Copyright Warning & Restrictions

The copyright law of the United States (Title 17, United States Code) governs the making of photocopies or other reproductions of copyrighted material.

Under certain conditions specified in the law, libraries and archives are authorized to furnish a photocopy or other reproduction. One of these specified conditions is that the photocopy or reproduction is not to be "used for any purpose other than private study, scholarship, or research." If a, user makes a request for, or later uses, a photocopy or reproduction for purposes in excess of "fair use" that user may be liable for copyright infringement,

This institution reserves the right to refuse to accept a copying order if, in its judgment, fulfillment of the order would involve violation of copyright law.

Please Note: The author retains the copyright while the New Jersey Institute of Technology reserves the right to distribute this thesis or dissertation

Printing note: If you do not wish to print this page, then select "Pages from: first page # to: last page #" on the print dialog screen

Order Number 9409124

Morphological operations in image processing and analysis

Pu, Chamim Christopher, Ph.D.

New Jersey Institute of Technology, 1993

U·M·I
300 N. Zeeb Rd.
Ann Arbor, MI 48106

MORPHOLOGICAL OPERATIONS IN IMAGE PROCESSING AND ANALYSIS

by
Chamim Christopher Pu

A Dissertation
Submitted to the Faculty of
New Jersey Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of
Doctor of Philosophy

Department of Computer and Information Science

October 1993

APPROVAL PAGE

Morphological Operations in Image Processing and Analysis

by

Chamim Christopher Pu

Dr. Frank Y. Shih, Dissertation Advisor — date

Associate Professor of Computer and Information Science, NJIT

Dr. James A. M. McHugh, Committee Member, — date

Associate Chairperson and Professor

of Computer and Information Science, NJIT

Dr. David Nassimi, Committee Member, — date

Associate Professor of Computer and Information Science, NJIT

Dr. DaoChaun Hung, Committee Member, — date

Assistant Professor of Computer and Information Science, NJIT

Dr. Nirwan Ansari, Committee Member, — date

Assistant Professor of Electrical Engineering, NJIT

ABSTRACT

Morphological Operations in Image Processing and Analysis

by

Chamim Christopher Pu

Morphological operations applied in image processing and analysis are becoming increasingly important in today's technology. Morphological operations which are based on set theory, can extract object features by suitable shape (structuring elements). Morphological filters are combinations of morphological operations that transform an image into a quantitative description of its geometrical structure which based on structuring elements. Important applications of morphological operations are shape description, shape recognition, nonlinear filtering, industrial parts inspection, and medical image processing.

In this dissertation, basic morphological operations are reviewed, algorithms and theorems are presented for solving problems in distance transformation, skeletonization, recognition, and nonlinear filtering. A skeletonization algorithm using the maxima-tracking method is introduced to generate a connected skeleton. A modified algorithm is proposed to eliminate non-significant short branches. The back propagation morphology is introduced to reach the roots of morphological filters in only two-scan. The definitions and properties of back propagation morphology are discussed. The two-scan distance transformation is proposed to illustrate the advantage of this new definition.

G-spectrum (geometric spectrum) which based upon the cardinality of a set of non-overlapping segments in an image using morphological operations is presented to be a useful tool not only for shape description but also for shape recognition. The G-spectrum is proven to be translation-, rotation-, and scaling-invariant. The shape likeliness based

on G-spectrum is defined as a measurement in shape recognition. Experimental results are also illustrated.

Soft morphological operations which are found to be less sensitive to additive noise and to small variations are the combinations of order statistic and morphological operations. Soft morphological operations commute with thresholding and obey threshold superposition. This threshold decomposition property allows gray-scale signals to be decomposed into binary signals which can be processed by only logic gates in parallel and then binary results can be combined to produce the equivalent output. Thus the implementation and analysis of function-processing soft morphological operations can be done by focusing only on the case of sets which not only are much easier to deal with because their definitions involve only counting the points instead of sorting numbers, but also allow logic gates implementation and parallel pipelined architecture leading to real-time implementation. In general, soft opening and closing are not idempotent operations, but under some constraints the soft opening and closing can be idempotent and the proof is given. The idempotence property gives us the idea of how to choose the structuring element sets and the value of index such that the soft morphological filters will reach the root signals without iterations. Finally, summary and future research of this dissertation are provided.

BIOGRAPHICAL SKETCH

Author: Chamim Christopher Pu

Degree: Doctor of Philosophy in Computer Science

Date: October 1993

Undergraduate and Graduate Education:

- Doctor of Philosophy in Computer Science,
  New Jersey Institute of Technology, Newark, NJ, 1993
- Master of Science in Computer Science,
  The University of Arizona, Tucson, AZ, 1988
- Bachelor of Science in Computer Engineering,
  National Chiao Tung University, Hsinchu, Taiwan, 1983

Major: Computer Science

Presentations and Publications:

F. Shih and C. Pu, "Medial axis transformation with single-pixel and connectivity preservation using Euclidean distance computation," *Proc. IEEE Inter. Conf. Pattern Recog.*, Atlantic City, New Jersey, pp. 723-725, June 1990.

F. Shih and C. Pu, "A maxima-tracking method for skeletonization from Euclidean distance function," *Proc. IEEE Computer Soc. Conf. on Tools for Artif. Intell.*, San Jose, CA, pp 246-253, Nov. 1991.

F. Shih and C. Pu, "Morphological shape description using geometric spectrum on multidimensional binary images," *Pattern Recognition*, Vol. 25, No. 9, pp 921-927, 1992.

F. Shih, C. King and C. Pu, "Pipeline architectures for recursive morphological operations," *IEEE Trans. on Image Processing*, to appear.

C. Pu and F. Shih , "Soft mathematical morphology: binary and gray scale," *Proc. EURASIP Math. Morphology Appl. Signal Proc.*, Spain, May 1993.

F. Shih and C. Pu, "Threshold decomposition of soft morphological filters," *Proc. IEEE Inter. Conf. on CVPR*, New York City, June 1993.

This dissertation is dedicated to
God, our heavenly Father

ACKNOWLEDGMENT

The doctoral research should be enjoyed and indeed it was joyful. This was partly due to the helpful faculty and colleagues in the Department of Computer and Information Science, from whom, the author received encouragement. The author is grateful to the friends made during this research and would like to extend special thanks to them.

First of all, the author would like to thank his research advisor, Professor Frank Shih, for his advising and guidance during the five years of doctoral research. His valuable time was instrumental to the author in processing this research. The author is proud to co-author with him in many conferences, journals, and papers.

Special thanks to Professor Peter Ng, the chairperson of CIS and the CIS department for the financial support given to me during the period of this research.

The author appreciates the help and suggestions from Professors James McHugh, David Nassimi, Dao-Chaun Hung, and Nirwan Ansari for serving as members of the committee.

Sincere thanks to the author's parents for their never ending support, understanding, and encouragement during the years of advance studies. The author is grateful to his wife Beatrice for her love and consistent support in sharing her life with him.

Finally, the author give heartful thanks to God, our Heavenly Father. "Unless the Lord builds the house, its builders labor in vain. Unless the Lord watches over the city, the watchmen stand guard in vain." (Psalm 127:1-2) It is He who made it possible for the author to complete this dissertation.

From the rising of the sun until its going down, praise the LORD.

TABLE OF CONTENTS

Chapter **Page**

LIST OF FIGURES

Figure **Page**

CHAPTER I

INTRODUCTION

1.1 Motivation

Image processing and analysis is important in many of today's technology: aerial surveillance photographs, slow-scan television images of the moon or of planets gathered from space probes, television images taken from an industrial robot's eyes, chromosome scans, X-ray images, computerized axial tomograph scans, and fingerprint analysis.

Mathematical morphology is becoming increasingly important in image processing and computer vision applications for object representation, recognition, and defect inspection. Morphological operations applyed in image processing and analysis are briefly reviewed. The morphological operations can be employed for many purposes, including digitalization, compression, enhancement, restoration, reconstruction, matching, segmentation, representation, and description.

Mathematical morphology based on the geometric shape, provides a particular approach to the processing and analysis of digital images. This approach of the analysis and processing of digital images is generally based upon predetermined geometric shape known as *structure elements*. The underlying strategy is to understand the characteristics of an object by probing its microstructure with various forms which are structuring elements. The analysis is geometric in character and it approaches image processing from the vantage point of human perception. Appropriately used, morphological operations also tend to simplify image data while preserving their essential shape characteristics and eliminating irrelevancies. Morphology is a set-theoretical method which was first introduced by Matheron and Serra at the Paris School of Mines, France around 1964 who wanted to analyze the structure of microscopic images from geologic and metallic

specimens and to relate the results of this analysis to the physical properties of rocks and minerals. It becomes one of the active and vigorous fields in image processing and analysis in recent years.

One purpose of mathematical morphology is the quantitative description of geometric structures and it achieves this goal by probing and transforming an image with different patterns of predefined shapes to extract different pieces of information.

Morphological operations can be employed for edge detection, segmentation, and enhancement of images which provides for the systematic alteration of the geometric content of an image while maintaining the stability of important geometric characteristics. Moreover, there exist a well-developed morphological algebra that can be employed for representation and optimization and it is possible to express digital algorithms in terms of a very small class of primitive morphological operations. Finally, there exist rigorous representation theorems by means of which one can obtain the expression of morphological filters in terms of the primitive morphological operations.

The digitization problem (or sampling problem) can be stated as followed: given a Euclidean image mapping Ψ, can one find a digital mapping which preserves the same topological features. More generally, what is the relationship between performing a less costly morphological filtering operation on the sampled image, and performing the more costly equivalent morphological filtering operations on the original image. Haralick *et al* presented a *morphological sampling theorem* which stated how a digital image must be morphologically filtered before sampling in order to preserve the relevant information after sampling, to what precision an appropriately morphologically filtered image can be reconstructed after sampling, and the relationship between morphological filters operating before sampling and the more computationally efficient scheme of morphologically operating on the sampled image with a sampled structuring element [32, 35, 36].

The steady of modern communication requirements has resulted in an increase in

the volume of pictorial data that must be transmitted from one location to another. Although image transformation to a remote location is not necessary in some cases, one does need to store the images for future retrieval and analysis with as little memory as possible. In the past, most of the image transmission has been accomplished through the conventional analog technique. Today, the trend in image transmission and storage is to use digital technique. Shih [80] proposed an image compression technique which uses the mathematical morphology. This approach can be extended into three-dimensional images and can be improved with adopting variable length coding. Other technique adopted in representation can be used as compression techniques, such as morphological skeleton [50, 56] and the morphological representation [29].

Whenever a picture or image is converted from one to another, e.g., copied, scanned, transmitted, or displayed, the "quality" of the output image may be lower than that of the input. These include methods of modifying the gray scale (e.g., increasing contrast), deblurring, smoothing, and removing noise. An important problem in image processing and analysis applications is to develop an efficient filtering procedure that restores a binary image from its noisy version [75, 76, 77, 57, 58]. There are two issues we have to consider: 1) the filtering operators should be efficient enough to eliminate the noise degradation, and 2) the filtering operators should be able to restore important aspects of the shape-size content of the noise-free image as well as to preserve its crucial geometric and topological image description. Schonfeld and Goutsias [74] presented a theoretical analysis of morphological filters for the *optimal* restoration of noisy binary images. The problem is formulated in a general form and an optimal solution is obtained by using fundamental tools from mathematical morphology and decision theory. The important result of their analysis is the fact that the class of alternating sequential filters is the class of parametric, smoothing morphological filters that optimally restore noisy binary images in the least mean difference sense.

The objective of scene segmentation is to separate the components of an image into

subsets that correspond to the physical objects in the scene. The segmented components are then used by higher-level processes for interpretation and recognition. It should be emphasized that there is no single standard approach to segmentation. Many different types of picture or scene parts can serve as the segments on which descriptions are based, and there are many different ways in which one can attempt to extract these parts from the image.

Maragos and Ziff showed that many composite morphological systems, such as morphological edge detection, peak/valley extraction, skeletonization, and shape-size distributions obey a weak linear superposition which is called *threshold-linear superposition* [60]. Namely, the output is the sum of outputs due to input binary images that result from thresholding the input gray-level image at all levels. The threshold decomposition architecture and stacking property are introduced by Shih and Mitchell [84] which allows the implementation of the architecture that gray-scale operations can be decomposed into binary operation with the same dimensionality as the original operations. In [85, 86], Shih and Mitchell presented techniques for decomposing big gray-scale morphological structuring elements into combined structures of segmented small components. The decomposition is suitable for parallel pipelined architecture and the technique will allow us to design any kind and size of gray-scale structuring elements.

Edge operators based on gray-scale morphological operations are introduced by Lee, Haralick and Shapiro [48]. These operators can be efficiently implemented in near real time machine vision system with gray-scale morphological operations hardware support. The simplest morphological edge detectors are the dilation residue and erosion residue operators. Different combinations of these two simple operators is also introduced and justified. The blur-minimum morphological edge operator is defined whose inherent noise sensitivity is less than the dilation or erosion residue operators.

In the case of binary images which are mainly perceived as geometrical patterns.

There is a need for image representations that emphasize geometric structure. One of such representations is the *skeleton*. In general, skeleton are used to describe a line-thinned caricature of binary image that summarizes its shape and conveys information about its size, orientation, and connectivity. The most important property of distance transformation is the concept of skeleton and reconstruction of the original picture from its skeleton. Skeletonization in image processing is a very important issue. The skeleton not only save the memory space but also save the computation cost of processing and recognition. In [87, 91], the idea of using Euclidean distance was proposed, which provides the global information about object shape and can be used to reconstruct the original object. Moreover, the skeleton sets can be used as the base of shape decomposition which hierarchical skeletons will be discussed.

Shape decomposition is a very important issue in image processing and pattern recognition. Shape decomposition is to decompose binary objects into a union of *simple* objects. The decomposition should be unique, translation, scaling, and rotation invariant. Some morphological approach [67, 68] is to use the structuring elements as the simplest object component and to analyze an image as a union of the structuring elements. This approach is based on the structuring elements used, therefore, an object is represented by those structuring elements.

Shape description describes the object according to its shape geometric features. The shape of an object refers to its profile and physical structure. These characteristics can be represented by the boundary, region, moment, and structural representations. These representations can be used for matching shapes, recognizing objects, or making measurements on the shape characteristics. Therefore, the shape description is a very active and important issue in image processing, computer vision, and pattern recognition during recent decades.

Maragos [50, 51, 55] proposed a shape-size descriptor which is called *pattern*

spectrum. The discrete version of pattern spectrum is a very useful quantity for shape analysis. Bronskill and Venetsanopoulos proposed *pecstrum* [8] of which the discrete version is defined as the pattern spectrum divided by the cardinality of original image. Another variety called *probability distribution function* [20] is defined as one minus the pecstrum.

Shih and Pu [88] presented a useful morphological shape description tool called *geometric spectrum* or *G-spectrum*, for quantifying the geometric features on multidimensional binary images. The G-spectrum which is not only useful in shape description but also shape recognition is a measurement for quantifying the geometric shape of discrete multidimensional images.

1.2 Introduction of Mathematical Morphology and Notation

The notations used in this dissertation are introduced as follows. Let X be a set in a multidimensional Euclidean space which represents a binary image and $Card(X)$ be the cardinality of X (i.e., the total number of elements in the set X). The cardinality of a one-dimensional set is the length, of a two-dimensional set is the area, of a three-dimensional set is the volume, and so forth. Let $X \ominus Y$, $X \oplus Y$, $X \circ Y$, and $X \bullet Y$ denote the morphological erosion, dilation, opening, and closing, respectively. The set difference is denoted by $X - Y$.

A brief introduction of mathematic morphology is going to give as followed. The simplest morphological operations are the *erosion* and *dilation*. The dilation of a set A by a set of structuring elements B is

$$A \oplus B = \{a + b \mid a \in A,\ b \in B\} = \bigcup_{b \in B} A_b. \tag{1.2.1}$$

where A_b is the translation of a set A by a vector b:

$$A_b = \{z \mid z = a + b,\ a \in A,\ z \in E\}. \tag{1.2.2}$$

An example of dilation is given in Fig. 1.1.

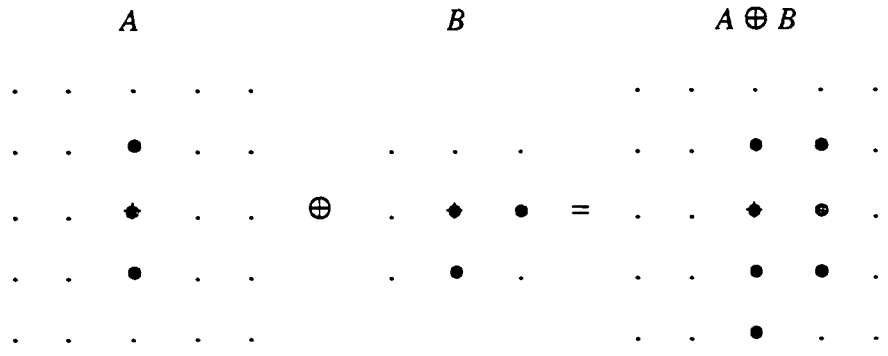


Fig. 1.1 Example of dilation of set A by structuring elements B.

The erosion of a set A by a set of structuring elements B is

$$A \ominus B = \{a \mid a+b \in A,\ b \in B\} = \bigcap_{b \in B} A_b. \tag{1.2.3}$$

An example of erosion is given in Fig. 1.2.

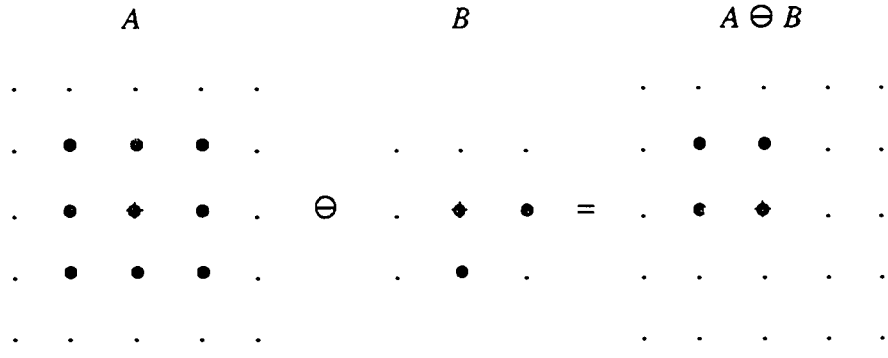


Fig. 1.2 Example of erosion of set A by structuring elements B.

Dilation and erosion posses several interesting properties. They are given as followed but without proofs. Interested readers would refer the references [75, 76, 77, 18, 67 Chapter 6, 34].

(1) The dilation and erosion are dual operations:

$$A \oplus B = (A^c \ominus B)^c$$

$$A \ominus B = (A^c \oplus B)^c \tag{1.2.4}$$

(2) Commutative property:

$$A \oplus B = B \oplus A \tag{1.2.5}$$

But erosion won't posses this property that is $A \ominus B \neq B \ominus A$.

(3) Associative property:

$$A \oplus (B \oplus C) = (A \oplus B) \oplus C \tag{1.2.6}$$

(4) Translation invariance

$$A_z \oplus B = (A \oplus B)_z \tag{1.2.7}$$

$$A \oplus B_z = (A \oplus B)_z \tag{1.2.8}$$

$$A_z \ominus B = (A \ominus B)_z \tag{1.2.9}$$

$$A \ominus B_z = (A \ominus B)_z \tag{1.2.10}$$

(5) Increasing property

$$A \subseteq B \Rightarrow A \oplus C \subseteq B \oplus C \tag{1.2.11}$$

$$A \subseteq B \Rightarrow A \ominus C \subseteq B \ominus C \tag{1.2.12}$$

(6) Distributive property:

$$(A \cup B) \oplus C = (A \oplus C) \cup (B \oplus C) \tag{1.2.13}$$

$$A \oplus (B \cup C) = (A \oplus B) \cup (A \oplus C) \tag{1.2.14}$$

$$A \ominus (B \cup C) = (A \ominus B) \cap (A \ominus C) \tag{1.2.15}$$

$$(B \cap C) \ominus A = (B \ominus A) \cap (C \ominus A) \tag{1.2.16}$$

Another two important operations are *opening* and *closing*.

$$A \circ B = (A \ominus B) \oplus B \tag{1.2.17}$$

$$A \bullet B = (A \oplus B) \ominus B \tag{1.2.18}$$

Examples of opening and closing are given as follows.

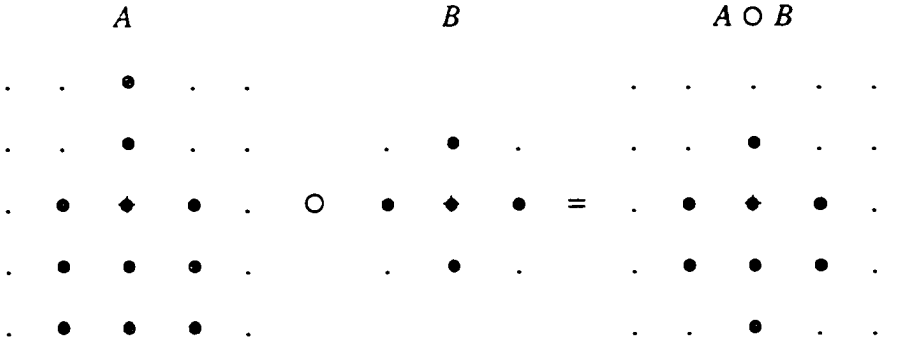


Fig. 1.3 Example of opening of set A by structuring elements B.

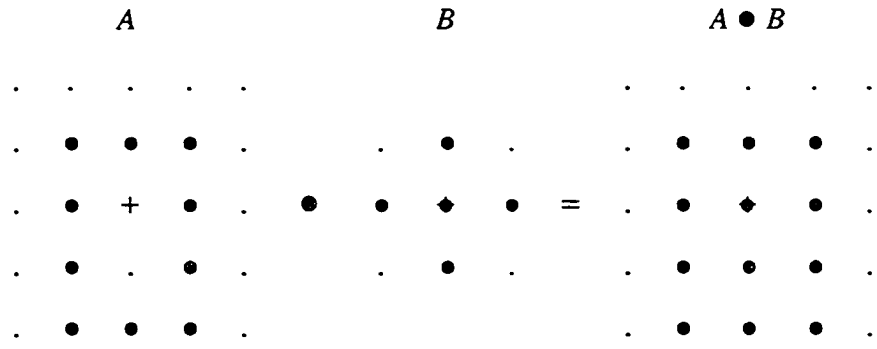


Fig. 1.4 Example of closing of set A by structuring elements B.

The notions and morphological transformations of a binary image can also be extended to gray-scale images. Let f and k be the gray-scale image and structuring elements. The gray-scale morphological dilation, erosion, opening, and closing are defined as follows, where $z \in f$ and $x-z \in k$.

$$f \oplus k = \max\{f(z) + k(x-z)\} \tag{1.2.19}$$

$$f \ominus k = \min\{f(z) - k(x-z)\} \tag{1.2.20}$$

$$f \circ k = (f \ominus k) \oplus k \tag{1.2.21}$$

$$f \bullet k = (f \oplus k) \ominus k \tag{1.2.22}$$

Interested readers would refer [75, 76, 77] for more in detail about gray-scale morphology and its properties.

1.3 Useful Morphology Applications

In this section, we give examples of useful applications based upon morphological operations. Given a gray-level image $f(m, n)$ and a small 2-D symmetric structuring set K containing the origin.

A. Morphological Edge Gradient

- Erosion Gradient:

$$EG(f) = f - (f \ominus k) \tag{1.2.23}$$

- Dilation Gradient:

$$DG(f) = (f \oplus k) - f \tag{1.2.24}$$

- Morphological Gradient:

$$MG(f) = (f \oplus k) - (f \ominus k) \tag{1.2.25}$$

- Morphological Edge-strength Operator:

$$EO(f) = \min\ [EG(f), DG(f)] \tag{1.2.26}$$

B. Peak and Valley Extraction

During the peak extraction, the peaks that can not contain K remains, while the rest get eliminated.

- Peak Extraction

$$PE(f) = f - (f \circ K) \tag{1.2.27}$$

- Valley Extraction

$$VE(f) = (f \bullet K) - f \tag{1.2.28}$$

C. Skeletonization

A morphological skeleton for a gray-level image f is denoted by $SK(f)$ and expressed as follows.

$$SK(f) = \sum_{n=0}^{N} SK_n(f) \tag{1.2.29}$$

where $N = \max \{ n \mid f \ominus nB \neq 0 \}$. The nth skeleton component of f is

$$SK_n(f) = (f \ominus nB) - [(f \ominus nB) \circ B] \quad 0 \leq n \leq N \tag{1.2.30}$$

where $nB = B \oplus B \oplus B \cdots \oplus B$ (n times).

1.4 Organization of This Dissertation

The organization of this dissertation will be given in this section. The outline of this dissertation is as follows.

Chapter 1 Introduction

Chapter 2 A maxima-tracking method for skeletonization

Chapter 3 The roots of morphological filters

Chapter 4 Geometric spectrum and shape description

Chapter 5 Morphological shape recognition using G-spectrum

Chapter 6 Soft mathematical morphology and its properties

Chapter 7 Gray-scale soft mathematical morphology

Chapter 8 Summary and future research

The brief statements of each chapter is also given which will be the abstract of the topic discussed in each chapter.

In chapter 2, the skeletonization algorithm based on the Euclidean distance function

using the sequential maxima-tracking method is described. When applied to a connected image, this method generates a connected skeleton composed of simple digital arcs. With a slight modification, the algorithm can preserve the more important features in the skeleton by eliminating nonsignificant short skeletal branches which touch the object boundary at corners. Therefore its application to shape recognition can be achieved easier.

In chapter 3, a morphological filter in gray-scale applications is a nonlinear digital filter which consists of a bounded mask (called the structuring element) with predefined values that slides over a signal of finite length. For each input sample, the output is the minimum or maximum of all neighboring input samples in the mask offset by their corresponding mask values centered at that input sample. If a finite signal that ends iteratively with a structuring element, converges to an invariant signal in a finite number of passes, such an invariant signal is called a *root* for the structuring element. A new concept to reach the roots called back-propagation morphology is defined. By using this new concept, a two-scan algorithm is developed to reach the roots for only two scans.

In chapter 4, a useful morphological shape description tool called *geometric spectrum* or *G-spectrum* is presented, for quantifying the geometric features on multidimensional binary images. The basis of this tool relies upon the cardinality of a set of non-overlapping segments in an image using morphological operations. The G-spectrum preserves the translation invariance property. With a chosen set of isotropic structuring elements the G-spectrum also preserves the rotation invariance. After the procedure of normalization, the G-spectrum can also preserve the scaling invariance. The properties and proofs of the G-spectrum are discussed.

In chapter 5, the shape recognition algorithm using G-spectrum is presented. Experimental results are shown to be satisfied. The shape likeliness measurement which is based upon G-spectrum is defined to measure the likeliness between shapes in order to

identify the two shapes that are the most alike. The less of shape likeness measurement is, the more these two shape are alike.

In chapter 6 the generation of soft morphological operations was motivated by Koskinen, Astola, and Neuvo [44, 45]. The soft morphological operations are less sensitive to additive noise and to small variations. New definitions of binary soft morphological operations are given. The properties and proofs of soft morphological operations are studied and discussed. It also shows that soft morphological operations commute with thresholding and obey threshold-linear superposition. In general, soft closing and soft opening are not idempotent operations, but under some constraint the soft operations can be idempotent and the proof is given. The properties of *idempotent soft morphological filters* will be studied and discussed.

In chapter 7 gray-scale soft mathematical morphology is the extension of binary soft mathematical morphology which is found to be less sensitive to additive noise and to small variations. In this chapter, binary soft morphological operations are reviewed and the definitions of gray-scale soft morphological operations are given. The properties of gray-scale soft morphological operations are developed. It has been shown that the soft morphological operations of functions by sets or functions can commute with thresholding and obey threshold superposition. The threshold superposition property allows gray-scale signals to be decomposed into multiple binary signals which can be processed by only logic gates in parallel and then the binary results can be combined to produce the equivalent output.

In chapter 8 we state the further research using morphological operations applied in image processing and analysis and give summary of this dissertation.

CHAPTER II

A MAXIMA-TRACKING METHOD FOR SKELETONIZATION

In this chapter, the skeletonization algorithm based upon the Euclidean distance function is described. This algorithm uses maxima-tracking which will generate a connected skeleton. A modified algorithm is proposed to eliminate non-significant short branches.

2.1 Introduction

The study of distance transformation and skeletonization was motivated by the need for methods of converting a digital picture into a linear form in a near natural manner. From the skeleton, the contour of the figure can be easily regenerated, so the amount of information involved is the same. However, the skeleton seems to emphasize some properties of the picture; for instance, curvature properties of the contour correspond to topological properties of the skeleton. The concept of a skeleton was first proposed by Blum [4]. Skeleton, medial axes, or symmetrical axes have been extensively used for characterizing the objects satisfactorily by structures composed of line or arc patterns. Applications include the representation and recognition for handwritten or printed characters, fingerprint ridge patterns, chromosomes and biological cell structures, circuit diagrams and engineering drawings, and the like.

If we regard a connected object as grass and tend a fire starting from its contour, assume that this fire burning spreads uniformly in all directions but in such a way that the waves generated do not flow through each other. The *skeleton* (or *medial axis*, abbreviated MA) is where waves collide with each other in a frontal or circular manner [4]. It has this name because the centers are located at midpoints, or along local symmetrical axes of the region. The points in the MA bear the same distance from at least two

contour points and this distance is the smallest among those computed from all background this distance is the smallest among those computed from all background points (equidistance property).

Information on the size of all foreground points is retained by associating each point of the transformed image (MA) with a label representing its distance value from the background. As a consequence, it is possible to recover the original binary image as the union of the circular neighborhoods centered at the skeleton points and having a radius equal to the associated labels (i.e. reverse transformation). A skeleton will not immediately appear in the wave-front if it is a smooth curve locally. The appearance of a skeleton starts at the minimum radius of curvature of the contour. The disappearance of a skeleton represents the largest circle that can be drawn in the local region. If the boundary has a regional concave contour, the skeleton will be generated only by those points of the contour that are not on the convex hull of the boundary. For those stimuli on convex hull, the wave-fronts will not collide with each other. Thus, it will not generate a skeleton within this area. In fact, the skeleton will be located outside the regional concave contour if the skeletonization is dealing with the background of the connected object, i.e., the complement of the picture.

Many algorithms have been developed to extract the skeleton. Generally speaking, the procedure accomplishing a transformation of this kind involves an iterative process which shrinks the input image step-by-step until a one-element thick figure is obtained [65]. An algorithm was proposed [63] which practically utilizes the strategy: visiting all the pixels in the bitmap to iteratively delete the edge points which are classified as non-safe points (the points being deleted without the effectness of connectivity). Zhang and Suen [109] proposed a parallel thinning algorithm which consists of two subiterations: one is to delete the south-east boundary points and the north-west corner points and the other is to delete the north-west boundary points and the south-east corner points.

Lu and Wang [49] modified Zhang and Suen's algorithm to preserve the 2-pixel wide diagonal lines. Holt *et al.* [41] made some improvement to preserve the 2×2 square patterns. Hall [31] evaluated Holt's improved algorithm and proposed another improvement to the preservation of certain diagonal lines. Guo and Hall [30] presented two parallel thinning algorithms each using two-subiteration approaches: one algorithm is to alternatively delete the top-to-bottom right-to-left boundary points, and then bottom-to-top left-to-right ones, and the other algorithm is to apply a thinning operator to one of the diagonal and side subfields. Kwok [46] proposed the thinning algorithm by contour generation in terms of a chain code. The advantage of this algorithm is that the contour tracking is performed only once during iterations and only edge points are visited. Arcelli and Sanniti di Baja [2] used the city-block distance transform instead of the original object to detect the skeleton. Rather than peeled, the skeleton are identified as multiple pixels based on defined multiplicity conditions, and recursive procedure calls are applied and regarded as a one-pass two-operation process to detect those multiple pixels. Maragos and Schafer [56] utilized morphological operators to extract the skeleton and to optimize the skeleton for image coding. However, their skeletons are not connected and have more than one-pixel in width.

This chapter is organized as followed. In Section 2.2, an example of thick skeleton generation is presented. In Section 2.3, related definitions such as base points, apex points, directional-uphill generation, and directional-downhill generation are given to illustrate our approach. In Section 2.4, the new algorithm is proposed and its connectivity properties are proved. In Section 2.5, a modified algorithm is described to eliminate non-significant skeletal branches. At last section, we summarize this chapter.

2.2 An Example of Thick Skeleton Generation

For each pixel P in an image, let us associate a set of disks or circles with various radii centered at P. Let C_P be the largest disk which is completely contained in the

foreground, and let r_P be the radius of C_P. There may exist other pixels Q such that C_Q contains C_P. If no such Q exists, D_P is called a *maximal disk*. The set of centers and radii (i.e. Euclidean distance value) of the maximal inscribed disks is called the *Maximal-Disk-Skeleton*, abbreviated MD-Skeleton.

A. The Skeleton from Distance Function

We define a function at each point P of the foreground S whose value is the smallest distance of P from the background $\bar{S}$:

$$d(P) = \min_{Q \in \bar{S}} \{d(P, Q)\}, \tag{2.2.1}$$

where $d(P, Q)$ is the Euclidean distance between two points P and Q. Let us visualize the distance function as the altitude on a surface; the "ridges" of the surface namely the points where we cannot define a tangent plane, together with the corresponding distances, constitute the skeleton. It can be easily derived that a point P belongs to the skeleton of a binary image S, if and only if, the maximal disk with the radius $d(P)$ hits the boundary of S at two or more different places. Let us define the set

$$A(P) = \{Q \mid d(P, Q) = d(P),\ Q \in \bar{S}\}. \tag{2.2.2}$$

If A(P) contains more than one point, P is a skeleton point.

B. Detection of the MD-Skeleton and the Ridge Points

From above discussion, the set of all pixels associated with their distance values, having no neighboring points of higher altitude within a local 3×3 window, is the MD-Skeleton which is completely disconnected. In order to achieve connectedness, the following defined ridge points are added into the MD-Skeleton. The *ridge* points are all points, having the altitude higher than their neighboring two pixels in any one of the horizontal,

vertical, 45°-diagonal, and 125°-diagonal directions. Note that the MD-Skeleton is the set of pixels having the altitude higher than or equal to their 8 neighbors; however the ridge points require strictly higher altitude. Fig. 2.1 represents the Euclidean distance function with both MD-Skeleton and ridge points underlined. It has to be noted that the result is still not connected.

C. Trivial Uphill Generation

The trivial uphill of a point P is the set of all neighbors with higher altitude. Fig. 2.2 shows what happens if we initially add the uphill of the MD-Skeleton and ridge points, and continuously add the uphill of the new skeleton points until no further uphill can be generated. We could obtain a connected skeleton but a very thick one. In order to obtain a skeleton with the necessary thinness, one has to take into account the cases of the directional-neighborhood which will be discussed in the next section.

2.3 An Illustration of Our Approach

According to eq. (2.2.2), the following defined base and apex points which satisfy the condition that $A(P)$ contains more than one point, are the skeleton points.

A. Base Points

The base points are those corner points which have the distance value 1 and are surrounded by the majority 0's. They belong to one of the following three configurations which can vary up to eight 45°-rotations, respectively:

1	0	0
1	<u>1</u>	0
0	0	0

1	1	0
1	<u>1</u>	0
0	0	0

1	1	1
1	<u>1</u>	0
0	0	0

where the central underlined pixels represent the 45°-, 90°-, and 125°-corner points,

respectively, and the labeled "1" indicates a foreground pixel and "0" a background pixel. In Euclidean Geometry, the corner point can have a great deal of degree. However in digital image data, only the above three degrees may exist within a local 3×3 window.

It is easy to see that if all three configurations are considered as base points, more base points will induce more nonsignificant short branches of the skeleton. That is to say that an approach, according to which branches are originated from each corner point in correspondence with every convexity indiscriminately, would lead to unmanageable complexity in the skeleton structure. Therefore, the number of major or minor branches of a skeleton should reach a compromise among the representativity of the connected object structure, the required reversibility and the cost of detecting the significant branches and deleting the noisy branches.

In this chapter, detecting strong curvature has led us to consider sharp convexities as more significant. In more detail, if the amplitude of the angle formed by two intersecting wave-fronts of the fireline is viewed as the parameter characterizing the sharpness, the values smaller than or equal to 90° are identified as detectable convexities. More strictly in getting rid of minor branches, even only 45°-convexities are detected. Using these base points as the source to grow up the skeleton, the remaining elements are acquired to preserve the skeletal connectedness.

B. Apex Points

The apex points are those points being the local maxima in their 3×3 neighborhood. It has to be noted that the apex points are in fact the MD-Skeleton mentioned in Section 2; the reason for renaming is because the local maximum pixels only construct a small portion of and a very disconnected skeleton. From a geometrical point of view, that the local maxima could be either 45°-corner points or interior elements having the highest altitude in the local neighborhood. The 45°-corner points, since their distance values are 1's, have been detected as the base points. Some of the apex points may have the highest altitude

not only in a local 3×3 neighborhood, but also in a larger neighborhood. The base and the apex points are considered as sources or elementary cells of the skeleton and grow up out of them to be 8-connected. Fig. 2.3 represents again the Euclidean distance function with the base points labeled "**B**" and the apex points underlined.

C. Directional-Uphill Generation

The set of points $\{P_i\}$ is called the *directional-neighborhood* of P, i.e., $D_P = \{P_i\}$, if they are in the 8-neighborhood of P and are also located within the ±45° slope change of the local medial axis direction of P. For example, using the neighborhood labeled $P_1, P_2, \cdots$, and P_8 counterclockwise from the positive x-axis of P, if P_7 and P are the skeleton points, the points P_2, P_3, and P_4 are the directional-neighbors of P, i.e., $D_P = \{P_2, P_3, P_4\}$. The case and some other cases are illustrated below.

$$\begin{array}{ccc} P_4 & P_3 & P_2 \\ . & P & . \\ . & P_7 & . \end{array} \qquad \begin{array}{ccc} . & . & P_2 \\ P_5 & P & P_1 \\ . & . & P_8 \end{array}$$

$$\begin{array}{ccc} P_4 & P_3 & . \\ P_5 & P & . \\ . & . & P_8 \end{array} \qquad \begin{array}{ccc} . & P_3 & P_2 \\ . & P & P_1 \\ P_6 & . & . \end{array}$$

It has to be noted that the directional-neighborhood always contains three points. The directional-uphill generation is to add these points which are the maxima of the central point P plus its directional-neighborhood. That is to say

$$P_{next}^{U} = \max_{P_i \in D_P \cup \{P\}} \{P_i\}. \tag{2.3.1}$$

Fig. 2.4 shows the result of directional-uphill generation of Fig. 2.3.

D. Directional-Downhill Generation

From Fig. 2.4, it can be easily seen that there must exist a vertical path of the skeleton between two apex points underlined on the left side, in which the altitude changes are not always increasing; instead they are mixture of decreasing and increasing. The directional-downhill generation which is similar to the directional-uphill generation except the maxima tracking of the set excluding the central point P, is used to produce this type of skeletal branch. That is to say

$$P_{next}^{D} = \max_{P_i \in D_P} \{P_i\}. \tag{2.3.2}$$

The directional-downhill generation is initialized from the apex points which are the local maxima. Hence the altitude of the next directional-downhill should be lower. However, the tracking procedure is continued without taking into account a comparison of neighbors with the central point; the next directional-downhill altitude could be lower or even higher. Fig. 2.5 shows the result of directional-downhill generation of Fig. 2.4. The skeleton is now connected and single-pixel wide except a few two pixels with the same Euclidean distance.

2.4 The Algorithm and the Connectivity Properties Proof

The maxima tracking algorithm traces the skeleton points by choosing the local maxima on the Euclidean distance transform in terms of the least slope change in medial axes. The algorithm is described as follows:

1) The base points and the apex points are detected as the initial skeleton points.
2) Starting from these initial skeleton points, the directional-uphill generation in eq. (2.3.1) is used to add continuously more skeleton points, and the points which can not be further tracked, are marked (The marked points are always apex points).

3) Starting from the marked points, the directional-downhill generation in eq. (2.3.2) is used to complete the skeleton tracking.

Fig. 2.6. illustrates the resulting skeletons by using the above algorithm. In this chapter, the 8-connectedness is applied on the foreground, therefore the 4-connectedness for the background must be used to be consistent [72]. There are three fundamental connectivity properties which must be considered:

C1) After skeletonization, an originally connected object should not be disconnected into two or more sub-objects.

C2) After skeletonization, a connected object should not disappear at all.

C3) After skeletonization, these originally disconnected background components should not be 4-connected.

The proof for the above three properties is given as follows:

Proposition 2.1: The maxima-tracking skeletonization will not disconnect an originally connected object into two or more sub-objects.

[(*Proof*)]:

Induction hypothesis: For any m ($m < n$) marked apex points of an object, the skeleton obtained will preserve the property C1, where n could be any number.

The base case is $m = 1$. The algorithm starts tracking from each of the base and the apex points, and it will stop when the current point is connected with another skeleton point or touches with the point marked. That means every subskeleton branch starting from the base point will connect with some skeleton point or meet with other branch at the marked apex point.

When $m = 2$, it is regarded as two sub-objects, each sub-object contains a marked apex point. As discussed above, each sub-object is 8-connected. The algorithm will trace from each marked-apex and stop when it is connected with some skeleton point

contained in the other skeleton subset. The reason that it would not connect with a skeleton point which belongs to the same skeleton subset is the directional-neighbors we use. Using directional-neighbors will enforce the tracking direction to go towards the region which has not been tracked up to now. Finally, it will lead tracking toward another skeleton subset.

As the induction hypothesis claim, when $m = n$, the skeleton obtained is 8-connected. Now consider the case when $m = n+1$. Those $n+1$ marked-apex points can be regarded as two sub-objects: one contains n marked-apex points and the other contains only a marked-apex (i.e. the $(n+1)th$ point). As discussed above, the one with n marked-apex points is 8-connected, and the other with a marked-apex is also 8-connected. Tracking from the $(n+1)th$ marked-apex, it will lead the tracking to go towards the skeleton subset with n marked-apex points. That means the whole object is 8-connected. □

Proposition 2.2: The maxima-tracking skeletonization will not make an originally connected object disappeared at all.

[(*Proof*)]:

Tracking starts from each base and each apex point, the algorithm marks base points as the skeleton points. For any size of an object, there must be at least one base point or at least one apex point, and the skeleton obtained must have at least one point. For the ideal case of a circle, it there is no base point but one apex point is present which is the local maximum representing the center of the circle. That is after skeletonization, an 8-connected object will not disappear at all, and the skeleton at least contains a pixel. □

Proposition 2.3: The maxima-tracking skeletonization will not let the originally disconnected background components 4-connected.

[(*Proof*)]:

According to the definition of the apex points, there exists at least one apex in the object between any two background components. As the algorithm is performed, it will

trace from each apex point which is not 8-connected to any skeleton point. After tracking, there will be one skeleton branch that will make these two background components disconnected. Therefore the skeleton obtained will not allow the originally disconnected background components to be 4-connected. □

The algorithm including three procedures and two functions are described in pseudo-codes as follows:

```
procedure SKEPIK
    /* trace starting from each base and each apex point */
    for  i, j in 1 ··· N, 1 ··· N  loop
        if  BASE_APEX ( p (i,j) )  then
            UpHill-Generation ( p (i, j), p (i, j) );
        end if
    end loop

    /* trace from each marked apex point in the procedure UpHill-Generation */
    while  ( marked pixel )
        DownHill-Generation ( marked_apex, marked_apex);
    end while
end SKEPIK

function  APEX ( p ) : Boolean
/* apex point is the local maximum point */
    if  ( p is local maxima )  then
        return TRUE;
    else
        return FALSE;
    end APEX

function  BASE_APEX ( p ) : Boolean
/* base point is the point with distance 1 and has 4 or more zeros in its 8-neighborhood */
    if  ( distance of p = 1 )  then
        find 8-neighbors of p;
        if  ( number of 8-neighbors with distance 0 ) ≥ 4  then
            return TRUE;
        else
            if  APEX ( p )  then
                return TRUE;
            else
                return FALSE;
            end if
        end if
    end if
end BASE_APEX
```

```
procedure  UpHill-Generation ( current-skeleton-pixel, previous-skeleton-pixel )
      if  ( number of maximum in 8-neighbors > 1 and
          distance of maximum ≥ distance of current-skeleton-pixel )  then
            maximum-pixel = the maximum of the directional-neighbors;
            UpHill-Generation ( maximum-pixel, current-skeleton-pixel);
      else
            if  ( number of maximum in 8-neighbors = 1 and
                distance of maximum ≥ distance of current-skeleton-pixel )  then
                  maximum-pixel = the maximum;
                  UpHill-Generation ( maximum-pixel, current-skeleton-pixel );
            else
                  mark-apex; /* mark current-skeleton-pixel for later processing */
            end if
      end if
end UpHill-Generation

procedure  DownHill-Generation ( current-skeleton-pixel, previous-skeleton-pixel )
      maximum-pixel = the maximum of the directional-neighbors;
      if  ( maximum-pixel is not a skeleton point )  then
            DownHill-Generation ( maximum-pixel, current-skeleton-pixel );
      end if
end DownHill-Generation
```

2.5 A Modified Algorithm

Skeleton points can have a velocity associated with them. This velocity is determined by the angle formed — the sharper the angle, the faster the velocity. In such circumstance, an accelerating point will generate from the high curvature point toward this skeletal locus. On the other hand, a decelerating point would result in the reverse direction. Whenever the skeleton is a straight line, the velocity is a constant.

The modified maxima tracking algorithm will obtain the skeleton by eliminating nonsignificant short skeletal branches which touch the object boundary at corners. It is different from the previous algorithm that detects the base and the apex points as initial skeleton points. Instead the maxima tracking starts from the apex points only. The algorithm will recursively repeat the same procedure which selects the maxima in the directional-neighborhood as the next skeleton point until another apex points are reached. further skeleton point.

The modified maxima-tracking algorithm is given as follows:

1) The apex points are detected as the initial skeleton points.
2) Starting from each apex point, we use the directional-uphill generation to generate the skeleton points. Recursively repeat this procedure until an apex point is reached, and then the apex point is marked.
3) Starting from those marked-apex points, the directional-downhill generation is used to track the skeleton points.

Fig. 2.7 illustrates the resulting skeletons by using the modified algorithm. The result could lead us to achieve the rule-based pattern recognition and shape decomposition. As shown in Fig. 2.7a, a skeleton of a straight line can be interpreted as a rectangular shape. In Fig. 2.7b, a triangle can be represented as a staircase. The distance value of the toppest stair is the radius of the largest enclosed circle in this triangle. Fig. 2.7c is a airplane-like shape which is composed of three rectangles. In Fig. 2.7d, there is a closed curve which indicates a hole inside. The skeleton in Fig. 2.7e can be represented by a bent curve, which gives us the information that there is a notch. Fig. 2.8 shows the skeleton of a wrench. The character "e" and its rotations by 30°, 45°, and 90° shown in Fig. 2.9, demonstrates that the modified algorithm will produce the output independent of the orientations of the input patterns, provided the digitization error is disregarded.

2.6 Summary

Obtaining the skeleton of a binary set in three steps or two steps is of course of great practical value. It is perhaps even of higher theoretical value. The clear identification of the various articulations in the skeleton generation facilitates the adaptation of the skeleton algorithm to other situations, i.e. conditional of geodesic skeletons, skeletons on square raster, skeletons of functions. The Euclidean distance transform whose attributes to the skeleton have the most equidistance property analogy with the definition for

continuous images and possesses the rotation invariance property. This will help a lot in object recognition not only for simple skeletons but also for the skeletons having distance information with them.

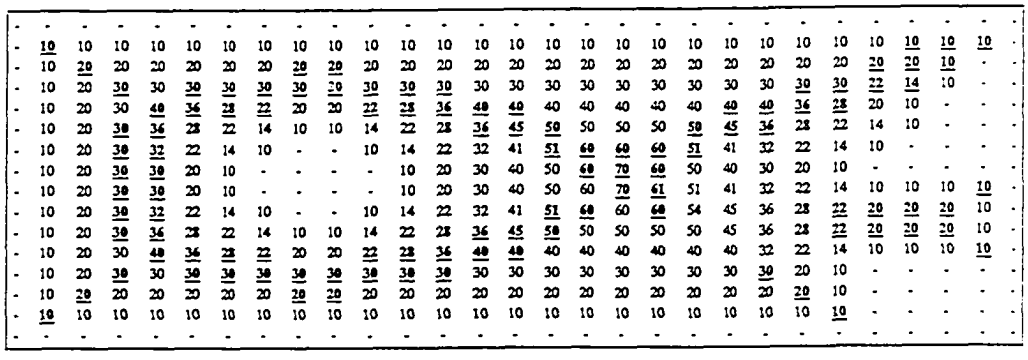


Fig. 2.1 The Euclidean distance with both MD-Skeleton and ridge points underlined.

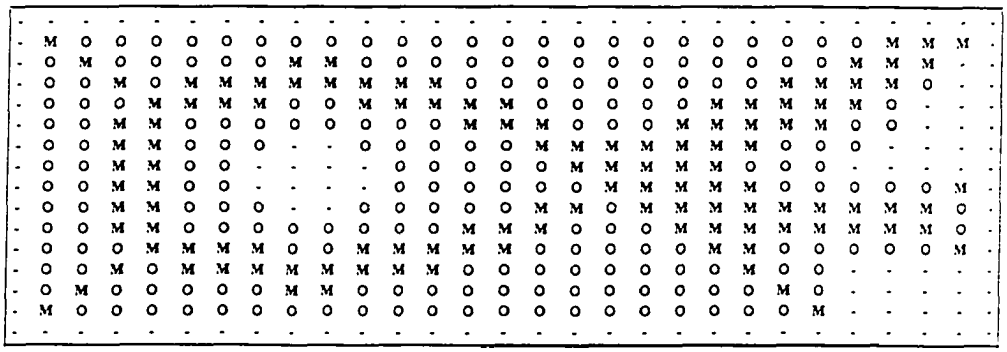


Fig. 2.2 The uphill generation of the MD-Skeleton and ridge points.

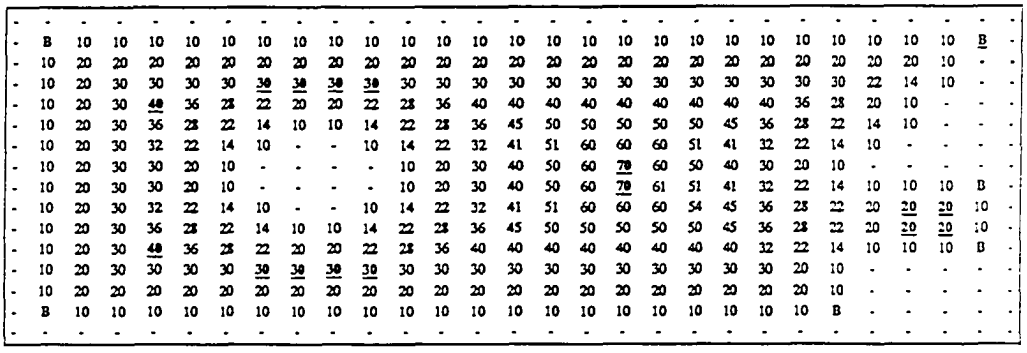


Fig. 2.3 The Euclidean distance with base points labeled **B** and apex points underlined.

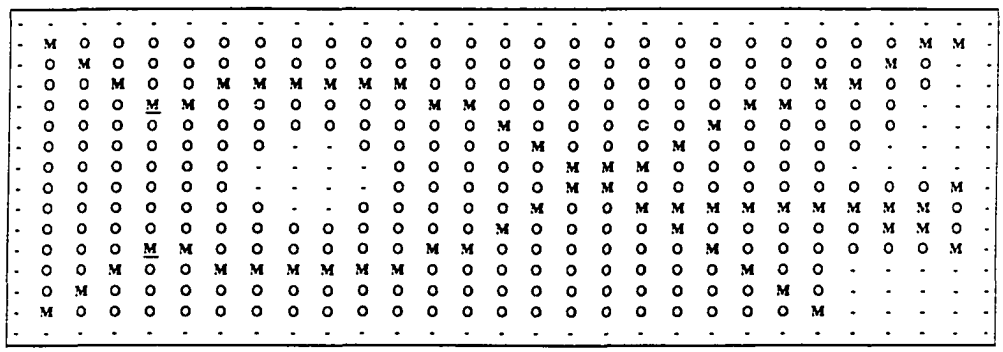


Fig. 2.4 The directional-uphill generation.

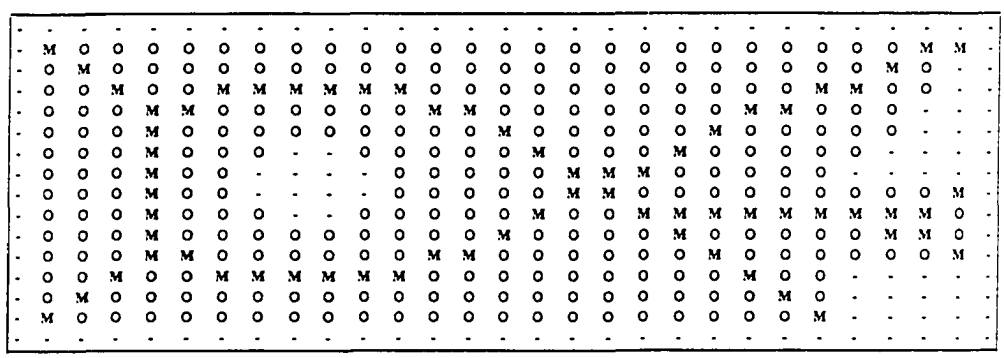


Fig. 2.5 The directional-downhill generation of Fig. 2.4.

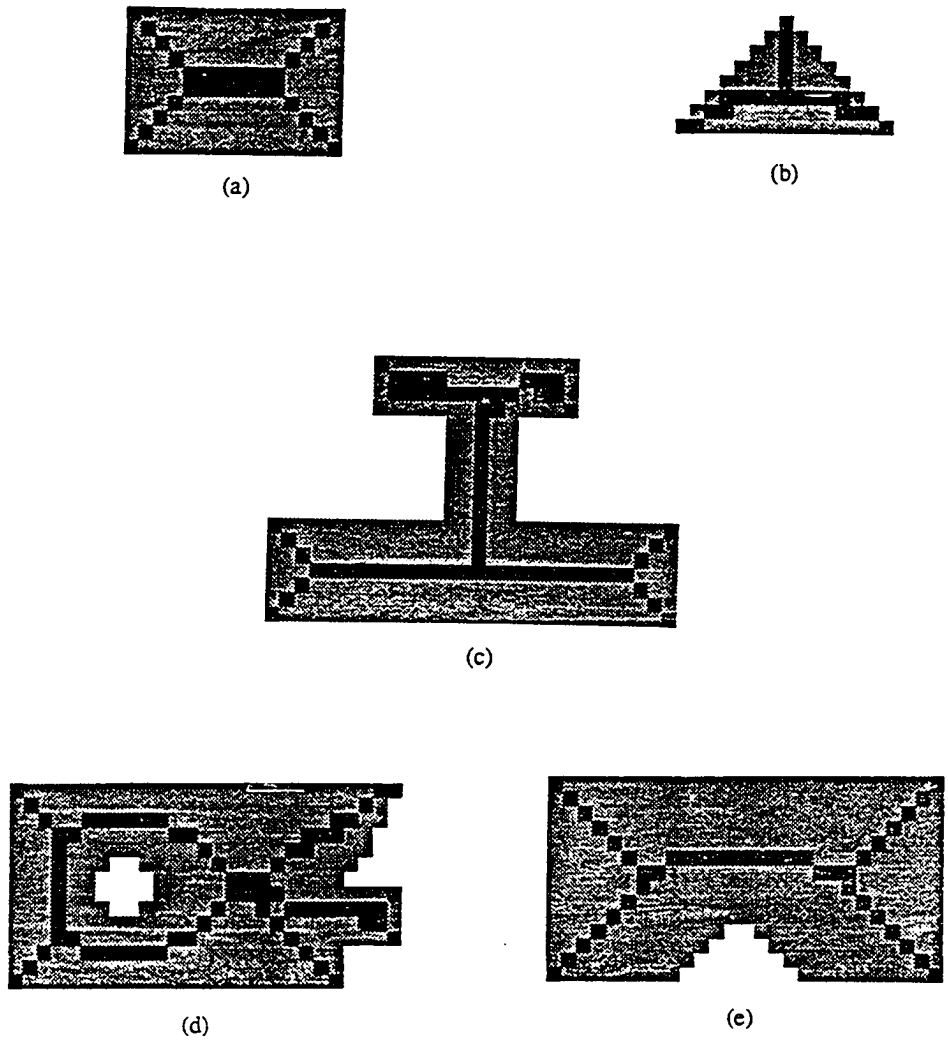


Fig. 2.6 The resulting skeleton by using the algorithm described in section 2.4.

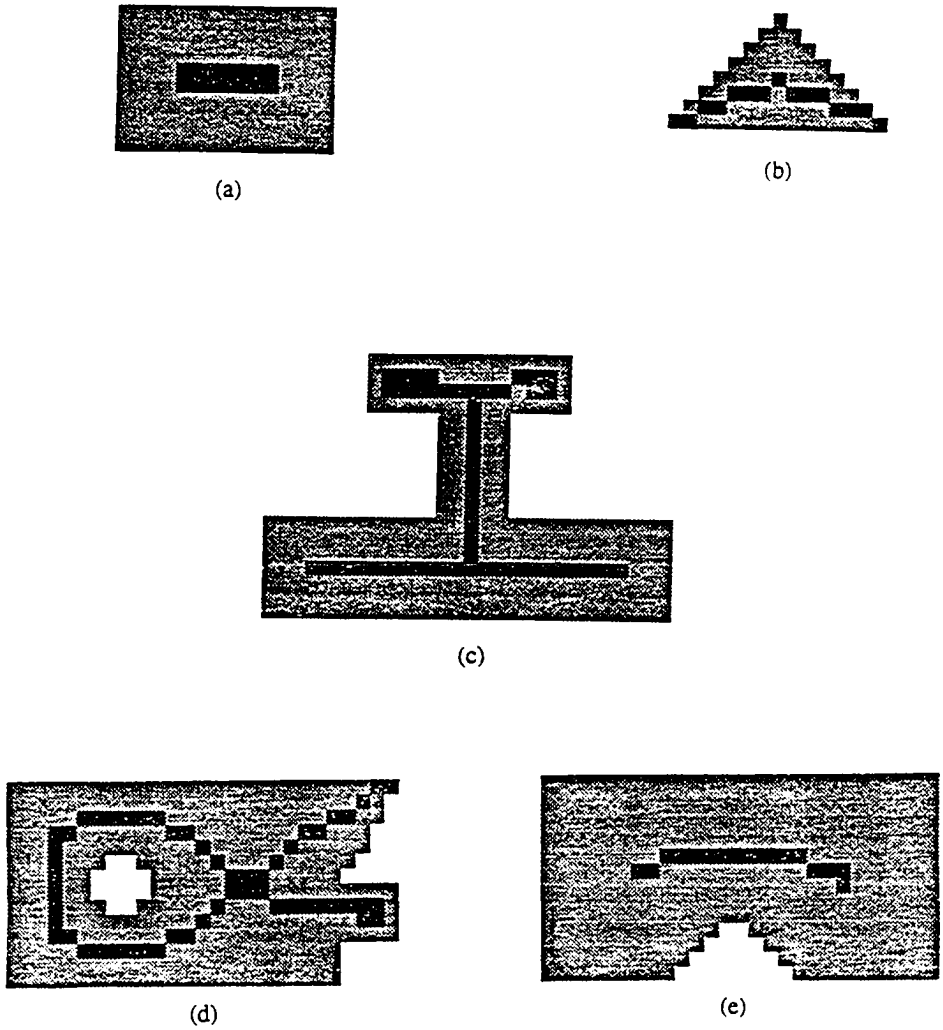


Fig. 2.7 The resulting skeleton by using the modified algorithm.

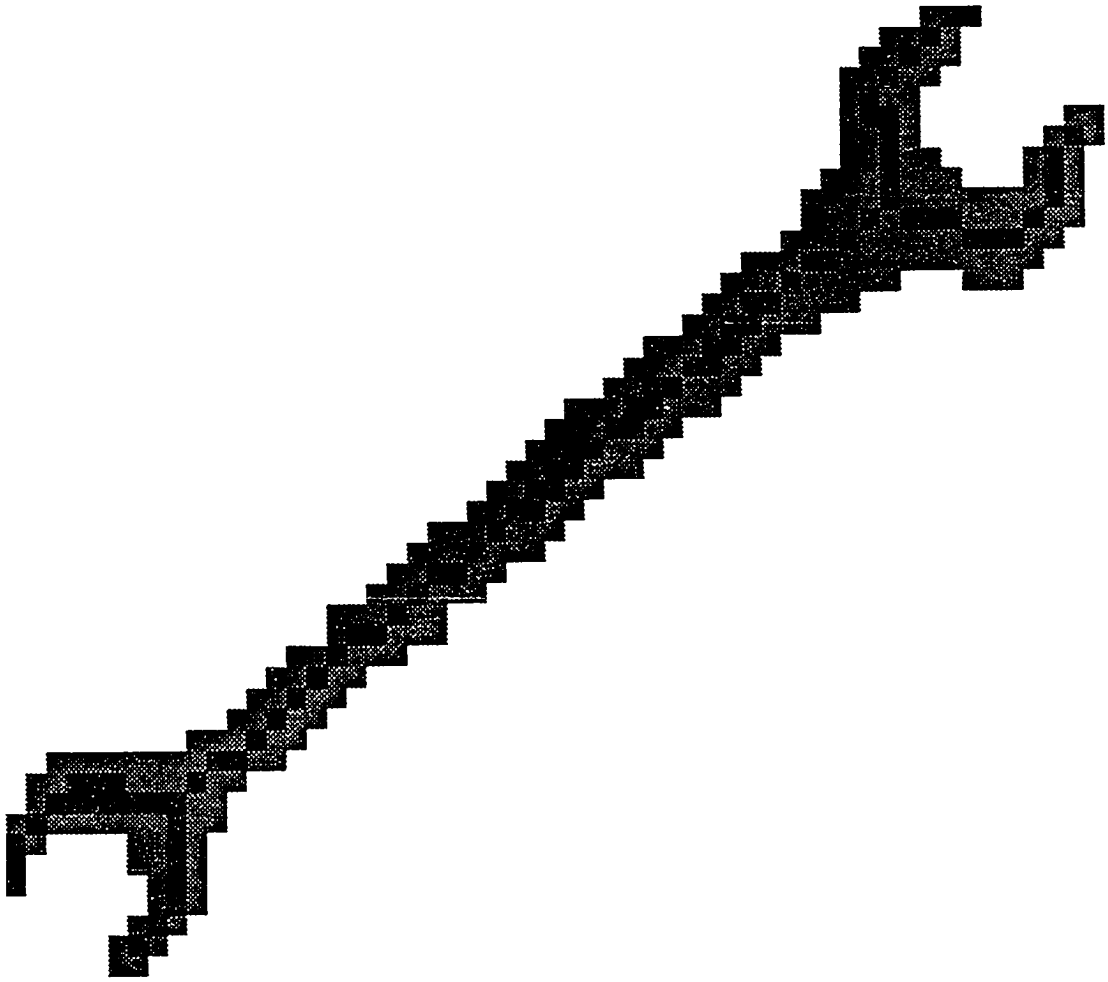
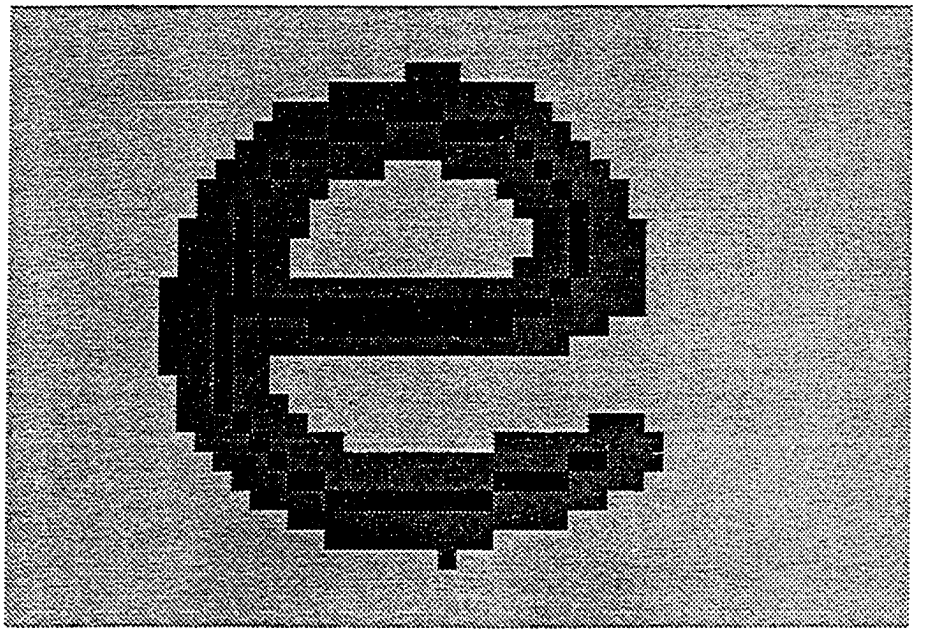
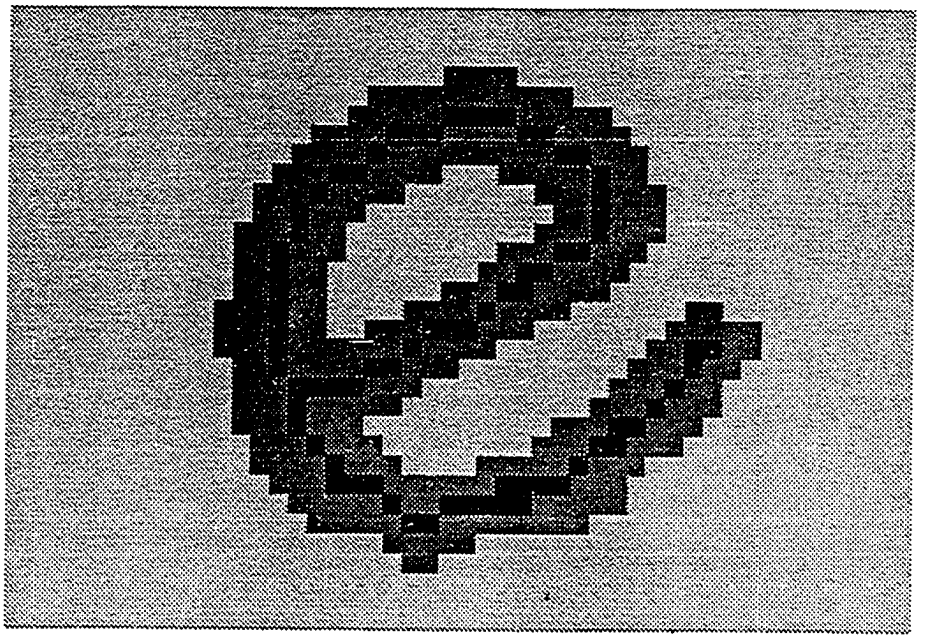


Fig. 2.8 An example of the skeleton for a wrench.

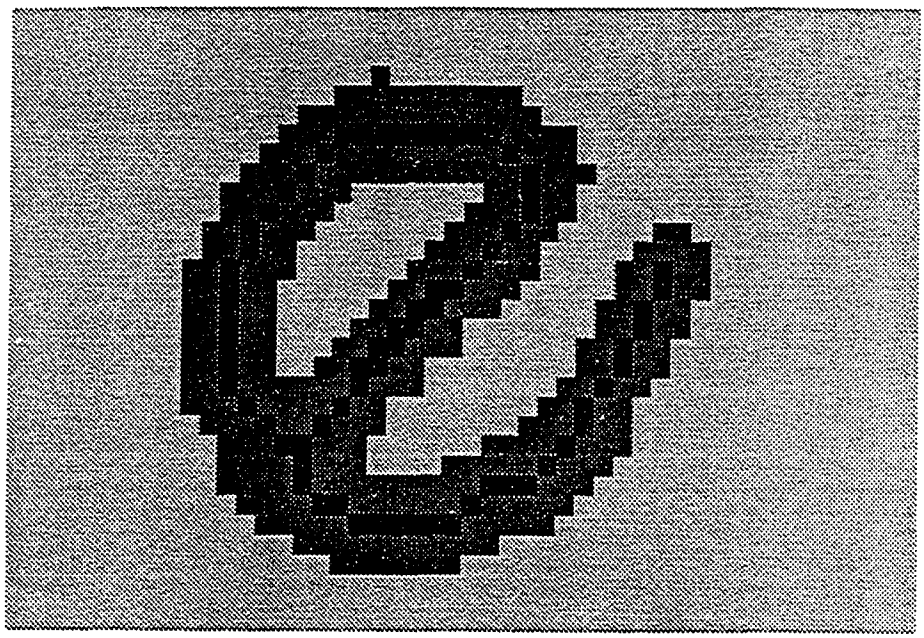


(a)

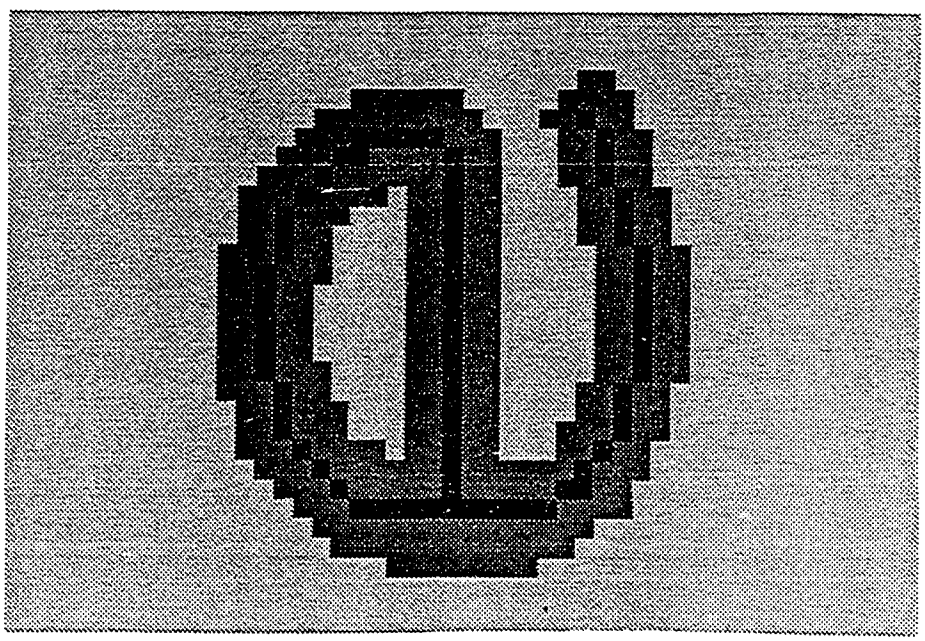


(b)

Fig. 2.9 The skeleton of "e" in (a) 0° (b) 30° (c) 45° (d) 90° rotations.



(c)



(d)

Fig. 2.9 (Continued)

CHAPTER III

THE ROOTS OF MORPHOLOGICAL FILTERS

The *back-propagation morphology* is defined to reach the roots of morphological filters. Its definition is given and properties are discussed in this chapter. The two-scan distrance transformation is proposed to show the advantage of this new definition.

3.1 Introduction

A morphological filter in gray-scale applications is a nonlinear digital filter which consists of a bounded mask (called the structuring element) with predefined values that slides over a signal of finite length. For each input sample, the output is the minimum or maximum of all neighboring input samples in the mask offset by their corresponding mask values centered at that input sample. If a finite signal that ends iteratively with a structuring element, converges to an invariant signal in a finite number of passes, such an invariant signal is called a *root* of the structuring element.

We define a new concept to reach the root called *back-propagation morphology* different from the traditionally defined morphology which we call *forward morphology*. During image scanning, the back-propagation morphological operations intend to feed back the output at each pixel to overwrite its input and continue in the same way until all pixels are scanned. We have developed several theorems of a two-scan algorithm using the back-propagation morphology to derive the root generation without recursively applying such forward morphology. Its operation is independent of the object size and saves significantly much computational time compared with the proportion to the number of iterations in forward morphology. A systolic array for efficiently processing the two-scan operation has been implemented by Shih *et al* [81]. In this chapter, we are not

going to discuss the properties of morphological roots. But we propose a new approach to reach the roots of morphological filters. The properties of morphological roots and their applications are the further research topics.

The best known and widely used filter based on order statistics is the *median filter*. A great effort has been made by many researchers to find signals (i.e. roots) that are invariant under median filtering. Roots of the median filters have been successfully used in edge enhancement [72] and image coding [67] The properties and convergence rates of median root can be found in [67]. Maragos and Schafer proved that median roots can be obtained by open-closing (opening followed by closing by the same structuring elements) and close-opening (closing followed by opening) filtering [58] which is the linkage between median filters and morphological filters.

Morphological filters posses certain nice syntactic and statistical properties, which are related to the corresponding properties of the median filters. Since morphological filters become more and more important in image processing and analysis. The root of morphological filters is interesting to investigate. One of its application is the distance transformation which will be discussed later.

This chapter will be organized as followed. We will give definitions and propositions of back-propagation morphology in Section 3.2. One example of morphological root application: distance transformation is given and discussed in Section 3.3. We give the summary and propose potential further research in Section 3.4.

3.2 Roots of Morphological Filters

Repeated application of the morphological filter on a quantized signal of finite length ultimately results in a sequence (*root signal* or *fixed point*) which is invariant to additional passes of the filters. A root of a filter Ψ, viewed as a mapping of transformation, is any signal f such that $\Psi(f) = f$.

A morphological filter in gray-scale applications is a nonlinear digital filter which consists of a bounded mask (called the structuring element) with predefined values that slides over a signal of finite length. For each input sample, the output is the minimum or maximum of all neighboring input samples in the mask offset by their corresponding mask values centered at that input sample. If a finite signal that ends iteratively with a structuring element k, converges to an invariant signal in a finite number of passes, such an invariant signal is called a *root* of the structuring element k. Let $\Psi(f, k)$ denote the morphological filter Ψ working on the image f and the structuring element k. We have the following definition for a morphological root:

Definition: Let us define $\Psi^n(f, k)$ is the output of morphological filter Ψ with respect to structuring elements k iterated n times and $\Psi^\infty(f, k)$ is the morphological root of Ψ with respect to k. That means $\Psi(f, k)$ will remain as an invariant output through a finite number of iterative operations. That is

$$\Psi^\infty(f, k) = \Psi^i(f, k) = \Psi^{(i+1)}(f, k), \tag{3.2.1}$$

where i is a finite integer and $\Psi^{(i+1)}(f, k) = \Psi(\Psi^i(f, k))$.

We call the traditionally defined morphology in chapter 1 as *forward morphology* since their outputs do not feed back to their inputs to affect their succeeding pixels' computation. The forward morphological operations are time-consuming when applied to find a root such that the computational complexity depends on the number of iterations. To speed up and avoid the iterations, a new definition called *back-propagation morphology* has been proposed. The new back-propagation morphological operations will compute the current scanning pixel result, output, and simultaneously feed back the result to overwrite its input in order to affect the succeeding pixels' computation.

Let $N(p)$ be the set of the neighbors that precede P in a scanning sequence of the picture within the window of a structuring element. We denote the back-propagation

filter as Φ. The *back-propagation dilation* of f by k is denoted by $\Phi_+(f, k)$ and defined as:

$$\Phi_+(f, k)(x,y) = \max \{ [\Phi_+(f, k)(x-m,y-n)] + k(m, n) \}, \qquad (3.2.2)$$

for all $(m, n) \in K$ and $(x-m, y-n) \in (N \cap F)$. The *back-propagation erosion* of f by k is denoted by $\Phi_-(f, k)$ and defined as:

$$\Phi_-(f, k)(x, y) = \min \{ [\Phi_-(f, k)(x+m, y+n)] - k(m, n) \}, \qquad (3.2.3)$$

for all $(m, n) \in K$ and $(x+m, y+n) \in (N \cap F)$.

Since back-propagation dilation (erosion) adopts the dilated (eroded) results of the preceding scanned neighbors to be involved in its computation, its output inherently depends on the image scanning sequence. In general, an image scanning can be classified into 1-D scan and 2-D scan. In moving from point to point, we always go from a point to one of its eight neighbors. Let us measure the moving direction counterclockwise from the positive x-axis. Thus, the 1-D scan can be separated into eight directions: 1) "*L*" denotes left-to-right or 0°; 2) "*R*" denotes right-to-left or 180°; 3) "*T*" denotes top-to-bottom or 270°; 4) "*B*" denotes bottom-to-top or 90°; 5) "45°"; 6)"135°"; 7)"225°"; 8)"315°". The 2-D scan can have four scanning sequences starting at a corner pixel: 1) "*LT*" denotes left-to-right and top-to-bottom (Note: This is a usual television raster scan.); 2) "*RB*" denotes right-to-left and bottom-to-top; 3) "*LB*" denotes left-to-right and bottom-to-top; 4) "*RT*" denotes right-to-left and top-to-bottom.

Assume a 3×3 structuring element k is denoted by

$$k = \begin{bmatrix} A_1 & A_2 & A_3 \\ A_4 & A_5 & A_6 \\ A_7 & A_8 & A_9 \end{bmatrix}. \qquad (3.2.4)$$

Because of the back-propagation effect with respect to the scanning direction, all the nine

elements $A_1, \cdots, A_9$ in k actually are not used. The redefined structuring element for the back-propagation morphology must satisfy the following criterion: wherever a pixel is being dealt with, all its neighbors defined within the structuring element k must have already been visited before by following the scanning sequence used. Thus, the k in one dimension is redefined as:

$$k_L = [A_4 \; A_5 \; x], \quad k_R = [x \; A_5 \; A_6], \quad k_T = \begin{bmatrix} A_2 \\ A_5 \\ x \end{bmatrix}, \quad k_B = \begin{bmatrix} x \\ A_5 \\ A_8 \end{bmatrix},$$

$$k_{45°} = \begin{bmatrix} & & x \\ & A_5 & \\ A_7 & & \end{bmatrix}, \quad k_{135°} = \begin{bmatrix} x & & \\ & A_5 & \\ & & A_9 \end{bmatrix},$$

$$k_{225°} = \begin{bmatrix} & & A_3 \\ & A_5 & \\ x & & \end{bmatrix}, \quad k_{315°} = \begin{bmatrix} A_1 & & \\ & A_5 & \\ & & x \end{bmatrix}; \tag{3.2.5}$$

the k in two dimensions is redefined as:

$$k_{LT} = \begin{bmatrix} A_1 & A_2 & A_3 \\ A_4 & A_5 & x \\ x & x & x \end{bmatrix}, \quad k_{RB} = \begin{bmatrix} x & x & x \\ x & A_5 & A_6 \\ A_7 & A_8 & A_9 \end{bmatrix},$$

$$k_{LB} = \begin{bmatrix} x & x & x \\ A_4 & A_5 & x \\ A_7 & A_8 & A_9 \end{bmatrix}, \quad k_{RT} = \begin{bmatrix} A_1 & A_2 & A_3 \\ x & A_5 & A_6 \\ x & x & x \end{bmatrix}, \tag{3.2.6}$$

where "x" means *don't care* or it can be defined as "-∞" according to mathematical morphological properties [85].

A complete omnidirectional scanning in the image space can be achieved by using eight-scan in one dimension and two-scan in two dimensions. The eight-scan method adopts eight back-propagation morphological filters sequentially in a non-specific order of eight scanning directions, *L*, *R*, *T*, *B*, 45°, 135°, 225°, 315° working on corresponding

structuring elements k_L, k_R, k_T, k_B, $k_{45°}$, $k_{135°}$, $k_{225°}$, $k_{315°}$, respectively. The two-scan method adopts two back-propagation morphological filters, the first being any one of four 2-D scans *LT*, *RB*, *LB*, *RT* working on corresponding structuring element, and the second being the opposite scanning sequence of the first scan and its scan-related structuring element. For example, the opposite scanning of *LT* is *RB*. From the followings, we will focus our attention on the theorem development of the two-scan method. These theorems can also apply to the eight-scan method.

We denote Φ_1 and Φ_2 as a pair of two-scan morphological filters such that Φ_1 and Φ_2 operate in opposite scanning sequences. Let k_1 and k_2 be the morphological structuring elements corresponding to the scanning sequences of operators Φ_1 and Φ_2, respectively. Let k_1 be the k_{LT} and k_2 be the k_{RB} as defined of eq. 3.2.6. We have the following proposition:

Proposition 3.1: If $\Phi^{\infty}(f, k_1)$ is the root of filter Φ with respect to k_1 then

$$\Phi^{\infty}(f, k_1) = \Phi_1(f, k_1). \tag{3.2.7}$$

[Proof]: It is obvious that the first pixel in an image in any scanning sequence satisfies

$$(\Phi^{\infty}(f, k_1))(P_1) = (\Phi_1(f, k_1))(P_1). \tag{3.2.8}$$

We assume that the *Nth* pixel P_n satisfies this proposition. Now consider the *(N+1)th* pixel P_{n+1}. P_{n+1} is determined by its neighbors defined by the domain of structuring element k_1, and k_1 satisfies the condition that when we scan P_{n+1}, all its neighbors have been previously scanned. Thus neighbors of P_{n+1} are:

$$\{P_i \mid i \le n,\ P_i = f(p+b),\ \ p \in P_{n+1},\ b \in k_1\}. \tag{3.2.9}$$

When the $\Phi^{\infty}(f, k_1)(P_{n+1})$ is computed, all its neighbors $\{P_i\}$ will not be changed. That

means $\{P_i\}$ have been computed and updated. Hence P_{n+1} will satisfy

$$(\Phi^{\infty}(f, k_1))(P_{n+1}) = (\Phi_1(f, k_1))(P_{n+1}). \quad \square$$

If the structuring elements k_1 is used, only one scan of back-propagation is enough to reach the morphological root. But usually the structuring elements used in morphological filters will not be as one of eq. 3.2.5 or eq. 3.2.6. Therefore, we need another operation in opposite scanning direction of Φ_1 which will complete the morphological filtering operation.

Proposition 3.2: If $\Phi^{\infty}(f, k)$ is the root of filter Φ with respect to k then

$$\Phi^{\infty}(f, k) = \Phi_2(\Phi_1(f, k_1), k_2). \tag{3.2.10}$$

[Proof]: Apparently, the neighborhood P_j of a pixel P within the window of the structuring element k may exist in two cases:

case (1): P_j belongs to k_1. In the first scan for computing Φ_1, according to *Proposition 3.1*, we can get the result $\Phi_1(f, k_1)$. In the second scan for computing Φ_2, the final result will not be affected because $(\Phi^{\infty}(f, k))(P)$ is determined by its neighbor P_j which is defined with structuring element k_1. Hence, we can obtain the correct result after computing Φ_2.

case (2): P_j belongs to k_2. According to *Proposition 3.1*, the result is easily obtained. Computing the Φ_2 in the scanning direction defined with structuring element k_2, we can get the result on the pixel P as *Proposition 3.1* stated.□

From proposition 3.1 and 3.2, we Since k_1 and k_2 are both independent (dealing with different neighbors), the order of applying k_1 and k_2 to a pixel is indifferent as long as the most up-to-date pixel value is used. The computation of two filters $\Phi_1(f, k_1)$ and $\Phi_2(f, k_2)$ can be initiated in parallel. That means the result will be generated only within a frame scanning time.

3.3 An Application: Distance Transformation

An application of the forward- and back-propagation morphology can be illustrated on the distance transformation. A distance transformation converts a binary image which consists of object (foreground) and non-object (background) pixels into an image where every object pixel has a value corresponding to the minimum distance to the background. The distance computation is in fact a global operation. Morphological erosion is an operation which selects the minimum value from the combination of an image and the predefined weighted structuring element within a window. Hence, mathematical morphology is a most appropriate approach to distance transformation. Applying the well-developed decomposition properties of mathematical morphology, we can significantly reduce the tremendous cost of global operations to that of small neighborhood operations, suitable for parallel pipelined computers [85, 86].

By setting the origin point as P, we can represent all the points by their distances from P (so that P is represented by 0). We select the weights of the structuring element to be entirely negative of the related distance measures, because gray-scale erosion is the minimum selection after applying the subtraction operation [86]. This will ensure that the output distance values stay within the same gray level range as the original gray-scale image. Three types of distance measures in digital image processing are usually used: Euclidean, city-block and chessboard [72]. We are only concerned with city-block and chessboard since their structuring elements can be decomposed into iterative dilations of small ones [85].

A binary image f (we use a small letter instead of a capital because of the gray level operation) consists of two classes, object pixels (foreground) and non-object pixels (background). Let the object pixels have the value "$+\infty$" (or any number larger than the object's highest distance) and the non-object pixels have the value "0." Let k be a 3×3 distance structuring element as follows:

$$k_{city-block} = \begin{bmatrix} -2 & -1 & -2 \\ -1 & 0 & -1 \\ -2 & -1 & -2 \end{bmatrix}, \quad k_{chessboard} = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 0 & -1 \\ -1 & -1 & -1 \end{bmatrix}.$$

The distance transformation algorithm by the forward morphology is as follows. Interested reader may refer to [86] for more detail.

Distance Transformation Algorithm:

(1) Do $d = f \ominus k$.

(2) Repeat (1) until the result does not change any more or until all "$+ \infty$" are removed.

The distance transformation has wide applications in image analysis. One of the applications is to compute a shape factor which is a measure of the compactness of the shape based on the ratio of the total number of object pixels to the summation of all distance measures [14]. Another is to obtain the medial axis (or skeleton) which is used for features extraction [4, 7, 40, 47, 50, 56].

We may apply the two-scan algorithm employing the back-propagation erosion to implement the distance transformation algorithm. The algorithm does not require iteration and only two scans are performed: one (k_1) is in the left-to-right, top-to-bottom scan and the other (k_2) is in the opposite (right-to-left, bottom-to-top) scan. Hence this algorithm can be implemented very fast, no matter what size the object is.

Two-Scan Distance Transformation Algorithm:

(1) Do $d_1 = \Phi_1(f, k_1)$.

(2) Do $d_2 = \Phi_2(d_1, k_2)$.

An example of chessboard distance transformation is given as follows.

Example: Let structuring elements k and original image f of which the object pixel value is set to be "255" be shown on fig. 3.1.

-1	-1	-1
-1	0	-1
-1	-1	-1

(a)

0	0	0	0	0	0	0
0	255	255	255	255	255	0
0	255	255	255	255	255	0
0	255	255	255	255	255	0
0	255	255	255	255	255	0
0	255	255	255	255	255	0
0	0	0	0	0	0	0

(b)

Fig. 3.1 (a) the structuring elements k and (b) the original image f.

The k applied here is chessboard distance measure. After the distance transformation, the result is shown in Fig. 3.2. Instead of applying iterative forward erosions, we use the back-propagation erosion with two decomposed structuring elements k_1 and k_2 in 2-D *LT* and *RB*, respectively.

$$k_1 = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 0 & x \\ x & x & x \end{bmatrix}, \qquad k_2 = \begin{bmatrix} x & x & x \\ x & 0 & -1 \\ -1 & -1 & -1 \end{bmatrix}.$$

By applying the two-scan algorithm to the original image f, we obtain the results of Φ_1

0	0	0	0	0	0	0
0	1	1	1	1	1	0
0	1	2	2	2	1	0
0	1	2	3	2	1	0
0	1	2	2	2	1	0
0	1	1	1	1	1	0
0	0	0	0	0	0	0

Fig. 3.2 The result of chessboard distance transformation.

and Φ_2 as shown in Fig. 3.3 and 3.4. Certainly, the result of two-scan distance transformation algorithm is exactly the same as original distance transformation algorithm.

0	0	0	0	0	0	0
0	1	1	1	1	1	0
0	1	2	2	2	2	0
0	1	2	3	3	3	0
0	1	2	2	2	2	0
0	1	1	1	1	1	0
0	0	0	0	0	0	0

Fig. 3.3 The result of $\Phi_1(f, k_1)$ where f is the original image.

3.4 Summary

We have defined a new concept which is called back-propagation morphology to reach the roots of morphological filters. Back-propagation morphology is different from the traditional morphology. During image scanning, the back-propagation morphological

0	0	0	0	0	0	0
0	1	1	1	1	1	0
0	1	2	2	2	1	0
0	1	2	3	2	1	0
0	1	2	2	2	1	0
0	1	1	1	1	1	0
0	0	0	0	0	0	0

Fig. 3.4 The result of $\Phi_2(\Phi_1(f, k_1), k_2)$.

operations intend to feed back the output at each pixel to overwrite its input and continue in the same way until all pixels are scanned and computed. We have developed several theorems of a two-scan algorithm using the back-propagation morphology to reach the root in only two scans without recursively applying such forward morphological operations. Its operation is independent of the size of a object and saves significantly much computational time compared with the proportion to the number of iterations in forward morphology. A systolic array implementation will make this two-scan operations much more efficient [81]. An example of applications of two-scan operations is shown in Section 3.3 which showed the computational advantages of back-propagation morphological filters in only two scans.

CHAPTER IV

GEOMETRIC SPECTRUM AND SHAPE DESCRIPTION

A useful morphological shape description tool called *G-spectrum* is defined, its properties are discussed and its application will be pointed out as an ongoing research.

4.1 INTRODUCTION

Mathematical morphology based on the geometric shape, provides a particular approach to the processing and analysis of digital images. The underlying strategy is to understand the characteristics of an object by probing its microstructure with various forms which are known as *structuring elements*. The analysis is geometric in character and it approaches image processing from the vantage point of human perception. Appropriately used, morphological operations also tend to simplify image data while preserving their essential shape characteristics and eliminating irrelevancies.

Shape description describes the object shape according to its geometric features. The shape of an object refers to its profile and physical structure. These characteristics can be represented by the boundary, region, moment, and structural representations. These representations can be used for matching shapes, recognizing objects, or making measurements on the shape characteristics. Therefore, the shape description is a very active and important issue in image processing, computer vision, and pattern recognition during recent decades. Many algorithms have been proposed to represent the shape. The skeleton representation and shape decomposition are the two most important categories. The idea of measuring the successive results of the morphological openings on an image by different sized structuring elements was initially suggested by Matheron and Serra.

Matheron explored the properties of these successive openings for the size distributions called *granulometries* [59]. Intuitively, a binary image is treated as a collection of grains (or particles) and the grains are sieved through a filter of increasing mesh size. After each filter is applied, the total number of remaining pixels is counted and the distribution of these counts reflects the grain size distribution. Serra applied the size distribution for the continuous size and the shape description on two-dimensional binary images [75].

Let $\{B(n) \mid n = 0, 1, \cdots, N-1\}$ be a sequence of structuring elements, such that the origin $(0, 0) \in B(n)$, and $B(n)$ contains at least one element, where

$$N = \max \{n \mid X \ominus A(n) \neq \varnothing\}, \tag{4.1.1}$$

and

$$A(0) = (0, 0), \tag{4.1.2}$$

$$A(n+1) = A(n) \oplus B(n), \qquad \text{for } n = 0, 1, \cdots, N-1. \tag{4.1.3}$$

More rigorously, the set $\{X \circ A(n) \mid n = 0, 1, \cdots, N\}$ of the morphological openings of an image X by the sequence $\{A(n) \mid n = 0, 1, \cdots, N\}$ of structuring elements provides a great deal of the shape and size information of the given image X. Due to the anti-extensive property of the morphological opening, the following is obtained.

$$X \circ A(0) \supseteq X \circ A(1) \supseteq \cdots \supseteq X \circ A(N). \tag{4.1.4}$$

Maragos proposed a shape-size descriptor which is called *pattern spectrum* [50, 51, 53, 55]. The discrete version of pattern spectrum is a very useful quantity for shape analysis which is given by

$$PS_X(n, B(n)) = Card(X \circ A(n) - X \circ A(n+1)) \quad \text{for } n = 0, 1, \cdots, N-1, \quad (4.1.5)$$

where $A(n+1)$ is computed by eq. (4.1.3).

Bronskill and Venetsanopoulos proposed *pecstrum* [8] of which the discrete version is defined as

$$P_n(X) = \frac{PS_X(n, B(n))}{Card(X)} \quad \text{for } n = 0, 1, \cdots, N-1. \quad (4.1.6)$$

Another variety called *probability distribution function* [20] is defined as

$$\Phi(n) = 1 - \frac{Card(X \circ A(n))}{Card(X)} \quad \text{for } n = 0, 1, \cdots, N-1. \quad (4.1.7)$$

In this chapter we present a useful morphological shape description tool called *geometric spectrum* or *G-spectrum*, for quantifying the geometric features on multidimensional binary images. The G-spectrum is proved to be a superior shape descriptor than the above descriptor in eqs. (4.1.5), (4.1.6), and (4.1.7) because of its less redundancy property.

This chapter will be organized as followed. The G-spectrum based on the morphological erosion and set transformation, is defined and some examples are given in Section 4.2. Its properties are discussed in Section 4.3. We propose a shape recognition algorithm using G-spectrum in Section 4.4. In the final Section, we give summary and further research.

4.2 G-SPECTRUM

The G-spectrum is a measurement for quantifying the geometric shape of discrete multidimensional images. From eq. (4.1.3), we know that

$$X \ominus A(n+1) = X \ominus (A(n) \oplus B(n))$$

$$= (X \ominus A(n)) \ominus B(n)$$

$$\subseteq X \ominus A(n), \qquad \text{for } n = 0, 1, \cdots, N. \tag{4.2.1}$$

Let $\{\Psi_n \mid n = 0, 1, , \cdots, N\}$ be a sequence of set transformations which satisfy that

$$X \ominus A(n+1) \subseteq \Psi_n[X \ominus A(n+1)] \subseteq X \ominus A(n) \quad \text{for } n = 0, 1, \cdots, N. \tag{4.2.2}$$

Some widely-used examples of the Ψ_n transformation [29] are given as follows.

Example 1: $\Psi_n(X) = X \oplus B(n)$ for $n = 0, 1, \cdots, N$.

Example 2: $\Psi_n(X) = X \oplus (B(n) \bullet A(n))$ for $n = 0, 1, \cdots, N$.

Example 3: $\Psi_n(X) = (X \oplus B(n)) \bullet A(n)$ for $n = 0, 1, \cdots, N$.

We introduce the G-spectrum as a shape descriptor which is based on a set of size distributions. The formal definition of G-spectrum is given as follows.

Definition 1: The G-spectrum is a set of values defined by

$$\text{G-spectrum} = \{ G_0(X), G_1(X), \cdots, G_N(X) \}, \tag{4.2.3}$$

where

$$G_n(X) = \frac{Card(X \ominus A(n)) - Card(\Psi_n[X \ominus A(n+1)])}{Card(X)},$$

$$\text{for } n = 0, 1, \cdots, N. \tag{4.2.4}$$

If we can find a sequence of transformations $\{\Psi_n \mid n = 0, 1, \cdots, N\}$ that satisfy eq. (4.2.2) then

$$X \ominus A(n) - \Psi_n[X \ominus A(n+1)] \subseteq X \ominus A(n) - X \ominus A(n+1),$$

$$\text{for } n = 0, 1, \cdots, N. \tag{4.2.5}$$

It is clear that

$$Card(X \ominus A(n) - \Psi_n[X \ominus A(n+1)]) \leq Card(X \ominus A(n) - X \ominus A(n+1)),$$

$$\text{for } n = 0, 1, \cdots, N. \tag{4.2.6}$$

According to eqs. (4.2.4) and (4.2.6) we can observe that our defined G-spectrum is less redundant than $R_n(X)$; that is

$$G_n(X) \leq \frac{Card(X \ominus A(n)) - Card(X \ominus A(n+1))}{Card(X)} = R_n(X). \tag{4.2.7}$$

It has been proved [29] that the upper-bound of the set transformations $\{\Psi_n\}$ which satisfy eq. (4.2.2), is $X \ominus A(n) \circ B(n) \bullet A(n)$ and also the following equation is satisfied by:

$$X \circ A(k) = \bigcup_{n=k}^{N} \left[(X \ominus A(n) - \Psi_n[X \ominus A(n+1)]) \oplus A(n) \right]. \tag{4.2.8}$$

The difference between two successive openings is

$$X \circ A(n) - X \circ A(n+1) = (X \ominus A(n) - \Psi_n[X \ominus A(n+1)]) \oplus A(n)$$

$$\supseteq X \ominus A(n) - \Psi_n[X \ominus A(n+1)]. \tag{4.2.9}$$

It is obviously that eq. (4.2.10)

$$Card(X \circ A(n) - X \circ A(n+1)) \geq Card(X \ominus A(n) - \Psi_n[X \ominus A(n+1)]), \tag{4.2.10}$$

implies $G_n(X) \leq P_n(X)$ according to eqs. (4.1.5), (4.1.6), and (4.2.4), where $P_n(X)$ is the nth element of "pecstrum". Hence, our defined G-spectrum has the least redundant size distribution, such that $G_n(X) \leq R_n(X)$ and $G_n(X) \leq P_n(X)$.

4.3 THE PROPERTIES OF G-SPECTRUM

The properties of G-spectrum are now presented and discussed in this section.

Proposition 4.1: For a given image X, each element of G-spectrum is a positive valued function. That is

$$G_n(X) \geq 0 \quad \text{for } n = 0, 1, \cdots, N . \tag{4.3.1}$$

[*Proof*]: From eq. (4.2.2), we know that

$$X \ominus A(n) \supseteq \Psi_n[X \ominus A(n+1)] . \tag{4.3.2}$$

By applying the cardinality to both sides yields

$$Card(X \ominus A(n)) \geq Card(\Psi_n[X \ominus A(n+1)]) . \tag{4.3.3}$$

Because $Card(X) \geq 0$, we have

$$\frac{Card(X \ominus A(n)) - Card(\Psi_n[X \ominus A(n+1)])}{Card(X)} \geq 0 . \tag{4.3.4}$$

According to eq. (4.2.4), the result in eq. (4.3.1) is obtained.□

Definition 2: The redundant reduction rate (*RRT*) of a given image X is defined as

$$RRT(X) = \frac{1}{Card(X)} \sum_{n=0}^{N} Card(\Psi_n[X \ominus A(n+1)] - X \ominus A(n+1)) . \tag{4.3.5}$$

As stated in Proposition 4.1, the G-spectrum is a set of positive values which gives the quantitative feature of an image based upon geometry. The redundant reduction rate is an indicator of how much redundant information can be reduced by using the G-spectrum. It is found that *RRT* also can be used in the matching procedure in object recognition.

Proposition 4.2: With a compact region of support, the summation of G-spectrum is equal to 1 minus the redundant reduction rate (RRT). That is

$$\sum_{n=0}^{N} G_n(X) = 1 - RRT(X). \tag{4.3.6}$$

[*Proof*]: From the definition of G-spectrum in eq. (4.2.4), we have

$$\sum_{n=0}^{N} G_n(X) = \frac{1}{Card(X)} \sum_{n=0}^{N} (Card(X \ominus A(n)) - Card(\Psi_n[X \ominus A(n+1)])) \tag{4.3.7}$$

$$= \frac{1}{Card(X)} \sum_{n=0}^{N} Card(X \ominus A(n) - \Psi_n[X \ominus A(n+1)]) \tag{4.3.8}$$

$$= \frac{1}{Card(X)} \sum_{n=0}^{N} Card(X \ominus A(n) - X \ominus A(n+1) - (\Psi_n[X \ominus A(n+1)] - X \ominus A(n+1))) \tag{4.3.9}$$

$$= \frac{1}{Card(X)} \sum_{n=0}^{N} Card(X \ominus A(n) - X \ominus A(n+1)) - \frac{1}{Card(X)} \sum_{n=0}^{N} Card(\Psi_n[X \ominus A(n+1)] - X \ominus A(n+1)) \tag{4.3.10}$$

Because $X \ominus A(0) = X$ and $X \ominus A(N+1) = \varnothing$, we have

$$\sum_{n=0}^{N} Card(X \ominus A(n) - X \ominus A(n+1)) = Card(X \ominus A(0) - X \ominus A(N+1))$$

$$= Card(X). \tag{4.3.11}$$

Therefore,

$$\sum_{n=0}^{N} G_n(X) = \frac{Card(X)}{Card(X)} -$$

$$\frac{1}{Card(X)} \sum_{n=0}^{N} Card(\Psi_n[X \ominus A(n+1)] - X \ominus A(n+1)) \tag{4.3.12}$$

$$= 1 - RRT(X) \qquad \square$$

The summation of G-spectrum is used to determine the degree of redundancy for an image representation. If $\sum_{n=0}^{N} G_n(X)$ is smaller, it means that the more redundant information is removed from the image. For an image X, the $RRT(X)$ will be varied with respect to the different sets of transformations. By employing the above concept, we are able to select a suitable set transformation which leads to the best performance on image coding.

Proposition 4.3: If $\Psi_n[X \oplus \{z\}] = \Psi_n[X] \oplus \{z\}$, for $n = 0, 1, \cdots, N$, then the G-spectrum is translation invariant. That is

$$G_n(X \oplus \{z\}) = G_n(X) \quad \text{for } n = 0, 1, \cdots, N. \tag{4.3.13}$$

[*Proof*]: The proof is derived from the fact of the translation invariance of erosion and cardinality, and from the assumption of the translation invariance of Ψ_n. That is

$$G_n(X \oplus \{z\}) = \frac{Card((X \oplus \{z\}) \ominus A(n)) - Card(\Psi_n[(X \oplus \{z\}) \ominus A(n+1)])}{Card(X \oplus \{z\})} \tag{4.3.14}$$

$$= \frac{Card((X \ominus A(n)) \oplus \{z\}) - Card(\Psi_n[X \ominus A(n+1)] \oplus \{z\})}{Card(X \oplus \{z\})} \tag{4.3.15}$$

$$= \frac{Card(X \ominus A(n)) - Card(\Psi_n[X \ominus A(n+1)])}{Card(X)} \tag{4.3.16}$$

$$= G_n(X). \qquad \square$$

The translation invariance property of the G-spectrum is useful in object recognition. Given any two G-spectrums $\{G_n(X_1) | n = 0, 1, \cdots, N\}$ and $\{G_n(X_2) | n = 0, 1, \cdots, N\}$, if both are the same or differ within a tolerance range, we say these two objects are matched. Although the G-spectrum is not scaling invariant, we can normalize the object into a pre-defined size. After normalization, the G-spectrum can also preserve the scaling invariance.

Proposition 4.4: The G-spectrum is scaling invariant if the set X is normalized. That is

$$G_n(Nr(\xi X)) = G_n(Nr(X)) \qquad \text{for } n = 0, 1, \cdots, N, \tag{4.3.17}$$

where ξ is an unknown scaling factor and $Nr(X)$ is a normalization function which is defined as

$$Nr(X) = \frac{\tau X}{Card(X)}, \tag{4.3.18}$$

where τ is a pre-defined value.

[*Proof*]: Because

$$Nr(\xi X) = \frac{\tau \xi X}{Card(\xi X)} = \frac{\tau \xi X}{\xi Card(X)} = Nr(X), \tag{4.3.19}$$

we have

$$G_n(Nr(\xi X)) = \frac{Card(\, Nr(\xi X) \ominus A(n)\,) - Card(\, \Psi_n[Nr(\xi X) \ominus A(n+1)]\,)}{Card(Nr(\xi X))} \tag{4.3.20}$$

$$= \frac{Card(\, Nr(X) \ominus A(n)\,) - Card(\, \Psi_n[Nr(X) \ominus A(n+1)]\,)}{Card(Nr(X))} \tag{4.3.21}$$

$$= G_n(Nr(X)) \qquad \square$$

From Proposition 4.4, if we perform the normalization (note that

$Card(Nr(X)) = \tau$) on the images with various scaling factors ξ, the G-spectrums of ξX and X are the same. This implies that the normalization according to a pre-defined value τ can produce the scaling invariant version of the G-spectrum.

Proposition 4.5: The first k elements of G-spectrum are zeros. That is

$$G_n(X) = 0 \qquad \text{for } n = 0, 1, \cdots, k-1 , \tag{4.3.22}$$

if and only if eqs. (4.3.23) and (4.3.24) are satisfied:

$$X \circ A(k) = X . \tag{4.3.23}$$

$$\Psi_n[X \ominus A(n+1)] = X \ominus A(n) \circ B(n) \bullet A(n) \qquad \text{for } n = 0, 1, \cdots, k-1 . \tag{4.3.24}$$

[*Proof*]: Case 1: assuming that eq. (4.3.22) is true, we prove that eqs. (4.3.23) and (4.3.24) are also true.

According to eq. (4.2.4), we have

$$Card(X \ominus A(n)) = Card(\Psi_n[X \ominus A(n+1)]) , \qquad \text{for } n = 0, 1, \cdots, k-1 . \tag{4.3.25}$$

This implies that

$$X \ominus A(n) = \Psi_n[X \ominus A(n+1)] , \qquad \text{for } n = 0, 1, \cdots, k-1 . \tag{4.3.26}$$

We replace X by $X \circ A(n+1)$ on the left-hand side of eq. (4.3.26) and obtain

$$(X \circ A(n+1)) \ominus A(n) = (X \ominus A(n+1) \oplus A(n+1)) \ominus A(n) \tag{4.3.27}$$

$$= (X \ominus (A(n) \oplus B(n)) \oplus (A(n) \oplus B(n))) \ominus A(n) \tag{4.3.28}$$

$$= X \ominus A(n) \ominus B(n) \oplus B(n) \oplus A(n) \ominus A(n) \tag{4.3.29}$$

$$= X \ominus A(n) \circ B(n) \bullet A(n) . \tag{4.3.30}$$

We replace X by $X \circ A(n+1)$ on the right-hand side of eq. (4.3.26) and obtain

$$\Psi_n[(X \circ A(n+1)) \ominus A(n+1)] = \Psi_n[X \ominus A(n+1) \oplus A(n+1) \ominus A(n+1)] \quad (4.3.31)$$

$$= \Psi_n[X \ominus A(n+1) \bullet A(n+1)] . \quad (4.3.32)$$

According to the mathematical morphology property, a set which is eroded by a structuring element is the same as the eroded result followed by a closing with the same structuring element. We then have

$$\Psi_n[(X \circ A(n+1)) \ominus A(n+1)] = \Psi_n[X \ominus A(n+1)] . \quad (4.3.33)$$

From eqs. (4.3.30) and (4.3.33), we can obtain

$$\Psi_n[X \ominus A(n+1)] = X \ominus A(n) \circ B(n) \bullet A(n) . \quad (4.3.34)$$

From eqs. (4.2.5) and (4.3.26), we have

$$X \circ A(k) = \bigcup_{n=0}^{N} \left[(X \ominus A(n) - \Psi_n[X \ominus A(n+1)]) \oplus A(n) \right] -$$

$$\bigcup_{n=0}^{k-1} \left[(X \ominus A(n) - \Psi_n[X \ominus A(n+1)]) \oplus A(n) \right] \quad (4.3.35)$$

$$= X \circ A(0) - \varnothing \quad (4.3.36)$$

$$= X \quad (4.3.37)$$

Case 2: assuming that eqs. (4.3.23) and (4.3.24) are true, we prove that eq. (4.3.22) is also true.

$$\Psi_n[X \ominus A(n+1)] = X \ominus A(n) \circ B(n) \bullet A(n) \quad (4.3.38)$$

$$= X \ominus A(n) \ominus B(n) \oplus B(n) \oplus A(n) \ominus A(n) \tag{4.3.39}$$

$$= (X \ominus (A(n) \ominus B(n)) \oplus (A(n) \oplus B(n))) \ominus A(n) \tag{4.3.40}$$

By applying eq. (4.1.3), we obtain

$$= [X \circ A(n+1)] \ominus A(n) . \tag{4.3.41}$$

According to eqs. (4.1.4) and (4.3.23), we obtain

$$X \circ A(n+1) = X \qquad \text{for } n = 0, 1, \cdots, k-1 . \tag{4.3.42}$$

Hence,

$$\Psi_n[X \ominus A(n+1)] = X \ominus A(n) \qquad \text{for } n = 0, 1, \cdots, k-1 . \tag{4.3.43}$$

From the definition of $G_n(X)$ in eq. (4.2.4), we obtain

$$G_n(X) = \frac{Card(X \ominus A(n)) - Card(\Psi_n[X \ominus A(n+1)])}{Card(X)} \tag{4.3.44}$$

$$= \frac{Card(X \ominus A(n)) - Card(X \ominus A(n))}{Card(X)} \tag{4.3.45}$$

$$= 0 \qquad \text{for } n = 0, 1, \cdots, k-1 . \qquad \square$$

If we can find a sequence of the set $\{A(n) \mid n = 0, 1, \cdots, N\}$ which satisfies eqs. (4.3.23) and (4.3.24), the recognition problem can be simplified by matching only $N-k+1$ elements of the G-spectrum. That means, if the G-spectrums of two sets X_1 and X_2 satisfy the following equation these two sets are regarded as the same.

$$G_n(X_1) - G_n(X_2) \leq \sigma \qquad \text{for } n = k, k+1, \cdots, N . \tag{4.3.46}$$

Proposition 4.6: If the set of structuring elements is chosen to be isotropic, the G-spectrum can be regarded as the rotation invariance.

The proof is straightforward. Since the structuring elements are isotropic, the G-spectrum defined as erosions by them, is naturally rotation-invariant. There is a relationship between pattern spectrum and G-spectrum. The next proposition will explore the relationship and prove that under some constraints, the elements of the G-spectrum and those of the pattern spectrum are equal to zero.

Proposition 4.7: There exists that for some n

$$G_n(X) = 0\,, \tag{4.3.47}$$

if and only if, the following are satisfied:

$$\Psi_n[X \ominus A(n+1)] = X \ominus A(n) \circ B(n) \bullet A(n)\,, \tag{4.3.48}$$

and

$$PS_X(\,n,\, B(n)\,) = 0\,. \tag{4.3.49}$$

[*Proof*]:

Case 1: assuming that eq. (4.3.47) is true, we prove that eqs. (4.3.48) and (4.3.49) are also true. According to eqs. (4.2.4) and (4.3.47), we have $Card(X \ominus A(n)) = Card(\,\Psi_n[X \ominus A(n+1)]\,)$. Since both sides are operating as the erosion on the same set X, we obtain

$$X \ominus A(n) = \Psi_n[X \ominus A(n+1)]\,. \tag{4.3.50}$$

The proof of eq. (4.3.48) is the same as the proof of eq. (4.3.24) in Proposition 4.5. To derive eq. (4.3.49), we first give

$$X \circ A(n) = X \ominus A(n) \oplus A(n) \tag{4.3.51}$$

$$= (\Psi_n[X \ominus A(n+1)]) \oplus A(n) \tag{4.3.52}$$

From eq. (4.3.48) we have

$$X \circ A(n) = (X \ominus A(n) \circ B(n) \bullet A(n)) \oplus A(n) \tag{4.3.53}$$

$$= X \ominus A(n) \ominus B(n) \oplus B(n) \oplus A(n) \ominus A(n) \oplus A(n) \tag{4.3.54}$$

$$= X \ominus (A(n) \oplus B(n)) \oplus (A(n) \oplus B(n)) \ominus A(n) \oplus A(n) \tag{4.3.55}$$

$$= X \ominus A(n+1) \oplus A(n+1) \circ A(n) \tag{4.3.56}$$

$$= X \circ A(n+1) \circ A(n) \tag{4.3.57}$$

$$\subseteq X \circ A(n+1) \tag{4.3.58}$$

From eq. (4.1.4), we know $X \circ A(n) \supseteq X \circ A(n+1)$. Hence, $X \circ A(n) = X \circ A(n+1)$. From eq. (4.1.5), we conclude $PS_X(n, B(n)) = 0$.

Case 2: assuming that eqs. (4.3.48) and (4.3.49) are true, we prove that eq. (4.3.47) is also true. From eqs. (4.1.5) and (4.3.49) we have

$$X \circ A(n) = X \circ A(n+1). \tag{4.3.59}$$

From the mathematical morphology property we obtain

$$X \ominus A(n) = (X \circ A(n)) \ominus A(n) \tag{4.3.60}$$

$$= (X \circ A(n+1)) \ominus A(n) \tag{4.3.61}$$

$$= X \ominus A(n) \circ B(n) \bullet A(n). \tag{4.3.62}$$

From eqs. (4.1.4), (4.3.48), and (4.3.62), we have

$$G_n(X) = \frac{Card(X \ominus A(n) \circ B(n) \bullet A(n)) - Card(X \ominus A(n) \circ B(n) \bullet A(n))}{Card(X)} \quad (4.3.63)$$

$$= 0 \qquad \square$$

The Proposition 4.7 tells us that if the transformation Ψ_n is constrained by eq. (4.3.48) and the *n*th element of the pattern spectrum is equal to zero, then the *n*th element of G-spectrum will be equal to zero or vice versa.

As many quantitative measures with respect to shape such as turning angles, length of sides, area, perimeter, radial lengths, and boundary coordinates have been proposed, it is difficult to standardize a minimum set of the shape descriptor to adequately quantify various object forms. The shape description techniques can be broadly classified into *external* and *internal*. The external shape description is based on the contour of a region under consideration, whereas the internal description deals with the region under consideration as an enclosed space.

The skeleton is one of the most important internal shape descriptors. The idea of transforming a binary image into an object skeleton and using the skeleton as a shape descriptor was first introduced by Blum [4]. If we set up the grass firing starting from the contour of a region, the fire burning spreads uniformly in all directions but in such a way that the waves generated do not flow through each other. The skeleton (or *medial axis*) is where the waves collide with each other in a frontal or circular manner. The skeleton of an object is the locus of the maximal inscribed circles within this object region. In other words, the medial axis is the symmetrical central description of the space whose boundary is the stimulus contour. The appearance of a skeleton starts at the minimum radius of curvature in the figure. The disappearance of a skeleton represents the largest circle that can be drawn in the figure. The skeleton sets can be used as the base for the shape description, classification and decomposition [50, 56, 87, 91].

The G-spectrum $\{G_n(X) \mid n = 0, 1, \cdots, N\}$ can be used as an internal shape descriptor which is based on the quantified geometric features instead of a sequence of discrete or connected pixels (e.g. the skeleton representation). Although the original image can not be reconstructed from the G-spectrum (i.e., G-spectrum is not an information-preserving descriptor), the G-spectrum is more useful than the skeleton in the shape recognition. It is not straightforward to apply a set of skeletons in solving the matching problem during shape recognition, however it is easier when a set of quantified geometric features, such as G-spectrum, is used. In other words, the G-spectrum is not only a shape descriptor, but also a tool for object recognition.

4.4 SUMMARY

In this chapter we define the G-spectrum not only as a shape descriptor for describing the quantified geometric features of multidimensional binary images, but also as a tool for shape recognition. The basis of G-spectrum relies upon the cardinality of a set of non-overlapping segments of an image using morphological operations. We have proved that the G-spectrum can preserve the invariance property for the transformations such as translation, rotation, and scaling. The G-spectrum can be easily extended and applied to multidimensional images.

CHAPTER V

MORPHOLOGICAL SHAPE RECOGNITION USING G-SPECTRUM

Properties of G-spectrum are presented and discussed in chapter 4. In this chapter, the application of G-spectrum is presented. It is found that G-spectrum is very useful in shape recognition.

5.1 The Recognition Algorithm using G-spectrum

By using the G-spectrum, a shape recognition algorithm is proposed as follows.

1. For every model object M_i, its G-spectrum and $RRT(M_i)$ are computed.
4. Given an object O, its G-spectrum and $RRT(O)$ are computed.
3. For every M_i if the following equation is satisfied where ε is a pre-defined tolerance value, we regard it as a candidate and denote as C_j.

$$RRT(O) - RRT(M_i) \leq \varepsilon \tag{5.1.1}$$

4. For every candidate C_j we got in step 3, we analysis and compare its G-spectrum with given object's G-spectrum. If they are matched (i.e. the following equation is satisfied, where σ is a pre-defined tolerance value), we recognize it.

$$\sum_{i=0}^{n}\left[G_i(O) - G_i(C_j)\right] \leq \sigma \tag{5.1.2}$$

In this algorithm, *RRT* is used as a pre-probe which can simplify the recognition procedure. According to proposition 4.7, it shows that if the structuring elements set and transformation set $\{\Psi_n\}$ satisfied the constraint in eq. (4.3.48), the recognition procedure will become much simplier.

The shape likeliness between two shapes can be defined by G-spectrum as follows.

Definition 5.1.1: The shape likeliness between two shapes p and q is defined as

$$\sum_{i=0}^{n}\left[G_i(S_p) - G_i(S_q)\right] \tag{5.1.3}$$

5.2 Experimental Results

In this section, experimental results will be presented and evaluated. Six different shapes are shown in Fig. 5.2 and structuring element sets are shown in Fig. 5.1. The G-spectrum (whose values are multiplied by 1000 in order to be illustrated clearly) of these six shapes using different Ψ transformation are shown in Figs. 5.3 - 5.12. The matching results of these six shapes using different Ψ transformation are shown in Figs. 5.13 - 5.22, which two shapes i and j are the most alike among these six shapes by selecting the minimal value of their shape likeliness of $\sum_{i=0}^{n}\left[G_i(O) - G_i(C_j)\right]$.

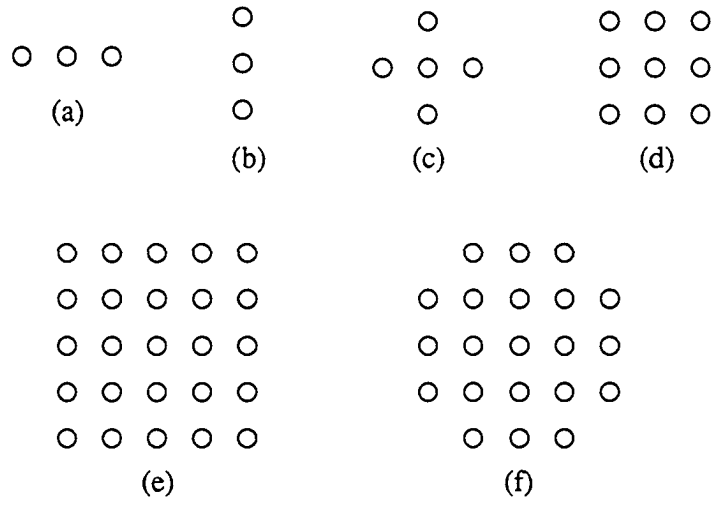
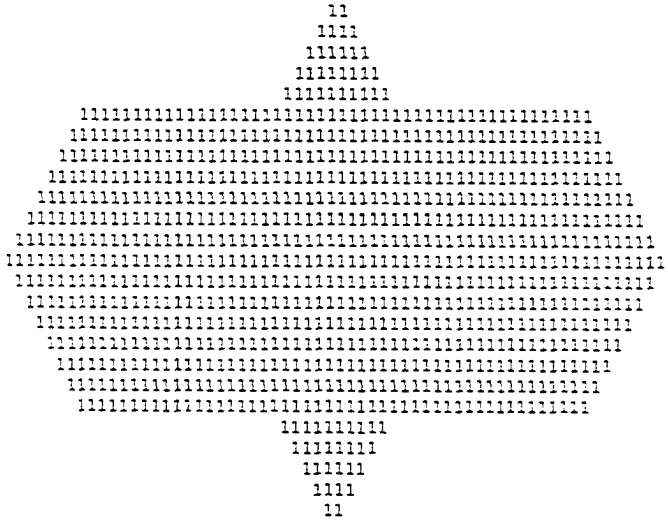
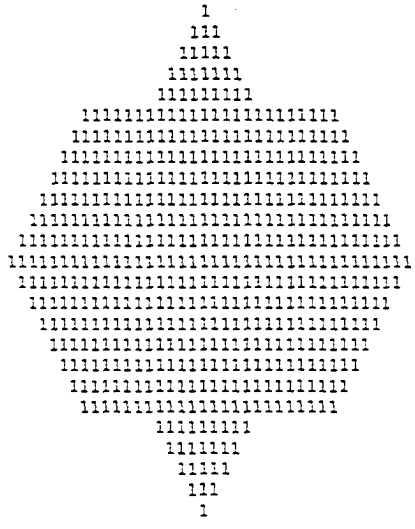


Fig. 5.1 Structuring element sets (a) h3, (b) v3, (c) din3, (d) sq3, (e) sq5, and (f) r5.

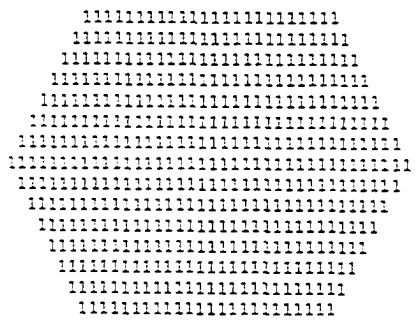


(a)

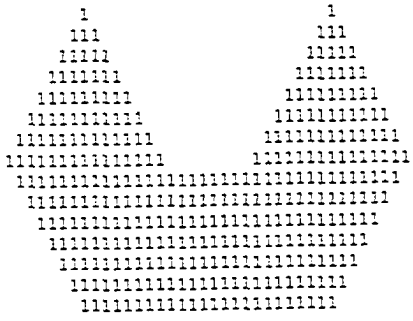


(b)

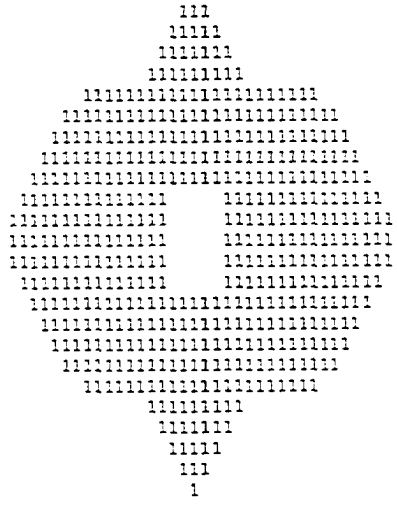
Fig. 5.2 (a) shape1, (b) shape2, (c) shape3, (d) shape4, (e) shape5, and (f) shape6.



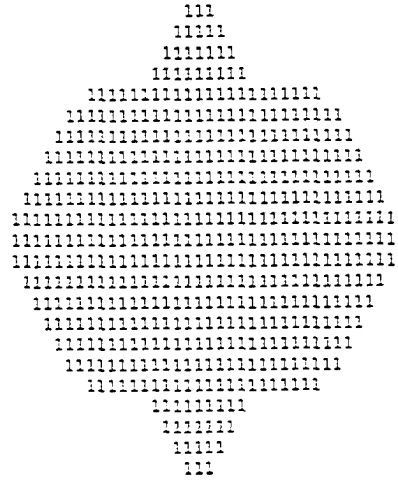
(c)



(d)



(e)



(f)

Fig. 5.2 (Continued)

Object	G_0	G_1	G_2	G_3	*RRT*
s_1	23	23	14	125	815
s_2	31	31	20	74	843
s_3	17	17	17	83	865
s_4	47	373			580
s_5	27	109	74		790
s_6	24	32	20	68	856

Fig. 5.3 G-spectrum of six shapes where $\Psi(f) = f \oplus sq\,5$ and $A(n) = sq\,5$.

Object	G_0	G_1	G_2	G_3	G_4	G_5	G_6	G_7	G_8	G_9	G_{10}	G_{11}	G_{12}	*RRT*
s_1	87	87	87	87	87	87	87	46	5	5	5	5	2	328
s_2	55	55	55	55	55	55	55	30	4	4	4	4	2	567
s_3	105	105	105	105	105	105	105	52						214
s_4	95	95	95	77	47	36	24	6						527
s_5	76	76	76	126	69	38	21	6						513
s_6	64	64	64	64	64	64	56	34	12	12	12	6		484

Fig. 5.4 G-spectrum of six shapes where $\Psi(f) = f \oplus h\,3$ and $A(n) = din\,3$.

Object	G_0	G_1	G_2	G_3	*RRT*
s_1	326	271	212	125	66
s_2	366	272	173	75	114
s_3	362	275	188	83	92
s_4	538	373			89
s_5	464	303	74		160
s_6	364	276	176	68	116

Fig. 5.5 G-spectrum of six shapes where $\Psi(f) = f \oplus h\,3$ and $A(n) = sq\,5$.

Object	G_0	G_1	G_2	G_3	*RRT*
s_1	241	187	132	125	314
s_2	315	220	130	75	260
s_3	279	210	140	83	288
s_4	450	373			176
s_5	410	315	74		202
s_6	312	224	132	68	264

Fig. 5.6 G-spectrum of six shapes where $\Psi(f) = f \oplus v3$ and $A(n) = sq5$.

Object	G_0	G_1	G_2	G_3	G_4	G_5	G_6	G_7	*RRT*
s_1	7	7	7	7	7	2	2	39	923
s_2	8	8	8	8	8	4	4	20	933
s_3	4	4	4	4	4	4	4	22	948
s_4	12	12	12	130					834
s_5	2	8	42	17	17	11			903
s_6	0	8	8	8	8	4	8	16	940

Fig. 5.7 G-spectrum of six shapes where $\Psi(f) = f \oplus sq3$ and $A(n) = sq3$.

Object	G_0	G_1	G_2	G_3	G_4	*RRT*
s_1	7	7	7	130	2	847
s_2	8	8	8	85	2	890
s_3	4	4	4	122		865
s_4	12	107	77			805
s_5	2	126	34	6		832
s_6	0	8	8	78	2	904

Fig. 5.8 G-spectrum of six shapes where $\Psi(f) = f \oplus r5$ and $A(n) = r5$.

Object	G_0	G_1	G_2	G_3	G_4	G_5	G_6	G_7	*RRT*
s_1	144	134	125	116	107	93	84	39	157
s_2	150	134	118	102	87	67	51	20	272
s_3	162	144	127	109	92	74	57	22	214
s_4	225	201	178	130					266
s_5	174	172	147	71	46	11			378
s_6	144	136	120	104	88	68	48	16	276

Fig. 5.9 G-spectrum of six shapes where $\Psi(f) = f \oplus h3$ and $A(n) = sq3$.

Object	G_0	G_1	G_2	G_3	G_4	*RRT*
s_1	278	251	223	155	2	91
s_2	287	240	193	120	2	157
s_3	314	262	210	122		92
s_4	432	331	77			160
s_5	372	319	118	6		185
s_6	284	244	196	114	2	160

Fig. 5.10 G-spectrum of six shapes where $\Psi(f) = f \oplus h3$ and $A(n) = r5$.

Object	G_0	G_1	G_2	G_3	G_4	G_5	G_6	G_7	*RRT*
s_1	59	50	41	32	23	14	9	39	733
s_2	98	83	67	51	35	24	16	20	606
s_3	70	61	52	44	35	26	17	22	672
s_4	136	112	89	130					533
s_5	120	118	160	84	50	11			458
s_6	92	84	68	52	36	24	20	16	608

Fig. 5.11 G-spectrum of six shapes where $\Psi(f) = f \oplus v3$ and $A(n) = sq3$.

Object	G_0	G_1	G_2	G_3	G_4	RRT
s_1	194	166	139	153	2	346
s_2	236	189	142	120	2	311
s_3	223	179	135	122		341
s_4	343	320	77			260
s_5	317	324	113	6		239
s_6	232	192	144	114	2	316

Fig. 5.12 G-spectrum of six shapes where $\Psi(f) = f \oplus v3$ and $A(n) = r5$.

	s_1	s_2	s_3	s_4	s_5	s_6
s_1		73	57	513	275	73
s_2			40	452	210	14
s_3				486	242	40
s_4					358	452
s_5						202
s_6						

Fig. 5.13 Likeliness among six shapes where $\Psi(f) = f \oplus sq5$ and $A(n) = sq5$.

	s_1	s_2	s_3	s_4	s_5	s_6
s_1		244	154	250	267	205
s_2			390	242	241	87
s_3				312	341	355
s_4					133	253
s_5						234
s_6						

Fig. 5.14 Likeliness among six shapes where $\Psi(f) = f \oplus h3$ and $A(n) = din3$.

	s_1	s_2	s_3	s_4	s_5	s_6
s_1		130	106	651	433	136
s_2			30	521	303	16
s_3				545	327	30
s_4					218	515
s_5						297
s_6						

Fig. 5.15 Likeliness among six shapes where $\Psi(f) = f \oplus h3$ and $A(n) = sq5$.

	s_1	s_2	s_3	s_4	s_5	s_6
s_1		159	111	652	480	165
s_2			64	493	321	16
s_3				557	385	70
s_4					172	487
s_5						315
s_6						

Fig. 5.16 Likeliness among six shapes where $\Psi(f) = f \oplus v3$ and $A(n) = sq5$.

	s_1	s_2	s_3	s_4	s_5	s_6
s_1		28	36	188	111	42
s_2			22	170	89	16
s_3				184	103	30
s_4					185	178
s_5						85
s_6						

Fig. 5.17 Likeliness among six shapes where $\Psi(f) = f \oplus sq3$ and $A(n) = sq3$.

	s_1	s_2	s_3	s_4	s_5	s_6
s_1		48	19	307	277	61
s_2			51	259	231	15
s_3				306	270	58
s_4					78	260
s_5						220
s_6						

Fig. 5.18 Likeliness among six shapes where $\Psi(f) = f \oplus r5$ and $A(n) = r5$.

	s_1	s_2	s_3	s_4	s_5	s_6
s_1		125	115	538	401	122
s_2			58	455	290	21
s_3				437	286	63
s_4					227	450
s_5						289
s_6						

Fig. 5.19 Likeliness among six shapes where $\Psi(f) = f \oplus h3$ and $A(n) = sq3$.

	s_1	s_2	s_3	s_4	s_5	s_6
s_1		85	95	537	418	81
s_2			70	474	355	16
s_3				442	323	72
s_4					119	470
s_5						351
s_6						

Fig. 5.20 Likeliness among six shapes where $\Psi(f) = f \oplus h3$ and $A(n) = r5$.

	s_1	s_2	s_3	s_4	s_5	s_6
s_1		165	94	370	378	171
s_2			77	263	247	18
s_3				340	324	81
s_4					200	267
s_5						249
s_6						

Fig. 5.21 Likeliness among six shapes where $\Psi(f) = f \oplus v3$ and $A(n) = sq3$.

	s_1	s_2	s_3	s_4	s_5	s_6
s_1		101	79	520	456	108
s_2			34	425	361	15
s_3				441	377	41
s_4					72	422
s_5						358
s_6						

Fig. 5.22 Likeliness among six shapes where $\Psi(f) = f \oplus v3$ and $A(n) = r5$.

5.3 Summary

The experimental results show that the shape likeliness is minimum which indicating that these two shapes are the most alike and their *RRT* are very close. Refining the algorithm by real image, discussing the tolerance, and defining the thresholding value of equivalence will be the future research topics.

CHAPTER VI

SOFT MATHEMATICAL MORPHOLOGY AND ITS PROPERTIES

The generation of soft morphological operations was motivated by Koskinen, Astola, and Neuvo [44, 45]. The soft morphological operations are less sensitive to additive noise and to small variations. New definitions of binary soft morphological operations are given. The properties and proofs of soft morphological operations are studied and discussed. It also shows that soft morphological operations commute with thresholding and obey threshold-linear superposition. In general, soft closing and soft opening are not idempotent operations, but under some constraint the soft operations can be idempotent and the prove is given. The properties of *idempotent soft morphological filters* will be studied and discussed.

6.1 INTRODUCTION

Mathematical morphology which is based on set theory, provides an algebraic approach to manifest structuring shapes on binary or gray scale images [75, 77]. The underlying strategy is to expose the characteristics of an object by probing its microstructure with various forms which are known as *structuring elements*. The analysis is geometrical in nature and it approaches to image processing from the vantage point of human perception. The essential morphological operations are dilation, erosion, opening, and closing. Dilation by a structuring element corresponds to expansion, and erosion corresponds to shrinking. Appropriately combined, the functionality of opening and closing closely corresponds to the specification of a filter by its bandwidth. They tend to simplify image data while preserving their primary shape characteristics and eliminating irrelevancies. The opening of an image will remove all of the pixels in regions that are too small to

contain the probe. The opposite sequence, closing, will fill in holes and concavities smaller than the probe. Such morphological filters can be used to suppress spatial features or discriminate against objects based on their size distribution [20]. As an example, if a disk structuring element is used, the opening of an image is equivalent to a low-pass filter. The opening residue is of course a high-pass filter. The difference of two openings of an image with two structuring elements with different radii corresponds to a band-pass filter.

Morphological filters constitute a highly nonlinear system. Another popular family of nonlinear filters is the order statistic filters which are based on statistics theory and have been applied to signal detection and image enhancement. Maragos and Schafer [57] extended the theory of median, order statistic and stack filters to mathematical morphology. They have shown that the order statistic filters can be used in both function- and set-processing and can commute with thresholding.

Soft morphological filters are the combination of the order statistic filters and morphological filters. The primary difference to standard morphological filters is that the maximum and minimum operations are replaced by more general weighted order statistics and the "soft" boundary is added to the structuring element. Koskinen *et al.* [44, 45] have shown that soft morphological operations are less sensitive to additive noise and to small variations in object shape, and they preserve most of the desirable properties of standard morphological operations.

In this chapter, the properties of threshold-linear superposition, idempotence, and binary soft morphological operations are developed. The soft morphological operations are proved to be capable of commuting with thresholding and the threshold-linear superposition property allows fast implementation of gray scale soft morphological operations by using only logic gates. The implementation and the analysis of gray scale soft morphological operations can be interpreted by focusing only on the case of sets that are

much easier to deal with since the binary soft morphological operations only involve counting the number of pixels instead of sorting the values. The idempotence property provides criteria to select the structuring elements and the order index, that leads the soft morphological filters to reach the root signal without iteration.

This chapter is organized as follows. In section 6.2 the definitions of soft morphological operations are given. In section 6.3 the properties of soft morphological operations, including increasing and translation-invariant, are discussed. In section 6.4 the threshold decomposition and threshold-linear superposition are developed. In section 6.5 the class of idempotent soft morphological filters are presented. Finally, summary is provided.

6.2 Definitions of Soft Morphological Operations

Let f be a function defined on m-dimensional discrete Euclidean space Z^m. Let W be a window, a finite subset of Z^m containing N points where $N = Card(W)$, the cardinality of W. The kth order-statistics (OS) of a function $f(x)$ with respect to the window W is a function whose value at location x is obtained by sorting in descendent order the N values of $f(x)$ inside the window W whose origin is shifted to location x and picking up the kth number from the sorted list, where k ranges from 1 to N. Let W_x denote the translation of the origin of a set W to location x. The kth OS is represented by

$$k\text{th } OS(f, W)(x) = k\text{th largest of } \{f(a) | a \in W_x\}. \tag{6.2.1}$$

Note that if N is an odd number and $k = (N+1)/2$, the kth OS corresponds to a median filter. If $k = 1$, the kth OS becomes a maximum filter. That is

$$1\text{st } OS(f, W)(x) = \max \{f(a) | a \in W_x\}. \tag{6.2.2}$$

If $k = N$, the kth OS becomes a minimum filter. That is

$$N\text{th } OS(f, W)(x) = \min \{ f(a) | a \in W_x \}. \tag{6.2.3}$$

The soft morphological operations adopt the concept of order statistics to replace the maximum and minimum in standard morphological operations. The basic idea of soft morphological operations is that the structuring element B is split into two subsets: the core set A and the soft boundary set $B \setminus A$, where "\" denotes the set difference. Soft morphological dilation (erosion) of a function with respect to a finite set are regarded as the gray scale soft morphological dilation (erosion) whose output value at location x is obtained by sorting in descendent (ascendent) order of the $Card(B \setminus A) + k \times Card(A)$ values of the input image that includes the pixels inside $B \setminus A$ and k times repetition of the pixels inside A and then selecting the k-th order from the sorted list. Let $\{k \Diamond f(a)\}$ denote the repetition k times of $f(a)$ which means $\{k \Diamond f(a)\} = \{f(a), f(a), \cdots, f(a)\}$ (k times). The definitions are given as follows, where f is a gray scale image and B is a flat structuring elements.

Definition 6.2.1: The soft morphological dilation of f by $[B, A, k]$ is defined as

$$(f \oplus [B, A, k])(x) = k\text{th largest of } (\{ k \Diamond f(a) | a \in A_x \} \cup \{ f(b) | b \in (B \setminus A)_x \}). \tag{6.2.4}$$

Definition 6.2.2: The soft morphological erosion of f by $[B, A, k]$ is defined as

$$(f \ominus [B, A, k])(x) = k\text{th smallest of } (\{ k \Diamond f(a) | a \in A_x \} \cup \{ f(b) | b \in (B \setminus A)_x \}). \tag{6.2.5}$$

Definition 6.2.3: The soft morphological closing of f by $[B, A, k]$ is defined as

$$f \bullet [B, A, k] = f \oplus [B, A, k] \ominus [B, A, k]. \tag{6.2.6}$$

Definition 6.2.4: The soft morphological opening of f by $[B, A, k]$ is defined as

$$f \circ [B, A, k] = f \ominus [B, A, k] \oplus [B, A, k]. \tag{6.2.7}$$

Note that if $k = 1$, the soft morphological operations in eqs. (6.2.4) - (6.2.7) are equivalent to standard morphological operations.

6.3 Properties of Soft Morphological Operations

In this section we present and prove general properties of soft morphological filters which are valid for both set- and function-processing. The notations used in the previous sections will be continuously used through this chapter.

Definition 6.3.1: A transformation Ψ is said to be *idempotent* if $\Psi(\Psi(f)) = \Psi(f)$.

Definition 6.3.2: A transformation Ψ is said to be *increasing* if for any two input signals f and g, such that $f(x) \le g(x)$ for every x, the resultant outputs $\Psi(f(x)) \le \Psi(g(x))$.

Proposition 6.3.1: The soft morphological dilation and erosion are increasing.

[Proof]: Let $f(x) \le g(x)$ for every x. From eq. (6.2.4) we have

$$k\text{th largest of } (\{k \lozenge f(a) \,|\, a \in A_x\} \cup \{f(b) \,|\, b \in (B \backslash A)_x\})$$

$$\le k\text{th largest of } (\{k \lozenge g(a) \,|\, a \in A_x\} \cup \{g(b) \,|\, b \in (B \backslash A)_x\}) \qquad (6.3.1)$$

that implies

$$(f \oplus [B, A, k])(x) \le (g \oplus [B, A, k])(x) \quad \text{for every } x. \qquad (6.3.2)$$

Thus, the soft morphological dilation is increasing. The proof for soft morphological erosion can be similarly derived. □

From Proposition 6.3.1, we can easily prove that the soft morphological closing and opening are also increasing.

Proposition 6.3.2: The soft morphological closing and opening are increasing.

[Proof]: Denote soft morphological dilation be Ψ_1 and soft morphological erosion be Ψ_2.

Let $f \leq g$. Since Ψ_1 and Ψ_2 are increasing, we have

$$\Psi_1(f) \leq \Psi_1(g), \quad \text{and } \Psi_2(f) \leq \Psi_2(g). \tag{6.3.3}$$

Therefore,

$$\Psi_2(\Psi_1(f)) \leq \Psi_2(\Psi_1(g)) \tag{6.3.4}$$

and

$$\Psi_1(\Psi_2(f)) \leq \Psi_1(\Psi_2(g)). \tag{6.3.5}$$

Eqs. (6.3.4) and (6.3.5) states that the soft morphological closing and opening are increasing respectively. □

Proposition 6.3.3: The soft morphological operations are translation-invariant. That is for any soft morphological operation Ψ with respect to any $z \in Z^m$, we have

$$\Psi_z(f) = \Psi(f_z). \tag{6.3.6}$$

The proof of translation invariance property is straightforward and therefore skipped.

Proposition 6.3.4: The soft morphological dilation propagates the local maximum of the pixels within B to all the pixels within A and fills up the valley whose area is less than or equal to $Card(A)$.

[Proof]: Let $f(x)$ be the local maximum in the neighborhood of the translated B_x. According to eq. (6.2.4), $f(x')$ will be changed to $f(x)$ if $x' \in A_x$. In other word, the local maximum $f(x)$ will be expanded to the neighborhood whose area is $Card(A)$.

Let $f(x)$ be the valley whose area is less than or equal to $Card(A)$. It means that some values in $B \setminus A$ are greater than $f(x)$. Thus the value of $f(x)$ must be increased after soft morphological dilation is applied because the kth largest of $\{k \lozenge f(a) \mid a \in A_x\} \cup \{f(b) \mid b \in (B \setminus A)_x\}$ must exist in $\{f(b) \mid b \in (B \setminus A)_x\}$. In other words, the valley is filled up. □

Proposition 6.3.5: The soft morphological erosion propagates the local minimum of the pixels within B to all the pixels within A and eliminates the peak whose area is less than $Card(A)$.

The above proof can be similarly derived from the proof of Proposition 6.3.4 and is skipped. The following provides an interesting property that the soft morphological dilation is equivalent to the soft morphological erosion if $k = Card(B)$ and $Card(A) = 1$.

Proposition 6.3.6: If $k = Card(B)$ and $Card(A) = 1$, then

$$f \oplus [B, A, k] = f \ominus [B, A, k]. \tag{6.3.7}$$

[Proof]: From the definition in eq. (6.2.4), if $k = Card(B)$ and $Card(A) = 1$, for any x, $f \oplus [B, A, k](x)$ is the kth element of the sorted list in which the total number of elements is $2\times k - 1$. It means that $f \oplus [B, A, k](x)$ is the middle one of the sorted list. According to eq. (6.2.5), $f \ominus [B, A, k](x)$ is the middle one of the sorted list, which is exactly the same as $f \oplus [B, A, k](x)$. □

6.4 Threshold Decomposition and Superposition

The soft morphological operations can be applied to both for set- and function-processing. Because of the increasing property, they can be threshold decomposed. By using the concept of cross-sections of a function and the definition of soft morphological operations for set-processing, we can prove that the function-processing soft morphological operations commute with thresholding. The set-processing soft morphological operations can be equivalently defined as follows, that only involve in counting the number of pixels instead of sorting numbers. Let X be a binary image. Also let $n = Card(A)$ and $N = Card(B)$. Thus, the total number of pixels within B after repetition k times is $k\times n + (N-n)$.

Definition 6.4.1: The binary soft morphological dilation of X by $[B, A, k]$ is defined as

$$X \oplus [B, A, k] = \{ y \mid k \times Card(X \cap A_y) + Card(X \cap (B \setminus A)_y) \geq k \}. \quad (6.4.1)$$

Definition 6.4.2: The binary soft morphological erosion of X by $[B, A, k]$ is defined as

$$X \ominus [B,A,k] = \{ y \mid k \times Card(X \cap A_y) + Card(X \cap (B \setminus A)_y) \geq N+(k-1)\times n-k+1 \}. \quad (6.4.2)$$

Note that $k \times Card(X \cap A_y) + Card(X \cap (B \setminus A)_y)$ is always less than or equal to $N + (k-1)\times n$ which is the total number of pixels involved in computation.

Example: Let $X = \{100101111\}$, $B = \{111\}$, $A = \{1\}$ which is the center point of B, and $k = 2$. The results after the soft morphological dilation and erosion are:

$$X \oplus [B,A,2] = \{100111111\}, \text{ and}$$

$$X \ominus [B,A,2] = \{100001111\}.$$

Let X^c denote the *set complement* of X. We will show the interaction of the soft morphological dilation of a set X with its complementation.

Proposition 6.4.3: The soft morphological dilation of a set X followed by a complement is equivalent to the dilation on X^c. iff $k = Card(B) - 1$ and $Card(A) = 1$, i.e., the core set contains only one center point.

$$X^c \oplus [B, A, Card(B)-1] = (X \oplus [B, A, Card(B)-1])^c. \quad (6.4.3)$$

[Proof]:

$$y \in X^c \oplus [B, A, Card(B)-1] \quad (6.4.4)$$

$$\Leftrightarrow (Card(B)-1) \times Card(X^c \cap A_y) + Card(X^c \cap (B \setminus A)_y) \geq Card(B)-1 \quad (6.4.5)$$

$$\Leftrightarrow (Card(B)-1) \times Card(X \cap A_y) + Card(X \cap (B \setminus A)_y) < Card(B) - 1 \quad (6.4.6)$$

$$\Leftrightarrow y \notin X \oplus [B, A, Card(B)-1] \tag{6.4.7}$$

$$\Leftrightarrow y \in [X \oplus [B, A, Card(B)-1]]^c \tag{6.4.8}$$

□

The *cross-section* $X_t(f)$ of f at level t is the set obtained by thresholding f at level t:

$$X_t(f) = \{x \mid f(x) \geq t\}, \text{ where } -\infty < t < \infty. \tag{6.4.9}$$

Theorem 6.4.1: The soft morphological operations of functions commute with thresholding. That is for any f, A, B and $k = 1, 2, \cdots, N-1$, we have

$$X_t(f \oplus [B, A, k]) = X_t(f) \oplus [B, A, k]. \tag{6.4.10}$$

[Proof]: We only prove that the soft morphological dilation of functions commutes with thresholding. The proofs for the other soft morphological operations are similar. Let $g(x) = (f \oplus [B, A, k])(x)$. Then

$$z \in X_t(g) \Leftrightarrow g(z) \geq t \tag{6.4.11}$$

$$\Leftrightarrow k \times Card(X_t(f) \cap A_z) + Card(X_t(f) \cap (B \setminus A)_z) \geq k \tag{6.4.12}$$

$$\Leftrightarrow z \in X_t(f) \oplus [B, A, k] \tag{6.4.13}$$

□

The essence of Theorem 6.4.1 is that the soft morphological operation of a function followed by thresholding at level t is equivalent to thresholding of the function at level t followed by soft morphological filtering of the resultant cross section. Both ways should yield the same result. The implementation and analysis of function-processing soft morphological operations can be achieved by the set-processing operations which are much

easier to deal with because their definitions involve only in counting the number of pixels instead of sorting numbers.

Consider the thresholded binary images

$$f_a(x) = \begin{cases} 1 & f(x) \geq a \\ 0 & f(x) < a, \end{cases} \tag{6.4.14}$$

where $0 \leq a \leq L$ and most of the case $L = 255$. It is simple to show that f can be reconstructed from its thresholded binary images. That is

$$f(x) = \sum_{a=1}^{L} f_a(x) \tag{6.4.15}$$

$$= \max \{ a \mid f_a(x) = 1 \}. \tag{6.4.16}$$

A transformation Ψ obeys the *threshold-linear superposition* [60] when it satisfies:

$$\Psi(f) = \sum_{a=1}^{L} \Psi(f_a). \tag{6.4.17}$$

Such transformation Ψ can be realized by decomposing f into all its binary images f_a, processing each thresholded image by Ψ, and creating the output $\Psi(f)$ by summing up the processed f_a.

Theorem 6.4.2: The soft morphological operations obey the threshold-linear superposition. That is

$$f \oplus [B, A, k] = \sum_{a=1}^{L} \left[f_a \oplus [B, A, k] \right]. \tag{6.4.18}$$

[Proof]: We only prove the threshold-linear superposition for soft morphological dilation.

$$f \oplus [B, A, k] = \sum_{a=1}^{L} X_a(f \oplus [B, A, k]) \tag{6.4.19}$$

$$= \sum_{a=1}^{L} \Big[X_a(f) \oplus [B, A, k] \Big] \tag{6.4.20}$$

$$= \sum_{a=1}^{L} \Big[f_a \oplus [B, A, k] \Big] \tag{6.4.21}$$

□

The theorem 6.4.2 is very useful in the sense that the gray scale input image is thresholded at each gray level a to get f_a followed by the binary soft morphological operations for f_a, and then summing up all f_a's that is exactly the same as the gray scale soft morphological operations we performed. An example is shown in Fig. 6.2 such that the soft morphological dilation obeys the threshold superposition, where A_1 and B are shown in Fig. 6.1.

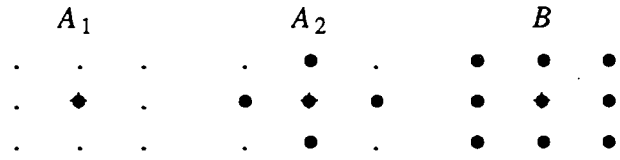


Fig. 6.1 Two core sets A_1, A_2, and the structuring element B.

6.5 Idempotent Soft Morphological Filters

An idempotent filter maps a class of input signals into an associated set of root sequences. Each of these root signals are invariant to additional filter passes. Since the nature of the soft morphological operations and their fair nonlinearity, idempotency usually dose not exist in such operations at the first stage. Koskinen *et al.* [44, 45] constructed a special class of idempotent soft morphological filters under the constraints that A only contains the origin of B and $k = Card(B) - 1$. In this section, we will show more general class of two-dimensional soft morphological closing and opening.

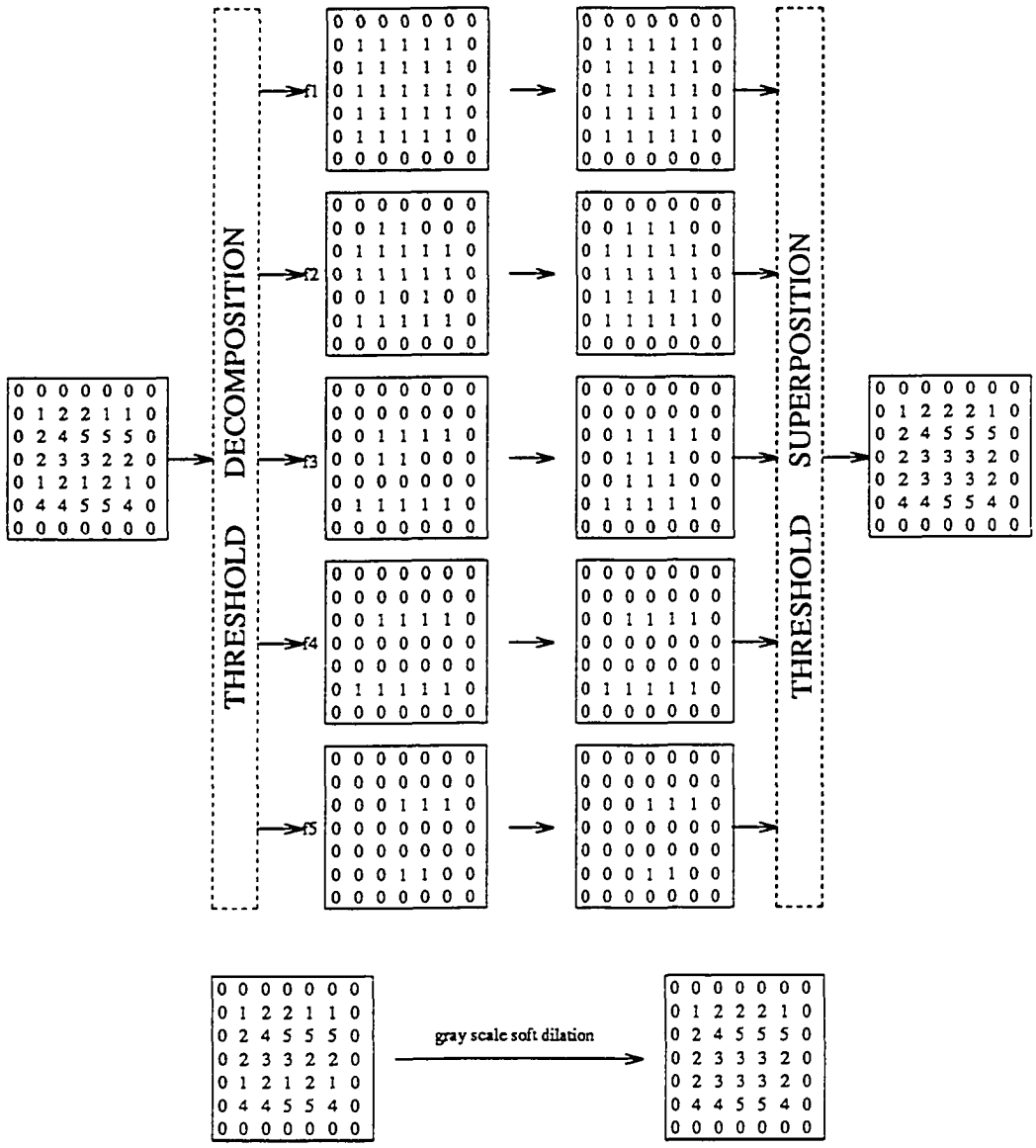


Fig. 6.2 The soft morphological dilation obeys threshold superposition.

Proposition 6.5.1: The soft morphological closing fills up the valley whose area is less than or equal to $Card(A)$, where $k = Card(B \setminus A)$.

[Proof]: According to Proposition 6.4.4, the soft morphological dilation fills up the valley whose area is less than or equal to $Card(A)$. The result followed by a soft morphological erosion will not be changed in those filled areas. □

Proposition 6.5.2: The soft morphological closing suppresses the local maximum where $k = Card(B \setminus A)$.

[Proof]: After the soft morphological dilation, the local maximum will be propagated to all the pixels within the core set A. According to Proposition 6.4.5, followed by the soft morphological erosion the peak will be eliminated. □

We have observed that when k is equal to $Card(B \setminus A)$, the soft morphological closing becomes idempotent. For simplicity, only one-dimensional expressions are given. Let $D[X]$ and $E[X]$ denote the positive Boolean functions of binary soft morphological dilation and erosion respectively, where X is the input binary image. Let the core set $A = \{-a, -a+1, \cdots, -1, 0, 1, \cdots, a-1, a\}$ and the soft boundary $B \setminus A = \{-b, -b+1, \cdots, -a-1, a+1, \cdots, b-1, b\}$. Since $k = Card(B \setminus A)$, $D[X]$ can be represented as a logical function as

$$D[X](0) = \sum_{i \in A} X(i) + \prod_{j \in (B \setminus A)} X(j), \tag{6.5.1}$$

where $X(i)$ is the value of input image X at location i, "+" is the logic **OR**, and "·" is the logic **AND**. The binary soft morphological erosion can be expressed as

$$E[X](0) = \prod_{i \in A} X(i) \cdot \sum_{j \in (B \setminus A)} X(j). \tag{6.5.2}$$

Let $\Psi[X]$ be the soft morphological closing. Replacing $X(i)$ by $D[X](i)$ yields [17]

$$\Psi[X](0) = E[D[X]](0) \tag{6.5.3}$$

$$= \prod_{i\in A} D[X](i) \cdot \sum_{j\in(B\backslash A)} D[X](j) \tag{6.5.4}$$

$$= \prod_{i\in A}\Big(\sum_{i'\in A_i} X(i') + \prod_{j'\in(B\backslash A)_i} X(j')\Big) \cdot \sum_{j\in(B\backslash A)}\Big(\sum_{i'\in A_j} X(i') + \prod_{j'\in(B\backslash A)_j} X(j')\Big) \tag{6.5.5}$$

$$= \left[\Big(\sum_{i'\in A_{-a}} X(i') + \prod_{j'\in(B\backslash A)_{-a}} X(j')\Big) \cdots \Big(\sum_{i'\in A_a} X(i') + \prod_{j'\in(B\backslash A)_a} X(j')\Big)\right] \cdot$$

$$\left[\Big(\sum_{i'\in A_{-b}} X(i') + \prod_{j'\in(B\backslash A)_{-b}} X(j')\Big) + \cdots + \Big(\sum_{i'\in A_b} X(i') + \prod_{j'\in(B\backslash A)_b} X(j')\Big)\right] \tag{6.5.6}$$

$$= X(0)\Psi_0[X] + \Psi_1[X]\,. \tag{6.5.7}$$

For increasing filters, idempotence can be characterized in terms of sum-of-products expressions. The minimal sun-of-products of $\Psi[X]$ contain $X(0)$ are separated from those do not contain $X(0)$.

Theorem 6.5.1: For any f and finite sets B and A, the soft morphological closing is idempotent if $k = Card(B\backslash A)$.

[Proof]: Applying Ψ a second time in eq. (6.5.7) yields

$$\Psi[\Psi[X]](0) = \Psi[X](0)\Psi_0[\Psi[X]] + \Psi_1[\Psi[X]]. \tag{6.5.8}$$

Idempotence takes the forms

$$\Psi[X](0) = \Psi[X](0)\Psi_0[\Psi[X]] + \Psi_1[\Psi[X]]\,. \tag{6.5.9}$$

Let $\Psi[X](0)$ be p, $\Psi_0[\Psi[X]]$ be q and $\Psi_1[\Psi[X]]$ be r. The logical identity of eq. (6.5.9) is $p = pq + r$. If we can show $r = p$, then $pq + r = pq + p = p$. When $r = 1$, we have

$$\Psi_1[\Psi[X]] = 1 \quad \rightarrow \prod_{i\in(B\backslash A)_x} \Psi[X](i) = 1 \qquad x\in A. \tag{6.5.10}$$

$$\text{If } \Psi[X](0) = 0 \rightarrow D[X](0) = 0 \text{ and } \sum_{i \in A} D[X](i) = 0 \tag{6.5.11}$$

$$\rightarrow \sum_{i \in (A \cup P)} X(i) = 0 \tag{6.5.12}$$

where $P = \{y \mid some\ y \in (B \backslash A)\}$. But when eq. (6.5.12) is true, according to Propositions 6.4.4 and 6.4.5, there must be some i such that $\Psi[X](i) = 0$ where $i \in (B \backslash A)$ which is contradictory to eq. (6.5.10). It implies that $\Psi[X](0) = 1$. That is

$$\Psi_1[\Psi[X]] = 1 \quad \rightarrow \Psi[X](0) = 1 \tag{6.5.13}$$

$$\rightarrow p = 1 \quad \text{and} \quad r = p \tag{6.5.14}$$

$$\rightarrow pq + r = p \tag{6.5.15}$$

$$\rightarrow \Psi[\Psi[X]](0) = \Psi[X](0)\,. \tag{6.5.16}$$

If $r = 0$ and $q \geq p$ imply $pq = p$. When $\Psi_1[\Psi[X]] = 0$, according to Proposition 6.5.2 that the local maximum will be suppressed after closing, we have

$$\sum_{i \in (B \backslash \{0\})} \Psi[X](i) \geq \Psi[X](0) \rightarrow \Psi_0[\Psi[X]] \geq \Psi[X](0) \tag{6.5.17}$$

$$\rightarrow q \geq p \tag{6.5.18}$$

$$\rightarrow pq = p \tag{6.5.19}$$

$$\rightarrow \Psi[\Psi[X]](0) = \Psi[X](0) \tag{6.5.20}$$

□

Examples are given to show soft morphological closing is idempotent when $k = Card(B \setminus A)$. Core sets and structuring elements set are shown in Fig. 6.1. Input image is shown in Fig. 6.3. Results shown in Figs. 6.4 and 6.5 are the idempotent cases,

i.e. if the operation is applied again, the same results are obtained. Fig. 6.6 shows that when $k \neq Card(B \setminus A)$ it is not idempotent and results will be changed by iterations.

0	0	0	0	0	0	0
0	1	2	2	1	1	0
0	2	4	5	7	9	0
0	2	3	3	2	2	0
0	1	2	1	2	1	0
0	4	4	5	5	4	0
0	0	0	0	0	0	0

Fig. 6.3 The original image f.

0	0	0	0	0	0	0
0	1	2	2	1	1	0
0	2	4	5	7	7	0
0	2	3	3	2	2	0
0	1	2	2	2	1	0
0	4	4	5	5	4	0
0	0	0	0	0	0	0

Fig. 6.4 The result of $\Psi(f)$ with A_1, B, and k=8.

6.6 Implementations of Soft Morphological Filters

Implementations of soft morphological operations are discussed and presented in this section. When the input image is binary, the block diagram of implementing the soft morphological dilation using eq. (6.2.1) is shown in Fig. 6.7. The idea of implementation is straightforward: provides desired signals in the neighborhood of each pixel x with repetition k time of signals in A_x, sorts these signals, and picks up the kth largest one.

0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	4	5	7	7	0
0	0	3	4	5	2	0
0	0	4	4	4	1	0
0	0	4	5	5	4	0
0	0	0	0	0	0	0

Fig. 6.5 The result of $\Psi(f)$ with A_2, B, and k=4.

The neighborhood generator will fetch the signals in the neighborhood of $x \in X$ in B_x and produce the output of signals in $(B \backslash A)_x$ and signals in A_x with repetition k times of. The binary sorter [11] is implemented by a pipelined multistage binary sorting network consisting of one-bit compare-and-swap elements which contain an AND gate and an OR gate. Fig. 6.8 shows the 5-bit binary sorter and the compare-and-swap (CS) element. The CS element sorts two binary inputs X_0 and X_1 in the descendent order to be $Y_0 \geq Y_1$. The truth table of CS elements illustrated what outputs will be produced with given inputs is shown in Fig. 6.9.

An example is given to show how the binary sorter works. The input of a 5-bit binary sorter is 01011 and the output will be 11100. The binary sorter in Fig. 6.7 sorts totally $Card(B \backslash A) + k \times Card(A)$ binary signals in the descendent order, and the kth largest signal will be output after the selection of the kth input by a multiplexer.

The improved block diagram of the soft morphological dilation using eq. (6.4.1) is shown in Fig. 6.10. A parallel counter is used to count the total number of ones in the $Card(B \backslash A) + k \times Card(A)$ input bits. The comparator will output 1 if the result of the parallel counter is greater than or equal to k and output 0 otherwise.

In the specific case of k equal to $Card(B \backslash A)$, the implementation can be much

0	0	0	0	0	0	0
0	0	1	2	1	0	0
0	1	2	3	2	1	0
0	2	3	3	3	2	0
0	2	3	3	3	2	0
0	0	2	3	2	0	0
0	0	0	0	0	0	0

(a)

0	0	0	0	0	0	0
0	0	0	1	0	0	0
0	0	2	2	2	0	0
0	1	2	3	2	1	0
0	0	2	3	2	0	0
0	0	0	2	0	0	0
0	0	0	0	0	0	0

(b)

Fig. 6.6 The result of (a) $\Psi[f\,]$ and (b) $\Psi[\Psi[f\,]]$ with A_1, B, and k=4.

easier. The block diagram of the soft morphological dilation and erosion in this case is shown in Fig. 6.11, and the logic-gate implementation of the dilation and erosion modules is shown in Fig. 6.12.

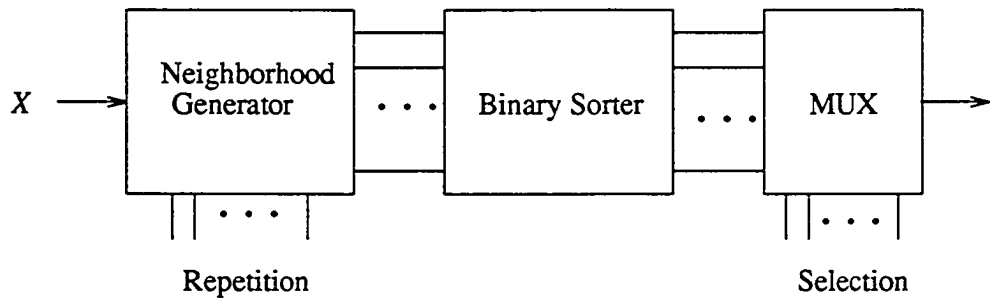


Fig. 6.7 The block diagram of the soft morphological dilation.

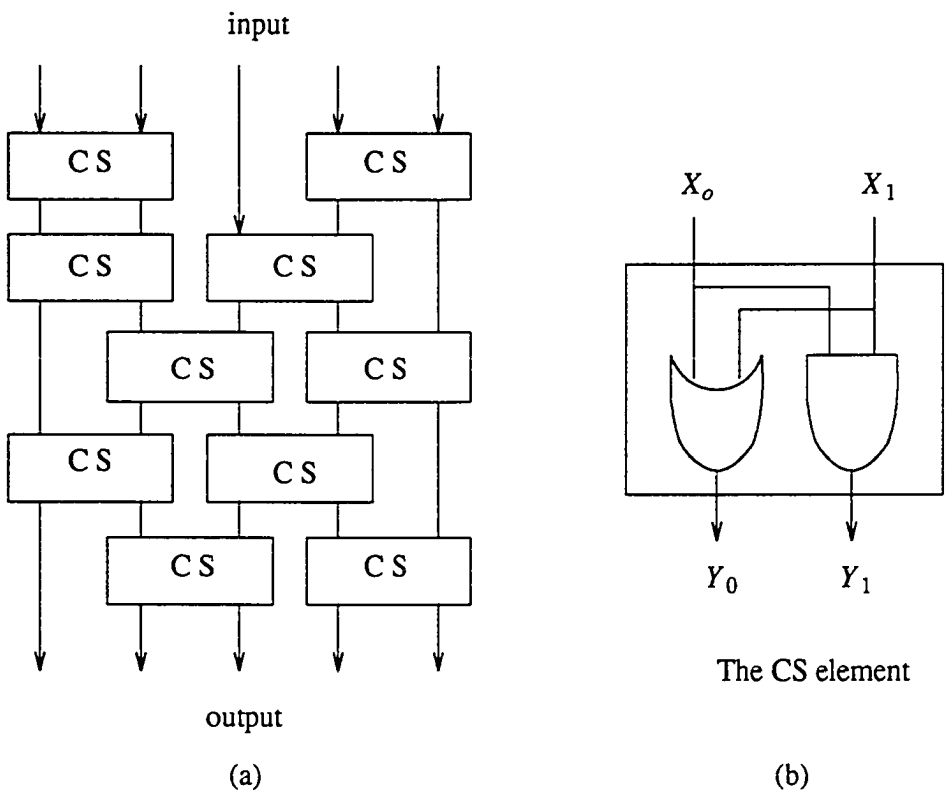


Fig. 6.8 (a) The 5-bit binary sorter, (b) the compare-and-swap (CS) element.

6.7 Summary

The important properties are discovered that soft morphological operations commute with threshold decomposition, obey threshold-linear superposition. The conditions for

Input		Output	
X_0	X_1	Y_0	Y_1
0	0	0	0
0	1	1	0
1	0	1	0
1	1	1	1

Fig. 6.9 The truth table of CS elements.

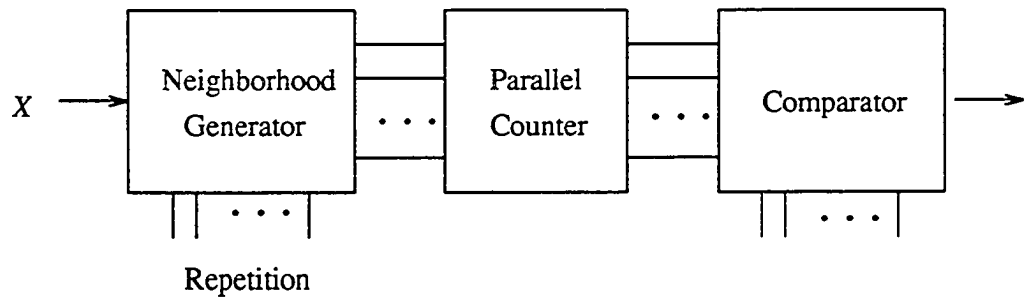


Fig. 6.10 The improved block diagram of the soft morphological dilation.

the idempotency property to be hold are also developed. New definitions soft morphological operations of sets by sets and functions by functions are given. The soft morphological operations commute with threshold decomposition and obey threshold-linear superposition which will lead us the fast implementation by using soft morphological operations of sets by sets instead of functions by sets or functions by functions operations. Thus the implementation and analysis of function-processing soft morphological operations can be done by focusing only on the case of sets which not only are much easier to deal with because their definitions involve only counting the points instead of sorting

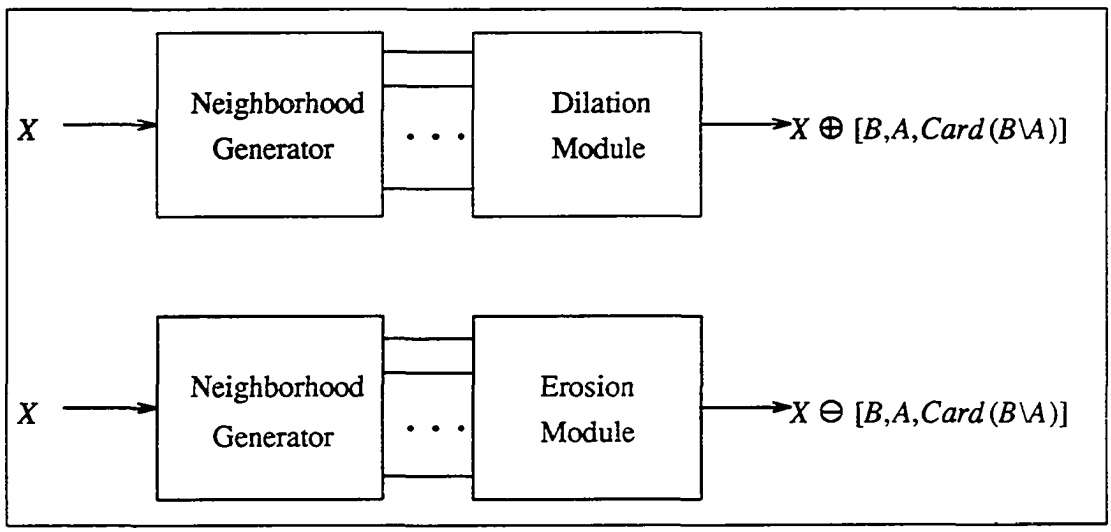
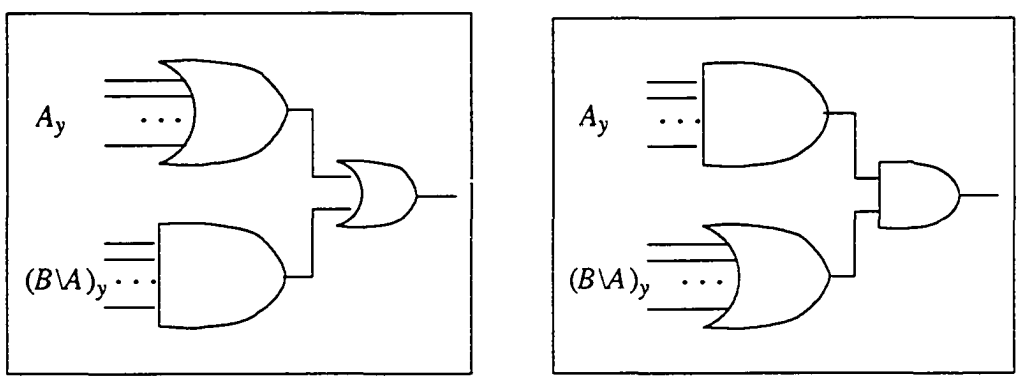


Fig. 6.11 The block diagram of soft dilation and erosion with $k = Card(B \backslash A)$.



Dilation Module Erosion Module

Fig. 6.12 The logic-gate implementation of the dilation and erosion modules.

numbers, but also allow logic gates implementation and parallel pipelined architecture leading to real-time implementation. The idempotence property gives us the idea of how to choose the structuring element sets and the value of k such that the soft morphological filters will reach the root signals without iterations.

CHAPTER VII

GRAY-SCALE SOFT MATHEMATICAL MORPHOLOGY

Gray-scale soft mathematical morphology is the extension of binary soft mathematical morphology which is found to be less sensitive to additive noise and to small variations. In this chapter, binary soft morphological operations are reviewed and the definitions of gray-scale soft morphological operations are given. The properties of gray-scale soft morphological operations are developed. It has been shown that the soft morphological operations of functions by sets or functions can commute with thresholding and obey threshold superposition. The threshold superposition property allows gray-scale signals to be decomposed into multiple binary signals which can be processed by only logic gates in parallel and then combined binary results to produce the equivalent output.

7.1 Introduction

Mathematical morphology which is based on set theory, provides an algebraic approach to manifest structuring shapes on binary or gray-scale images [74, 76]. Morphological filters constitute a highly nonlinear system. Another popular family of nonlinear filters is the order statistic filters which are based on order statistics [16] and have been applied to signal detection and image enhancement. Maragos and Schafer [57] extended the theory of median, order statistic and stack filters to mathematical morphology. They have shown that the order statistic filters can be used in both function- and set-processing and can commute with thresholding.

Soft morphological filters are the combination of the order statistic filters and morphological filters. The primary difference from standard morphological filters is that the maximum and minimum operations are replaced by the more general weighted order

statistics and the "soft" boundary is added to the structuring element. Koskinen *et al.* [44, 45] have shown that soft morphological operations are less sensitive to additive noise and to small variations in object shape and preserve most of the desirable properties of standard morphological operations.

In this chapter, the properties of threshold superposition and the new definitions of gray-scale soft morphological operations are presented. The soft morphological operations are proved to be capable of commuting with thresholding, and the property of threshold superposition allows fast implementation of function-processing soft morphological operations by using only logic gates. The implementation and the analysis of function-processing soft morphological operations can be interpreted by focusing only on the case of sets that are much easier to deal with since set-processing soft morphological operations only involve in counting the number of pixels instead of sorting the values.

The chapter is organized as follows. In section 7.2 the review of soft morphological operations is given. In section 7.3 the gray-scale soft morphological operations are defined. In section 7.4 the threshold decomposition for gray-scale soft morphological dilation is discussed. In section 7.5 the threshold decomposition for gray-scale soft morphological erosion is presented. Finally, conclusions are given.

7.2 Review of Soft Morphological Operations

The definitions of soft morphological operations of functions by sets are given as follows, where f is a gray-scale image and B is a flat structuring elements.

Definition 7.2.1: The soft morphological dilation of f by $[B, A, k]$ is defined as

$$(f \oplus [B,A,k])(x) = k\text{th largest of } (\{k \lozenge f(a) \mid a \in A_x\} \cup \{f(b) \mid b \in (B \backslash A)_x\}). \tag{7.2.1}$$

Definition 7.2.2: The soft morphological erosion of f by $[B, A, k]$ is defined as

$$(f \ominus [B,A,k])(x) = k\text{th smallest of } (\{k \lozenge f(a) \mid a \in A_x\} \cup \{f(b) \mid b \in (B \backslash A)_x\}). \tag{7.2.2}$$

Definition 7.2.3: The soft morphological closing of f by $[B, A, k]$ is defined as

$$f \bullet [B,A,k] = f \oplus [B,A,k] \ominus [B,A,k]. \tag{7.2.3}$$

Definition 7.2.4: The soft morphological opening of f by $[B, A, k]$ is defined as

$$f \circ [B,A,k] = f \ominus [B,A,k] \oplus [B,A,k]. \tag{7.2.4}$$

The soft morphological operations of sets by sets are defined as follows.

Definition 7.2.5: The soft morphological dilation of X by $[B, A, k]$ is defined as

$$X \oplus [B,A,k] = \{y \mid k \times Card(X \cap A_y) + Card(X \cap (B \setminus A)_y) \geq k\}. \tag{7.2.5}$$

Definition 7.2.6: The soft morphological erosion of X by $[B, A, k]$ is defined as

$$X \ominus [B,A,k] = \{y \mid k \times Card(X \cap A_y) + Card(X \cap (B \setminus A)_y) \geq N + (k-1) \times n - k + 1\}. \tag{7.2.6}$$

Referring to eq. (7.2.5) any one pixel within A_x is one or the number of ones within $(B \setminus A)_x$ is greater or equal to k will generate an one to the output. According to eq. (7.2.6) each pixel within A_x is one and the number of ones within $(B \setminus A)_x$ is greater or equal to $N-n-k+1$ (i.e. the number of zeros is greater than or equal to k) will generate an one to the output. The parallel counter and comparator are easy to implement by logic gates which will not be discussed here. The logic-gate implementation of soft dilation and erosion is illustrated in Fig. 7.1 where parallel counter counts the number of ones of input signals and comparator outputs one when the number outputted from parallel counter is greater than or equal to the index k. The general properties of soft morphological filters which are valid for both sets by sets and functions by sets.

7.3 Gray-Scale Soft Morphological Operations

Serra [75] used the cross section of the signal to generalize the morphological filtering of

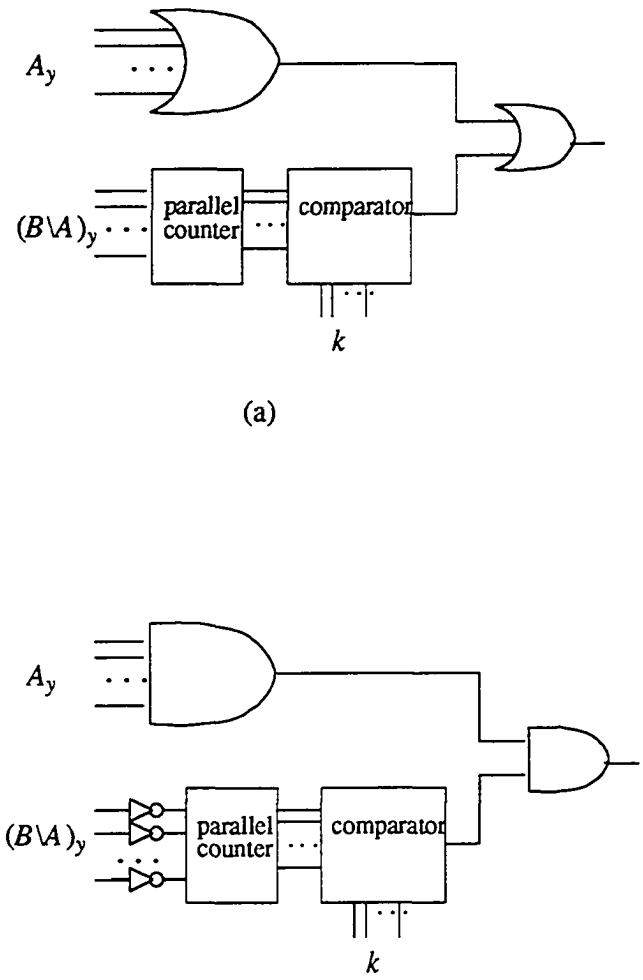


Fig. 7.1 The logic-gate implementation of soft dilation and erosion.

gray-scale signals. Sternberg [95] further generalized morphological filters by considering gray-scale images as surfaces of 3-D volumes. The standard binary morphological operations of dilation, erosion, opening, and closing are all naturally extended to gray-scale by the use of a *min* and *max* operation [95]. Similarly, the gray-scale soft morphological operations are the extension of the binary soft morphological operations. The definitions of soft morphological operations of functions by functions are defined as follows, where f is the input gray-scale image and α and β are the gray-scale structuring

element sets. Let the symmetric function $g^s(x)$ with respect to the origin be given by

$$g^s(x) = g(-x). \tag{7.3.1}$$

If F is the domain of f, B is the domain of β, and A is the domain of α. The definitions of soft morphological operations of functions by functions are given as follows.

Definition 7.3.1: The soft dilation of f by $[\beta, \alpha, k]$ is

$$f \oplus [\beta,\alpha,k](z) = k\text{th largest of } (\{ k \lozenge (f(y) + \alpha(z-y)\} \cup \{f(b) + \beta(z-b)\}), \tag{7.3.2}$$

where $z - y \in A$ and $z - b \in B \setminus A$.

Definition 7.3.2: The soft erosion of f by $[\beta, \alpha, k]$ is

$$f \ominus [\beta,\alpha,k](z) = k\text{–th smallest of } (\{ k \lozenge (f(y) - \alpha(z+y)\} \cup \{f(b) - \beta(z+b)\}), \tag{7.3.3}$$

where $z + y \in A$ and $z + b \in B \setminus A$.

Definition 7.3.3: The soft closing of f by $[\beta, \alpha, k]$ is

$$f \bullet [\beta,\alpha,k](x) = f \oplus [\beta,\alpha,k] \ominus [\beta,\alpha,k](x). \tag{7.3.4}$$

Definition 10: The soft opening of f by $[\beta, \alpha, k]$ is

$$f \circ [\beta,\alpha,k](x) = f \ominus [\beta,\alpha,k] \oplus [\beta,\alpha,k](x). \tag{7.3.5}$$

One of the most important links between sets and functions is the *umbra* which is introduced by Sternberg [95]. The *umbra* of a function f is defined as

$$U[f] = \{ (x, y) | 0 \le y \le f(x) \}, \tag{7.3.6}$$

where only positive value is considered. The *top surface* of a set A is defined as

$$T[A](x) = \max\{ y | (x, y) \in A \}. \tag{7.3.7}$$

Figs. 7.2 and 7.3 illustrate the top surface and unbra functions respectively.

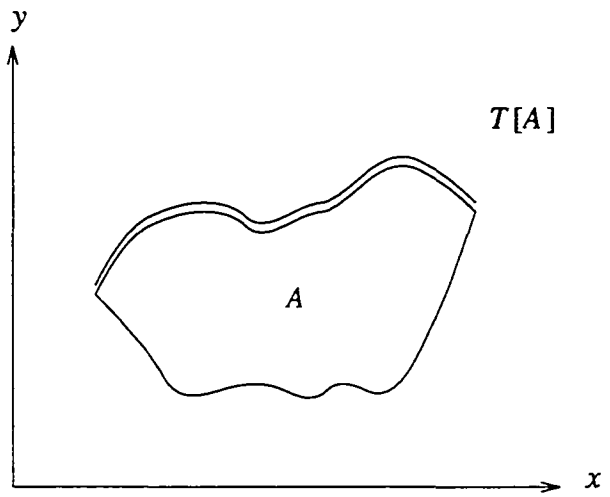


Fig. 7.2 The top surface $T[A]$ of a set A.

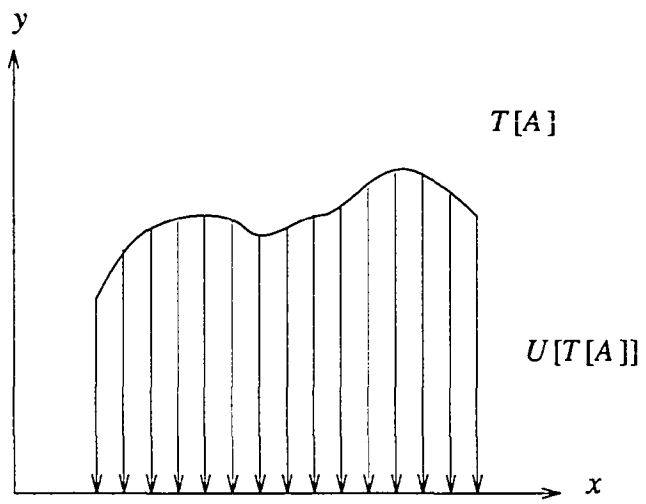


Fig. 7.3 The umbra $U[T[A]]$ of a function $T[A]$.

After the operations of taking a top surface of a set and an umbra of a surface are defined, we have the following theorems.

Theorem 7.3.1: The soft morphological dilation of functions by functions is the surface of the soft dilation of their umbras. That is

$$f \oplus [\beta, \alpha, k] = T\{ U[f] \oplus [U^s[\beta], U^s[\alpha], k]\}. \tag{7.3.8}$$

[Proof]: Let $X = U[f]$, $B = U^s[\beta]$, and $A = U^s[\alpha]$ (i.e., $B^s = U[\beta]$ and $A^s = U[\alpha]$). The soft morphological dilation of functions by functions can be derived as follows.

$$U[f] \oplus [U^s[\beta], U^s[\alpha], k](z) = X \oplus [B, A, k](z) \tag{7.3.9}$$

$$= k\text{th largest of} \left\{ \{k \Diamond X(a) \,|\, a \in A_z\} \cup \{X(b) \,|\, b \in (B \backslash A)_z\} \right\}. \tag{7.3.10}$$

By applying the umbra operation on eq. (7.3.9), we have

$$T(U[f] \oplus [U^s[\beta], U^s[\alpha], k]) \tag{7.3.11}$$

$$= k\text{th largest of} \left\{ \{ k \Diamond (f(x) + \alpha(z-x)\} \cup \{f(b) + \beta(z-b)\} \right\} \tag{7.3.12}$$

$$= f \oplus [\beta, \alpha, k]. \tag{7.3.13}$$

This is because $T[X](a) = f(x) + \alpha(z - x)$ (as shown in Fig. 7.4) where $x \in A_z$ and $z - x \in A$. □

Similarly, we have the following theorem without proof.

Theorem 7.3.2: The soft morphological erosion of functions by functions is the surface of the soft erosion of their umbras. That is

$$f \ominus [\beta, \alpha, k] = T\{ U[f] \ominus [U^s[\beta], U^s[\alpha], k]\}. \tag{7.3.14}$$

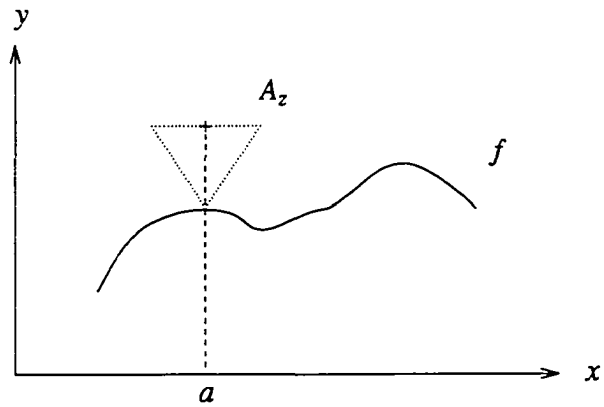


Fig. 7.4 $T[X](a) = f(x) + \alpha(z-x)$ where $a \in A_z$.

7.4 Threshold Decomposition for Soft Morphological Dilation

In this section, we will show that the soft morphological operations of functions by functions commute with thresholding and obey the threshold superposition. Let a *slice* of a function f, denoted by $S[f]$, be defined as [84]

Definition 7.4.1: A slice of a function f is

$$S[f_i](x,y) = \begin{cases} 1 & \text{if } y = i \text{ and } f(x) \geq y \\ 0 & \text{otherwise.} \end{cases} \tag{7.4.1}$$

Let I and J be the maximal gray-scale value of f and β respectively, i.e., $I = \max\{ a \,|\, f_a = 1 \}$ and $J = \max\{ a \,|\, \alpha_a = 1 \text{ or } \beta_a = 1 \}$. The umbra of a function f can be decomposed into I slices. That is

$$U[f] = \bigcup_{i=0}^{I} S[f_i]. \tag{7.4.2}$$

From Theorem 7.3.1, we have

$$f \oplus [\beta, \alpha, k] = T\{ U[f] \oplus [U[\beta^s], U[\alpha^s], k] \} \tag{7.4.3}$$

$$= T\left\{ \bigcup_{i=0}^{I} S[f_i] \oplus [U[\beta^s], U[\alpha^s], k] \right\} \tag{7.4.4}$$

$$= T\left\{ \bigcup_{i=0}^{I} \left[S[f_i] \oplus [U[\beta^s], U[\alpha^s], k] \right] \right\} \tag{7.4.5}$$

$$= T\left\{ \bigcup_{i=0}^{I} \left[S[f_i] \oplus [\bigcup_{j=0}^{J} S[\beta^s{}_j], \bigcup_{j=0}^{J} S[\alpha^s{}_j], k] \right] \right\} \tag{7.4.6}$$

$$= T\left\{ \bigcup_{i=0}^{I} \bigcup_{j=0}^{J} \left[S[f_i] \oplus [S[\beta^s{}_j], S[\alpha^s{}_j], k] \right] \right\} \tag{7.4.7}$$

$$= \max_{i=0}^{I} T\left\{ \bigcup_{j=0}^{J} \left[S[f_i] \oplus [S[\beta^s{}_j], S[\alpha^s{}_j], k] \right] \right\}. \tag{7.4.8}$$

The result after applying eq. (7.4.8) is shown in Fig. 7.5. Based upon the definition of slice, each slice of input signal and structure element sets consists of only one non-zero row. It is possible for us to perform the binary soft dilation in one less dimension. The top surface operation in eq. (7.4.8) can be replaced by a summation of all reduced dimensionality stacking signals.

$$f \oplus [\beta, \alpha, k] = \max\left\{ \sum_{j=0}^{J} \left[f_0 \oplus [\beta^s{}_j, \alpha^s{}_j, k] \right] - 1, \right.$$

$$\left. \sum_{j=0}^{J} \left[f_1 \oplus [\beta^s{}_j, \alpha^s{}_j, k] \right], \sum_{j=0}^{J} \left[f_2 \oplus [\beta^s{}_j, \alpha^s{}_j, k] \right] + 1, \cdots \right\}. \tag{7.4.9}$$

Let the input signal and structuring element sets be non-negative. The first term in eq. (7.4.9) is a constant. That is

$$\sum_{j=0}^{J} \left[f_0 \oplus [\beta^s{}_j, \alpha^s{}_j, k] \right] - 1 = J. \tag{7.4.10}$$

According to eq. (7.4.10), eq. (7.4.9) can be rewritten as

$f \oplus [\beta, \alpha, 2] = 79877974$

f: 05432510 $\quad$ β: 12421 $\quad$ α: ␣ $\quad$ $k=2$

$S[f_0]$	$S[f_1]$	$S[f_2]$	$S[f_3]$	$S[f_4]$	$S[f_5]$	$S[\beta_0]$	$S[\beta_1]$	$S[\beta_2]$	$S[\beta_3]$	$S[\beta_4]$
00000000	00000000	00000000	00000000	00000000	01000100	00000	00000	00000	00000	11111
00000000	00000000	00000000	00000000	01100100	00000000	00000	00000	00000	01110	00000
00000000	00000000	00000000	01110100	00000000	00000000	00000	00000	01110	00000	00000
00000000	00000000	01111100	00000000	00000000	00000000	00000	00100	00000	00000	00000
00000000	01111110	00000000	00000000	00000000	00000000	00100	00000	00000	00000	00000
11111111	00000000	00000000	00000000	00000000	00000000					

$S[f_5]S[\beta_0]$

					$\bigcup_{j=0}^{4}[S[f_5] \oplus [S[\beta_j], S[\alpha_j], 2]]$
01000100	00000000	00000000	00000000	00000000	01000100
00000000	01000100	00000000	00000000	00000000	01000100
00000000	00000000	11101110	00000000	00000000	11101110
00000000	00000000	00000000	11101110	00000000	11101110
01000100	01000100	01000100	01000100	11111110	11111110
00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000

$S[f_4]S[\beta_0]$

					$\bigcup_{j=0}^{4}[S[f_4] \oplus [S[\beta_j], S[\alpha_j], 2]]$
00000000	00000000	00000000	00000000	00000000	00000000
01100100	00000000	00000000	00000000	00000000	01100100
00000000	01100100	00000000	00000000	00000000	01100100
00000000	00000000	11111110	00000000	00000000	11111110
00000000	00000000	00000000	11111110	00000000	11111110
01100100	01100100	01100100	01100100	11111110	11111110
00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000

Fig. 7.5 An example of illustrating eq. (7.4.8).

$S[f_3]S[\beta_0]$ $\bigcup_{j=0}^{4}[S[f_3] \oplus [S[\beta_j],S[\alpha_j],2]]$

00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000
01110100	00000000	00000000	00000000	00000000	01110100
00000000	01110100	00000000	00000000	00000000	01110100
00000000	00000000	11111110	00000000	00000000	11111110
00000000	00000000	00000000	11111110	00000000	11111110
01110100	01110100	01110100	01110100	11111110	11111110
00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000

$S[f_2]S[\beta_0]$ $\bigcup_{j=0}^{4}[S[f_2] \oplus [S[\beta_j],S[\alpha_j],2]]$

00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000
01111100	00000000	00000000	00000000	00000000	01111100
00000000	01111100	00000000	00000000	00000000	01111100
00000000	00000000	11111110	00000000	00000000	11111110
00000000	00000000	00000000	11111110	00000000	11111110
01111100	01111100	01111100	01111100	11111110	11111110
00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000

$S[f_1]S[\beta_0]$ $\bigcup_{j=0}^{4}[S[f_1] \oplus [S[\beta_j],S[\alpha_j],2]]$

00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000
01111110	00000000	00000000	00000000	00000000	01111110
00000000	01111110	00000000	00000000	00000000	01111110
00000000	00000000	11111111	00000000	00000000	11111111
00000000	00000000	00000000	11111111	00000000	11111111
01111110	01111110	01111110	01111110	11111111	11111111
00000000	00000000	00000000	00000000	00000000	00000000

$S[f_0]S[\beta_0]$ $\bigcup_{j=0}^{4}[S[f_0] \oplus [S[\beta_j],S[\alpha_j],2]]$

00000000	00000000	00000000	00000000	00000000	00000000	01000100
00000000	00000000	00000000	00000000	00000000	00000000	01100100
00000000	00000000	00000000	00000000	00000000	00000000	11111110
00000000	00000000	00000000	00000000	00000000	00000000	11111110
00000000	00000000	00000000	00000000	00000000	00000000	11111110
11111111	00000000	00000000	00000000	00000000	11111111	11111111
00000000	11111111	00000000	00000000	00000000	11111111	11111111
00000000	00000000	11111111	00000000	00000000	11111111	11111111
00000000	00000000	00000000	11111111	00000000	11111111	11111111
11111111	11111111	11111111	11111111	11111111	11111111	11111111

$\bigcup_{i=0}^{5}\bigcup_{j=0}^{4}[S[f_i] \oplus [S[\beta^s{}_j],S[\alpha^s{}_j],2]]$

Fig. 7.5 (Continued)

$$f \oplus [\beta, \alpha, k] = \max\left\{ J, \sum_{j=0}^{J}\left[f_1 \oplus [\beta^s{}_j, \alpha^s{}_j, k]\right], \right.$$

$$\left. \sum_{j=0}^{J}\left[f_2 \oplus [\beta^s{}_j, \alpha^s{}_j, k]\right] + 1, \cdots \right\}. \tag{7.4.11}$$

$$= \max\left\{ J, J + \left[f_1 \oplus [\beta^s{}_J, \alpha^s{}_J, k]\right] (J-1) + \right.$$

$$\left. \sum_{j=J-1}^{J}\left[f_2 \oplus [\beta^s{}_j, \alpha^s{}_j, k]\right] + 1, \cdots \right\}. \tag{7.4.12}$$

$$= J + \max\left\{ f_1 \oplus [\beta^s{}_J, \alpha^s{}_J, k], \sum_{j=J-1}^{J}\left[f_2 \oplus [\beta^s{}_j, \alpha^s{}_j, k]\right], \cdots \right\}. \tag{7.4.13}$$

Based upon the eq. (7.4.13), the threshold decomposition algorithm for soft morphological dilation of functions by functions is described as follows.

1. Determine the highest gray-scale values of the input signal and the structuring element sets, denoted by I and J respectively.
2. Assume $I \geq J$. Decompose f into I binary images $\{f_a \mid 1 < a \leq I\}$ and decompose β into $J + 1$ binary structuring element sets $\{\beta_a \mid 0 \leq a \leq J\}$.
3. Compute the binary results Y_{ia}, that is

$$Y_{ia} = f_i \oplus [\beta_a, \alpha_a, k] \text{ for } \begin{cases} J-i+1 \leq a \leq J & \text{if } 1 \leq i \leq J \\ 0 \leq a \leq J & \text{if } J < i \leq I. \end{cases} \tag{7.4.14}$$

4. Sum up all the stacking binary results to obtain Y_i.

$$Y_i = \begin{cases} \sum_{a=J-i+1}^{J} Y_{ia} & \text{if } 1 \leq i \leq J \\ \sum_{a=0}^{J} Y_{ia} & \text{if } J < i \leq I. \end{cases} \tag{7.4.15}$$

5. Compute the gray-scale result (Y) by selecting the maximum value of Y_i at each position x and then add J to every position.

$$Y(x) = \max Y_i(x) + J. \tag{7.4.16}$$

An example is given in Fig. 7.6 to show the threshold decomposition algorithm for soft morphological dilation of functions by functions. There are two cases encountered in applying the threshold decomposition algorithm: (i) $I > J$ and (ii) $I \leq J$.

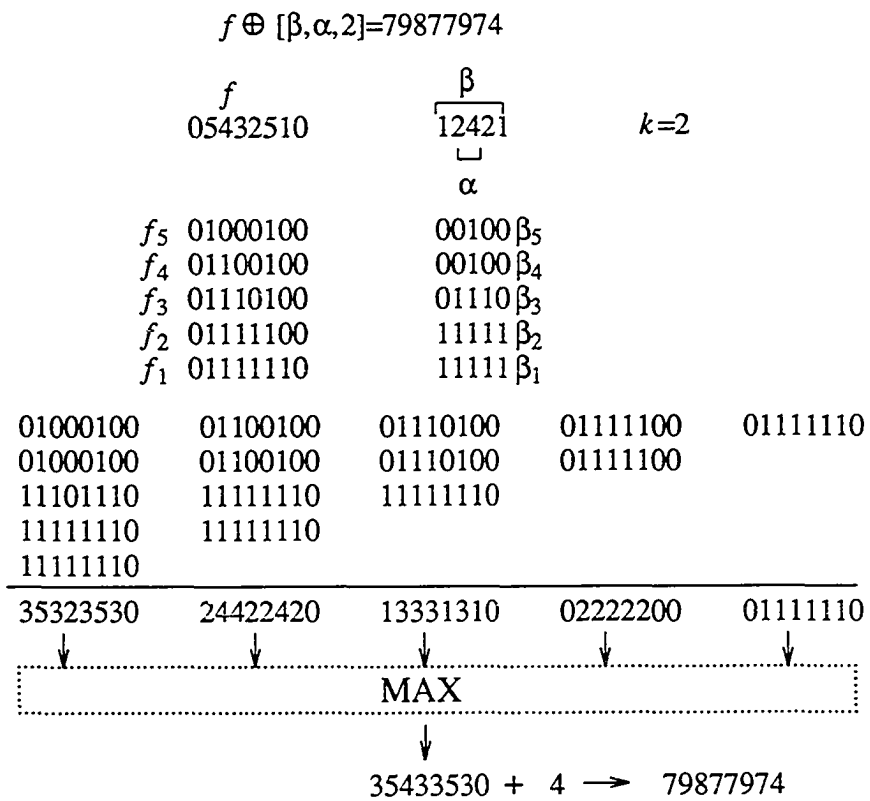


Fig. 7.6 Threshold decomposition for soft dilation of functions by functions.

Case (i): If $I > J$, eq. (7.4.13) becomes

$$f \oplus [\beta, \alpha, k] = J + \max \left\{ f_1 \oplus [\beta^s{}_J, \alpha^s{}_J, k], \sum_{j=J-1}^{J} \left[f_2 \oplus [\beta^s{}_j, \alpha^s{}_j, k] \right], \right.$$

$$\sum_{j=J-2}^{J}\left[f_3 \oplus [\beta^s{}_j, \alpha^s{}_j, k]\right], \cdots, \sum_{j=0}^{J}\left[f_{J+1} \oplus [\beta^s{}_j, \alpha^s{}_j, k]\right],$$

$$\sum_{j=0}^{J}\left[f_{J+2} \oplus [\beta^s{}_j, \alpha^s{}_j, k]\right] + 1, \cdots, \sum_{j=0}^{J}\left[f_I \oplus [\beta^s{}_j, \alpha^s{}_j, k]\right] + I - J - 1\Bigg\} \quad (7.4.17)$$

Case (ii): If $I \leq J$, eq. (7.4.13) becomes

$$f \oplus [\beta, \alpha, k] = J + \max\left\{ f_1 \oplus [\beta^s{}_J, \alpha^s{}_J, k], \sum_{j=J-1}^{J}\left[f_2 \oplus [\beta^s{}_j, \alpha^s{}_j, k]\right],\right.$$

$$\left.\sum_{j=J-2}^{J}\left[f_3 \oplus [\beta^s{}_j, \alpha^s{}_j, k]\right], \cdots, \sum_{j=J-I+1}^{J}\left[f_I \oplus [\beta^s{}_j, \alpha^s{}_j, k]\right]\right\}. \quad (7.4.18)$$

Note that it is time-consuming in implementation of the maximum selection over each position of all signals. By eliminating this drawback, a new implementation is presented as follows. From eq. (7.4.7), we have

$$f \oplus [\beta, \alpha, k] = T\left\{ \bigcup_{i=0}^{I}\bigcup_{j=0}^{J}\left[S[f_i] \oplus [S[\beta^s{}_j], S[\alpha^s{}_j], k]\right]\right\}. \quad (7.4.19)$$

Each subgroup from left to right in eq. (7.4.19) possesses the stacking property from bottom to top. The top surface of the union of those stacked subgroups is equal to a summation of the reduced-dimensionality stacking signals. Thus

$$f \oplus [\beta, \alpha, k]$$

$$= \left\{ \left[f_0 \oplus [\beta^s{}_0, \alpha^s{}_0, k]\right] + \left[f_1 \oplus [\beta^s{}_0, \alpha^s{}_0, k] \cup f_0 \oplus [\beta^s{}_1, \alpha^s{}_1, k]\right] + \cdots + \right.$$

$$\left[f_{J+1} \oplus [\beta^s{}_0, \alpha^s{}_0, k] \cup f_J \oplus [\beta^s{}_1, \alpha^s{}_1, k] \cup \cdots \cup f_1 \oplus [\beta^s{}_J, \alpha^s{}_J, k]\right]$$

$$+ \cdots + \left[f_I \oplus [\beta^s{}_{J-1}, \alpha^s{}_{J-1}, k] \cup f_{I-1} \oplus [\beta^s{}_J, \alpha^s{}_J, k]\right] +$$

$$\left\{ f_I \oplus [\beta^s{}_J, \alpha^s{}_J, k] \right\} - 1 \tag{7.4.20}$$

$$= \Big\{ 1 + 1 + \cdots + 1 +$$

$$\left[f_{J+1} \oplus [\beta^s{}_0, \alpha^s{}_0, k] + f_J \oplus [\beta^s{}_1, \alpha^s{}_1, k] + \cdots + f_1 \oplus [\beta^s{}_J, \alpha^s{}_J, k] \right]$$

$$+ \cdots + \left[f_I \oplus [\beta^s{}_{J-1}, \alpha^s{}_{J-1}, k] + f_{I-1} \oplus [\beta^s{}_J, \alpha^s{}_J, k] \right] +$$

$$\left\{ f_I \oplus [\beta^s{}_J, \alpha^s{}_J, k] \right\} - 1 \tag{7.4.21}$$

$$= \Big\{ (J+1) +$$

$$\left[f_{J+1} \oplus [\beta^s{}_0, \alpha^s{}_0, k] + f_J \oplus [\beta^s{}_1, \alpha^s{}_1, k] + \cdots + f_1 \oplus [\beta^s{}_J, \alpha^s{}_J, k] \right]$$

$$+ \cdots + \left[f_I \oplus [\beta^s{}_{J-1}, \alpha^s{}_{J-1}, k] + f_{I-1} \oplus [\beta^s{}_J, \alpha^s{}_J, k] \right] +$$

$$\left\{ f_I \oplus [\beta^s{}_J, \alpha^s{}_J, k] \right\} - 1 \tag{7.4.22}$$

$$= \Big\{ \left[f_{J+1} \oplus [\beta^s{}_0, \alpha^s{}_0, k] + f_J \oplus [\beta^s{}_1, \alpha^s{}_1, k] + \cdots + f_1 \oplus [\beta^s{}_J, \alpha^s{}_J, k] \right]$$

$$+ \cdots + \left[f_I \oplus [\beta^s{}_{J-1}, \alpha^s{}_{J-1}, k] + f_{I-1} \oplus [\beta^s{}_J, \alpha^s{}_J, k] \right] +$$

$$\left\{ f_I \oplus [\beta^s{}_J, \alpha^s{}_J, k] \right\} + J. \tag{7.4.23}$$

The implementation using OR gates and a Σ operation is shown in Fig. 7.7.

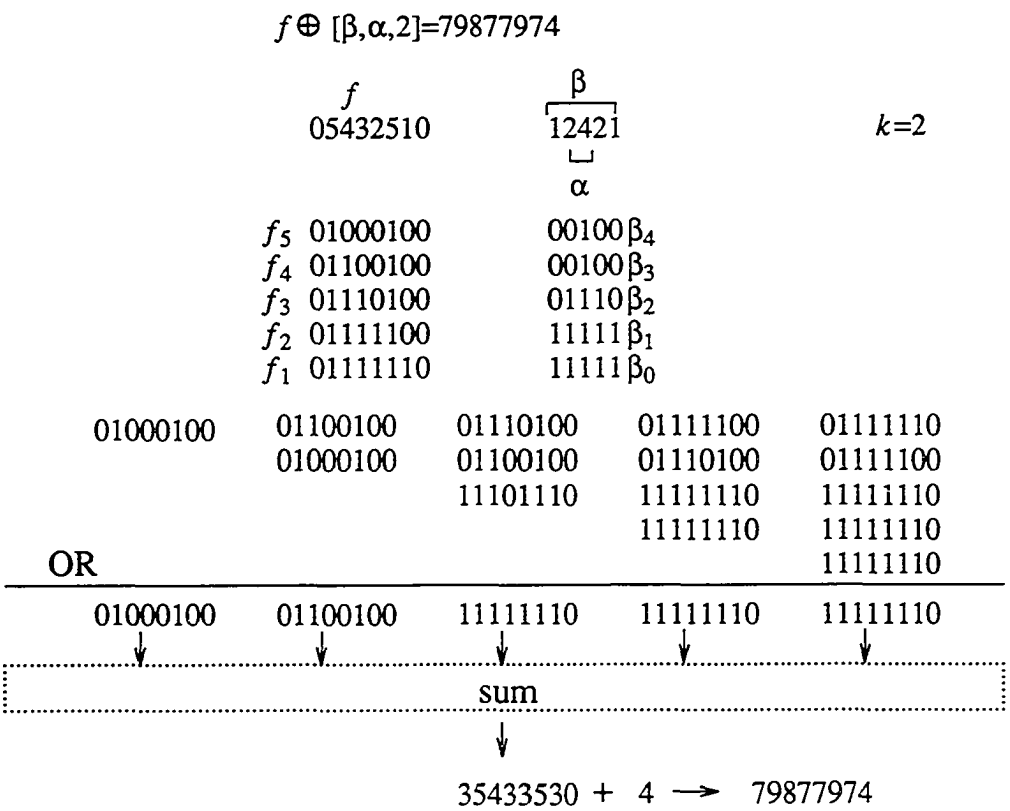


Fig. 7.7 New implementation of threshold decomposition for soft dilation.

7.5 Threshold Decomposition for Soft Morphological Erosion

The decomposition algorithm for soft erosion can be derived as follows. The *slice complement* [84] of a function f, denoted by $S'[f]$, is defined as

Definition 7.5.1: The slice complement of a function f is

$$S'[f_i](x,y) = \begin{cases} 0 & \text{if } y = i \text{ and } f(x) < y \\ 1 & \text{otherwise.} \end{cases} \tag{7.5.1}$$

The umbra of a function f can be decomposed into I slice complements. That is

$$U[f] = \bigcap_{i=0}^{I} S'[f_i]. \tag{7.5.2}$$

From Theorem 7.3.2, we have

$$f \ominus [\beta, \alpha, k] = T\left\{ U[f] \ominus [U^s[\beta], U^s[\alpha], k] \right\} \tag{7.5.3}$$

$$= T\left\{ \bigcap_{i=0}^{I} S'[f_i] \ominus \left[\bigcup_{j=0}^{J} S[\beta^s{}_j], \bigcup_{j=0}^{J} S[\alpha^s{}_j], k \right] \right\}. \tag{7.5.4}$$

$f \ominus [\beta, \alpha, k]$

$$= \Big\{ \Big[S'[f_0] \ominus [S[\beta^s{}_J], S[\alpha^s{}_J], k] \Big] \cap$$

$$\Big[S'[f_1] \ominus [S[\beta^s{}_J], S[\alpha^s{}_J], k] \cap S'[f_0] \ominus [S[\beta^s{}_{J-1}], S[\alpha^s{}_{J-1}], k] \Big] \cap \cdots \cap$$

$$\Big[S'[f_J] \ominus [S[\beta^s{}_J], S[\alpha^s{}_J], k] \cap S'[f_{J-1}] \ominus [S[\beta^s{}_{J-1}], S[\alpha^s{}_{J-1}], k] \cap \cdots \cap$$

$$S'[f_0] \ominus [S[\beta^s{}_0], S[\alpha^s{}_0], k] \Big] \cap \cdots \cap$$

$$\Big[S'[f_I] \ominus [S[\beta^s{}_J], S[\alpha^s{}_J], k] \cap S'[f_{I-1}] \ominus [S[\beta^s{}_{J-1}], S[\alpha^s{}_{J-1}], k] \cap \cdots \cap$$

$$S'[f_{I-J}] \ominus [S[\beta^s{}_0], S[\alpha^s{}_0], k] \Big] \cap \cdots \cap \Big[S'[f_I] \ominus [S[\beta^s{}_0], S[\alpha^s{}_0], k] \Big] \Big\} \tag{7.5.5}$$

$$= \Big\{ \Big[f_0 \ominus [\beta^s{}_J, \alpha^s{}_J, k] \Big] + \Big[f_1 \ominus [\beta^s{}_J, \alpha^s{}_J, k] \cap f_0 \ominus [\beta^s{}_{J-1}, \alpha^s{}_{J-1}, k] \Big] + \cdots +$$

$$\Big[f_J \ominus [\beta^s{}_J, \alpha^s{}_J, k] \cap f_{J-1} \ominus [\beta^s{}_{J-1}, \alpha^s{}_{J-1}, k] \cap \cdots \cap f_0 \ominus [\beta^s{}_0, \alpha^s{}_0, k] \Big]$$

$$+ \cdots +$$

$$\left[f_I \ominus [\beta^s_J, \alpha^s_J, k] \cap f_{I-1} \ominus [\beta^s_{J-1}, \alpha^s_{J-1}, k] \cap \cdots \cap f_{I-J} \ominus [\beta^s_0, \alpha^s_0, k] \right]$$

$$+ \cdots + \left[f_I \ominus [\beta^s_0, \alpha^s_0, k] \right] \Bigg\} - (J+1) \tag{7.5.6}$$

$$= \Bigg\{ 1 + \left[f_1 \ominus [\beta^s_J, \alpha^s_J, k] \right] + \cdots + \left[f_1 \ominus [\beta^s_J, \alpha^s_J, k] \cap f_0 \ominus [\beta^s_{J-1}, \alpha^s_{J-1}, k] \right]$$

$$+ \cdots +$$

$$\left[f_J \ominus [\beta^s_J, \alpha^s_J, k] \cap f_{J-1} \ominus [\beta^s_{J-1}, \alpha^s_{J-1}, k] \cap \cdots \cap f_1 \ominus [\beta^s_1, \alpha^s_1, k] \right]$$

$$+ \cdots +$$

$$\left[f_I \ominus [\beta^s_J, \alpha^s_J, k] \cap f_{I-1} \ominus [\beta^s_{J-1}, \alpha^s_{J-1}, k] \cap \cdots \cap f_{I-J} \ominus [\beta^s_0, \alpha^s_0, k] \right]$$

$$+ 0 + \cdots + 0 \Bigg\} - (J+1) \tag{7.5.7}$$

$$= \Bigg\{ \left[f_1 \ominus [\beta^s_J, \alpha^s_J, k] \right] + \cdots + \left[f_1 \ominus [\beta^s_J, \alpha^s_J, k] \cap f_0 \ominus [\beta^s_{J-1}, \alpha^s_{J-1}, k] \right]$$

$$+ \cdots +$$

$$\left[f_J \ominus [\beta^s_J, \alpha^s_J, k] \cap f_{J-1} \ominus [\beta^s_{J-1}, \alpha^s_{J-1}, k] \cap \cdots \cap f_1 \ominus [\beta^s_1, \alpha^s_1, k] \right]$$

$$+ \cdots +$$

$$\left\{ f_I \ominus [\beta^s{}_J, \alpha^s{}_J, k] \cap f_{I-1} \ominus [\beta^s{}_{J-1}, \alpha^s{}_{J-1}, k] \cap \cdots \cap f_{I-J} \ominus [\beta^s{}_0, \alpha^s{}_0, k] \right\}$$

$$- J. \tag{7.5.8}$$

The threshold decomposition of soft morphological erosion of functions by functions using eq. (7.5.8) is illustrated in Fig. 7.8.

7.6 Summary

In this chapter the gray-scale soft morphological operations are studied and their properties are discussed. The important properties are discovered that soft morphological operations commute with threshold decomposition, obey threshold superposition which will lead us the fast implementation by using binary soft morphological operations instead of gray-scale operations. Thus the implementation and analysis of the function-processing soft morphological operations can be done by focusing only on the case of sets which not only are much easier to deal with because their definitions involves only counting the points instead of sorting numbers, but also allow logic gates implementation and parallel pipelined architecture leading to real-time implementation. The implementations of soft morphological dilation and erosion by using threshold decomposition give much improvement in computation cost.

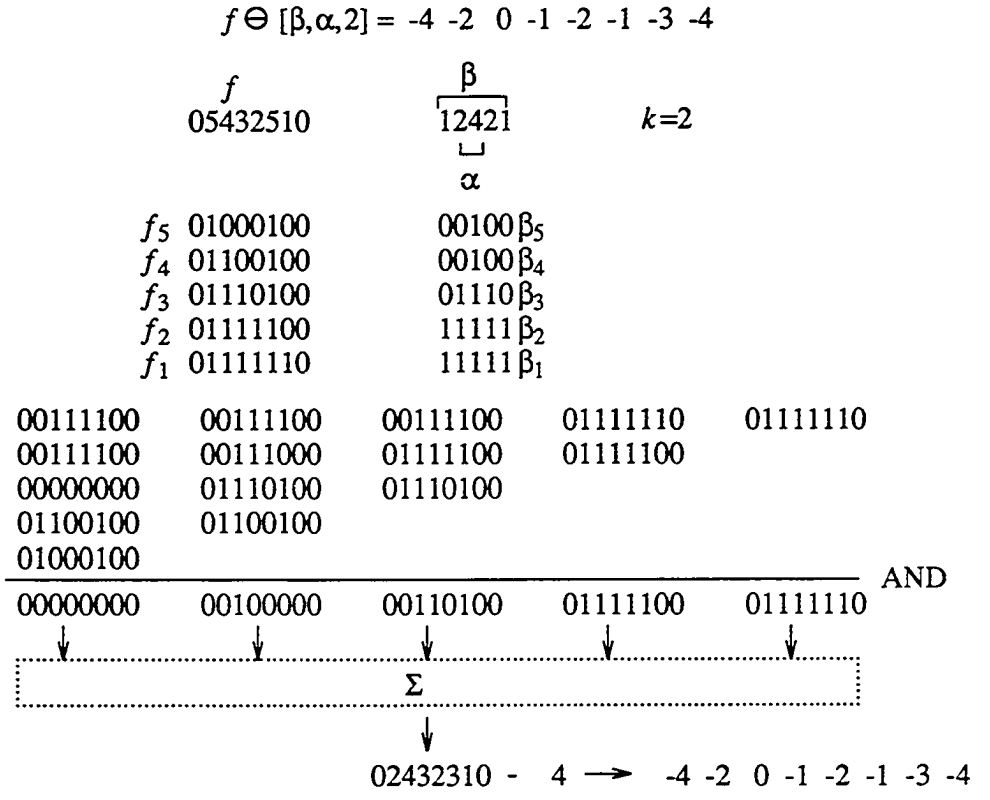


Fig. 7.8 Threshold decomposition for soft erosion of functions by functions.

CHAPTER VIII

SUMMARY AND FUTURE RESEARCH

This dissertation is aimed to investigate a powerful mathematical tool – mathematical morphology and to develop new useful morphological operations which can be applied to image analysis and object recognition. In the final Chapter we summarize the contributions of our research and briefly discuss the further potential research.

8.1 Contribution of this Dissertation

Our research is oriented toward aspects in both theory and application. Therefore, our contributions in this dissertation can be divided into two major parts: the theoretical research and the application research.

A. Theoretical Research

1. *Back-propagation morphology.* The new back-propagation morphological operations will compute the current scanning pixel results and simultaneously feed back the results to overwrite its input in order to affect the succeeding pixels' computation and to continue in the same way until all the pixels are scanned.
2. *Two-scan algorithm for roots of morphological operations.* The two-scan algorithm for reaching roots of morphological operations is based upon the back-propagation morphology. It needs only two scans without iteratively applying traditional morphological operations.
3. *G-spectrum.* The defined G-spectrum can not only be a shape descriptor for describing the quantified geometric features of multidimensional binary images, but also be a tool for shape recognition. We have proved that the G-spectrum can preserve the

invariance property for the transformations such as translation, rotation, and scaling. The G-spectrum can be easily extended and applied to multidimensional images.

4. *Soft mathematical morphology.* Soft morphological filters are the combination of the order statistic filters and morphological filters. The primary difference to standard morphological filters is that the maximum and minimum operations are replaced by more general weighted order statistics and the soft boundary is added to the structuring element. It have been shown that soft morphological operations are less sensitive to additive noise and to small variations in object shape, and they preserve most of the desirable properties of standard morphological operations.
5. *Gray-scale soft mathematical morphology.* The new definitions of soft morphological operations of functions by functions are given. Their properties are investigated and discussed.
6. *Threshold decomposition of soft mathematical morphology.* It is discovered that both functions by functions and functions by sets of soft mathematical operations commute with thresholding and obey threshold superposition. This property allows logic-gate implementation which leads to real time processing.
7. *Idempotent property of soft morphological filters.* It is shown that under some conditions the soft morphological filters can be idempotent.

B. Applications Research

1. *The maxima-tracking skeletonization.* This skeletonization algorithm is based upon Euclidean distance function and uses the sequential maxima-tracking method. The skeleton generated is connected and composed of simple digital components (arcs). With some slight modification, the skeleton generated is eliminated the non-significant short skeletal branches.
2. *Distance transformation using the two-scan algorithm.* A new approach for

generating distance functions is by using two-scan algorithm which is based upon back-propagation erosion. Using only two scans instead of iterations the two-scan distance transformation algorithm can save significant computation time.

3. *Shape recognition algorithm using G-spectrum.* The shape recognition using G-spectrum is presented. The experimental results are satisfied.
4. *Logic-gate implementation of soft morphological filters.* The logic-gate implementations of soft morphological filters using the properties of threshold decomposition and threshold superposition are developed.
5. *Threshold decomposition algorithms of soft morphological dilation and erosion of functions by functions.* The threshold decomposition algorithms of gray-scale soft morphology into binary soft morphology are developed which allows gray-scale operations be processed by only sets processing.

8.2 Future Research

The shape recognition algorithm using G-spectrum is presented. Refining the algorithm by real image, discussing the tolerance, and defining the thresholding value of equivalence should be investigated further. The practical system for shape recognition that uses the G-spectrum will be the future research topic.

We have shown some properties of soft mathematical morphology of both binary and gray-scale. It still exists other properties which should be investigated and discussed. The threshold decomposition algorithms for soft morphological dilation and erosion are presented. The VLSI implementations of these algorithms will be one of the future research topic which makes these theoretical issues more practical. Applications using soft morphological filters are other future research topics.

8.3 Epilogue

From the research of this dissertation, morphological operations are studied, new algorithms are developed, and new theorems are discovered. The maxima-tracking skeletonization algorithm is proposed to generate skeletons which are connected, composed of simple arcs, and equidistant. G-spectrum are proved to be a useful tool not only in shape description but also in shape recognition. Shape recognition algorithm is presented to recognition shape in the model-based approach. Soft mathematical morphology is the combination of order statistics and mathematical morphology. It was introduced and investigated in these two years. New definitions of soft morphological operations of sets by sets are introduced. Threshold decomposition and superposition properties of soft morphological operations lead to a faster implementation by using binary soft morphological operations consisting of only logic gates. The idempotency property gives us an idea in choosing the suitable structuring element sets and the value of index k, such that the soft morphological filters will produce the root signals without iterations.

My original goal in this dissertation was to study the properties and applications of mathematical morphology. The final result is that this dissertation has discovered G-spectrum and soft morphological filters, developed shape recognition algorithm and threshold decomposition for soft morphological dilation and erosion, and presented the implementations of soft morphological filters. I strongly hope that this dissertation will be useful for problems in skeletonization, shape recognition, non-linear filtering, and morphological filtering.

REFERENCES

[1] L. Abbott, R. Haralick and X. Zhuang, "Pipeline architectures for morphologic image analysis," *Journal of Machine Vision and Applications* , vol. 1, pp. 23-40, 1988.

[2] C. Arcelli and G. Sanniti di Baja, "A one-pass two-operation process to detect the skeletal pixels on the 4-distance transform," *IEEE Trans. Pattern Analysis and Machine Intelligence,* pp. 411-414, April 1989.

[3] S. Behrens and J. Dengler, "Analysing the structure of medical images with morphological size distributions," *Proc. Inter. Conf. Pattern Recognition*, pp. 886-890, 1990.

[4] 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: MIT Press, 1967, pp. 362-380.

[5] G. Borgefors, "Distance transformations in arbitrary dimensions," *Computer Vision, Graphics, and Image Processing,* 27, pp. 321-345, 1984.

[6] G. Borgefors, "Distance transformations in digital images," *Computer Vision, Graphics, and Image Processing,* 34, pp. 344-371, 1986.

[7] K. L. Boyer, A. J. Vayda, and A. C. Kak, "Robotic manipulation experiments using structural stereopsis for 3D vision," *IEEE Expert*, pp. 73-94, Fall 1986.

[8] J. Bronskill and A. Venetsanopoulos, "Multidimensional shape description and recognition using mathematical morphology," *Journal of Intell. and Robotics Sys.*, vol. 1, pp. 117-143, 1988.

[9] C. Chu and E. Delp, "Impulsive noise suppression and background normalization of electrocardiogram signals using morphological operators," *IEEE Trans. Biomedical Engineering*, vol. 36, no. 2, pp. 262-273, 1989.

[10] M. Chen and P. Yan, "A multiscaling approach based on morphological filtering," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 11, no. 7, pp. 694-700, 1989.

[11] L. Chang and S. Yu, "A new implementation of generalized order statistics filter by threshold decomposition," *IEEE Trans. Signal Processing*, vol. 40, no. 12, pp. 3062-3066, Dec. 1992.

[12] S. Crabtree, L. Yuan and R. Ehrlich, "A fast and accurate erosion-dilation method suitable for microcomputer," *CVGIP: Graph. Model Image Proc.*, vol. 53, no. 3, pp. 283-290, May 1991.

[13] A. Darwish and A. Jain, "A rule based approach for visual pattern inspection," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 10, no. 1, pp. 56-68, 1988.

[14] P. E. Danielsson, "A new shape factor," *Comput. Graphics Image Processing*, pp. 292-299, 1978.

[15] P. E. Danielsson, "Euclidean distance mapping," *Comput. Graphics Image Processing*, 14, pp. 227-248, 1980.

[16] H. David, *Order Statistics*, John Wiley, New York, 1981.

[17] E. Dougherty and R. Haralick, "The logical context of nonlinear filtering," *Proc. SPIE Symp. on Image Algebra and Morphological Image Processing*, vol. 1658, pp. 234-244, 1992.

[18] E. Doughherty and C. Giardina, *Morphological methods in image and signal processing*, Prentice-Hall, Englewood Cliffs, NJ, 1988.

[19] E. Doughherty and C. Giardina, "Closed-form representation of convolution, dilation, and erosion in the context of image algebra," *Proc. CVPR*, pp. 754-759, 1988.

[20] E. Dougherty and J. Pelz, "Morphological granulometric and analysis of electrophotographic images – size distribution statistics for process control," submitted to *Optical Engineering*.

[21] E. Dougherty and P. Sehdev, "A robust image processing language in the context of image algebra," *Proc. CVPR*, pp. 748-753, 1988.

[22] E. Doughherty, "The dual representation of gray-scale morphological filters," *Proc. CVPR*, pp. 172-177, 1989.

[23] E. Dougherty, "Hausdorff-metric interpretation of convergence in the Matheron topology for binary mathematical morphology," *Proc. 10 Inter. Conf. Pattern Recognition*, pp. 870-875, 1990.

[24] P. Ghosh, "A mathematical model for shape description using Minkowski operators," *CVGIP*, vol. 44, pp. 239-269, 1988.

[25] P. Ghosh, "A solution of polygon containment, spatial planning, and other related problems using Minkowski operations," *Comp. Vision Graph. Image Pro.*, vol. 49, pp. 1-35, 1990.

[26] P. Ghosh, "An algebra of polygons through the notion of negative shapes," *CVGIP: Image Understanding*, vol. 54, no. 1, pp. 119-144, 1991.

[27] M. Gokmen and R. W. Hall, "Parallel shrinking algorithm using 2-subfields approaches," *Computer Vision, Graphics, and Image Processing*, 52, pp. 191-209, 1990.

[28] W. Gong, "On decomposition of structure element for mathematical morphology,"

Proc. ICPR, pp. 836-838, 1988.

[29] J. Goutsias and D. Schonfeld, "Morphological representation of discrete and binary images," *IEEE Trans. Signal Processing*, vol. 39, pp. 1369 - 1379, 1991.

[30] Z. Guo and R. W. Hall, "Parallel thinning with two-subiteration algorithms," *Comm. ACM,* Vol. 32, No. 3, pp. 359-373, March 1989.

[31] R. W. Hall, "Fast parallel thinning algorithm: parallel speed and connectivity preservation,"*Comm. ACM,* Vol. 32, No. 1, pp. 124-131, January 1989.

[32] R. Haralick, C. Lin, J. Lee and X. Zhuang, "Multi-resolution Morphology," *IEEE Proc.. Inter. Conf. Comput. Vision*, London, pp 516-520, June 1987.

[33] S. Hazout and N. Nguyen, "Image analysis by morphological automata," *Pattern Recognition*, vol. 24, no. 5, pp 401-408, 1991.

[34] R. Haralick, S. Sternberg and X. Zhuang, "Image analysis using mathematical morphology," *IEEE Trans. Pattern Anal. Machine Intel.*, vol. 9, no. 4, pp. 532-550, July 1987.

[35] R. Haralick, X. Zhuang, C. Lin and J. Lee, "Binary Morphology: working in the sampled domain," *IEEE Proc. CVPR*, pp. 780 - 791, 1988.

[36] R. Haralick, X. Zhuang, C. Lin and J. Lee, "The digital Morphological sampling theorem," *IEEE Trans. Acoust. Speech Signal Processing*, vol. 37, pp. 2067 - 2090, 1989.

[37] H. Heijmans and C. Ronse, "The algebraic basis of mathematical morphology I. dilations and erosions," *Computer Vision Graphics Image Proc.*, vol. 50, pp. 245-295, 1990.

[38] H. Heijmans, "Theoretical aspects of gray-level morphology," *IEEE Trans. Pattern Anal. Machine Intel.*, vol. 13, 1991.

[39] H. Heijmans and A. Toet, "Morphological sampling," *Computer Vision Graphics Image Proc.: Image Understanding*, vol. 54, no. 3, November, pp. 384-400, 1991.

[40] J. Hilditch, "Linear skeletons from square cupboards," in *Machine Intelligence*, vol. 4, (B. Meltzer and D. Michie Eds.), American Elsevier, pp. 403-420, 1969.

[41] C. M. Holt, A. Stewart, M. Clint, and R. H. Perrott,"An improved parallel thinning algorithm," *Comm. ACM,* Vol. 30, No. 2, pp. 156-160, February 1987.

[42] C. Huang and O. Mitchell, "Shape analysis using morphology," *Proc. of Midcon 90*, Dallas, Texas, pp 436-440, 1990.

[43] B. Jang and R. Chin, "Analysis of thinning algorithms using mathematical morphology," *IEEE Trans. Pattern Anal. Machine Intel.*, vol. 12, No. 6, pp 541-551, June 1990.

[44] L. Koskinen, J. Astola, and Y. Neuvo, "Soft morphological filters," *Proc. SPIE Symp. on Image Algebra and Morphological Image Processing*, vol. 1568, pp. 262-270, 1991.

[45] L. Koskinen and J. Astola, "Statistical properties of soft morphological filters," *Proc. SPIE Symp. on nonlinear Image Processing*, vol. 1658, pp. 25-36, 1992.

[46] P. C.K. Kwok,"A thinning algorithm by contour generation,"*Comm. ACM,* Vol 31, No. 11, pp. 1314-1324, Nov. 1988.

[47] C. Lantuejoul, "Skeletonization in quantitative metallography," in *Issues In Digital Image Processing*, R. M. Haralick and J. C. Simon Eds., Maryland: Sijthoff & Noordhoff, pp. 107-135, 1980.

[48] J. Lee, R. Haralick and L. Shapiro, "Morphological edge detection," *IEEE Trans. Robotics and Automation*, vol. 3, No. 2, pp 142-150, April 1987.

[49] H. E. Lu and P.S. P. Wang, "A comment on: a fast thinning algorithm for thinning digital patterns," *Comm. ACM,* Vol. 29, No. 3, pp. 239-242, March 1986.

[50] P. Maragos, "A unified theory of translation-invariant system with applications to morphological analysis and coding of images," Ph.D. dissertation, School Elec. Eng., Georgia Inst. Technol., Atlan, July 1985.

[51] P. Maragos, "Pattern spectrum of images and morphological shape-size complexity," *Proc. Inter. Conf. ASSP*, April, 1987.

[52] P. Maragos, "Optimal morphological approaches to image matching and object detection," *Proc. ICCV*, pp. 695-699, 1988.

[53] P. Maragos, "Morphology-based symbolic image modeling, multi-scale nonlinear smoothing, and pattern spectrum," *Proc. CVPR*, pp. 766-773, 1988.

[54] P. Maragos, "A representation theory for morphological image and signal processing," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 11, no. 6, pp. 586-599, 1989.

[55] P. Maragos, "Pattern spectrum and multiscale shape representation," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 11, pp. 701-716, 1989.

[56] P. Maragos and R. Schafer, "Morphological skeleton representation and coding of binary images," *IEEE Trans. Acoust. Speech Signal Processing*, vol. 34, pp. 1228-1244, 1986.

[57] P. Maragos and R. Schafer, "Morphological filters - part I: their set-theoretic analysis and relations to linear shift-invariant," *IEEE Trans. Acoust. Speech Signal Processing*, vol. 35, no. 8, pp. 1153-1169, 1987.

[58] P. Maragos and R. Schafer, "Morphological filters - part II: their relations to median, order-statistic, and stack filters," *IEEE Trans. Acoust. Speech Signal Processing*, vol. 35, no. 8, pp. 1170-1184, 1987.

[59] G. Matheron, *Random Sets and Integral Geometry*. New York: Wiley, 1975.

[60] P. Maragos and R. Ziff, "Threshold superposition in morphological image analysis systems," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 12, no. 5, pp. 498-504, 1990.

[61] U. Montanari, "A method for obtaining skeletons using a quasi-Euclidean distance in N dimensions," *J. Assoc. Comput.*, 15, pp. 600-624, 1968.

[62] C. Moran, "A morphological transformation for sharpening edges of features before segmentation," *Computer Vision Graphics Image Proc.*, vol. 49, pp. 85-94, 1990.

[63] N. J. Naccache and R. Shinghal, "SPTA: a proposed algorithm for thinning binary patterns," *IEEE Trans. Systems, Man, and Cybernetics*, Vol. SMC-14, No. 3, pp. 409-418, May 1984.

[64] J. Noble, "Morphological feature detection," *Proc. ICCV*, pp. 112-116, 1988.

[65] L. O'Gorman, "K×K thinning," *Computer Vision, Graphics, and Image Processing*, 51, pp. 195-215, 1990.

[66] I. Pitas and A. Maglara, "Range image analysis by using morphological signal decomposition," *Pattern Recognition*, vol. 24, no. 2, pp. 165-181, 1991.

[67] I. Pitas and A. Venetsanopoulos, "Nonlinear digital filters," Kluwer Acamedic Publishers, Boston, 1990.

[68] I. Pitas and A. Venetsanopoulos, "Morphological shape decomposition," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 12, pp. 38-45, 1990.

[69] C. Pu and F. Shih , "Soft mathematical morphology: binary and gray scale," *Proc. EURASIP Math. Morphology and its Application to Signal Processing*, Barcelona, Spain, May, 1993.

[70] C. Richardson and R. Schafer, "A lower bound for structuring element decompositions," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 13, no. 4, pp. 365-369, 1991.

[71] C. Rose and H. Heijmans, "The algebraic basis of mathematical morphology II. openings and closings," *CVGIP: Image Understanding*, vol. 54, no. 1, pp.74-97, June 1991.

[72] A. Rosenfeld and A. C. Kak, *Digital Picture Processing*, vol. 2, New York: Academic, 1982.

[73] A. Rosenfeld and J. L. Pfaltz, "Sequential Operations in Digital Picture Processing," *J. ACM*, 13, pp. 471-494, 1967.

[74] D. Schonfeld and J. Goutsias, "Optimal morphological pattern restoration from noisy binary images," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 13, no. 1, pp. 14-29, 1991.

[75] J. Serra, *Image Analysis and Mathematical Morphology*. New York: Academic, 1982.

[76] J. Sera, "Introduction to mathematical morphology," *Comput. Vision, Graphics, Image Processing*, vol. 35, pp. 283-305, Sep. 1986.

[77] J. Serra, *Image Analysis and Mathematical Morphology vol 2: theoretical advances*. New York: Academic, 1988.

[78] M. Sato, T. Wada, and H. Kawarada, "A morphological study on sturcture line," *Proc. ICPR*, pp. 559-562, 1988.

[79] F. Shih and H. Don, "Machine tools recognition by image morphology," *Proc. Inter. Computer Symposium*, 1988.

[80] F. Shih, "A new image compression technique based on geometric features," *Proc. Inter. Conf. Image Processing*, Singapore, September 1989.

[81] F. Shih, C. King and C. Pu, "A two-scan algorithm and architecture to a root for morphological filters," *Proc. Inter. Phoenix Conf. Computer Comm.*, Scottsdale, Arizona, pp. 78-84, March 1990.

[82] F. Shih and O. Mitchell, "Industrial parts recognition and isnpection by image morphology," *Proc. Inter. Robotics Automation*, Phila., April 1988.

[83] F. Shih and O. Mitchell, "Automated fast recognition and location of arbitrarily shaped object by image morphology," *Proc. CVPR*, pp. 774-779, 1988.

[84] F. Shih and O. Mitchell, "Threshold decomposition of gray-scale morphological into binary morphology," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 11, no. 1, pp. 31-42, 1989.

[85] F. Shih and O. Mitchell, "Decomposition of gray-scale morphological structuring elements," *Pattern Recognition*, vol. 24, no. 3, pp 195-203, 1991.

[86] F. Shih and O. Mitchell, "A mathematical morphology approach to Euclidean distance transformation," *IEEE Trans. Image Processing*.

[87] F. Shih and C. Pu, "A maxima-tracking method for skeletonization from Euclidean distance function," *IEEE Proc. Inter. Conf. Tools for Artif. Intell.*, San Jose, CA, pp. 246-253, Nov. 1991.

[88] F. Shih and C. Pu, "Morphological shape description using geometric spectrum on multidimensional binary images," *Pattern Recognition* Vol. 25, No. 9, pp. 921-927, 1992.

[89] F. Shih and C. Pu, "Analysis of the properties of soft morphological filtering using threshold decomposition," submitted to *IEEE Trans. Signal Processing*, 1993.

[90] F. Shih and C. Pu, "Threshold decomposition of soft morphological filters," *Proc. IEEE Inter. Conf. on Computer Vision and Pattern Recognition*, New York City, June, 1993.

[91] F. Shih and C. Pu, "Medial axis transformation with single-pixel and connectivity preservation," Submitted to *Internal Journal of Computer Vision*, 1993

[92] D. Sinha and C. Giardina, "Discrete black and white object recognition via morphological functions," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 12, no. 3, pp. 275-293, March 1990.

[93] M. Skolnick, S. Kim and R. O'Bara, "Morphological algorithms for computing non-planar point neighborhoods on cellular automata," *Proc. ICCV*, pp. 106-111, 1988.

[94] J. Song and E. Delp, "The analysis of morphological filters with multiple structuring elements,"*Computer Vision, Graphics, Image Processing*, vol. 50, pp. 308-328, 1990.

[95] S. R. Sternberg, "Grayscale morphology," *Comput. Vision, Graphics, Image Processing*, vol. 35, pp. 333-355, 1986.

[96] J. Thomas, R. Peter II and P. Jeanty, "Automatic segmentation of ultrasound images using morphological operators," *IEEE Trans. Medical Image*, vol. 10, no. 2, pp. 180-186, June 1991.

[97] A. Toet, "Image fusion by a ratio of low-pass pyramid," *Pattern Recognition Letter*, 9, pp. 245-253, 1989.

[98] A. Toet, "A morphological pyramidal image decomposition," *Pattern Recognition Letter*, 9, pp. 255-261, 1989.

[99] A. Toet, "Hierarchical image fusion," *Machine Vision and Application*, vol. 3, pp. 1-11, 1990.

[100] J. Toriwaki and S. Yokoi, "Distance Transformations and Skeletons of Digitized Pictures with Applications," pp. 187-264 , in *Progress in Pattern Recognition,* Ed. by L.N. Kanal and A. Rosenfeld, North-Holland Publisher, 1981.

[101] L. Vincent and P. Soille, "Watersheds in digital spaces: an efficient algorithm based on immersion simulations," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 13, no. 6, pp. 583-598, June 1991.

[102] R. Vogt, "Automatic generation of simple morphological algorithms," *Proc. CVPR*, pp. 760-765, 1988.

[103] A. M. Vossepoel, "A note on distance transformations in digital images," *Computer Vision, Graphics, and Image Processing,* 43, pp. 88-97, 1988.

[104] X. Wang and G. Bertrand, "An algorithm for a generalized distance transformation based on Minkowski operations," *Proc. ICPR*, pp. 1164-1168, 1988.

[105] M. Wu, "Fuzzy morphology and image analysis," *Proc. ICPR*, pp. 453-455, 1988.

[106] J. Xu, "The optimal implementation of morphological operations on neighborhood connected array processors," *Proc. CVPR*, pp. 166-171, 1989.

[107] J. Xu, "Morphological skeleton and shape decomposition," *Proc. 10 Inter. Conf. Pattern Recognition*, pp. 876-880, 1990.

[108] J. Xu, "Decomposition of convex polygonal morphological structuring elements into neighborhood subsets," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 13, no. 2, pp. 153-162, Feb. 1991.

[109] T. Y. Zhang and C. Y. Suen, "A fast thinning algorithm for thinning digital patterns," *Comm. ACM,* Vol. 27, No. 3, pp. 236-239, March 1984.