MORPHOLOGICAL SIGNATURE TRANSFORM
FOR SHAPE REPRESENTATION AND MATCHING

A Dissertation submitted to the

Division of Research and Advanced Studies

of the University of Cincinnati

in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

in the Department of Electrical and Computer Engineering

of the College of Engineering

1994

by

Sven Lončarić

M.S.E.E., University of Zagreb, Croatia, 1989

B.S.E.E., University of Zagreb, Croatia, 1985

Committee Chair: Atam P. Dhawan

Abstract

A novel shape description method based on the Morphological Signature Transform (MST) is presented in this dissertation. The MST uses multiresolution morphological image processing by non-convex multiple structuring elements. A binary image which contains the object shape to be described is represented by means of a multiresolution pyramid. The method is based on the successive morphological erosions of the input image at different resolutions by primary and rotated structuring elements. The areas of successively eroded images are computed for each structuring element at each pyramid level. The obtained set of numbers is arranged into vectors, ordered, and used as a shape descriptor. Experimental results demonstrate that the method is robust against noise and invariant to translation, rotation, and scale change.

A new method for the selection of the optimal structuring element is presented in the second part of dissertation. For a given class of shapes the optimal structuring element for MST method is selected by means of a genetic algorithm. The optimization criteria is formulated to enable a robust shape matching. Experiments have been performed on a class of model shapes. The proposed optimal shape description method is applied to the problem of shape matching which evolves in many object recognition applications. Here, an unknown object is matched to a set of known objects in order to classify it into one of finite number of classes.

An application of the developed theory to the medical image registration problem constitutes the third part of dissertation. A new method for 3-D registration of Magnetic Resonance (MR) and Positron Emission Tomographic (PET) images of the brain is presented in this paper. It is

based on the combination of the Iterative Principal Axis Registration (IPAR) method and the shape description method based on the Morphological Signature Transform (MST). The new method is called the IPAR/MST registration method. Experimental results using a physical brain phantom and a real data demonstrate that the IPAR/MST method improves the accuracy of the conventional IPAR method.

Dedicated to my wife, Ljiljana,

and to our children, Dora and Philip.

Acknowledgments

Instead of just *working* on my Ph.D. thesis I could say that I have been *living* with it for last three and a half years. So demanding is this task that it can be completed only with help and support of many people.

In the first place I want to thank my advisor Dr. Atam P. Dhawan for all the effort and energy that he put into leading me through the labyrinth of scientific research. Without his friendship and support this work would not be possible. I am very grateful to him for introducing me to an exciting field of research. I also want to express my gratitude to the members of my Doctoral Committee Dr. Arthur Helmicki, Dr. Ravi Kothari, Dr. Marios Polycarpou, and Dr. Joseph Broderick who all helped me with excellent suggestions regarding this dissertation.

As a Fulbright fellow, I would like to thank the Fulbright program for giving me an opportunity to study in the USA and for providing financial support for the first year of my study. I also want to thank the Croatian Ministry of Science and Technology for their financial support during the second year of my study.

My colleagues from the laboratory were all very helpful with invaluable discussions. Lou Arata has been very helpful in clarifying many aspects of MR and PET imaging. I would also like to thank Prashanth Kini, Tom Dufresne, Chad Peck, Yateen Chitre, and Kevin Wheeler for their help and discussions.

The support of my family has been very important in these trying times. My wife, Ljiljana, has spent a lot of time taking care of our children, Dora and Philip, and other family matters

while I was at work. Without her support this dissertation definitely could not be completed. Dora and Philip had to spend many hours without the presence of their father. Special thanks to my mom and dad (Radojka and Vjenceslav Lončarić) and my mother and father-in-law (Ivanka and Krunoslav Fagač) who have helped us very much with our children and have been a constant source of emotional and financial support.

Finally, I would like to thank many other people who may have not been mentioned but their help and support was of great value for the completion of this dissertation.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Perception capability is a characteristic of intelligent beings. Perception and intelligence are means for adaptation to varying conditions of the environment. To develop artificial systems with the ability to perform tasks that require interaction with the environment, it is necessary to study biological systems and emulate their functionality. The human visual perception system is one of the most complex and advanced in nature. For these reasons, a considerable amount of research is done today to understand the underlying neuropsychological processes. The most important task of the visual perception system and higher cognitive levels is recognition (understanding) of objects. The components of shape understanding process are shape description, discrimination, and recognition. In this dissertation we address the problem of shape description and matching. New methods and experimental results are presented which demonstrate the usefulness of the proposed techniques.

A novel shape description method based on the multiresolution morphological image processing is presented in this dissertation. The method is useful for description of both binary and gray-scale images. Multiple convex and/or non-convex structuring elements are used to create unique shape signatures used to describe a shape. Simple shape descriptors like area are used to describe shape signatures. The usability of the method is tested on two applications: shape matching and medical image registration. Shape matching is performed by computing the distance between shape descriptor vectors.

A new approach for optimal shape matching is presented. The shape matching is optimal in the sense of the most robust shape discrimination. An advanced genetic algorithm is developed to determine the optimal structuring element for shape matching.

The developed morphological shape description method is applied to a medical multi-modality image registration problem. Here, multiple images from medical imaging modalities are registered. Registration combines the information coming from different sources to enable a better study and diagnosis of various diseases.

1.1 Outline of Dissertation

An overview of mathematical morphology is given in Chapter 2. It provides some historical notes and a formal introduction to mathematical morphology. Binary and gray-scale morphology are treated in separate sections and basic properties presented.

Chapter 3 provides an overview of existing shape description methods. Several theories of visual form perception are briefly addressed. An attempt is made here to classify shape

description methods into several groups. Most representative methods with applications are briefly described.

A new framework for morphological shape description [157, 159] is described in Chapter 4. It explains a new method for morphological shape description which is based on Morphological Signature Transform (MST). It also provides experimental results which demonstrate the feasibility of method for shape matching.

A method for optimal shape matching [161, 160] is presented in Chapter 5. An advanced genetic algorithm for optimal 2-D structuring element is explained and used in determining the optimal structuring element for morphological signature transform based shape description and matching. Experimental results are provided which demonstrate the quality of optimal shape matching.

A concept of multi-modality multi-reference registration is proposed in Chapter 6. Here, a MR (Magnetic Resonance) image of the brain is registered to a PET (Positron Emission Tomography) image of the same brain. Images are registered by a combination of IPAR (Iterative Principal Axis Registration) [4, 2] and morphological signature transform shape description based technique [158]. A novel multi-grid algorithm with high computational efficiency is presented. Experimental results demonstrating the feasibility of the approach are presented.

A new method for selection of the optimal 3-D structuring element for multi-modality image registration is developed in Chapter 7. The obtained optimal structuring element is used for MST-based medical image registration. Experimental results demonstrate the accuracy of the method.

Finally, a discussion on the dissertation work is presented in Chapter 8. It summarizes the methods and the results presented in this work and discusses possible directions for future research.

Chapter 2

Mathematical Morphology: An Overview

2.1   Introduction

The foundations of mathematical morphology were established by Minkowski [192] at the beginning of this century. Later, Matheron [184] and Serra [262, 263, 264] continued to develop the field. Recently, a new wave of interest for morphological operations and their applications to divergent fields has emerged.

A number of concepts closely related to mathematical morphology have been investigated in literature. Neighborhood operations with applications in cellular architectures for image processing are studied in [93, 232, 235]. Other operations related to mathematical morphology include min-max neighborhood operators [197, 213, 300] and order statistic filters [129, 5, 224, 227].

Mathematical morphology is a powerful tool for image processing. It has no equivalent among conventional linear image processing methods. The non-linear nature of morphological operations represents both advantage and disadvantage. The advantage is in powerful shape processing capabilities that come from their non-linear nature. The disadvantage of non-linearity lies in the fact that it is not possible to construct a conventional algebraic framework with all conveniences of linear systems. However, some weaker algebraic structures (most importantly lacking inverse elements) can be formed [44, 57, 61, 69, 115, 242, 243].

A number of researchers investigated applications of mathematical morphology to biomedical image processing [187, 188, 277, 140, 39], geodesic image analysis [147, 148], range image analysis [222], edge enhancement [194], texture analysis [300, 221], image smoothing [170], segmentation [190], interpolation [212], and decomposition [210, 211].

Many contributions have been made in the field of shape analysis: shape recognition [44, 295, 171, 270, 314, 313, 288], shape description [259, 248, 275, 159, 161], shape matching [267, 159, 161],

Numerous papers have been published on theoretical development of the mathematical morphology [110, 111, 244, 279, 116, 280, 58, 247, 293, 292, 64, 294]. Morphological operations have also been implemented using neural networks [274].

The foundations of the mathematical morphology are in set theory, integral geometry, convex analysis, stereology, and geometrical probabilities [191, 169, 175]. Additional material on the mathematical morphology can be found in [263, 281, 109, 88, 175].

In the following sections an introduction to Euclidean binary and gray scale morphological

operations is presented.

2.2 Theoretical Background

The mathematical morphology is based on the *Boolean algebra* structure constructed using the notion of *lattice*. Therefore to establish the basis for the development of morphology we first define the necessary mathematical structures [263, 62].

For generality, let W be a set which contains sets (objects) on which morphological operations are performed. Some common examples for W are: $W = R^2$ (two-dimensional space) or $W = R^3$ (three-dimensional space), where R denotes the set of real numbers. Examples of discrete case are $W = Z^2$, and $W = Z^3$, where Z denotes a set of positive and negative integers. Discrete case describes the situation that occurs in practical realizations on computers.

Definition 2.1 *A relation M is a subset of the Cartesian product of the sets $L \times L$.*

$$M \subset L \times L \tag{2.1}$$

Then for $a, b \in L$ we say that

$$aMb \Longleftrightarrow (a, b) \in M \tag{2.2}$$

The left side of Equation 2.2 is read "a is in relation M with b". □

Having defined the notion of relation we proceed with the definition of a partial order rela-
tionrelation.

Definition 2.2 *Let L be a set, M be a relation on $L \times L$, and $a, b, c \in L$. A partial order relation M is a relation having the following properties.*

1. *aMa (reflexivity)*

2. *$aMb \iff bMa$ (symmetry)*

3. *$aMb \ \& \ bMc \implies aMc$ (transitivity)*

A total order relation is a partial order relation which satisfies $\forall a, b \in L$, either aMb, or bMa.

$\square$

Definition 2.3 *The pair (L, M) is called a partially ordered set if M is a partial order relation on L. The pair (L, M) is called a totally ordered set if M is a total order relation on L.* $\square$

The following are the definitions for general,distributive,bounded, and complemented lattice which are required to construct a Boolean algebra structure.

Definition 2.4 *A quartet $(L, M, \vee, \wedge)$ is called a (general) lattice if the following conditions are met.*

1. *(L, M) is a partially ordered set.*

2. *$\vee$ is a least upper bound (supremum) operation.*

3. *$\wedge$ is a greatest lower bound (infimum) operation.*

4. *Every subset $K \subset L$ has a least upper bound and a greatest lower bound which belong to L.*

Supremum and infimum operations $\vee$ and $\wedge$ must have the following properties.

18

1. $x \vee y = y \vee x$ and $x \wedge y = y \wedge x$ *(commutation)*

2. $x \vee (y \vee z) = (x \vee y) \vee z$ and $x \wedge (y \wedge z) = (x \wedge y) \wedge z$ *(association)*

3. $x \vee x = x$ and $x \wedge x = x$ *(idempotence)*

4. $x \vee (y \wedge x) = x$ and $x \wedge (y \vee x) = x$ *(absorption)*

$\square$

Definition 2.5 *A quartet $(L, M, \vee, \wedge)$ is called a distributive lattice if the following conditions are met.*

1. $(L, M, \vee, \wedge)$ *is a general lattice*

2. $x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$ and $x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$ *(distributivity)*

$\square$

Definition 2.6 *A sextuple $(L, M, \vee, \wedge, p, q)$ is called a bounded lattice if the following conditions are met.*

1. $(L, M, \vee, \wedge)$ *is a general lattice.*

2. $\exists\, p \in L \mid pMx \; \forall x \in L$ *(smallest element)*

3. $\exists\, q \in L \mid xMq \; \forall x \in L$ *(largest element)*

$\square$

Definition 2.7 *A septet* $(L, M, \vee, \wedge, ', p, q)$ *is called a complemented lattice if the following conditions are met.*

1. $(L, M, \vee, \wedge, p, q)$ *is a bounded lattice.*

2. $\forall x \in L \; \exists \; x' \in L \mid x \vee x' = q$ *and* $x \wedge x' = p$ *(complement)*

$\square$

Finally, we define the notion of Boolean algebra which is the foundation of mathematical morphology.

Definition 2.8 *A septet* $(L, M, \vee, \wedge, ', p, q)$ *is called a Boolean algebra if the following conditions are met.*

1. $(L, M, \vee, \wedge, p, q)$ *is a bounded and a distributive lattice.*

2. $(L, M, \vee, \wedge, ', p, q)$ *is a complemented lattice.*

$\square$

The following example illustrates the definitions of relation, lattice, and Boolean algebra on a familiar n-dimensional real vector space.

Example 2.9 Let $W = R^n$ and let $L = \mathcal{P}(W)$ be a partitive set of W. The familiar set subset operation $\subset$ is a partial order relation on L. The pair $(L, \subset)$ is a partially order set. The quartet $(L, \subset, \cup, \cap)$ is a lattice, where the operators $\cup$ and $\cap$ are the standard set union and intersection operators. $(L, \subset, \cup, \cap)$ is also a distributive lattice. The sextuple $(L, \subset, \cup, \cap, \Phi, \Omega)$ is

a bounded lattice, where Φ denotes the empty set and Ω denotes the full set. Using the standard set complement operation ^{c} we have that $(L, \subset, \cup, \cap, ^c, \Phi, \Omega)$ is a complemented lattice. Finally, $(L, \subset, \cup, \cap, ^c, \Phi, \Omega)$ is also a Boolean algebra. $\qquad \square$

2.3   Binary Morphology

2.3.1   Definitions

In this section, definitions of basic morphological operations are presented. We again consider a set W whose subsets are sets we want to manipulate. Typical examples are $W = R^2$ (images) and $W = R^3$ (volumes). Note that $W = R^n$ is a real vector space.

Definition 2.10 *The multiplication of a set by a real number is defined as*

$$rX = \bigcup_{x \in X} \{rx\} \tag{2.3}$$

where $r \in R$, and $X \subset W$. The multiplication of a vector by a real number which appears in Equation 2.3 is the conventional real number by vector multiplication. $\qquad \square$

Definition 2.11 *The translation of a set by a real vector is defined as*

$$X_y = \{x + y \mid x \in X\} \tag{2.4}$$

where $y \in W$, and $X \subset W$. The vector addition which appears in Equation 2.4 is the conventional vector addition. $\qquad \square$

Definition 2.12 *The symmetric set $\check{X}$ of a set X is defined as*

$$\check{X} = \{x \in W \mid -x \in X\} \tag{2.5}$$

where $X \subset W$. $\qquad\qquad\square$

The basis of mathematical morphology are Minkowski set addition and subtraction operations. In the following definitions let $\mathcal{P}$ be the partitive set (a set of all subsets) of W.

Definition 2.13 *Minkowski addition $\oplus : \mathcal{P} \times \mathcal{P} \to \mathcal{P}$ and Minkowski subtraction $\ominus : \mathcal{P} \times \mathcal{P} \to \mathcal{P}$ are defined by*

$$
\begin{aligned}
X \oplus Y &= \bigcup_{y \in Y} X_y \tag{2.6} \\
X \ominus Y &= \bigcap_{y \in Y} X_y \tag{2.7}
\end{aligned}
$$

where $X, Y \in \mathcal{P}$. $\qquad\qquad\square$

The notation $\oplus : \mathcal{P} \times \mathcal{P} \to \mathcal{P}$ means that the function $\oplus$ takes two elements of the set $\mathcal{P}$ as arguments and produces another element of the set $\mathcal{P}$ as result. Note that the elements of the set $\mathcal{P}$ are subsets of W.

Morphological erosion and dilation [184, 262] operations are the basic and among the most useful operations for image processing purposes. Most other operations are derived from these two.

Definition 2.14 *The morphological dilation $\mathcal{D} : \mathcal{P} \times \mathcal{P} \to \mathcal{P}$ and its dual the morphological erosion $\mathcal{E} : \mathcal{P} \times \mathcal{P} \to \mathcal{P}$ are defined by*

$$\mathcal{D}(X, B) \;=\; X \oplus \check{B} = \bigcup_{y \in B} X_{-y} \tag{2.8}$$

$$\mathcal{E}(X, B) \;=\; X \ominus \check{B} = \bigcap_{y \in B} X_{-y} \tag{2.9}$$

where $X, B \subset W$. $\qquad\square$

Morphological opening and closing were introduced by Matheron [184] and are among the most powerful in mathematical morphology.

Definition 2.15 *Morphological opening and closing are defined as*

$$X \circ B \;=\; X \ominus \check{B} \oplus B \tag{2.10}$$

$$X \bullet B \;=\; X \oplus \check{B} \ominus B \tag{2.11}$$

$\qquad\square$

Proposition 2.16 *Morphological opening can also be computed using the following expression.*

$$X \circ B = \bigcup_{x \in W} \{B_x \mid B_x \subset X\} \tag{2.12}$$

$\qquad\square$

fig/morphology/image.ps

fig/morphology/kernel.ps

Figure 2.1: The original image and the structuring element.

Equation 2.12 reveals the nature of the opening operation. If set B has the disk shape, the opening can be thought of as a "rolling ball" algorithm where the set B (often called a structuring element) is rolled along the interior border of X to eliminate all fine structure of X which cannot be accessed by B.

2.3.2 Examples

In this section, the examples of the basic morphological operations are presented.

An example image and a structuring element are shown in Figure 2.1. Figures 2.2, 2.3, 2.4, and 2.5 illustrate the morphological erosion, dilation, opening, and closing, respectively. The image on the left shows the result of operation while the image on the right shows the resulting image superimposed on the original image.

fig/morphology/erosion.ps   fig/morphology/erosion-comb.ps

Figure 2.2: Morphological erosion operation.

fig/morphology/dilation.ps   fig/morphology/dilation-comb.ps

Figure 2.3: Morphological dilation operation.

fig/morphology/opening.ps   fig/morphology/opening-comb.ps

Figure 2.4: Morphological opening operation.

fig/morphology/closing.ps   fig/morphology/closing-comb.ps

Figure 2.5: Morphological closing operation.

2.3.3 Basic Properties

One of the main properties of morphological operations is non-invertibility. Shape processing capabilities of morphological operations are a result of how these operations are defined. However, these operations are not invertible. The reason for this is the fact that a certain amount of information gets lost in the process of non-linear computation. The goal of any morphological processing is a controlled loss of irrelevant information through successive transformations [263].

Some of the properties that any morphological operation is tested for are the following:

1. Increasing property.

2. Anti-extensivity.

3. Idempotence.

4. Homotopy.

We provide definitions of above properties as follows.

Definition 2.17 *A morphological operation ψ is increasing if it preserves inclusion*

$$X \subset Y \Longrightarrow \psi(X) \subset \psi(Y) \tag{2.13}$$

$\square$

The increasing property is significant because the increasing mapping will preserve the inclusion operations (ordering with respect to $\subset$).

Definition 2.18 *A morphological operation ψ is anti-extensive if it shrinks*

$$\psi(X) \subset X \tag{2.14}$$

□

The operation which shrinks reduces the size of the result so that the operation result is a subset of the original set.

Definition 2.19 *A morphological operation ψ is idempotent if*

$$\psi[\psi(X)] = X \tag{2.15}$$

□

Idempotence is a very important property of morphological operations. This property of certain morphological operations shows their fundamental difference compared to linear operations. The important difference lies in the fact that a filtering operation can be performed in only one step. Successive applications of an idempotent operation do not make any difference.

Definition 2.20 *A morphological operation ψ is homotopic if it preserves the operand's homotopic tree.* □

The homotopic tree shows the topological information about the object topology. A homotopic operation will preserve the topological properties of the operand.

A concept of duality is very important in mathematical morphology. Each morphological operation has its dual operation. Actually, as soon as we define a new operation ψ we can automatically define its dual operation ψ^* in the following way

$$\psi^*(X) = [\psi(X^c)]^c \qquad (2.16)$$

The dual operation of an increasing operation is an increasing operation.

Proposition 2.21 *Minkowski addition is an increasing but not idempotent operation. It is also an extensive operation if the structuring element contains the origin. Minkowski addition is a commutative, associative, and translation invariant operation. It distributes over the set union.*

$$X \subset Y \implies X \oplus Z \subset Y \oplus Z \qquad (2.17)$$

$$X \oplus Y = Y \oplus X \qquad (2.18)$$

$$(X \oplus Y) \oplus Z = X \oplus (Y \oplus Z) \qquad (2.19)$$

$$(X \oplus Y)_t = X_t \oplus Y \qquad (2.20)$$

$$X \oplus (Y \bigcup Z) = (X \oplus Y) \bigcup (X \oplus Z) \qquad (2.21)$$

where $X, Y, Z \in \mathcal{P}$ and $t \in W$. $\qquad \square$

Proposition 2.22 *Minkowski subtraction is not idempotent and it does not preserve homotopy. It is anti-extensive only if coordinate system origin belongs to the set that is subtracted. It is an*

increasing operation.

$$X \subset Y \implies X \ominus Z \subset Y \ominus Z \qquad (2.22)$$

$$(X \ominus Y) \ominus Z = X \ominus (Y \oplus Z) \qquad (2.23)$$

$$(X \ominus Y)_t = X_t \ominus Y \qquad (2.24)$$

$$X \ominus (Y \bigcup Z) = (X \ominus Y) \bigcap (X \ominus Z) \qquad (2.25)$$

$$(X \bigcap Y) \ominus Z = (X \ominus Z) \bigcap (Y \ominus Z) \qquad (2.26)$$

where $X, Y, Z \in \mathcal{P}$ *and* $t \in W$. $\square$

Minkowski add and subtract are dual operations and the following relation holds.

Theorem 2.23 *Duality theorem for Minkowski addition and subtraction.*

$$(X \oplus B)^c = X^c \ominus B \qquad (2.27)$$

$\square$

This means that instead computing the Minkowski addition of a set it is possible to compute the

Minkowski subtraction of the set complement and then complement the result.

Theorem 2.24 *Duality theorem for dilation and erosion.*

$$\mathcal{D}(X, B)^c = \mathcal{E}(X^c, B) \qquad (2.28)$$

□

The duality relation between erosion and dilation in Equation 2.28 is a direct consequence of Proposition 2.23. In other words, it is sufficient to implement only one morphological operation (erosion or dilation) and its dual operation can be performed according to Equation 2.16.

The erosion of set X by a set B can be visualized as a union of points by which B translated fits within X. The following proposition illustrates this point.

Proposition 2.25 *Another formulation of dilation.*

$$\mathcal{D}(X, B) = \{x \in W \mid B_x \subseteq A\} \tag{2.29}$$

□

Proposition 2.26 *Opening is an anti-extensive, idempotent, and increasing operation. Closing is an extensive, idempotent, and increasing operation.* □

Theorem 2.27 *Duality theorem for opening and closing.*

$$(X \circ B)^c = X^c \bullet B \tag{2.30}$$

□

Here we introduce a short notation that will be used later in the text.

Definition 2.28 *A short notation B^n is defined as*

$$B^n = \overbrace{B \oplus B \oplus \cdots \oplus B}^{n \; times} \qquad (2.31)$$

where $B \subset W$ and $n \in N$. □

Note that in general $B^n \neq nB$. However we have the following result.

Proposition 2.29 *The following statement holds*

$$B \quad \text{is convex} \quad \Longleftrightarrow B^n = nB \qquad (2.32)$$

where $B \subset W$ and $n \in N$. □

2.4   Gray Scale Morphology

Gray scale morphology is a generalization of binary morphology to gray scale functions so that the power of morphology can also be applied to functions. The concept of gray scale morphology can be related to and derived from the binary morphology using an elegant approach proposed by Sternberg [281, 109]. In his classical paper [281], he proposed a notion of umbra which can be used for representing gray scale morphological operations in terms of binary morphological operations. An alternative way of generalizing from binary to gray scale morphology is by a threshold decomposition of gray scale images into binary images [262, 118, 176]. In the following sections a formal development of gray scale morphology using the concept of umbra is presented.

2.4.1   Umbra and Surface Concepts

The concepts of top surface of a set and the umbra of the surface are introduced first. Here, a set in N-dimensional Euclidean space $W = E^N$ can be thought of as a gray scale function of $N - 1$ variables. This is done by assuming that first $N - 1$ coordinates of N-tuple $x \in W$ are in the domain of a gray scale function while N-th coordinate is the gray scale value. The top surface of a set W is defined on the first $N - 1$ coordinates and has the value equal to the maximum value of corresponding gray scale values.

Definition 2.30 *The top surface of a set. Let $A \subseteq E^N$ be a set, where E^N is N-dimensional Euclidean space and let $F = \{x \in E^{N-1} \mid (x, y) \in A \text{ for some } y \in E\}$. The top surface of A is a function $T[A] : F \longrightarrow E$ defined as*

$$T[A](x) = \max\{y \mid (x, y) \in A\} \tag{2.33}$$

$\square$

Definition 2.31 *The basic definition of umbra. A set $S \subseteq E^{N-1} \times E$ is an umbra if and only if*

$$(x, y) \in S \implies (x, z) \in S \; \forall z \leq y \tag{2.34}$$

$\square$

Definition 2.32 *The umbra of a function. Let $F \subseteq E^{N-1}$ and $f : F \longrightarrow E$. The umbra of f,*

$U[f] \subseteq E^{N-1} \times E$, *is defined as*

$$U[f] = \{(x, y) \in F \times E \mid y \leq f(x)\} \qquad (2.35)$$

$\square$

Note that the umbra of a function is an umbra.

2.4.2   Gray Scale Operations

Now that the concept of the surface of a set and the umbra of a function is established we proceed

with a definition of gray scale equivalents of Minkowski addition, Minkowski subtraction, dilation,

and erosion.

Definition 2.33 *The gray scale versions of the basic morphological operations. Let $F, G \subseteq E^{N-1}$*

and $f : F \longrightarrow E$ and $g : G \longrightarrow E$. Minkowski addition and subtraction, dilation, and erosion

are defined by

$$f \oplus g \;=\; T[U[f] \oplus U[g]] \qquad (2.36)$$

$$f \ominus g \;=\; T[U[f] \ominus U[g]] \qquad (2.37)$$

$$\mathcal{D}(f, g) \;=\; T[D(U[f], U[g])] \qquad (2.38)$$

$$\mathcal{E}(f, g) \;=\; T[E(U[f], U[g])] \qquad (2.39)$$

34

□

The expressions in Definition 2.33 are an elegant way of defining gray scale operations through the use of binary morphology. However, they are not convenient for practical computation. Before we proceed with expressions for practical use the notion of the symmetric function operator is introduced.

Definition 2.34 *The symmetric function f^s of a function f is given by*

$$f^s(x) = f(-x) \tag{2.40}$$

□

Note that the gray scale dilation and erosion are defined consistently with their binary equivalents.

Proposition 2.35 *Alternative ways for expressing dilation and erosion.*

$$\mathcal{D}(f, g) = f \oplus g^s \tag{2.41}$$

$$\mathcal{E}(f, g) = f \ominus g^s \tag{2.42}$$

□

Proposition 2.36 gives practical expressions for the computation of gray scale operations.

Proposition 2.36 *Practical expressions for basic gray scale operations. Let $F, G \subseteq E^{N-1}$ and $f : F \longrightarrow E$ and $g : G \longrightarrow E$.*

$$(f \oplus g)(x) = \max_{z \in F, x-z \in G} [f(z) + g(x - z)] \tag{2.43}$$

$$(f \ominus g)(x) = \min_{z \in D, x-z \in G} [f(z) - g(x - z)] \tag{2.44}$$

$$\mathcal{D}(f, g)(x) = \max_{z \in F, z-x \in G} [f(z) + g(z - x)] \tag{2.45}$$

$$\mathcal{E}(f, g)(x) = \min_{z \in F, z-x \in G} [f(z) - g(z - x)] \tag{2.46}$$

$\square$

The Equations 2.44 and 2.45 are similar to convolution expressions from linear system theory where addition is replaced by max (min) operations and where multiplication is replaced by addition (subtraction). However, this is where similarity ends since, unlike convolution, they exhibit a nonlinear behavior.

Definition 2.37 *Gray scale opening and its dual closing are defined by*

$$(f \circ g)(x) = [(f \ominus g^s) \oplus g](x) \tag{2.47}$$

$$(f \bullet g)(x) = [(f \oplus g^s) \ominus g](x) \tag{2.48}$$

$\square$

The opening operation performs a 'rolling ball' algorithm by creating resulting function points by translating the function g 'under' the surface defined by the function f. The closing is the

36

dual operation of opening.

2.4.3   Basic Properties

We begin with the basic properties of umbra and surface. The surface and the umbra operations
are inverses of each other in the following sense.

Proposition 2.38 *The surface operation is an inverse of the umbra operation. Let $F \subseteq E^{N-1}$
and $f : F \longrightarrow E$. Then*

$$T[U[f]] = f \tag{2.49}$$

$\square$

On the other hand the umbra operation is not an inverse of the surface operation

Proposition 2.39 *The composition of surface and umbra operations. Let $A \subseteq E^N$. Then*

$$A \subseteq U[T[A]]] \tag{2.50}$$

The equality in Equation 2.50 holds if and only if A is an umbra. $\square$

The umbra homomorphism theorem [109] establishes the homomorphism between binary and
grayscale operations.

Theorem 2.40 *The umbra homomorphism theorem. Let $F, G \subseteq E^{N-1}$ and $f : F \longrightarrow E$ and
$g : G \longrightarrow E$. Then*

$$U[f \oplus g] \;\; = \;\; U[f] \oplus U[g] \tag{2.51}$$

$$U[f \ominus g] \quad = \quad U[f] \ominus U[g] \tag{2.52}$$

□

Let us introduce some notation before we proceed further.

Definition 2.41 *Maximum and minimum operations for functions. Let $F, G \subseteq E^{N-1}$ and $f : F \longrightarrow E$ and $g : G \longrightarrow E$. Then*

$$(f \max g)(x) \quad = \quad \max(f(x), g(x)) \tag{2.53}$$

$$(f \min g)(x) \quad = \quad \min(f(x), g(x)) \tag{2.54}$$

□

Definition 2.42 $\leq$ *and $\geq$ relations for functions. Let $F, G \subseteq E^{N-1}$ and $f : F \longrightarrow E$ and $g : G \longrightarrow E$. Then*

$$f \leq g \iff f(x) \leq g(x) \ \forall x \in E^{N-1} \tag{2.55}$$

$$f \geq g \iff f(x) \geq g(x) \ \forall x \in E^{N-1} \tag{2.56}$$

□

The relations between the umbra and min and max operators are shown bellow.

Proposition 2.43 *The relation between the umbra and* min *and* max *operators. Let* $F, G \subseteq$ E^{N-1} *and* $f : F \longrightarrow E$ *and* $g : G \longrightarrow E$. *Then*

$$U[f \max g] \;\; = \;\; U[f] \bigcup U[g] \tag{2.57}$$

$$U[f \min g] \;\; = \;\; U[f] \bigcap U[g] \tag{2.58}$$

$\square$

Proposition 2.44 *The relation between the umbra and function* $\leq$ *operator. Let* $F, G \subseteq E^{N-1}$ *and* $f : F \longrightarrow E$ *and* $g : G \longrightarrow E$. *Then*

$$f \leq g \Longleftrightarrow U[f] \subseteq U[g] \tag{2.59}$$

$\square$

Proposition 2.45 *The relation between the top surface and set* $\subseteq$ *operator. Let* $A, B \subseteq E^{N}$.

$$A \subseteq B \Longrightarrow T[A](x) \leq T[B](x) \tag{2.60}$$

$\square$

The following basic properties of grayscale operations are a consequence of the umbra homomorphism theorem.

Proposition 2.46 *Gray scale Minkowski addition is an extensive operation. It is a commutative, associative, and translation invariant operation. It distributes over the set union. Let* $F, G, H \subseteq$

E^{N-1} and $f : F \longrightarrow E$, $g : G \longrightarrow E$, and $h : H \longrightarrow E$.

$$f \leq g \implies f \oplus h \leq g \oplus h \tag{2.61}$$

$$f \oplus g = g \oplus f \tag{2.62}$$

$$(f \oplus g) \oplus h = f \oplus (g \oplus h) \tag{2.63}$$

$$(f \oplus g)_t = f_t \oplus g \tag{2.64}$$

$$f \oplus (g \max h) = (f \oplus g) \max (f \oplus h) \tag{2.65}$$

where $t \in E^{N-1}$. $\qquad\qquad\square$

Proposition 2.47 *Minkowski subtraction is not idempotent and it does not preserve homotopy. It is anti-extensive only if coordinate system origin belongs to the set that is subtracted. It is an increasing operation. Let $F, G, H \subseteq E^{N-1}$ and $f : F \longrightarrow E$, $g : G \longrightarrow E$, and $h : H \longrightarrow E$.*

$$f \subset g \implies f \ominus h \subset f \ominus h \tag{2.66}$$

$$(f \ominus g) \ominus h = f \ominus (g \oplus h) \tag{2.67}$$

$$(f \ominus g)_t = f_t \ominus g \tag{2.68}$$

$$f \ominus (g \max h) = (f \ominus g) \min (f \ominus h) \tag{2.69}$$

$$(f \min g) \ominus h = (f \ominus h) \min (g \ominus h) \tag{2.70}$$

where $t \in E^{N-1}$. $\qquad\qquad\square$

Proposition 2.48 *Gray scale opening is an anti-extensive, idempotent, and increasing operation. Gray scale closing is an extensive, idempotent, and increasing operation.* □

Theorem 2.49 *Duality theorem for gray scale opening and closing. Let $F, G \subseteq E^{N-1}$ and $f : F \longrightarrow E$ and $g : G \longrightarrow E$. Then*

$$- (f \circ g) = (-f) \bullet g \tag{2.71}$$

□

2.5   Discussion and Comments

A certain amount of confusion has been present in literature on mathematical morphology. The reason for this is a presence of two different approaches to definitions of the basic morphological operations. Actually, there are three levels of differences that introduce confusion to definitions.

1. The way how operations are defined (mathematical definitions).

2. The way how operations are called (names).

3. The notation that is used (symbols).

The first approach has been proposed by Serra [262, 263]. The second approach has been proposed by Sternberg [281]. In an attempt to present and clarify differences of these two approaches in three levels mentioned above we present a table which contains a summary of both approaches. Table 2.1 has three main columns corresponding to three crucial levels of difference (actual

41

	Serra		Sternberg	
Definition	Name	Notation	Name	Notation
$\bigcup_{b \in B} A_b$	Minkowski addition	$A \oplus B$	Minkowski addition or dilation	$A \oplus B$
$\bigcup_{b \in B} A_{-b}$	dilation	$A \oplus \check{B}$	-	-
$\bigcap_{b \in B} A_b$	Minkowski subtraction	$A \ominus B$	Minkowski subtraction	-
$\bigcap_{b \in B} A_{-b}$	erosion	$A \ominus \check{B}$	erosion	$A \ominus B$
$\bigcup_{y \mid B_y \subseteq A} B_y$	opening	$(A \ominus \check{B}) \oplus B$	opening	$(A \ominus B) \oplus B$
$\bigcap_{y \mid B_y \subseteq A^c} B_y^c$	closing	$(A \oplus \check{B}) \ominus B$	closing	$(A \oplus B) \ominus B$

Table 2.1: A comparative table of definitions, names, and notations.

definition, name, and notation). Rows of the table correspond to operations. The table shows that there are not many differences in operation definitions vs. names. The only difference is in definition of dilation. The erosion, opening, and closing are defined consistently in both approaches. There are many more differences in operation notation vs. name or definition. For example, the symbol $\ominus$ denotes Minkowski subtraction in Serra's notation but the same symbol denotes erosion in Sternberg's notation. The names dilation and Minkowski addition denote the same operation in Sternberg's work but denote different operations in Serra's work.

Both approaches have its advantages and disadvantages. However, these name and notation differences mainly influence the way how ideas are formalized in written form and do not change the underlying operation. The Serra's approach is slightly more complicated but gives nicer duality relations. In Sternberg's approach the expressions for opening and closing are more simple but the resulting duality relations are not ideal.

The approach used in this work is the same as used by Serra. The author feels that this question is important to the extent of its influence to the formal formulation of the theory. However, the importance of these differences must not be overestimated as it is the underlying philosophy of morphological operations which is important and common for both approaches.

Chapter 3

Algorithms for Shape Description: An Overview

3.1 Introduction

The shape of the object is a silhouette of the object (e.g. obtained by illuminating the object by an infinitely distant light source). The procedures (segmentation, etc.) used to obtain the shape from a given gray-scale image are not discussed here. Algorithms that we present here assume that they are given a binary image i.e. the shape of the object. A shape description method generates a descriptor vector (also called a feature vector) from a given shape (binary image). The required properties of a shape description scheme are invariance to translation, scale, and rotation. This is required because these three transformation do not change the *shape* of the object.

The problem of shape description is one of the most important problems in image analysis. It has been pursued by many authors and there is a huge amount of work related to it. A number of review papers have been written on the topic of shape analysis [206, 207, 1]. Books dealing with shape analysis include [289, 67, 198, 253, 254, 149, 126, 308, 13, 133, 258, 108, 107].

In an attempt to create a systematic overview of the subject it is possible to classify methods according to several different criteria. Pavlidis [206] has used the following classifications. One classification is based on whether the algorithm uses only boundary points to create a description or it also looks in the interior of the shape. Two classes of algorithms are called *boundary* (or *external*) and *global* (or *internal*), respectively. Examples of the former class are algorithms which parse the shape boundary [66, 309, 137, 199, 99, 47, 136, 284] and various Fourier transforms of the boundary [312, 102, 240, 217, 296]. Examples of global methods include the medial axis (also called symmetric axis) transform (MAT) proposed by Blum and described in [23, 196, 24, 66, 25, 213, 199, 47], moment based approaches [128, 285, 17, 238], and methods of shape decomposition into other primitive shapes [73, 200, 8, 138, 132].

Another classification of shape description algorithms can be made on the basis of whether the result of the analysis is a scalar or a non-scalar. For example, MAT produces another image (containing symmetric axis) and is therefore called a *space-domain* technique. On the other hand, *scalar transform* techniques produce numbers (scalars) as result. Examples of later methods include various Fourier [312, 102, 240, 217, 296] and moment-based [128, 285, 17, 238] procedures for shape analysis.

A third classification of the shape analysis methods can be made on the basis of information

45

preserving. Methods which allow the accurate (or at least sufficiently accurate) reconstruction of the shape from its descriptor are called *information preserving* methods as opposed to methods only capable of partial reconstruction called *information nonpreserving* techniques. An example of information nonpreserving methods is area to perimeter square ratio. Many significantly different shapes can have the same area to perimeter square ratio. Therefore it is not possible to reconstruct the original shape knowing only its area to perimeter square ratio. Many simple shape descriptors suffer from the same problem.

Some of the most important shape descriptor methods are briefly described in the following sections. Section 3.2 emphasizes the role and importance of research in interdisciplinary fields like visual perception, cognition, psychology, and physiology. Sections 3.3 and 3.4 present boundary scalar and space domain techniques, respectively. Sections 3.5 and 3.6 address most characteristic global scalar and space domain methods, respectively. Finally, Section 3.7 presents various approaches to shape matching.

3.2   Visual Form Perception

To understand the nature of the human processing of shape one has to study the cognitive science. In particular, the visual cognition and perception are of interest for study of shape description. There is a huge amount of interdisciplinary literature available on the topic. Introductory and more advanced books and articles dealing with visual perception and cognition include [289, 297, 317, 43, 103, 164, 21, 134, 231].

The human visual perception and cognition system is one of the most sofisticated and versatile

in nature. Its ability to understand the organization of surrounding nature has been unsurpassed by a large margin by artificially created reasoning systems. The human visual perception and psychology are studied intensively in order to understand complex underlying processes and to develop artificial systems for visual perception and cognition.

3.2.1 Theories of Visual Form Perception

There have been several schools in psychology attempting to formalize the mechanisms of behavior, in general, and visual perception as one aspect of it. Hake [106] discussed several approaches to representation of natural forms. Zusne [317] presented an overview of contemporary theories of visual form. An in-depth discussion of visual perception theories in psychology is outside the scope of this work, therefore we limit ourselves to a brief discussion of selected topics.

The Gestalt school of psychology [318] has played a revolutionary role with its novel approach to visual form. A more detailed exposition of the subject of Gestalt theory can be found in [301, 70, 141, 144]. The Gestalt theory is a non-computational theory of visual form, this being a disadvantage for practical engineering applications. However, according to Zusne "it is still the only theory to deal with form in a comprehensive fashion" ([317], p. 108). There have been many books on Gestalt laws presenting various lists of principles. These lists range from six to more then hundred. Here, we give a list of laws for visual forms as proposed by Zusne [317].

1. Visual form is the most important property of a configuration.

2. Visual form is either dynamic or the outcome of dynamic processes which underlie them.

47

3. All visual forms possess at least two aspects, a figured portion called figure and a background called ground.

4. Visual forms may possess one or several centers of gravity about which the form is organized.

5. Visual forms are transposable (with respect to translation, size, orientation, and color) without loss of identity.

6. Visual forms resist change. They tend to maintain their structure against disturbing forces.

7. Visual forms will always be as good (regular, symmetric, simple, uniform, exhibiting the minimum amount of stress) as the conditions (pattern stimulus) allow.

8. Forms may fuse to produce new ones.

9. A change in one part of form affects other parts of the form (law of compensation).

10. Visual forms tend to appear and disappear as wholes.

11. Visual forms leave aftereffect that make them easier to remember (law of reproduction).

12. Space is anisotropic, it has different properties in different directions.

Another approach to a theory of visual form is found in Hebb's work. Hebb presented a neuropsychological theory of behavior in his book *The Organization of Behavior* [113]. In his theory, Hebb emphasized the role of the neural structures in the mechanism of visual perception. His work influenced a number of researchers in the field of artificial neural networks. As opposed to Gestalt school, Hebb argues that form is not perceived as a whole but consists of parts. The

organization and mutual spatial relation of parts must be learned for successful recognition. This learning aspect of perception is the central point in Hebb's theory. The main mechanism which integrates simpler elements of perception are eye movements. The simpler elements are angles and lines. Hebb also introduced a notion of *cell assemblies*. Cell assemblies are neurons grouped together so that repeated firings will lower the synaptic resistance causing neurons in the group to mutually excite each other. Hebb's theory was mostly qualitative and not computational.

Gibson [89] developed a third comprehensive theory of visual perception. First principle of his theory is that the space is not a geometric or abstract but a real *visual* space characterized by forms that are in it. Gibson's theory is centered around perceiving real three-dimensional objects, not their two-dimensional projections. Second principle is that there is a real world stimulus behind any simple or complex visual perception. The stimulus is in the form of gradient which is a property of surface. The examples of physical gradients are changing size of texture elements (depth dimension), degree of convergence of parallel edges (perspective), hue and saturation of colors, and shading. Gibson points out that the Gestalt school has been occupied with the study of two-dimensional projections of three-dimensional world and that its dynamism is no more then the ambiguity of interpretation of projected images. In his classifications, there are ten different kinds of form.

1. Solid form. (Seeing an object means seeing a solid form.)

2. Surface form. (Slanted and forms with edges.)

3. Outline form. (A drawing of edges of a solid form.)

4. Pictorial form. (Representations which are drawn, photographs, paintings, etc.)

5. Plan form. (A drawing of edges of a surface projected on a flat surface.)

6. Perspective form. (A perspective drawing of a form.)

7. Nonsense form. (Drawings which do not represent a real object.)

8. Plane geometric form. (An abstract geometric form not derived from or attempting to make a solid form visible.)

9. Solid geometric form. (An abstract part of a three-dimensional space bounded with imaginary surfaces.)

10. Projected form. (A plane geometric form which is a projection of a form.)

These forms are grouped into three classes as follows.

1. Real objects: solid and surface forms.

2. Representations of real objects: outline, pictorial, plan, perspective, and nonsense forms.

3. Abstract (non-real) objects: Plane geometric forms, solid geometric forms, and projected forms.

The first class is the "real" class consisting of objects form the real world. The second class are representations of real objects. The third class are abstractions which can be represented using symbols but do not correspond to real objects (because they have no corresponding stimulus in the real world). This partition is very clear in Gibson's work.

At least three separate levels of theory will be required: first a theory of how we perceive the surface of objects–a theory of slant-shape or, in older words, of shape-constancy; second, a theory of how we perceive representations, pictures, displays, and diagrams; and third, a theory of how we apprehend symbols. ([90], p. 412.)

A tutorial on the visual cognition with emphasize on shape recognition was written by Pinker [219] (see also [41]).

3.2.2 Recent Advances

Marr et al. [177, 178, 181, 182, 180, 183] made significant contributions with the study of human visual perception system. In Marr paradigm [179] a focus of research is shifted from applications to topics corresponding to modules of the human visual system. An illustration of this point is *shape from x* research which represent an important part of the total research in computer vision [1]. Papers dealing with shape from x techniques include: shape from shading [127], shape from contour [125, 290], shape from texture [89], shape from stereo [120], and shape from fractal geometry [37].

In [178] Marr developed a primal sketch paradigm for early processing of visual information. In his method, a set of masks is used to measure discontinuities in first and second derivative of the original image. This information is then processed by subsequent procedures to create a *primal sketch* of the scene. The primal sketch contains locations of edges in the image and is used by subsequent stages of the shape analysis procedure. Marr and Hildreth [180] developed further the concept of the primal sketch and proposed a new edge detection filter based on the zero crossings

of the Laplacian of the two-dimensional Gaussian distribution function. In his approach, zeros of Laplacian indicate the inflection point in the edge to detect edge positions. Marr and Nishihara [181] proposed a set of criteria for evaluation of shape description methods: accessibility, scope and uniqueness, and stability and sensitivity. Accessibility shows how easy (or difficult) is to compute a shape descriptor in terms of memory requirements and computational time. Scope shows the class of shapes that can be described by the method. Uniqueness shows whether a one-to-one mapping exists between shapes and shape descriptors. Stability and sensitivity show how sensitive a shape descriptor is to a "small" changes in shape.

Brady [30] proposed another set of criteria for representations of shape.

- *Rich local support.* This criteria requires that a representation is information preserving (rich) and can be computed locally. Local support is important for representation of occluded objects.

- *Smooth extension* and *subsumption.* This criteria ensures that local descriptions can easily produce global descriptions. This is a kind of continuity of representation.

- *Propagation.* This criteria adds a hierarchical property to representation in the sense that perceptual subparts are propagated from local to global levels of representation.

Brady illustrated this criteria on the Generalized Cylinder representation for three-dimensional objects and the Smoothed Local Symmetries representation for two-dimensional shapes [31].

Koenderink [142] and Koenderink and van Doorn [143] have studied the psychological aspects of visual perception and proposed several interesting paradigms. Conventional approaches to

shape are often static in the sense that they treat all shape details equally as global shape features [143]. A dynamic shape model was developed where the visual perception is performed on several scales of resolution. Such notions of order and relatedness are present in visual psychology and absent in conventional geometrical theories of shape. It has been argued that there exist the manuals of art theory [96] which have not been given the attention they deserve and which contain the practical knowledge accumulated over centuries [143]. In art as well as in perception, a shape is viewed as a hierarchical structure. A procedure for morphogenesis based on a multiple levels of resolution was developed [143]. Any shape can be embedded in a "morphogenetic sequence" based on the solution of the partial differential equation which describes the evolution of the shape through multiple resolutions.

Many authors agree about the significance of high curvature points for visual perception. Attneave and Arnoult [10, 11] performed psychological experiments to investigate the significance of corners for perception. In the famous Attneave's cat experiment a drawing of a cat was used to locate points of high curvature and connect them to create a simplified drawing of the cat. After a brief presentation the cat would be recognized reliably in the drawing. It was suggested that such points have high information content. To approximate curves by straight lines the high curvature points are the best place to brake the lines. In that case the resulting image retains the maximal amount of information necessary for successful shape recognition. For the purpose of shape description points of high curvature are taken to be a corners and shape can be approximated by a polygon. A variety of algorithms for polygonal approximation of the shape [204, 206, 87, 166, 307, 306] has been developed. Davis [46] combined the use of high curvature

points and line segment approximations for polygonal shape approximations.

Hoffman and Richards [121, 241] argue that when the visual system decomposes objects it does so at the points of high negative curvature. This agrees with the principal of transversality [104] which is found in nature. The principal of transversality says that when two arbitrarily shaped convex objects interpenetrate each other the meeting point is a boundary point of concave discontinuity of their tangent planes.

Leyton [152] demonstrated the *Symmetry-Curvature* theorem which claims that any curve section that has only one curvature extremum has one and only one symmetric axis which terminates at the extremum itself. (For more information on symmetric axis see Section 3.6.1.) This is an important result because it establishes the connection between to important notions in visual perception. In [153], Leyton developed a theory which claims that all shapes are basically circles which changed form as a result of various deformations cause by external forces like pushing, pulling, stretching, etc. Two problems were considered. The first was the inference of the shape history from a single shape, and the second was the inference of shape evolution between two shapes. A concept of symmetry-curvature was used to explain the process which deformed the object. Symmetric axes show directions along which deformation process most likely took place. In [152], Leyton proposed a theory of nested structures of control which, he argues, governs the functioning of the human perceptual system. It is a hierarchical system where at each level of control all levels bellow are also included in information processing.

Pentland [215, 216] investigated methods for representation of natural form by means of fractal geometry [167, 168, 14, 53, 37, 75]. He argued that fractal functions are appropriate for

natural shape representation because many physical processes produce fractal surface shapes. This is due to the fact that natural forms repeat whenever possible and non-animal objects have a limited number of possible forms [287, 282]. Most existing schemes for shape representation were developed for engineering purposes as opposed to the purpose of perception. Fractal representation can produce objects which much better correspond to the human model of visual perception and cognition.

Lowe [165] proposed a computer vision system that can recognize three-dimensional objects from unknown viewpoints and single two-dimensional images. The procedure is non-typical and uses three mechanisms of perceptual grouping to determine the three-dimensional knowledge about the object as opposed to a standard bottom-up approach. The disadvantage of bottom-up approaches is that it takes an extensive amount of information to perform recognition of the object. Instead, the human visual system is able to perform recognition even with very sparse data or partially occluded objects. The conditions that must be satisfied by perceptual grouping operations are as follows.

- The viewpoint invariance condition. This means that observed primitive features must remain stable over a range of viewpoints.

- The detection condition. There must be enough information available to avoid accidental mis-interpretations.

The grouping operations used by Lowe are the following. Grouping on the basis of proximity of line end points was used as one viewpoint invariant operation. The second operation was grouping on the basis of parallelism which is also viewpoint independent. The third operation

was grouping based on collinearity. Preprocessing operation consisted of edge detection using Marr's zero crossings in the image convolved with a Laplacian of Gaussian filter. In the next step a line segmentation was performed. Grouping operations on a line segmented data were performed to determine possible locations of objects.

3.3   Boundary Scalar Transform Techniques

Boundary scalar transform algorithms typically consist of two major steps. In the first step, a one-dimensional function is constructed from the two-dimensional shape boundary. The constructed one-dimensional function is used in the second step to describe the shape of the two-dimensional boundary. Note that, in this approach, the shape is described *indirectly* by means of a one-dimensional characteristic function of the boundary instead of the two-dimensional boundary itself. Techniques used in the second step of this approach are divided into those based on the Fourier transform of the characteristic function and those based on a stochastic process modeling of the characteristic function.

3.3.1   From 2-D Shape to 1-D Boundary Representation

The methods in this group first generate a one-dimensional real or complex function from the shape boundary. 2-D shape can be represented using a real or complex 1-D function. In this Section we present several possible 1-D boundary representations of the shape that have been used in the literature.

Zahn and Roskies [312] and Bennet and McDonald [19] have used a tangent angle versus arc

fig/centroid.ps

Figure 3.1: The centroid-to-boundary distance approach.

length function. The tangent angle at some point is measured relative to the tangent angle at the initial point.

Granlund [102], Richards and Hemami [240], and Persoon and Fu [217] have used a complex function of the form $x(t) + jy(t)$, where t is the arc length parameter.

Another way to create a one-dimensional representation of the boundary is to measure the distance from the shape centroid to the points on the shape boundary. There are several possible ways to construct the required one-dimensional function from the given shape using the centroid-to-boundary distance approach. Here, we mention three centroid based procedures for constructing the required function. The shape, its centroid and boundary points are shown in Figure 3.1.

In the first method, lines are drawn from the centroid O to boundary points $A_0, \ldots, A_{N-1}$ so

that angles between lines are constant and equal $2\pi/N$. The function is now constructed as

$$f(i) = d(O, A_i) \qquad (3.1)$$

where $i = 0, \ldots, N - 1$ and $d(A, B)$ is the Euclidean distance between the points A, B.

In the second method, the distance between subsequent boundary points is used for the one-dimensional function values. Using the same notation we have

$$f(i) = \begin{cases} d(A_i, A_{i+1}) & \text{for} \quad 0 \leq i < N - 1 \\ d(A_{N-1}, A_0) & \text{for} \quad i = N - 1 \end{cases} \qquad (3.2)$$

where $i = 0, \ldots, N - 1$.

In the third method, a shape boundary is approximated with a polygon of N sides so that all sides have the same length. Let A_i be the polygon vertices where $i = 0, \ldots, N - 1$. In addition, let $a(\overline{AB})$ be the function which returns the angle between the line $\overline{AB}$ and a reference line. The one-dimensional function is then given by

$$f(i) = \begin{cases} a(\overline{A_i A_{i+1}}) & \text{for} \quad 0 \leq i < N - 1 \\ a(\overline{A_{N-1} A_0}) & \text{for} \quad i = N - 1 \end{cases} \qquad (3.3)$$

where $i = 0, \ldots, N - 1$.

Chang, Hwang, and Buehrer [35] constructed the function of distance from the centroid to the feature points. In their method, the feature points are the points of high curvature. There are

two approaches for detection of high curvature points. The first is to compute curvature directly from the boundary curve. The second approach is to perform a polygonal approximation of the shape and use the polygon vertices as feature points. This is based on the fact that corners are points of high curvature. The computed distances are saved in an linked list. The distances are ordered to achieve rotation invariance. The distances are also divided by the minimal distance to achieve scale invariance. Translation invariance is automatically achieved through the use of centroid.

3.3.2   Fourier Transform of Boundary

The shape description methods under this category use the Fourier Transform of the 1-D boundary representation to characterize the shape.

The Zahn and Roskies method [312] is presented in more detail to illustrate this approach to shape description. Let f_k, $k = 0, \ldots, L - 1$ be the tangent angle at the point k measured relative to the angle at point zero (i.e. $f_0 = 0$). An example shape and its tangent angle are shown in Figure 3.2. Let l_k be the arc length from the initial point to the point k and L be the total arc length. The boundary is parsed in a clockwise direction producing negative angles relative to initial point. Consider a new function which depends on parameter ranging from zero to 2π.

$$a(t_k) = f_k + t_k \tag{3.4}$$

fig/ftbound.ps

Figure 3.2: The tangent angle function.

where

$$t_k = \frac{2\pi}{L} l_k \tag{3.5}$$

and $0 \leq t_k < 2\pi$. The second summand in Equation 3.4 is added to compensate for the negative slope of function f_k. The behavior of the resulting function is such that in the case of the circle or regular polygon the function a_k is equal to zero. If non-regular shape is described a non-zero a_k function results. It can be shown that

$$a_0 = a_{2\pi} = 0 \tag{3.6}$$

The Fourier transform is now applied to the function a and resulting coefficients are used for

shape description. The complex shape descriptors are given as

$$S_n = \frac{1}{2\pi} \int_0^{2\pi} a(t)e^{-jnt}dt \tag{3.7}$$

Due to the arc length normalization the shape descriptor is invariant to scale change. The shape descriptor is invariant to translation because the tangent angle function is invariant to the shape position. Rotation of the object (i.e. the variation of the starting point) causes phase change in the resulting Fourier transform, therefore looking at the magnitude of the Fourier coefficients will ensure rotation invariance of the method.

The major advantage of the method is that it is easy to implement and is based on a well developed theory of Fourier analysis. The disadvantage is in the fact that Fourier transform does not provide local shape information. After the Fourier transform the local shape information gets distributed into all coefficients and is not localized in the frequency domain. Tangent angle versus arc length representations suffer from a very high noise sensitivity because it is difficult to determine the tangent angle for noisy contours.

3.3.3 Bending Energy Concept

Young, Walker, and Bowie [309] proposed an interesting notion of bending energy. According to this approach a shape can be represented by its bending energy defined by

$$E = \frac{1}{P} \int_0^P |K(p)|^2 dp \tag{3.8}$$

61

where $K(p)$ is the curvature function, p is the arc length parameter, and P is the total curve length. To actually compute the bending energy the Equation 3.8 was not used directly but instead the Fourier transform of the boundary was computed first. Using Fourier coefficients and Parseval's relation the bending energy was computed in a more efficient way. In addition, the authors proved that the circle was the shape having the minimum average bending energy.

3.3.4   Stochastic Methods

Most methods in this class are based on the modeling of the centroid to boundary points distance function. The process of distance function formation from the object shape is presented in the Section 3.3.1. The distance function is modeled using various stochastic process models. The parameters of the model are then used as shape descriptors.

On the terminology side, a notion of *time-series* is often used in stochastic processes for signals that depend on time. In stochastic shape boundary analysis, a distance function is modeled instead a time function. Regardless of this fact, in some literature, the term "time-series modeling" is still used to refer to the shape boundary problem as mentioned above.

Kashyap and Chellappa [137] proposed a use of circular autoregressive models (CAR) for representation of centroid to boundary points distance function. A CAR model is characterized by a set of unknown parameters and an independent noise sequence. Let r_t be a distance function where $t = 1, \ldots, N$. Since boundary is closed, r_t is a periodic function

$$r_k = r_{k+N} \tag{3.9}$$

A particular CAR model that was utilized is the same one used by Huang [130]. It is a stochastic process defined with the following m-th order difference equation

$$r_t = \alpha + \sum_{j=1}^{m} \theta_j r_{t-j} + \sqrt{\beta} \omega_t \tag{3.10}$$

where $t = 1, \ldots, N$ and

$$\omega_i = \omega_{i+N} \tag{3.11}$$

$$E[\omega_i] = 0 \tag{3.12}$$

$$E[\omega_i \omega_j] = \beta \delta_{ij} \tag{3.13}$$

and $\{\omega_1, \ldots, \omega_N\}$ are independent random variables with normal distribution $N(0, 1)$. The Kronecker's delta function is defined as

$$\delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases} \tag{3.14}$$

The parameters $\{\alpha, \theta_1, \ldots, \theta_m, \beta\}$ are unknown and need to be estimated. The maximum likelihood (ML) parameter estimation [230, 305, 201] was used. The ML estimated parameters θ_i are translation, rotation, and scale invariant. Note that the rotation invariance holds only for angles which are multiples of $2\pi/N$. Parameters α and β are not scale invariant but the quotient $\alpha/\sqrt{\beta}$ is. Therefore, the vector $[\theta_1, \ldots, \theta_m, \alpha/\sqrt{\beta}]^T$ is used as a shape descriptor. Kashyap and Chellappa further investigated coding and reconstruction schemes and showed that stochastic

63

methods could be used for description of closed boundaries.

Dubois and Glanz [65] used the same autoregressive model as in [137] but investigated three additional methods for improving pattern classification (shape matching) performance. The classification was then performed by computing the weighted Euclidean distance between unknown object descriptors and training objects. The first improvement was the weighting of the descriptor vector so that components which were common within a training class were emphasized while components which vary were de-emphasized. It has been shown that the optimal feature weight is inversely proportional to the standard deviation of the feature of the class training set [261]. The second improvement consisted in the rotation of the coordinate system and scaling the rotated axes [261]. This groups the members of one pattern class closer in the new coordinate system. The third modification included the use of hyperplanes to divide the pattern space. The least mean squared error procedure [54] yields the optimal hyperplane parameters. The experimental result showed that the normalized AR model parameters useful shape descriptors.

Das, Paulik, and Loh [45] developed a bivariate technique for autoregressive modeling of shape boundary. Using their technique they obtained even better experimental results.

The linear AR model has been compared with the non-linear model of the quadratic Volterra type given by

$$r_t = \sum_{j=1}^{m} a_j r_{t-j} + \sum_{u=0}^{p} \sum_{v=0}^{q} g_{uv} r_{t-u} r_{t-v} + e_t \tag{3.15}$$

where r_t is the centroid to boundary points distance function [136]. According to Kartikeyan and Sarkar, the linear AR models may not be sufficient for recognition of more complicated (non-convex) shapes. For better accuracy a higher order linear model is necessary to increase

fig/archeight.ps

Figure 3.3: The arc height concept.

dimension of the shape descriptor vector. The use of overfitted AR models may lead to poor recognition performance. However, non-linear models can provide higher accuracy necessary to describe more complicated shapes. The experiments demonstrated that the quadratic Volterra models performed better classification then the linear AR model [136].

The disadvantage of AR boundary modeling is that in the case of complex boundaries a small number of AR parameters is not sufficient for description. For this reason, He and Kundu [112] combined the use of the AR model with the hidden Markov model. The shape boundary was partitioned into a number of segments and each segment was described using the AR modeling. The obtained vectors were analyzed using the hidden Markov model.

3.3.5 Arc Height Method

Lin, Dou, and Wang [155] used the Arc Height Function (AHF) to characterize the shape boundary. The arc height concept is shown in Figure 3.3 and is defined as follows. An arc chord $\overline{AB}$

of predefined length is positioned on the boundary. The symmetric axis passing through $\overline{OC}$ is drawn perpendicular to the arc chord $\overline{AB}$. The length of the line segment $\overline{OC}$ is called the arc height at the position A. As arc chord is moved along the curve a mapping between the arc length and the arc height defines the AHF. The AHF is then used to characterize the shape.

3.4 Boundary Space Domain Techniques

The boundary space domain methods take shape boundary as input and produce result in the pictorial or graph form. Space domain techniques often appear in various structural approaches to shape recognition. The reason for this is that in structural methods a recognition system processes the visual information in stages starting from the early preprocessing phase to a higher levels where the final interpretation of the visual scene is performed. A characteristic of these processing stages is that they produce an image, a graph, or other non-scalar as opposed to approaches described in Section 3.3 which produce a scalar result. This is why such methods are called the space domain methods. Some examples of the space domain techniques are presented in the following sections.

3.4.1 Chain Code

Freeman [77] has proposed a method for coding of line drawings called the *chain coding*. A more detailed overview of the chain code methods and algorithms by Freeman can be found in [78]. The generalized chain code [80] is shown in Figure 3.4. The nodes surrounding the node A are enumerated counter-clockwise in ascending order from inside out. Ring 1 contains nodes

fig/chaincode.ps

Figure 3.4: The generalized chain code.

$0 - 7$, Ring 2 contains nodes $8 - 23$, Ring 3 contains nodes $24 - 47$, etc. A generalized chain representation is denoted by rings that are used. For example, a $(1,3)$ code uses line segments from point A to all points from ring 1 and ring 3. Code (1) is the basic chain code with eight segments from A to points from ring 1. A conventional code (1) is used in the following text to introduce some basic notation [78]. A *link* a_i is a directed line segment of length $\sqrt{2}$ for directions $i = 1, 3, 5, 7$ and of length one for directions $i = 0, 2, 4, 6$ (assuming grid spacing is equal to one). A *chain* is an ordered sequence of links written in the form

$$A = a_1 a_2 a_3 \ldots a_n \tag{3.16}$$

The links a and b form an *inverse pair* if

$$a = b + 4(mod8) = b^{-1} \tag{3.17}$$

67

Note that the addition in Equation 3.17 is performed modulo eight. Inverse of a chain is computed as

$$(a_1 \ldots a_n)^{-1} = a_n^{-1} \ldots a_1^{-1} \tag{3.18}$$

Length of the chain can be computed as

$$L = n_e + n_o \sqrt{2} \tag{3.19}$$

assuming the grid spacing of one. n_e is the number of links having even code while n_o is the number of links with odd code. Integration of a function specified by its chain code and the function value at the starting point of the chain is given by

$$I = \sum_{i=1}^{N} a_{ix} \left(y_{i-1} + \frac{1}{2} a_{iy} \right) \tag{3.20}$$

where a_{ix} and a_{iy} are x and y components of the link a_i. and

$$y_i = y_{i-1} + a_{iy} \tag{3.21}$$

and y_0 is the ordinate of the chain starting point. The first moment about x axis is computed as

$$m_{01} = \sum_{i=1}^{N} \frac{1}{2} a_{ix} \left[y_{i-1}^2 + a_{iy} y_{i-1} + \frac{1}{3} a_{iy}^2 \right] \tag{3.22}$$

68

The second moment about x axis is computed as

$$m_{02} = \sum_{i=1}^{N} \frac{1}{3} a_{ix} [y_{i-1}^3 + \frac{3}{2} a_{iy} y_{i-1}^2 + a_{iy}^2 y_{i-1} + \frac{1}{4} a_{iy}^3] \tag{3.23}$$

The distance between two points connected by the chain A can be computed as

$$m_{02} = [(\sum_{i=1}^{N} a_{ix})^2 + (\sum_{i=1}^{N} a_{iy})^2]^{1/2} \tag{3.24}$$

The above basic operations should illustrate the flexibility and versatility of the chain code for algorithm realization.

In [79], Freeman used a chain code description of the boundary to extract the critical points which he use then to generate shape description which is invariant to translation, rotation, and scale. He also presented another chain-code approach which is based on the centroidal profile of the shape. The centroidal profile is a plot of centroid to boundary points distance.

Parui and Majumder [202] used a modified chain code to perform a symmetry analysis. The shape boundary is represented in a hierarchical way so that at highest level a lower number of polygon vertices is used while at the lowest level the finest polygonal approximation is utilized. The search for the symmetric axis position is done starting at highest level with transition to lower levels as the position of the symmetry axis is more accurately determined.

3.4.2   Syntactic techniques

Structural descriptions may be viewed as graphs and as such are suitable for formulation in terms of formal languages [81, 205, 83, 84, 117]. Syntactic methods have several important advantages. They attempt to emulate the structural and hierarchical nature of the human visual perception system. Another advantage is that the theory of formal languages, that syntactic methods rely on, is a well developed field. The main disadvantage of this approach is that shape (or shape boundary) must first be encoded to provide the input to the parser. Typically a low level segmentation of the image must be performed to extract different types of line and curve segments and corners [209]. The obtained boundary features are then formed in a string.

$$S = s_1, s_2, \ldots, s_n \qquad (3.25)$$

String element s_i can represent different things like a chain-code element, a side of polygonal approximation, or an arc. The string of feature symbols is then parsed according to a grammar to detect the shape of the object. In addition to deterministic (nonstochastic) grammars stochastic grammars have been investigated [81]. Fu [82] presented the method for image modeling based on stochastic grammars. Attributed grammars manipulate attributes (semantics) at the time of parsing the grammar (syntax). Attributes are usually some primitive shape features.

The theory of formal languages, established by Noam Chomsky, has been used in many fields including compiler design, automata theory, computer languages, pattern recognition, and image processing. First, the basic terminology of formal languages is introduced. An *alphabet* is a set

of words (symbols). Words are combined together to form a *sentence*. A *language* is a set of sentences that can be composed using the words from the alphabet. Formal languages are defined using grammars. Grammars are sets of syntax rules describing how sentences can be generated using available vocabulary. A grammar G is a the quartet $G = (N, \Sigma, P, S)$, where N denotes the set of nonterminal symbols, Σ is the set of terminal symbols, P is the set of production rules, and $S \in N$ is the start symbol. The language $L(G)$ is a set of sentences generated by the grammar G. The sentences have the following properties.

1. Each sentence is composed of terminal symbols only.

2. Each sentence can be derived from S using the appropriate sequence of production rules from P.

Formal grammars are divided into four classes. Type 0 grammars have the unrestricted form of production rules. Type 1 are context-sensitive grammars where productions depend on the context. Type 2 are context-free grammars and type 3 grammars are regular grammars. Context-free and regular grammars have been used most in practise [84].

3.4.3 Boundary Approximations

Two most popular schemes for curve approximation are polygonal and spline approximations.

Polygonal approximations are used to approximate the shape boundary using the polygonal line. Methods are based on the use of the minimal error, the minimal polygon perimeter, the maximal internal polygon area, or the minimal external polygon area as approximation criteria.

The error measures which are used most are maximal error (yielding various minimax methods) and integral square error.

One of the most popular methods in this group is the *split-and-merge* algorithm [208]. In this approach, a curve is split in segments until some acceptable error is obtained. At the same time split polygonal segments are merged if the resulting segment approximates the curve within some maximum error. Pavlidis [204] used partial derivatives of the integral square error function to direct Newton's method in search for the optimal breakpoints.

Wu and Leou [306] used a different optimization criteria to derive polygonal approximations. The internal maximum area, the external minimum area, or the minimum area deviation polygonal approximations were used in their work.

Bengtsson and Eklundh [18] presented a hierarchical method where the shape boundary is represented by polygonal approximation. The split-and-merge algorithm was used to create the polygonal approximation. The scale-space approach [303] was used to track the position of inflection point on the boundary curve. Stable shape features are those which remain unchanged over scale. Tangents at border points are estimated using polynomial approximation to yield a multi-scale representation of the curve.

Splines have been very popular for interpolation of functions and approximation of curves. Ikebe and Miyamoto [131] wrote an overview of spline applications for shape design, representation, and restoration. Mathematical treatment of splines is presented in several books [50, 9, 139] while the computer graphics perspective is presented in [91, 195, 15, 245].

Let $t_0, t_1, \ldots, t_n$ be points satisfying $t_0 < t_1 < \cdots < t_n$. These points are called knots. A

function S of degree $k \in N$ is called a spline function of degree k if it satisfies the following conditions.

1. S is a polynomial of degree $\leq k$.

2. S has a continuous $k - 1$-st derivative on $[t_0, t_n]$.

The above conditions are used to determine the actual spline coefficients for each interval $[t_i, t_{i+1}]$. Cubic splines $(k = 3)$ are used most in applications. Splines have a very nice property of minimizing curvature. In other words, they approximate a given function with a curve having the minimum average curvature. This makes them perfect candidates for a "natural" representation of curves. The disadvantage of splines in interpolation problems is that a local function value change changes complete spline representation. B-splines are constructed so that a local function value change does not spread to the rest of the intervals. B-splines have the following properties.

1. $N_{i,k}(t) = 0$ for $t \leq t_i$ and $t/geqt_{i+k+1}$.

2. $N_{i,k}(t) > 0$ for $t \in (t_i, t_{i+k+1})$.

3. $\sum_i N_{i,k}(t) = 1$.

4. $N_{i,k}(t)$ has a continuous $k - 1$-st derivative.

5. $N_{i,k}(t)$ can be computed using a recursive formula.

where $N_{i,k}(t)$ is a k-th order B-spline.

B-splines can be used for interpolation of plane curves given by parametric equations $(x(t), y(t))$. Each parametric equation is interpolated independently in that case.

Surface are often represented by means of a family of curves. The simplest solution is just a pointwise linear interpolation between curves. This technique is called *lofting* and is used widely in shipbuilding and aircraft industries [131].

3.4.4 Scale-Space Techniques

This groups contains methods which rely on the scale-space representation. Witkin [303] proposed a scale-space filtering approach which provides a representation useful for representing significant object features. The representation was created by tracking the position of inflection points in signals filtered by low-pass Gaussian filters of variable widths. The inflection points which remain present in the representation were expected to be "significant" object characteristics.

Babaud et al. [12] proved the uniqueness of Gaussian kernel for scale-space filtering. Gaussian kernel has the desirable property of saving inflection points when width of the filter is increased. In other words, contours in scale-space image cannot disappear as filter width is increased. The only filter having such property is the Gaussian filter.

Asada and Brady [7] proposed a new approach based on ideas developed by Marr et al. to introduce a representation called the *curvature primal sketch*. It is a scale-space approach for representation of curvature and is similar to Marr's primal sketch for edge detection. The shape boundary is filtered with Gaussian functions of increasing width to obtain a multi-scale representation of shape boundary. The curvature is then computed at different scales to obtain the curvature primal sketch.

Mokhtarian and Mackworth [193] applied the scale-space approach to the description of planar shapes using the shape boundary. The curvature along the contour was computed and smoothed with variable width Gaussian filters. The scale space image of the curvature function was used as a hierarchical shape descriptor which is invariant to translation, scale, and rotation.

The concept of multiscale filtering is also present in mathematical morphology. Chen and Yan [36] used a variable size structuring element to perform various morphological operations.

Dill et al. [56] studied the role of the leukocyte locomotion. They created multiple smoothed versions of a leukocyte boundary to extract skeletons. The boundary was represented using the chain 1-code. A Gaussian filter of variable width was used to create smoothed versions of the boundary. The skeleton was computed using a technique based on Arcelli's algorithm [6]. This algorithm improves the noise sensitivity and the sensitivity to global convexities.

3.4.5   Boundary Decomposition

The methods in this group decompose the shape boundary into curve segments.

H. Liu and M. Srinath [156] developed a procedure for shape classification using contour matching. The input to the procedure was an object shape. The Sobel edge detector was used to compute x and y direction gradients and the tangent angle along the boundary by

$$\theta = \tan^{-1} \frac{\Delta_y}{\Delta_x} \tag{3.26}$$

The tangent angle function was convolved with the derivative of Gaussian function to find the smoothed curvature function. The boundary was segmented at the points of high curvature. The

75

curve matching was performed in two steps. In first step, individual segments for two shapes were compared. In second step, groups of segments were compared and group was disqualified if less then three consecutive segments matched. The final step was used to measure the distance between two shapes. This was done using the chamfer 3/4 distance transform [28, 29] which approximates the Euclidean distance transform very good but is not so computationally intensive. The chamfer 3/4 distance transform was computed for the first boundary. The second boundary was superimposed on the distance transform of the first boundary and the average distance value was computed. Experimental results demonstrated the feasibility of the method for shape matching.

3.5 Global Scalar Transform Techniques

Methods classified here compute a scalar result based on the global shape. Moment based methods are among most popular methods from the group of global (or internal) scalar transform methods. Shape vectors and matrices are among less known methods for shape description.

3.5.1 Moments

Moments were first used in mechanics for purposes other then shape description. Recent surveys of the field are written by Prokop and Reeves [238] and with more general approach by Weiss [299].

Two-dimensional Cartesian moment m_{pq} of order $p + q$ of a function $f(x, y)$ is defined as

$$m_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q f(x, y) dx dy \qquad (3.27)$$

76

In discrete case we have

$$m_{pq} = \sum_{0}^{M-1} \sum_{0}^{N-1} x^p y^q f(x, y) \tag{3.28}$$

where M, N are the image size parameters. The use of moments for shape description was initiated by Hu [128]. He proved that the moment-based shape description is information preserving. Moments m_{pq} are uniquely determined by the function $f(x, y)$ and vice versa the moments m_{pq} are sufficient to accurately reconstruct original function $f(x, y)$. The zeroth order moment m_{00} is equal to the shape area assuming that $f(x, y)$ is a silhouette function having value one within the shape and zero outside the shape. First order moments can be used to compute the coordinates of the center of mass as

$$x_c = \frac{m_{10}}{m_{00}} \tag{3.29}$$

$$y_c = \frac{m_{01}}{m_{00}} \tag{3.30}$$

Second order moments are called moments of inertia. They can be used to determine the principal axes of the shape. Principal axes are axes with respect to which there are maximum and minimum second order moments.

Moments of projections are actually one-dimensional moments of projection functions. Let

$$v(x) = \int_{-\infty}^{\infty} f(x, y) dy \tag{3.31}$$

be a projection of function $f(x, y)$ to the x axis. The projection moments to x axis are then

77

given as

$$m_p = \int_{-\infty}^{\infty} x^p v(x)dx = m_{p0} \tag{3.32}$$

For successful application to shape description a method must be invariant to translation, rotation, and scale. Therefore the moments defined in Equation 3.27 are not useful for shape description. Hu [128] proposed seven invariant moments (also called moment invariants). These moments do not depend on position, orientation, or scale of the shape.

A generalization of moment transform to other transform kernels is possible if Equation 3.27 is rewritten as

$$m_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} P_p(x)P_p(y)f(x,y)dxdy \tag{3.33}$$

A conventional *transform kernel* $x^p y^q$ is replaced by a more general form $P_p(x)P_p(y)$. Particularly appealing is to use *orthogonal* polynomials for moment transform kernel [285]. In this case the moments produce a minimal information redundancy [238]. This is important for the optimal utilization of the information available in a given number of moments. Some of the orthogonal polynomial systems that have been investigated [285] include Legandre, Zernike, and pseudo-Zernike polynomials.

The advantage of moment methods is that they are mathematically concise. The disadvantage is that it is difficult to correlate high-order moments to shape features. As with most scalar transform methods the local information and shape features cannot be detected.

Another transform approach is the Fourier transform of the shape. Similarly to moment approach it is impossible to 'sense' local shape features and it is also computationally intensive.

fig/shapematrix.ps

Figure 3.5: The shape matrix concept.

The problem of invariance to translation, scale, and rotation is also present in this approach.

3.5.2 Shape Matrices and Vectors

Shape matrix and vector approaches use a global shape information to create a numerical (matrix or vector) description of the shape.

Goshtasby [99] used a matrix to represent the pixel values corresponding to a polar raster of coordinates centered in the shape center of mass. The idea is illustrated in Figure 3.5. A polar raster of concentric circles and radial lines are positioned in the shape center of mass. Line $\overline{OA}$ represents the axis of the polar coordinate system. The binary value of the shape is sampled at intersections of circles and radial lines. The shape matrix is formed so that the circles correspond to matrix columns and radial lines correspond to matrix rows. The scheme is

invariant to translation, rotation, and scaling.

Algorithm 3.1 *The shape matrix description method.*

1. *Let L be the maximum radius of the shape S*

2. *Create an $m \times n$ matrix and call it M*

3. **For** $i = 0$ **to** $n - 1$

   (a) **For** $j = 0$ **to** $m - 1$

      i. **if** $(iL/(n-1), 2\pi j/m) \in S$

         then $M(i, j) = 1$

         else $M(i, j) = 0$

The maximum radius of the shape is equal to the radius of the circle centered in the shape center of mass which contains the shape.

Example 3.1 The shape matrix for shape shown in Figure 3.5 is given as follows. In this

example, $m = 8$ and $n = 7$.

$$M = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 \end{bmatrix} \qquad (3.34)$$

□

In the method used by Taza and Suen [284], shapes were described by means of shape matrices and a comparison of matrices was performed to classify unknown shapes into a one of the known classes. A scheme for the weighting of matrix entries was developed for more objective comparison. The weighting was based on the fact that sampling density is not constant with the polar sampling raster. Without weighting the innermost shape samples are implicitly given much more importance then peripheral shape pixels since sampling density is much higher in the center of the shape.

Parui, Sarma, and Majumder [203] proposed a shape description scheme based on the relative areas of the shape contained in concentric rings located in the shape center of mass. Let L be the maximum radius of the shape S to be described. Let T_k be the k-th ring of n concentric rings

obtained by sectioning the maximum radius L into n equal segments. Note that $S \subset \cup_{i=1}^{n} T_i$. Let

$$x_i = \frac{A(S \cap T_i)}{A(T_i)} \tag{3.35}$$

where $A(.)$ is the function that returns the area of its argument. In other words x_i is the area of shape S contained in ring T_i relative to the area of the ring itself. The shape descriptor vector is formed as $x = [x_1 \ldots x_n]^T$. The authors demonstrated that the shape vector scheme can be used for shape matching.

3.5.3   Morphological Methods

A number of approaches to shape description based on mathematical morphology have been investigated. In this section, shape description methods based on the area of morphologically processed images are presented.

Maragos [170, 173] proposed the concept of pattern spectrum. The pattern spectrum for continuous images is defined by

$$PS_X(r, B) = \frac{-dA(X \circ rB)}{dr} \tag{3.36}$$

where B is a unit disk structuring element and $A(X)$ denotes the area of the set X. The pattern spectrum of a set X is obtained by opening X by a disk of variable size. The areas of the resulting sets were measured. The pattern spectrum is defined as the derivative of the area function with respect to r, the radius of the disk structuring element. The definition for discrete images is as

82

follows.

$$PS_X(n, B) = A[X \circ nB - X \circ (n+1)B] \text{ for } n \geq 0 \tag{3.37}$$

$$PS_X(-n, B) = A[X \bullet nB - X \bullet (n-1)B] \text{ for } n \geq 1 \tag{3.38}$$

Similar expressions are also presented for gray scale morphology.

The pattern spectrum approach is related to a notion of granulometries first studied by Matheron [184] and more recently by Dougherty [60, 59, 63]. Granulometries are a result of a sieving operation applied to binary images. It is a sieving operation because the structure in the image is filtered according to components (or particles) size. Let E_k be a sequence of structuring elements such that $E_{k+1} \circ E_k = E_{k+1}$. Then

$$S \circ E_1 \supset S \circ E_2 \supset \cdots \tag{3.39}$$

The sequence of openings $S \circ E_k$ is called a granulometry. Let $A(k) = Area(S \circ E_k)$ be a sequence of areas of successive openings. The decreasing function $A(k)$ is called a size distribution. Note that the negative derivative (or difference in discrete case) of the area sequence is equal to the Maragos' pattern spectrum.

Shih and Pu [275] proposed another "spectrum" transformation which is called G-spectrum. It is an extension of the work of Goutsias and Schonfeld (see Section 3.6.2). G-spectrum is defined by

$$G_n(X) = \frac{Card(X \ominus A_n) - Card(\psi_n[(X \ominus A_{n+1})])}{Card(X)} \tag{3.40}$$

The authors proved that their representation is less redundant then granulometric size distributions or pattern spectrum.

Another set of techniques were derived from the concept of morphological covariance [262]. Loui et al. [163] used the geometrical correlation function (GCF) for representation of two-dimensional shapes. Let $X \subset R^n$ be a compact set and B_h^ϕ be a structuring element consisting of two points at a distance h at an angle ϕ relative to 0^0. The morphological covariance [262] is defined by

$$C(h) = \text{Mes} \ [X \ominus B_h^\phi] \tag{3.41}$$

where Mes is a set measure which returns the area of the binary set. Note that

$$X \ominus B_h^\phi = X \bigcap X_{-h}^\phi \tag{3.42}$$

is the intersection of the original set X with its translated version. The translation is done into the direction ϕ. Some properties of covariance function are

$$C(0) = \text{Mes} \ [X] \tag{3.43}$$

$$C(\infty) = 0 \tag{3.44}$$

$$C(h) = C(-h) \tag{3.45}$$

$$C(h) \leq C(0) \tag{3.46}$$

The GCF is defined by

$$K_\phi(h) = \frac{\text{Mes } [X \ominus B_h^\phi]}{\text{Mes } Y} \tag{3.47}$$

where Y is a predefined shape. It is also possible to use $Y = X$. The authors used GCF for shape description and matching. The method has rotation and translation invariance. If scale invariance is desired a preprocessing step of rescaling can be added. The shape descriptor is either maximum or minimum K_ϕ function. The maximum $K_{\phi max}$ and minimum $K_{\phi min}$ have the following property.

$$A[K_{\phi max}] \geq A[K_\phi] \ \forall \phi \in [0, \pi] \tag{3.48}$$

$$A[K_{\phi min}] \leq A[K_\phi] \ \forall \phi \in [0, \pi] \tag{3.49}$$

where $A[K_\phi]$ is computed by

$$A[K_\phi] = \sum_{h=0}^{N-1} K_\phi(h) \tag{3.50}$$

The rotation invariance was achieved through the use of minimum or maximum correlation functions. The experimental results have shown that the method is useful for shape matching.

Maragos [171, 172] related the mean absolute error criterion for signal matching to a morphological correlation function. The morphological cross-correlation function is defined by

$$\mu_{fg}(k) = \sum_{n=-\infty}^{\infty} \min[f(n+k), g(n)] \tag{3.51}$$

where $n, k \in Z^d$, $d \in N$ and $f, g : Z^d \longrightarrow R$. $\mu_{ff}(k)$ is the morphological auto-correlation

function. The morphological cross-correlation function can be related to the minimum absolute error matching as follows. Since $|a - b| = a + b - 2\min(a, b)$ the minimization of the absolute error is equivalent to a maximization of $\min(a, b)$. If mean error is minimized then the later term has the form of the morphological cross-correlation defined in Equation 3.51. A similar argument has been used in the past for minimization of the mean square error. In that case $(a - b)^2 = a^2 + b^2 - 2ab$ and minimization of the mean square error leads to a maximization of conventional linear cross-correlation function between signals.

3.6 Global Space Domain Techniques

Global space domain methods are based on the analysis of the global shape. The resulting shape descriptor is a non-scalar (e.g. a graph or an image). Most representative methods from this group are discussed in the following sections.

3.6.1 Medial Axis Transform

The most popular and the most studied global space domain method is the *medial axis* transform (MAT) originally proposed by Blum [23, 24]. The idea of these approaches is to represent the shape using a graph and hope that important shape features are preserved in the graph. On terminology side, a term the medial axis was originally used by Blum. A skeletal pair consisting of the skeleton and the "quench" function is used by Calabi [33]. The terms the *prairie fire* transform, *symmetric axis* transform, and *skeleton* transform have been used in literature to refer to the same approach. Additional material on the topic can be found in [196, 66, 213, 199, 47].

This approach is motivated by the study of neural physiology and visual psychology. In particular, Blum hypothesized that a process of image formation on the retina is a chain reaction in the following sense. When an object image is formed on the retina a certain number of excited neurons are fired, lowering the excitation levels of neighboring neurons and causing them to fire a short interval of time later (inhibition). The process is repeated until the whole area of the object is "tiled" with firing neurons. The inhibited neurons cannot fire again for some short time due to underlying neurophysiological processes [98]. Therefore the wave front of the firing cells cannot move back towards the retinal areas containing inhibitory neurons. This mechanism is similar to the spreading of prairie fire. In fact, the first approach Blum used was a temporal function showing the arc length of wave front versus time. This approach did not show very useful for shape description purposes. Blum's second concept - a concept of medial axis has proven to be more useful for shape description purposes.

The purpose of the medial axis transform is to extract the skeletal, stick-like figure from the object. This figure can, hopefully, be used to describe the object shape. The formation of the skeleton can be explained on the following example. Let object interior be composed of a burnable dry grass while object exterior be composed of unburnable wet grass. If fire is set simultaneously at all points of the shape boundary, it will propagate towards the center of the shape. However, at some positions the wave front from one direction will intersect the fire wave front coming from another direction. Points of wave front collision are called quench points. The skeleton of the figure is defined as the set of quench points. More formally, let $d(x, B)$ be the

fig/skeleton.ps

Figure 3.6: The medial axis transform.

closest distance from the point x to the boundary B of the shape. It is defined as

$$d(x, B) = inf\{d(x, y) \mid y \in B\} \tag{3.52}$$

There are some points in the object shape with the following property

$$\text{for some } x \in B \ \exists \ a, b \in B \mid d(x, B) = d(x, a) = d(x, b) \tag{3.53}$$

The union of such points is called the medial axis (or skeleton) S of the shape. An illustration of this method is shown in Figure 3.6. The quench function of the skeleton is defined by

$$q(x) = d(x, B) \tag{3.54}$$

The pair (S, q) is called a skeletal pair. The skeletal pair is important because it contains sufficient information for complete shape reconstruction. The reconstruction process is as follows.

1. **for** each $x \in S$ **do**

   (a) Position a disk of radius $q(x)$ at location x

The union of all disks is equal to the original shape.

The wave front propagation can be computed using the distance transform methods [218, 255] or using the shrinking operation [196]. Gray scale images have been processed using the min-max operators [197, 213, 92]. The shrinking and related Boolean-nature local neighborhood operators have been used intensively in cellular array image processors [68, 237, 233, 249, 250, 234, 86, 236].

The disadvantage of the medial axis transform is that it is very sensitive to noise in the shape boundary. Even small changes in the shape can cause significant changes in the topology of the MAT graph. Another difficulty is in the practical realization in discrete spaces [189]. The change from continuous to discrete space causes some difficulties. For example, the MAT graph of a connected object may not be connected.

To fight the problem of noise, Blum and Nagel [25] proposed a *generalized medial axis* transform, based on the touching circle defined as a circle which is tangent to the shape boundary without intersecting it. The *r-symmetric axis* of the shape is defined as union of all points which have a touching circle of radius greater then r and has at least two touching points with the boundary. The requirement that the radius is greater then r prevents little noisy spikes to generate skeleton segments. The two touching point requirement limits the choice to the skeleton points only. The generalized skeleton has shown to have the better noise robustness [25]. The concept of symmetry and symmetric axis was developed further in works of Brooks [32] and Brady [31].

89

The problem of determining the axis of symmetry is related to an inverse problem - a generation of shapes from their symmetric axis. Object generated in such a way are called *ribbons*. The synthesis of ribbon-like object is done by means of the axis and the *generator*. The generator shape is a geometric figure which moves along the axis and changes size as it moves. In Blum's approach a disk was used as the generator and constructed shapes are called *Blum ribbons*. *L-ribbons* are generated using the line segment. L-ribbons with fixed angle between the line segment and the axis are called *Brooks ribbons* [32]. L-ribbons with fixed angle between the line segment and the side of the ribbon are called *Brady ribbons* [31]. Rosenfeld [252] and Ponce [229] did a comparison of various ribbon generating procedures.

In the following text we present several definitions of symmetry which have been proposed in the literature. The symmetry is always defined in terms of a condition that has to be satisfied for two points to form a symmetry. The line connecting two points is called a line of symmetry. The middle point is a point in the middle of the line of symmetry. The axis of symmetry is formed as a union of middle points.

The *skew* symmetry [135] is a symmetry where lines connecting corresponding (symmetrical) points are perpendicular to the axis of symmetry. This is a mirror-type symmetry. The notion of *parallel* symmetry was proposed by Ulupinar and Nevatia [290]. This symmetry assumes that there is a continuous monotonic mapping between parameters in parametric representation of symmetric curves so that tangent angles of both curves are the same. Let $c_i(p) = (x_i(p), y_i(p))$ where $i = 1, 2$ be the parametric curve equations and $\phi_i(p)$ be the tangent angle equation. For c_1 and c_2 to be parallel symmetric there must exist a function $f(p)$ such that $c_1(p) = c_2(f(p))$.

Brady [31] developed a notion of *smooth local* symmetry. Here, two points form a local symmetry if the angles between the curve normals at two given points and the line connecting the points are the same.

A hierarchical (multiresolution) approach to the skeleton analysis was presented by Pizer, Oliver, and Bloomberg [228]. In his approach, a hierarchy by scale was used to construct a series of skeletons. The multiscale approach has been demonstrated to improve the noise properties of the representation [228].

Rom and Medioni [246] proposed a hierarchical representation of shape using axial shape description. This approach combines several paradigms mentioned above. The original shape was broken into parts at negative curvature minima of the shape boundary [121]. The obtained parts were represented using smooth local symmetry ribbons. To capture the global relationship between parts the parallel was used. Typically, the procedure to determine the axes is computationally expensive and produces many unwanted axes. The unwanted axes can be eliminated using various approaches [42]. In the final stage, a recursive procedure was performed for shape decomposition. In each step of the recursive procedure the axial representation of the shape was made and its smallest parts removed. The shape was then recreated and the procedure repeated. The series of produced shapes represent the decomposition. In [256] B-splines were used to find a boundary approximation from the edge map. The procedures for computation of skew, parallel, and smooth local symmetries were presented for B-spline approximated boundary.

Leymarie and Levine [151] developed a new method for extraction of symmetry axis which does not suffer of discretization problems as many other algorithms do. The method was based

on the use of *snakes* for active contour representation, high curvature points on the boundary, and symmetric axis transform. The result was a dynamic multi-scale skeleton representation.

Axes of symmetry were primarily extracted from binary images but Gauch and Pizer [85] proposed a method for extraction of the *intensity axis* of symmetry from gray scale images. The method was applied to a shape-based image segmentation where it was possible to identify parts of the object corresponding to different components of the intensity axis of symmetry.

Maragos and Schafer [174] used mathematical morphology to extract skeleton subsets for efficient coding of binary images. Lantuejoul showed [262] that, for subsets of R^2, the skeleton could be computed by

$$SK(X) = \bigcup_{r>0} [(X \ominus rB) - (X \ominus rB) \circ drB] \tag{3.55}$$

where rB is the open disk of radius r and drB denotes a closed disk of infinitesimally small radius dr. Serra [262] showed that, for discrete binary images sampled on a hexagonal grid, the skeleton could be computed by

$$S_n(X) = (X \ominus nB) - (X \ominus nb) \circ B \tag{3.56}$$

$$SK(X) = \bigcup_{n=0}^{N} S_n(X) \tag{3.57}$$

where B is a unit disk consisting of seven pixels in hexagonal grid. The morphological skeleton

transformation has an exact inverse given by

$$X = \bigcup_{n=0}^{N} [S_n(X) \oplus nB] \tag{3.58}$$

This transform pair was the basis for binary image decomposition. The skeleton subsets S_n were encoded using block and run-length coding methods [130]. The coding was efficient because skeleton subsets are thin binary images.

3.6.2   Shape Decomposition

In shape decomposition techniques a shape is represented as a combination of component shapes. The idea is to represent a complex shapes in terms of simple components. Pavlidis stated the problem of global shape analysis (decomposition) as follows.

> Among the boundary points find sets of points which are closely related. Such sets
>
> may be used to assign labels to corresponding parts of the object [207].

In this approach, shape decomposition is based on the properties of boundary points. Several authors have used this approach for shape decomposition. Decomposition criteria can be defined in terms of the medial-axis transform, can require convex components, or visibility of boundary points. In the medial axis transform approach two boundary points are labeled related if they are both on the circle which is contained in the object shape. A decomposition criteria can be formulated to require that the line segment between two points on the boundary is contained in the shape that is described. This kind of criteria leads to decomposition in the convex components.

Boundary point clustering [266, 265] is a probabilistic method which requires that decomposition is done so that each point in the boundary is visible by most other points. Decomposition on the basis of k-nearest neighbors [22] related atoms corresponding to approximating polygon sides instead of boundary points. In stroke detector approaches points are related if they are close to each other across the boundary. Semantic considerations about shapes being described were taken into account in the method for shape decomposition by collinearity [138]. The method was based on heuristics based on human concepts of collinearity. Experiments verified the effectiveness of approach. Vanderheydt et al. [291] used fuzzy subset theory [310, 311] to direct decomposition based on convex and concave boundary points and polygonal approximation of the boundary. The method was applied to the problem of the chromosome shape analysis.

Pitas et al. [223, 220, 225, 222, 226] developed a morphological method for shape decomposition. This is an alternative for skeleton subset decomposition from given in Equation 3.57. Let B be a disk structuring element. Then the morphological decomposition proposed by Pitas et al. is given by the following recursive expression.

$$X_i \;=\; (X - X'_{i-1}) \circ r_i B \tag{3.59}$$

$$X'_i \;=\; \bigcup_{j=1}^{i} X_j \tag{3.60}$$

$$X'_0 \;=\; \emptyset \tag{3.61}$$

where r_i is the radius of the structuring element at level i. The object can be reconstructed using the following expression.

$$X = \bigcup_{i=1}^{k} X_i \tag{3.62}$$

k is the number of components $X_i \neq \emptyset$. Sets X_i are simple objects of the form $X_i = L_i \oplus r_i B$ where L_i is a point or a line. If B has a circular shape then X_i are called Blum ribbons (see Section 3.6.1). Hence, this decomposition technique breaks the shape into "simple" components which are Blum ribbons. This is an advantage since one of desired properties of a decomposition scheme is that the resulting components are intuitively simple.

Schonfeld and Goutsias [259, 260, 100] proposed another scheme for morphological shape decomposition. Let B_n, $n = 0, \ldots, N - 1$ be a sequence of structuring elements such that $(0,0) \in B_n$ and $B_n \neq \{(0,0)\}$ $\forall n$. Let ψ_n, $n = 0, \ldots, N-1$ be a sequence of set transformations such that

$$X \ominus A_{n+1} \subseteq \psi_n[X \ominus A_{n+1}] \subseteq X \ominus A_n \tag{3.63}$$

A morphological representation is given by

$$R_n(X) = X \ominus A_n - \psi_n[X \ominus A_{n+1}] \tag{3.64}$$

where

$$A_0 = \{(0,0)\} \tag{3.65}$$

$$A_{n+1} = A_n \oplus B_n \tag{3.66}$$

95

and

$$N = \max_{n}\{n \mid X \ominus A_n \neq \emptyset\} \tag{3.67}$$

The original set X can be reconstructed using

$$X = \bigcup_{n=0}^{N} (R_n(X) \oplus A_n) \tag{3.68}$$

A simple example for set transformation ψ is $\psi(X) = X$. In that case Equation 3.64 becomes

$$R_n(X) = X \ominus A_n - X \ominus A_{n+1} \tag{3.69}$$

The obtained decomposition is called a generalized morphological segmentation. Another choice for transformation ψ is $\psi(X) = (X \oplus B_n) \circ A_n$. This decomposition is called a generalized reduced morphological skeleton. The authors prove that the later decomposition is the "optimal" in the class of invertible morphological decompositions. It is optimal in the sense of minimizing the probability of error in reconstruction in the presence of noise.

3.7 Shape Matching

Shape matching is a procedure for classifying an unknown shape to one of predefined classes. Its name comes from the fact that an unknown shape is matched to a number of known shape candidates (class representatives). The shape matching procedure is a basis for object recognition. This section presents several representative approaches to shape matching.

3.7.1   Minimum Distance Classifier

The most popular way of matching shapes described by their "feature" vectors is using various distance measures. A feature vector (also called a shape descriptor) consisting of n shape features is a vector $x \in R^n$. The problem of shape classification is a problem of assigning an unknown vector x to a one of the classes characterized by shape descriptors x_i where $i = 1, \ldots, m$. The problem is solved by measuring a distance $d_i = \| x - x_i \|$ for $i = 1, \ldots, m$ and selecting the class using the minimal distance criteria. In other words, the unknown shape is classified to a class k such that $d_k \leq d_i \; \forall i = 1, \ldots, m$.

Mathematically speaking, a distance is a function $d : R^n \times R^n \to R$ having the following properties.

1. $d(x, y) \geq 0$ and $d(x, y) = 0 \iff x = y$

2. $d(x, y) = d(y, x)$ (symmetry)

3. $d(x, y) \leq d(x, z) + d(z, y)$ (triangle inequality)

where $x, y, z \in R^n$. There are several distance functions used in practise. Euclidean distance describes the conventional distance that is used in everyday life. It has the advantage of representing conventional distance but is computationally expensive to compute. For that reason, other distance measures like city-block, chessboard, octagonal, or chamfer are often used in practise [28, 29].

3.7.2 Distance Transforms

A distance transform is a transform of a binary image which shows the distance of each pixel to the object in the image. Distance transforms can be computed using sequential or parallel algorithms [29].

The parallel algorithm is given by

$$v^m(i,j) = \min (k,l) \in F(v^{m-1}(i+k,j+l) + c(k,l)) \tag{3.70}$$

where $v^m(i,j)$ is the pixel value at step m and F is the region of support of a mask $c(k,l)$ of the size $M \times M$. The initial image $v^0(i,j)$ if size $N \times N$ is created from the binary image as follows

$$v^0(i,j) = \begin{cases} 0 & \text{for pixels inside the shape} \\ \infty & \text{for pixels outside the shape} \end{cases} \tag{3.71}$$

The mask depends on the distance computed. The procedure is repeated until no pixel value changes in the distance image.

The sequential algorithm parses the image in two steps with two different masks.

1. **for** $i = \frac{M+1}{2}, \ldots, N$ **do**

    (a) **for** $j = \frac{M+1}{2}, \ldots, N$ **do**

        i. $v^m(i,j) = \min_{(k,l) \in F}(v^{m-1}(i+k,j+l) + c(k,l))$

2. **for** $i = N - \frac{M+1}{2}, \ldots, 1$ **do**

fig/distmask.ps

Figure 3.7: A parallel, forward, and backward distance transform masks.

(a) **for** $j = N - \frac{M+1}{2}, \ldots, 1$ **do**

    i. $v^m(i,j) = \min_{(k,l) \in B}(v^{m-1}(i+k, j+l) + c(k,l))$

where F and B are the regions of support of the forward and the backward masks, respectively.

An example of a forward and a backward masks of size 5×5 is shown in Figure 3.7. The center

pixel is marked with 0.

Shih and Mitchell [273] developed a morphological distance transform. They used the gray

scale morphological erosion which is defined in Equation 2.46. It has a very similar structure to

Equation 3.70 and that is why it is appropriate for distance transform computation. Gray scale

structuring elements of the form shown in Figure 3.8 were used for the computation of Euclidean

distance, city-block distance, and chessboard distance transforms [273]. It is very important that

the structuring element size (region of support) is larger then any object in the binary image.

This introduces a difficulty because morphological operations with large structuring elements are

fig/dist-euclid.ps   fig/dist-city.ps   fig/dist-chess.ps

Figure 3.8: Gray scale structuring elements for distance transform.

not efficient and most hardware implementations of morphological operations can deal with fixed
(small) size structuring elements. The authors performed a decomposition of large structuring
elements into smaller ones [315, 269, 271, 272, 276]. The size of small structuring elements was
3×3. The city-block distance is given by $d_4(x, y) = |x_1 - x_2| + |y_1 - y_2|$ and the corresponding
structuring element is given by $k(x, y) = -(|x| + |y|)$. An example city-block distance mask of
the size 5×5 is shown bellow.

$$
k_5 = \begin{bmatrix}
-4 & -3 & -2 & -3 & -4 \\
-3 & -2 & -1 & -2 & -3 \\
-2 & -1 & 0 & -1 & -2 \\
-3 & -2 & -1 & -2 & -3 \\
-4 & -3 & -2 & -3 & -4
\end{bmatrix}
\tag{3.72}
$$

This (and any larger size city-block) mask can be decomposed in the following way.

$$k_5 = k_3 \oplus k_3 \tag{3.73}$$

and in general for any larger size structuring element

$$k_{2n+1} = \overbrace{k_3 \oplus \cdots \oplus k_3}^{n \text{ times}} \tag{3.74}$$

where

$$k_3 = \begin{bmatrix} -2 & -1 & -2 \\ -1 & 0 & -1 \\ -2 & -1 & -2 \end{bmatrix} \tag{3.75}$$

is the 3×3 city-block structuring element. Similarly, the chessboard distance structuring element can be decomposed using the following simple structuring element.

$$k_3 = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 0 & -1 \\ -1 & -1 & -1 \end{bmatrix} \tag{3.76}$$

Further examples of the Chamfer 3-4 and the Chamfer 5-7-11 distance structuring elements can be found in [273]. For approximation of Euclidean distance, Shih and Mitchell propose a more complex method since the Euclidean structuring element cannot be decomposed using the above scheme.

101

3.7.3 String Matching

Many boundary space domain shape description methods produce a description of the shape in the form of the string of symbols denoting elements of the shape boundary. The classification of shapes (patterns) in the pattern space then reduces to comparison of strings. The comparison of strings has found applications in many approaches including syntactic methods [84], polygonal methods [166], and chain code methods. A general overview of string matching techniques can be found in the book by Sankoff and Kruskal [257] or in the review paper by Kruskal [146].

Strings can be classified into attributed and non-attributed. Attributed strings have some additional information associated with each element of the string. Additional information is used to provide more accurate shape description and discrimination. Another classification of strings is in linear and cyclic strings. Cyclic strings solve the problem of starting point selection in description of the shape boundary [166]. Shape matching using linear strings is more sensitive to the choice of starting point.

Non-attributed strings can be compared using the Levenshtein distance. The Levensthein distance between two strings is defined as the minimum number of editing operations (insert, delete, or change) required to transform one string to the other. The weighted Levensthein distance assigns some weight (or cost) to each editing operation. For attributed strings a distance between attribute vectors is combined with the distance between non-attributed part of the string to produce the total distance [83].

A simple measure of string similarity is presented in [97]. In this method, the number of differences in the string elements is counted and compared to the number of matches. The

ratio of the number of matched elements to the ratio of non-matched ones is used as a distance measure. The problem of starting point selection is solved so that the distance is computed for each circularly shifted relative position of strings. A good shape discrimination performance has been demonstrated.

3.7.4 Graph Matching

The model-based approach to recognition is based on the relational model of the object [286, 32, 268, 38]. The relational model describes the relations between the object primitives. The primitives are simple shape components in terms of which a complex shape is described. The relational model can be represented by means of a graph. Hence, a graph matching becomes an important tool for model shape matching.

Wong [304] developed a technique for model matching by subgraph isomorphism. Three-dimensional objects were represented using attributed graphs. An attributed graph is a "conventional" graph with attributes added to its branches and nodes. The input image was first represented using a two-dimensional line drawing. The characteristic points on the drawing were vertices and edges. Vertices are a result of intersection of three surfaces while edges are formed at junction of two surfaces. Several line and junction labeling schemes have been proposed in the literature [40, 34]. Wong further developed a new labeling scheme having thirteen generalized junction types. A model graph is a graph showing the structure of a three-dimensional object. A model graph has the following properties.

1. The total number of nodes is equal to the total number of vertices.

2. The total number of branches is equal to the total number of edges.

3. Exactly three branches are incident with each node.

4. The total number of nodes is even.

5. The total number of branches is $3m/2$ where m is the total number of nodes.

The attributes assigned to each node are the set of allowable junction types and the set of allowable neighbor junction types. A projection graph is a graph showing the structure of the two-dimensional drawing that is a projection of a three-dimensional object. Projection graphs are the subgraph isomorphisms of the three-dimensional model graphs. There are several known algorithms for graph matching presented in the literature [26]. A high complexity of available algorithms is reduced here because junction type constraints were used to eliminate wrong matches. Two nodes matched if they were in the allowable junction types table of each other. The recursive matching procedure worked as follows. Nodes in both graphs were enumerated and each node was matched to other nodes until the match was found. If a mismatch is found the algorithms backtracks and examines other possibilities.

Among many other applications, neural networks [114, 316, 76, 119] have also been used for shape matching. Lin et al. [154] used the Hopfield network [123, 124, 186] to solve the surface and vertex correspondence problems in three-dimensional object recognition systems. The network consisted of neurons organized into a two-dimensional array where rows corresponded to the features of the input shape and columns corresponded to the model shape features. An energy

function was a sum of three terms.

$$E = -\sum_i \sum_k \sum_j \sum_l C_{ikjl} V_{ik} V_{jl} + \sum_i (1 - \sum_k V_{ik})^2 + \sum_k (1 - \sum_i V_{ik})^2 \qquad (3.77)$$

where C_{ikjl} is the coefficient determining the connection strength between the neuron ik and the neuron jl. It is positive if the feature i is similar to the feature k and if the feature j is similar to the feature l, otherwise it is negative. The first term in Equation 3.77 represents the compatibility constraint. The second and third terms are constraints which force that there is only one active output in each row and column. This is required so that each input feature can correspond to a one and only one model feature. The authors specified different coefficients C_{ikjl} for the problems of surface and vertex matching.

Basak er al. [16] have used a similar Hopfield network to perform a matching of shapes represented using the relational model [268, 38]. The relational shape descriptions can be specified in terms of graphs. The authors formulated an energy function which imposes the following constraints. The following assumes $N \times N$ array of neurons where rows and columns correspond to candidate and model shape primitive. Each primitive was a relational shape descriptor.

1. Each candidate primitive must correspond to a one model primitive.

2. The total number of matched primitive is equal to N.

3. Two primitives are matched if they have nearly the same number of attributes and similar attribute values.

4. There exist identical spatial constraints between candidate and model primitives.

105

The proposed technique was successfully applied to the problem of character recognition which has been studied intensively in literature [101].

3.7.5   Medial Axes Matching

Trahanias [288] developed a method for matching of medial axes. In his approach, a medial axis was represented in terms of its components (Equation 3.57). A skeleton function $[SKF(X)](x,y)$ was constructed so that to each pixel belonging to the component n a value $n+1$ was assigned.

$$[SKF(X)](x,y) = \begin{cases} n+1 & \text{for } (x,y) \in S_n(X) \\ \\ 0 & \text{otherwise} \end{cases} \tag{3.78}$$

Pixels outside of skeleton were assigned value zero. The skeleton function was used for weighting of pixel pairs during matching process. The algorithm worked as follows

1. $D \leftarrow 0$

2. **for** each pixel $(x,y) \in S_1$ **do**

    (a) find its nearest pixel $(x',y') \in S_2$

    (b) compute $d = W[(x,y),(x',y')]R[(x,y),(x',y')]$

    (c) $D \leftarrow D + d$

    (d) mark (x',y') as visited

3. **for** each unvisited pixel $(x',y') \in S_2$ **do**

(a) find its nearest pixel $(x, y) \in S_1$

(b) compute $d = W[(x, y), (x', y')]R[(x, y), (x', y')]$

(c) $D \leftarrow D + d$

where $W[(x, y), (x', y')]$ is a weighting function defined by

$$W[(x, y), (x', y')] = |[SKF(X)](x, y) - [SKF(X)](x', y')| + 1 \qquad (3.79)$$

and $R[(x, y), (x', y')]$ is the Euclidean distance between points (x, y) and (x', y'). The resulting value D was the distance between two medial axes which was used for shape matching. The experiments have shown the usability of the method for shape matching.

Chapter 4

Morphological Signature Transform for Shape Description

4.1 Introduction

Mathematical morphology has evolved as a useful tool for various image processing tasks [262, 109, 88]. It is suitable for shape-related processing since morphological operations are directly related to the object shape. Here, a novel approach for shape representation, based on the multiresolution morphological processing, is presented. The proposed method uses multiple structuring elements to obtain the successively eroded versions of input images as a result of multiresolution multiple structuring element morphological operations. The areas of the successively eroded images are then used for shape representation.

The proposed shape representation method is applied to the problem of shape matching which

108

evolves in many object recognition applications. Here, an unknown object must be matched to a set of known objects in order to classify it into one of finite number of possible classes. In practical applications, the unknown object may be corrupted by noise. Results of the proposed shape matching method in the presence of noise are included in this work.

Shapiro et al. [267] used residual approach to shape matching, but the algorithm they used was restricted to one resolution, with a single structuring element shape (disk). In their method the residual image centroid, area, and ratio of minor to major axis of the best fitting ellipse, were used to represent the shape. The method proposed in this work represents the image by its area only, and uses several structuring elements and successive morphological erosions to achieve better accuracy.

One of the basic underlying ideas of this work is the 'divide and conquer' principle applied to shape properties. Shape description methods [99, 284, 312, 217, 73, 309, 200, 136, 137, 8, 296] often attempt to directly extract shape properties from a complex shape. The drawback of such approaches is that the extraction of complex shape properties may not be efficient. The approach taken in this work is to transform the initial (single) complex object to a set of (multiple) objects having simple shape properties. These shape properties from multiple transformed images are then extracted in order to obtain the shape descriptors. The advantage of such an approach is that it extracts simple shape properties from complex objects. Transformed images enable simpler shape descriptors to accurately describe the given initial shape. The proposed approach is called the Morphological Signature Transform and is described below.

4.2 Morphological Signature Transform

In this section, the morphological signature transform (MST) is presented. The *binary* MST and a generalization to the *gray scale* MST are treated in separate subsections.

4.2.1 Binary MST

The shape representation method described in this work is based on the decomposition of a *complex* shape to multiple *simple* signature shapes. Here, the simple shape is described as a shape which can be accurately represented by a shape property that can be computed through simple morphological processing. The idea of this approach is to process decomposed, multiple shapes instead of processing the original shape. The decomposed shapes are called the *signature* shapes because they contain the information about the original shape which is extracted through a property decomposition process. In this section we introduce a general shape property decomposition process called the Morphological Signature Transform (MST).

A definition of the Morphological Signature Transform [157, 159] is given bellow.

For generality, let W be a set which contains shapes to be described. Some common examples for W are: $W = R^2$ (planar shapes) or $W = R^3$ (three-dimensional shapes), where R denotes the set of real numbers. The multiplication of a set by a real number is defined as

Definition 4.1 *Multiplication of a set by a real number.*

$$rX = \bigcup_{x \in X} \{rx\} \tag{4.1}$$

where $r \in R$, and $X \subset W$. $\qquad \square$

Another notation that we use in this work is defined as follows.

Definition 4.2 *A short notation S^n.*

$$S^n = \overbrace{S \oplus S \oplus \cdots \oplus S}^{n \ times} \qquad (4.2)$$

where there are n summands in the Minkowski addition on the right side of the equation. $\qquad \square$

Note that generally $S^n \neq nS$, but in the special case when S is a convex set it holds that $S^n = nS$.

The binary MST is defined as follows.

Definition 4.3 *The MST of shape X with respect to a (not necessarily convex) structuring element S is defined as:*

$$X_S(r, n) = \mathcal{M}(rX, S^n) \qquad (4.3)$$

where $r \in R$, and $n \in N$. $\mathcal{M}$ is a binary morphological operator, i.e. $\mathcal{M} : \mathcal{P} \times \mathcal{P} \to \mathcal{P}$, where $\mathcal{P}$ is the set of all subsets (i.e. the partitive set) of W. $\qquad \square$

In other words $\mathcal{M}$ takes two sets from $\mathcal{P}$ as arguments and produces a third set as result. Some common examples for binary operator $\mathcal{M}$ are morphological erosion, dilation, opening, and closing.

To compute the MST a morphological operator $\mathcal{M}$ must be selected. The morphological erosion has been used in this work. For a given set X, and structuring element S a family of sets $X_S(r, n)$ is generated by varying the parameters r and n. The parameter r is a scale

parameter. The parameter n plays the role in changing the size and the shape of the structuring element S. Note that in the case of convex structuring elements the parameter n determines the structuring element size, while in the case of non-convex structuring elements it changes both size and shape. For fixed scale r the sequence of images $X_S(r, 0), X_S(r, 1), \ldots$ represents the image shape signature with respect to the structuring element S, at some particular image shape scale r. Generated shapes $X_S(r, n)$ are called the signature shapes of X with respect to S.

Signature shapes $X_S(r, n)$ can be used to characterize the original shape X.

4.2.2   Gray Scale MST

In this section, a generalization of the binary MST defined in Section 4.2.1 to a gray scale MST is presented. The generalization is straight forward by substitution of binary morphological operators with their gray scale versions.

In the following text we define several notions which are needed for development of the grayscale MST.

The familiar notion of convexity for real functions is defined as follows.

Definition 4.4 *A real function $f : R^n \longrightarrow R$ is convex if the following condition holds.*

$$\frac{1}{2}[f(a) + f(b)] \leq f(\frac{a + b}{2}) \; \forall a, b \in R^n \tag{4.4}$$

$\square$

The multiplication of a real function by a real number is defined as follows.

112

Definition 4.5 *The multiplication of a real function f by a real number r.*

$$(rf)(x) = \begin{cases} rf(\frac{1}{r}x) & r \neq 0 \\ 0 & r = 0 \end{cases} \tag{4.5}$$

where $r \in R$, and $f : R^n \longrightarrow R$. □

Note that this definition of multiplication includes scaling of the region of support in addition to the scaling of function values. The resulting function is an enlarged (or a reduced) version of the original function. Another short notation that is used here is defined bellow.

Definition 4.6 *A short notation f^n.*

$$f^n = \overbrace{f \oplus f \oplus \cdots \oplus f}^{n \ times} \tag{4.6}$$

where there are n summands in the gray scale Minkowski addition ($\oplus$) on the right side of the equation. □

Note that generally $f^n \neq nf$, but in the special case when f is a convex function it holds that $f^n = nf$.

The MST of a real function (the gray scale MST) is defined as follows.

Definition 4.7 *The MST of a real function f with respect to a (not necessarily convex) real structuring function g is defined as:*

$$MST(r, n) = \mathcal{M}(rf, g^n) \tag{4.7}$$

113

where $f, g : R^n \longrightarrow R$, $r \in R$, $n \in N$, and $\mathcal{M}$ is a gray scale morphological operator. □

In other words, $\mathcal{M}$ takes two real functions as arguments and produces a third real function as result. Some common examples for gray scale morphological operator $\mathcal{M}$ are gray scale morphological erosion, dilation, opening, and closing.

For a given function f and a structuring function g a family of functions $MST(r, n)$ is generated by varying the parameters r and n. The parameter r is a scale parameter. The parameter n plays the role in changing the scale and the shape of the structuring function g. Note that in the case of convex structuring functions the parameter n determines the structuring function size, while in the case of non-convex structuring functions it changes both size and shape. For fixed scale r the sequence of images $MST(r, 0), MST(r, 1), \ldots$ represents the image signature with respect to the structuring function g, at some particular image scale r. Generated images $MST(r, n)$ are called the signature images of f with respect to g.

4.2.3   Example

An example of the morphological signature transform is presented in this section. A binary image and a structuring element are shown in Figure 4.1. The MST of the given binary image with respect to the given structuring element is shown in Figure 4.2. For simplicity, only the MST for the original resolution ($MST(1, n)$) is computed in this example. The shrinking pattern of the MST image sequence depends on the shape of the original image and on the structuring element.

fig/mst/img

fig/mst/kern

Figure 4.1: A binary image (256 × 256), a structuring element (8 × 8).

.

fig/mst/img.1   fig/mst/img.2   fig/mst/img.3   fig/mst/img.4

fig/mst/img.5   fig/mst/img.6   fig/mst/img.7   fig/mst/img.8

fig/mst/img.9   fig/mst/img.10   fig/mst/img.11   fig/mst/img.12

fig/mst/img.13   fig/mst/img.14   fig/mst/img.15   fig/mst/img.16

Figure 4.2: Morphological signature transform sequence.

4.3   MST Shape Description Method

A shape representation method must exhibit three properties:

- translation invariance

- size invariance

- invariance to object rotation

The MST representation method has all three of the above mentioned properties. The method uses Equation 4.3 to obtain the signature shapes from the original shape. The family of image shape signatures obtained in such a way are then used for further shape processing instead of the original image. In the case of shape description the signature images are used for extraction of shape properties.

Another important feature of the MST approach to shape description is that since multiple signatures are available it is possible to use a *simple* shape property to describe the signatures. In such a way, a problem of extracting complex shape properties from a *single* object is replaced with a simpler problem of extracting a simple shape property (e.g. shape area) from derived *multiple* signature shapes.

The input binary image is represented by means of multiresolution pyramid with reduction in resolution by a factor of two across the pyramid levels. The multiresolution approach has been proven to be an efficient one for many image processing algorithms [251]. The image pyramid of the quadtree form is used in this work. This means that the image resolution changes by a factor of two across the pyramid levels. In this case, the input image scale factor r from the

Equation 4.3 has the form $r = 2^{-n}$, where n denotes the pyramid level. The original image pyramid level has the index $n = 0$.

4.3.1 Erosion and Area Based MST Shape Description

A shape property that is used to describe signature shapes in this work is the area. It is both simple and easy to compute - for binary images it is just a count of pixels with values equal to one. The signature shapes are created by means of the MST based on the morphological erosion. In that case, the signature shape Equation 4.3 becomes

$$Z(r, n) = \mathcal{E}(rX, S^n) \tag{4.8}$$

and the area $A(Z(r, n))$ of the signature image $Z(r, n)$ is computed for some $r \in R$ and $n \in N$. In addition, multiple rotated versions of the structuring element S are used in Equation 4.8, as explained below, to ensure additional shape information as well as the rotation invariance of the method.

In other words, the image (at some scale r) is successively morphologically eroded by a structuring element S. The areas of the successively eroded images are used to represent a shape. The same procedure is performed using the input image at different resolutions of the multiresolution pyramid. It is also possible to use multiple structuring elements to generate a families of signature shapes, i.e. to extract different shape properties according to used structuring elements. Following such a procedure a set of numbers is obtained which is used to represent the image.

Note that in the special case of a convex structuring element S we have $S^n = nS$ and therefore

$$Z(r,n) = \mathcal{E}(rX, S^n) = \mathcal{E}(rX, nS) = r\mathcal{E}(X, \frac{n}{r}S) = rZ(1, \frac{n}{r}) \qquad (4.9)$$

If the shape area $A(Z(r,n))$ is used to describe signature shapes $Z(r,n)$ we have from Equation 4.9 that

$$A(Z(r,n)) = A(rZ(1, \frac{n}{r})) = r^2 A(Z(1, \frac{n}{r})) \qquad (4.10)$$

and we see from Equation 4.10 that for convex structuring elements it is sufficient to look at signature images of the form

$$Z(1,p) = \mathcal{E}(X, pS) \qquad (4.11)$$

In other words, with convex structuring elements there is no need of input image scaling, only structuring element scaling is necessary to produce the signature images. This is because the image scaling can be translated into a scaling of the structuring element and as Equation 4.10 shows it is sufficient to scale only a convex structuring element. However, in case of non-convex structuring elements, the scaling of the structuring element only does not provide all the information. The image scaling can, by no means, be translated into a non-convex structuring element scaling and therefore it is necessary to change the image scale, in addition to a structuring element scaling.

The method proposed in this work uses multiple (signature) images to obtain the shape descriptor. The increased dimensionality (the increased amount of the data) allows the use of a simple shape descriptor such as the area to represent an image, as opposed to the approach

taken by Shapiro, et al. in [267] where more complex shape descriptor (a combination of centroid, area, and ratio of minor to major axis length of the best fitting ellipse) has been computed. The additional information, obtained in such a way was found to be sufficient to enable accurate and robust shape representation.

No systematic way or criteria for the choice of a structuring element has been proposed in literature yet. In this work we use non-convex structuring elements which were created to 'sense' both detailed and coarse structure of the image. The rationale behind the use of unconventional (non-convex and non-symmetrical) structuring elements and multiresolution approach is described bellow.

According to Equation 4.10 multiple (successive) erosions computed at single image-resolution are equivalent to single erosions computed at multiple resolutions if *convex* structuring elements (e.g. disk) are used. In that case there is no need for the use of both multiresolution approach and successive erosions because it is sufficient to look at single resolution only. In other words, multiresolution and successive erosions are redundant in case of *convex* structuring elements. In case of *non-convex* structuring elements multiple resolutions and successive erosions provide additional information about the object shape. The method proposed here takes advantage of multiple resolutions and successive erosions simultaneously and it is able to do so due to the use of non-convex structuring elements. Another important feature is that this method uses multiple structuring elements in order to achieve both rotational invariance and better accuracy. Rotational invariance is achieved through rotation of one fixed structuring element. From the morphological processing viewpoint these rotated structuring elements are *different* and therefore

provide additional shape information when used for successive erosions.

The structuring elements used in this work are created "empirically" to detect both detailed and coarse structure of the object. This is achieved through the use of structuring elements which have sharp (elongated) structure on one side and flat (disk-like) structure on the other side (see Figure 4.5). Through rotation of such a structuring element the complete structure of the object can be analyzed to extract more shape information compared to that obtained using a disk structuring element. Since the disk structuring element is rotationally symmetric, rotated versions of it do not provide any additional information.

The inherent property of the basic morphological operations is that they are not invariant to rotation, unless the structuring element is rotationally symmetric by itself. Therefore, a 'direct' use of basic morphological operations together with non-symmetric structuring elements would not satisfy the shape description method property of invariance to rotation listed above. The method that we describe in this work solves this difficulty by using rotated versions of non-symmetric structuring elements, as described below. Such an approach has a double advantage:

- It solves the problem of rotational invariance.

- It provides different structuring elements which enable the extraction of different shape properties (i.e. generation of signature images).

The structuring element must have a 360^0 period of rotational symmetry (i.e. it takes a rotation by 360^0 to put the structuring element in the same position again). Such a requirement comes from the nature of the method that we further describe here. The proposed method could also utilize structuring elements with some other period (e.g. $180^0, 120^0, \ldots$) of rotational

symmetry, although the 360^0 case provides the maximum amount of information, given some fixed rotational resolution.

In the rest of the work a rotational symmetry of 360^0 is assumed. From the initially chosen structuring element we derive a set of N structuring elements by rotating the initial one by the angle α

$$\alpha = i\frac{360}{N}, \ i = 1, \ldots, N-1$$

In such a way, we are able to achieve rotational invariance of the method. The basic steps of the algorithm are as follows:

Algorithm 4.1 *MST algorithm for shape description.*

1. *The input binary image is linearly scaled in both dimensions in such a way that the shape area is equal to some constant value. This step assures size invariance of the method.*

2. *The multiresolution pyramid is created by subsampling by a factor of two from the normalized version of the original image obtained in previous step.*

3. *The rotated versions of the structuring element are created from the original one.*

4. **For each** *multiresolution pyramid level* **do**

    (a) **For each** *rotated version of the structuring element* **do**

        i. *Compute successive erosions of the input image by a given structuring element and at a given pyramid level and store the areas of the successively eroded images in a vector.*

(b) *Circularly shift the vectors (to achieve rotation invariance) so that the largest one comes to zero-th position.*

(c) *Take one or more vectors from the shifted set as a representative for this particular pyramid level.*

5. *Take the vector(s) resulting from each pyramid level and store them in one final shape descriptor vector.*

6. *To achieve more realistic distance measure vectors from different pyramid levels may be weighted, yielding the final morphological shape descriptor vector.*

The vector obtained in the final step of the algorithm is the final result of the algorithm and is used as a morphological shape descriptor. As described above, such a shape descriptor has all the desired invariance properties - invariance to rotation, size, and translation.

4.4 MST Shape Descriptor Algorithm Implementation

In this section a more detailed description of the practical implementation of the morphological shape description algorithm is presented.

In practical implementation, the images, instead of structuring elements, are rotated for better accuracy. Structuring elements are typically of small size (16 by 16 is used in this work). Rotating an image of such a small size by a small angle typically gives an inaccurate result. This restricts the maximum number of rotated versions of the structuring elements, limiting the accuracy of the method. To overcome this practical difficulty, the *images* are rotated instead of

the structuring elements. Input images typically have larger dimensions and the error that is introduced by rotation operation and discrete nature of an image will be reduced.

The more detailed description of the algorithm is shown bellow. Multiple structuring elements simply mean the repetition of this identical procedure for each of the structuring elements.

Algorithm 4.2 *A more detailed MST shape description algorithm*

1. *Choose the initial structuring element, call it S.*

2. *Read the input binary image, call it X. The object of interest in the image has the pixel value of one, while the background has the pixel value of zero.*

3. *Scale the input binary image X by a linear factor f, $f = \sqrt{\frac{F}{KA(X)}}$ where F is the total frame area, K is the scaling security factor, and A(X) is the input binary image area. Call the scaled image Y_0^0, where the subscript denotes the rotated version of the scaled input image while the superscript denotes the pyramid level.*

4. *For i=0 to N-1 do (The index i denotes rotated versions of the scaled input image.)*

   (a) *Generate i-th rotated version of the scaled input image Y_i^0 by rotating Y_0^0 by the angle of $\frac{360^0}{N}i$.*

   (b) *$A_i^0(n) \leftarrow A(\mathcal{E}(Y_i^0, S^n))$, where $n = 1, \ldots, L$ (Compute the areas of successive erosions for 0-th pyramid level. The constant L is the number of successive erosions computed.)*

   (c) *Find the index p of the last non-zero element of the successive area sequence $A_i^0(n)$. (That is, find p such that $A_i^0(p) \neq 0$ while $A_i^0(p+1) = 0$.)*

123

(d) $A_i^0(n) \leftarrow A_i^0(n)w_p(n)$, where $n = 1, \ldots, L$ (Weight the areas sequence with respect to a successive areas index n. Note that the weighting function $w_p(n)$ depends on the parameter p.)

(e) $a_0 \leftarrow [A_1^0 A_2^0 \ldots A_L^0]^T$, where the superscript T denotes the vector transposition. (Form a vector $a_0 \in \mathcal{R}^L$ from L successive areas computed for 0-th pyramid level.)

(f) For $j=1$ to R-1 do (The index j denotes pyramid levels.)

    i. Produce j-th level of the multiresolution pyramid for the i-th rotated image Y_i^j by shrinking (subsampling) the image in the previous pyramid level Y_i^{j-1} by a factor of two.

    ii. $A_i^j(n) \leftarrow A(\mathcal{E}(Y_i^j, S^n))$, where $n = 1, \ldots, L$. (Compute L successive erosions.)

    iii. Find the index p of the last non-zero element of the successive area sequence $A_i^j(n)$.

    iv. $A_i^j(n) \leftarrow A_i^j(n)w_p(n)$, where $n = 1, \ldots, L$ (Weight the areas sequence with respect to a successive areas index n.)

    v. $A_i^j(n) \leftarrow 4^j A_i^j(n)$, where $n = 1, \ldots, L$ (Weight the image areas with respect to a pyramid level index j.)

    vi. $a_j \leftarrow [A_1^j A_2^j \ldots A_L^j]^T$, where the superscript T denotes the vector transposition. (Form a vector $a_j \in \mathcal{R}^L$ from L successive areas computed for j-th pyramid level.)

(g) $b_i^T = [a_0^T a_1^T \ldots a_{R-1}^T]^T$, where the superscript T denotes the vector transposition. (Form a vector $b_i \in \mathcal{R}^{RL}$ from R vectors a_j computed for pyramid levels of the i-th rotated image.

124

5. *Find the largest descriptor vector, call it b_g, with respect to Euclidean vector norm. That is, $|b_g| \geq |b_i|$, where $0 \leq i < N$.*

6. *$c_i \leftarrow b_{(i+g)modN}$, where $0 \leq i < N$. (Form a new vector sequence c_i so that the largest vector, b_g, with respect to Euclidean norm, is in the 0-th position.)*

7. *$d_X \leftarrow [c_0^T c_1^T \ldots c_{N-1}^T]^T$. (Form the final morphological shape descriptor $d_X \in \mathcal{R}^{NRL}$ of the shape X by concatenating N descriptor vectors for each rotated position of the image.)*

The final result of the above algorithm is a vector d_X, which we call the morphological shape descriptor of shape X with respect to a structuring element S. In case of multiple different structuring elements the above algorithm is simply repeated once for each structuring element and the corresponding partial shape descriptor vectors d are concatenated into a one large vector. In such a way it is possible to utilize various structuring elements to extract various shape features and to combine results into one unique morphological shape descriptor.

The scaling security factor K is introduced in Step 3 of the algorithm shown above in order to prevent image scaling out of frame limits. Note that the area of the object contained in image will after scaling be equal to $\frac{F}{K}$, where F is the total frame area.

The multiresolution pyramid (quadtree) is constructed by subsampling as follows. Let $Y^n(i,j)$ denote the image at the pyramid level n, where i and j are integer pixel coordinates. The binary image at the next pyramid level is then constructed by subsampling the image from the previous level as $Y^{n+1}(i,j) = Y^n(2i,2j)$. It is known from literature [251] that the subsampling process may introduce spectrum-aliasing related problems in subsampled images. Some approaches use (possibly weighted) four pixel average to get the resulting pixel in the lower resolution pyramid

level, what can be viewed as a kind of anti-aliasing low-pass filtering. The low-pass nature of binary images used in the experiment presented here did not require any more elaborate anti-aliasing method. Instead, a simple subsampling by a factor of two was successfully used with no aliasing problems.

A special attention is paid to the last non-zero entry in the decreasing sequence of successive image areas. This entry varies randomly and is not used for shape description purposes. It is replaced by zero in each sequence of successive areas.

Two kinds of sequence weighting are performed in the above algorithm. The successive area sequence is weighted for some particular pyramid level and rotation position. This weighting is done to put more emphasize to higher-index elements of the decreasing successive area sequence [55]. The sequence elements decrease to zero approximately as

$$a(n) = a(0)(1 - \frac{n}{p})^2, \ n = 1, \dots, p \tag{4.12}$$

where p is the index of the last non-zero element in the sequence, and $a(0)$ is the area of the original image. The previous formula is derived for the case of circular image and disk-shaped structuring element but can be used as a rule-of-thumb approximation in the cases of other shapes. The formula shows that the first few areas in the sequence will make the highest influence since higher-order elements decrease to zero. To avoid this effect an emphasizing is performed

126

by weighting the sequence by the empirically derived formula

$$w_p(n) = \begin{cases} 1/(1 - \frac{n}{c(p-d)}) & 1 \le n < p \\ 0 & p \le n \le L \end{cases} \qquad (4.13)$$

where c and d are real constants. The values of $c = 1.5$ and $d = 1$ were used in this experiment. The linear nature of the Equation 4.13 cannot completely cancel the quadratic behavior of the Equation 4.12 but does a good trade-off between enlarging elements in the middle of the sequence and keeping higher-order elements value low. Most shape information is contained in the middle of the area sequence due to a nature of the method described here. This is because images are pre-scaled to bring them to a constant area and therefore low-order index elements are relatively similar. On the other side high-order index elements and especially the last non-zero element of the decreasing area sequence contain less shape information and should not have strong influence. The use of weighting expression in Equation 4.13 enhances exactly the middle and most important range of the sequence.

The pyramid level weighting is done with respect to pyramid level. Since area decreases quadratically as resolution decreases by a factor of two this weighting is performed by multiplying all areas in pyramid level j by 4^j, $j = 1, \ldots, R - 1$, where R is the total number of pyramid levels. In such a way, the contribution of each level of the multiresolution pyramid to the shape descriptor vector is roughly equal.

Steps 5 and 6 of the algorithm are necessary to ensure the rotational invariance of the method. The Euclidean vector norm is used to find the largest vector and then the sequence of signatures

127

is circularly shifted so that the largest vectors comes to zeroth position.

One possible optimization of required disk storage and execution time includes generation of the rotated versions of image from zero to ninety degrees only instead of the full 360^0 range. In other words, a set of $\frac{N}{4}$ rotated versions of the scaled input image is generated by rotating Y_0^0 by the angle of $\frac{360^0}{N}i$, $i = 0, ..., \frac{N}{4} - 1$. Missing rotations are then obtained indirectly by rotating structuring element S by the angles of 90, 180, and 270 degrees. A rotation by a multiple of 90 degrees is performed without any error in spite of small resolution of the structuring element. Such an approach results in less disk space requirement and smaller execution time.

4.5 Shape Matching

In this section we present the shape matching procedure based on the minimum distance classifier (see 3.7). Using the MST shape descriptor algorithm we obtain the vector corresponding to each input object. The vector is called the morphological descriptor. The dimension of the morphological descriptor depends on the number of multiresolution pyramid levels, the number of the rotated versions of the original scaled image, and the number of successive erosions computed. Theoretically, the more rotated versions we have, the closer to rotation invariance we get. In practise, however, there exist some maximum number of rotations (maximum resolution) after which additional rotations do not improve accuracy, due to finite precision and errors of the rotation operation.

The study of similarities and differences between shapes now reduces to the study of similarities and differences between morphological descriptor vectors. If we have two shape descriptors

x, and y, of dimension k, then we can take a distance between them as a measure of the similarity of the objects they represent. The smaller the distance, the more similar the objects. The Euclidean distance has been used in this work.

$$d_2(x, y) = (\sum_{i=1}^{k}(x_i - y_i)^2)^{\frac{1}{2}} \tag{4.14}$$

4.6 Experimental Results

A set of eighteen test images was created and used as original images. These images of size 256×256 are shown in Figure 4.3. Another set of eighteen images was created from the original images, by shape distortion. Shape distortion includes changes of size, orientation (rotation), and addition of noise to the shape boundary. Distorted images are shown in Figure 4.4. Two structuring elements of size 16×16 are shown in Figure 4.5.

Four pyramid levels were used ($R = 4$) to represent the input image. The images were rotated into 64 different positions from 0^0 to 360^0 ($N = 64$). The successive area sequence length was 8 elements at the highest resolution pyramid level ($L = 8$) with lengths of 4, 2, and 1, for subsequent lower resolutions. This gave a total descriptor length of 15 elements for one rotation position. The total shape descriptor size for all 64 rotations was then equal to 960 ($64 \times 15 = 960$). Two structuring elements were used resulting in a 1920-dimensional shape descriptor. The multiresolution pyramid resolutions have ranged from 256×256 to 32×32 therefore giving the total frame area $F = 65536$ at the highest resolution pyramid level. To prevent image scaling out of frame bounds a factor $K = 4$ has been used in Step 3 of the

algorithm.

The structuring elements used here were chosen empirically. In this work, non-convex structuring elements were used as opposed to convex structuring elements used in [157]. Due to the multiresolution nature of the MST shape description method the method is particularly suitable for use with non-convex structuring element as discussed in section 4.3.1.

The experiment consisted in shape matching between distorted and original images. In real world applications, it is often necessary to select unknown and noisy object into one of several classes of known objects. The objective is to obtain a shape descriptor which is robust against noise. Euclidean distances between morphological descriptor vectors (normalized to a value of 50,000) are shown in Table 1. The original images are on the horizontal axis while the distorted images are on the vertical axis of the table. The value on the diagonal of the table should be the minimal value in the corresponding row. This means that for some particular distorted image, the distance to its original is smallest compared to distances between that particular distorted image and other images and therefore the image will be correctly classified. The results have shown the accurate shape matching.

fig/pr/part01 fig/pr/part02 fig/pr/part03 fig/pr/part04 fig/pr/part05 fig/pr/part06

S01   S02   S03   S04   S05   S06

fig/pr/part07 fig/pr/part08 fig/pr/part09 fig/pr/part10 fig/pr/part11 fig/pr/part12

S07   S08   S09   S10   S11   S12

fig/pr/part13 fig/pr/part15 fig/pr/part16 fig/pr/part17 fig/pr/part18 fig/pr/part20

S13   S14   S15   S16   S17   S18

Figure 4.3: Original shapes (S01-S18).

fig/pr/part01nfig/pr/part02nfig/pr/part03nfig/pr/part04nfig/pr/part05nfig/pr/part06n

D01   D02   D03   D04   D05   D06

fig/pr/part07nfig/pr/part08nfig/pr/part09nfig/pr/part10nfig/pr/part11nfig/pr/part12n

D07   D08   D09   D10   D11   D12

fig/pr/part13nfig/pr/part15nfig/pr/part16nfig/pr/part17nfig/pr/part18nfig/pr/part20n

D13   D14   D15   D16   D17   D18

Figure 4.4: Distorted shapes (D01-D18).

fig/pr/y16.ps   fig/pr/v16.ps

Y   V

Figure 4.5: Structuring elements used for shape description (Y,V)

	S01	S02	S03	S04	S05	S06	S07	S08	S09
D01	0.16	1.68	3.25	3.43	0.35	1.19	4.93	4.68	0.55
D02	1.65	0.89	1.91	2.02	1.51	1.01	3.64	3.37	1.42
D03	3.16	1.80	0.76	0.92	3.00	2.38	2.25	1.96	2.87
D04	3.36	1.93	0.97	0.30	3.21	2.49	2.28	1.95	3.08
D05	0.52	1.55	3.10	3.27	0.45	1.14	4.79	4.53	0.54
D06	1.81	0.86	1.88	1.87	1.68	0.78	3.65	3.36	1.55
D07	4.94	3.60	2.17	2.16	4.79	4.25	0.10	0.81	4.66
D08	4.33	2.94	1.55	1.39	4.18	3.57	1.28	0.90	4.05
D09	0.74	1.49	2.99	3.14	0.69	1.08	4.67	4.41	0.41
D10	1.40	0.92	2.31	2.44	1.30	1.00	4.00	3.73	1.09
D11	4.46	3.03	1.43	1.45	4.31	3.71	1.27	1.02	4.17
D12	3.16	2.08	2.07	1.80	3.03	2.35	3.24	2.99	2.85
D13	5.40	4.06	2.48	2.65	5.25	4.76	1.21	1.39	5.12
D14	2.68	1.87	2.40	2.22	2.57	1.98	3.79	3.52	2.38
D15	1.82	0.97	1.99	2.04	1.68	1.05	3.72	3.44	1.50
D16	2.77	1.40	1.07	1.05	2.64	1.99	2.66	2.37	2.47
D17	3.42	1.98	0.77	0.58	3.26	2.55	2.21	1.88	3.13
D18	2.58	1.23	1.30	1.31	2.43	1.79	2.91	2.62	2.26

	S10	S11	S12	S13	S14	S15	S16	S17	S18
D01	1.49	4.61	3.02	5.30	2.63	1.32	2.52	3.23	2.37
D02	0.98	3.26	2.08	4.01	1.87	1.20	1.18	1.84	1.19
D03	2.02	1.70	2.08	2.50	2.25	2.42	1.12	0.83	1.33
D04	2.22	1.79	1.82	2.67	2.07	2.56	1.18	0.61	1.39
D05	1.32	4.46	2.90	5.16	2.49	1.12	2.35	3.07	2.19
D06	1.00	3.27	1.78	4.08	1.58	1.12	1.22	1.66	1.07
D07	3.84	1.20	3.37	1.12	3.74	4.24	2.87	2.39	3.10
D08	3.20	1.08	2.73	1.72	3.07	3.59	2.18	1.62	2.43
D09	1.15	4.33	2.73	5.04	2.34	0.91	2.24	2.94	2.02
D10	0.34	3.63	2.20	4.36	1.80	0.70	1.50	2.25	1.34
D11	3.29	0.45	2.87	1.29	3.21	3.71	2.25	1.63	2.48
D12	2.18	2.95	0.56	3.68	1.48	2.38	1.76	1.81	1.66
D13	4.28	1.15	3.92	0.34	4.27	4.71	3.30	2.83	3.52
D14	1.81	3.47	1.28	4.23	0.47	1.90	1.88	2.14	1.60
D15	0.88	3.34	1.87	4.11	1.50	0.86	1.37	1.82	0.98
D16	1.62	2.20	1.77	2.98	1.89	2.04	0.71	0.97	1.11
D17	2.26	1.64	1.97	2.54	2.20	2.60	1.26	0.38	1.43
D18	1.36	2.45	1.66	3.24	1.60	1.69	0.98	1.14	0.53

Table 4.1: Shape matching distance matrix. (S are original and D are distorted shapes.)

Chapter 5

Optimal Shape Matching

A new method for the selection of the optimal structuring element for shape description and matching based on the morphological signature transform (MST) is presented in this paper. For a given class of shapes the optimal structuring element for MST method is selected by means of a genetic algorithm. The optimization criteria is formulated to enable a robust shape matching.

Experiments have been performed on a class of model shapes. The proposed optimal shape description method is applied to the problem of shape matching which evolves in many object recognition applications. Here, an unknown object is matched to a set of known objects in order to classify it into one of finite number of classes. Experimental results are presented and discussed.

5.1 Introduction

Mathematical morphology has been successfully used for various image processing tasks [262, 88, 109]. It is useful for processing of shape because morphological operations directly affect the shape. A new approach for morphological shape representation was presented in [157, 159, 161]. It is based on the morphological signature transform (MST). The method uses a structuring element to compute the MST of objects to be described but does not solve the problem of the selection of the structuring element.

In this paper, we present a new method for selection of optimal structuring elements for use in the MST-based method for shape representation and shape matching. The proposed method uses the optimal structuring element for multiresolution morphological operations. The optimal SE is selected by means of genetic algorithm. The optimality criteria is defined to enable best discrimination of shapes in certain class. In other words, the structuring element provides the best shape matching capability for shapes in given class.

Experiments were performed where the proposed method for optimal structuring elements was applied to a certain class of test shapes. The optimal structuring element was determined by means of genetic algorithm and used for shape description using the MST-based algorithm. The proposed shape description method is applied to the problem of shape matching which evolves in many object recognition applications. Here, an unknown object must be matched to a set of known objects in order to classify it into one of finite number of possible classes. In practical applications, the unknown object may be corrupted by noise.

5.1.1 MST-based shape description

A new approach for MST-based shape representation was presented by Loncaric and Dhawan in [157, 159]. The method is based on multiresolution morphological processing. A brief description of the method is given bellow.

In the preprocessing step original images are resized so that all shapes have the same area. This step insures size invariance of the method. The MST is used to generate signature shapes of the original shape. The signature shapes contain various shape-features of the original shape. Instead of describing the original shape, the signature shapes are described. The underlying idea is that it is easier to describe several simple shapes than one complex shape. The signature shapes are simple in the sense that they contain only features which are extracted by a structuring element at a level of the multiresolution pyramid. Another advantage of the method is that due to multiple signature shapes a better accuracy can be achieved then when describing a single (original) shape. Even better description can be achieved if multiple structuring elements are used. In that case, the MST of the object to be described is computed with respect to several structuring elements yielding even more signature shapes. Multiple structuring elements are created by means of rotation of the initial structuring element. This also enables the rotation invariance of the method.

In the next step of the procedure the obtained signature shapes are described using any conventional shape description method. The simple shape descriptor used in this work is the area of the shape.

Simple shape descriptors are reordered (for rotational invariance) and merged into a large

shape descriptor which is used as the descriptor of the original shape.

The MST-based shape description method has the following properties:

1. translation invariance

2. size invariance

3. rotation invariance

5.2   Introduction to Genetic Algorithms

Genetic algorithm [122, 94, 48, 145] is a powerful tool for solving problems which involve a search through a complex solution space. It is is particularly useful for problems with a large solution space. An exhaustive search of such a complex space is not acceptable in practice because it would take too much time to complete. For such problems, a genetic algorithm can find a near-optimal solution in a much shorter time. In practice, it is often not critical or necessary to find *the* best solution.

Searching a complex space of problem solutions involves a trade-off between two conflicting objectives: *exploring* the search space and *exploiting* the best solutions available [27]. The power of genetic algorithms lies in the search strategy that they use. It is a strategy of creating new solution candidates from the best known solutions using random process of recombination. It is a approach that facilitates both exploration of the space and exploitation of existing best solutions. This approach is opposed to hill-climbing strategies which use only the best currently known solution to find an improved one. Hill-climbing strategies are therefore vulnerable to

being trapped in the region of the search space which may not contain the best solution. The reason for such performance is a weak exploration property of hill-climbing algorithms.

Genetic algorithms have been applied to a wide range of problems including function minimization [51], dynamic system control [95], image registration and pattern recognition [74, 71, 302], and game playing [185, 278].

A genetic algorithm attempts to emulate the mechanism of natural evolution and selection. In nature, species are attempting to adapt to a changing environment. The knowledge accumulated in each generation is stored in chromosomes. By reproduction these chromosomes are modified so that chromosomes of both parents are combined together in order to achieve more competitive child. If the result of reproduction is not effective in the natural environment it will be eliminated by natural selection. Only best specimens will survive. A genetic algorithm utilizes the same idea. In order to solve a problem using a genetic algorithm the following components are necessary [49]:

1. Solutions to the problem must be represented by chromosomes

2. An initial population of solutions must be created

3. An evaluation function for rating solutions in terms of their fitness

4. Genetic operators that alter the composition of children during reproduction

Chromosomes are strings of genes (bits of information) which can have a value of 0 or 1. Problem variables must be represented using chromosomes. This is called coding. A choice of coding is not an easy one. It has been shown [94] that an inappropriate coding choice can

make a search process more difficult. The initial population is usually chosen randomly but it can be also used to test a quality of the solution. The problem solution should not change if initial population is changed. The evaluation function is used to measure a fitness of a solution. There are many possible genetic operators that can be used to alter the composition of children's chromosomes during reproduction. They are typically tailored to suit the particular problem that is being solved. A wrong choice of genetic operators can lead to a failure to determine the solution. Some conventional operators include crossover, inversion and mutation. Crossover interchanges substrings of genes from two chromosomes at randomly selected position in the chromosomes. Inversion inverts a substring of bits in a chromosome. Mutation flips the value of a number of bits at randomly chosen positions in a chromosome.

5.3 A Genetic Algorithm for Optimal Structuring Elements

Genetic algorithm is a useful tool for optimization of functions with large search space which are difficult to search for the optimal solution using exhaustive search methods. In such cases exhaustive search methods would take too much time to complete. Morphological signature transform based algorithm for shape description uses structuring elements to compute the areas of iteratively morphologically eroded images. The problem of selecting the optimal structuring elements is difficult to solve. Consider a binary structuring element of size 8×8. The number of different structuring elements of this size can be as high as 2^{64} what eliminates the possibility

of exhaustive search to determine the optimal structuring element. A genetic algorithm (GA) is an ideal tool in this case.

A method proposed here uses genetic algorithm to search for the optimal (or nearly optimal) solution. The initial generation of structuring elements is created randomly. Each structuring element is represented by a chromosome. The chromosome consists of genes which can be viewed as bits of information. Genes correspond to structuring element pixels. The iteration procedure consists of two main steps:

1. mating of pairs of individuals

2. fitness computation for new individuals

The mating phase consists of three subphases:

1. reproduction

2. crossover

3. mutation

The reproduction phase examines each individual in the current generation and passes it to a next generation with a probability that depends on the fitness (optimization criteria) of individual for some particular purpose. The individual can be either passed (copied) to a next generation or removed. The crossover phase takes pairs of individuals which are represented by their chromosomes and crosses them over in a certain way so that properties of both individuals are combined. In other words certain genes will be exchanged between the two individuals. The mutation phase

randomly changes the value of one or few genes in the chromosome of each individual. The result of these steps is the next generation of individuals. A fitness of each of these new individuals is computed and the next step of iterative procedure is started. In many optimization problems GAs proved to be an effective tools for finding the optimal or nearly-optimal solution. It is especially appropriate in cases where exhaustive search cannot be applied.

However, a choice of representation of variable that is being optimized in a chromosome form suitable for use of GA (also known as coding) is known to be a difficult task [94]. Moreover, it is critical for many optimization problems to choose the best coding in order to find an optimal solution. In other words, failing to do so can cause divergence of the GA from the optimal solution. The coding choice has shown to be a challenging task in this application, too.

5.4   Implementation of Genetic Algorithm

5.4.1   Chromosome representation of the problem

The classical representation of a chromosome as a string of genes has shown not to be appropriate for coding of structuring elements. The most intuitive coding of a binary structuring element is in the string form where rows of the two-dimensional SE are concatenated into a string and each bit corresponds to a gene. The problem with this coding scheme is that the crossover phase makes so significant changes in structuring elements such that even sub-optimal individuals are destroyed in each generation and no optimal solution can be ever reached. In other words, even small changes in chromosome cause significant changes in the shape of the structuring element.

Because of the high sensitivity, this simple coding scheme is not suitable for this optimization problem. In genetic algorithm terms the exploration is dominating over exploitation. Known solutions are not exploited but are destroyed in each generation to facilitate exploration.

The coding scheme that is desirable here must have better exploitation quality but still be able to explore the solution space. It must be such that small modifications in the chromosome caused by crossover make a small change in the shape of structuring element. Another desirable property is that topological properties of the shape do not change a lot in the process of reproduction. For the above mentioned reasons, a special coding method and crossover operator have been used in this work.

The coding scheme uses square chromosomes where genes correspond to pixels of a square structuring element. A new patch-type crossover operation is used in this genetic algorithm. Unlike conventional crossover where strings of genes are swapped, in this scheme rectangular gene patches are swapped. Patch-type crossover operation uses two crossover points instead of a conventional one crossover point operator. It is easier to see this in the case of a conventional one-dimensional chromosome representation which uses a single point at which chromosomes are swapped. A more advanced one-dimensional crossover selects both starting and ending point of a chromosome segment to be swapped. Patch-type crossover used here is a two-dimensional two-point crossover operator. The upper-left and the lower-right patch corners are selected randomly. This assures that less distortion of the structuring element is done in crossover operation. Consequently, the quality of the partial optimization solution will be preserved better across generations.

5.4.2 Variable crossover rate

The chromosomes representing two individuals are crossed-over with some predefined probability. If crossover-probability is higher more pairs will be crossed-over then if probability is low. Basic genetic algorithms use fixed probability of crossover. In order to achieve better exploration of the search space, a variable crossover is used in this work. The crossover probability at each generation is determined as

$$p_{cross} = 1 - h^3 \tag{5.1}$$

where $0 \leq h \leq 1$ represents the normalized entropy of the population. Entropy measures the diversity of the population. Entropy is larger for diverse populations and smaller for populations consisting of more uniform individuals. Low entropy early in the evolution process indicates a premature convergence problem. The value 3 for the exponent in Equation 5.1 is chosen to force higher crossover probability (and avoid premature convergence) even if entropy is relatively high.

The premature convergence problem [27] causes a genetic algorithm not to reach the global optimal solution. The algorithm converges to a local solution. To avoid the premature convergence it is necessary to ensure a good exploration property of the reproduction process.

In this work the entropy of the fitness values is used to measure the diversity of the population. If two population members are different but yield the same fitness value they are considered to be the same with respect to their performance in this application. The fact that they have the same fitness means simply that they are both equally suited for the purpose of shape description. On the other hand if two population members yield different fitness values they certainly have different shapes, too. Therefore the entropy of the population fitness values is appropriate for

143

measuring the diversity of population members with respect to their ability to perform a good

shape description.

Let N be the number of individuals in population. Assume that the fitness f_i, $i = 1, \ldots, N$

of the population members has the value $f_i \in I = [0, f_{max}]$ and divide the interval I into L

subintervals $I_j = [jd, (j + 1)d]$, $j = 0, \ldots, L - 1$ of the length $d = \frac{f_{max}}{L}$. The entropy is then

computed as

$$h = \frac{1}{lnN} \sum_{i=0}^{L-1} p_i ln p_i \tag{5.2}$$

where p_i is the relative frequency of fitness value falling in interval I_i

$$p_i = \frac{n_i}{N} \tag{5.3}$$

where n_i is the number of individuals in population having fitness value $f \in I_i$. The factor lnN in

the denominator of Equation 5.2 is added to normalize entropy values to interval $[0, 1]$. If there is

a lot of diversity in population relative frequencies will have approximately uniform distribution

and the resulting entropy will be high (close to 1) according to Equation 5.2. If fitness values in

population are grouped together around one value the corresponding relative frequency p_i will

be large while other relative frequencies will be low yielding low entropy value (close to 0).

5.4.3   Fitness function

The fitness of each individual is computed for each generation in order to eliminate non-efficient

(non-optimal) individuals in the next generation. In this particular application the fitness criteria

is defined as how suitable a SE is for the purpose of shape matching. The fitness is computed for a certain predefined class of input shapes. This means that the resulting structuring element will be optimal for that particular class of shapes. In practical applications like shape matching there is often a predefined set of model shapes that have to be recognized. In that case a procedure described above can provide the optimal structuring element. The optimal structuring element is the one that will provide best discrimination between the shapes in the class. In other words, a SE is desired which will result in shape descriptors which are as different as possible. The idea behind this definition is that the optimal SE will provide the best noise robustness and shape matching capabilities.

This can be described in a more formal way as follows. Assume that there are M shapes in the class. Let $x_i \in R^p$, $i = 1, \ldots, M$ be shape descriptors for M shapes in the class. Note that shape descriptors are p-dimensional real vectors. There are many possible choices for a fitness function. The properties of the final solution are determined by the nature of the GA fitness function. Several possible choices for the fitness function are presented in the following paragraphs.

The rationale behind the first choice of fitness function is as follows. It is desirable that the shape descriptors are arranged in p-dimensional space in such way that no two vectors are close in order to assure maximal noise robustness and shape discrimination. Therefore the fitness function for structuring element S can be defined as

$$f(S) = \frac{1}{M} \sum_{i=1}^{M} \| x_i - x_m \|_2^2 \tag{5.4}$$

where $\| \cdot \|_2$ denotes the Euclidean vector norm, and x_m denotes the median vector, i.e.

$$x_m = \frac{1}{M} \sum_{i=1}^{M} x_i \tag{5.5}$$

It can be easily shown that this fitness function can also be expressed as

$$f(S) = \frac{1}{M} \sum_{i=1}^{M} \| x_i \|_2^2 - \| x_m \|_2^2 \tag{5.6}$$

Another expression equivalent to the expressions in Equations 5.4 and 5.6 is

$$f(S) = \frac{1}{2M^2} \sum_{i=1}^{M} \sum_{j=1}^{M} \| x_i - x_j \|_2^2 \tag{5.7}$$

These fitness functions will have larger value if shape descriptors are far one from another. If shape descriptors (vectors) are grouped together (in real p-dimensional space) the fitness value will be lower, because it is more difficult to distinguish between shapes. Lower fitness value will result in higher probability for elimination of this particular structuring element in the next generation. The drawback of this type of fitness function is in the fact that it measures the average distance. This means that although average distance is high there may be a pair of shapes with a very small distance what will cause a difficulty in shape matching. To overcome this disadvantage another fitness measure can be defined as

$$f(S) = \min_{i=1}^{M} \min_{j=1, j \neq i}^{M} \| x_i - x_j \|_2^2 \tag{5.8}$$

It is called the minimum distance fitness measure. It looks at the worst case and that is the minimum distance between pairs of shapes. In other words, although the average distance may be high the minimum distance is used as the fitness value. This type of fitness function does a better job of selecting optimal shapes of structuring elements.

5.4.4   Iterative GA procedure

The steps of the procedure for searching for the optimal structuring element are the following. Note that the minimum distance fitness function shown in Equation 5.8 is used here.

Algorithm 5.1 *Optimal MST shape matching.*

1. *Create a class of M model shapes to be described*

2. *Create initial generation of N chromosomes representing structuring elements*

3. **repeat**

   (a) **comment** *loop bellow is for all structuring elements*

   (b) **for** $i = 1, \ldots, N$ **do**

      i. **comment** *loop bellow is for all model shapes*

      ii. **for** $j = 1, \ldots, M$ **do**

         A. **comment** *all model shapes*

         B. *Compute MST-based shape descriptor*

         C. *Let x_j^i be the shape descriptor for shape j using structuring element i*

iii. $x_m^i = \sum_{j=1}^{M} x_j^i$ *(median shape descriptor)*

iv. $f_i = \min_{i=1}^{M} \min_{j=1, j \neq i}^{M} \| x_i - x_j \|_2^2$ *(fitness of structuring element i)*

(c) *Compute population fitness h*

(d) *Compute crossover rate p_{cross}*

(e) *Perform fitness scaling*

(f) *Perform reproduction of chromosomes on the basis of computed scaled fitness*

(g) *Perform crossover of chromosomes using crossover rate p_{cross}*

(h) *Perform mutation of chromosomes*

(i) *Decode chromosomes to create a new generation of structuring elements*

4. **until** *satisfied*

Fitness scaling is performed to prevent domination of one outstanding individual in the generation over the others [94]. It keeps the average fitness value of the generation unchanged but scales maximum fitness value to the value equal to two times the average fitness value.

The block-diagram of the genetic algorithm optimization procedure is also represented in Figure 5.1 for better understanding.

fig/genalg.ps

Figure 5.1: Block-diagram representation of the GA optimization procedure.

5.5 Experimental Results

5.5.1 Optimal Structuring Element

A set of model shapes was created for which the optimal structuring element had to be determined. Model shapes are shown in Figure 5.2. The following convention is used in all figures: black color represents pixel value "zero" and white color represents pixel value "one". A square 8×8 chromosome was used to represent a structuring element of size 8×8. The initial generation of structuring elements is shown in Figure 5.3. Thirty generations of genetic algorithm were computed. For a given shape and structuring element the morphological signature transform based method described in [157, 159, 161] was used to compute shape descriptors. The fitness of each structuring element for shape description of given shapes was computed using the fitness function. The genetic algorithm crossover phase was based on a square patches of genes which are swapped to perform crossover operation. The position within the gene and the size of the patch were determined randomly. The plot of the average fitness in each generation is shown in Figure 5.4. The average fitness of structuring elements in early generations is lower then in newer generations. The fitness value shows the quality of structuring element for shape matching purpose. Therefore the structuring elements in last generations have much better properties then those from early generations. The resulting shape matching performance is better with structuring elements that have higher fitness values. The optimal structuring element for the selected class of shapes is shown in Figure 5.5. The resulting structuring element obtained using genetic algorithm has better shape description and discrimination capabilities then the randomly

fig/pr2/part00.ps fig/pr2/part01.ps fig/pr2/part02.ps fig/pr2/part03.ps fig/pr2/part04.ps

S00   S01   S02   S03   S04

fig/pr2/part05.ps fig/pr2/part06.ps fig/pr2/part07.ps fig/pr2/part08.ps fig/pr2/part09.ps

S05   S06   S07   S08   S09

Figure 5.2: Original shapes (S00-S09).

chosen structuring element. It is the result of a search through the structuring element space given criteria for the optimal shape matching capability.

5.5.2 Shape Matching

The shape matching experiment was performed as follows:

1. MST shape description (based on a structuring element S) of original and noisy shapes.

2. Shape matching of noisy to original shapes.

The experiment was performed using several different structuring elements in order to compare the performance. The optimal structuring element and several non-optimal structuring elements were used in experiment.

151

fig/pr2/se.00.ps fig/pr2/se.01.ps fig/pr2/se.02.ps fig/pr2/se.03.ps fig/pr2/se.04.ps fig/pr2/se.05.ps

K00   K01   K02   K03   K04   K05

fig/pr2/se.06.ps fig/pr2/se.07.ps fig/pr2/se.08.ps fig/pr2/se.09.ps fig/pr2/se.10.ps fig/pr2/se.11.ps

K06   K07   K08   K09   K10   K11

Figure 5.3: The initial generation of structuring elements.

fig/pr2/avgfitness.ps

Figure 5.4: Average fitness vs. generation.

fig/pr2/optimal.ps

Figure 5.5: The optimal structuring element for the given class of model shapes.

To quantitatively measure the shape matching performance a measure called the noise margin ratio is defined as described bellow. The noise margin ratio shows the shape matching performance with respect to a given structuring element and shapes. The shape matching table (matrix) is formed as follows:

$$D = [d_{ij}] \tag{5.9}$$

where $i = 1, \ldots, M$, and $j = 1, \ldots, M$. M denotes the number of shapes. d_{ij} is the distance between the distorted shape i and the original shape j. The noise margin ratio is defined as follows:

$$n_m = \min_{i=1, d_{ii} \neq 0}^{M} \left(\frac{1}{d_{ii}} \min_{j=1, j \neq i}^{M} d_{ij} \right) \tag{5.10}$$

where d_{ij} is the shape matching table entry in the row i and column j. In other words, for each row in the shape distance table the minimum non-diagonal to diagonal entry ratio is computed. The final ratio is equal to the minimal ratio of all rows. The ratio should be as high as possible because it provides a better noise robustness.

The model shapes that were used in the genetic algorithm part of the experiment are now used as original shapes. These images of size 128×128 are shown in Figure 5.2. Another set of shapes was created from the model shapes, by shape distortion. Shape distortion included

fig/pr2/part00n.fig/spr2/part01n.fig/spr2/part02n.fig/spr2/part03n.fig/spr2/part04n.ps

D00   D01   D02   D03   D04

fig/pr2/part05n.fig/spr2/part06n.fig/spr2/part07n.fig/spr2/part08n.fig/spr2/part09n.ps

D05   D06   D07   D08   D09

Figure 5.6: Distorted shapes (D00-D09).

rotation, scale change, and addition of noise to the boundaries of the shapes. Distorted images are shown in Figure 5.6.

Four pyramid levels were used in morphological signature transform algorithm to represent the input image. The images were rotated into 16 different positions from 0^0 to 360^0. The successive area sequence length was 16 elements at the highest resolution pyramid level. This gave a total descriptor length of 30 elements for one rotation position. The total shape descriptor size for all 16 rotations was equal to 480 ($16 \times 30 = 480$). All distances between shape descriptors are normalized to 10000.

The shape matching table obtained using the optimal structuring element is shown in Table 5.1. The optimal structuring element is determined using the genetic algorithm based method explained above. The original image indices are on the horizontal axis while the distorted image indices are on the vertical axis of the table. The value on the diagonal of the table should be the minimal value in the corresponding row. This means that for a particular distorted image,

154

	S00	S01	S02	S03	S04	S05	S06	S07	S08	S09
D00	0.30	0.49	0.59	0.60	0.86	0.57	1.17	0.60	0.56	0.49
D01	0.41	0.35	0.46	0.42	0.82	0.69	1.14	0.55	0.64	0.58
D02	0.60	0.44	0.30	0.51	0.52	1.03	1.34	0.54	1.01	0.61
D03	0.63	0.56	0.54	0.29	0.80	0.82	1.01	0.45	0.77	0.59
D04	0.61	0.46	0.48	0.69	0.30	1.12	1.49	0.60	1.12	0.58
D05	0.76	0.92	0.98	0.83	1.32	0.23	0.91	0.85	0.39	0.79
D06	1.74	1.82	1.85	1.51	2.20	1.24	0.70	1.59	1.21	1.64
D07	0.76	0.78	0.83	0.52	1.11	0.66	0.75	0.44	0.59	0.65
D08	0.95	1.06	1.10	0.91	1.50	0.49	0.92	1.02	0.41	1.03
D09	0.51	0.61	0.70	0.66	0.84	0.69	1.10	0.54	0.71	0.25

Table 5.1: Shape matching distance matrix using the optimal structuring element. (S are original and D are distorted shapes.)

the distance to its original is smallest compared to distances between that particular distorted image and other candidate images and therefore the image will be correctly classified. The results using the optimal structuring element have shown the accurate shape matching. The noise margin ratio for the optimal structuring element is equal to 1.15.

For robust shape matching it is important that the diagonal table entry is as small as possible relative to non-diagonal entries in the same row. This provides a noise margin so that noisy shapes are correctly classified. To understand why is the optimal structuring element really more effective then other elements the shape matching experiment is repeated using a non-optimal structuring element K09, from the initial generation, which is shown in Figure 5.3. The MST shape description is computed and the shape matching between original and noisy shapes is performed. The normalized shape distance is shown in Table 5.2. The noise margin ratio defined by Equation 5.10 is computed. The noise margin ratio for non-optimal element K09 is equal to 1.07. This shows that the optimal structuring element provides better noise robustness then

a non-optimal one. This is not surprising since the optimizing criteria (the genetic algorithm fitness function) was designed to favor noise robustness quality. Another experiment is performed using a non-optimal structuring element K00. The resulting noise margin ratio is 1.06.

It is also possible that a structuring element is not appropriate for shape matching using a given MST shape descriptor size and a given noise level. To illustrate this point the experiment is repeated with the structuring element K01. Shape matching yields the noise margin ratio of 0.77 for a given shape descriptor size and shape noise. This means that an erroneous match took place because in some row of the shape matching table a non-diagonal entry was smaller then the diagonal entry. In general, there are several steps that can be taken in order to overcome the problem of an erroneous shape matching. The shape descriptor size can be increased to obtain a better discrimination property. This can be done by increasing some parameters in MST shape description method such as the number of rotated versions of original structuring element, and/or the number of iterations, and/or the number of image pyramid levels. Another possibility is the use of multiple structuring elements.

	S00	S01	S02	S03	S04	S05	S06	S07	S08	S09
D00	0.18	0.43	0.56	0.72	0.73	0.62	1.23	0.77	0.76	0.59
D01	0.52	0.42	0.48	0.45	0.77	0.65	0.99	0.49	0.53	0.68
D02	0.56	0.45	0.34	0.47	0.49	0.74	1.15	0.50	0.77	0.56
D03	0.67	0.56	0.43	0.25	0.71	0.68	0.83	0.35	0.56	0.63
D04	0.55	0.54	0.47	0.66	0.25	0.87	1.32	0.66	0.97	0.46
D05	0.71	0.81	0.71	0.68	1.01	0.38	0.90	0.74	0.47	0.73
D06	1.70	1.69	1.58	1.27	1.91	1.25	0.59	1.32	1.06	1.56
D07	0.91	0.84	0.71	0.39	1.02	0.62	0.56	0.36	0.42	0.77
D08	0.96	0.95	0.84	0.65	1.22	0.56	0.70	0.71	0.20	0.96
D09	0.56	0.66	0.64	0.63	0.65	0.61	1.09	0.62	0.77	0.18

Table 5.2: Shape matching distance matrix using a non-optimal structuring element. (S are original and D are distorted shapes.)

Chapter 6

Application of Morphological Shape Description to Image Registration

In this chapter, a medical application based on the use of Morphological Signature Transform for shape description is presented.

6.1   Introduction

The registration of MR and PET images has become an important method for study of internal structures of the brain. In particular, MR imaging modality provides the anatomical structure of the brain while PET imaging modality provides information about the metabolic activity of the brain. In order to utilize both sources of information it is necessary to correlate independently obtained MR and PET brain images. Several approaches for registration of MR and PET have been investigated and presented in literature such as techniques based on the use of stereotactic

frames or external markers [20], anatomic landmarks [72], and surface or volume matching [214, 150]. An accurate and practical volume-based registration method called Iterative Principal Axis Registration (IPAR) has been developed by Arata et al. [4, 2, 3]. In this method a principal axis transform of MR and PET binary volumes is used to align the volumes. Here, we present an enhanced method for 3-D brain registration which uses morphological preprocessing and segmentation of MR and PET brain images and an enhanced version of iterative principal axis transformation. The method relies on the mathematical morphology and three-dimensional affine transformations of the brain. A good reference text on geometric modeling of curves, surfaces, and volumes can be found in [195].

6.2 An Overview of Iterative Principal Axis Registration

In this section we give a short overview of the IPAR method [4, 2, 3]. The features of IPAR method are as follows

- Requires no external markers.

- Works even if only partial volumes are available.

- Based on 3-D principal axis transform.

One of the main advantages of this procedure is that it works for partial volumes. In PET imaging modality a view is 10.8 cm high providing only a partial view of the brain. On the

contrary in MR imaging modality there is not a limited view problem. For these reasons, it is often necessary in practise to match a partial brain volume obtained by PET to a full brain volume obtained by MR. This situation makes the registration problem difficult. An outline of IPAR procedure is given by

Algorithm 6.1 *The iterative principal axis registration algorithm.*

1. *Do the segmentation of the total MR and PET brain volume.*

2. *Perform the shape-based interpolation of MR and PET volume.*

3. *Compute initial principal axis parameters for the total MR volume. Let T_{MR} be the matrix representing affine transform which aligns MR volume with the coordinate system origin.*

4. *Compute initial principal axis parameters for the total PET volume. Let T_{PET} be the matrix representing affine transform which aligns PET volume with the coordinate system origin.*

5. **Do**

    (a) *Transform PET volume to the position of MR volume using transformation matrix $T_{MR}^{-1} T_{PET}$.*

    (b) *$T_{MR}^{old} \leftarrow T_{MR}$*

    (c) *Compute principal axis parameters for the partial MR volume corresponding to the PET volume.*

    (d) *Let T_{MR} be the transformation matrix corresponding to the partial MR volume principal axis parameters.*

160

6. **While** $\| T_{MR} - T_{MR}^{old} \| > epsilon$

7. *To obtain the final result transform PET volume to the position of MR volume using transformation matrix $T_{MR}^{-1} T_{PET}$.*

6.3   IPAR and MST Based Registration

A new approach [158, 162] for registration of medical images is described in this section. To motivate the new approach let us consider the rationale behind the multi-modality imaging. A revolutionary step was done when it was realized that it is possible to obtain more information by using multiple imaging modalities at the same time. However, the IPAR and many other registration methods are based on a single source of anatomical references to perform registration. The anatomical reference is either the brain volume, or the brain surface. The approach proposed here is the use of multiple anatomical references. We call such a scheme the *multi-reference* registration for analogy to the multi-modality imaging.

The multiple anatomical references used in this work are the brain volume and the ventricle volume. The IPAR method uses brain volume (shape) to perform the initial registration of the MR and PET brain images. In the method proposed here a ventricular structure of the brain is used as the source for additional information for the subsequent registration phase.

6.3.1   Basic Algorithm

The basic idea for improvement of the original IPAR procedure is as follows.

Algorithm 6.2 *The basic IPAR and MST based registration.*

1. *Perform segmentation of the total MR and PET brain.*

2. *Perform IPAR registration based on the total brain volumes.*

3. *Perform segmentation of the MR and PET brain ventricles.*

4. *Perform MST registration based on the ventricles.*

The "conventional" registration using IPAR procedure is done to perform the initial registration. In the second step, MST based procedure is used to improve the accuracy of registration.

A more detailed outline of the algorithm is presented in Algorithm 6.3.

Algorithm 6.3 *A more detailed brain registration procedure is given as follows.*

1. *Read and unpack files containing MR and PET images from tape.*

2. *Combine MR and PET images to obtain 3-D gray scale MR and PET volume data.*

3. *Perform segmentation of the total MR and PET brain.*

4. *Perform IPAR based registration to obtain the affine transform parameter vector x_{IPAR} which maps the PET volume to the position of the MR volume.*

5. *Create matrix T_{IPAR} corresponding to x_{IPAR}.*

6. *Perform segmentation of the MR brain ventricle.*

7. *Perform segmentation of the PET brain ventricle on the registered PET volume.*

162

8. *Compute MST-based shape descriptor d_{mr} of the MR ventricle.*

9. *Create variation set $V = \cup_{i=1}^{2^n} v_i$ where $v_i \in R_n$ are the vertices of n-dimensional. cube of size g and centered at point x_{IPAR}.*

10. $v_0 \leftarrow x_{IPAR}$

11. *Create matrices T_{MST}^i, $i = 0, \ldots, 2^n$ corresponding to parameter vectors v_i.*

12. *Create fine positional variations of the PET ventricle using matrices T_{MST}^i, $i = 0, \ldots, 2^n$.*

13. *Compute MST-based shape descriptors d_{pet}^i, $i = 0, \ldots, 2^n$ of the PET ventricle variations.*

14. *Perform a matching of PET ventricle variation shapes to MR ventricle shape, i.e. choose $k \in [0, 2^n]$ such that $\| d_{pet}^k - d_{mr} \| \leq \| d_{pet}^i - d_{mr} \|$, $i = 0, \ldots, N$.*

15. T_{MST}^k *is the matrix which maps the PET volume to the position of the MR volume.*

The outline of the procedure is also shown in Figure 6.1.

6.3.2 Multi-Grid Algorithm

Here, an extension of Algorithm 6.3 is made. The extension consists in iterative repetition of the Algorithm 6.3 with different neighborhood sizes. The improved algorithm works as follows. At first a large neighborhood around initial approximation is examined to determine the best solution which is used as starting point of the next iteration. In the next iteration the neighborhood size is cut in half and the whole process is repeated. The algorithm is called the Multi-Grid MST

fig/diagram.ps

Figure 6.1: IPAR/MST registration procedure block diagram.

Registration. The advantage of the multi-grid approach is that the search through the optimization space is faster since the solution is searched for at the coarse resolution first followed by the fine search. To achieve the same resolution using the Algorithm 6.3 much more computation time would have to be used.

Algorithm 6.4 *A multi-grid MST-based registration.*

1. *Read files containing MR and PET images.*

2. *Combine MR and PET images to obtain 3-D gray scale MR and PET volume data.*

3. *Perform segmentation of the total MR and PET brain.*

4. *Perform IPAR based registration to obtain the affine transform parameter vector x_{IPAR} which maps the PET volume to the position of the MR volume.*

5. *Create matrix T_{IPAR} corresponding to x_{IPAR}.*

6. *Perform segmentation of the MR brain ventricle.*

7. *Perform segmentation of the PET brain ventricle on the IPAR registered PET volume.*

8. *Compute MST-based shape descriptor d_{mr} of the MR ventricle.*

9. *$x \leftarrow x_{IPAR}$*

10. *$g \leftarrow$ initial grid size*

11. **for** *$i = 1, \ldots, N$* **do**

(a) *Create parameter variation set* $V = \cup_{i=1}^{2^n} v_i$ *where* $v_i \in R_n$ *are the vertices of n-dimensional cube of size g and centered at point x.*

(b) $v_0 \leftarrow x.$

(c) *Create matrices* T_{MST}^i, $i = 0, \dots, 2^n$ *corresponding to parameter vectors* v_i.

(d) *Create fine positional variations of the PET ventricle using matrices* T_{MST}^i, $i = 0, \dots, 2^n$.

(e) *Compute MST-based shape descriptors* d_{pet}^i, $i = 0, \dots, 2^n$ *of the PET ventricle variations.*

(f) *Perform a matching of PET ventricle variation shapes to MR ventricle shape, i.e. choose* $k \in [0, 2^n]$ *such that* $\| d_{pet}^k - d_{mr} \| \leq \| d_{pet}^i - d_{mr} \|$, $i = 0, \dots, 2^n$.

(g) $x \leftarrow v_k$

(h) $g \leftarrow g/2$

12. T_{MST}^k *is the matrix which maps the PET volume to the position of the MR volume.*

Let us now examine, for Algorithm 6.4 the number of points in the search space that have to be parsed for a given accuracy. Let g be the initial grid size and N be the number of multi-grid levels (see Algorithm 6.4). For simplicity, let the dimension n of the parameter space be equal to one. The minimum grid size is equal to $\epsilon = g2^{-N+1}$ and represents the maximum accuracy.. The size l of the parameter (search) space that can be examined by the multi-grid algorithm is then equal to

$$l = g + g2^{-1} + \cdots + g2^{-N+1} = 2g(1 - 2^{-N}) \tag{6.1}$$

The number of points that have to be parsed is equal to two for each level of the multi-grid algorithm. Therefore we see in this one-dimensional example that to search a subset of the search space of size l with accuracy ϵ the number of search points is equal to

$$N_{s1} = 2N \tag{6.2}$$

In order to demonstrate the computation efficiency of the above multigrid algorithm we compute the required number of points to be parsed with the Algorithm 6.3. We assume that the same search space subset has to be searched and with the same accuracy. The number of points to be parsed is simply given as

$$N_{s2} = l/\epsilon + 1 = 2^N \tag{6.3}$$

Note that the above numbers are obtained for the case of the one-dimensional parameter search space. It can be easily verified that in the case of n-dimensional parameter search space the numbers of points become $(2N)^n$ and 2^{nN} for multi-grid and basic algorithms, respectively. Since a MST shape descriptor has to be computed for each point, this shows the proposed multi-grid algorithm has a huge advantage in computational efficiency. The number of shape descriptors that has to be computed for a given accuracy is shown in Table 6.1. In this example, $n = 3$ (three-dimensions). We can see that the number of search space points to be parsed increases slower for the multi-grid algorithm then for the basic algorithm.

N	N_{s1}	N_{s2}
1	8	8
2	64	64
3	216	512
4	512	4096
5	1000	32768

Table 6.1: Numbers of operations for the multi-grid and the basic algorithm.

6.4   Methods and Procedures

Four steps of the brain registration procedure are described in the following subsections. The preprocessing step is applied to original MR and PET images in order to eliminate varying brightness. An automatic brain segmentation procedure is used to extract the position of the brain. Additional slices are interpolated between the original brain slices. Finally, the iterative principal axis procedure is performed to correlate MR and PET images.

6.4.1   Preprocessing

The original brain images vary in brightness. For example, peripheral slices (bottom or top) have a lower brightness then slices located in the middle of the brain. To achieve uniform brightness a histogram-based K-means clustering algorithm is applied to original input slices. It determines the position of background pixel values, brain tissue pixel values, and brain skull pixel values in the histogram. The determined characteristic values are used to perform a non-linear histogram mapping. The resulting image has three characteristic gray scale ranges shifted to a predefined positions. The preprocessed set has much smaller brightness variation enabling easier

segmentation.

6.4.2   Brain Segmentation

The segmentation step has the role of extracting the brain i.e. creating a binary volume showing
the position of the brain. The difficulty of this step lies in the fact that other parts of the head
anatomy are contained in the scan, too. Therefore an intelligent procedure is necessary in order
to eliminate extraneous parts of the anatomy which must not be included in the brain volume.
The problem is that other anatomical structures are often of similar brightness and/or connected
to the main brain volume. A user interaction is necessary here to direct the decision process of
the program. The segmentation procedure is designed in such a way to minimize necessary user
interaction. The first step is image thresholding. The thresholding level is determined by the
parameters of the previous histogram-modification preprocessing step. The value of the 90 % of
the standardized brain gray tissue pixel value was used here. The resulting thresholded image
typically has 'holes' and 'islands' which are eliminated by region growing processes. Most of
the bays are eliminated using morphological 2-D closing operation [88] with disk-like structuring
element of size 16 pixels in diameter. This size was found to be a good compromise for eliminating
bays but not connecting extraneous parts of the anatomy to the brain volume. After each slice
is processed a final step of 3-D morphological processing is applied to eliminate possible errors
in previous steps. A 3-D morphological closing is applied in order to fill possible gaps in the
volume between two slices which may have occurred as a result of erroneous segmentation. A
subsequent step of 3-D morphological opening is performed for the same reason. The described

169

procedure segments out brain volume correctly in 90 % of MR slices and 97 % of PET slices while user correction is required for erroneously segmented slices. PET segmentation is more accurate because the skull is not present in PET scans.

The first step extracts head mask:

Algorithm 6.5 *Brain segmentation: Part I.*

1. **for each** *image in a scan set* **do**

   (a) *threshold image at level* 50.

   (b) *morphologically erode to break "rings" outside the skull which are sometimes present.*

   (c) *grow a region from outside to fill holes in the brain.*

   (d) *morphologically erode to shrink the extent of the mask.*

The head mask is then used as follows:

Algorithm 6.6 *Brain segmentation: Part II.*

1. **for each** *image in a scan set* **do**

   (a) *threshold image at level* 130.

   (b) *invert image. (This gives all pixels with values less then* 130.

   (c) *multiply by the mask.*

   (d) *morphologically dilate to isolate ICH regions close to the skull.*

   (e) *grow a region from outside to fill holes in the brain.*

(f) morphologically erode to shrink the extent of the brain.

(g) multiply the binary result with the original gray scale image to extract the brain without skull.

6.4.3   Ventricle Segmentation

In this section, procedures for extraction of the ventricle from MR and PET brain images are presented. The extraction of the MR ventricle is an easier problem because the ventricle can be extracted using T2-weighted and proton density images. The extraction is based on the fact that the ventricle appears white in a T2-weighted image while it is dark in a proton density image. The rest of the brain anatomy is approximately of the same brightness. This fact gives rise to a segmentation based on subtraction of a proton density image from a corresponding T2-weighted image.

The outline of the MR ventricle segmentation procedure is as follows.

Algorithm 6.7 *The basic MR ventricle segmentation.*

1. *Subtract proton density image from T2-weighted image.*

2. *Threshold the result.*

3. *Perform morphological closing to fill holes and connect ventricle structure.*

4. *Correct the result manually, if necessary.*

On the other hand, a segmentation of PET ventricle is more difficult because the ventricle is not easily recognized in a PET image. For this reason, the segmentation of the PET ventricle is

performed after the IPAR registration is finished since at that time the approximate position of PET relative to MR is known.

Algorithm 6.8 *The basic PET ventricle segmentation.*

1. *Use MR ventricle to mask gray scale voxels within the PET ventricle.*

2. *Compute the histogram of the masked PET volume to determine the optimal threshold for PET ventricle segmentation.*

3. *Threshold the PET volume.*

4. *Perform morphological closing to fill holes and connect ventricle structure.*

5. *Correct the result manually, if necessary.*

6.4.4 Interpolation

A shape-based interpolation [239] is used to create a full volume with slice thickness equal to the slice pixel size. Typically, the pixel size is of the order of 0.5 mm while the original slice thickness is of the order of 1 cm. Shape-distance maps are computed and cubic interpolation is used to add new slices at required positions. Interpolated gray scale images are then thresholded to get the full MR and PET binary volumes. The procedure steps are summarized as follows.

Algorithm 6.9 *Shape-based interpolation.*

1. *Input a binary image.*

2. Compute the Euclidean distance transform of the input image.

3. Interpolate additional slices between the distance transform images of the original slices.

4. Threshold the result at zero.

The distance transform is computed so that each pixel in output image has the value equal to the distance of that pixel to the closest border. In addition, pixels "inside" the shape are assigned positive values while pixels "outside the shape are assigned negative values. This is followed by a cubic interpolation which adds an additional slice at any position between middle two of any four consecutive slices. Four slices are required to uniquely determine the third-order polynomial. The result is thresholded at zero level to obtain the resulting binary images.

6.4.5 IPAR

The iterative principal axis algorithm is used to correlate MR brain volume and partial PET brain volume. In this work, an improved IPAR algorithm was used. The improved version uses the original idea [4, 2] but has an improved iteration algorithm. To eliminate of problem of possible oscillating and not converging to a solution a weighting of partial solutions is introduced which insures convergence towards the solution. Assume that vector S_i denotes the PAR parameters of the MR volume in the iteration i of the algorithm. The original algorithm used this vector to perform translations and rotations in order to align the volume. The improved iteration algorithm takes the previous value of the principal axis parameters in the account in order to prevent possible divergence from the solution. The actual PAR parameters at iteration i are

computed as

$$S_i^{'} = \alpha S_i + (1 - \alpha)S_{i-1} \tag{6.4}$$

where $\alpha \in (0, 1)$. The result is a more uniform and monotone convergence to the solution.

6.4.6   MST-based Registration

The MST-based multi-modality registration is based on the matching of a MR ventricle shape with a PET ventricle shape. The accuracy of the method depends on the accuracy of the ventricle segmentation. The ventricle segmentation is performed as explained in Section 6.4.3. The multi-grid algorithm based on the recursive search of the shape descriptor space is used as described in Section 6.3.

6.5   Physical Brain Phantom

To perform an objective evaluation of a registration algorithm it is necessary to use a brain phantom. The evaluation using the real patient data is difficult because of the lack of clear landmarks that can be located in both MR and PET brain images. A brain phantom can be designed to contain landmarks which can be clearly detected in both MR and PET scans. In this section a brain phantom used for experimental evaluation of IPAR and IPAR/MST procedures is described.

The phantom utilized here is actually a combination of two phantoms. The first phantom is a phantom with anatomical details (gray, white matter and bone) shown in Figure 6.2. It

fig/phan-graywhite.ps

Figure 6.2: Brain phantom with anatomical details.

is a much simplified model of the human brain structure. The second phantom is a phantom developed by Derenzo et al. [52]. It consists of tubes of variable diameters and spacings as shown in Figure 6.3. The tubes are used as landmarks because their position can be easily determined in both MR and PET images. The tube diameters (spacings) in millimeters are as follows: 6.25 (25), 5 (20), 4 (16), 3.5 (14), 3 (12), and 2.5 (10). A cross section detail of Derenzo phantom (as seen by MR) is shown in Figure 6.4. The overall diameter of the phantom is 200 mm.

The anatomical phantom is required for IPAR and MST algorithms while Derenzo phantom is needed for registration error estimation. These requirements are met in a combination of the two phantoms mentioned above. The combined phantom is formed by rigidly joining two phantoms as shown in Figure 6.5. The resulting phantom is scanned using MR and PET imaging modality as a single phantom.

fig/phan-derenzo.ps

Figure 6.3: Derenzo brain phantom.

fig/phan-derenzo-slice.ps

Figure 6.4: A MR slice of Derenzo brain phantom.

fig/phan-comb.ps

Figure 6.5: A combined brain phantom.

6.6 Experimental Results

A validation of the proposed registration method is conducted in two different ways using both a physical phantom and a real patient data. An objective evaluation of the method is performed through the use of a combined physical brain phantom described in Section 6.5. It provides a *quantitative* comparison between the IPAR and the IPAR/MST registration procedures. A subjective evaluation of the registration is performed using the real patient brain images. The evaluation is performed through a visual comparison of the IPAR and the IPAR/MST registration results by a human observer.

The experiments are conducted as follows. Multi-modality brain images were first registered using the IPAR technique based on the MR and PET brain volume. In the second step, the

IPAR/MST registration method is used as described in Section 6.3.

The six parameters determining the position of the brain in the space are x, y, and z coordinates and α, β, and γ angles of principal axis. The multigrid algorithm described in Section 6.3.2 is used to make variations of the three angle parameters and search recursively for the best match. Only three parameters are variated in order to reduce the number of shape variations to be examined. Using three parameters, there are $2^3 = 8$ variations at each step of the multi-grid algorithm. With six degrees of freedom the number of combinations grows to $2^6 = 64$ at each step of the algorithm causing much longer execution time. In support of this argument, there is a much higher uncertainty associated with the principal axis angle estimation then with the centroid estimation. The compact, ball-shaped brain is the main reason for this claim. Ball-shaped forms are difficult to characterize using the principal axes because of the symmetric nature of the form. For this reason, the centroid parameters estimated by the IPAR method are kept constant and only principal axis angles are variated.

The physical brain phantom and real data experiments are presented in the next two sections.

6.6.1   Brain Phantom Experiment

The brain phantom experiment is performed as follows. The combined physical phantom described in Section 6.5 is scanned using MR and PET scanners and two sets of phantom images are obtained. The anatomical section of the phantom is used to perform IPAR and MST registrations, while Derenzo part of the phantom is used for error measurement. In the first step the IPAR algorithm is performed. The phantom brain is segmented in MR and PET images to

create the binary volume data for IPAR algorithm. The IPAR registration is performed and the PET data transformed (rotated and translated) to align it with the MR data. In the second step the MST-based registration is performed as follows. The internal anatomical phantom structure is segmented in MR and PET images and used as the brain ventricle in the real brain. The MST registration is then performed and another set of registered PET images is created by rotation and translation. The I-shaped three-dimensional structuring element is used for MST shape description.

The original phantom MR images are shown in Figure 6.6 and the original phantom PET images are shown in Figure 6.7. The registered images are shown with MR brain contour superimposed on the PET image data. The IPAR registered images are shown in Figure 6.8 while the IPAR/MST registered images are shown in Figure 6.9.

The MR pixel size is equal to 0.9375 mm and the MR slice thickness is equal to 5 mm. The PET pixel size and slice thickness is equal to 1.56 mm and 6.75 mm, respectively. MR and PET images are interpolated in all three dimensions so that the cubic voxels of size equal to 0.9375 mm are created.

The resulting images reflect the accurate registration. However, a quantitative analysis presented bellow is required for an objective evaluation of results.

6.6.2   Brain Phantom Registration Error Analysis

The mismatch between Derenzo tube positions in registered MR and PET images is used to measure the error. The tubes with the largest diameter are used since smaller diameter tubes

fig/phantom/orig.mr/slice.01 fig/phantom/orig.mr/slice.02 fig/phantom/orig.mr/slice.03 fig/phantom/orig.mr/slice.04 fig/phantom/orig.mr/slice.05

fig/phantom/orig.mr/slice.06 fig/phantom/orig.mr/slice.07 fig/phantom/orig.mr/slice.08 fig/phantom/orig.mr/slice.09 fig/phantom/orig.mr/slice.10

fig/phantom/orig.mr/slice.11 fig/phantom/orig.mr/slice.12 fig/phantom/orig.mr/slice.13 fig/phantom/orig.mr/slice.14 fig/phantom/orig.mr/slice.15

fig/phantom/orig.mr/slice.16 fig/phantom/orig.mr/slice.17 fig/phantom/orig.mr/slice.18 fig/phantom/orig.mr/slice.19 fig/phantom/orig.mr/slice.20

Figure 6.6: Original phantom MR images.

fig/phantom/orig.pet/slice.01 fig/phantom/orig.pet/slice.02 fig/phantom/orig.pet/slice.03 fig/phantom/orig.pet/slice.04 fig/phantom/orig.pet/slice.05

fig/phantom/orig.pet/slice.06 fig/phantom/orig.pet/slice.07 fig/phantom/orig.pet/slice.08 fig/phantom/orig.pet/slice.09 fig/phantom/orig.pet/slice.10

fig/phantom/orig.pet/slice.11 fig/phantom/orig.pet/slice.12 fig/phantom/orig.pet/slice.13 fig/phantom/orig.pet/slice.14 fig/phantom/orig.pet/slice.15

Figure 6.7: Original phantom PET images.

180

fig/phantom/nfig/phantom/slifeg0/2hantom/slifeg0/3hantom/slifeg0/4hantom/slina.05ned.pet/slice.06

fig/phantom/nfig/phantom/slifeg0/7hantom/slifeg0/8hantom/slifeg0/9hantom/slina.10ned.pet/slice.11

fig/phantom/nfig/phantom/slifeg12hantom/slifeg13hantom/slifeg14hantom/slina.15ned.pet/slice.16

fig/phantom/nfig/phantom/slifeg17hantom/slifeg18hantom/slina.19ned.pet/slice.20

Figure 6.8: IPAR registered phantom PET images. MR brain contours are superimposed on PET brain image.

fig/phantom/veigt/phantom/vet/gl/phantom/vet/gl/phantom/vet/gl/phantom/vet/slice.05ned.pet/slice.06

fig/phantom/veigt/phantom/vet/gl/phantom/vet/gl/phantom/vet/gl/phantom/vet/slice.10ned.pet/slice.11

fig/phantom/veigt/phantom/vet/gl/phantom/vet/gl/phantom/vet/gl/phantom/vet/slice.15ned.pet/slice.16

fig/phantom/veigt/phantom/vet/gl/phantom/vet/gl/phantom/vet/slice.19ned.pet/slice.20

Figure 6.9: IPAR/MST registered phantom PET images. MR brain contours are superimposed on PET brain image.

fig/phanregerr/mrslice.13   fig/phanregerr/petslice.13

Figure 6.10: Registered Derenzo phantom MR and PET slices (IPAR algorithm).

are not registered by PET due to a limited PET resolution. To determine the position of the tubes registered gray scale images are thresholded so that the resulting binary images have the same area. The *exclusive or* operation is performed between binary images to reveal mismatched areas. A larger registration error will generate a larger mismatched area. The area of the mismatched regions is computed to obtain a quantitative measure of the registration accuracy.

The following example is used to better illustrate the steps of the error estimation procedure. An example of registered MR and PET images is shown in Figure 6.10. The registered images are thresholded and the largest diameter tubes are used for error estimation as shown in Figure 6.11. To determine the amount of mismatch the exclusive or operation is performed between the thresholded images and the result is shown in Figure 6.12. It is obvious that the mismatched region area is smaller for the IPAR/MST algorithm then for the IPAR algorithm. This means that the IPAR/MST algorithms improves the accuracy of the IPAR algorithm.

fig/phanregerr/mrslice.13.thr

fig/phanregerr/petslice.13.thr

Figure 6.11: Thresholded registered Derenzo phantom MR and PET slices (IPAR algorithm).

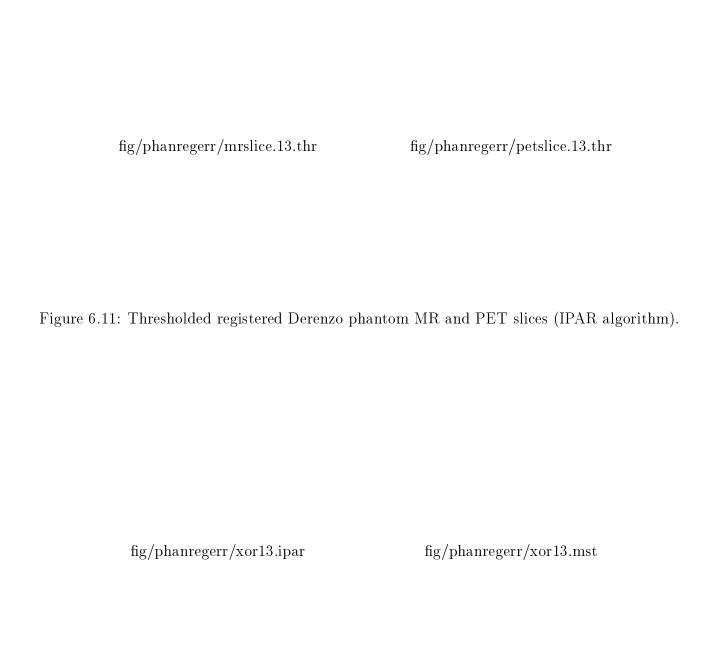fig/phanregerr/xor13.ipar

fig/phanregerr/xor13.mst

Figure 6.12: Mismatch between MR and PET registered Derenzo phantom slices for IPAR and IPAR/MST algorithms.

The Derenzo slices with odd indices are used to perform error analysis. The odd slices are selected to reduce the number of slices required for computation.

Let t_i^{MR} and t_i^{PET} be the binary images obtained by thresholding the i-th registered MR and PET images, respectively. Note that $A(t_i^{MR}) = A(t_i^{PET})$ where $A(.)$ denotes the area operator. The relative error is obtained by dividing the mismatched area by the double tube area as follows.

$$\epsilon_i = \frac{A(t_i^{PET} \underline{\vee} t_i^{MR})}{2A(t_i^{MR})} \tag{6.5}$$

where i denotes the image index and $\underline{\vee}$ operator denotes the *exclusive or* operation. Note that $0 \leq \epsilon_i \leq 1$. ϵ_i is also called the *relative mismatched area*. The average relative error for a set of N slices is determined as follows.

$$\epsilon = \frac{1}{N} \sum_i \epsilon_i \tag{6.6}$$

Relative errors for odd Derenzo slices for IPAR and IPAR/MST registration are shown in Table 6.2. It is evident that the IPAR/MST algorithm reduces registration error, i.e. improves registration accuracy. The difference in accuracy is also evident in Figure 6.12 where it is obvious that the mismatched region area is larger for IPAR then for IPAR/MST algorithm.

A different measure of registration error is the distance between the centroids of the corresponding Derenzo disks [3]. It can be shown that the relative mismatched area and the centroid distance measures are not independent. Moreover, the two measures are related by an expression shown in Equation 6.7. The expression is easily derived by computing the area of intersection of two disks and subtracting it from the total area. Consider two overlapping disks of radius r

Slice	IPAR	IPAR/MST
11	0.68	0.46
13	0.90	0.64
15	0.78	0.56
17	0.78	0.60
19	0.76	0.60
avg	0.78	0.57

Table 6.2: The relative errors for Derenzo slices (11,13,15,17,19) and the average error for IPAR and IPAR/MST algorithms.

fig/phanregerr/twocirc.ps

Figure 6.13: The relative mismatched area.

as shown in Figure 6.13. The distance between their centroids (centers) is equal to $2h$, where $h \leq r$. The relative mismatched area is equal to the ratio of the mismatched area to the double disk area as shown in Equation 6.5.

$$\epsilon = 1 - \frac{2}{\pi}(\arccos x - x\sqrt{1 - x^2}) \tag{6.7}$$

186

fig/phanregerr/errplot.ps

Figure 6.14: The relative mismatched area vs. relative centroid distance.

where

$$x = \frac{h}{r} \tag{6.8}$$

is the *relative centroid distance*. If $h = 0$ no error is present and $\epsilon = 0$. On the other side, if $h = r$ the relative error $\epsilon = 1$ results. Note that the relative mismatched area is only defined for the centroid distance range $0 \leq 2h \leq 2r$. In other words, it can only be used when the centroid distance is within the range $[0, 2r]$. This is not a problem since the radius r can be selected large enough to satisfy the above requirement. The plot of ϵ as a function of x is shown in Figure 6.14. It is evident from the plot that the relative mismatched area changes "almost" linearly with the relative centroid distance. Due to a nonlinear nature of Equation 6.7 it is impossible to explicitly

express the relative centroid distance in terms of the relative mismatched area.

Having developed the relation between the two error measures it is possible to estimate indirectly the centroid distance from the mismatched area. For example, the average relative mismatched area for IPAR method shown in Table 6.2 is equal to 0.78. Reading the graph in Figure 6.14 we estimate the corresponding relative centroid distance to approximately 0.66. It follows that the centroid distance is equal to $2h = 0.662r$. The average disk radius in the phantom data experiment is $r = 3.95$ pixels what leads to the centroid distance of $2h = 5.21$ pixels. Since the pixel size is equal to 0.9375 mm, the centroid distance is equal to $5.21 \times 0.9375 = 4.88$ mm. The centroid distance error measure for IPAR/MST method is computed similarly. The average mismatched area is equal to 0.57 and the graph in Figure 6.14 gives the corresponding relative centroid distance of 0.47. The resulting centroid distance is equal to $2h = 3.71$ pixels. The pixel size of 0.9375 mm means that the centroid distance is equal to 3.47 mm.

6.6.3   Real Patient Data Experiment

The experimental results obtained using real patient data are presented in this section. Here, a subjective evaluation by a human viewer is performed. The original MR brain data is shown in Figure 6.15 and the original PET data is shown in Figure 6.16. The registration result is represented by superimposing the MR brain contour on the PET image data. Figure 6.17 shows the IPAR registered images. The IPAR/MST registered images are shown in Figure 6.18. Table 6.3 shows the distances between MST shape descriptor for MR ventricle and MST shape descriptors for rotationally variated PET ventricles. The PET variation corresponding to the

fig/real/orig.mr/slice.01 fig/real/orig.mr/slice.02 fig/real/orig.mr/slice.03 fig/real/orig.mr/slice.04 fig/real/orig.mr/slice.05

fig/real/orig.mr/slice.06 fig/real/orig.mr/slice.07 fig/real/orig.mr/slice.08 fig/real/orig.mr/slice.09 fig/real/orig.mr/slice.10

fig/real/orig.mr/slice.11 fig/real/orig.mr/slice.12 fig/real/orig.mr/slice.13 fig/real/orig.mr/slice.14 fig/real/orig.mr/slice.15

fig/real/orig.mr/slice.16 fig/real/orig.mr/slice.17

Figure 6.15: Original MR images.

P0	P1	P2	P3	P4	P5	P6	P7	P8
7.67	7.45	7.47	7.49	7.47	7.56	7.51	7.47	7.47

Table 6.3: PET ventricles to MR ventricle shape distance using MST shape description.

smallest distance is selected as the most accurate one.

Visual inspection of the resulting registered images shows an improvement in registration quality of IPAR/MST algorithm over IPAR algorithm. However, this evaluation is subjective while an objective method is presented in the previous section.

fig/real/orig.pet/slice.00 fig/real/orig.pet/slice.01 fig/real/orig.pet/slice.02 fig/real/orig.pet/slice.03 fig/real/orig.pet/slice.04 fig/real/orig.pet/slice.05

fig/real/orig.pet/slice.06 fig/real/orig.pet/slice.07 fig/real/orig.pet/slice.08 fig/real/orig.pet/slice.09 fig/real/orig.pet/slice.10

fig/real/orig.pet/slice.11 fig/real/orig.pet/slice.12 fig/real/orig.pet/slice.13 fig/real/orig.pet/slice.14 fig/real/orig.pet/slice.15

Figure 6.16: Original PET images.

fig/real/matched.pet/slice.01 fig/real/matched.pet/slice.02 fig/real/matched.pet/slice.03 fig/real/matched.pet/slice.04 fig/real/matched.pet/slice.05 fig/real/matched.pet/slice.06

fig/real/matched.pet/slice.07 fig/real/matched.pet/slice.08 fig/real/matched.pet/slice.09 fig/real/matched.pet/slice.10 fig/real/matched.pet/slice.11

fig/real/matched.pet/slice.12 fig/real/matched.pet/slice.13 fig/real/matched.pet/slice.14 fig/real/matched.pet/slice.15 fig/real/matched.pet/slice.16

Figure 6.17: IPAR registered PET images. MR brain contours are superimposed on PET brain image.

fig/real/ventmatched.pet/slice.01fig/real/ventmatched.pet/slice.02fig/real/ventmatched.pet/slice.03fig/real/ventmatched.pet/slice.04fig/real/ventmatched.pet/slice.05ed.pet/slice.06

fig/real/ventmatched.pet/slice.06fig/real/ventmatched.pet/slice.07fig/real/ventmatched.pet/slice.08fig/real/ventmatched.pet/slice.09fig/real/ventmatched.pet/slice.10ed.pet/slice.11

fig/real/ventmatched.pet/slice.11fig/real/ventmatched.pet/slice.12fig/real/ventmatched.pet/slice.13fig/real/ventmatched.pet/slice.14fig/real/ventmatched.pet/slice.15ed.pet/slice.16

Figure 6.18: IPAR/MST registered PET images. MR brain contours are superimposed on PET brain image.

Chapter 7

Optimal Morphological Shape Description for Brain Image Registration

A new method for selection of the optimal 2-D structuring element for MST shape description and matching is presented in Chapter 5. A novel approach to brain image registration is proposed in Chapter 6. In this chapter, we describe an improved version of the algorithm for MST brain image registration. The algorithm is improved in the sense that it uses the *optimal* structuring element. The method is called IPAR/OMST registration, where the letter O stands for optimal.

In this section, a genetic algorithm procedure for the selection of a near-optimal 3-D structuring element is described. The principle is similar to one described in Chapter 5.

7.1 Introduction

A new genetic algorithm procedure has been used to determine a near-optimal three-dimensional structuring element for OMST-based shape description method. The OMST-based method is used to improve the accuracy of the IPAR registration as described in Section 6.3. The most significant difference between the genetic algorithm presented here and the one presented in Chapter 5 is the number of dimensions.

In the brain registration application, a three-dimensional structuring element is required for OMST-based shape description of a three-dimensional (3-D) brain ventricular structure. The 3-D structuring element is used to describe fine 3-D positional variations of the PET ventricle as explained in Section 6.3. Fine positional variations represent just minor changes in the shape of the object (ventricle). For this reason it is necessary to select a structuring element which is able to distinguish between several positions of the object. (The objects produced by fine positional variations are very similar with respect to morphological erosion.) The optimal structuring element is defined as the one which provides the best discrimination between the fine 3-D positional variations of the ventricle. It is important to emphasize that the MST shape description algorithm used here is designed *not* to use rotated structuring elements (see Section 4.3.1). This is necessary because in this (registration) application a sensitivity to rotational changes is desired as opposed to a shape description application (Chapter 4) where invariance to rotation is desired.

A genetic algorithm is the method of choice for the problem of selecting the 3-D optimal structuring element. In the 3-D case there are even more arguments for the use of genetic

algorithm then in the 2-D case. The following example shows why is this so. With 2-D structuring elements of size 8×8 there are $2^{8 \times 8} = 2^{64}$ possible combinations (different structuring elements). In case of 3-D structuring elements this number grows to $2^{8 \times 8 \times 8} = 2^{512}$. This also means that the 3-D search space is much larger the the 2-D search space. However, a genetic algorithm is especially suited for optimization problems in large search spaces where an exhaustive search is not feasible. In that case, the genetic algorithm provides a near-optimal solution which is often satisfactory in practical applications.

7.2   Implementation of Genetic Algorithm

The basic structure of the 3-D genetic algorithm is similar to that of the 2-D genetic algorithm described in Chapter 5. However, there are several changes due to a difference in dimensions (two vs. three). The cubic chromosome is used in 3-D which is analogous to the square chromosome in the 2-D application. Reproduction, crossover, and mutation algorithms from Chapter 5 are extended to three dimensions. The same kind of a variable crossover rate controlled by the population entropy is used in 3-D algorithm. A block-diagram of the 3-D genetic algorithm optimization procedure is represented in Figure 7.1. The principal difference between 2-D and 3-D algorithms as compared to Figure 5.1 is that the input shapes in 3-D case are obtained from a single shape by means of fine positional variations of the original 3-D shape. This is because we are looking for the 3-D structuring element which best discriminates between fine positional variations of the single (original) 3-D shape.

A primary positional variation used here is 3-D rotation. To understand why is this so it

194

fig/genalg3d.ps

Figure 7.1: Block-diagram representation of the 3-D GA optimization procedure.

is important to look at the principal axis registration algorithm and its sensitivity to noise. Six parameters of the volume obtained by principal axis transform are three center of mass coordinates and three angles of principal axes with respect to coordinate axes. The center of mass coordinates are robust to noise. The principal axes angles are more sensitive to noise since brain volume is approximately ball-shaped. For round shapes principal axes angles are more sensitive to changes in shape due to non-ideal segmentation and/or noise. For this reason only the three principal axes angles are variated to achieve better registration.

The resulting structuring element is called the *optimal* structuring element. The second principal difference compared to the shape description application described in Chapter 5, as explained in the previous section, is that in this application the MST shape description algorithm

195

is made sensitive to rotational change. This is achieved by the use of the single structuring element for shape description (no rotated versions).

7.3 Experimental Results

A validation of the proposed registration method is conducted in the same way as in Chapter 6. Both a physical phantom and a real patient data are used. The combined physical brain phantom is described in Section 6.5. It provides a *quantitative* comparison between the IPAR and the IPAR/OMST registration procedures. A subjective evaluation of the registration is performed using the real patient brain images. The experiments are conducted using the same data and in the same way as in Chapter 6.

7.3.1 Brain Phantom Experiment

The brain phantom experiment is performed as explained in Section 6.6.1 and the same phantom MR and PET data is used. The only difference is that the optimal structuring element is used which is determined by means of a genetic algorithm as described in Section 7.2. The change of the average fitness vs. generation is shown in Figure 7.2. The slices of the optimal 3-D SE obtained by means of GA are shown in Figure 7.3. The 3-D structuring element size is $4 \times 4 \times 4$. The original phantom MR images are shown in Figure 6.6 and the original phantom PET images are shown in Figure 6.7. The registered images are shown with MR brain contour superimposed on the PET image data. The IPAR registered images are shown in Figure 6.8 while the IPAR/OMST registered images are shown in Figure 7.4. The resulting images demonstrate

fig/phantom.opt/avgfitness.ps

Figure 7.2: Average fitness vs. generation.

fig/phantomfig/p/se/sl.00fig/p/se/sl.01fig/p/se/sl.02opt/se/sl.03

Figure 7.3: The slices of the optimal 3-D structuring element.

fig/phantom.opfig/xphtamtaxmhoqfig/tphsdimtatk2eqfig/tphsdimtatk3eqfig/tphsdimtatk4eqtp/etersdina.t05ed.pet/slice.06

fig/phantom.opfig/xphtamtaxmhoqfig/tphsdimtatk7eqfig/tphsdimtatk8eqfig/tphsdimtatk9eqtp/etersdina.t10ed.pet/slice.11

fig/phantom.opfig/xphtamtaxmhoqfig/tphsdimtatk12eqfig/tphsdimtatk13eqfig/tphsdimtatk14eqtp/etersdina.t15ed.pet/slice.16

fig/phantom.opfig/xphtamtaxmhoqfig/tphsdimtatk17eqfig/tphsdimtatk18eqtp/etersdina.t19ed.pet/slice.20

Figure 7.4: IPAR/OMST registered phantom PET images. MR brain contours are superimposed on PET brain image.

the accurate registration.

7.3.2   Brain Phantom Registration Error Analysis

The error analysis is performed as explained in Section 6.6.1. The mismatch between Derenzo tube positions in registered MR and PET images is used to measure the error. Relative errors for odd Derenzo slices for IPAR and IPAR/OMST registration are shown in Table 7.1. The results are slightly better then in the experiment described in Chapter 6. This is due to a fact that a better structuring element is used for MST registration. The genetic algorithm procedure enabled the selection of the better, optimal structuring element. Using the relation between

198

Slice	IPAR	IPAR/OMST
11	0.68	0.45
13	0.90	0.57
15	0.78	0.56
17	0.78	0.60
19	0.76	0.60
avg	0.78	0.56

Table 7.1: The relative errors for Derenzo slices (11,13,15,17,19) and the average error for IPAR and IPAR/OMST algorithms.

the two error measures (the centroid distance and the average relative mismatched area) which is developed in Chapter 6 it is possible to estimate the centroid distance from the mismatched area. The centroid distance error measure for IPAR/OMST method is computed similarly. The average mismatched area is equal to 0.56 and the graph in Figure 6.14 gives the corresponding relative centroid distance of 0.46. The resulting centroid distance is equal to $2h = 3.63$ pixels. The pixel size of 0.9375 mm means that the centroid distance is equal to 3.40 mm.

7.3.3 Real Patient Data Experiment

The experimental results obtained using real patient data are presented in this section. The same patient data is used as in real patient data experiment in Chapter 6.

In the first step, the optimal structuring element is determined using the genetic algorithm procedure as presented in Section 7.2. The brain ventricle is extracted from MR images and used to generate fine positional variations as shown in Figure 7.1. The change of the average fitness vs. generation is shown in Figure 7.5. The slices of the optimal 3-D SE obtained by means of GA are shown in Figure 7.6. The 3-D structuring element size is $4 \times 4 \times 4$. Due to a

199

fig/real.opt/avgfitness.ps

Figure 7.5: Average fitness vs. generation.

fig/real.opt/se/sl.00 fig/real.opt/se/sl.01 fig/real.opt/se/sl.02 fig/real.opt/se/sl.03

Figure 7.6: The slices of the optimal 3-D structuring element.

fig/real.opt/ver/fig/real.opt/ver/fig/real.opt/ver/fig/real.opt/ver/fig/real.opt/ver/slice.03ed.pet/slice.06

fig/real.opt/ver/fig/real.opt/ver/fig/real.opt/ver/fig/real.opt/ver/fig/real.opt/ver/slice.10ed.pet/slice.11

fig/real.opt/ver/fig/real.opt/ver/fig/real.opt/ver/fig/real.opt/ver/fig/real.opt/ver/slice.15ed.pet/slice.16

Figure 7.7: IPAR/OMST registered PET images. MR brain contours are superimposed on PET brain image.

lack of objective evaluation method for real patient images, a subjective evaluation by a human viewer is performed here. The original MR brain data is shown in Figure 6.15 and the original PET data is shown in Figure 6.16. The registration result is represented by superimposing the MR brain contour on the PET image data. Figure 6.17 shows the IPAR registered images. The IPAR/OMST registered images are shown in Figure 7.7. Visual inspection of the resulting registered images shows an improvement in registration quality of IPAR/OMST algorithm over IPAR algorithm.

Chapter 8

Conclusions and Further Work

In this chapter, the presented material is summarized and possible directions for the future work are discussed.

8.1 Overview

An introduction to the foundations of mathematical morphology is given in Chapter 2. Basic morphological operations have been defined and their properties examined.

An overview of shape description methods is presented in Chapter 3. Most influential theories of visual perception are briefly mentioned. Shape description methods have been divided into classes and most important and characteristic approaches addressed. Typical shape matching procedures are reviewed.

The main topic of this dissertation is the morphological shape description. The following new methods and applications are proposed.

1. Morphological Signature Transform (MST) based method for shape description (Chapter 4).

2. Genetic algorithm based method for the optimal shape matching (Chapter 5).

3. Application of MST-based shape description to medical image registration problem (Chapter 6)

4. Genetic algorithm based method for the optimal medical image registration (Chapter 7).

8.2   Proposed Methods and Applications

8.2.1   Morphological Shape Description

A novel shape description method has been proposed in Chapter 4. The proposed shape description algorithm is based on the Morphological Signature Transform (MST) for decomposition of a shape to a number of *signature* shapes. The method uses the areas of successively morphologically eroded images and the multiresolution representation of the input image to obtain morphological descriptors. The morphological descriptor is computed using rotated and scaled versions of the images, and one (or more) structuring elements. The resulting morphological descriptor is invariant to rotation, scaling, and translation. The shape description method has been applied to the shape matching problem. The proposed MST can be classified into the global space domain methods but the MST-shape description is a global scalar transform technique. Experiments have shown that the method exhibits invariance to rotation, scaling and translation

and is robust to the noise.

Maragos [173] used the derivative of the expression similar to Equation 4.11 with respect to scale p to characterize shape properties (the morphological opening instead of erosion and the convex disk-shaped structuring element are used). However, in general case of non-convex structuring elements, the approach presented here provides additional shape information. That is because the approach described here uses *double* multiscaling to vary both the scale of the object and the scale of the structuring element, as opposed to a scaling of the structuring element only in [173]. In addition, the MST shape description method uses multiple structuring elements to provide an increased sensitivity to the various shape properties.

The proposed shape description algorithm is based on decomposition of the initial shape to a number of *signature* shapes. The question that might be posed here is whether the areas of signature shapes or the areas of the residues (that is differences between the successive signature shape areas) should be used as a shape descriptors. The real number sequence of the successive areas can be viewed as a discrete signal. The sequence of the residues (first order differences of the successive areas sequence) can then be viewed as a derivative of the areas sequence. It is well known from the linear filtering theory that derivation procedure enhances high frequency noise in the signal. Therefore, the residual approach would yield the shape descriptor which would contain more noise then the area approach. Therefore the algorithm proposed here uses area as the basic shape descriptor instead of residue which is more prone to noise.

The proposed MST-based method can be applied to any n-dimensional object representation and recognition problem.

8.2.2　Optimal Shape Matching

A new method for the selection of the optimal structuring element for use with the morphological signature transform (MST) based shape representation method is proposed in Chapter 5. The method uses genetic algorithm to search for the optimal structuring element for description of a predefined class of model shapes. A novel two-dimensional crossover operator with two crossover points is used in genetic algorithm. The entropy of fitness values is used to estimate the population diversity. The crossover rate is variable and is determined by the population diversity in order to avoid the problem of premature convergence. The obtained optimal structuring element has been evaluated on the shape matching problem. Experiments have shown that the optimal structuring element enables accurate shape matching and is robust to the noise. The matching is optimal in the sense of best shape discrimination. The best shape discrimination is achieved when the minimal distance between the shape descriptors of a given class of shapes is maximized.

8.2.3　Application of Morphological Shape Description to Medical Image Registration

The IPAR and MST-based shape description method has been applied to the problem of multi-modality medical image registration. The method consists of two major steps. In the first step, the IPAR algorithm is used for initial registration of MR and PET volumes. In the second step, the MST-based brain ventricle shape matching is performed for more accurate registration. A multi-grid method for the search of the ventricle principal axis angle space has been developed.

205

The multi-grid algorithm has a higher computational efficiency then a simple search algorithm.

The validation of the new IPAR/MST method is conducted in two different ways. The experiments were performed using a physical brain phantom and a real patient MR and PET brain data. The physical phantom was used for an objective validation and error estimation of the registration procedures. It has been demonstrated quantitatively that the IPAR/MST algorithm improves the accuracy of the original IPAR algorithm. The real patient data was used for a subjective visual evaluation of the registration performance.

8.2.4   Optimal Medical Image Registration

A genetic algorithm procedure for selection of the optimal 3-D structuring element for IPAR/MST registration is presented. The procedure provides the structuring element which is optimal in the sense of the best ventricle position discrimination. This allows the optimal selection of one of the fine ventricle positional variations.

Experimental results show that the optimal IPAR/MST method performs better then without the optimization of the structuring element. In the worst case, without optimization, a registration may be inaccurate if unsuitable structuring element is selected. However, the use of optimization procedure guarantees that the optimal structuring element is used and that accurate registration results are achieved.

8.3 Future Work

The future work on MST-based shape description will include investigation of other morphological operations for generation of signature shapes. A shape descriptor other then area could be used for description of signature shapes.

Possible genetic algorithm improvements include different chromosome representations of the problem, other crossover operators, and different fitness functions (optimization criteria).

Application of MST-based description to medical image registration requires measurements on a larger data set for a better verification of the technique.

Bibliography

[1] Y. Aloimonos. Visual shape computation. *Proceedings of the IEEE*, 76:899–916, 1988.

[2] L. Arata, A. P. Dhawan, J. Broderick, M. Gaskil, A. V. Levy, and N. D. Volkow. Multi-modality three-dimensional principal axis brain image registration and analysis: Part1: Model based segmenation, labeling, and analysis of internal structure of mr brain images. Submitted to *Journal of Nuclear Medicine*, 1992.

[3] L. K. Arata. *Multi-modality three-dimensional brain image registration and analysis*. PhD thesis, University of Cincinnati, 1992.

[4] L. K. Arata and A. P. Dhawan. Iterative principal axes registration: A new algorithm for retrospe ctive correlation of mr-pet brain images. In *Proceedings of IEEE EMBS Conference*, 1992.

[5] G. Arce, N. Gallagher, and T. Nodes. Median filters: Theory for one- and two-dimensional filters. In T. Huang, editor, *Advances in Computer Vision and Image Processing, vol.2*, pages 89–166. JAI Press Inc., 1986.

[6] C. Arcelli. Pattern thinning by contour tracing. *Computer Graphics and Image Processing*, 17:130–144, 1981.

[7] H. Asada and M. Brady. The curvature primal sketch. *IEEE Transactions on PAMI*, 8:2–14, 1986.

[8] T. Asano and T. Asano. Minimum partition of polygon regions into trapezoids. In *Symposium on Foundations Computer Science*, pages 233–241, 1983.

[9] K. E. Atkinson. *An Introduction to Numerical Analysis*. Wiley, 1989.

[10] F. Attneave. Some informational aspects of visual perception. *Psychol. Rev.*, 61:183–193, 1954.

[11] F. Attneave and M. D. Arnoult. The quantitative study of shape and pattern perception. In L. Uhr, editor, *Pattern Recognition*, pages 123–141. Wiley, 1966.

[12] J. Babaud, A. Witkin, M. Baudin, and R. Duda. Uniqueness of the gaussian kernel for scale-space filtering. *IEEE Transactions on PAMI*, 8:26–33, 1986.

[13] Ballard and Brown. *Computer Vision*. Prentice Hall, 1988.

[14] M. Barnsley. *Fractals Everywhere*. Academic Press, 1988.

[15] R. Bartels, J. Beatty, and B. Barsky. *An Introduction to Splines for use in Computer Graphics and Geometric Modeling*. Morgan Kaufmann, 1987.

[16] J. Basak, S. Chaudhury, S. K. Pal, and D. Majumder. Matching of structural shape descriptions with hopfield net. *International Journal of Pattern Recognition and Artificial Intelligence*, 7:377–404, 1993.

[17] S. O. Belkasim, M. Shridhar, and M. Ahmadi. Pattern recognition with moment invariants: A comparative study and new results. *Pattern Recognition*, 24:1117–1138, 1991.

[18] A. Bengtsson and J. Eklundh. Shape representation by multiscale contour approximation. *IEEE Transactions on PAMI*, 13:85–93, 1991.

[19] J. R. Bennet and J. S. McDonald. On the measurement of curvature in a quantized environment. *IEEE Transactions on Computers*, 24:803–820, 1975.

[20] M. Bergstrom and et al. Head fixation device for reproducible position alignment in transmission ct and positron emmision tomography. *Journal of Computer Assisted Tomography*, 5:136–141, 1981.

[21] I. Biederman. Human image understanding: Recent research and a theory. *Computer Vision, Graphics, and Image Processing*, 32:29–73, 1985.

[22] C. Bjorklund and T. Pavlidis. Global shape analysis by k-syntactic similarity. *IEEE Transactions on PAMI*, 3:144–155, 1981.

[23] H. Blum. A transformation for extracting new descriptors of shape. In Whaten-Dunn, editor, *Models for the Perception of Speech and Visual Forms*, pages 362–380. MIT Press, 1967.

[24] H. Blum. Biological shape and visual science (part i). *J. theor. Biology*, 38:205–287, 1973.

[25] H. Blum and R. Nagel. Shape description using weighted symmetric axis features. *Pattern Recognition*, 10:167–180, 1978.

[26] J. A. Bondy and U. S. R. Murty. *Graph Theory with Applications*. North-Holland, 1981.

[27] Lashon Booker. Improving search in genetic algorithms. In Lawrence Davis, editor, *Genetic Algorithms and Simulated Annealing*, pages 61–73. Pitman Publishing, 1987.

[28] G. Borgefors. Distance transformations in arbitrary dimensions. *Computer Vision, Graphics, and Image Processing*, 27:321–345, 1984.

[29] G. Borgefors. Distance transformation in digital images. *Computer Vision, Graphics, and Image Processing*, 34:344–371, 1986.

[30] M. Brady. Criteria for representations of shape. In J. Beck, B. Hope, and A. Rosenfeld, editors, *Human and Machine Vision*, pages 39–84. Associated Press, 1983.

[31] M. Brady and H. Asada. Smoothed local symmetries and their implementation. *Int'l Journal of Robotics Research*, 3:36–61, 1984.

[32] R. A. Brooks. Symbolic reasoning among 3-d models and 2-d images. *Artificial Intelligence*, 17:285–348, 1981.

[33] L. Calabi and W. Hartnett. Shape recognition, prairie fires, convex deficiencies, and skeletons. *The American Mathematical Monthly*, 75:335–341, 1968.

[34] I. Chakravarty. A generalized line and junction labeling scheme with application to scene analysis. *IEEE Transactions on PAMI*, 1:202–205, 1979.

[35] C. C. Chang, S. M. Hwang, and D. J. Buehrer. A shape recognition scheme based on relative distances of feature points from the centroid. *Pattern Recognition*, 24:1053–1063, 1991.

[36] M. Chen and P. Yan. A multiscaling approach based on morphological filtering. *IEEE Transactions on PAMI*, 11:694–700, 1989.

[37] S. Chen, J. Keller, and R. Crownover. Shape from fractal geometry. *Artificial Intelligence*, 43:199–218, 1990.

[38] R. Chin and C. Dyer. Model-based recognition in robot vision. *ACM Computing Surveys*, 18:67–108, 1986.

[39] C. Chu and E. Delp. Impulsive noise supression and background normalization of electrocardiogram signals using morphological operators. *IEEE Transactions on Biomedical Engineering*, 36:262–273, 1989.

[40] M. B. Clowes. On seeing things. *Artificial Intelligence*, 1:79–115, 1970.

[41] Special issue: Visual cognition, 1984.

[42] J. H. Connell and M. Brady. Generating and generalizing models of visual objects. *Artificial Intelligence*, 31:159–183, 1987.

[43] Cornsweet. *Visual Perception*. Associated Press, 1970.

[44] T. Crimmins and W. Brown. Image algebra and automatic shape recognition. *IEEE Transactions on Aerospace Engineering*, 21:60–69, 1985.

[45] M. Das, M. J. Paulik, and N. K. Loh. A bivariate autoregressive modeling technique for analysis and classification of planar shapes. *IEEE Transactions on PAMI*, 12:97–103, 1990.

[46] L. Davis. Understanding shape: Angles and sides. *IEEE Transactions on Computers*, 26:236–242, 1977.

[47] L. Davis. Two-dimensional shape representation. In Young and Fu, editors, *Handbook of Pattern Recognition and IP*, pages 233–245. Associated Press, 1986.

[48] Lawrence Davis, editor. *Handbook of Genetic Algorithms*. Van Nostrand Reinhold, 1971.

[49] Lawrence Davis. Genetic algorithms and simulated annealing: An overview. In Lawrence Davis, editor, *Genetic Algorithms and Simulated Annealing*, pages 1–11. Pitman Publishing, 1987.

[50] C. de Boor. *A Practical Guide to Splines*. Springer-Verlag, 1978.

[51] A. K. DeJong. Adaptive system design: A genetic approach. *IEEE Transactions on SMC*, 10, 1980.

[52] S. E. Derenzo, T. F. Budinger, J. L. Canoon, W. L. Greenberg, R. M. Huesman, and T. Vulotich. The donner 280-crystal high resolution positron tomograph. *IEEE Transactions on Nuclear Science*, 26:2790–2793, 1979.

[53] R. L. Devaney and L. Keen, editors. *Chaos and Fractals: The Mathematics Behind the Computer Graphics.* American Mathematical Society, 1988.

[54] P. A. Devijver and J. Kittler. *Pattern Recognition: A Statistical Approach.* Prentice Hall, 1982.

[55] A. Dhawan, S. Loncaric, and S. Ramachandran. Optimal morphological image processing. In *Proceedings of the IEEE International Conference on Systems Engineering*, 1991.

[56] A. Dill, M. Levine, and P. Noble. Multiple resolution skeletons. *IEEE Transactions on PAMI*, 9:495–504, 1987.

[57] E. Dougherty and C. Giardina. Closed-form representation of convolution, dilation, and erosion in the context of image algebra. In *Proceedings of the IEEE CVPR*, pages 754–759, 1988.

[58] E. Dougherty and E. Kraus. Filtering by morphological pseudoconvolutions. In *SPIE-1247*, 1990.

[59] E. Dougherty and J. Pelz. Morphological granulometric analysis of electrophotographic images - size distribution statistics for process control. *Optical Engineering*, 30:438–445, 1991.

[60] E. Dougherty, J. Pelz, and D. Marsh. Morphological granulometric analysis of electrophotographic images. In *SPIE-1153*, 1989.

[61] E. Dougherty and P. Sehdev. A robust image processing language in the context of image algebra. In *Proceedings of the IEEE CVPattern Recognition*, pages 748–753, 1988.

[62] E. R. Dougherty and Charles R. Giardina. *Mathematical Methods for Artificial Intelligence and Autonomous Systems*. Prentice Hall, 1988.

[63] E. R. Dougherty, J. Pelz, F. Sand, and A. Lent. Morphological image segmentation by local granulometric size distributions. *Journal of Electronic Imaging*, 1:46–60, 1992.

[64] Edward R. Dougherty. Optimal mean-square n-observation digital morphological filters i. optimal binary filters, ii. optimal gray-scale filters. *Computer Vision, Graphics, and Image Processing: Image Understanding*, 55:36–72, 1992.

[65] S. Dubois and F. Glanz. An autoregressive model approach to two-dimensional shape classification. *IEEE Transactions on PAMI*, 8:55–66, 1986.

[66] R. Duda and P. Hart. *Descriptions of Line and Shape*, chapter 9, pages 327–378. John Wiley, New York, 1973.

[67] R. Duda and P.Hart. *Pattern Classification and Scene Analysis*. John Wiley, New York, 1973.

[68] M. Duff, D. Watson, T. Fountain, and G. Shaw. A cellular logic array for image processing. *Pattern Recognition*, 5:229–247, 1973.

[69] D. Eberly and D. Wenzel. Adaptation of group algebras to signal and image processing. *Computer Vision, Graphics, and Image Processing: Graphical Models and Image Processing*, 53:340–348, 1991.

[70] W. D. Ellis, editor. *A Source Book of Gestalt Psychology*. Harcourt Brace Jovanovic, 1923.

[71] A. C. Englander. Machine learning of visual recognition using genetic algorithms. In J. Grefenstette, editor, *Proceedings of an International Conference on Genetic Algorithms and Their Applications*, pages 197–201. Pittsburgh, PA: The Robotics Institute of Carnegie-Mellon University, 1985.

[72] A. C. Evans and et al. Mri-pet correlation in three dimensions using a volume-of-interest (voi) atlas. *Journal of Cerebral Blood Flow and Metabolism*, 11:A69–A78, 1991.

[73] Ferrari, Shankar, and Sklansky. Minimal rectangular partition of digitized blobs. In *5th Intl. Conf. Pattern Recognition*, pages 1040–1043, 1980.

[74] J. M. Fitzpatrick, J. J. Grefenstette, and D. Van Gucht. Image registration by genetic search. In *Proceedings of IEEE SOUTHEASTCON '84*, pages 460–464, 1984.

[75] G. Freeland and T. Durrani. Ifs fractals and the wavelet transform. In *Proceedings of the ICASSP-90*, pages 2345–2348, 1990.

[76] Freeman and Skapura. *Neural Networks*. Addison Wesley, 1991.

[77] H. Freeman. On the encoding of arbitrary geometric configurations. *IRE Transactions*, 10:260–268, 1961.

[78] H. Freeman. Computer processing of line-drawing images. *ACM Computing Surveys*, 6:57–97, 1974.

[79] H. Freeman. Shape description via the use of critical points. *Pattern Recognition*, 10:159–166, 1978.

[80] H. Freeman and J. Saghri. Comparative analysis of line-drawing modeling schemes. *Computer Graphics and Image Processing*, 12:203–223, 1980.

[81] K. S. Fu. *Syntactic Methods in Pattern Recognition*. Associated Press, 1974.

[82] K. S. Fu. Syntactic image modeling using stohastic tree grammars. *Computer Graphics and Image Processing*, 12:136–152, 1980.

[83] K. S. Fu. A general (syntactic-semantic) approach to picture analysis. In K. S. Fu and T. Kunii, editors, *Picture Engineering*, pages 56–74. Springer Verlag, 1982.

[84] K. S. Fu. *Syntactic Pattern Recognition and Applications*. Prentice Hall, 1982.

[85] J. M. Gauch and S. M. Pizer. The intensity axis of symmetry and its application to image segmentation. *IEEE Transactions on PAMI*, 15:753–770, 1993.

[86] F. Gerritsen and P. Verbeek. Implementation of cellular-logic operators using 3x3 convolution and table lookup hardware. *Computer Vision, Graphics, and Image Processing*, 27:115–123, 1984.

[87] P. Ghosh. A solution of polygon containment, spatial planning, and other related problems using minkowski operations. *Computer Vision, Graphics, and Image Processing*, 49:1–35, 1990.

[88] C. R. Giardina and E. R. Daugherty. *Morphological Methods in Image Processing*. Prentice Hall, 1988.

[89] J. J. Gibson. *The perception of the visual world*. Houghton, 1950.

[90] J. J. Gibson. What is form ? *Psychology Review*, 58:403–412, 1951.

[91] Gilio. *Interactive Computer Graphics*. Prentice Hall, 1978.

[92] V. Goetcherian. From binary to grey tone image processing using fuzzy logic concepts. *Pattern Recognition*, 12:7–15, 1980.

[93] M. Golay. Hexagonal parallel pattern transformations. *IEEE Transactions on Computers*, 18:733–740, 1969.

[94] D. E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, 1989.

[95] David Goldberg. Genetic algorithms and rule learning in dynamic system control. In J. Grefenstette, editor, *Proceedings of an International Conference on Genetic Algorithms and Their Applications*, pages 8–15. Pittsburgh, PA: The Robotics Institute of Carnegie-Mellon University, 1985.

[96] H. E. Gombrich. *Art and Illusion*. Phaidon Press, London, 1960.

[97] R. Gonzales and P. Wintz. *Digital Image Processing*. Addison Wesley, 1987.

[98] Sheperd Gordon. *The Synaptic Organization of the Brain*. Oxford, 1990.

[99] A. Goshtasby. Description and descrimination of planar shapes using shape matrices. *IEEE Transactions on PAMI*, 7:738–743, 1985.

[100] J. Goutsias and D. Schonfeld. Morphological representation of discrete and and binary images. *IEEE Transactions on Signal Processing*, 39:1369–1379, 1991.

[101] V. Govindan and A. Shivaprasad. Character recognition - a review. *Pattern Recognition*, 23:671–683, 1990.

[102] G. H. Granlund. Fourier preprocessing for hand print character recognition. *IEEE Transactions on Computers*, 21:195–201, 1972.

[103] D. Granrath. The role of human visual models in image processing. *Proceedings of the IEEE*, 69:552–561, 1981.

[104] V. Guillemin and A. Pollack. *Differential Topology*. Prentice Hall, 1974.

[105] H. Hadwiger. *Vorlesungen uber Inhalt, Oberflaeche, und Isoperimetrie*. Springer Verlag, 1957.

[106] H. W. Hake. Form discrimination and the invariance of form. In L. Uhr, editor, *Pattern Recognition: Theory, Experiments, Computer Simulations, and Dynamic Models of Form, Perception, and Discovery*, pages 142–173. John Wiley, 1966.

[107] R. Haralick and L. Shapiro. *Computer and Robot Vision, vol. 2.* Addison Wesley, 1992.

[108] R. Haralick and L. Shapiro. *Comuter and Robot Vision, vol. 1.* Addison Wesley, 1992.

[109] R. Haralick, S. Sternberg, and X. Zhuang. Image analysis using mathematical morphology. *IEEE Transactions on PAMI*, 9:532–550, 1987.

[110] R. Haralick, X. Zhuang, C. Lin, and J. Lee. The digital morphological sampling theorem. *IEEE Transactions on ASSP*, 37:2067–2090, 1989.

[111] R. Haralick, X. Zhuang, C. Lin, and J. Lee. The digital morphological sampling theorem. In Kasturi and Trivedi, editors, *Image Analysis Applications*, pages 373–418. MD, 1990.

[112] Y. He and A. Kundu. 2-d shape classification using hidden markov model. *IEEE Transactions on PAMI*, 13:1172–1184, 1991.

[113] D. O. Hebb. *The organization of behavior.* Wiley, 1949.

[114] R. Hecht-Nielsen. *Neurocomputing.* Addison Wesley, 1990.

[115] H. Heijmans. Morphological filtering and iteration. In *SPIE-1360*, pages 166–175, 1990.

[116] H. Heijmans and C. Ronse. The algebraic basis of mathematical morphology i. dilations and erosions. *Computer Vision, Graphics, and Image Processing*, 50:245–295, 1990.

[117] T. Henderson. Syntactic and structural methods i and ii. In O. Faugeras, editor, *Fundamentals of CV*, pages 273–292. Cambridge University Press, 1983.

[118] J. Hereford and W. Rhodes. Nonlinear optical image filtering by time-sequential decomposition. *Optical Engineering*, 27:274–279, 1988.

[119] J. Hertz, A. Krogh, and R. G. Palmer. *Introduction to the theory of neural computation.* Addison Wesley, 1991.

[120] W. Hoff and N. Ahuja. Surfaces from stereo: Integrating feature matching, disparity estimation, and contour detection. *IEEE Transactions on PAMI*, 11:121–136, 1989.

[121] D. D. Hoffman and W. A. Richards. Parts of recognition. *Cognition*, 18:65–96, 1984.

[122] John H. Holland. *Adaption in Neural and Artificial Systems.* Ann Arbor: University of Michigan Press, 1975.

[123] J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proc. of the National Academy of Sciences*, 79:2554–2558, 1982.

[124] J. Hopfield and D. Tank. Neural computation of decisions in optimization problems. *Biological Cybernetics*, 52:141–152, 1985.

[125] R. Horaud and M. Brady. On the geometric interpretation of image contours. *Artificial Intelligence*, 37:333–353, 1988.

[126] B. Horn. *Robot Vision.* MIT Press and McGraw-Hill, 1986.

[127] B. Horn. Height and gradient from shading. In *IJCV-5*, pages 37–75, 1990.

[128] M. K. Hu. Visual pattern recognition by moment invariants. *IRE Transactions on Information Theory*, 8:179–187, 1962.

[129] Huang, editor. *Two-dimensional Digital Signal Processing II: Transforms and Median Filters*. Springer Verlag, 1981.

[130] T. Huang. Coding of two-tone images. *IEEE Transactions on Communications*, 25:1406–1424, 1977.

[131] Y. Ikebe and S. Miyamoto. Shape design, representation, and restoration with splines. In K. Fu and T. Kunii, editors, *Picture Engineering*, pages 75–95. Springer Verlag, 1982.

[132] H. V. Jagadish and A. M. Bruckstein. On sequential shape descriptions. *Pattern Recognition*, 25:165–172, 1992.

[133] A. K. Jain. *Fundamentals of Digital Image Processing*. Prentice Hall, 1989.

[134] P. N. Johnson-Laird. *The Computer and the Mind: An Introduction to Cognitive Science*. Harvard University Press, 1988.

[135] T. Kanade. Recovery of the three-dimensional shape of an object from a single view. *Artificial Intelligence*, 17:409–460, 1981.

[136] B. Kartikeyan and A. Sarkar. Shape description by time series. *IEEE Transactions on PAMI*, 11:977–984, 1989.

[137] R. Kashyap and R. Chellappa. Stohastic models for closed boundary analysis: Representation and reconstruction. *IEEE Transactions on IT*, 27:627–637, 1981.

[138] H. Kim, K. Park, and M. Kim. Shape decomposition by collinearity. *Pattern Recognition Letters*, 6:335–340, 1987.

[139] D. Kincaid and W. Cheney. *Numerical Analysis*. Brooks/Cole, 1991.

[140] J. Klingler, C. Vaughan, T. Fraker, and L. Andrews. Segmentation of echocardiographic images using mathematical morphology. *IEEE Transactions on Biomedical Engineering*, 35:925–934, 1988.

[141] W. Koehler. *Gestalt psychology*. Liverlight, 1929.

[142] J. Koenderink. The structure of images. *Biological Cybernetics*, 50:363–370, 1984.

[143] J. Koenderink and A. Van Doorn. Dynamic shape. *Biological Cybernetics*, 53:383–396, 1986.

[144] K. Koffka. *Principles of Gestalt psychology*. Harcourt Brace Jovanovic, 1935.

[145] J. R. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, 1992.

[146] J. Kruskal. An overview of sequence comparison: Time warps, string edits, and macro-molecules. *SIAM Review*, 25:201–237, 1983.

[147] C. Lantuejoul. Geodesic segmentation. In L. Uhr, editor, *Multicomputers and Image Processing Algorithms and Programs*, pages 111–124. Associated Press, 1982.

[148] C. Lantuejoul and F. Maisonneuve. Geodesic methods in quantitative image analysis. *Pattern Recognition*, 17:177–187, 1984.

[149] M. Levine. *Vision in Man and Machine*. McGraw-Hill, 1985.

[150] A. V. Levy and et al. The metabolic centroid method for pet brain image analysis. *Journal of Cerebral Blood Flow and Metabolism*, 9:388–397, 1989.

[151] F. Leymarie and M. D. Levine. Simulating the grassfire transform using an active contour model. *IEEE Transactions on PAMI*, 14:56–75, 1992.

[152] M. Leyton. Symmetry-curvature duality. *Computer Vision, Graphics, and Image Processing*, 38:327–341, 1987.

[153] M. Leyton. Inferring causal history from shape. *Cognitive Science*, 13:357–387, 1989.

[154] W. Lin, F. Liao, and T. Lingutla. A hierarchical multiple-view approach to three-dimensional object recognition. *IEEE Transactions on Neural Networks*, 2:84–92, 1991.

[155] Y. Lin, J. Dou, and H. Wang. Contour shape description based on an arch height function. *Pattern Recognition*, 25:17–23, 1992.

[156] H. Liu and M. Srinath. Partial shape classification using contour matching in distance transformation. *IEEE Transactions on PAMI*, 12:1072–1079, 1990.

[157] S. Loncaric and A. Dhawan. A morphological residue approach to shape representation. In *SPIE Proceedings 1658: Nonlinear Image Processing III*, 1992.

[158] S. Loncaric and A. Dhawan. 3-d brain image registration using morphological processing and iterative principal axis transform. In *Proceedings of the 15th Annual International Conference of IEEE Engineering in Medicine and Biology Society*, 1993.

[159] S. Loncaric and A. Dhawan. A morphological signature transform for shape description. *Pattern Recognition*, 26:1029–1037, 1993.

[160] S. Loncaric and A. Dhawan. Optimal mst-based shape description using genetic algorithm. *Submitted to Pattern Recognition*, 1993.

[161] S. Loncaric and A. Dhawan. Optimal shape description using morphological signature transform via genetic algorithm. In *SPIE Proceedings 2030: Image Algebra and Morphological Image Processing IV*, 1993.

[162] S. Loncaric, A. Dhawan, and S. Narayan. 3-d brain image registration using morphological processing and ite rative principal axis transform. In *Proceedings of the Emerging Technologies in Biomedical Engineer ing Conference*, 1993.

[163] A. Loui, A. Venetsanopoulos, and K. Smith. Two-dimensional shape representation using morphological correlation functions. In *ICASSP-90*, pages 2165–2168, 1990.

[164] D. G. Lowe. *Perceptual Organization and Visual Recognition*. Kluwer Academic, 1985.

[165] D. G. Lowe. Three-dimensional object recognition from single two-dimensional images. *Artificial Intelligence*, 31:355–395, 1987.

[166] M. Maes. Polygonal shape recognition using string-matching techniques. *Pattern Recognition*, 24:433–440, 1991.

[167] B. Mandelbrot. *Fractals : form, chance, and dimension*. Freeman, 1977.

[168] B. Mandelbrot. *The fractal geometry of nature*. Freeman, 1983.

[169] P. Maragos. Tutorial on advances in morphological image processing and analysis. *Optical Engineering*, 26:623–632, 1987.

[170] P. Maragos. Morphology-based symbolic image modelling, multi-scale nonlinear smoothing, and pattern spectrum. In *ICCVPR-88*, pages 766–773, 1988.

[171] P. Maragos. Optimal morphological approaches to image matching and object recognition. In *ICCV-88*, pages 695–699, 1988.

[172] P. Maragos. Morphological correlation and mean absolute error criteria. In *Proceedings of the ICASSP-89*, pages 1568–1571, 1989.

[173] P. Maragos. Pattern spectrum and multiscale shape representation. *IEEE Transactions on PAMI*, 11:701–716, 1989.

[174] P. Maragos and R. Schafer. Morphological skeleton representation and coding of binary images. *IEEE Transactions on ASSP*, 34:1228–1244, 1986.

[175] P. Maragos and R. Schafer. Morphological systems for multidimensional signal processing. *Proceedings of the IEEE*, 78:690–710, 1990.

[176] P. Maragos and D. Ziff. Treshold superposition in morphological image analysis systems. *IEEE Transactions on PAMI*, 12:498–504, 1990.

[177] D. Marr. A theory for cerebral neocortex. *Proceedings of the Royal Society of London*, B176:161–234, 1970.

[178] D. Marr. Early processing of visual information. *Proceedings of the Royal Society of London*, B275:483–519, 1976.

[179] D. Marr. *Vision*. Freeman, San Francisco, 1982.

[180] D. Marr and E. Hildreth. Theory of edge detection. *Proceedings of the Royal Society of London*, B207:187–217, 1980.

[181] D. Marr and H. Nishihara. Representation and recognition of the spatial organization of three-dimensional shapes. *Proceedings of the Royal Society of London*, B200:269–294, 1978.

[182] D. Marr and T. Poggio. A computational theory of human stereo vision. *Proceedings of the Royal Society of London*, B204:301–328, 1979.

[183] D. Marr and S. Ullman. Directional selectivity and its use in early visual processing. *Proceedings of the Royal Society of London*, B211:151–180, 1981.

[184] G. Matheron. *Random Sets and Integral Geometry*. John Wiley, 1975.

[185] M. L. Mauldin. Maintaining diversity in genetic search. In *Proceedings of the National Conference on Artificial Intelligence*, pages 247–250, 1984.

[186] R. McEliece, E. Posner, E. Rodemich, and S. Venkatesh. The capacity of the hopfield associative memory. *IEEE Transactions on Information Theory*, 33:461–482, 1987.

[187] F. Meyer. Iterative image transformations for an automatic screening of cervical smears. *The J. of Histochemistry and Cytochemistry*, 27:128–135, 1979.

[188] F. Meyer. Automatic screening of cytological specimens. *Computer Vision, Graphics, and Image Processing*, 35:356–369, 1986.

[189] F. Meyer. Digital euclidean skeletons. In *SPIE-1360 Visual Communications and Image Processing*, pages 251–262, 1990.

[190] F. Meyer and S. Beucher. Morphological segmentation. *Journal of Visual Communication and Image Representation*, 1:21–46, 1990.

[191] R. Miles. A survey of geometric probability in the plane, with emphasis on stohastic image modeling. *Computer Graphics and Image Processing*, 12:1–24, 1980.

[192] H. Minkowski. Volumen and oberflaeche. *Math. Ann.*, 57:447–495, 1903.

[193] F. Mokhtarian and A. Mackworth. Scale-based description and recognition of planar curves and two-dimensional shapes. *IEEE Transactions on PAMI*, 8:34–43, 1986.

[194] C. Moran. A morphological transformation for sharpening edges of features before segmentation. *Computer Vision, Graphics, and Image Processing*, 49:85–94, 1990.

[195] M. E. Mortenson. *Geometric Modeling*. John Wiley, 1985.

[196] J. Mott-Smith. Medial axis transformations. In Lipkin and Rosenfeld, editors, *Picture Processing and Psychopictorics*, pages 267–283. Associated Press, 1970.

[197] Y. Nakagawa and A. Rosenfeld. A note on the use of local min and max operations in digital picture processing. *IEEE Transactions on SMC*, 8:632–635, 1978.

[198] R. Nevatia. *Machine Perception*. Prentice Hall, 1982.

[199] R. Nevatia. *Shape Analysis and Recognition*, chapter 5, pages 61–89. Prentice Hall, 1982.

[200] J. O'Rourke and N. Badler. Decomposition of three-dimensional objects into spheres. *IEEE Transactions on PAMI*, 1:295–305, 1979.

[201] Athanasios Papoulis. *Probability, Random Variables, and Stohastic Processes*. McGraw-Hill, 1984.

[202] S. Parui and D. Majumder. Symmetry analysis by computer. *Pattern Recognition*, 16:6367, 1983.

[203] S. Parui, E. Sarma, and D. Majumder. How to discriminate shapes using the shape vector. *Pattern Recognition Letters*, 4:201–204, 1986.

[204] T. Pavlidis. Polygonal approximations by newton's method. *C*, 26:800–807, 1977.

[205] T. Pavlidis. *Structural Pattern Recognition*. Springer Verlag, 1977.

[206] T. Pavlidis. A review of algorithms for shape analysis. *Computer Graphics and Image Processing*, 7:243–258, 1978.

[207] T. Pavlidis. Algorithms for shape analysis of contours and waveforms. *IEEE Transactions on PAMI*, 2:301–312, 1980.

[208] T. Pavlidis. *Algorithms for Graphics and Image Processing*. Springer Verlag, 1982.

[209] T. Pavlidis and F. Ali. A hierarchical syntactic shape analyzer. *IEEE Transactions on PAMI*, 1:2–9, 1979.

[210] S. Pei and F. Chen. Subband image decomposition by mathematical morphology. In *SPIE-1360*, pages 293–301, 1990.

[211] S. Pei and F. Chen. Subband decomposition of monochrome and color images by mathematical morphology. *Optical Engineering*, 30:921–933, 1991.

[212] S. Pei and H. Tsai. Image mosaic and interpolation by multiresolution morphological pyramids. In *SPIE-1360*, pages 284–292, 1990.

[213] S. Peleg and A. Rosenfeld. A min-max medial axis transformation. *IEEE Transactions on PAMI*, 3:208–210, 1981.

[214] C. A. Pelizzari and et al. Accurate three-dimensional registration of ct, pet, and/or mr image s of the brain. *Journal of Computer Assisted Tomography*, 13:20–26, 1989.

[215] A. Pentland. Fractal-based description of natural scenes. *IEEE Transactions on PAMI*, 6:661–674, 1984.

[216] A. Pentland. Perceptual organization and the representation of natural form. *Artificial Intelligence*, 28:293–331, 1986.

[217] E. Persoon and K. Fu. Shape discrimination using fourier descriptors. *IEEE Transactions on SMC*, 7:170–179, 1977.

[218] J. Pfaltz and A. Rosenfeld. Computer representation of planar regions by their skeletons. *Communications of the ACM*, 10:119–125, 1967.

[219] S. Pinker. Visual cognition: An introduction. *Cognition*, 18:1–63, 1984.

[220] I. Pitas. Morphological signal decomposition. In *Proceedings of the ICASSP-90*, pages 2169–2172, 1990.

[221] I. Pitas and C. Kotropoulos. Texture analysis and segmentation of seizmic images. In *ICASSP-89*, pages 1437–1440, 1990.

[222] I. Pitas and A. Maglara. Range image analysis by using morphological signal decomposition. *Pattern Recognition*, 24:165–181, 1991.

[223] I. Pitas and A. Venetsanopoulos. Shape decomposition by mathematical morphology. In *Proceedings of the ICCV-87*, pages 621–625, 1987.

[224] I. Pitas and A. Venetsanopoulos. *Non-Linear Digital Filters*. Kluwer Academic, 1989.

[225] I. Pitas and A. Venetsanopoulos. Morphological shape decomposition. *IEEE Transactions on PAMI*, 12:38–45, 1990.

[226] I. Pitas and A. N. Venetsanopoulos. Morphological shape representation. *Pattern Recognition*, 25:555–565, 1992.

[227] Ioannis Pitas and Anastasios Venetsanopoulos. Order statistics in digital image processing. *Proceedings of the IEEE*, 80:1893–1921, 1992.

[228] S. Pizer, W. Oliver, and S. Bloomberg. Hierarchical shape description via the multiresolution symmetric axis transform. *IEEE Transactions on PAMI*, 9:505–511, 1987.

[229] J. Ponce. On characterizing ribbons and finding skewed symmetries. *Computer Vision, Graphics, and Image Processing*, 52:328–340, 1990.

[230] H. Vincent Poor. *An Introduction to Signal Detection and Estimation*. Springer-Verlag, 1988.

[231] M. I. Posner, editor. *Foundations of Cognitive Science*. MIT Press, 1989.

[232] K. Preston. Feature extraction by golay hexagonal pattern transforms. *IEEE Transactions on Computers*, 20:1007–1014, 1971.

[233] K. Preston. Some notes on cellular logic operators. *IEEE Transactions on PAMI*, 3:476–481, 1981.

[234] K. Preston. Cellular logic computers for pattern recognition. *IEEE Computer*, 16:36–47, 1983.

[235] K. Preston. ξ-filters. *IEEE Transactions on ASSP*, 31:861–876, 1983.

[236] K. Preston. Cellular logic arrays for image processing. In Young and Fu, editors, *Handbook of Pattern Recognition and Image Processing*, pages 395–436. Associated Press, 1986.

[237] K. Preston, M. Duff, S. Levialdi, P. Norgren, and J. Toriwaki. Basics of cellular logic with some applications in medical image processing. *Proceedings of the IEEE*, 67:826–859, 1979.

[238] R. J. Prokop and A. P. Reeves. A survey of moment-based techniques for unoccluded object representation and recognition. *CVGIP: Graphical Models and Image Processing*, 54:438–460, 1992.

[239] S. P. Raya and J. K. Udupa. Shape-based interpolation of multidimensional objects. *IEEE Transactions on Medical Imaging*, 9:32–42, 1990.

[240] C. W. Richards and H. Hemami. Identification of three-dimensional objects using fourier descriptors of the boundary curve. *IEEE Transactions on SMC*, 4:371–378, 1974.

[241] W. Richards and D. Hoffman. Codon constraints on closed 2d shapes. *Computer Vision, Graphics, and Image Processing*, 31:265–281, 1985.

[242] G. Ritter and J. Wilson. Image algebra in a nutshell. In *ICCV-87*, pages 641–645, 1987.

[243] G. Ritter, J. Wilson, and J. Davidson. Image algebra: An overview. *Computer Vision, Graphics, and Image Processing*, 49:297–331, 1990.

[244] J. Roerdink. Mathematical morphology on the sphere. In *SPIE-1360*, pages 263–271, 1990.

[245] D. F. Rogers and J. A. Adams. *Mathematical Elements for Computer Graphics, 2nd ed.* McGraw-Hill, 1990.

[246] H. Rom and G. Medioni. Hierarchical decomposition of axial shape description. *IEEE Transactions on PAMI*, 15:973–981, 1993.

[247] C. Ronse and H. Heijmans. The algebraic basis of mathematical morphology. openings and closings. *Computer Vision, Graphics, and Image Processing: Image Understanding,* 54:74–97, 1991.

[248] C. Ronse and B. Macq. Morphological shape and region description. *Signal Processing,* 25:91–105, 1991.

[249] A. Rosenfeld. Cellular architectures: From automata to hardware. In L. Uhr, editor, *Multicomputers and IP Algorithms and Programs,* pages 253–260. Associated Press, 1982.

[250] A. Rosenfeld. Parallel image processing using cellular arrays. *IEEE Computer,* 16:14–20, 1983.

[251] A. Rosenfeld, editor. *Multiresolution Image Processing and Analysis.* Springer-Verlag, 1984.

[252] A. Rosenfeld. Axial representations of shape. *Computer Vision, Graphics, and Image Processing,* 33:156–173, 1986.

[253] A. Rosenfeld and A. C. Kak. *Digital Picture Processing, vol. 1.* Associated Press, 1982.

[254] A. Rosenfeld and A. C. Kak. *Digital Picture Processing, vol. 2.* Associated Press, 1982.

[255] A. Rosenfeld and J. Pfaltz. Sequential operations in digital picture processing. *Journal of ACM,* 13:471–494, 1966.

[256] P. Saint-Marc, H. Rom, and G. Medioni. B-spline contour representation and symmetry detection. *IEEE Transactions on PAMI,* 15:1191–1197, 1993.

[257] D. Sankoff and J. B. Kruskal, editors. *Time Warps, String Edits and Macromolecules: The Theory and Practise of Sequence Comparison.* Addison Wesley, 1983.

[258] R. Schalkoff. *Digital Image Processing and Computer Vision.* Wiley, 1989.

[259] D. Schonfeld and J. Goutsias. Robust morphological representation of binary images. In *ICASSP-90*, pages 2065–2068, 1990.

[260] D. Schonfeld and J. Goutsias. Optimal morphological pattern restoration from noisy binary images. *IEEE Transactions on PAMI*, 13:14–29, 1991.

[261] G. S. Sebestyen. *Decision-Making Processes in Pattern Recognition.* Macmillan, New York, 1962.

[262] Jean Serra. *Image Analysis and Mathematical Morphology.* Academic Press, 1982.

[263] Jean Serra. Introduction to mathematical morphology. *Computer Vision, Graphics, and Image Processing*, 35:283–305, 1986.

[264] Jean Serra, editor. *Image Analysis and Mathematical Morphology, Vol.2: Theoretical Advances.* Academic Press, 1988.

[265] L. Shapiro. A structural model of shape. *IEEE Transactions on PAMI*, 2:111–126, 1980.

[266] L. Shapiro and R. Haralick. Decomposition of two-dimensional shapes by graph-theoretic clustering. *IEEE Transactions on PAMI*, 1:10–20, 1979.

[267] L. Shapiro, R. MacDonald, and S. Sternberg. Ordered structural shape matching with primitive extraction by mathematical morphology. *Pattern Recognition*, 20:75–90, 1987.

[268] L. Shapiro, J. Moriarty, R. Haralick, and P. Mulgaonkar. Matching three-dimensional objects using a relational paradigm. *Pattern Recognition*, 17:385–405, 1984.

[269] F. Shih and O. Mitchell. Decomposition of gray scale morphological structuring elements. In *IEEE CS Workshop on Computer Vision 87*, pages 304–306, 1987.

[270] F. Shih and O. Mitchell. Automated fast recognition of arbitrarily shapede objects by image morphology. In *ICCVPR-88*, pages 774–779, 1988.

[271] F. Shih and R. Mitchell. Treshold decomposition of gray-scale morphology into binary morphology. *IEEE Transactions on PAMI*, 11:31–42, 1989.

[272] F. Shih and R. Mitchell. Decomposition of gray-scale morphological structuring elements. *Pattern Recognition*, 24:195–203, 1991.

[273] F. Y. Shih and O. R. Mitchell. A mathematical morphology approach to euclidean distance transformation. *IEEE Transactions on Image Processing*, 1:197–204, 1992.

[274] Frank Y. Shih and Jenlong Moh. Implementing morphological operations using programmable neural networks. *Pattern Recognition*, 25:89–99, 1992.

[275] Frank Y. Shih and Christopher C. Pu. Morphological shape description using geometric spectrum on multidimensional binary images. *Pattern Recognition*, 25:921–927, 1992.

[276] Frank Y. Shih and Hong Wu. Decomposition of geometric-shaped structuring elements using morphological transformations on binary images. *Pattern Recognition*, 25:1097–1106, 1992.

[277] M. Skolnick. Application of morphological transformations to the analysis of two-dimensional electrophoretic gels of biological materials. *Computer Vision, Graphics, and Image Processing*, 35:306–332, 1986.

[278] S. F. Smith. Adaptive learning systems. In R.Forsyth, editor, *Expert Systems - Principles and Case Studies*. Associated Book Publishers Ltd., 1984.

[279] P. Soille and L. Vincent. Determining watersheds in digital pictures via flooding simulations. In *SPIE-1360*, pages 240–250, 1990.

[280] J. Song and E. Delp. The analysis of morphological filters with multiple structuring elements. *Computer Vision, Graphics, and Image Processing*, 50:308–328, 1990.

[281] S. Sternberg. Grayscale morphology. *Computer Vision, Graphics, and Image Processing*, 35:333–355, 1986.

[282] S. Stevens. *Patterns in Nature*. Atlantic-Little Brown Books, 1974.

[283] Stoyan, Kendall, and Mecke. *Stohastic Geometry and its Applications*. John Wiley, 1987.

[284] A. Taza and C. Suen. Discrimination of planar shapes using shape matrices. *IEEE Transactions on SMC*, 19:1281–1289, 1989.

[285] M. R. Teague. Image analysis via the general theory of moments. *Journal of the Optical Society of America*, 70:920–930, 1980.

[286] J. Tenenbaum, M. Fischler, and H. Barrow. Scene modeling: A structural basis for image description. *Computer Graphics and Image Processing*, 12:407–425, 1980.

[287] D'Arcy Thomson. *On Growth and Form.* Cambridge University Press, 1942.

[288] P. E. Trahanias. Binary shape recognition using the morphological skeletom transform. *Pattern Recognition*, 25:1277–1288, 1992.

[289] L. Uhr, editor. *Pattern Recognition:Theory, Experiments, Computer Simulations, and Dynamical Models of Form, Perception, and Discovery.* Wiley, 1966.

[290] F. Ulupinar and R. Nevatia. Inferring shape from contour for curved surfaces. In *Proceedings of the Int'l Conference on Pattern Recognition*, pages 147–154, 1990.

[291] L. Vanderheydt, F. Dom, and H. Van den Berghe. Two-dimensional shape decomposition using fuzzy subset theory applied to automated chromosome analysis. *Pattern Recognition*, 13:147–157, 1981.

[292] L. Vincent. Morphological transformations of binary images with arbitrary structuring elements. *Signal Processing*, 22:3–23, 1991.

[293] L. Vincent and P. Soille. Watersheds in digital spaces: An efficient algorithm based on immersion simulations. *IEEE Transactions on PAMI*, 13:583–598, 1991.

[294] Luc Vincent. Morphological grayscale reconstruction in image analysis: Applications and efficient algorithms. *IEEE Transactions on Image Procesing*, 2:176–201, 1993.

[295] R. Vogt. Automatic generation of simple morphological algorithms. In *ICCVPR-88*, pages 760–765, 1988.

[296] T. Wallace and P. Wintz. An efficient three-dimensional aircraft recognition algorithm using normalized fourier descriptors. *Computer Graphics and Image Processing*, 13:99–126, 1980.

[297] Wathen-Dunn, editor. *Models for the Perception of Speech and Visual Forms*. MIT, 1967.

[298] *Webster's Collegiate Dictionary, 9th ed.*, 1991.

[299] I. Weiss. Geometric invariants and object recognition. *International Journal of Computer Vision*, 10:207–231, 1993.

[300] M. Werman and S. Peleg. Min-max operators in texture analysis. *IEEE Transactions on PAMI*, 7:730–733, 1985.

[301] M. Wertheimer. Laws of organization in perceptual forms. In W. D. Ellis, editor, *A Source Book of Gestalt Psychology*. Harcourt Brace Jovanovic, 1923.

[302] S. W. Wilson. Machine learning of visual recognition using genetic algorithms. In J. Grefenstette, editor, *Proceedings of an International Conference on Genetic Algorithms and Their Applications*, pages 188–196. Pittsburgh, PA: The Robotics Institute of Carnegie-Mellon University, 1985.

[303] A. P. Witkin. Scale-space filtering. In *Proceedings of the 8th Int'l Joint Conference on Artificial Intelligence*, pages 1019–1022, 1983.

[304] E. K. Wong. Model matching in robot vision by subgraph isomorphism. *Pattern Recognition*, 25:287–303, 1992.

[305] Eugene Wong and Bruce Hajek. *Stohastic Processes in Engineering Systems.* Springer-Verlag, 1985.

[306] J. Wu and J. Leou. New polygonal approximation schemes for object shape representation. *Pattern Recognition*, 26:471–484, 1993.

[307] J. Xu. Decomposition of convex polygonal morphological structuring elements into neighborhood subsets. *IEEE Transactions on PAMI*, 13:153–162, 1991.

[308] Young and Fu, editors. *Handbook of Pattern Recognition and IP.* Associated Press, 1986.

[309] I. Young, J. Walker, and J. Bowie. An analysis technique for biological shape i. *Computer Graphics and Image Processing*, 25:357–370, 1974.

[310] L. Zadeh. Fuzzy sets. *Information and Control*, 8:338–353, 1965.

[311] L. Zadeh. Outline of a new approach to the analysis of complex systems and decision processes. *IEEE Transactions on SMC*, 3:28–44, 1973.

[312] C. Zahn and R. Roskies. Fourier descriptors for plane closed curves. *Computer Graphics and Image Processing*, 21:269–281, 1972.

[313] D. Zhao and D. Daut. Automated shape recognition in noisy environments based on morphological algorithms. In *SPIE-1295 Real-Time Image Processing II*, page 1990, 223-234.

[314] Y. Zhao and R. Haralick. Binary shape recognition based on an automatic morphological shape decomposition. In *ICASSP-89*, pages 1691–1694, 1989.

[315] X. Zhuang and R. Haralick. Morphological structuring element decomposition. *Computer Vision, Graphics, and Image Processing*, 35:370–382, 1986.

[316] S. F. Zornetzer, J. L. Davis, and C. Lau, editors. *An Introduction to Neural and Electronic Networks*. Academic Press, 1990.

[317] Zusne. *Visual Perception of Form*. Associated Press, 1970.

[318] L. Zusne. *Contemporary Theory of Visual Form Perception: III. The Global Theories*, chapter 4, pages 108–174. Associated Press, 1970.