indicate Y_{n+1}. The dark points cannot be Skeleton points in S_0, according to Corollary 7.

suitable to *progressive transmission*, as the second one. Moreover, it is based on the new theoretical properties of the Skeleton Representation, presented in last section, 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 8.6 below).

The proposed Coding scheme is restricted to discrete-family Generalized-step binary Skeletons, defined on $\mathcal{Z}^2$ (discrete binary images). An adaptation of this scheme for discrete-family Generalized-Step *Grayscale* Skeletons is presented in Appendix C, section C.1.

8.4.1 The Algorithm

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 radius.

Initially, for each of the Ultimate Erosions U_n, $0 \leq n \leq N$, a set $\tilde{U}_n$ is formed, in such a way that if a point s belongs to $\tilde{U}_n$, then it does not descend from any other point in $\tilde{U}_n$, under $C(n)$. If $C(n)$ is symmetric, the above means that $\tilde{U}_n$ contains only one point of each connected component of U_n, under $C(n)$. The points in the above sets $\tilde{U}_n$ have their position and radius 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 $\tilde{U}_n$. External boundary of a set A is considered here to be the points outside A, which are direct descendants of points in A, under $C(n)$.

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 5; 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 4), and their position are coded by an arithmetic coder [16]. 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, $0 \leq n \leq N$. Form the sets $\tilde{U}_n$ as specified above.
2. $n \leftarrow N-1$. $Y_N \leftarrow \emptyset$.
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 5) 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.[2]
7. If a "1" was sent, $Z \leftarrow (Z \cup \{p\})$. Otherwise, go to step 4.
8. Recursively, scan the direct descendants 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. $n \leftarrow (n-1)$. $Y_{n+1} \leftarrow Z \oplus B(n)$. Go to step 3.

8.5 Coding after Redundancy Reduction

The theoretical results and the algorithm presented above are related to Skeletons with *all* the centers of maximal elements, i.e., for which no redundant point is removed. Before presenting simulation results (sections 8.6 and 8.7) for the above coding scheme, let us consider adaptations of the algorithm for Redundancy Reduced Skeletons, or Minimal Skeletons.

8.5.1 Coding of Reduced Skeletons

Let us consider a Redundancy-Reduced Generalized-Step Skeleton, obtained by the general approach presented in chapter 6. In this case, equation (6.14), on page 74, is written in the following form:

$$RS_n = X_n - \begin{pmatrix}\text{redundant}\\ \text{region of}\\ \text{order } n\end{pmatrix} \ominus A(n) \tag{8.14}$$

[2] A *probability model* in an arithmetic coder is the collection of the probabilities of appearance of each of the symbols to be coded. An adaptive model is an empiric probability model, for which all the probabilities are initially set to $1/M$, where M is the number of symbols, and, during the coding process, updated to $(m_i+1)/M$, where m_i is the number of appearances of the symbol i.

Let us redefine Ultimate Erosions in the following way:

$$U_n = X_n - \mathrm{Rec}\left\{X_n, \begin{pmatrix}\text{redundant}\\ \text{region of}\\ \text{order } n\end{pmatrix} \ominus A(n)\right\}_{C(n)} \tag{8.15}$$

With this redefinition, Theorem 4 (page 98) is valid also for the above Reduced Skeleton, whatever is set as "redundant region of order n". The proof is very similar to that of Theorem 4.

Moreover, since the resulting Reduced Skeleton is a *subset* of the original Skeleton, Theorem 5 too remains valid.

Therefore, the proposed coding scheme can be directly adapted for Reduced Skeletons, such as the versions of $RS^{(1)}$ and $RS^{(2)}$ for Generalized-Step Skeletons.

8.5.2 Coding of Minimal Skeletons

Let us consider now a Minimal Generalized-Step Skeleton. In this case, as before, Theorem 5 remains valid, and an adaptation of the definition of Ultimate Erosions makes Theorem 4 stay valid as well.

The difference, now, is that the adapted Ultimate Erosions is such that it can be calculated only after the whole Minimal Skeleton $\{MS_n\}$ is computed. Moreover it has to be calculated in decreasing order of n. The redefinition in this case is:

$$U_n = MS - \mathrm{Rec}\{MS \cup Z_n, Z_n\}_{C(n)} \tag{8.16}$$

where:

$$MS \triangleq \bigcup_{m\geq 0} MS_m \tag{8.17}$$

$$Z_n \triangleq \bigcup_{m>n} MS_m \tag{8.18}$$

With the above considerations, the coding scheme is suitable also for Minimal Generalized-Step Skeletons.

8.5.3 Which is Preferable?

Does the proposed coding scheme perform better with the original Generalized-Step Skeleton, a Redundancy-Reduced version of it, or one of its Minimal Skeletons?

The answer is not totally clear because of the following trade-off between redundancy and connectivity. The less redundancy, the fewer points have to be coded, but part of the connectivity is lost and the number of Ultimate Erosions grows. The more connected is the Skeleton, fewer Ultimate Erosions are expected to be found, but all the redundancy has to be coded.

On the other hand, the connectivity is mainly related to *Intralevel* redundancy. Therefore, an Interlevel-Free Skeleton ($RS^{(2)}$) seems to be the most suitable type of Skeleton, among the above indicated ones, for the proposed coding scheme, because it removes as much as possible redundant points, but keeping the Intralevel redundancy, and therefore preserving connectivity.

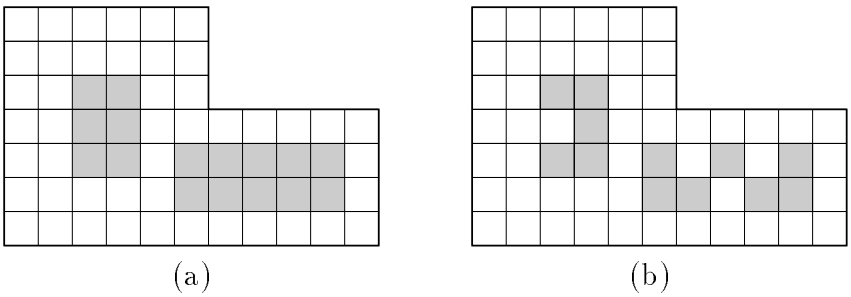


Figure 8.8: A Redundancy Reduction scheme for a Skeleton calculated with a 3×3-squared structuring element. (a) Original Skeleton, (b) Reduced Skeleton.

8.5.4 Coding with the Squared Structuring Element

The simulation results, presented in the next sections, relate to a Skeleton calculated with a 3×3-square structuring element, as constant generator. This structuring element was chosen, so that Corollary 7 can be used.

For the above structuring element, most of the redundancy is Intralevel. Therefore, although a $RS^{(2)}$ kind of Skeleton seems to be the most appropriate for the proposed coding scheme, it was not used in the simulations presented below, since $RS^{(2)}$ is expected to be very close to the original Skeleton, on one hand, and its computation time is expected to be much longer, on the other hand.

A particular Redundancy scheme, suitable only for the Skeleton with the 3×3 square structuring element, was used instead. It removes a large part of the Intralevel redundant points, without affecting the connectivity. The redundancy scheme consists of *sequentially* discarding points of the Skeleton (with radius greater than 0), for which at least 3 out of the 4 closest neighbors are also Skeleton points. Fig. 8.8 demonstrate the result of applying the scheme in a simple image. Note that the representation remains error-free.

8.6 Simulation Results (Binary Images)

The simulations of the proposed scheme, presented here, relate to the a Skeleton with constant generator $B(n) = B$, equal to the 3×3 squared structuring element centered at the origin. Moreover, $B(0) = B$.

Two sets of simulation tests are presented.

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. 5.5(b), on page 62), and the results, in bits-per-pixel, are presented in Table 8.1. 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 scanned documents (fax), and compares it to existing standard coders [1, 14]. 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 8.2 compares the size of the obtained

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 8.1: 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 8.2: File sizes of compressed facsimile standard CCITT documents, obtained by the proposed Skeleton algorithm, compared to previous Skeleton-Based coder and existing standards.

	Lena				
	8 bpp	5 bpp	3 bpp	Mosaic	Map
Ziv-Lempel ("Compress")	4.74	2.12	1.35	1.02	2.39
Error-free JPEG	7.06	2.67	1.04	-	1.61
Binary Skeleton	-	-	1.21	-	1.51
Grayscale Skeleton	**7.55**	**2.37**	**0.72**	**0.52**	**1.42**

Table 8.3: Simulation results for the proposed Grayscale Skeleton-based coding scheme.

coded files with the results given in [1] 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.

8.7 Simulation Results (Grayscale Images)

The same algorithm presented above for Skeleton-based coding of binary images can be adapted for coding Grayscale Skeletons. Such a generalization is described in Appendix C, section C.2. In this section, we present simulation results for the Grayscale coding scheme.

The Skeleton used in the simulations was with a constant and flat 3×3-pixel squared family generator (structuring element). Only *error-free* coding was investigated. This is because the influence of quantization on the theoretical results (Theorems 4 and 5) are still to be studied.

The results are summarized in Table 8.3. As the results in the first column indicate for the 512×512-pixel, 8 bit-per-pixel image "Lena" (Fig. 8.9(a)), the proposed Grayscale scheme does not provide good compression for *natural images*. By natural images, we mean images with large areas of slow variation in the gray levels. The reason for the incompatibility is: Images with slow gray-level variation require a great number of flat decomposition elements for a perfect representation, as demonstrated in Fig. 8.10(a).

On the other hand, images containing large flat areas, and abrupt gray-level variations, can be more efficiently represented by flat Skeleton elements, as one can see in Fig. 8.10(b). This assumption is confirmed by the remaining results in Table 8.3. Column 2 presents the results for the 5 bit-per-pixel "Lena" seen in Fig. 8.9(b), obtained by discarding the 3 least-significant bit-planes of "Lena". We can see that the proposed Skeleton, in this case, provides results similar to those of the "compress" algorithm, and the error-free JPEG. The advantage of the Grayscale Skeleton over the other methods for flat images is stressed in the third column of the table, where the results for "Lena" with only its 3 most-significant bit-planes (Fig. 8.9(c)) is presented. Note also the advantage of the Grayscale Skeleton-based

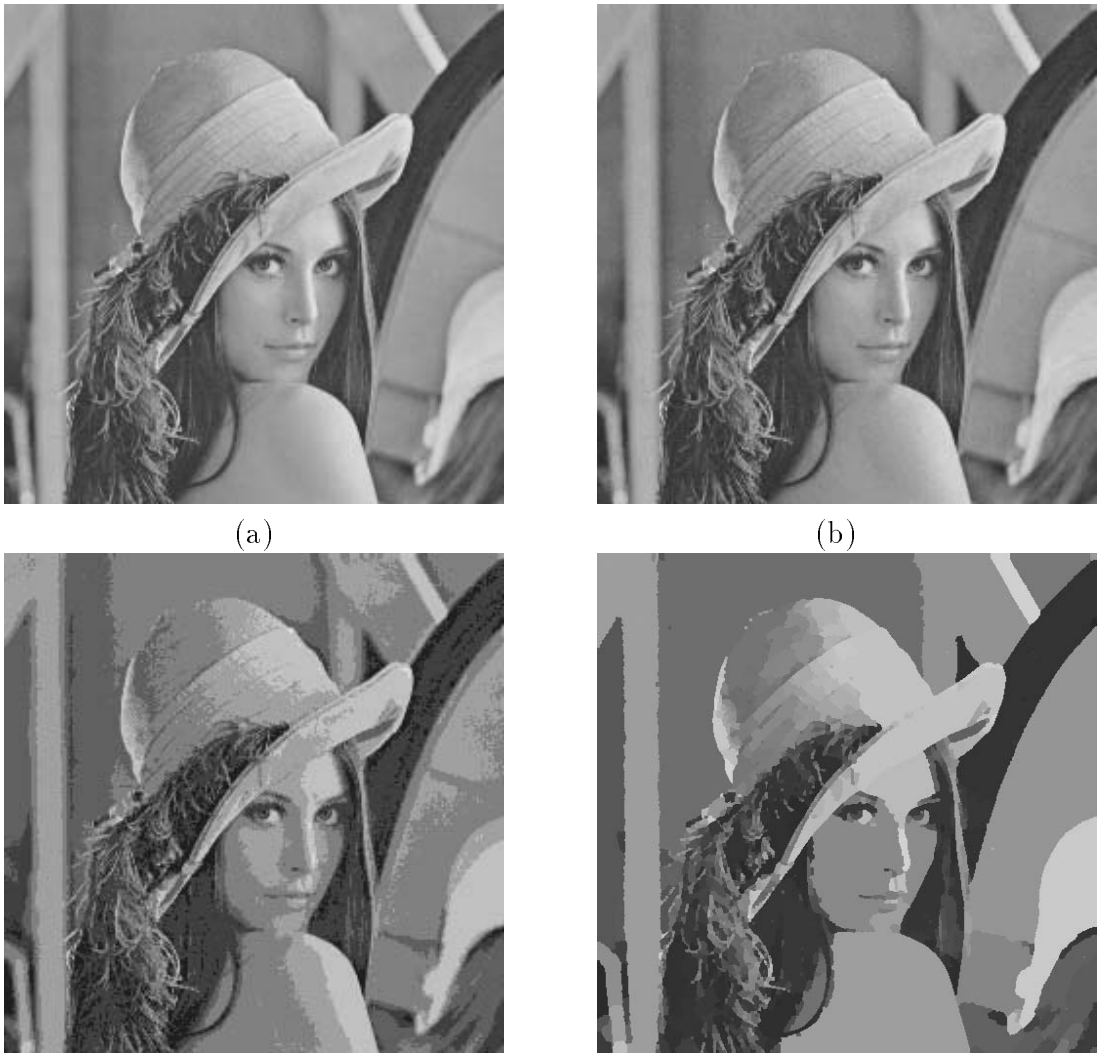


Figure 8.9: Grayscale images, used for the simulations. (a) 8 bit-per-pixel, 512×512 "Lena", (b) 5 bit-per-pixel "Lena", (c) 3 bit-per-pixel "Lena", and (d) "Mosaic" of "Lena", obtained by means of a segmentation procedure.

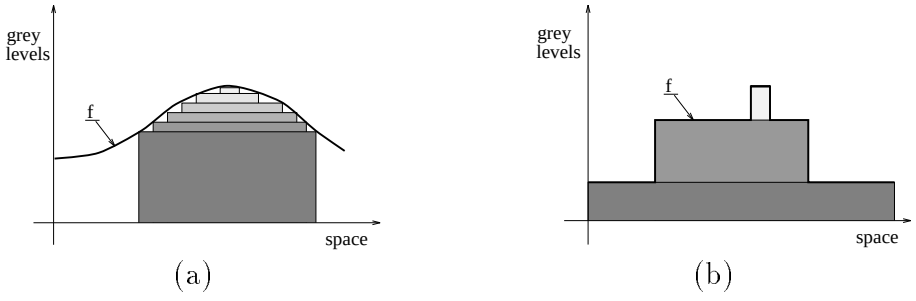


Figure 8.10: The efficiency of the Grayscale Skeleton Representation. (a) Slow gray-level varying functions are not well represented, whereas (b) flat and abrupt varying functions are more efficiently represented.

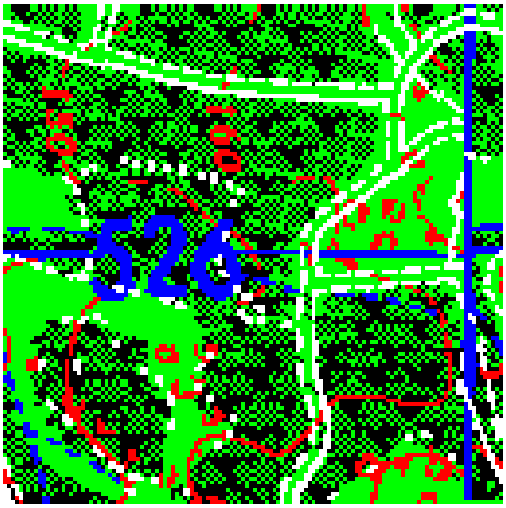


Figure 8.11: A 5-layer 128 × 128 multi-layer map.

method over the Binary Skeleton-based method, applied to each of the 8 layers separately.

A 3-bit-per-pixel version of natural images are not typically of interest; on the other hand some important applications do relate to flat images. This is the case of Segmentation-based Coding, described and considered in chapter 9, for which the result of a segmentation process can be summarized in the form of a "Mosaic" image (see Fig. 8.9(d)). The gray level of a pixel in the presented "Mosaic" image is equal to the average of gray levels in the segment to which it belongs. According to column 4 in Table 8.3, such an image is more efficiently coded by the proposed scheme than by the other investigated methods. As seen in chapter 9, this can be used as part of a Segmentation-based coding scheme of natural images.

Another suitable kind of images for Grayscale Skeleton-based coding is that of *multi-layer maps*. The layers are arranged in the form of a grayscale image, and then coded by the method. The results for the 5-layer 128 × 128-pixel map, presented in Fig. 8.11, is shown in the last column of Table 8.3, and the advantage of the proposed scheme can be observed.

Chapter 9

Linear Versus Morphological Methods in Image Representation

In this chapter, a comparison is performed between the Morphological methods for image representation, considered throughout this thesis, and the Linear methods, which are the most common and well-known ones in Image Processing and Representation. Furthermore, hybrid methods, combining both approaches, are considered.

Section 9.1 below reviews the many algebraic *similarities* between the two methods; more particularly, between the Linear mathematical framework (Linear Spaces) and the Morphological mathematical framework (Complete Lattices), between Convolution and Dilation and Erosion, and between some Linear representations (Wavelets and Laplacian Pyramids) and the Skeleton representation.

Section 9.2, on the other hand, stresses the basic *differences* between the methods, from the Image Processing point of view. It turns out that Linear methods better represent *smooth* regions in a natural grayscale image, whereas Morphological methods better represent *edges*.

Section 9.3 considers hybrid methods, which combine the two methods in order to better represent both smooth regions and edges. The rapidly growing field of Segmentation-based coding is considered as a particular case of this approach. The potential contribution of the Grayscale Skeleton-based coding algorithm presented in chapter 8 to this area is demonstrated by a computer simulation.

9.1 Algebraic Similarities

9.1.1 Framework comparison

Linear Image Processing is based on *Linear Spaces*. These are sets, to which a *sum* operation is attached. This sum operation induces a certain structure for the space, characterized by a scaling operation called *multiplication by a scalar*. The most important operators in a Linear Space $\mathcal{L}$ (let us denote them generically by ψ) are those which:

1. Commute with the sum operation, i.e., $\psi(\sum x_i) = \sum \psi(x_i)$, $\forall \{x_i\} \subset \mathcal{L}$.

2. Preserve the scaling structure of the elements in the space, i.e., $x = a \cdot y \Rightarrow \psi(x) = a \cdot \psi(y)$, $x, y \in \mathcal{L}$, $a \in \mathcal{C}$ (set of complex numbers).

Such an operator is called a *linear operator*.

As described in chapter 2, Mathematical Morphology is based on *Complete Lattices*. These are sets, to which *two* operations are attached: *supremum* and *infimum*. These operations induce in the space a certain structure, characterized by the *order relation* ($\leq$). The most important operators in a Complete Lattice $\mathcal{P}$ are those which:

1. Commute either with the supremum operation, i.e., $\psi(\vee X_i) = \vee\psi(X_i)$, or with the infimum operation, i.e., $\psi(\wedge X_i) = \wedge\psi(X_i)$, $\forall\{X_i\} \subset \mathcal{P}$.

2. Preserve the ordering structure of the elements in the space, i.e., $X \leq Y \Rightarrow \psi(X) \leq \psi(Y)$, $X, Y \in \mathcal{P}$. Or, in other words, ψ is *increasing*.

Among such operators, the ones which commute with the supremum are called *dilations*, and those which commute with the infimum are called *erosions*.

This analogy between the two frameworks, pointed out by Serra in the preface of [52], makes several notions and structures be common for both approaches.

For instance, the most useful operator in Linear Image Processing is *convolution*. It is the only one in a Linear space, which is both *linear* and *translation invariant*. (A translation invariant operator satisfies: $\psi[T(x)] = T[\psi(x)]$, where $T(\cdot)$ is a translation). In the Linear Space of 2-D functions from $\mathcal{Z}^2$ to $\mathcal{R}$, the convolution has the form:

$$[f * g](x) = \sum_{y \in \mathcal{Z}^2} [f(y) \cdot g(x - y)] \tag{9.1}$$

In Binary and Grayscale Morphology, the most useful operators are the Minkowski sum and subtraction, $\oplus$ and $\ominus$, respectively. The Minkowski sum is the only operator which is both a Dilation and translation invariant, whereas the Minkowski subtraction is the only translation invariant Erosion. As pointed out in chapter 2, for Grayscale Morphology, these operators assume the following form, in the discrete case:

$$[f \oplus g](x) = \bigvee_{y \in \mathcal{Z}^2} [f(y) + g(x - y)] \tag{9.2}$$

$$[f \ominus g](x) = \bigwedge_{y \in \mathcal{Z}^2} [f(y) - g(y - x)] \tag{9.3}$$

Note the resemblance between the structure of the convolution and the above operators.

In [31, 32], Maragos transposes many of the concepts known in Linear Image Processing to Mathematical Morphology of functions. The notion of a *Dilation System* was defined, as an analogy to Linear Systems, as an operator $\mathcal{D}$ satisfying:

$$\mathcal{D}\left\{\bigvee_i [c_i + x_i(v)]\right\} = \bigvee_i \{c_i + \mathcal{D}[x_i(v)]\} \tag{9.4}$$

The *impulse signal* for Grayscale Morphology is defined there in the following way:

$$\mu(v) \triangleq \begin{cases} 0, & v = 0, \\ -\infty, & v \neq 0. \end{cases} \tag{9.5}$$

In classical linear processing, Translation-Invariant Linear Systems are totally characterized by their *impulse response*. This is also the case for Translation-Invariant Dilation Systems. Maragos showed [32] that any Translation-Invariant Dilation System is of the form:

$$\mathcal{D}(x) = x \oplus g \tag{9.6}$$

where g is the impulse response $\mathcal{D}(\mu)$. As the reader can notice, this is the same Minkowski sum as considered above, where g is the structuring element. Therefore, a structuring element is the analogous in Morphology for the impulse response in Linear Systems.

A similar transposition can be performed for *Erosions Systems*, since erosion is dual to dilation.

Also in [31, 32], Maragos proposed an analogy in Morphology for the *frequency response* of signals. It turns out that, just like harmonic signals are the eigenvectors for Linear Translation-Invariant Systems, hyperplanes are the eigenvectors for Dilation and Erosion Translation-Invariant Systems. Since harmonic signals are characterized by constant *frequency*, and hyperplanes are characterized by constant *slope*, one concludes that *slope* in Morphological systems is the analogous of *frequency* in Linear systems. An *A-transform*, which is the analogous of the *Fourier transform*, was defined in [31], but this can be applied consistently only to *convex functions*.

Parallel and independently to Maragos, Dorst and Boomgaard [8, 9] proposed the same analogy of frequency response by slope response, which they, called *Slope Transform*. Their transform "contains" the A-transform proposed by Maragos, in the sense that the former can be applied for any function and is invertible.

9.1.2 Generic Image Representations

Representation Models

Linear Image Representation is based on the following model:

$$f(x) = \sum_{i \in I} a_i \cdot \varphi_i(x) \tag{9.7}$$

where $\{\varphi_i(x)\}$ is a family of pre-defined functions, and $\{a_i\}$ (called representation coefficients) is a representation of $f(x)$ according to that family.

If the family $\{\varphi_i(x)\}$ is real and *orthonormal*, and supposing that there exists a set of coefficients $\{a_i\}$ that satisfies (9.7), then the representation coefficients can be calculated by the scalar product:

$$a_i = \sum_{x \in \mathcal{R}^2} f(x) \cdot \varphi_i(x) \tag{9.8}$$

If the functions $\varphi_i(x)$ are harmonic functions each in a spatial frequency, then $\{a_i\}$ is the frequency response of $f(x)$.

The above is well-known in Linear Image Processing, and has the following analogy in Mathematical Morphology. The most common model for morphological image representation is the following:

$$f(x) = \bigvee_{i \in I} [a_i + \varphi_i(x)] \tag{9.9}$$

If the functions $\varphi_i(x)$ have *disjoint* regions of support (which is the analogous of saying that the family is orthogonal), and if there exists $\{a_i\}$ for which (9.9) is satisfied, then the representation "coefficients" are given by:

$$a_i = \bigwedge_{x \in \mathcal{R}^2} [f(x) - \varphi_i(x)] \tag{9.10}$$

If the functions $\varphi_i(x)$ are hyperplanes each with a different spatial slope, then $\{a_i\}$ is the slope response of $f(x)$.

Projections and Partial Representations

Equation (9.7) can be seen as characterizing a map from the space of indices I to the image space $\mathcal{R}^2$. This map is a *linear* operator because it preserves linear superposition. In addition, (9.8) characterizes another linear map, this time from $\mathcal{R}^2$ to I. If there exists a unique $\{a_i\}$ satisfying (9.7) for each function $f(x)$, then this second map is the inverse of the first.

Similarly, equation (9.9) characterizes a *dilation* from I to $\mathcal{R}^2$, since it preserves superposition by supremum. Equation (9.10) is actually the *adjoint erosion* from $\mathcal{R}^2$ to I, of the above dilation.

Let us consider now the case where there does not exist a collection of coefficients $\{a_i\}$ which satisfy the linear representation (9.7) for some $f(x)$. Still, a set of coefficients $\{a_i\}$ can be calculated by the inverse mapping (9.8). The question which is asked is: What is the relation between the function $\tilde{f}(x)$, obtained by the direct mapping characterized by (9.7), and the original function $f(x)$? It is well known in Functional Analysis, that $\tilde{f}(x)$ is the *projection* of $f(x)$ on the subspace generated by $\{\varphi_i(x)\}$. This process of projection in idempotent, and the resulting projected element is the element in the subspace which is the closest to $f(x)$.

Analogous conclusions can be drawn for the Morphological Representation. Since (9.9) and (9.10) characterize, respectively, a dilation and an erosion, which we denote by $\hat{\delta}$ and $\hat{\varepsilon}$, we can rewrite those equations in the form:

$$f = \hat{\delta}(a) \tag{9.11}$$

$$a = \hat{\varepsilon}(f) \tag{9.12}$$

where f denote the original function $f(x)$, and a the set of coefficients $\{a_i\}$. Usually, though, the above system cannot be satisfied for all f, and the first equation should be written:

$$\tilde{f} = \hat{\delta}(a) \tag{9.13}$$

By substituting (9.12) in (9.13), we get:

$$\tilde{f} = \hat{\delta}\hat{\varepsilon}(f) = \hat{\gamma}(f) \tag{9.14}$$

where $\hat{\gamma}$ is the morphological opening associated with $\hat{\delta}$ (see section 2.4.4).

Like any opening, $\hat{\gamma}$ is idempotent. The function $\tilde{f}$ is an element of the subspace of functions which can be represented by means of $\{\varphi_i\}$, therefore this subspace is "generated"

by that family. Furthermore, $\tilde{f}(x) \leq f(x)$, $\forall x \in \mathcal{R}^2$, since $\hat{\gamma}$, like every opening, is anti-extensive. And finally, $\tilde{f}(x)$ is the "biggest" function (that with the highest values for each x) in the above subspace, which is "smaller" than $f(x)$ (again this is a property of every opening). The conclusion is that $\tilde{f}(x)$ is the projection of $f(x)$ in the space of functions morphologically represented by $\{\varphi_i(x)\}$, such that it is the "closest" one to $f(x)$ (the error $f(x) - \tilde{f}(x)$, which is always positive, is the smallest, for all x).

Therefore, a provides in this case the "best" partial representation of f in terms of $\{\varphi_i(x)\}$.

If, on the other hand, the family $\{\varphi_i(x)\}$ in *not* disjoint, then generally the representation obtained by (9.10) is *redundant*, and a subset of this representation is to be calculated (see, for example, the discussion and general approach in chapter 6).

9.1.3 Wavelets and Skeleton Representations

Let us consider here a Wavelets Representation and a Grayscale Skeleton Representation as particular cases of the above Linear (9.7) and Morphological (9.9) Image Representations, respectively.

A simple Grayscale Skeleton $\{S_n(x)\}_{n\in\mathcal{N}}$, generated with a constant structuring element $g(x)$, and calculated using the umbra approach (see section 4.3), represents the original image $f(x)$ in the following manner:

$$f(x) = \bigvee_{n\in\mathcal{N}} [S_n \oplus ng](x) = \bigvee_{n\in\mathcal{N}} \bigvee_{y\in\mathcal{R}^2} [S_n(y) + ng(x-y)] \tag{9.15}$$

where $ng(x)$ is the function obtained by the n-fold grayscale translation-invariant dilation of $g(x)$ by itself.

Now, if we assume I to be $\mathcal{R}^2 \times \mathcal{N}$, so that each index i is in the form (y, n), where y is a point in the 2-D space, and n is a non-negative integer, then by comparing (9.9) with (9.15) we conclude that, in the case of the Grayscale Skeleton:

$$\varphi_{(y,n)}(x) = ng(x-y) \tag{9.16}$$

and that $S_n(y)$ are the representation "coefficients".

Therefore, a grayscale image is represented, in terms of a Grayscale Skeleton, by the *superposition of dilated and translated versions of a single function*, $g(x)$. A *Wavelet Representation*, on the other hand, can be described in the very same way as the superposition of dilated and translated versions of a "Mother Wavelet". The basic differences rely on the nature of the superposition, which is linear for Wavelets, and morphological for the Skeleton, and the nature of the "dilation", which is actually a *scaling* for Wavelets, and a morphological dilation for the Skeleton.

9.1.4 Laplacian Pyramids and The Skeleton Representation

Although intuitively similar, the Wavelet and the Skeleton representations differ in many of their algebraic properties. For instance, the Wavelet representation is not translation invariant, whereas the Skeleton representation is. On the other hand, in the discrete case, the

number of representation coefficients in the Wavelet representation is identical to the number of pixels in the original image, whereas for the Skeleton representation it is much larger.

A Linear representation which is closer to the Skeleton, in terms of the above properties, is the Laplacian Pyramid. In section 4.1.1, the Skeleton has already been considered as a pyramid. Here we stress this, by comparing the way in which it is calculated to that of the Laplacian Pyramid.

Let us consider here the same discrete-family, constant generator, grayscale Skeleton, as that of the last section. The calculation of the Skeleton "subsets" $S_n(y)$ in this case is given by the following version of Lantuéjoul's Formula:

$$S_n = f \ominus ng - (f \ominus ng) \circ g \tag{9.17}$$

A simple algebraic manipulation on (9.17) leads to:

$$S_n = f \ominus ng - [f \ominus (n+1)g] \oplus g \tag{9.18}$$

Equation (9.18), together with the fact that $f \ominus ng = [f \ominus (n-1)g] \ominus g$, permit us to implement the Skeleton decomposition according to the algorithm described in Fig. 9.1(a). The reconstruction can be implemented as shown in Fig. 9.1(b).

Let us consider now the decomposition diagram (Fig. 9.1(a)), a little more closely. Notice that, if one replaces the "erosion" operations by a "decimation" (linear low-pass filtering followed by a down-sampling), and the "dilation" operation by "interpolation" (up-sampling followed by a linear low-pass filtering), then the resulting algorithm is that of the Laplacian Pyramid. Therefore, the Skeleton decomposition can be seen as a Laplacian-like pyramid, with the erosion acting as a decimation, and the dilation acting as an interpolation.

The similar structures of both representations lead to the above mentioned similar algebraic properties.

9.2 Differences Between the Approaches

We have seen that on one hand, there is a great similarity between the algebraic structures of Linear and Morphological Image Representations; yet, on the other hand, the two approaches lead to qualitatively very different results.

For demonstration, consider the two series of filtering operations presented in Fig. 9.2. The left column of Fig. 9.2(a) consists of a linear low-pass pyramid. Each image was obtained by averaging the pixel values of the original image (at the top), with a moving squared window with increasing sizes. The right column represents a band-pass pyramid, obtained by subtracting each pair of images of the low-pass pyramid. The left column of Fig. 9.2(b) consists of a morphological "low-pass" pyramid, obtained by performing a grayscale opening with squared structuring elements of increasing sizes. The sizes of the structuring elements are identical to the sizes of the windows used in the linear filtering operations. The right column represents the respective "band-pass" pyramid.

Notice that the edges of the image are smoothed and blurred as we move down along the Linear low-pass pyramid. Consequently, all the edges of the image appear in *all* the levels of the band-pass pyramid. In other words, the "energy" of each edge is spread out across

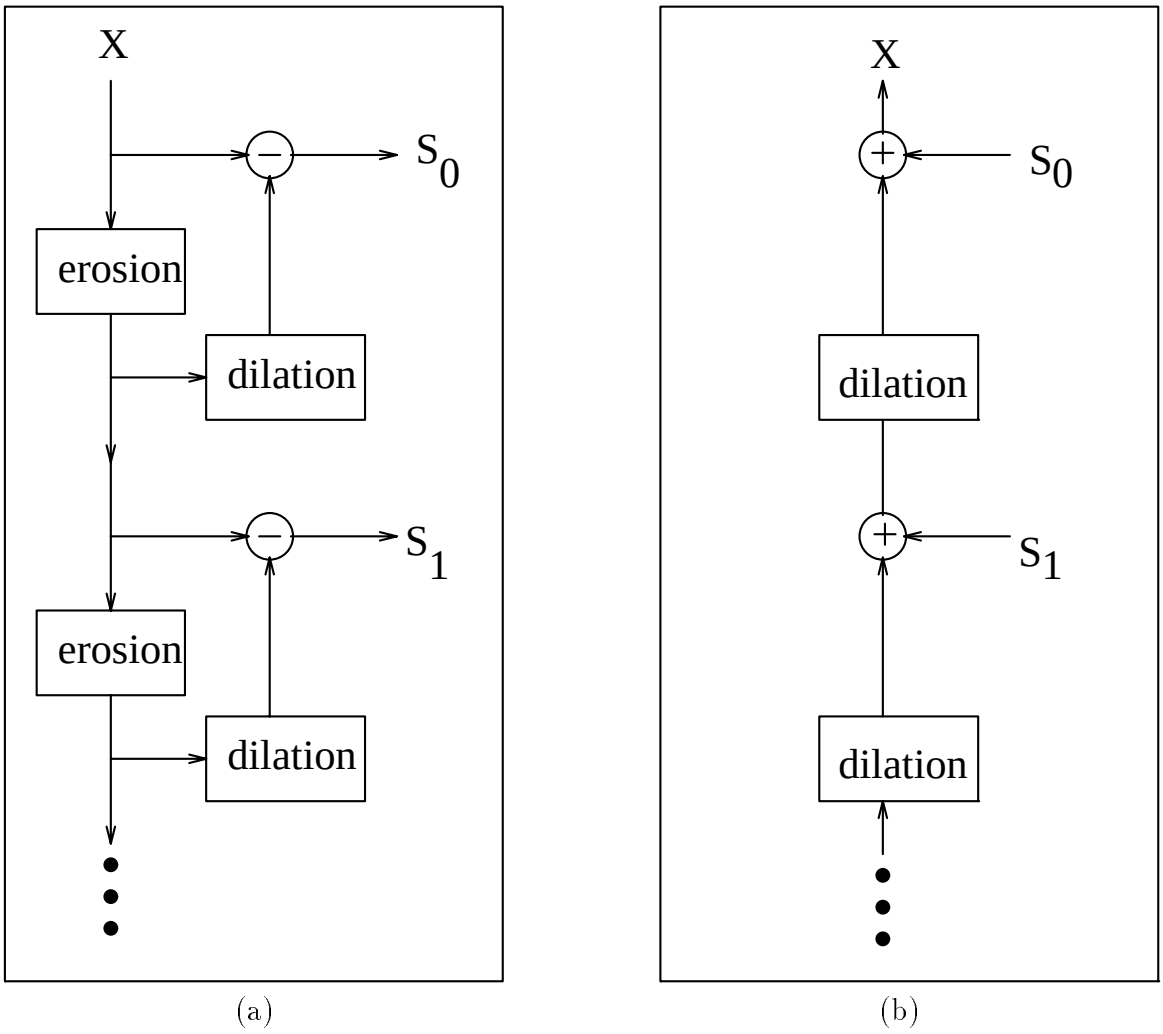
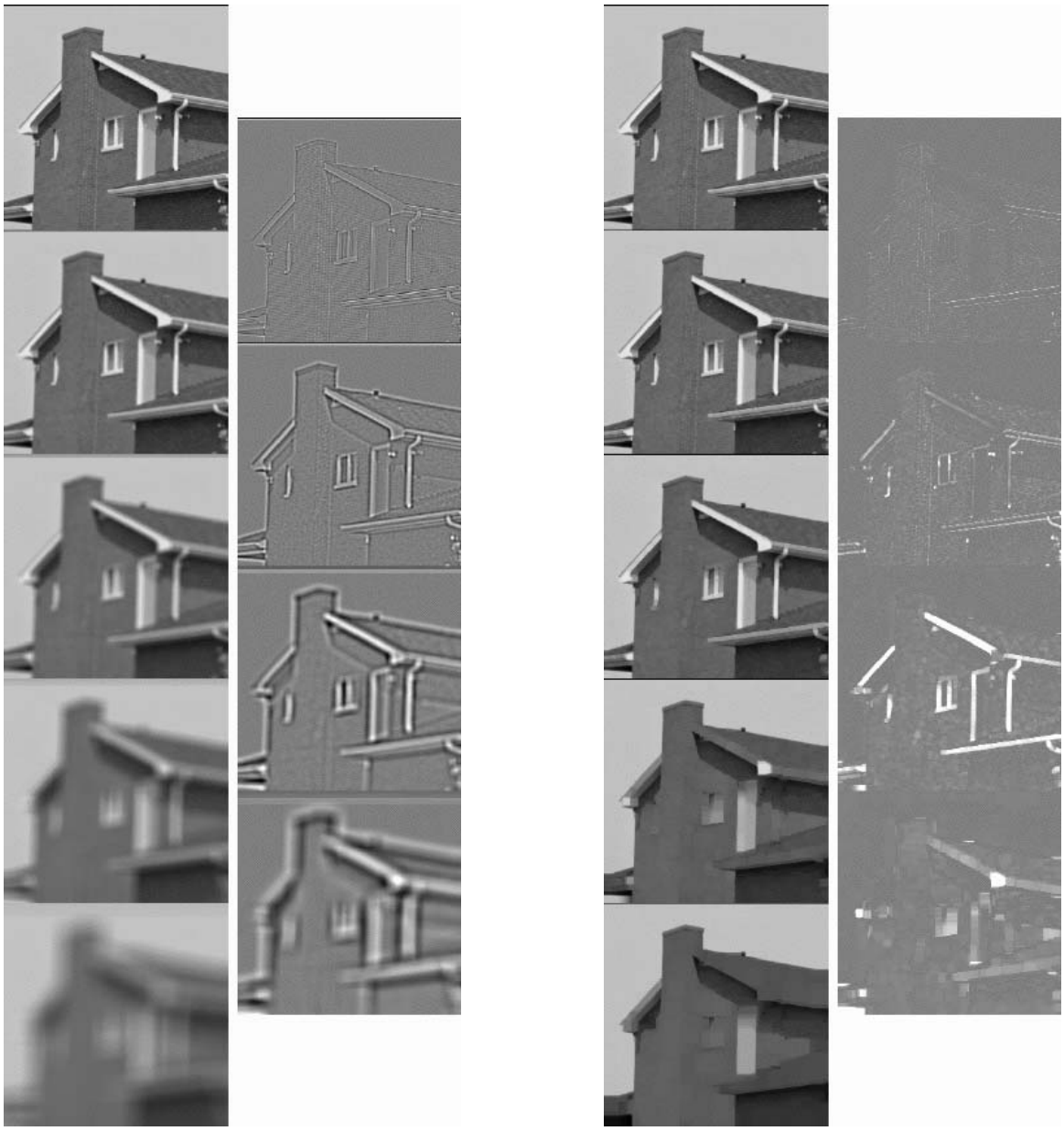


Figure 9.1: Calculation and reconstruction of the Skeleton decomposition. (a) Calculation diagram, where "erosion" denotes $(\cdot) \ominus g$, and "dilation" denotes $(\cdot) \oplus g$, (b) reconstruction diagram.



(a) (b)

Figure 9.2: Linear and Morphological Filtering. (a) Left column: Linear low-pass pyramid. Right column: Linear band-pass pyramid. (b) Left column: Morphological "low-pass" pyramid". Right column: Morphological "band-pass" pyramid.

the band-pass pyramid, and the conclusion is that representations based on multi-level linear filtering do not efficiently represent edges.

On the other hand, no smoothing or blurring is noticed in the low-pass morphological pyramid. The morphological filtering operations remove bright features of the original image, without affecting the edge contrast. The bright features that are removed, are retained in the band-pass morphological pyramid levels, according to their size (or width). Therefore, each edge does not have its "energy" spread out, but, on the contrary, it is concentrated in a specific band-pass pyramid level. The conclusion is that morphological representations are usually more suitable for representing edges.

The opposite conclusion is obtained for the *smooth regions* in an image, as already explained in section 8.7 (see Fig. 8.10 on page 108). That is, the morphological representations do not efficiently represent slow-varying regions of the image, whereas it is well known that the linear representations usually do. Similarly to the linear representation in relation to edges, the morphological representation usually makes the "energy" of a smooth region to be spread out across its representation levels, whereas the linear representation keeps is concentrated in a certain representation level.

9.3 Hybrid Methods

As discussed in the last section, the morphological and the linear representations are *complementary*, because the former efficiently represents edges and does not efficiently represent smooth regions, whereas the latter efficiently represents smooth regions and does not efficiently represent edges.

In this section, we consider *hybrid methods*, combining both approaches in order to obtain an efficient representation of both edges and smooth regions in images.

Combined Framework

The ideal hybrid representation would be one, which is based on a *combined mathematical framework*, containing as particular cases both linear spaces and complete Lattices. Such framework would make possible the construction of a combined image model, and its study.

However, this combined framework is now just an ideal. It looks like that such a unification is not possible at all. Therefore, other approaches, based at a less extent on a global mathematical framework are the ones in which we have to base ourselves.

Morphological Representation of a Linear Decomposition

A first alternative is to perform a linear decomposition of an image and, then, to represent the resulting representation by morphological methods.

The aim of such an approach is to try to gather back, by means of the morphological representation, the energy of the edges, which was spread out by the linear representation.

The method know as "Zero-tree coding of Wavelet Decompositions" [55] (which lately has found much interest, in the Wavelet-based coding field) could be seen as a representative

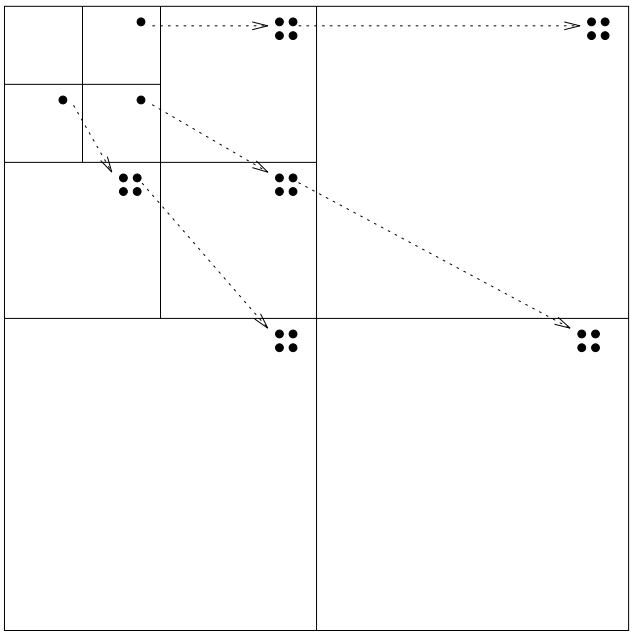


Figure 9.3: The relation between the coefficients in a Wavelet Decomposition, as considered by the zero-tree coding method. This relation can be considered as a *structuring function* in a Boolean Lattice.

of this approach, although the zero-tree coding is not recognized by the image processing community as a morphological method.

The zero-tree coding method takes into consideration that the edge energy is spread out in a Wavelet representation in an ordered manner. Since the original image is recursively down-sampled at a 2 by 2 ratio, each representation coefficient in a lower Wavelet subband is spatially related to 4 other coefficients in the immediately higher Wavelet subband (see Fig. 9.3). This relation structure can be carried out recursively, so that each point in a low subband can be seen as the root of a quad-tree. Since the energy spread in the Wavelet decomposition follows this same tree structure, then any coefficient having an absolute value below a given threshold is likely to be the root of a zero-tree, in which all the nodes are also coefficients with absolute value below the same threshold.

In summary, in the zero-tree method, one searches, for each threshold value (which assumes the values, 2^m, $m = 7, 6, \ldots, 0$), the zero-tree roots and codes them efficiently.

If we consider the Wavelet representation as a function over a Boolean Lattice, and the relation, depicted in Fig. 9.3, as a *structuring function* in that Boolean Lattice, then the zero-tree representation can be seen as a morphological representation, closely related to the Skeleton. Notice the similarity between the above structuring function and that of the Quadtree Decomposition, presented in section 5.2.1. Furthermore, by applying thresholds of the form 2^m, one is actually performing a *bit-plane decomposition* of the absolute value of the Wavelet Representation, and, as seen in chapter 5, this can also be considered as a Skeleton decomposition.

Therefore, in our opinion, the zero-tree coding can be considered as a morphological representation, and its study and generalization (which are outside the scope of this thesis)

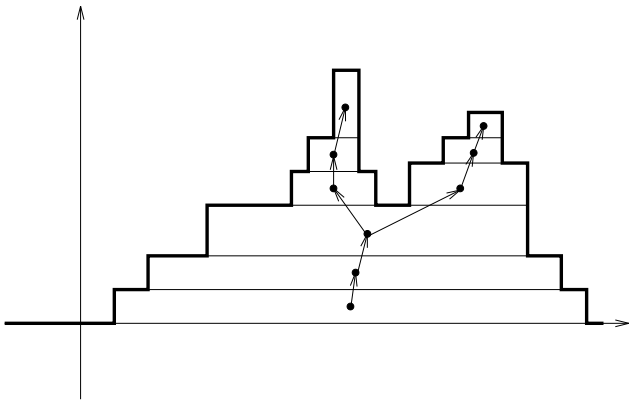


Figure 9.4: A simple Skeleton decomposition of a 1-D discrete signal, and the arrangement of its points into a connected graph.

can be performed by means of morphological methods.

Linear Representation of a Morphological Decomposition

Another alternative is the dual approach; first to perform the morphological decomposition and, then, to calculate its linear representation. The aim with this approach could be to try to gather back the "energy" of the smooth regions, which was spread out by the morphological representation.

For binary images (even though they do not contain "smooth regions"), the method of linear approximation of the Skeleton lines by parametrized arcs or splines (see [3]), could be considered as a representative of this approach.

For grayscale images, a similar method could be tried. First, a connectivity criterion should be defined, and connected Skeleton points under this criterion should be arranged in a graph (see Fig. 9.4). Then, a numerical function (or functions), describing this graph, should be defined. For example: Let $(x(t), y(t), z(t))$ describe the differences in coordinates, and gray level, between a Skeleton point t and its successor $t+1$. If a Skeleton point has more than one successor, then all of them (but one) can be disconnected from the graph and considered as roots of new graphs. The functions (in the above example $x(t)$, $y(t)$, and $z(t)$) would, then, be represented by linear methods. The implementation and analysis of the above proposed method is outside the scope of this thesis.

9.3.1 Segmentation-Based Representation

Probably the most popular hybrid scheme, in the last few years, is the segmentation-based representation and coding [43, 44]. This is part of the "Second Generation Image Coding Techniques", proposed by Kunt, Ikonomopoulos, and Kocher [26], who suggest to separate features of different nature in the image into different classes, and code them separately.

In segmentation-based coding, a segmentation of the original image into disjoint regions is first performed. Then the contents of each region (called the *texture* of the region) are coded individually, and the boundaries of the segments are coded separately. The most usual

linear method for coding the "texture" is polynomial approximation. The boundary of the segments are usually coded by chain code [13] or by the binary Skeleton [4].

Here we propose a segmentation-based coding scheme, using the Grayscale Skeleton-coding scheme proposed in chapter 8. The segmentation procedure is a version of the morphological segmentation algorithm, called "Watershed algorithm" [34, 61], which is recently of increasing interest [6, 33, 44, 13, 4].

After a segmentation is performed, a "Mosaic Image" is created, in the same way as described in section 8.7: each of its pixels is set to the average of the gray values inside the segment to which it belongs. A Mosaic image for "Lena", shown in Fig. 8.9(d), is presented here again in Fig. 9.5(b) (The original "Lena" is seen in Fig. 9.5(a)). This Mosaic image, as mentioned in section 8.7, is suitable for efficient coding by the proposed Grayscale Skeleton-based coding scheme.

On the other hand, the difference between the original image and the Mosaic image is coded by a linear method. Since the high contrasted edges of the original image are retained in the Mosaic image, this difference image is expected to contain mainly the slow-varying features of the smooth regions. For "Lena", the difference image is shown in Fig. 9.5(c).

The result of a very simplified implementation of the above idea is shown here. The difference image was *lossy* coded by JPEG, with a high "q-factor", at 0.28 bits per pixel. The use of a high q-factor in this case is possible since mainly low frequency features are assumed to be found in the image. The high frequencies are allowed to be distorted, since they usually belong to the interface between the segments, which are retained in the Mosaic image. The Mosaic image, on the other hand, can be errorless coded at 0.52 bits per pixel, as seen in section 8.7. The overall coding rate is therefore 0.8 bits per pixel, and the reconstructed image (with PSNR of 30.4dB) is seen in Fig. 9.5(d).

The above scheme is just a preliminary experiment, pointing out a possible approach. Much is still to be done. First, 2-D JPEG is certainly not the appropriate linear scheme for linearly coding the difference image. The texture inside each segment should be coded individually, avoiding the high frequencies due to the discontinuities at the interface of segments. Furthermore, the segmentation scheme should be improved and adapted to coding purposes (see [43, 44]). And, finally, a lossy coding of the Mosaic image should be considered, instead of a lossless one.

(a) (b)

(d) (c)

Figure 9.5: Segmentation-based coding scheme, based on the Grayscale Skeleton. (a) Original Image "Lena", (b) Mosaic image, (c) difference image, (d) Reconstructed image.

Chapter 10

Conclusions and Future Research

10.1 Conclusions

In this work the Morphological Skeleton Representation of images, and some of its applications (especially for Image Coding), are studied. The results and conclusions can be divided into three main groups:

Generalization of the Algebraic Framework: The Skeleton is not only a symmetry axis of planar shapes, consisting of a collection of connected lines. In its most generalized sense, it is also an algebraic hierarchical decomposition, and a pyramidal geometrical representation of several types of images (binary, grayscale, sequences of images, etc.).

The algebraic framework of the Skeleton was generalized several times in recent years, aiming to extend the scope of the Skeleton representation as much as possible. This work first adds a new evolutionary branch to the above development. Its basic contribution is in showing that an image can be decomposed into *multi-parameter* families of elements, using a generalization of the classical and efficient Lantuéjoul's Formula. Then it proposes a *Generalized Skeleton framework*, which unifies all the previous generalizations, including the Generalized Skeleton on Lattices proposed by Serra [53], and the Generalized MSES introduced by us. The above framework also permits the definition of new representations as particular cases, like the Hybrid Skeleton. The extension of such framework for grayscale images is also analyzed.

In addition, applications and special cases are presented and discussed. It is shown that the generalized Skeleton framework contains, as particular cases, image representations previously known, but not recognized as Skeletons, such as the Quadtree and the Bit-Plane decompositions. Moreover, multi-parameter generalizations of these decompositions are presented (specifically, a multi-parameter Quadtree decomposition for binary images, and a Quadtree/Bit-Plane decomposition for grayscale images).

Another Morphological Representation, called the *Two-Sided Skeleton*, is also proposed here. It does not belong to the Generalized Skeleton framework, but it seems to us that is has a great potential for Coding and Pattern Recognition. Its advantage is in yielding a Skeleton-like, self-dual (almost), morphological representation for binary and grayscale images.

Morphological Reduction of Skeleton Redundancy: An algorithm for obtaining Minimal Skeletons had been previously proposed [29]. However, this algorithm cannot be described in terms of morphological operations only, and therefore it is not amenable for a fully parallel implementation. In this work, we consider redundancy reduction by means of morphological operations only, which can be efficiently implemented on parallel machines.

Previously, all the redundant points in a Skeleton representation were considered as part of a single group. In this work, we first define *redundancy categories*, and then classify each Skeleton point according to these categories. Points in different categories usually have different characteristics; for instance, *Intralevel* redundant points are often responsible for connectivity preservation, whereas *Interlevel* redundant points are those which usually compose the undesired long Skeleton branches, which are formed when the boundary of the shape is perturbated. Therefore, removing Interlevel redundancy, while keeping the Intralevel one, often leads to a more robust Skeleton, without affecting its connectivity.

A generic approach for obtaining *Redundancy-Reduced Skeletons* is also proposed and studied. It leads to closed-form morphological formulæ for removing redundant points from most of the proposed categories. The Intralevel redundant points can not be removed by this method.

A second approach for redundancy removal is proposed, which usually removes *most* of the Intralevel redundancy, by means of a morphological algorithm. The approach is based on a new generalization of the concept of "Convexity", called B-Convexity, also developed and studied in this work. A shape is called B-Convex when it is invariant under a morphological closing using a structuring element B. We show that the morphological closing can be seen as a generalization of the traditional *Convex-Hull* operation, and that many of properties of the latter are shared by the former.

Moreover, we show that the *Essential Points* of the Skeleton (which are proved here to be the intersection of all the Minimal Skeletons) can be obtained by means of closed-form morphological formulæ. These formulæ are obtained from the above approaches for redundancy reduction.

Coding of the Skeleton Representation: In the last years, many researchers lost their interest in Skeleton-based coding of binary images, because simpler coding schemes, like Chain-Coding of the original binary image, showed similar (or even better) compression results. Furthermore, no coding scheme using a Grayscale-Skeleton is proposed at all (as far as we know). On the other hand, the Skeleton coding schemes, used in the above simulations for binary images, have not taken into consideration the strong correlation existing between Skeleton subsets. Actually, this correlation was not properly characterized, and it was not clear how to efficiently use it.

In this work, new theoretically-based properties, characterizing the above correlation, are found, and a coding scheme, taking them into consideration, is proposed, for both binary and grayscale images. The proposed coding scheme showed, in simulations, substantial improvement in the Skeleton-based compression efficiency, as compared to

previous Skeleton-based coding schemes, and better results than those presented by the classical coding methods, including Chain-Coding.

The proposed coding-scheme is found to have similar performance as the Group 3 standard coder for facsimile, but is less efficient than the Group 4 and JBIG standard algorithms. It seems to us that much can still be done regarding Skeleton-based coding, so that it is not clear yet if it can or cannot achieve (or even improve) the performance of the above mentioned recent standards.

10.2 Future Research

Non-linear models are usually hard to handle, from a mathematical point of view. This often makes their analysis difficult, and does not permit their study, as profoundly as the study of linear models.

Mathematical Morphology, although being a non-linear theory, is based on a solid algebraic structure, and this gives us several mathematical tools for analysis, which permit its study as profoundly as linear theories do.

The above reality, together with the fact that Mathematical Morphology is relatively new, indicates that there is still much to be discovered in that field, not only in applications, but in theory as well. The more one reveals Mathematical Morphology's "secrets", the more he finds that there is still a lot to be revealed.

In the sequel, possible directions for future research are suggested, as a continuation of this work.

10.2.1 Hybrid Representations

In section 9.3, some hybrid approaches, combining morphological with linear methods for image representation, were considered. Further analysis and possible generalization of those approaches are suggested.

The zero-tree coding of Wavelets decomposition, considered here as a "linear-followed-by-a-morphological" approach, could have the zero-tree part of the scheme formulated in terms of Mathematical Morphology. Its morphological analysis should be then performed. An adaptation of the zero-tree coding could be tried, in order to make it a particular case of the Generalized Skeleton Representation. Such a representation is expected to be a "bi-parameter" one, since it should incorporate both the bit-plane decomposition and an adapted tree decomposition in one structure. This should be possible, since each of them can be considered as a particular case of a "one-parameter" Skeleton. Such adaptation could give better results in Coding.

The "morphological-followed-by-a-linear" approach proposed in section 9.3 should be formally defined, and analyzed. First, the connectivity criterion should be determined. Then, an appropriate set of functions, describing the connectivity graph, should be sought and analyzed. The effect of lossy linear coding of these functions on the reconstructed image should also be studied, and simulations of coding should be performed.

Probably the most promising among the hybrid methods is the Segmentation-based one. As mentioned at the end of chapter 9, much has to be done yet in this area. First, an adaptation of the proposed grayscale Skeleton-based coding algorithm (chapter 8) from lossless to *lossy* should be performed, and theoretically analyzed. Furthermore, coding-oriented (possibly multi-resolution) segmentation of images should be implemented. Moreover, linear coding schemes which code the contents of each segment separately should be investigated. These are just some of the topics which have to be studied in this area.

10.2.2 Sub-Family Optimal Determination

A more theoretical, and of much interest, research direction is the determination of a sub-family of elements, from a given family, for optimal skeletonization, according to some criterion. More specifically, given a family of elements (for example, the family of rectangles with sides n and m, both non-negative integers), what is the specific sub-family, for which a Skeleton representation would give the best coding results, or would most efficiently permit image analysis? This problem can be seen as the transposition, to Morphology, of the "Best-Basis Selection" [7] and/or the "Matching Pursuit" [28] problems for Wavelet Representations.

Suppose we consider the "bi-parameter" family of rectangles, mentioned above. The skeletonization of any binary set X by this family provides a lossless representation of X. If we take *any* sub-family of rectangles, containing the rectangle of sides 0 and 0 (a single point), then the skeletonization by this sub-family is also error-free.

If we search, for instance, for the 10 rectangles, which best represent X (in terms of area covering), then we would be trying to solve a "Matching Pursuit" problem.

A "Best-Basis Selection" kind of algorithm would be as follows: for each index (n, m), during the skeletonization process, decide among proceeding to the index $(n+1, m)$, the index $(n, m+1)$, or to both. This procedure selects a connected subset of the set of indices, determining a sub-optimal sub-family of rectangles. The question to be answered: "Under which conditions is this sub-family the optimal one?"

10.2.3 Generalization and Analysis of the Two-Sided Skeleton

The Two-Sided Skeleton, presented in Appendix D, opens up a new direction in Morphological Representation of Images. Its properties, its redundancy, and its efficiency should be further studied.

First of all, its generalization, in the same sense as performed for the original Skeleton in chapter 3, should be developed, and the scope and the conditions under which it is a perfect representation should be determined. Its redundancy should also be studied. And, finally, theoretical properties similar to those presented in chapter 8 should be sought. Upon discovery, an algorithm similar to the one proposed here for Skeleton-based Coding could be developed and tested.

Appendix A

Proofs

A.1 Theorems in Chapter 3

Proof of Theorem 1, page 46 - *A maximal element from a family* $\{\delta_i(x), i \in I, x \in E\}$ *inside a set* X *is an element* $\delta_i(x)$ *contained in* X*, such that, for any* $j \neq i$ *and* $y \in E$*, if* $\delta_j(y) \subseteq X$*, then* $\delta_i(x) \not\subseteq \delta_j(y)$*. But since condition 2 of the theorem makes it impossible for* $\delta_j(y)$ *to contain* $\delta_i(x)$ *if* $j \not> i$ *(unless* $i = j$ *and* $x = y$*), we need to check only for* $j > i$*. The skeleton subset* $S_i(X)$ *is the set of points* x*, such that* $\delta_i(x)$ *is maximal in* X*.*

The following is always true:

$$\delta_i(x) \subseteq X \Leftrightarrow x \in \varepsilon_i(X) \tag{A.1}$$

$$\delta_i(x) \not\subseteq \delta_j(y) \Leftrightarrow x \notin \varepsilon_i\delta_j(y) = \delta_{[i,j]}(y) \tag{A.2}$$

It is always true also that $\delta_j(y) \subseteq X$ *implies* $\varepsilon_i\delta_j(y) \subseteq \varepsilon_i(X)$*, but the equivalence is usually not assured. However, since the family of structuring-functions is granulometry-generating (condition 1 of the theorem), for* $j > i$ *the equivalence is obtained, which can be written in the following way:*

$$\delta_j(y) \subseteq X \Leftrightarrow \delta_{[i,j]}(y) \subseteq \varepsilon_i(X) \Leftrightarrow y \in \varepsilon_{[i,j]}\varepsilon_i(X) \tag{A.3}$$

Therefore, according to (A.1), (A.2) and (A.3), $x \in S_i(X)$ *iff* $x \in \varepsilon_i(X)$ *and* $x \notin \gamma_{[i,j]}\varepsilon_i(X)$*,* $\forall j > i$*. This leads to (3.9).* □

Proof of Theorem 2, page 47 - *The left side of (3.10) can be written in the following way:*

$$\bigcup_{i \in J} \delta_i(S_i(X)) = \bigcup_{i \in J} \bigcup_{x \in S_i(X)} \delta_i(x) \tag{A.4}$$

which means that it is equal to the union of all the maximal elements contained in X*, with "radius" in* J*. Therefore, we need to proof that* $x \in \gamma_J(X)$ *iff* x *belongs to some maximal element with "radius" in* J*.*

If $x \in X$ *belongs to a maximal element* $\delta_j(y)$*,* $j \in J$*, then* $x \in \gamma_j(X) \subseteq \gamma_J(X)$*, which proves one way.*

If $x \in \gamma_J(X)$ *then there is* $j \in J$ *such that* $x \in \delta_j(y) \in L_X$ *(for some* $j \in E$*), which is not necessarily a maximal element. But, since* L_X *is inductive for inclusion (condition of*

the theorem), every increasing chain in L_X converges to a unique element, and therefore, by Zorn's Lemma, $\delta_j(y)$ (and hence x too) is contained in a maximal element. The radius of this maximal element is obviously greater or equal to j and belongs to I. Thus, since J is an anti-umbra in I (condition of the theorem), the radius of the maximal element containing x belongs to J, which proves the way back. □

The above proofs are an extension of those in [53].

A.2 Relations in Chapter 6

Proof of relation (6.26), page 80 -

- *The region represented by Skeleton points of order i is:*

$$\delta_i(S_i) \tag{A.5}$$

- *The region represented by all the Skeleton points of order i,* except *a Skeleton point $s \in S_i$ is:*

$$\delta_i(S_i - \{s\}) \tag{A.6}$$

- *The region represented* only by the point *s is the difference of the above sets:*

$$\delta_i(S_i) - \delta_i(S_i - \{s\}) \tag{A.7}$$

- *The union of the above sets, for all the Skeleton points $s \in S_i$ is:*

$$\bigcup_{s \in S_i} [\delta_i(S_i) - \delta_i(S_i - \{s\})] \stackrel{\triangle}{=} Y. \tag{A.8}$$

  It gives the union of those regions which are each represented by only one point of order i.

- *$\delta_i(S_i) - Y$ is the region represented by more than one point of order i, i.e.,*

$$\delta_i(S_i) - Y = R_i. \tag{A.9}$$

- *Y may also be written as:*

$$Y = \delta_i(S_i) - \bigcap_{s \in S_i} \delta_i(S_i - \{s\}) \tag{A.10}$$

- *relation (6.26) is then obtained considering the last two items, and the fact that Y is contained in $\delta(S_i)$.*

□

Proof of relation (6.27), page 80 - *It is enough to prove that for any sets A and B:*

$$\bigcap_{a\in A}(A-\{a\})\oplus B=\bigcap_{b\in B}A\oplus(B-\{b\}) \tag{A.11}$$

First, let us denote the left hand of the above equation as $\mathcal{H}$, and then write the dilation explicitly in the following way:

$$\mathcal{H}=\bigcap_{a\in A}\bigcup_{b\in B}\bigcup_{\tilde{a}\in A,\tilde{a}\neq a}\{\tilde{a}+b\} \tag{A.12}$$

Then, after some logical and set manipulations, we notice that a point $z=\tilde{a}+b$ belongs to $\mathcal{H}$ iff there is another pair of points a and $\tilde{b}$, in A and B respectively, such that $a+\tilde{b}=z$. In other words:

$$\mathcal{H}=\{z=\tilde{a}+b=a+\tilde{b}\in A\oplus B\mid\ a\neq\tilde{a},\tilde{b}\neq b\} \tag{A.13}$$

Since equation (A.13) is symmetric, i.e., the roles of $A,a,\tilde{a}$ and $B,\tilde{b},b$ are respectively interchangeable, then we can interchange the above sets and elements also in the original expression, which provides (A.11). □

A.3 Proposition in Chapter 7

Proof of the Proposition 3, page 90 - *Like we did in the proof of Proposition 2 (page 89), we can rewrite the definition of Global Essential Points in terms of $X\ominus nB$: $t\in S_n$ is a Global Essential Point iff:*

$$\{\tilde{P}_n\cup[(X\ominus nB)-\{t\}]\oplus nB\}\ominus nB\neq X\ominus nB \tag{A.14}$$

On the other hand, the left side of equation (A.14) can assume only two values: $X\ominus nB$ or $X\ominus nB-\{t\}$, for any $t\in X\ominus nB$. This is because, for any $t\in X\ominus nB$,

1.

$$\begin{gathered}\{\tilde{P}_n\cup[(X\ominus nB)-\{t\}]\oplus nB\}\ominus nB\supseteq\\(\tilde{P}_n\ominus nB)\cup[(X\ominus nB)-\{t\}]\bullet nB=\\[(X\ominus nB)-\{t\}]\bullet nB\supseteq\\(X\ominus nB)-\{t\}\end{gathered} \tag{A.15}$$

2.

$$\begin{gathered}\{\tilde{P}_n\cup[(X\ominus nB)-\{t\}]\oplus nB\}\ominus nB\subseteq\\\{\tilde{P}_n\cup(X\ominus nB)\oplus nB\}\ominus nB=\\\{\tilde{P}_n\cup X\circ nB\}\ominus nB=\\(\textstyle\bigcup_m S_m\oplus mB)\ominus nB=\\X\ominus nB\end{gathered} \tag{A.16}$$

Therefore, $t \in S_n \subseteq X \ominus nB$ is Global Essential iff

$$\{\tilde{P}_n \cup [(X \ominus nB) - \{t\}] \oplus nB\} \ominus nB = (X \ominus nB) - \{t\} \tag{A.17}$$

We note that there are 4 types of points in $X \ominus nB$:

1. *The points that satisfy (A.17) and belong to S_n - which are the Global Essential Points, $GE_n(X)$.*
2. *The points that* **do not** *satisfy (A.17) and belong to S_n - which are called the* Global Redundant Points.
3. *The points that satisfy (A.17) and* **do not** *belong to S_n - which we denote Y_n.*
4. *The points that* **do not** *satisfy (A.17) and* **do not** *belong to S_n.*

The points that do not satisfy (A.17) satisfy:

$$\{\tilde{P}_n \cup [(X \ominus nB) - \{t\}] \oplus nB\} \ominus nB = X \ominus nB \tag{A.18}$$

If we intersect the left side of equation (A.17) for every t in $X \ominus nB$, we obtain:

$$\bigcap_{t \in X \ominus nB} \{\tilde{P}_n \cup [(X \ominus nB) - \{t\}] \oplus nB\} \ominus nB = (X \ominus nB) - [GE_n(X) \cup Y_n]$$
$$\Downarrow$$
$$\Big\{\tilde{P}_n \cup \bigcap_{t \in X \ominus nB} \{[(X \ominus nB) - \{t\}] \oplus nB\}\Big\} \ominus nB = $$
$$= (X \ominus nB) - [GE_n(X) \cup Y_n] \tag{A.19}$$

And since

$$\bigcap_{t \in X \ominus nB} \{[(X \ominus nB) - \{t\}] \oplus nB\} = \mathcal{I}^{nB}(X \ominus nB) \oplus nB \tag{A.20}$$

we get:

$$\Big\{\tilde{P}_n \cup \mathcal{I}^{nB}(X \ominus nB) \oplus nB\Big\} \ominus nB = (X \ominus nB) - [GE_n(X) \cup Y_n] \tag{A.21}$$

By set-subtracting both sides of (A.21) from S_n, we obtain the proof. □

A.4 Lemma in Chapter 8

Proof of Lemma 1, page 97 - *Let us define:*

$$R_n \stackrel{\triangle}{=} Rec\{S \cup Y_{n+1}, Y_{n+1}\}_{C(n)} \tag{A.22}$$
$$= Rec\{S \cup S_n \cup Y_{n+1}, Y_{n+1}\}_{C(n)} \tag{A.23}$$
$$= Rec\{S \cup X_n, Y_{n+1}\}_{C(n)}. \tag{A.24}$$

We note that:

$$Rec\{S \cup X_n, Y_{n+1}\}_{C(n)} \supseteq \tag{A.25}$$
$$\supseteq Rec\{X_n, Y_{n+1}\}_{C(n)} = \tag{A.26}$$
$$= X_n - U_n. \tag{A.27}$$

Therefore:

$$R_n \cup U_n \supseteq X_n. \tag{A.28}$$

On the other hand:

$$Rec\{S \cup X_n, Y_{n+1}\}_{C(n)} \subseteq \tag{A.29}$$
$$\subseteq Rec\{S \cup X_n, X_n\}_{C(n)}, \tag{A.30}$$

And we shall show that:

$$Rec\{S \cup X_n, X_n\}_{C(n)} = X_n, \tag{A.31}$$

which gives, together with (A.30):

$$R_n \subseteq X_n. \tag{A.32}$$

Let us prove (A.31).

For $n = 0$, (A.31) holds trivially, since $X_0 = X$ (the original images), and $S \subseteq X$.

Suppose, therefore, $n > 0$. It holds:

$$\begin{cases} X_n \oplus C_n = X_n \oplus B_{n-1} \\ [X_n \oplus B_{n-1}] \cap S_{n-1} = \emptyset \end{cases} \tag{A.33}$$

Which leads to:

$$[X_n \oplus C(n)] \cap S_{n-1} = \emptyset. \tag{A.34}$$

Similarly,

$$[X_n \oplus C(n)] \cap S_m = \emptyset, \quad \forall m < n. \tag{A.35}$$

Therefore, since $S_m \subseteq X_n$ for all $m \geq n$, we get:

$$[X_n \oplus C(n)] \cap [X_n \cup S] = X_n. \tag{A.36}$$

The conclusion is the validity of equation (A.31).

Now, from (A.28) and (A.32), we get $R_n \cup U_n = X_n$, which proves the Lemma. □

Appendix B

B-Convexity Theory

B.1 Convex Sets and Some of its Properties.

One way to define a Convex Set in $\mathcal{R}^2$ is the following:

Definition 10 *Let $X \subseteq \mathcal{R}^2$. X is Convex iff it is identical to its Convex-Hull, i.e., $X = CH(X)$.*

The above definition demands another definition, for the operation Convex-Hull, which should be not based on the concept of Convexity. Such a definition is:

Definition 11 *The Convex-Hull of a set X, $CH(X)$, is the intersection of all the half-planes that contain X.*

The two main **properties of the Convex-Hull** operation are:

1. $CH(\cdot)$ is idempotent.
2. For any set X, $CH(X)$ if the "smallest" convex set that contains X.

Some of the **basic topological properties of the Convex Sets** are:

1. X is convex iff any two points x and y belonging to X are connected by a segment contained in X. We may also formulate this property in the following equivalent way: X is convex iff for any to points x and y belonging to X, the Convex-Hull of $\{x, y\}$ is contained in X.
2. The intersection of convex sets is a convex set.
3. X is convex iff every point outside X can be separated from X by a half-plane, i.e., $x \notin X \Rightarrow$ exists a half-plane that contains x and does not intersect with X.

B.2 B-Convex Sets and its Properties.

The proposed generalization of Convexity is obtained by replacing the Convex-Hull operation in Definition 10 by a generalized Convex-Hull operation, called *B-Convex-Hull*. The definition of the B-Convex-Hull, in its turn, is similar to the definition of the conventional Convex-Hull (Definition 11), having only the half-planes of Definition 11 replaced by an arbitrary shape B.

Definition 12 *The B-Convex-Hull of a set X, $CH^B(X)$, is the intersection of all the translations of $(B^s)^c$ that contain X.*

In the above definition, B^c is the complement of B, i.e., $B^c = \{b \in \mathcal{R}^2 \mid b \notin B\}$, and B^s is the symmetric of B, i.e., $B^s = \{-b \mid b \in B\}$.

Actually, the B-Convex-Hull, as defined above, is not a new operation; it is known in Mathematical Morphology as the Morphological **Closing**. In other words:

Proposition 4

$$CH^B(X) = X \bullet B \tag{B.1}$$

where $\bullet$ denotes binary morphological Closing.

Proof

$$\begin{aligned}
CH^B(X) &= \bigcap_{\{z \mid [(B^s)^c]_z \supseteq X\}} [(B^s)^c]_z \\
&= \bigcap_{\{z \mid \{z\} \oplus B^s \subseteq X^c\}} [(B^s)^c]_z \\
&= \bigcap_{z \in X^c \ominus B^s} [(B^s)^c]_z \\
&= \left[\bigcup_{z \in X^c \ominus B^s} (B^s)_z \right]^c \\
&= [(X^c \ominus B^s) \oplus B^s]^c \\
&= (X^c \circ B^s)^c \\
&= X \bullet B
\end{aligned} \tag{B.2}$$

□

We denote the generalized Convex sets as B-Convexity and define them as follows:

Definition 13 *A set X is B-Convex iff $X = X \bullet B$.*

B.2.1 Properties of the B-Convex-Hull

The B-Convex-Hull has the following two basic properties:

1. It is idempotent.

2. $CH^B(X)$ is the "smallest" B-Convex set that contains X.

Note the analogy with the properties of Convex-Hull described above. **Proof**

1. *Idempotency is a property of the morphological Closing, hence it is also of the B-Convex-Hull.*

2. *(a)* $CH^B(X)$ *contains* X*, since this is a property of the morphological Closing.*

   (b) $CH^B(X)$ *is a B-Convex set, since, according to the idempotency,* $CH^B[CH^B(X)] = CH^B(X)$.

   (c) Let Y *be a B-Convex set, containing* X*, such that* $X \subseteq Y \subseteq CH^B(X)$*. Since Closing is an* increasing *operation, i.e.,* $A \subseteq B \Rightarrow A \bullet C \subseteq B \bullet C$*, for all* A*,* B *and* C*, we obtain* $X \bullet B \subseteq Y \bullet B \subseteq CH^B(X) \bullet B = (X \bullet B) \bullet B = X \bullet B$*, and therefore,* $CH^B(X) \subseteq Y \subseteq CH^B(X)$*, which means that* $CH^B(X)$ *and* Y *are identical.*

   From 2a, 2b and 2c, we get the proof.

□

B.2.2 Properties of the B-Convex Sets

The topological properties of B-Convex sets, analogous to those of the Convex sets, are:

1. If X is B-Convex then for any pair of points x and y in X, $\{x, y\} \bullet B$ is contained in X.

2. The intersection of B-Convex sets is a B-Convex set.

3. X is B-Convex iff every point outside X can be separated from X by a translation of B^s, i.e., $x \notin X \Rightarrow$ exists $z \in \mathcal{R}^2$ such that $(B^s)_z$ contains x and does not intersect with X.

Proof

1. *It follows from the increasing property of the Closing operation:*

$$\{x, y\} \subseteq X \Rightarrow \{x, y\} \bullet B \subseteq X \bullet B = X \tag{B.3}$$

2. *Let* $\{X_i\}$ *be a set of B-Convex sets. Each* X_i *can be written as follows:*

$$X_i = \bigcap_{z \in Z_i} [(B^s)^c]_z \tag{B.4}$$

   for some Z_i*. Therefore,*

$$\bigcap_i X_i = \bigcap_{z \in \bigcup_i Z_i} [(B^s)^c]_z \tag{B.5}$$

   and hence, it is also a B-Convex set.

3.

$$
\begin{aligned}
X = CH^B(X) = \\
&= \bigcap_{\{z \mid \ [(B^s)^c]_z \supseteq X\}} [(B^s)^c]_z \\
&= \bigcap_{\{z \mid \ (B^s)_z \cap X = \emptyset\}} [(B^s)^c]_z \\
&= \left[\bigcup_{\{z \mid \ (B^s)_z \cap X = \emptyset\}} (B^s)_z \right]^c
\end{aligned} \tag{B.6}
$$

Therefore,

$$
X^c = \bigcup_{\{z \mid \ (B^s)_z \cap X = \emptyset\}} (B^s)_z \tag{B.7}
$$

This means that, $x \notin X$ if and only if x belongs to some translation of B^s such that this translation does not intersect with X.

□

B.3 Extreme Points

The Extreme Points of Convex Sets are points with many special properties. Among them, there is the ability to fully represent the original set, if it is bounded. In this work we also extend the definition of Extreme Points, for B-Convex sets as well.

B.3.1 Extreme Points of Convex Sets

One way of defining Extreme Points of Convex sets is:

Definition 14 *A point t is an Extreme Point of a Convex set X iff the set $(X - \{t\})$ is also convex.*

The ability of the Extreme Points to fully represent the original shape is expressed in the following proposition:

Proposition 5 *Let $\mathcal{E}(X)$ be the set of the Extreme Points of a convex set X. If X is bounded, then $CH(\mathcal{E}(X)) = X$.*

B.3.2 Extreme and Internal Points of B-Convex sets

We now present our generalization of the above concepts, defining Extreme Points of B-Convex set in the following way:

Definition 15 *A point t is an Extreme Point of a B-Convex set X, iff the set $(X - \{t\})$ is also B-Convex.*

In this work, $\mathcal{E}^B(X)$ will denote the set of Extreme Points of a B-Convex set X.

We also define *Internal Points* of B-Convex sets. They are all the points in the set that are not Extreme. Denoting the set of Internal Points of a B-Convex set X, by $\mathcal{I}^B(X)$, we may write $\mathcal{I}^B(X) = X - \mathcal{E}^B(X)$. And, since $\mathcal{E}^B(X) \subseteq X$, it is also true that

$$\mathcal{E}^B(X) = X - \mathcal{I}^B(X). \tag{B.8}$$

The following equation provides a morphological formula for calculating the set of Internal Points of a given B-Convex set:

$$\mathcal{I}^B(X) = \bigcap_{x \in X} [X - \{x\}] \bullet B \tag{B.9}$$

Proof *According to the increasing property of the Closing,*

$$[X - \{x\}] \subseteq X \Rightarrow [X - \{x\}] \bullet B \subseteq X \bullet B = X, \tag{B.10}$$

which means that $[X-\{x\}]\bullet B$ *is never bigger than* X*. But it is never smaller than* $[X-\{x\}]$*, either. Therefore, it can be equal only to either one of these two possibilities. According to the definition,* $[X - \{x\}] \bullet B = [X - \{x\}]$ *iff* x *is an Extreme Point of* X*. Hence, we obtain that if (and only if)* x *is an Internal Point, then* $[X - \{x\}] \bullet B = X$*.*

Therefore,

$$\begin{aligned}
\bigcap_{x \in X} [X - \{x\}] \bullet B &= X \cap \left\{ \bigcap_{x \in \mathcal{E}^B(X)} [X - \{x\}] \bullet B \right\} \\
&= \bigcap_{x \in \mathcal{E}^B(X)} [X - \{x\}] \\
&= X - \bigcup_{x \in \mathcal{E}^B(X)} x \\
&= X - \mathcal{E}^B(X) = \mathcal{I}^B(X)
\end{aligned} \tag{B.11}$$

□

The set of Extreme Points is obtained using (B.8) and (B.9).

For a perfect reconstruction from the Extreme Points, the set B must be "big enough" in comparison to the set X. This situation is analogous to the requirement of X being bounded, in the last section, since in this case any half-plane (which is not bounded) is "big enough" in comparison to X. A *necessary* condition for the perfect reconstruction is stated in the following proposition:

Proposition 6 *If* $CH^B[\mathcal{E}^B(X)] = X$ *(perfect reconstruction from the Extreme Points), then* $B \ominus \mathcal{I}^B(X) = \emptyset$*.*

Proof *For all* A*,* B *and* C*, the following hold:* $(A - B) \oplus C \subseteq (A \oplus C) - (B \ominus C)$*, and* $(A - B) \ominus C = (A \ominus C) - (B \oplus C)$*. Together we get:* $(A - B) \bullet C \subseteq (A \bullet C) - (B \circ C)$*. Since* $\mathcal{E}^B(X) = X - \mathcal{I}^B(X)$*, we have:*

$$\mathcal{E}^B(X) \bullet B \subseteq X - \mathcal{I}^B(X) \circ B \tag{B.12}$$

By hypothesis, the left member of equation (B.12) is equal to X*. Therefore,* $\mathcal{I}^B(X) \circ B$ *must be empty and this happens iff* $\mathcal{I}^B(X) \ominus B$ *is also empty.* □

B.3.3 Properties of the Extreme and the Internal Points

Proposition 7 *If X is a B-Convex set, then $\mathcal{I}^B(X)$ is also a B-Convex set.*

Proof *In the proof of the formula (B.9), we showed that*

$$\mathcal{I}^B(X) = \bigcap_{x \in \mathcal{E}^B(X)} [X - \{x\}] \tag{B.13}$$

where each set $[X - \{x\}]$ is a B-Convex set, since x is an Extreme Point. Since intersection of B-Convex sets is also a B-Convex set, $\mathcal{I}^B(X)$ is B-Convex. □

Proposition 8 *A point p is an Extreme Point of a B-Convex set X, iff there exists z such that*

$$(B^s)_z \cap X = \{p\}. \tag{B.14}$$

Proof *The point p is extreme iff $[X - \{p\}]$ is B-Convex (definition of Extreme Point). $[X - \{p\}]$ is B-Convex iff every point outside X can be separated from X by a translation of B^s (proposition 3). Therefore, p can be separated from $[X - \{p\}]$, i.e., there exists z such that $p \in (B^s)_z$ and $(B^s)_z \cap [X - \{p\}] = \emptyset$. This equivalent to (B.14). All the other points (other than p) can also be separated from $[X - \{p\}]$ because, since X itself is B-Convex, they all can be separated from X.* □

The next property agrees with one's intuition that the Extreme Points are located at the boundary of the shape. Before we state it, let us define *Isolated Point* and *Boundary*:

Definition 16 *A point $x \in X$ is an Isolated Point of X, according to a shape B_0* containing the origin, *iff $(\{x\} \oplus B_0) \cap X = \{x\}$.*

The shape B_0 is called neighborhood.

Definition 17 *The boundary of a shape X, according to a neighborhood B_0, is the set $\partial^{B_0} X$ defined by $\partial^{B_0} X = X - X \ominus B_0$.*

Now we can state the property:

Proposition 9 *If B^s has no Isolated Points according to the neighborhood B_0, then the Extreme Points of* any *B-Convex set X belong to the boundary of X, according to B_0, i.e., $\mathcal{E}^B(X) \subseteq \partial^{B_0} X$.*

Proof *Let p be an Extreme Point of X. Then, according to proposition 8, there exists a point z for which relation (B.14) holds.*

Let's now assume that p belongs to $X \ominus B_0$. In that case, obviously,

$$\{p\} \oplus B_0 \subseteq X. \tag{B.15}$$

Intersecting both sides of (B.15) with $(B^s)_z$, using (B.14), and taking into consideration that p belongs to both B_0 and $(B^s)_z$, we obtain:

$$(\{p\} \oplus B_0) \cap (B^s)_z = \{p\} \tag{B.16}$$

Which means that p is an Isolated Point of $(B^s)_z$. *But since* B^s *has no Isolated Points, none of its translations has, either, and we have a contradiction. Therefore, p cannot belong to* $X \ominus B_0$.

The conclusion is that, if p is an Extreme Point, than it does not belong to $X \ominus B_0$. *In other words,* $\mathcal{E}^B(X) \subseteq (X \ominus B_0)^c$. *But since* $\mathcal{E}^B(X) \subseteq X$, *we obtain that* $\mathcal{E}^B(X) \subseteq X - X \ominus B_0$. □

Proposition 10 *A B-Convex set X can be perfectly reconstructed from its Extreme Points by the formula* $CH^B[\mathcal{E}^B(X)] = X$ *if and only if no Internal Point of X can be separated from* $\mathcal{E}^B(X)$ *by a translation of* B^s, *i.e.,* $\forall r \in \mathcal{I}^B(X)$, $\forall z$ *such that* $r \in (B^s)_z$, $(B^s)_z \cap \mathcal{E}^B(X) \neq \emptyset$.

Proof

$$\begin{aligned} CH^B[\mathcal{E}^B(X)] = X \Leftrightarrow \\ [\mathcal{E}^B(X)] \bullet B = X \Leftrightarrow \\ [\mathcal{E}^B(X)] \oplus B = X \oplus B \end{aligned} \tag{B.17}$$

Therefore every translation of B^s *that hits X (or any of its subsets, like* $\mathcal{I}^B(X)$*) hits also* $\mathcal{E}^B(X)$. □

Appendix C

Generalizations of the Coding Theorems

C.1 Coding Theorems Adapted to a Generalized Skeleton Representation

This part of the appendix transposes the theoretical results, which are presented in chapter 8 for the Generalized-Step Skeleton, to a form suitable for the Generalized Skeletons proposed in chapter 3.

Not all the theorems in chapter 8 can be directly adapted. For instance, the version of Theorem 4 presented here (Theorem 6) requires the set of indices I to be *totally* ordered, among other requirements.

C.1.1 Basic Definitions and Notation

Let $X \in \mathcal{P}(E)$ and the family of structuring functions $\{\delta_i(x)\}_{i \in I}$ (I being any partial ordered set) satisfy the conditions required for a Generalized Skeleton decomposition (column 3 in Table 3.5).

We adopt here the following notation:

$$X_i \stackrel{\triangle}{=} \varepsilon_i(X) \tag{C.1}$$

$$Y_i \stackrel{\triangle}{=} \bigcup_{j>i} \gamma_{[i,j]}\varepsilon_i(X) \tag{C.2}$$

Therefore,

$$S_i = X_i - Y_i \tag{C.3}$$

Definition 18 (Descendance) *Let $\delta(x)$ be a structuring function in a Boolean Lattice $\mathcal{P}(E)$. A point $y \in E$ is a direct* descendant *of a point $x_0 \in E$, under the give structuring function, iff:*

$$y \in \delta(x_0) \tag{C.4}$$

Definition 19 (Connectivity) *Two points x_0 and y are connected (under a pre-defined structuring function $\delta(x)$) iff each one is a direct descendant of the other, under $\delta(x)$, i.e.:*

$$x_0 \in \delta(y) \text{ and } y \in \delta(x_0) \tag{C.5}$$

Definition 20 (Reconstruction) *Let A, D be two sets in a Boolean Lattice, such that $D \subseteq A$, and $\delta(x)$ be a pre-defined structuring function. The* Reconstruction *of A from D using $\delta(x)$, $Rec\{A, D\}_\delta$ is given by the following recursive formula:*

$$Rec\{A, D\}_\delta \triangleq \{\delta[\delta(\delta(D) \cap A) \cap A] \cap A \ldots \tag{C.6}$$

Definition 21 (Ultimate Erosions) *If $i \in I$ is not a first point, i.e., $\exists j \in I \mid\ j < i$, then the* Ultimate Erosion *of order i, denoted as U_i, of the given set X, is defined by:*

$$U_i \triangleq X_i - \bigcap_{j<i} Rec\left\{X_i, Y_i\right\}_{\delta_{[j,i]}} \tag{C.7}$$

where $\delta_{[j,i]}(x) \triangleq \varepsilon_i \delta_j(x)$, $\forall i, j \in I, j < i, \forall x \in E$.

If $i \in I$ is a first point, then U_i is defined by:

$$U_i \triangleq X_i - Rec\left\{X_i, Y_i\right\}_{\delta^{(i)}} \tag{C.8}$$

where $\{\delta^{(i)}(x)\}$ is any pre-defined set of structuring functions.

C.1.2 Skeleton Properties

Quench Function Sampling

Lemma 2 *Let I be a* discrete *set; $\{S_i\}$ be the Skeleton subsets of a Skeleton satisfying the conditions in Table 3.5; and let X_i and Y_i be as defined in (C.1) and (C.2), respectively.*

Then, it holds:

$$\bigcap_{j<i} Rec\left\{Y_i \cup \left[\bigcup_{j' \le i} S_{j'}\right], Y_i\right\}_{\delta_{[j,i]}} \cup U_i = X_i \tag{C.9}$$

Proof *The proof is a generalization of that of Lemma 1.*

Let us define:

$$R_i \triangleq \bigcap_{j<i} Rec\left\{Y_i \cup \left[\bigcup_{j' \le i} S_{j'}\right], Y_i\right\}_{\delta_{[j,i]}} \tag{C.10}$$

$$= \bigcap_{j<i} Rec\left\{Y_i \cup S_i \cup \left[\bigcup_{j' \le i} S_{j'}\right], Y_i\right\}_{\delta_{[j,i]}} \tag{C.11}$$

$$= \bigcap_{j<i} Rec\left\{X_i \cup \left[\bigcup_{j' \le i} S_{j'}\right], Y_i\right\}_{\delta_{[j,i]}} \tag{C.12}$$

We note that:

$$\bigcap_{j<i} Rec\left\{X_i \cup \left[\bigcup_{j' \le i} S_{j'}\right], Y_i\right\}_{\delta_{[j,i]}} \supseteq \tag{C.13}$$

$$\bigcap_{j<i} Rec\left\{X_i, Y_i\right\}_{\delta_{[j,i]}} = \tag{C.14}$$

$$= X_i - U_i. \tag{C.15}$$

Therefore:

$$R_i \cup U_i \supseteq X_i. \tag{C.16}$$

On the other hand:

$$\bigcap_{j<i} Rec\left\{X_i \cup \left[\bigcup_{j'\leq i} S_{j'}\right], Y_i\right\}_{\delta_{[j,i]}} \subseteq \tag{C.17}$$

$$\subseteq \bigcap_{j<i} Rec\left\{X_i \cup \left[\bigcup_{j'\leq i} S_{j'}\right], X_i\right\}_{\delta_{[j,i]}}. \tag{C.18}$$

And we shall show that:

$$\bigcap_{j<i} Rec\left\{X_i \cup \left[\bigcup_{j'\leq i} S_{j'}\right], X_i\right\}_{\delta_{[j,i]}} = X_i. \tag{C.19}$$

which gives, together with (C.18):

$$R_i \subseteq X_i. \tag{C.20}$$

Let us prove (C.19).

It holds:

$$\bigcap_{j<i} \delta_{[j,i]}(X_i) \cap S_j = \emptyset \tag{C.21}$$

Therefore, for all $j < i$:

$$\delta_{[j,i]}(X_i) \cap \left[\bigcup_{j'<i} S_{j'} \cup X_i\right] = X_i \tag{C.22}$$

The conclusion is the validity of equation (C.19).

Now, from (C.16) and (C.20), we get $R_i \cup U_i = X_i$, which proves the Lemma. □

Theorem 6 *Consider Ultimate Erosions as defined in Definition 21, and the Skeleton as defined in Lemma 2. Moreover, we require that the decomposition family $\{\delta_i(x)\}$ be* increasing, *i.e., $i < j \Rightarrow \delta_i(x) \subseteq \delta_j(x), \forall x \in E$ (condition 7 for the Generalized Skeleton on Lattices, see column 2 of Table 3.5, on page 41).*

If I is totally ordered *and* discrete, *and the above conditions are satisfied, then the radius of the Skeleton points which are Ultimate Erosions, together with the position of all the Skeleton points, are sufficient for perfectly representing the original set X.*

Proof *Since I is totally ordered and discrete, it is isomorphic to $\mathcal{N}$, and therefore we will consider $I = \mathcal{N}$, with no lost of generality.*

We use induction in the following way: (i) If N is the maximal radius in the Skeleton, then $X_N = U_N$. (ii) Once X_{i+1} is known, each set X_i, $N > i \geq 0$, can be calculated (see below), and (iii) the original image X is equal to X_0.

In order to obtain the second part of the above induction, suppose that X_{i+1} is available. Therefore Y_i is also available, and since $I = \mathcal{N}$, it holds:

$$Y_i = \gamma_{[i,i+1]}\varepsilon_i(X) = \delta_{[i,i+1]}(X_{i+1}) \tag{C.23}$$

From the hypothesis, the Skeleton $S \triangleq \bigcup_{i\in I} S_i$ *and the Ultimate Erosions* $\{U_n\}$ *are provided. Since the Skeleton points with radius greater than* i *are available at this point, and since the Skeleton subsets are disjoint (decomposition family is increasing), the set of Skeleton points with radius smaller or equal to* i, $\bigcup_{j'\leq i} S_{j'}$, *is also available. Then* X_i *is obtained from the above by Lemma 2.* □

The following corollaries are direct consequences of (C.9).

Corollary 8 *For a Skeleton as defined in Theorem 6, if* s *is a Skeleton point with index* i*, then all the Skeleton points which descend from it, (under* $\delta_{[i-1,i]}(x)$*, if* $i > 0$*, or* $\delta^{(0)}(x)$*, otherwise), have also index* i*.*

Corollary 9 *Suppose that* $\delta_i(x)$ *is* symmetric*, i.e.,* $\forall x, y \in E$, $y \in \delta_i(x) \Leftrightarrow x \in \delta_j(x)$.

In this case, for a Skeleton as defined in Theorem 6, if s *is a Skeleton point with index* i*, then all the Skeleton points in the connected component to which it belongs (with connectivity being under* $\delta_{[i-1,i]}(x)$*, if* $i > 0$*, or* $\delta^{(0)}(x)$*, otherwise) have also index* i*.*

Corollary 10 *For a Skeleton as defined in Theorem 6, a Skeleton point* s *has index* i if and only if $s \notin Y_i$ *and* s *descends either from* U_i *or* Y_i*.*

Deterministic Prediction

As opposed to the results in the previous section, the results in this section relate to any Generalized Skeleton representation, i.e., I is *not* restricted either to be totally ordered, or to be discrete.

Theorem 7 *Let* $p \in E$*. If the following holds:*

$$\bigcup_{j>i} \gamma_{[i,j]}\left[\phi_i(Y_i \cup \{p\})\right] \supset \{p\} \tag{C.24}$$

then p *cannot belong to* S_i*.*

Proof *The proof is by contradiction. Suppose that* p *is in* S_i*, and let us define the following operator:*

$$\rho(Z) \triangleq \bigcup_{j>i} \gamma_{[i,j]}\varepsilon_i(Z). \tag{C.25}$$

By definition of Y_i, $\rho(X) = Y_i$*. Also the set* $\delta_i(Y_i)$ *gives* Y_i *when operated upon by* $\rho(\cdot)$*.*

Therefore, since $\rho(\cdot)$ *is increasing, any set* Z_0*, such that* $\delta_i(Y_i) \subseteq Z_0 \subseteq X$*, satisfies* $\rho(Z_0) = Y_i$*. In particular,* $Z_0 = \delta_i(Y_i \cup \{p\})$, $p \in S_i$*, satisfies it.*

However, according to (C.24) $\rho(Z_0) \supset \{p\}$*, and, therefore,* $p \in Y_i$*, which contradicts that* $p \in S_i$*.* □

Corollary 11 *Let* $\delta_F(x)$ *be a structuring function satisfying, for all* $x \in E$*:*

1. $x \notin \delta_F(x)$,
2. $$\bigcup_{j>i} \gamma_{[i,j]}\left[\phi_i[\{x\} \cup \delta_F(x)]\right] \supset \{x\} \tag{C.26}$$

and let $p \in E$*.*

If $\delta_F(p) \subseteq Y_i$*, then* p *cannot belong to* S_i*.*

C.2 Coding Algorithm Adapted to a Grayscale Skeleton Representation

In this part of the Appendix, we briefly propose a generalization for *grayscale functions* of the coding algorithm presented in Chapter 8 for binary images.

For the generalization, the *umbra approach* (Section 4.3, on page 52) is used. In this case, a Generalized-Step Grayscale Skeleton decomposition is considered as a Generalized-Step Skeleton on the Boolean Lattice $\mathcal{P}(E \times \mathcal{R})$, and all the definitions and propositions of Chapter 8 are valid.

The following is an outline of the algorithm for coding a grayscale image. It is based on the generalization in the umbra domain of the binary algorithm presented in Chapter 8, and it is transposed to the function domain (see section 4.5). The given image is denoted by f. We suppose here that the structuring element g is symmetric and flat. The function Z is an intermediate image, initially null, which is equal to f at the end of the process.

1. Calculate the Skeleton Subsets S_n, $0 \leq n \leq N$. Form the functions $\tilde{U}_n$ by taking one pixel from each local maximum of the Ultimate Erosions U_n.
2. $n \leftarrow N - 1$; $Y_N \leftarrow \emptyset$.
3. $Z \leftarrow (Y_n \vee \tilde{U}_n)$.
4. $p \leftarrow$ (a non-null point from the set $Z \oplus g - Z$). If there are no more points to scan, go to step 9.
5. Check (by means of the generalization of Theorem 5) if $S_n(p)$ can be different of 0 (i.e., if p "belongs" to S_n) or not. If it cannot, go to step 4.
6. Send to the Arithmetic Coder a "0", if p is "belongs" to S_n, or a "1", otherwise. Furthermore, if p belongs to S_n, send also to the Arithmetic Coder the difference between its gray value and the maximum gray value of Z in the neighborhood (defined by the region of support of g) of it. Use an adaptive probability model.
7. If a "1" was sent, $Z \leftarrow (Z \vee S_n(p))$. Otherwise, go to step 4.
8. Recursively, scan the direct descendants 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. The Skeleton points must have the difference between its gray value and the maximum of Z in its neighborhood (defined by the region of support of g) coded as well. After the whole connected component is scanned and coded, go to step 4.
9. If $n = 0$, STOP.
10. $n \leftarrow (n - 1)$; $Y_{n+1} \leftarrow Z \oplus B(n)$; Go to step 3.

Appendix D

Two-Sided Skeleton

Appendix D.

TWO-SIDED SKELETON
- A Representation Composed of Both Positive and Negative Morphological Elements
Reference [19]

ABSTRACT

This work presents a novel morphological representation structure - the Two-Sided Morphological Skeleton*. It represents a shape not only by the centers of* "positive elements" *(foreground features), as the ordinary Morphological Skeleton does, but also by the centers of* "negative elements" *(background features, such as* **holes***). I.e., it represents an image by elements from both sides of the Pattern Spectrum.*

The Two-Sided Morphological Skeleton can be a very efficient tool in areas such as Multi-Resolution Representation, Shape Analysis and Pattern Recognition, since negative elements are as much important to image comprehension as positive elements. It has also a potential in Coding, because it is a compact error-free representation of the original image.

In this work, the Two-Sided Morphological Skeleton is defined and studied for both binary and grayscale images.

I. INTRODUCTION

The Morphological Skeleton has been successfully applied in many Image Processing application areas for efficient shape representation, and as a feature extractor and classifier (according to size). However, it does not take directly into account background features such as "holes" and "negative shapes", making it less efficient in such cases. A simple example is shown in Fig. 1: Fig. 1(a) is a binary picture, where the bigger white circle is a "positive shape" and the smaller black circle is a "negative shape"; its ordinary Morphological Skeleton, calculated with a disk as structuring-element, is a circle between the two circular edges as shown in Fig. 1(b). There is a need for more efficient representations that consider both positive and negative disks, which would represent the same image with just two points (the center of each circle), as shown in Fig. 1(c), by the black and white dots.

The basic morphological operators which are able to extract both positive and negative features from given shapes are the Opening-Closing and the Closing-Opening filters. Several representation structures have been proposed based on these operators, e.g., [1-3].

(a) (b) (c)

Fig. 1: (a) A binary image, (b) its Morphological Skeleton, (c) a more natural representation of the image.

Toet's "band-pass" pyramid [1], for instance, is an error-free representation where the Closing-Opening filter replaces the LPF in linear pyramids, prior to the decimation process, and either a Closing or a Dilation filter replaces the LPF needed in the interpolation process. This pyramid does produce a two-sided error-free representation, in which both positive and negative features are selected and classified according to resolution. On the other hand, Toet's pyramid levels contain not only genuine image features, extracted according to size by the Closing-Opening filter, but also spurious features, originated from by down-sampling process [2]. This is because subsampling a morphologically filtered image is not an invertible process [4, 5]. The genuine features and the spurious ones are indistinguishable, as demonstrated in Fig. 2. Fig. 2(a) shows a grayscale image and Fig. 2(b) shows its genuine morphological features, obtained at each step by the difference of the images in the input and output of the Closing-Opening decimation filter. Fig.2(c) shows the related Toet's "band-pass" pyramid, which contains the features from Fig. 2(b) plus spurious features. The pyramid in Fig. 2(b) is not error-free, and the non-morphological spurious features of Toet's pyramid compensate its lossy nature. Throughout this paper, all the "band-pass" pyramidal levels are represented with a shift of 128 in their graylevels, so that negative features can also be shown. The structuring-function used is a 2×2 flat square.

To avoid the generation of spurious features, Zhou and Venetsanopoulos suggested in [2] a different pyramidal representation based on Alternating Sequential Filters (ASF), with no down-sampling. ASF's were introduced by Sternberg [6], and analyzed by Serra [7,

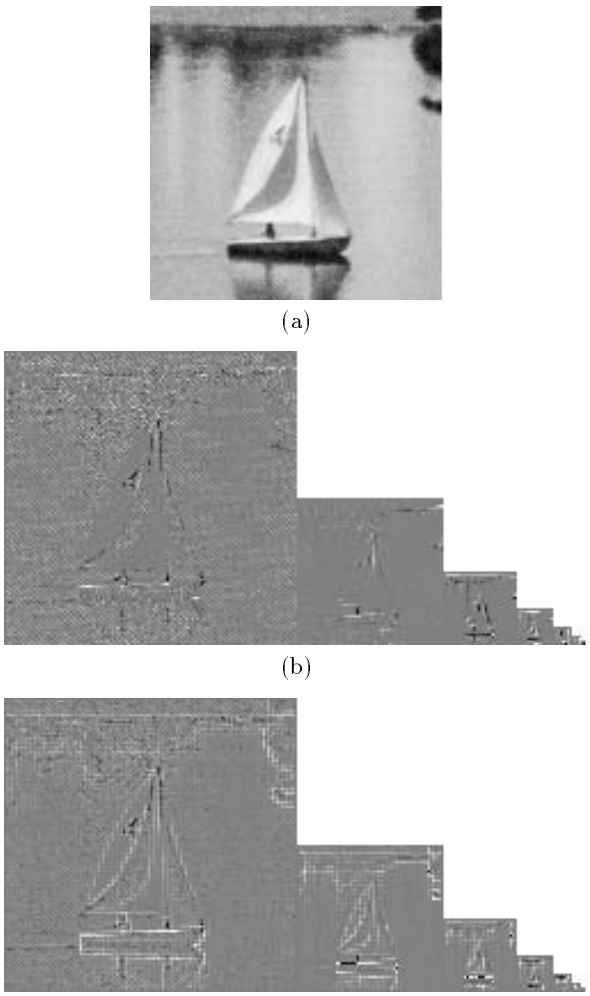


Fig. 2: (a) A 128×128 grayscale image, (b) genuine morphological features, extracted by the Opening-Closing filter at each step, (c) Toet's pyramid, with spurious features originated by the down-sampling process.

chapter 10]. An ASF is obtained by the composition of Opening-Closing (or Closing-Opening) filters, each one using an element bigger than the one in the previous stage. ASF's have been used extensively in image filtering.

In [2, 3], the ASF-based "low-pass" pyramid $\{f_n\}$ was generated in the following way:

$$\begin{aligned} f_n &\equiv (f_{n-1} \circ n \cdot g) \bullet n \cdot g, \quad n \geq 1 \\ f_0 &\equiv f \end{aligned} \tag{1}$$

where $f = f(x,y)$ is the original image, $g = g(x,y)$ is a pre-defined structuring-function, $\oplus$, $\ominus$, $\circ$ and $\bullet$ denote, respectively, grayscale dilation, grayscale erosion, grayscale opening and grayscale closing, and $n \cdot g$ stands for $g(x,y)$ dilated $n-1$ times by itself.

The related "band-pass" pyramid, called *Feature-Width Morphological Pyramid* [2], was defined as the difference between each level f_n and the next level f_{n+1}. Fig. 3 shows the first four levels of the Feature-Width Pyramid.

It is indeed an error-free representation of the image, which takes into account both positive and negative elements, and does not have the disadvantage caused by down-sampling, as in Toet's pyramid. On the other hand, the extracted features (both positive and negative) in level n of this "band-pass" pyramid have width equals to n, as seen in Fig. 3. Which means that for an efficient representation, as needed in coding, and for precision in determining the position of the features, as needed in pattern recognition, *thinning must be performed*. The thinning, which is obtained through erosion, *is not invertible*, thus producing spurious features like the down-sampling in Toet's pyramid.

The Two-Sided Morphological Skeleton presented in this paper is an invertible process, providing a thinned alternative to Zhou and Venetsanopoulos's pyramid. It is an error-free representation, with no spurious features, formed by the centers of both the positive and the negative elements extracted at each level of the ASF-based pyramid defined in (1).

II. TWO-SIDED SKELETON

In this paper we consider only discrete pictures, i.e., sets in Z^2 (binary discrete pictures) or functions over Z^2 (grayscale discrete pictures). The definition and properties of the Two-Sided Skeleton can be extended to continuous pictures (sets or functions over R^2), but this extension is not in the scope of this paper.

A. Binary Pictures

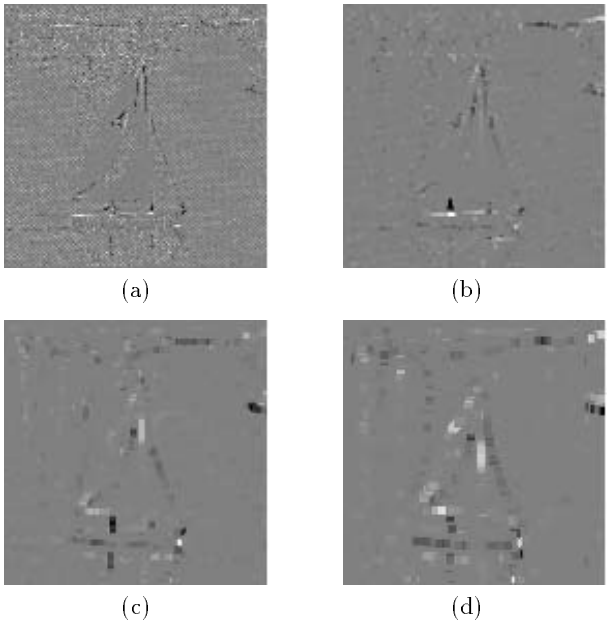


Fig. 3: (a)-(d) The first four levels of the Feature-Width Morphological Pyramid of the image shown in Fig. 2(a).

In the binary case, the ASF-based "low-pass" pyramid is defined as follows:

$$\begin{aligned} X_n &= (X_{n-1} \circ nB) \bullet nB, \quad n \geq 1 \\ X_0 &= X \end{aligned} \tag{2}$$

where, X is the original shape, B is the structuring-element, $\circ$ and $\bullet$ are respectively the binary opening and the binary closing operations.

As opposed to the thinned version of the Feature-Width Pyramid, in which the levels are obtained by *first* taking the difference between consecutive "low-pass" pyramid levels, and *then* performing *thinning*; the Two-Sided Skeleton performs *first* the thinning of each "low-pass" pyramid level and *then* takes the difference. The formal definition is as follows.

The Two-Sided Skeleton of a set $X \subseteq Z^2$ with a given structuring-element $B \subseteq Z^2$ is two collections of sets $\{S_n^+\}_{n=0}^{\infty}$ and $\{S_n^-\}_{n=0}^{\infty}$, where S_n^+ is called *the Positive Skeleton Subset of order* n and S_n^- is called *the Negative Skeleton Subset of order* n. For every natural n, S_n^+ and S_n^- are given by:

$$S_n^- \triangleq (X_{n+1} \oplus nB) - (X_n \oplus nB) \tag{3}$$

$$S_n^+ \triangleq (X_n \ominus nB) - (X_{n+1} \ominus nB) \tag{4}$$

where $\{X_n\}$ is the "low-pass" pyramid defined in (2). In (3) and (4), $\oplus$ denotes binary dilation, $\ominus$ denotes binary erosion, and the minus-sign denotes here the *set-difference* operation. Note that the thinning of *negative* features is obtained by *dilation*, whereas the thinning of *positive* features is obtained by *erosion*.

The *Positive* Skeleton Subsets $\{S_n^+\}$ correspond to the positive side of the Pattern Spectrum; they contain the centers of positive features that represent the original image, where by *feature* we mean a dilated and translated version of the structuring-element. The *Negative* Skeleton Subsets $\{S_n^-\}$ correspond to the negative side of the Pattern Spectrum; they contain the centers of negative features.

Let us define *the Positive Skeleton* of a shape as the union of all its *positive* skeleton subsets, and *the Negative Skeleton* as the the union of all its *negative* skeleton subsets. Figures 4(c) and 4(d) show the Positive and the Negative Skeletons, respectively, of the shape in Fig. 4(a). If we compare them with the ordinary Morphological Skeleton, shown in Fig.4(b), we notice that they represent the shape in a more meaningful and efficient way.

The Two-Sided Skeleton subsets fully represent the original image X. As shown in the Appendix, every level n of the pyramid defined in (1) can be recovered from the lower-resolution level $n+1$ by "adding" the information in the positive and the negative skeleton subsets of order n in the following way:

$$\begin{aligned} X_n = \{&[X_{n+1} \\ &- (S_n^- \oplus nB^s)] \circ nB \\ &\cup (S_n^+ \oplus nB)\} \bullet nB \end{aligned} \tag{5}$$

where B^s denotes the set symmetric to B: $B^s \triangleq \{-b \mid b \in B\}$. Fig. 5 shows the block diagrama of the reconstruction process (5).

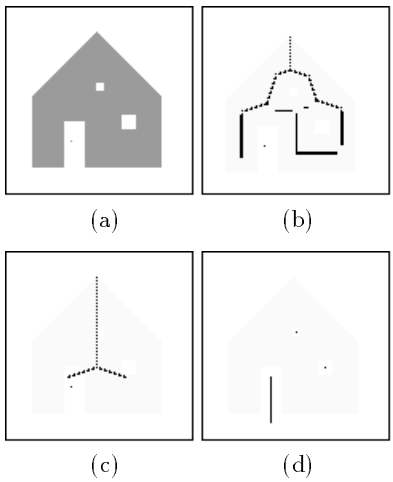


Fig. 4: The Two-Sided Skeleton versus the ordinary Skeleton. (a) A binary Image, (b) its ordinary Morphological Skeleton with a 3x3 square as structuring-element, (c) its *positive* and (d) its *negative* Two-Sided Skeleton subsets with the same structuring-element.

Since, for a bounded X, there exists a natural N such that $\forall n \geq N$, $X_n = \emptyset$, all the information is retained in the sets $\{S_n^+\}_{n=0}^{N-1}$ and $\{S_n^-\}_{n=0}^{N-1}$. By applying (5) successively from $n = N-1$ down to 0, the original image $X = X_0$ is reconstructed.

A partial reconstruction can be obtained by applying (5) from $n = N-1$ down to a given number $k > 0$. The image obtained by this process is the pyramid level X_k, which is a smoothed version of the original image.

B. Grayscale Pictures

A Two-Sided Skeleton may also be defined for a function $f(x,y)$, with a given structuring-function $g(x,y)$. The Positive and Negative Skeleton Function of order n, $s_n^+(x,y)$ and $s_n^-(x,y)$, respectively, are defined as follows:

$$s_n^- \triangleq \begin{cases} (f_{n+1} \oplus n \cdot g) - (f_n \oplus n \cdot g), & \text{if positive} \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

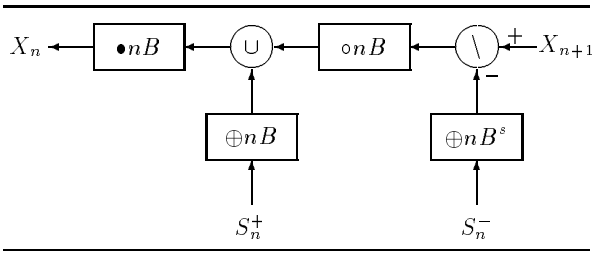


Fig. 5: Block diagram of the recursive reconstruction process. ("\" denotes *set-difference*).

$$s_n^+ \triangleq \begin{cases} (f_n \ominus n \cdot g) - (f_{n+1} \ominus n \cdot g), & \text{if positive} \\ 0, & \text{otherwise} \end{cases} \quad (7)$$

where $\{f_n\}$ is the "low-pass" ASF-based pyramid defined in (1). In (6) and (7), $\oplus$, $\ominus$ denote, respectively, grayscale dilation, and grayscale erosion. Note that all the Positive and Negative Skeleton Functions have non-negative values.

Fig. 6(a) shows a grayscale picture $f(x,y)$ of size 128×128 pixels (the same one as shown in Fig. 2(a)), Fig. 6(b)-(e) its skeleton functions for $n = 0, \ldots, 3$, and Fig. 6(f) the 4^{th} level of the pyramid defined in (1). The image functions in Fig. 6(b)-(e) were obtained by the formula $s_n^+(x,y) - s_n^-(x,y) + 128$, for $n = 0, \ldots, 3$ respectively, so that the darker lines belong to the negative subsets and the brighter lines belong to the positive subsets. Note that all the lines are thin, even for higher values of n. The structuring-function used is flat with the shape of a 2x2 square.

As in the binary case, the reconstruction process is performed iteratively from pyramidal level $n+1$ to the higher-resolution level n with the "addition" of the information of the positive and negative skeleton functions:

$$\begin{aligned} f_n = \{\{[f_{n+1} \oplus n \cdot g \\ - s_n^-] \ominus 2n \cdot g \\ + s_n^+\} \oplus n \cdot g\} \bullet n \cdot g \end{aligned} \quad (8)$$

If $f(x,y)$ is spatially-bounded, i.e., if there exists a positive real M such that $f(x,y) = 0$ for $x^2 + y^2 > M$, then there exists a natural N such that $f_n \equiv 0, \forall n \geq N$. This means that all the information is retained in the Skeleton Functions of orders less or equal to $N-1$, and that an error-free reconstruction from those functions can be obtained.

Partial reconstructions can be obtained as well, by stopping the reconstruction process at any level $k > 0$.

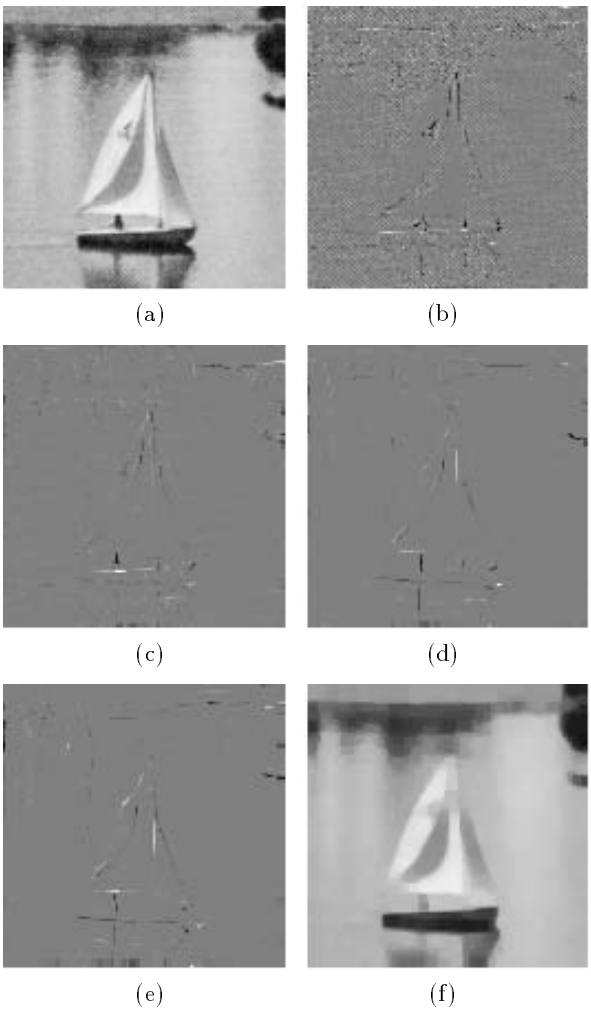


Fig. 6: The Two-Sided Skeleton of grayscale images. (a) The same grayscale image as in Fig. 2(a), (b)-(e) composition of its Positive and Negative Skeleton Functions of order 0 to 3, respectively, (d) the fourth level of the ASF-based pyramid.

III. APPLICATIONS

Some of the application areas, in which the Two-Sided Skeleton can be applied, are examined bellow.

A. Multi-Resolution Analysis

The positive and negative skeleton subsets (in the case of binary images) or functions (in the case of grayscale images) constitute an error-free "band-pass" pyramid, when the related "low-pass" pyramid is the ASF-based pyramid defined in (1) or (2). In this context, the concepts "band-pass" and "low-pass" relate not to frequencies, but to size. But like frequency "band-pass" pyramids, the Two-Sided Skeleton contains at its lower levels (i.e., its subsets - or functions - of lower orders) the finest details of the image, and at its higher levels the largest components of the image.

The information contained at some level n of this "band-pass" pyramid may be viewed as related to those features that belong to *resolution level* n, but not to the lower *resolution level* $n+1$. In this sense, a multi-resolution analysis (and/or processing) can be performed based on the Two-Sided Skeleton subsets (or functions).

As pointed out in section I, the Two-Sided Skeleton is preferable to the "band-pass" pyramid presented in [1] (Fig. 2(b)) and the to *thinned* version of the Feature-Width Pyramid presented in [2, 3], because its implicit thinning process is error-free. This prevents the pyramidal levels to contain spurious features, and therefore provides a more consistent representation.

B. Robust Representation

Although it is not a robust representation, the ordinary Skeleton has some degree of insensitivity to *positive noise*, which is noise that contaminates only the *background* of a binary image, or that only *increments* the values of a grayscale image. However, *negative noise*,

which contaminates the *foreground* of a binary image or *decrements* the values of a grayscale image, often alters completely the ordinary Skeleton of an image. Fig. 7(a)-(d) illustrate the behavior of the ordinary Skeleton of a binary image in presence of negative noise: Fig. 7(a) is the original binary image and Fig. 7(b) is the same image with 1% negative binary noise. Fig. 7(c) and Fig. 7(d) are the ordinary Skeleton of the images in Fig. 7(a) and Fig. 7(b), respectively. Note how the shape of the skeleton changes.

The Two-Sided Skeleton has the same degree of insensitivity to both positive and negative noise. Since in many applications the corrupting noise has both positive *and* negative components, the Two-Sided Skeleton should be preferred for working under noisy conditions. Fig. 7(e-f) demonstrate the behavior of the Two-Sided Skeleton in the presence of noise. Fig. 7(e) and Fig. 7(f) are the Two-Sided Skeleton of Fig. 7(a) and Fig. 7(b), respectively, where the black points refer to the negative skeleton and the white points to the positive skeleton. The structuring-element used in the simulation was *Rhombus* (the origin and its 4-pixel neighborhood).

C. Compression and Progressive Transmition

The Two-Sided Skeleton subsets (or functions) provide a compact representation of the original image, which permits also efficient Coding of the image.

In the grayscale case, compression may be achieved by techniques similar to these applied to other "band-pass" pyramids. Some of those techniques are:

- Allowing a different quantization error at each resolution level. The higher the resolution level, the greater the permitted quantization error. The highest resolution level(s) may sometimes be totally discarded. (The highest resolution levels are related to the lowest function orders).
- Applying different coding methods at different groups of resolution levels.

For instance, the Two-Sided Skeleton can replace the thinned version of the Feature-Width Pyramid in the coding method proposed in [2] and [3]. This is expected to improve the quality of the reconstructed image because there is no loss of information in the Two-Sided Skeleton implicit thinning process. After the Two-Sided Skeleton is obtained, the coding is done as in [2] of [3]: the *highest resolution levels* are decomposed by a directional morphological filter bank into directional images, which are scanned in the respective direction and have their run-lengths coded. The remaining coarse image is coded by a vector quantization scheme.

The interest in using a *morphological* pyramid, such as the Two-Sided Skeleton for image coding, is that it represents image edges in a more efficient way than linear pyramids. Since the ordinary Skeleton represent explicitly the edges of *positive elements* only, it is less suitable than two-sided representations for efficient coding.

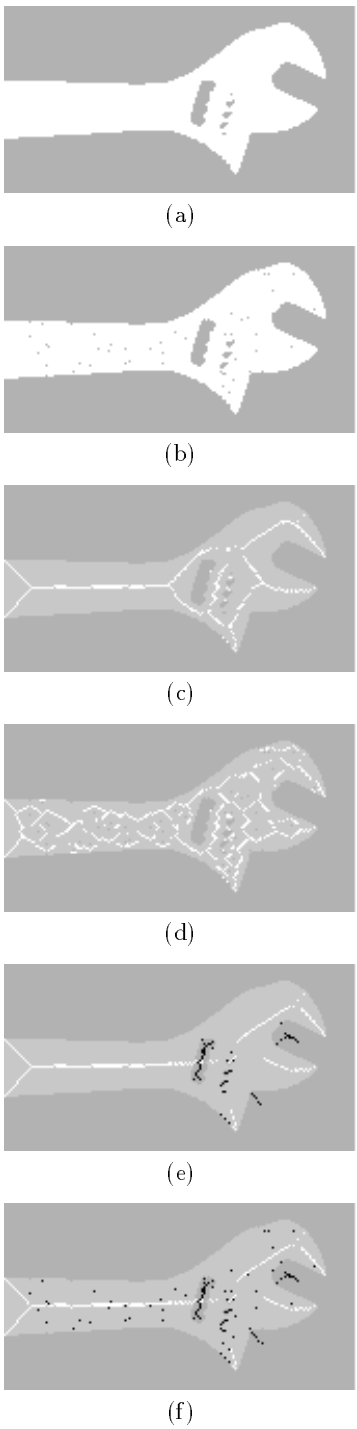


Fig. 7: The behavior of the ordinary and the Two-Sided Skeletons under noisy conditions: (a) A grayscale image, (b) The same image, but with 1% binary negative noise, (c) the ordinary Skeleton of (a), (d) the ordinary Skeleton of (b), (e) the Two-Sided Skeleton of (a), (f) the Two-Sided Skeleton of (b).

In the binary case, the Two-Sided Skeleton usually achieves compression rates similar to those of the ordinary Morphological Skeleton, even though there are

two Skeleton subsets at each step n, instead of only one as in the ordinary Skeleton. This is because there are usually less representation points in the Two-Sided Skeleton than in the ordinary Skeleton, as was demonstrated in Fig. 4.

In either case, the Two-Sided Skeleton is also suitable for Progressive Transmition (from the highest subset orders to the lowest ones), since the reconstruction is performed on a level-by-level basis and since each level "adds" more details.

D. Image Decomposition

Every Two-Sided-Skeleton point is accompanied by two parameters: size (the order of the skeleton subset or function) and side of the Pattern Spectrum (whether it belongs to a positive or to a negative subset or function). By selecting appropriate Two-Sided-Skeleton points, one can obtain decompositions of the original image, according to those parameters.

E. Pattern Recognition

Each Two-Sided Skeleton point is the center of a dilated version of the structuring-element. If the structuring-element is convex, the dilated versions are also scaled versions of it. Thus, it is possible to search for a certain convex shape in a given image by analyzing the Two-Sided Skeleton of that image, when the structuring-element used in the calculation is the convex shape to be found. Such method, *using the ordinary Skeleton*, is expected to detect positive elements only.

IV. CONCLUSIONS

A new morphological representation structure, the Two-Sided Skeleton, has been defined, both for binary and grayscale images, and its applications were discussed. It was shown that the new structure is able to represent image details in a more meaningful way than the ordinary Morphological Skeleton, since it considers both the positive and the negative features of the image. The Two-Sided Skeleton characteristics as a multi-resolution pyramid, and its advantages over two other similar morphological multi-resolution representations, were also pointed out.

APPENDIX

Proof of the reconstruction relation in equation (5).

First we notice that for all n:

$$X_{n+1} \bullet (n+1)B = X_{n+1} \qquad \text{(A.1)}$$

$$X_{n+1} \bullet nB = X_{n+1} \qquad \text{(A.2)}$$

Relation (A.1) is a direct consequence of the definition for $\{X_n\}$ in (2), and (A.2) is obtained from (A.1).

Using (A.2), and since $A \ominus B - C \oplus B^s = (A-C) \ominus B$, $\forall A, B, C$, we get:

$$\begin{aligned} X_{n+1} - (S_n^- \oplus nB^s) &= \\ &= [(X_{n+1} \oplus nB) \ominus nB] - (S_n^- \oplus nB^s) \\ &= [(X_{n+1} \oplus nB) - S_n^-] \ominus nB \end{aligned} \qquad \text{(A.3)}$$

Also $S_n^- = (X_{n+1} \oplus nB) - (X_n \oplus nB)$ (by definition), and $A - (A - B) = A \cap B$, $\forall A, B$. Therefore:

$$\begin{aligned} [(X_{n+1} \oplus nB - S_n^-] \ominus nB = \\ &= [(X_{n+1} \oplus nB) \cap (X_n \oplus nB)] \ominus nB \\ &= [(X_{n+1} \bullet nB) \cap (X_n \bullet nB)] \end{aligned} \qquad \text{(A.4)}$$

From (A.1), (A.2), (A.3) and (A.4), we obtain:

$$X_{n+1} - (S_n^- \oplus nB^s) = X_{n+1} \cap X_n \qquad \text{(A.5)}$$

Performing opening and "adding" now the information in S_n^+, it follows:

$$\begin{aligned} [(X_{n+1} \cap X_n) \circ nB] \cup (S_n^+ \oplus nB) = \\ &= [(X_{n+1} \cap X_n) \ominus nB \cup S_n^+] \oplus nB \\ &= [(X_{n+1} \ominus nB) \cap (X_n \ominus nB) \cup S_n^+] \oplus nB \end{aligned}$$

Using $S_n^+ = (X_n \ominus nB) - (X_{n+1} \ominus nB)$ (by definition) and noting that $(A \cap B) \cup (A - B) = A$, $\forall A, B$, we obtain:

$$\begin{aligned} [(X_{n+1} \ominus nB) \cap (X_n \ominus nB) \cup S_n^+] \oplus nB = \\ &= [(X_{n+1} \ominus nB) \cap (X_n \ominus nB) \cup \\ &\quad \cup (X_n \ominus nB) - (X_{n+1} \ominus nB)] \oplus nB \\ &= (X \ominus nB) \oplus nB = X_n \circ nB \end{aligned} \qquad \text{(A.6)}$$

Finally we use the fact that the opening-closing operation is idempotent, and that $X_n = X_{n-1} \circ nB \bullet nB$, to state that:

$$(X_n \circ nB) \bullet nB = X_n \qquad \text{(A.7)}$$

The reconstruction relation in (5) is then obtained from (A.5), (A.6) and (A.7).

References

[1] A. Toet, "A Morphological Pyramidal Image Decomposition", *Pattern Recognition Letters*, No.9, May 1989, pp. 255-261.

[2] Z. Zhou and A. N. Venetsanopoulos, "Directional Decomposition of Morphological Pyramid Representations for Low Bit-Rate Image Coding", *1991 International Conference on Digital Signal Processing*, Florence, Italy, September 4-6, 1991, pp. 404-409.

[3] Z. Zhou and A. N. Venetsanopoulos, "Morphological Methods in Image Coding", *ICASSP-92*, pp. III-481–III-484.

[4] R. M. Haralick et al., "The Digital Morphological Sampling Theorem". In: R. Kasture and M. M. Trivedi, *Image Analysis Applications*, Marcel Dekker, 1990.

[5] H. J. A. M. Heijmans and A. Toet, "Morphological Sampling", *CVGIP: Image Understanding*, Vol. 54, No. 3, November 1991, pp. 384-400.

[6] S. R. Sternberg, "Morphology for grey tone functions", *CVGIP* 35, 1986.

[7] J. Serra, *Image analysis and mathematical morphology, Volume 2: Theoretical Advances.* London: Academic Press, 1988.

Bibliography

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

[2] H. Blum, "A Transformation for Extracting New Descriptors of Shape", in *Models for the Perception of Speech and Visual Forms*, W. Wathen-Dunn Ed., Cambridge, MA: M.I.T. Press, 1967.

[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] P. Brigger, M. Kunt, and F. Meyer, "The Geodesic Morphological Skeleton and Fast Transformation Algorithms", *Proc. of the Intern. Symp. on Mathematical Morphology ISMM'94*, J. Serra and P. Soille (eds.), pp. 133-140, September 1994.

[5] P.J. Burt and E.H. Adelson, "The Laplacian Pyramid as a Compact Image Code", *IEEE Trans. Commun.*, Vol. COM-31, No. 4, pp. 532-540, April 1983.

[6] J.R. Casas, P. Esteban, A. Moreno, and M. Carrera, "Morphological Scheme for Morphometric Analysis of Epidermal Biopsy Images", *Proc. of the Intern. Symp. on Mathematical Morphology ISMM'94*, J. Serra and P. Soille (eds.), pp. 325-331, September 1994.

[7] R.R. Coifman and M.V. Wickerhauser, "Entropy-Based Algorithms for Best-Basis Selection", *IEEE Trans. Information Theory*, Vol. 38, No. 2, pp. 713-718, March 1992.

[8] L. Dorst and R. van den Boomgaard, "An Analytical Theory of Mathematical Morphology", *Proc. Int'l Work. Math. Morphology & Applic. to Signal Processing*, Barcelona, pp. 245-250, May 1993.

[9] L. Dorst and R. van den Boomgaard, "Morphological Signal Processing and Slope Transform", *Signal Processing*, Vol. 38, September 1994.

[10] E.R. Dougherty, *An Introduction to Morphological Image Processing*, SPIE Volume TT 9, Washington, 1992.

[11] D.A.F. Florêncio and R.W. Schafer, "Critical Morphological Sampling and Its Applications to Image Coding", *Proc. of the Intern. Symp. on Mathematical Morphology ISMM'94*, J. Serra and P. Soille (eds.), pp. 109-116, September 1994.

[12] C.R. Giardina and E.R. Dougherty, *Morphological Methods in Image and Signal Processing*, Prentice Hall, Englewood Cliffs, 1988.

[13] C. Gu and M. Kunt, "Application fo Morphological Filters for Contour Image Sequence Coding", *Proc. of the Intern. Symp. on Mathematical Morphology ISMM'94*, J. Serra and P. Soille (eds.), pp. 125-132, September 1994.

[14] 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. Yamazake and T. Yoshida, "Technical Features of the JBIG Standard for Pogressive Bi-Level Image Compression", *Signal Processing: Image Communication*, Vol. 4, pp. 103-111, 1992.

[15] H.J.A.M. Heijmans, "On the Construction of Morphological Operators Which are Self-dual and Activity-Extensive", *Proc. Intern. Workshop on Mathematical Morphology*, Barcelona, pp. 82-88, May 1993.

[16] P.G. Howard and J.S. Vitter, "Arithmetic Coding for Data Compression", *Proceedings of the IEEE*, Vol. 82, No. 6, pp. 857-865, June 1994.

[17] R. Kresch and D. Malah, "Morphological Multi-Structuring-Element Skeleton and its Applications", *Proc. of ISSSE'92*, Paris, pp. 166-169, September 1992.

[18] R. Kresch and D. Malah, "Morphological Reduction of Skeleton Redundancy", *Proc. Intern. Workshop on Mathematical Morphology*, Barcelona, pp. 145-150, May 1993.

[19] R. Kresch and D. Malah, "Two-Sided Skeleton - A Representation Composed of Both Positive and Negative Morphological Elements", *Proc. Intern. Workshop on Mathematical Morphology*, Barcelona, pp. 145-150, May 1993.

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

[21] R. Kresch and D. Malah, "Multi-Parameter Skeleton Decomposition", *Proc. of the Intern. Symp. on Mathematical Morphology ISMM'94*, J. Serra and P. Soille (eds.), pp.141-148, September 1994.

[22] R. Kresch and D. Malah, "Skeleton Redundancy Reduction Based on a Generalization of Convexity", *EUSIPCO-94*, Edimburgh, September 1994.

[23] R. Kresch and D. Malah, "An Efficient Coding Scheme for Binary Images Based on the Morphological Skeleton Representation", *Proc. IEEE 18th Convention of Electrical and Electronics Engineers in Israel*, Tel Aviv, pp. 2.2.3/1-5, March 1995.

[24] R. Kresch and D. Malah, "New Morphological Skeleton Properties Leading to Its Efficient Coding", to appear in *IEEE Workshop on Non-Linear Signal and Image Processing 1995*, Neos Marmaras, Halkidiki, Greece, June 1995.

[25] R. Kresch and D. Malah, "Quadtree and Bit-Plane Decomposition as Particular Cases of the Generalized Morphological Skeleton" to appear in *IEEE Workshopon Non-Linear Signal and Image Processing 1995*, Neos Marmaras, Halkidiki, Greece, June 1995.

[26] M. Kunt, A. Ikonomopoulos, M. Kocher, "Second-Generation Image-Coding Techniques", *Proc. of the IEEE*, Vol. 73, No. 4, pp. 549-574, April 1985.

[27] C. Lantuéjoul, *La Squelettisation et son application aux mesures topologuiques des mosaiques polycristallines*, theses de Docteur-Ingenieur, School of Mines, Paris, France, 1978.

[28] S.G. Mallat and Z. Zhang, "Matching Pursuits With Time-Frequency Dictionaries", *IEEE Trans. Signal Processing*, Vol. 41, No. 12, pp. 3397-3415, December 1993.

[29] 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.

[30] P. Maragos, "Pattern Spectrum and Multiscale Shape Representation", *PAMI*, Vol.11, No.7, pp. 701-716, July 1989.

[31] P. Maragos, "Max-Min Difference Equations and Recursive Morphological Systems", *Proc. Int'l Work. Math. Morphology & Applic. to Signal Processing*, Barcelona, pp. 168-173, May 1993.

[32] P. Maragos, "Morphological Systems: Slope Transforms and Max-Min Difference and Differential Equations", *Signal Procesing*, Vol. 38, September 1994.

[33] F. Marqués, T. Megía, N. Joshi, and A. Navarro-Quilis, "Automatic Quantification of Spine Parameters from X-Ray Images by Means of Morphological Tools", *Proc. of the Intern. Symp. on Mathematical Morphology ISMM'94*, J. Serra and P. Soille (eds.), pp. 333-340, September 1994.

[34] F. Meyer and S. Beucher, "Morphological Segmentation", *JVCIR*, Vol. 1, No. 1, pp. 21-46, September 1990.

[35] L.A. Overturf, M.L. Comer, and E.J. Delp, "Color Image Coding Using Morphological Pyramid Decomposition", *IEEE Trans. on Image Processing*, Vol. 4, No. 2, pp. 177-185, February 1995.

[36] S. Peleg and A. Rosenfeld, "A Min-Max Medial Axis Transformation", *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 3, No. 2, pp. 208-210, 1981.

[37] I. Pitas and A.N. Venetsanopoulos, "Morphological Shape Decomposition", *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 12, No. 1, pp. 38-45, 1990.

[38] I. Pitas and A.N. Venetsanopoulos, *Nonlinear Filters in Image Processing: Principles and Applications*, Kluwer Academic Publishers, Boston, 1990.

[39] S.A. Rajala, H.A. Peterson, and E.J. Delp, "Binary Morphological Coding of Grayscale Images", *ISCAS'88*, pp. 2807-2811, 1988.

[40] J.M. Reinhardt and W.E. Higgins, "Shape Representation: Comparison Between the Morphological Skeleton and Morphological Shape Decomposition", *ICIP'94*, Vol. 2, pp. 91-95, 1994.

[41] C. Ronse and B. Macq, "Morphological Shape and Region Description", *Signal Processing*, Vol. 25, No. 1, pp. 91-105, October 1991.

[42] A. Rosenfeld and A.C. Kac, *Digital Picture Processing*, Academic Press, 1976.

[43] Ph. Salembier, L. Torres, M. Pardas, F. Marqués, Hierro, and A. Gasull, "Morphological Segmentation-Based Coding of Image Sequences", *EECTD-93*, pp. 1245-1249, 1993.

[44] Ph. Salembier, "Morphological Multiscale Segmentation for Image Coding", *Signal Processing*, Vol.38, pp. 359-386, 1994.

[45] G. Sapiro and D. Malah, "Morphological Image Coding via Bit-Plane Decomposition and a New Skeleton Representation", *Proc. of the 17th IEEE convention of Electrical and Electronics Engineers in Israel*, pp.174-177, May 1991.

[46] G. Sapiro and D. Malah, "A Geometric Sampling Theorem and its Applications in Morphology Image Coding", *Proc. of the International Conference on Digital Signal Processing*, Florence, pp.410-415, September 1991.

[47] G. Sapiro, *Image Coding by Morphological Techniques*, M.Sc. Thesis (in Hebrew), Department of Electrical Engineering, Technion, September 1991.

[48] G. Sapiro and D. Malah, "Morphological Image Coding Based on a Geometric Sampling Theorem and a Modified Skeleton Representation", *Journal of Visual Communication and Image Representation*, Vol. 5, No. 1, pp. 29-40, March 1994.

[49] D. Schonfeld and J. Goutsias, "A Fast Algorithm for the Morphological Coding of Binary Images", *SPIE* Vol. 1001, pp.138-145, 1988.

[50] D. Schonfeld and J. Goutsias, "Morphological Representation of Discrete and Binary Images", *Trans. Signal Processing*, Vol. 39, No. 6, pp. 1369-1379, June 1991.

[51] J. Serra, *Image Analysis and Mathematical Morphology.* London: Academic Press, 1982.

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

[53] J. Serra, "Skeleton Decompositions", *SPIE Vol. 1769, Image Algebra and Morphological Image Processing III*, pp. 376-386, 1992.

[54] D. Shaked, *Symmetry Invariance and Evolution in Planar Shape Analysis*, D.Sc. Thesis (in English), Department of Electrical Engineering, Technion, June 1995.

[55] J. Shapiro, "An Embedded Wavelet Hierarchical Image Coder", *Proc. 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP-92*, Vol. IV, pp. 657-660, 1992.

[56] J. Song and E.J. Delp, "The Analysis of Morphological Filters with multiple structuring-elements", *CVGIP*, Vol.50, pp.308-328, 1990.

[57] S.R. Sternberg, "Grayscale Morphology", *CVGIP*, Vol.35, pp.333-355, 1986.

[58] F. Sun and P. Maragos, "Experiments on Image Compression Using Morphological Pyramids", *SPIE, Visual Communications and Image Processing*, Vol. 1199, pp. 1303-1312, 1989.

[59] A. Toet, "A Morphological Pyramid Image Decomposition", *Pattern Recognition Letters 9*, pp. 255-261, May 1989.

[60] G.T. Toussaint (Ed), *Computational Morphology*, North-Holland, 1988.

[61] L. Vincent and P. Soille, "Watersheds in Digital Spaces: An Efficient Algorithm Based on Immersion Simulations", *PAMI*, Vol. 13, No. 6, pp. 583-597, June 1991.

[62] Z. Zhou and A.N. Venetsanopoulos, "Directional Decomposition of Morphological Pyramid Representations for Low Bit-Rate Image Coding", *Proc. Int'l. Conf. Digital Signal Processing-91*, pp. 404-409, 1991.