SOME STUDIES ON SHAPE OF DOT PATTERNS

Anirban Ray Chaudhuri

INDIAN STATISTICAL INSTITUTE
203 BARRACKPORE TRUNK ROAD
CALCUTTA 700 035 INDIA
1998
(Revised 1999)

To My Parents and Teachers

Acknowledgment

I owe Prof. B. B. Chaudhuri, my supervisor, a tremendous debt. His advice, guidance, affection and his insistence on being meticulous in different aspects helped me in making my research objectives clear and keeping myself motivated throughout my tenure as a research fellow. His ideas in the subject left a long lasting influence on my own thinking.

There are hardly any words to express my gratitude to Dr. Ayan Basu who helped me in understanding and revealing the statistical aspect of our work. I thank Dr. Subir Bhandari for providing counter examples to my earlier and less mature hypothesis. I would like to thank Prof. Swapan Parui and Prof. Bhabatosh Chanda for introducing me to the topic of dot pattern analysis and mathematical morphology respectively and for many a helpful suggestions.

I am grateful to Prof. D.Dutta Majumder, Prof. J. Roy, Prof. J. K. Ghosh, Prof. S. B. Rao and Prof. J. Das for their encouragement, support during the course of this work.

I express my gratitude to Prof. K. V. Mardia, University of Leeds and Prof. Vi di Gesu, University of Palermo for discussions I had with them during their visits to our Institute.

I was benefited from knowledge, help, and above all friendship from many people. I thank Dr. Dipti Mukherjee, Dr. S. Vasulu, Dr. Probal Sengupta, Dr. Umapada Pal, Swapan Raha, Ujjwal Bhattacharya, Amitava Datta, Tamaltaru Pal, Utpal Garai, and Umesh Adiga for all the help they have extended.

I would like to take this opportunity to express my thanks to all staff members of Deans office, Reprography Unit and Library of the Institute for their regular services and assistance. I wish to thank one and all members of the CVPR Unit for the nice environment they provided for the work.

I acknowledge Council of Scientific and Industrial Research, CSIR for the research fellowship throughout my tenure as a research fellow at the Institute.

I convey my gratitude to my parents, Mr. Riten and Mrs. Rebecca Ray Chaudhuri, my brother Aniruddha for their affection and encouragement; especially, my grandmother Mrs. Bina Roy for her support and tolerance during my stay in Calcutta between 92-98.

During the tenure of the thesis revision –

I am thankful to all my colleagues at the Department of Computer & System Sciences and Computer Center of Visva-Bharati, Santiniketan.

I am grateful to Prof. Narendra Ahuja, University of Illinois at Urbana Champaign for his valuable suggestions regarding the applicability of the work and offering me a fellowship for further research at the Beckman Institute from the beginning of the coming millenium.

Finally, I owe my wife Debanjali, for all her patience, sacrifice and care that gives me strength to continue work on the old problem.

Anirban Ray Chaudhuri.

Lecturer in Computer Science,
Department of Computer & System Sciences
Visva-Bharati (A Central University)
Santiniketan. 731235 India.

CONTENTS

LIST OF FIGURES

Chapter 1.

Introduction

The important visual characteristics of an object are shape, size, color, brightness, contrast and texture. Of them, shape is a multi-dimensional concept that is difficult to define. It takes different meanings in different contexts. We try to explain it in terms of their attributes like elongation, roundness, and symmetry; although these terms do not capture the complete notion of shape.

Perhaps Gestalt theory [Koffka 35] is the first attempt to study the principles of visual perception in a systematic manner. The central concept of this theory is 'Gestalt' which means form or configuration. In this theory form is examined from physical, psychological and behavioral points of view. Some basic Gestalt principles are discussed in the next section. These principles help us to understand the notion of shape.

The introduction of information theory provided another tool for the study of shape. Attneave and others showed how and where the information in the shape of a particular object is concentrated [Attneave 59]. Gradually, quantitative approaches of evaluation of shape property were evolved with the advent of digital computers. Controlled synthesis of shape and forms were also considered side by side for graphical applications. An overview of shape analysis is presented in Section 1.1.

Our thesis concerns the *shape of dot patterns*[*]. The dot pattern shape is introduced in Section 1.2 where different approaches of shape computation are surveyed. The shape detection of dot pattern may also be viewed as a *set estimation* problem. In that case the *consistency* criterion become an important issue. In this thesis we have considered the set estimator aspect as well. Moreover, the *clustering* problem is dealt with for the shape extraction from complex point pattern.

The scope and layout of the thesis are presented in Section 1.3.

---

[*] We shall interchangably call point sets as point patterns or dot patterns.

1.1 Overview of Shape Analysis

The Gestalt approach of shape analysis started in 1912 when Wertheimer published a paper on visual illusion called *apparent motion* [Wertheimer 38]. To explain how perception of individual objects is formed, Wertheimer proposed that the visual system organize parts into wholes based on laws of grouping. Thereafter, the theory grew for about three decades. More than one hundred principles are found in the Gestalt literature [Koffka 35], [Zusne 70], [Rock & Palmer 90]. Of them, a few important principles are discussed below.

In 1915, Rubin pointed out that all visual forms possess at least two distinguishable aspects called figure and *background* [Rubin 15]. The judgment of figure and background is guided primarily by some contrasting properties. The figure has contours and distinguishable parts and it has a 'thing' like quality [Zusne 70]. The background, on the other hand, is formless, diffuse and appears to be more distant than the figure. When the strengths of these properties are nearly equal in the figure and background portion of the image, the resulting percept becomes ambiguous. A well-known example is Rubin's vase that may be seen as a flower vase or two human faces in front of each other. This *figure-ground principle* is applicable to all visual stimuli.

Another important principle is that of proximity – patterns having smaller distance among themselves tend to group together compared to the patterns separated by larger distances. Similarly, the *principle of similarity* states that similar patterns tend to form groups to our perception. Moreover the *principle of good continuation* conveys that if there are several ways of interpreting a pattern, the simplest and most regular way is chosen in visual perception. The above three configuration properties are not independent and may influence one another in the precept of a pattern.

The *transposition principle* is another important principle of visual perception, states that the shape of a pattern remains unchanged under few transformations namely, translation, rotation, scaling and reflection. This principle helps us define the shape of an object [Parui 84]. However, rotation and reflection can change the meaning of a figure. For example, the character '6' under 180° rotation appears as the character '9' while the mirror image of 'd' about the vertical axis is 'b'. Nevertheless, this principle is useful in computer recognition and classification of figures.

There exist other configuration properties such as *symmetry* and *closure*. The principle of symmetry states that patterns, which are symmetrically organized, tend to form groups while due to the principle of closure, the closed figures tend to be seen as units. Another notable principle is the *principle of compensation*, which states that change in one part of a visual form affects the perception of the whole form. The simpler the form the greater is the effect. This principle indicates that changes in more busy areas of the picture are more tolerated than changes in the less busy areas. Thus, during image enhancement, noise cleaning and data compression more care should be taken for the less busy areas of the image.

The Gestaltists have demonstrated through experiments that information in the two dimensional shape is contained in its contour; especially at any point of change in the gradient in the contour, i.e. vertices, corners and curves [Attneave 59]. For computer analysis of shape, the algorithms to detect these points are therefore useful and important.

One interesting area of study in visual perception is that of geometric illusions. There exist several theories of illusion that may be broadly classified as being psychological, judgmental or based on information sampling ideas [Over 68]. However, it is still not clear how the interaction in the visual field changes the percept of size and shape of objects in creating the illusions.

The Gestalt principles have a lasting influence on the shape analysis and recognition research of future generations. While Gestalt theorists tried to formalize visual perception of shape and form, several practical application oriented problems related to shape were attempted. Some of the earliest among them are optical character recognition, fingerprint processing and analysis of biomedical signals like ECG. Important breakthroughs were attained in these areas much before the introduction of digital computer. For example, character recognition was attempted as early as in 1900 by the Russian scientist Tyurin while Tausheck obtained a patent on character recognition system in 1929 [Mori et. al. 92].

However, work on pattern recognition in general and shape analysis in particular, received the strongest boost with the introduction of digital computer. Related disciplines like *image processing* and *computer vision* started to evolve with the need of enhancing the quality of remote sensing photographs including those sent from spacecraft. The work was extended to 3-D information extraction that led to the unfolding of 3-D object shape

properties and invariants from one or more pictures. In fact, finding 3-D shape is one of the major branches of computer vision. Shape extraction using stereo, shading, texture, etc. are well studied in literature [Kanade 81], [Horn 86], [Faugeras 93].

Another useful and important discipline that evolved deals with the synthesis of images by computer. The discipline, called *computer graphics* [Foley et. al. 90] has revolutionized the design and simulation process in heavy industry, among others. It paved the way for more recent discipline called *virtual reality*. Image synthesis is not our concern in this thesis and hence we do not review it further.

Several important and useful concepts and approaches about shape analysis were proposed in 1960s and 1970s [Pavlidis 78]. For example, in 1961, Freeman [Freeman 61] proposed a simple and effective representation of object border in the form of chain code. In its simplest form, *Freeman's chain code* is a three-bit number to represent the direction along which one can move from one pixel to the next along the border. Numerous shape recognition and data compression approaches were proposed on chain coded representation of objects.

Hough in 1962 proposed a novel approach of detecting lines in 2-D images [Hough 62]. The approach, called *Hough transform* has since been generalized to the detection of other shapes and used to many applied problems [Illingworth & Kittler 88]. On the other hand, in 1962 an important concept called the *medial axis* was proposed by Blum [Blum 64]. It is useful for vector representation, stroke feature detection, data compression and recognition of an object. In digital domain, many approximations to the medial axis have been proposed and the result is called the *skeleton* of the object. One class of approaches of skeletonization is called *thinning* [Jang & Chin 90], [Lam et.al. 92] while the other class is based on *distance transform* [Ragnemalm 93].

Many orthogonal transformations and series-transformation based features were proposed for shape description and recognition [Pavlidis 80], [Gonzalez&Woods 93]. Some of them are Fourier transform, Walsh-Hadamard and Haar series expansion, Karhunen Loeve expansion. In their original form, these features are not rotation and scale invariant. Hence some form of normalization is done to make them insensitive to rotation and scaling. Among other similar features, normalized moments attracted due attention [Hu 62],[Teh 88].

Depending on the problem, geometrical and topological features are used for shape recognition. Some of the features are area, perimeter, slope-curvature, elongatedness, compactness, convexity, straightness, end points, intersection points, number of holes, Euler number [Ballard & Brown 82], [Rosenfeld & Kak 82], [Jain 89].

Shape recognition approaches can be classified into a few broad categories namely, statistical, deterministic, syntactic, fuzzy set theoretic and morphological approaches. Of them, deterministic approaches employ classical inference rules while statistical approaches use probability theory in the decision process [Duda & Hart 73]. A syntactic approach employs the formal language grammar description and recognition. For example, *web grammar* was proposed by Pfaltz and Rosenfeld in 1969 for image modeling and description of inter-relation in objects [Pfalz & Rosenfeld 69]. Among others, graph grammars as well as attributed grammars may be mentioned. For a complete description see [Fu 74].

Fuzzy set theoretic approach was introduced by L. Zadeh in 1965 [Zadeh 65]. Here, an object or its shape properties are considered as *fuzzy sets*. A fuzzy set is a set whose constituents are characterized by membership values in [0,1] that denote the degree of belongings to the set. In this framework, the classifier decision can also be made soft, instead of crisp [Rosenfeld 84], [Pal & Majumder 86], [Chaudhuri 91].

The theory of mathematical morphology was formally introduced by Matheron in 1975 [Matheron 75] and later developed for binary as well as gray scale image analysis by Serra and others [Serra 82], [Haralick et.al. 87], [Sternberg 86], [Dougherty 94]. The language of classical (binary) mathematical morphology is set theory. Central to the idea of mathematical morphology is the *structuring element*, which is defined in the same (norm) space of the input pattern about a (local) center of reference, and its structure or shape is problem dependent. A morphological operator probes over the input pattern space (the input image) using the structuring element (template). The output of the operator depends on the geometric similarity between the structuring element (globally) and the object (locally). Most of our proposed approaches as well as algorithms are morphological in nature. The (binary) morphological operators that are used later in the thesis are defined in the Appendix.

Two more recent classes of techniques are based on artificial neural network and genetic algorithms. In artificial neural network, attempt is made to simulate human neural process of memorizing and inferencing by a set of interconnected simple processors called

artificial neural nodes [Carpenter & Grossberg 92], [Sabourin & Mitiche 93], [Datta et. al. 97]. Genetic algorithms exploit the process of genetic evolution by simulating the mutation, crossover and selection of best fitting chromosomes [Hill & Taylor 92], [Watabe & Okino 93].

Shape analysis is a very active area of research. For more comprehensive reviews on shape coding techniques, surveys by Pavlidis [Pavlidis 78], [Pavlidis 80] and more recent one by Marshall [Marshall 89] may be noted. A huge number of papers are being published regularly on shape analysis in different journals and conference proceedings. Among the journals – IEEE Trans. Pattern Recognition and Artificial Intelligence, IEEE Trans. Systems Man and Cybernatics, CVGIP Image Understanding, Pattern Recognition, Pattern Recognition Letters, ACM Computing Surveys, Image and Vision Computing, Signal Processing, etc., contain papers on shape analysis. Among the conference proceedings – Int. Conference on Pattern Recognition, Int. Conference on Computer Vision, Int. Conference on Computer Vision and Pattern Recognition may be referred.

1.2 Dot Pattern and its Shape

We often come across minute objects in a real scene appeared in the form of dot pattern. In almost all application sciences we encounter such patterns. Stars in the galaxy (Astronomy), cells or micro-organisms in the tissue (Biology), micro structures of impurities in metal-compound (Chemistry), X-ray dot maps (Particle Physics) are few such examples.

The other type of dot pattern available for computer analysis is not a direct image-like structure. Rather, a dot represents the feature vector of a physical object or a physical phenomenon [Duda & Hart 73].

Depending on the problem, for both types of patterns, it is necessary to discover the structure of the point set in the form of clusters [Jain & Dubes 88], directionality [Mardia 72], intrinsic dimensionality [Verveer & Duin 95], etc. Registration and matching of point sets are also needed in some problems [Ranade & Rosenfeld 80], [Ahuja & Charan 95].

Another problem of interest, which is more relevant to the first type of dot patterns, is the main concern of this thesis. That is to find the shape of the dot pattern. It involves detection of perceptual border as well as the skeletal representation. It is essentially a *low-*

level description, independent of any final conceptually driven description. According to Marr, such a description is called a *primal sketch* [Marr 82].

A human observer is quick to understand the *perceptual boundary* of the point pattern where points are clearly visible as well as fairly densely and homogeneously distributed over a planar region. Here perceptual boundary refers to a boundary which observer would perceive as a function of the relative location of the dots, under normal (*fronto-parallel*) viewing conditions, and without assigning any semantic or cognitive interpretations to the pattern. Although the perceptual boundaries are subjective, observers generally agree on the shape of boundaries for homogenous (at least locally) dot patterns which, in a Gestalt sense, cluster on the basis of proximity.

On the other hand, if the dot pattern is elongated or curvilinear then we perceive a linear form in which a notion of skeleton can be conceived.

With the advent of digital computers, the interest in the task of automated computing of perceptual shape of dot patterns arises from a continuing effort to simulate aspects of human visual performance. Moreover these methods can be utilized in some real life problems.

For example, the spatial density measure of certain cell-organelles in a tissue section is an essential part in pathological test. Moreover, shape and size of important organelles have to be studied individually. In some case the boundary of such an organelle is not distinct from the background but is surrounded by tiny organelles. Then, the *inner-border* (*hole*) of the representative dot pattern of those tiny organelles may effectively demarcate the boundary of the observed one. (An illustration on this problem is presented in Section 3.5.2.1)

In metallurgy, the density of impurities in the host metal determines the quality of an alloy. On the other hand, the spatial distribution of stars as well as its density are important towards finding the age and size of the galaxy. Note that in all these microstructure related density measure problems, the number of occurrence of these structures should be counted over the *region of support* than over the whole visual field. Otherwise, the estimated density may become low with respect to the true density. So, at first it is necessary to extract the overall border of the micro-structures. Next, to estimate the density, the number

of micro-structures may be counted and divided by the area enclosed within that extracted border. Several other identical problems of region estimation like sea-bed scanning for natural resources (remote sensing), land-mine detection (military reconnaissance) etc., may be encountered.

Since one of the most effective ways to estimate the region of support is to find the shape of the representative dot patterns, the problem considered in the thesis has true potential application to practical problems. However, in this thesis except one illustration in cytology as mentioned above, we have concentrated only towards developing methodologies.

It should be noted that the theories and methods involved in shape computation of dot patterns are shared with that in *spatial analysis*, the subject that examines the spatial structure and association of phenomena [Riplay 81], [Ahuja & Schachter 83], [Diggle 83],[Stoyan et.al. 87]. Geographic analysis [Taylor 77], [Matula & Sokal 84], remote sensing [Richards 86], astronomy [Geller & Huchra 89], [Barrow 92], spatial information system [Laurini & Thompson 92], biomedical imaging [Pernus 88], [Choi 96], and defense applications are some but not the only areas where structural analysis as well as studies on spatial distribution of dot patterns are useful.

In the following section we present a survey of dot pattern analysis.

1.2.1 Studies on analysis and shape computation of dot patterns

Two aspects of shape of a dot pattern are of interest, which may be called *external* shape and *internal* shape. The external shape descriptor tries to capture the perceptual border while the internal shape descriptor tries to capture the core of that underlying region, something like a skeleton.

Among early studies on automated detection of border of non-linear components, the work of O'Callaghan should be mentioned [O'Callaghan 74]. His approach is based on a generalized definition of neighborhood. Classically, neighbors are defined on the basis of Euclidean distance. Such neighbors are called *nearest neighbors*. O'Callaghan combined the 'distance' as well as 'behindness' in his definition. Thus, if P is a chosen neighbor of a candidate point O and if among Q and R, Q is 'behind' P then R will be preferred as the next neighbor of O. Using this definition and a few other heuristic measures he could find the boundary points of a dot pattern.

O'Callaghan's neighborhood is dependent on some ad hoc thresholds chosen apriori. Chaudhuri proposed a new definition called *nearest centroid neighborhood* [Chaudhuri 94] that is free from any threshold and yet captures the notion of proximity and surroundedness. Here, the neighbors are sequentially chosen such that the centroid of neighbors is nearest to the candidate point. *Shared nearest neighbors* [Jarvis & Patrick 71] and *mutual nearest neighbors* [Gowda & Krishna 78] are other examples of neighbors, which are used in classification and clustering problem.

Graph-theoretic approaches have been used to determine the shape of dot patterns. For example, one can draw the *Minimum Spanning Tree* (*MST*) which is a tree constructed by connecting all points in the pattern so that the total edge length is minimised. Then the nodes with single edge may be considered as those belonging to the border. For early work on MST related approach, see the celebrated work of Zahn [Zahn 71].

Various proximity graphs can be generated on the dot pattern by choosing various definitions of neighborhood. For earlier surveys see [Radke88],[Vincent 89], [Jaromczik&Toussiant 92].

Relative neighborhood graph (*RNG*) is one such proximity graph proposed by Toussiant [Toussaint 80]. Two points P and Q in the pattern are relative neighbors if the distance between P and Q is smaller than larger of the distances of any other point R from P and from Q. The *Gabriel graph* (*GG*) proposed by Gabriel and Sokal [Gabriel & Sokal 69] is another graph where the neighborhood relation is stronger than RNG. Two points P and Q are Gabriel neighbors if $\angle PRQ$ is not an obtuse angle for any other data point R in the pattern. In this connection two parameterised family of neighborhood graphs, namely β-*graph* [Krikpatrick & Radke 85] and γ-*graph* [Veltkamp 89] may be mentioned. These graphs contain RNG and GG as special cases [Jaromczik & Toussiant 92].

A unique shape description of dot pattern can be its convex hull, which is the smallest convex polygon to contain the dot pattern. On computation of convex hull see [Akl 85], [Ronse 89]. However, it is a poor shape descriptor if the dot pattern contains concavity. To overcome the limitation of patterns with concavity, Jarvis suggests decomposition of the pattern into subpatterns [Jarvis 77]. The union of convex hulls of resultant subpatterns represents the shape of the whole pattern. Recently, a similar approach is proposed by

Garai and Chaudhuri [Garai & Chaudhuri 98], which starts from the convex hull and converges to a polygonal shape by split and merge procedure.

An alternative definition of the boundary based on the concept of density is attempted by Medek [Medek 81] where the connectivity in regular lattice [Rosenfeld 70] is generalized. Other approaches of border point detection by density based methods are also available. See for example, [Chaudhuri & Chaudhuri 97].

Voronoi diagram (or *tessellation*) and its dual *Delaunay triangulation* were used by many researchers in shape analysis of dot patterns [Ahuja 82], [Edelsbrunner et. al. 83], [Aurenham 91], [Bertin et. al. 92], [Faugeras 93]. In 2-D, Voronoi tessellation partitions the space by convex polygons such that only one data point P is situated in one polygon and any other point Z within the polygon is nearer to P than any other data point. The tessellation induces a kind of neighborhood in the data space, called *Voronoi neighborhood*. Two data points P and Q are Voronoi neighbors if the corresponding polygons have at least one common edge. On the other hand, common edges of Voronoi neighbors constitute the Voronoi diagram. If the Voronoi neighbors are joined by lines then the resulting graph is called the Delaunay triangulation.

Perhaps the first work on external shape computation using Voronoi diagram was made by Fairfield [Fairfield 79]. Ahuja and his associates made more elaborate use of the approach to dot pattern analysis [Ahuja 82], [Ahuja & Tuceryan 89]. Shape properties and geometric attributes such as area, compactness, elongatedness, eccentricity are computed through Voronoi neighbors. A perceptual grouping is accomplished and correction of groupings are performed through constrained propagation using probabilistic relaxation process. The work was also extended to detect 3-D curves [Charan & Ahuja 95].

A popular external shape descriptor of dot patterns called α-*shape* (*graph*) with a sound mathematical basis was proposed by Edelsbrunner et. al. [Edelsbrunner et. al. 83]. The α-graph is a uniparametric subgraph of Delaunay triangulation. This graph is in fact derived from a natural generalisation of convex hull, called α-*hull* (for the formal definition and derivation see Chapter 3). The vertices (called α-*extremes*) and edges of the α-graph (called α-*edges*) for a given α can be characterised as follows. For every point P in the pattern set there exists a real number $\alpha_{max}(P)$ such that P becomes a vertex of the α-graph if and only if $\alpha \leq \alpha_{max}(P)$. On the other hand, for every edge e in the Delaunay triangulation there are

real numbers $\alpha_{min}(e)$ and $\alpha_{max}(e)$ such that e is an edge of the α-graph if $\alpha_{min}(e) \leq e \leq \alpha_{max}(e)$ and vice versa.

Some algorithms on α-graph in digital case [Parui et.al. 93], [Satio et. al. 91], [Pernus 88] as well as modification of α-graph like directed α-graph (which is defined over a connected set as well) were proposed [Worring & Smeulders 94]. Recently 3-dimensional extension of α-shape was also reported [Edelsbrunner & Mücke 94].

Conventional α-shape is not adequate if the dots are of not evenly distributed. To overcome this shortcoming, Edelsbrunner put forward a theory of *weighted α-shape* which is intuitively less appealing than the classical definition [Edelsbrunner 92]. Another practical difficulty is the lack of a automatic approach of finding appropriate α for a given dot pattern.

Toussiant proposed an external shape descriptor using the *sphere of influence graph* (*SIG*). Consider circle at each point of the dot pattern with radius equal to the distance of its nearest neighbor. P and Q are joined in the SIG if their circles intersect at least at two points. A properly chosen subgraph of SIG called *Sphere of influence shape* denotes the border of the pattern [Toussaint 88]. The main advantage of Toussiant's proposed extractor is that it is free from parameters. Moreover, it is simple and computationally attractive [Bently & Ottman 79]. However, since SIG considers only the nearest neighborhood distance and the point-wise circular influence zone is highly localized to its nearest neighbor, the result often does not agree with the perceived border if the dot pattern is dense [Toussaint 88].

Chaudhuri presented a different approach of external shape computation of dot patterns by an application of quadtree [Chaudhuri 85]. In this case a simple hierarchical partitioning of the space was made for a partitioning factor of 4 and the smallest square covering the dot pattern was decomposed into 4^i equal squares grids at the i-th level. Subsequently, at the i-th level the border of the dot pattern was interpreted as the border of the connected components in of non-empty grids. Our first shape descriptor presented in Chapter 2 is another quadrate method which is coupled with a stopping criteria that was not provided with the quadtree based algorithm.

There appears to have no standard framework or features based on which various dot pattern shape descriptors can be evaluated. However, we have attempted to make an

evaluation on methods of perceptual border extraction. The following six criteria are considered.

(a) *Type of data considered:* In a regular dense dot pattern the perceptual border is clearly recognized. However, the shape (border) of the pattern may be perceived with variation in density, with multiple components as well as in presence of additive 'noise' (especially, in the boundary region). Thus, a descriptor is more useful if it can also recover the perceptual border of a 'complex' dot pattern.

(b) *Output:* A method that provides 'smooth' border without any edge 'inconsistent' with the perceived shape is a good one.

(c) *Computational complexity:* Since now-a-days memory is affordable, the computational complexity of the algorithm is restricted to run time complexity only. Moreover, any algorithm that allows easy implementation is favored over others.

(d) *No of parameters/thresholds:* The lesser the number, the better is the method.

(e) *Automatic estimation:* Any method is highly desirable where the required parameters or thresholds are automatically selected.

(f) *Set consistency of the descriptor:* If a shape descriptor is also a consistent set estimator then it is a desirable additional property. For a consistent set estimator, it is statistically established that the region covered by the border generated by that descriptor gradually converges to the original region with increase in the number of dots.

In the following table we have attempted to give a bird's eye view on 6 external 2-D dot pattern shape descriptors that are comparable with our proposed 3 descriptors. (Any method that is an up-gradation without contributing any additional criterion or, has been considerably modified by another method is not separately mentioned in the table.)

	Kind of data considered	Output	Average Time Complexity Analysis	No of parameters/ Thresholds	Automatic Estimation	Set Consistency Aspect	General comments on the Method
Angle & Directional Continuity Based Approach [O'Callaghan'74]	Homogeneous noise free patterns with single component	Generated borders are perceptually compatible with perceived ones	Not Analyzed	5 thresholds	Values of thresholds are justified	Not Considered	Noted as an earliest computer based algorithm for border extraction
α-shape [Edelsbrunner et.al. '83]	Homogeneous noise free patterns with single component	Subject to the right choice of α the output is compatible to the perceived border.	$O(n \log n)$	1 parameter	Not Discussed	Partially covered in [Worring & Smeulders '94]	Analytically sound and the inherent idea is somewhat dual of intuitive one
Voronoi Neighbor Based Approach [Ahuja & Tuceryan'89]	Multi-component and varying density even within a component	Perceptual border is correctly extracted.	$O(n \log n)$	5 thresholds	Values of thresholds are justified	Not Considered	Perceptual aspects are extensively studied but the overall method is complicated.
Quad Tree Based Approach [Chaudhuri '85]	Homogeneous noise free patterns with single component	Border consists of only horizontal and vertical edges; compatible with the perceived shape under correctly chosen number of iteration	$O(n)$	1 parameter	Not Discussed	Not Considered	The algorithm is intuitively simple.
Sphere of Influence Graph [Toussiant '88]	Patterns with various shapes and densities	Results are good for low density patterns but for high density, the border with too many inconsistent edges is not compatible to the perceived border.	$O(n \log n)$	The method is free of parameters/ thresholds	—	Not Considered	Intuitively very simple. Very easy to implement
Minimum Spanning Tree Based Approach [Murthy '88]	The number of components to be supplied	No result on dot pattern is shown.	$O(n \log n)$	1 parameter	Not Discussed	Consistent estimator	Intuitively appealing but lack of result makes it difficult to evaluate
***s*-shape** [Ray Chaudhuri et.al. '97]	Multi-connected homogeneous patterns with varying density	Border consists of only horizontal and vertical edges; compatible to the perceived shape.	$O(n)$	1 parameter	Is proposed	Consistency maintained even under moderate noise	Intuitively simple. Very easy to implement
***r*-shape** [Ray Chaudhuri et.al.97]	Multi-connected homogeneous patterns with varying density	Perceptual border is correctly extracted	$O(n)$	1 parameter	Is proposed	Not Considered	Intuitively simple. Easy to implement
Generalized *r*-shape [Ray Chaudhuri &Chaudhuri97]	Multi-component and varying density even within a component	Perceptual border is correctly extracted	$O(n \log n)$	2 parameters	Not proposed	Not Considered	Easier to implement than Voronoi neighbor based approach

Figure 1.1 *A table showing comparative study of various dot pattern border descriptors*

The computation of internal shape (like skeleton) has received relatively less attention compared to the external shape. Zahn [Zahn 71] presented a novel MST based skeleton representation that conformed to Gestalt principles of perceptual organization for elongated patterns. Here, the longest path in the MST was found and a measure of diversity for each node in the middle half of this path was computed. The edge connected to the node with minimum density was removed. Further every edge whose length greater than the user-specified threshold were dropped out from the tree. Finally, the pruned path depicted a kind of skeleton of the dot pattern.

For curvi-linear patterns, RNG and SIG can be used for internal shape computation [Toussaint 80], [Toussaint 88]. The Voronoi diagram and Delaunay triangulation based skeletonization proposed respectively by Ogniewicz [Ogniewicz 91] and Faugeras [Faugeras 93] are also applicable with some preprocessing. (See Chapter 4.)

Recently, a new self-organizing neural network based approach on dot pattern skeletonization was presented by Datta and Parui [Datta & Parui 97]. The algorithm used Kohonen's self-organizing feature map principle [Kohonen 89] with few modifications of the original Kohonen's model. The network was dynamic in the sense that processors can be inserted or merged during the adaptive process. The number of processors need not be known apriori and the net can learn the topological relationship from the input pattern enabling the network to tend to a desirable approximation of the skeletal shape of the dot pattern.

Clustering is a popular approach in pattern classification and taxonomical grouping [Kendall 66]. Its' primary objective is to partition the point set into subsets that maximize both the 'similarity' among members of the same class, and 'dissimilarity' across classes [Anderberg 73]. The various algorithms which have been proposed differ as to what constitutes a better partition and what methods may be used for achieving improvements and have been grouped in some categories.

The problem of clustering arises in point pattern shape analysis if several subpatterns exist in the space and the isolation of 'component' becomes necessary. From that point of view, the clustering may be treated as a subproblem of shape detection of complex point patterns. There are quite a few interesting clustering methods motivated by the notion of shape (perceptual consideration) or the spatial structure of the data. According to Jain and

Dubes [Jain & Dubes 88], these techniques mostly belong to the category of *exclusive*, *unsupervised* clustering. Exclusive unsupervised clustering is subdivided into *hierarchical* and *partitional* clustering by the type of structure imposed on the data. A hierarchical clustering is a nested sequence of partitions, whereas a partitional clustering is using a single partition. For an extensive review of clustering algorithms see [Jain & Dubes 88].

Clustering based on Voronoi diagram and Delaunay triangulation were also attempted [Ahuja & Tuceryan 89] [Rosenfeld & Jolion 88], [Okabe et. al. 92] but such approaches should not be used for higher dimensional data because of computational complexity. The other proximity graphs described earlier have applications in clustering. Among these, several earlier attempts were made with nearest neighbor [Cover & Hart 67], shared nearest neighbors [Jarvis & Patrick 71], mutual nearest neighbors [Gowda & Krishna 78], k-nearest neighbors [Mizo & Shimura 80], Gabriel graph and RNG [Toussaint 80], [Urquhart 82].

Zahn suggested several heuristic tactics (the most important one has been already described in the context of computation of internal shape) for uncovering *inconsistent* edges in a MST whose removal can generate natural clusters. [Zahn 71]. Being more economical than other structural bases in higher dimensions, MST is a very popular tool in clustering [Murthy 88], [Chaudhuri 95a], [Chaudhuri & Chaudhuri 97]. In this connection, an interesting clustering approach based on the skeleton of influence zone may also be mentioned [Herbin et. al. 96].

Few researchers also studied clustering based on decomposition of the feature space by hyper-cubes. Some earlier attempts can be found in [Taylor 77] and [Warnekar & Krishna 79]. An efficient hierarchical process, which was the generalization of quadtree based decomposition technique described earlier, was presented in [Chaudhuri 85a]. By means of non-linear morphological filters as well as the nearest neighborhood classifier, a clustering approach was proposed by Postaire et. al. [Postaire et. al. 93]. Another recent clustering algorithm called *GRIDCLUS* may be mentioned which used a multidimensional grid data structure [Schikuta96]. In all of these studies, the data structure organizes the space surrounding the pattern, rather than to organize the patterns themselves. These methods are particularly useful for very large data sets.

External shape description of dot pattern may be viewed as an associated problem of set estimation from a finite number of sample points drawn from the set. One major problem

of interest in set estimation is its consistency. A set estimator is consistent if it converges to the original set when the number of points drawn from the original set tends to infinity. Like clustering, set estimation can be done over higher dimensional space. In this thesis we have proposed a class of consistent set estimators that are computationally efficient. One of the earliest consistent set estimators was due to Grenander [Grenander 75]. The estimator has computational inconvenience as it lacks the *scale equivariance* (dependency on the space) property. However, some attempts was made by Mandal et. al., where rectangular neighborhoods were considered instead of circular disks surrounding the sample points [Mandal et. al. 92]. For more detail, see Section 6.1.1.

Another novel approach based on MST was proposed by Murthy [Murthy 88]. This set estimator was the union of circular disks along the MST of the sample points and naturally possesses the scale equivariance property. The consistency of the result remains valid for any continuous distribution. However, the proposed estimator cannot be extended to the case of union of multiple compact regions unless the number of disjoint components is known.

There was an attempt to establish the set consistency of α-graph by Worring and Smeulders [Worring & Smeulders 94]. They established that the α-graph of a connected set converges to the connected set itself but the proof in case of sample points (instead of the original connected set) is not shown. This is somewhat like establishing the *Fisher consistency* in the context of parametric estimation [Cox & Hinkley 74].

Some prior knowledge about the distribution or complexity of the data through the testing of uniformity [Smith&Jain 84],[Banerjee et.al.91] as well as cluster validation [Dubes&Jain 79] etc., are useful for dot pattern analysis [Stoyan et. al. 87], [Jain & Dubes 88], [Bhasker et. al. 88]. For example, if a pattern is found to be uniformly distributed as well as well clustered, then some simple shape extractor may be employed.

Noise is a common phenomenon in any system of observation and inference. Some attempts have been made on clustering over noisy data [Jain & Dubes 88], [Yahil & Brown 76], [Huang & Shieh 90]. But the problem of set estimation with noisy data seems to be untouched.

For the sake of completeness, some work on human perceptual analysis of dot patterns may be mentioned here. Glass, Prazdny and others studied the perceptual process of global

structure identification in *Moire patterns* as a function of the global transformation, such as dilation, rotation or translation [Glass 69], [Prazdny 86], [Kass & Witkin 87].

One-dimensional arrangements of dots appear to group into lines. Uttal et. al. studied how detection of lines formed by dots on a noisy background is affected by inter-dot spacing [Uttal et.al. 70]. Corners are very salient and useful properties of contours. In [Link & Zucker 88], human sensitivity to corners is assessed and a theory to compute them are sketched. Dotted contours are taken as the stimuli in the psycho-visual experiments, which are designed to quantitatively evaluate the human sensitivity to orientation discontinuities. It is found that human sensitivity to corner points is dependent on a rough computation of the order of change in curvature at the point.

Zucker and Davis discovered that dots appear to be organized in a line when they are closer than a critical spacing [Zucker & Davis 88]. Since the spacing depends on the size of the dots (a real dot has always a finite size), they referred this critical value as *size/spacing constraint.* As the density of dots decreases, there is an abrupt point where some percepts change dramatically. A model for these earlier grouping processes was quantitatively consistent with the observed data: the shift takes place for dot to space ratios of 1:4 or, at most, 1:5. It has been claimed that this size/spacing constraint stands in direct opposition to grouping process formalized by Gestaltists.

1.3 Scope and Layout of the Thesis

In this thesis we have concentrated on the development of new algorithms for external shape (i.e., boundary detection) as well as internal shape (skeleton formation) of dot patterns. Our proposed algorithms are such that the complexity of algorithms should grow with the complexity in the scene under observation rather a general-purpose complex algorithm, which may result in wastage of resources. To start with, we have considered that pattern has a single component without any curve-like structure so that the boundary encloses a finite area in which the points are more or less evenly distributed. Later on dot patterns with having more than one bounded region as well as variable densities have been considered. In this connection, the concept of clustering has been utilized. Curve-like extension of the regions has been tackled successfully.

Another important problem examined in this thesis is the consistent set estimation from dot patterns. We have proposed a new and computationally efficient approach of set estimation, which work for data of any finite dimension drawn from a continuous distribution under certain constraints.

It is to be noted that lack of standard database of dot patterns is a problem in study of dot pattern shape analysis; especially, in evaluation of existing shape descriptors. All previous studies mentioned in the thesis were applied mostly to either hand generated dot patterns or dots randomly drawn from binary images. The random process generated most of the dot patterns we have shown in the thesis. A few complex ones (having some tail-like extensions and density varying even in a single component as in Chapter 5) were manually drawn. We have also illustrated a 'real dot pattern' resulting from representative point set of pigment granules in a squamous epithelium cell.

To describe the scope of the present thesis more systematically, a chapter-wise breakup is given below.

Chapter 2 - Let the pattern plane be partitioned by a lattice of square grids. Consider the union of grids containing points of the dot pattern. If the grid-length s is properly selected, the 'smooth' version of this union of grids approximates the underlying region of the pattern and its border can be considered as the border of the dot pattern. This is the intuition behind the proposed new shape descriptor called, *s-shape*. One can iteratively generate a finite sequence of s-shapes, called *s-shape spectrum*. The mathematical basis of the shape spectrum is worked out. From this spectrum the s-shape closer to the intuitive structure of a dot pattern can be selected using a parameter ε. Because of its inherent simplicity, the descriptor is easily implemented with linear (in terms of cardinality of dot pattern) time complexity in digital domain.

A new structural basis of dot pattern called a *dispersion matrix*[‡] is evolved in the connection of s-shape. Like other existing structural bases namely, Voronoi diagram, Delaunay triangulation, dispersion matrix is a useful tool for shape analysis and shape recovery of dot pattern. A derivative of this dispersion matrix called *binary projection* of

[‡] The term, 'dispersion matrix' is not used in the sense of dispersion matrix of a random variable [Rao 73].

dispersion matrix is a digital binary image, whose cardinality is less than that of the dot pattern [RayChaudhuri et.al. 95a], [RayChaudhuri et.al. 97].

Chapter 3 - We have proposed another external shape descriptor called *r-shape* [RayChaudhuri et.al. 94], which produces border smoother than that generated from the *s*-shape [RayChaudhuri et.al. 95a], [RayChaudhuri et.al. 97]. The idea behind this descriptor is as follows. Subject to proper selection of the structuring radius r, the union of disks with radius r and center at the dots can be considered as a representation of the underlying region of the pattern. The polygonal approximation of this union is the r-shape and is constructed by connecting the respective centers of each pair of intersecting disks, which are partially exposed. The structuring radius for an r-shape is selected from the same sequence of positive numbers that is used for s-shape spectrum. Inconsistent loops may exist in the r-shape, those when deleted partially or completely, gives the border of the dot pattern that is compatible with the perceived shape. An average $O(n)$ time complexity algorithm for extraction of perceived border of a dot pattern is presented in digital domain. We have also illustrated a 'real dot pattern' resulting from representative point set of pigment granules in a squamous epithelium cell. The 'inner border' of the dot pattern is a good demarcater of the nucleus membrane region where it is quite difficult otherwise to differentiate the nucleus border from the cytoplasm.

Chapter 4 - In this chapter, we consider (mathematical) morphology based approaches to recover the underlying structure of dot pattern. The internal shape is the main concern of this chapter, although external shape (borders) extraction is also considered.

The r-shape is interpreted by binary morphological operators. An algorithm on border extraction is presented where a simple extreme point extractor based on binary morphology is incorporated.

It is found that the binary digital image resulting from the dispersion matrix, if smoothed by morphological filters, is a scale reduced digital image of the smoothed s-shape. At first, this image is morphologically thinned and then stretched over the pattern. The proposed skeletonization of the dot pattern has three-fold advantage: (a) any smart binary image thinning algorithm can be used on the binary projection. (b) Due to search space reduction, it is fast; and (c) extraneous spurs (short branches) generally caused by thinning is minimized [RayChaudhuri & Chaudhuri 96].

Chapter 5 - Our approach of shape computation of regular dot patterns is extended to complex cases where the data density may vary in a single component, or overlapped patterns or a denser pattern embedded in a dispersed pattern. We have also developed the approach for detecting patterns where the dots are arranged like arc or line. First such curvilinear patterns are detected and separated out. Then a clustering algorithm is proposed to tackle rest of the pattern [RayChaudhuri & Chaudhuri 97].

Chapter 6 - In this chapter the utility of our approach in set estimation has been theoretically examined. A good set estimator should have the following desirable properties: (a) The estimator should be consistent, i.e. the Lebesgue measure of the symmetric difference of the actual region and the set estimator should go to zero in probability as the number of sample points increase arbitrarily; (b) it should be computationally efficient; (c) it should be automatic, in the sense that the method should be able to detect automatically the number of independent disjoint components making up the true region and (d) it should be robust in the presence of additive noise. None of the estimators mentioned in Section 1.2.2 combine all these properties. We develop a *s*-shape based class of set estimators which overcome this drawback. It is proved that the *s*-shape is a consistent estimator not just under the uniform distribution, but also when the points are drawn according to any continuous distribution. The order of error in estimation is independent of the dimensionality [RayChaudhuri et.al. 97a], [RayChaudhuri et.al. 98]. From the error rate analysis one interesting observation in connection to the *s*-shape spectrum is that for dense patterns there is no significant change of the *s*-shapes after the second iteration.

Chapter 7 - The summary and contribution of the thesis are discussed. Some of the open problems and direction of future work are mentioned.

Chapter 2.

The *s*-Shape : A New Shape Descriptor of Dot Patterns.

2.1 Introduction

In real plane $\Re^2$ one can perceive the border of a point set if the points are clearly visible as well as fairly densely populated and more or less evenly distributed. Such a point set is referred to as a regular dot pattern or simply a dot pattern (DP).

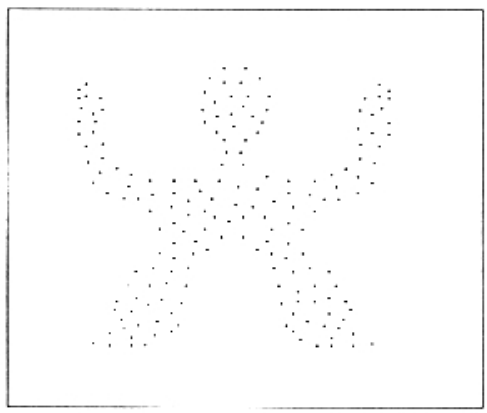


Figure 2.1 *An example of a 'human' shaped dot pattern.*

The present chapter deals with the shape of dot patterns. By the shape of a dot pattern in general, we mean the structure of the bounded underlying region in which the pattern is embedded. Our aim is to propose a intuitively appealing shape descriptor of dot pattern. For easy implementation, the shape descriptor should be simple. The literature survey on shape computation is provided in Section 1.2. We propose an unsupervised approach in $\Re^2$ that can be extended in $\Re^3$ and beyond (see Chapter 6).

The intuitive idea behind the new shape descriptor called *s*-shape (see Figure 2.2) is as follows. Let the pattern plane be partitioned by a lattice of square grids. Consider the union of grids containing points of the DP. If the grid-length s is properly selected, the 'smoothed' version of this union approximates the underlying region of the pattern and its border can be considered as the border of the DP. (See Figure 2.3.) One can iteratively generate a finite sequence of *s*-shapes, called the *s*-shape spectrum. (See Figure 2.4).

It has been shown that the proposed algorithm is computationally efficient (for a given s, it is linear, in terms of number of points) and robust.

In Section 2.2, the mathematical properties of the shape spectrum is worked out. From this spectrum, the s-shape closer to the intuitive structure of a DP can be selected automatically. A new structural basis of a dot pattern called dispersion matrix is evolved which is closely associated with the notion of s-shape (See Figure 2.5). The binary image resulting from this dispersion matrix if smoothed, successfully captures the structure of the dot pattern and the border of the dot pattern is derived from this image.

In Section 2.3, the s-shape based linear order border extraction algorithm is presented and its implementation in digital domain is described. The performance analysis of the proposed approach is done in Section 2.4. A few experimental results on some typical data sets are presented. Also, the possible extension of the above approach of border extraction is discussed in case of complex as well as noisy regular dot patterns.

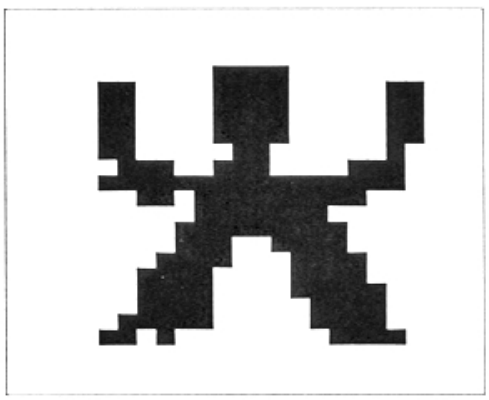


Figure 2. 2 *A typical s-shape of the dot pattern.*

2.2 The s-Shape and its Derivatives

Let S be a dot pattern containing n points. Let W be an optimal isothetic rectangular region (i.e. a rectangle whose sides are parallel to the horizontal and vertical axes of reference) containing S i.e. $S \subset W \subset \Re^2$. For a given grid side-length s, let $\boldsymbol{F}(s)$ denote a lattice of square grids, with horizontal and vertical sides on the real plane. Let for any grid g,

$$G(s) = \left\{ \cup g \,|\, g \in \boldsymbol{F}(s), g \cap W \neq \Phi \right\} \tag{2.1}$$

$$H(s) = \left\{ \cup g \,|\, g \in \boldsymbol{F}(s), g \cap S \neq \Phi \right\} \tag{2.2}$$

Thus, $G(s)$ denotes the set-union of grids over W; $H(s)$, a subset of $G(s)$ is the union of grids each containing at least one dot.

Let $\#H(s)$ denote the number of grids in $H(s)$. The area of $H(s)$ is defined as

$$A(H(s)) = \# H(s) \times s^2 \tag{2.3}$$

Definition 2.1 : The induced hull $H(s)$ with grid-length s is called an s-shape of S.

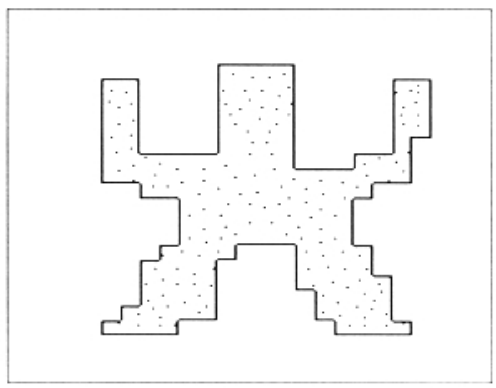


Figure 2.3 *The s-shape based border of the pattern.*

Subject to a proper choice of grid side-length s, the s-shape of S approximates underlying region of the DP. If the grid size is large, almost every non-empty grid contains more than one point. Conversely, if the grid size is relatively small, almost all non-empty grids are isolated. We are looking for a maximal grid size for which almost every non-empty grid will contain only one point of DP so that the connectivity is maintained as much as possible. This is achieved in the following way.

If the points of S are evenly distributed over the optimal rectangle W, then W gives the shape of the dot pattern and its border ∂W denotes the border of the DP. However, in general, points of S are distributed over only a small region of W. To get an approximation of the underlying region, $\bar{s} = \sqrt{A(W)/n}$ can be taken as an initial grid side-length and the s-shape $H(\bar{s})$ gives an crude approximation of the shape.[§] We make further approximation iteratively as follows where the area of the induced hull gradually decreases.

[§] We have established that when n is asymtotically increasing, $\bar{s}$ is the greatest lower bound of grid-lengths for s-shape based consistent set estimation. For details, Chapter 6 is referred.

Consider the sequence $\prec s_i \succ$ defined so that -

$$\left.\begin{aligned} s_i &= \bar{s} && \text{when } i = 1, \\ &= \sqrt{\frac{A\left(H\left(s_{i-1}\right)\right)}{n}} && i > 1 \end{aligned}\right\} \tag{2.4}$$

Since, $\#H(s_i) \leq n \quad \forall i,$

we have, $0 \leq s_{i+1} = \sqrt{\frac{\#H\left(s_i\right)}{n}} \times s_i$

$\leq s_i.$

So, $\prec s_i \succ$ is a non-negative monotonically non-increasing sequence. Moreover as this monotonically non-increasing sequence is bounded from below (0 is a lower bound), it is also convergent.

Let $$\lim_{i \to \infty} s_i = \underline{s}\,. \tag{2.5}$$

The induced hull $H(s_i)$ gradually coincides with $H(\underline{s})$ as s_i tends towards $\underline{s}$. Thus, the following proposition is established :

Proposition 2.1 : The sequence $\prec s_i \succ$ and $\prec A\left(H_i\right) \succ$ are both monotonically non-increasing and convergent and if $\underline{s}$ is the limiting value of $\prec s_i \succ$ then,

$$\lim_{i \to \infty} s_i = \underline{s} \Rightarrow \lim_{i \to \infty} A\left(H_i\right) \to A\left(H\left(\underline{s}\right)\right). \tag{2.6}$$

Note that s_i in $\prec s_i \succ$ will continue to decrease i.e. $s_i < s_{i-1}$ until each grid in $H(s_i)$ contains only one dot. The rate of convergence of the sequence mostly depends on homogeneity and structure of the DP. Now, we will show that $\prec s_i \succ$ attains its minimum value i.e. attains the limit, which is strictly positive after a finite number of steps.

Note, $$s_{i+1} = \sqrt{\frac{\#H\left(s_i\right)}{n}} \times s_i\,. \tag{2.7}$$

Which implies $$\sqrt{\frac{\#H\left(s_i\right)}{n}} = \frac{s_{i+1}}{s_i}\,.$$

By taking limits on both sides, $$\frac{\lim_{i\to\infty}\sqrt{\#H(s_i)}}{\sqrt{n}} = \frac{\lim_{i\to\infty} s_{i+1}}{\lim_{i\to\infty} s_i} = 1 .$$

In other words, $$\lim_{i\to\infty} \#H(s_i) = n .$$

So, $\prec \#H(s_i) \succ$ is a convergent sequence of positive integers. As any convergent sequence of integers contains at most a finite number of distinct integers, $\prec \#H(s_i) \succ$ has only finite elements which are distinct and the rest are equal to the limiting value n of the sequence.

Let $$j = min\{i \mid \#H(s_i) = n\}. \tag{2.8}$$

Note that this limiting value is attained only when all points of S are mutually separated by grids. Then according to definition, any s_i, for $i \geq j$ is constant and is equal to the limiting value; i.e. $s_i = \underline{s} \quad \forall i \geq j$.

If possible let, $\underline{s} = 0$. We shall show that this is not possible.

Since we have assumed that all points of S are not collinear,

$$A(W) > 0 \Rightarrow s_1 = \bar{s} > 0 . \tag{2.9}$$

Now, since $\prec s_i \succ$ is strictly monotonically decreasing sequence converging after finite number of steps, there exists a positive integer j such that

$$\begin{aligned} \bar{s} > s_i \;&> s_{i+1} && \forall i < j \\ &= \underline{s} && otherwise. \end{aligned} \tag{2.10}$$

But as $s_{j-1} > 0$ and number of non-null grids in $H(s_{j-1}) > 0$,

$$s_j = \sqrt{\frac{\#nonnull\ \ grids\ \ of\ \ H(s_{j-1})}{n}} \times s_{j-1} > 0 . \tag{2.11}$$

Which is a contradiction to the hypothesis that $\underline{s} = 0$. Hence, the following important result is established.

Proposition 2.2 : The sequence $\prec s_i \succ$ converges to $\underline{s} > 0$ after a finite number of steps.

Definition 2.2 : The sequence $\prec H(s_i) \succ$ is called the s-shape spectrum of the dot pattern.

The finiteness and positivity of $\prec s_i \succ$ ensures that the s-shape spectrum is *computable*.

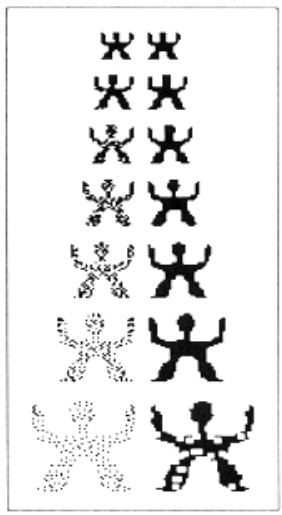


Figure 2.4 *s_i -times (s_i gradually decreasing) scale reduced shape spectrum and its smooth version (morphologically closed by unit digital disk)*

Being a union of fixed size squares, the border of the *s*-shape has a stair-case like appearance. Also, if the value of s_i is small, the *s*-shape may contain holes and its border may appear cracked. So, the border of a *s*-shape in the spectrum is considered as a crude border of the dot pattern. Any hole filling as well as border smoothing algorithm, conventional [Rosenfeld & Kak 82], [Jain 89], [Gonzalez & Woods 94] or morphological [Serra 82], [Haralick & Shapiro 92], [Schonfeld & Goutsias 91], [RayChaudhuri et.al. 95] gives a better appearance to the *s*-shapes. The binary morphological closing is used in the proposed algorithm (see Section 2.3.2) for that purpose. The binary closing is a simple morphological filter that smoothes the set from outside: all background structures that can not contain the structuring element are added to the foreground. (For definition of closing and other basic morphological operators see Appendix) It is desirable to know the value of *s* for which the *s*-shape of dot pattern agrees well with the perceived shape. An automatic choice of an appropriate *s* from $\prec s_i \succ$ is in order.

The terminal value $\underline{s}$ cannot give an appropriate grid-length unless points of S are highly regularly distributed, which is unlikely. For a random dot pattern with more or less uniform distribution, we can consider the sequence $\prec s_i \succ$ as a spectrum having $\bar{s}$ and $\underline{s}$ as upper and lower bound. A suitable grid from the upper values of $\prec s_i \succ$ where neighboring values (pre and post values in the sequence) are relatively nearly the same, is selected.

For a small $\varepsilon > 0$, let

$$\hat{s} = max\left\{s_k ; \left|\frac{s_{k-1} - s_{k+1}}{s_k}\right| \leq \varepsilon, s_{k-1} \in \prec s_i \succ\right\} \tag{2.11}$$

Definition 2.3 : The dispersion measure $\hat{s}$ is called the structuring grid-length of the s-shape.

We would like to point out here that 'perceptual structure' of dot pattern cannot be defined uniquely. It varies from person to person to a small extent. The ε factor introduces such a notion in the structuring grids. If we fix a value of ε in a suitable range we obtain an acceptable s-shape (based border). For a more or less evenly distributed DP, ε in the range (0.3-0.5) (typically 0.4) is a good choice (for details see Section 2.4).

Let the grids in $G(s)$ with grid side-length s be ordered in 2-dimensional array. Then it induces a matrix (say,) $\left((t_{lm})\right)$ whose (l,m)-th element denotes the number of dots in the grid situated at l-th row, m-th column position. The binary projection of $\left((t_{lm})\right)$ is another matrix (say,) $\left((b_{lm})\right)$ where

$$\left.\begin{aligned} b_{lm} &= 1 \qquad if\ \ t_{lm} > 0 \\ &= 0 \qquad otherwise. \end{aligned}\right\} \tag{2.12}$$

The matrix $\left((t_{lm})\right)$ can be considered as a gray scale digital image whereas its projection can be considered as a binary digital image where each object pixel represents a non-empty grid of $H(s)$. In case of a digital dot pattern (as discussed in Section 2.3), the projection $\left((b_{lm})\right)$ (corresponding to the binary image $G(s)$) is a s^2 times reduced size image of $H(s)$.

The sequence of such matrices correspond to $\prec s_i \succ$ are of particular interest and some quantitative analysis on homogeneity in dot pattern as well as measure of dispersion are possible. Consider the collection of binary projections $((b_{lm}))$ corresponding to $G(s_i)$, for $i = 1, 2, .. , k$. If the collection is ordered by i then the resulting sequence of binary images as well as their smooth versions show how s-shape spectrum captures the underlying structure of the DP. Such a spectrum and its smooth version, where each projection is morphologically closed, on the DP of Figure 2.1 is presented in Figure 2.4. The structuring element used for the closing is illustrated in Figure 2.6. Next, we will consider the matrix $((t_{lm}))$ which is induced by the $G(\hat{s})$.

Definition 2.4 : The induced matrix of $G(\hat{s})$, i.e. when the structuring grids are used, is said to be the induced dispersion matrix (or simply dispersion matrix) on the dot pattern.

The dispersion matrix with the structuring grid is denoted by $DMAT(S)$ whereas the binary projection of the dispersion matrix is denoted by $DMAT_{prj}(S)$.

Figure 2.5 illustrates the dispersion matrix of the dot pattern of Figure 2.1 where the value of $\varepsilon = 0.4$.

```
0 0 0 0 0 0 1 2 2 1 0 0 0 0 0 0 0
4 1 0 0 0 0 1 1 1 1 0 0 0 0 0 3 2
2 1 0 0 0 0 1 1 1 2 0 0 0 0 0 2 1
1 1 0 0 0 0 1 1 1 2 0 0 0 0 0 1 1
3 1 0 0 0 0 1 1 1 1 0 0 0 0 0 3 1
1 1 0 0 0 0 0 1 1 0 0 0 0 0 0 2 0
0 1 1 1 0 0 1 1 1 0 0 0 0 1 1 1 0
1 1 1 1 1 2 1 1 1 1 1 2 2 1 1 1 0
0 0 1 2 0 1 1 1 1 2 1 1 1 2 0 0 0
0 0 0 0 0 1 1 1 1 1 1 1 0 0 0 0 0
0 0 0 0 1 1 2 1 1 1 2 1 0 0 0 0 0
0 0 0 0 1 1 2 0 0 1 1 1 1 0 0 0 0
0 0 0 1 1 1 1 0 0 0 1 1 1 1 0 0 0
0 0 1 1 1 1 0 0 0 0 1 1 1 1 0 0 0
0 0 0 2 1 1 0 0 0 0 1 1 1 1 1 0 0
0 0 2 1 1 1 0 0 0 0 0 1 1 1 2 0 0
0 1 1 2 1 2 0 0 0 0 0 2 1 3 1 0 0
1 1 0 2 0 0 0 0 0 0 0 0 1 1 1 1 0
```

Figure 2.5 *The dispersion matrix of Figure 2.1* ($\varepsilon = 0.4$)

The dispersion matrix, which is invariant under translation and scaling can be used as a structural basis of dot patterns. It can be considered as a gray scale image whereas its projection can be considered as a binary image where each object pixel represents a non-empty grid of $H(\hat{s})$. The projection $DMAT_{prj}(S)$ may be porous and cracked due to local

irregularity of distribution of the pattern. It is closed with a *unit digital disk*. The unit digital disk is a 3×3 window having all entries valued 1 and the center of reference is situated at the central grid of the window (see Figure 2.6). The resulting image is a smoothed version that captures the underlying structure of DP. In Section 2.3, a digital implementation of an *s*-shape based algorithm on border extraction of a DP is presented. The computation is done over the binary projection of the dispersion matrix. Note, the cardinality of projection is less than that of the dot pattern. Another application of the $DMAT_{prj}(S)$ for skeletonization of DP, is presented in Chapter 4.

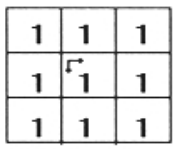


Figure 2.6 *A unit digital disk*

2.3 Digital Implementation

Let S be a binary image where almost all object pixels, $P_i = (x_i, y_i) \in \Im^2$, $i = 1, 2, \ldots n$ are disconnected but densely and more or less evenly distributed in a subregion. Then S is called a *digital dot pattern*. Let each P_i be colored or labeled by i.

2.3.1 Computation of dispersion matrix : Estimation of s

Step 1 :

Find

$$a \leftarrow \min_{i=1}^{n} x_i;$$

$$b \leftarrow \max_{i=1}^{n} x_i;$$

$$c \leftarrow \min_{i=1}^{n} y_i;$$

$$d \leftarrow \max_{i=1}^{n} y_i;$$

$$A(W) \leftarrow (b-a)\times(d-c);$$

(W is the optimal isothetic rectangular region whose four vertices are respectively, top-left $\omega_1(a, c)$; top-right $\omega_2(a, d)$; bottom-left $\omega_3(b, c)$; and bottom-right corner, $\omega_4(b, d)$.)

Compute

$$s_0 \leftarrow \sqrt{\frac{A(W)}{n}};$$
$$M_0 \leftarrow \left\lceil \frac{b-a}{s_0} \right\rceil;$$
$$N_0 \leftarrow \left\lceil \frac{d-c}{s_0} \right\rceil;$$

Set

$$t \leftarrow 1;$$
$$s_{-1} \leftarrow A(W);$$

Step 2 :

Initialize the entries of two matrices $((z_{lm}))$ and $((z'_{lm}))$ of order $M_{t-1} \times N_{t-1}$:

$$z_{lm} \leftarrow z'_{lm} \leftarrow 0;$$

Set count number of total non-null grids :

$$c_t \leftarrow 0;$$

At each pixel P_i ($i = 1,2, ..., n$) :

Compute

$$l \leftarrow \left\lceil \frac{x_i - a}{s_{t-1}} \right\rceil;$$
$$m \leftarrow \left\lceil \frac{y_i - c}{s_0} \right\rceil;$$
$$z_{lm} \leftarrow z_{lm} + 1;$$
$$if\left(z'_{lm} \equiv 0\right) \quad c_t \leftarrow c_t + 1;$$
$$z'_{lm} \leftarrow 1;$$

Step 3 :

Compute

$$((t_{lm})) \leftarrow ((z_{lm}));$$
$$((b_{lm})) \leftarrow ((z'_{lm}));$$
$$s_t \leftarrow s_{t-1} \times \sqrt{\frac{c_t}{n}};$$
$$M_t \leftarrow \left\lceil \frac{b-a}{s_t} \right\rceil;$$
$$N_t \leftarrow \left\lceil \frac{d-c}{s_t} \right\rceil;$$

Step 4 :

$$If \left(\frac{s_{t-2} - s_t}{s_{t-1}} \leq \varepsilon \right) \ then \quad (i.) \ \ s \ \leftarrow \ \lceil s_{t-1} \rceil;$$

$$(ii.) \ dispersion \ matix \ is \left((t_{lm}) \right);$$

$$(iii.) \ and \ its \ projection \ is \left((b_{lm}) \right);$$

$$else \quad t \leftarrow t+1; \ goto \ \text{Step 2}.$$

Note that for estimating the structuring grid-length, explicit computation of $\left((t_{lm})\right)$ or $\left((b_{lm})\right)$ is not necessary but at each iteration only the number of non-null grids is required.

2.3.2 Computation of s-shape based border of DP

Step 1. Find the projection of the dispersion matrix, $\left((t_{lm})\right)$.

Step 2. Smooth $\left((b_{lm})\right)$ by a binary morphological closing with a unit digital disk. (The definition of closing is given in the Appendix).

Step 3. The border of the resulting output of the closed version after its $\hat{s}$ times magnification followed by a translation to the point ω_1, is considered as an approximate border of the DP.

Thus, for the *s*-shape re-generation purpose, information in the binary closed projection of the dispersion matrix, the value of structuring grid-length $\hat{s}$ and the top-left corner of the optimal covering rectangle ω_1 may be stored.

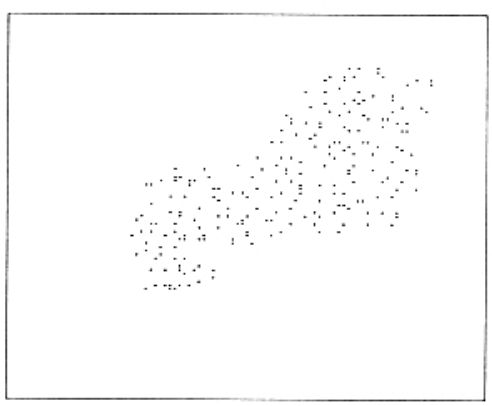


Figure 2.7 *A 'polar bear' shaped dot pattern*

2.3.3 Computational complexity analysis

It is evident that for every iteration, the computation of $((b_{lm}))$ is linear with respect to n. In proposition 2.2, we have shown that in analog case, limiting size of the bucket is attained only after a finite number of steps. However, in digital domain, since the structuring radius is an integer number, in the worst case, $(2\lceil s_2 \rceil + 1)$ is considered as an upper bound of iteration number. Thus, for a dense pattern where $s_2 << n$, on an average, the ε-measure of dispersion $\hat{s}$ can be computed in linear time with respect to n. The actual run-time for s-shapes (as well as r-shapes) computation for various values of n for a typical dot pattern is presented in Section 3.5.1.

From the above discussion it is clear that, the computation of dispersion matrix as well as its projection is linear with respect to the cardinality of DP. Note that for smoothing of s-shape, the main part of computation is binary closing of $DMAT_{prj}$ by a unit digital disk. To obtain the result, the worst case computation for each object pixel of $DMAT_{prj}$ is 64 binary comparisons. Thus, the average computational complexity of s-shape based border is $O(n)$ where n is the cardinality of the dot pattern.

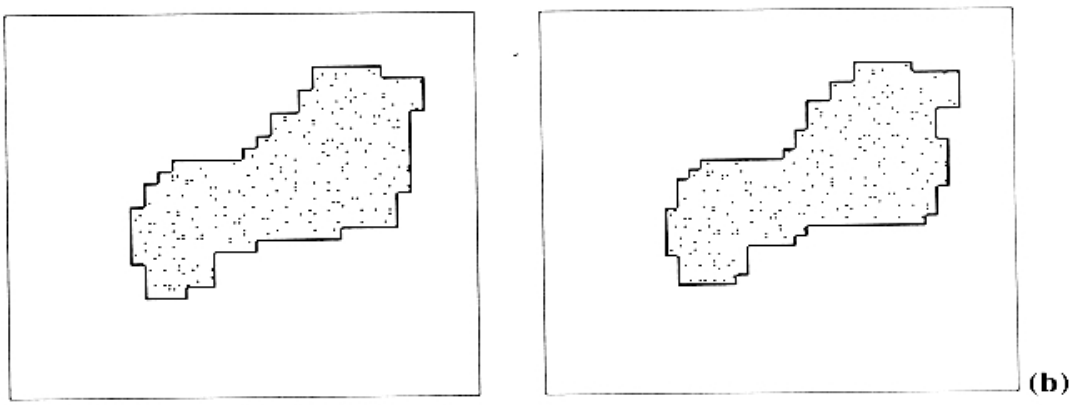


Figure 2.8 *Compatibility of s-shape with the perceived border of Figure 2.7*
(a) ε = 0.3(0.4) **(b)** ε = 0.5.

2.4 Performance Analysis

We have tested the proposed algorithms on several sets of synthetic dot patterns. We have experimentally found that for a simple regular dot pattern, if ε lies in the range 0.3 to 0.5, the smoothed s-shape successfully captures the underlying region of the dot pattern.

For the 'polar bear' shaped DP (Figure 2.7) the *s*-shape based extracted borders for ε = 0.3, 0.4, 0.5 (The shapes for ε = 0.3, 0.4 are, in fact the same) are shown in Figure 2.8(a) and Figure 2.8(b), respectively.

In Figure 2.9, the borders based on *s*-shape are presented for a few other typical dot patterns. In all these cases ε is fixed at 0.4.

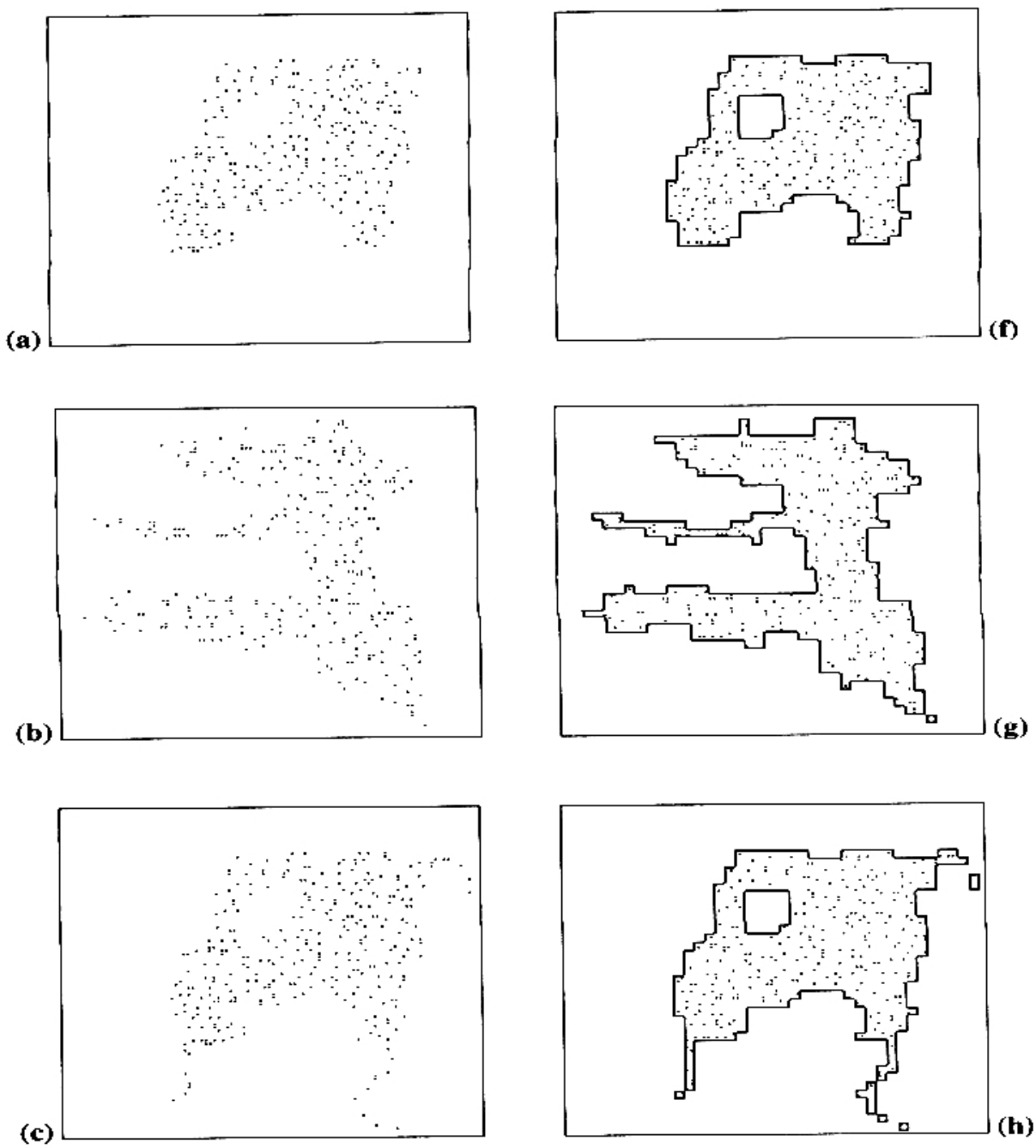


Figure 2.9 *Continued*

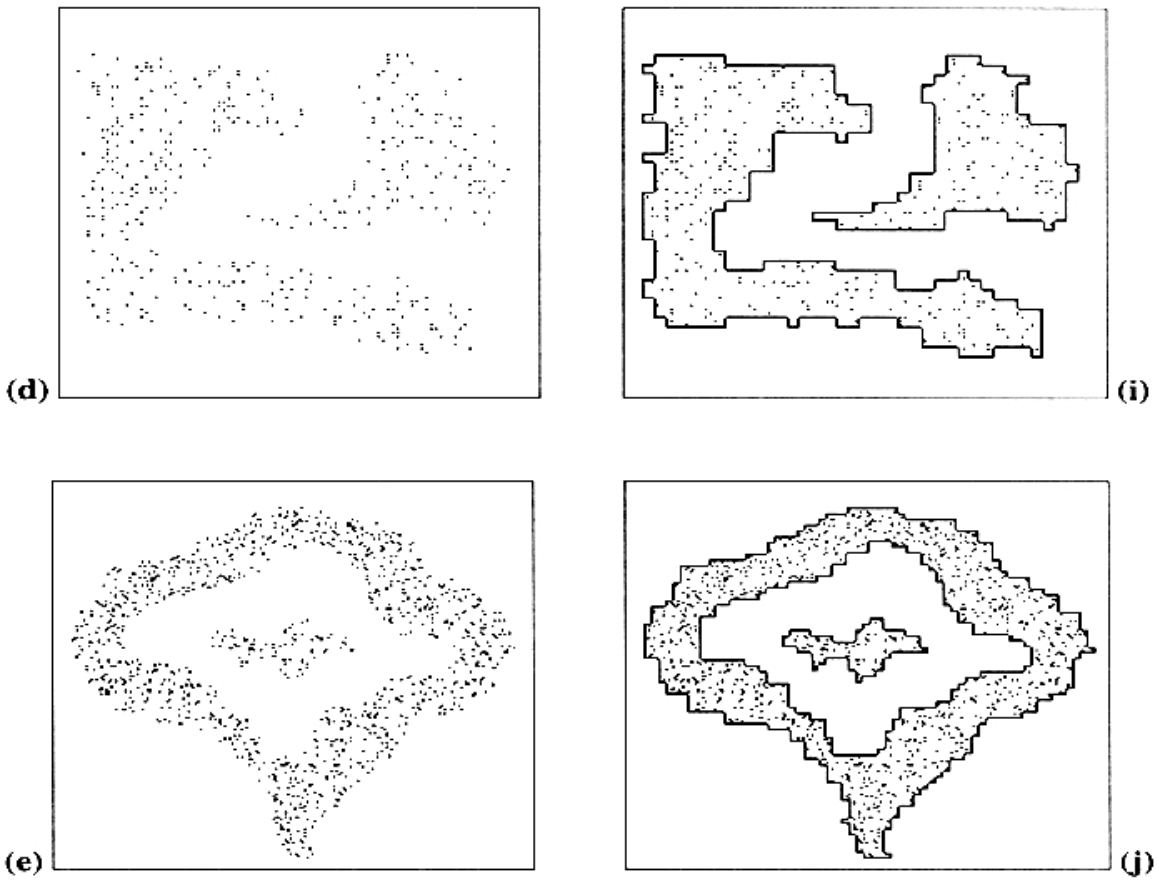


Figure 2.9 *A set of typical dot patterns (having holes, elongated limbs and disconnectivity) with their extracted borders from respective s-shapes.*

The boundary extraction algorithm is directly applicable for disconnected dot patterns also where the dispersion measures of individual components (subpatterns) are more or less same and pair-wise they are apart from each other by more than $2\sqrt{2}\,\hat{s}$ ($\hat{s}$ is the structuring grid-length for the whole pattern set). For example, in Figure 2.9(d) and Figure 2.9(e) the number of such disconnected patterns is 2. However, if the pattern set contains components of different ε-dispersion measures, a 'cut one out' approach is used as follows.

Step 1. Find the dispersion matrix for the mixed point patterns and find the *densest* component in it. (If the average number of dots in a component of the projection is maximum then that component is considered as the densest one.)

Step 2. Trace back the induced pattern of this most dense component of the matrix in the original mixed pattern and extract its border separately.

Step 3. Remove the extracted part from the mixed patterns and repeat the whole process.

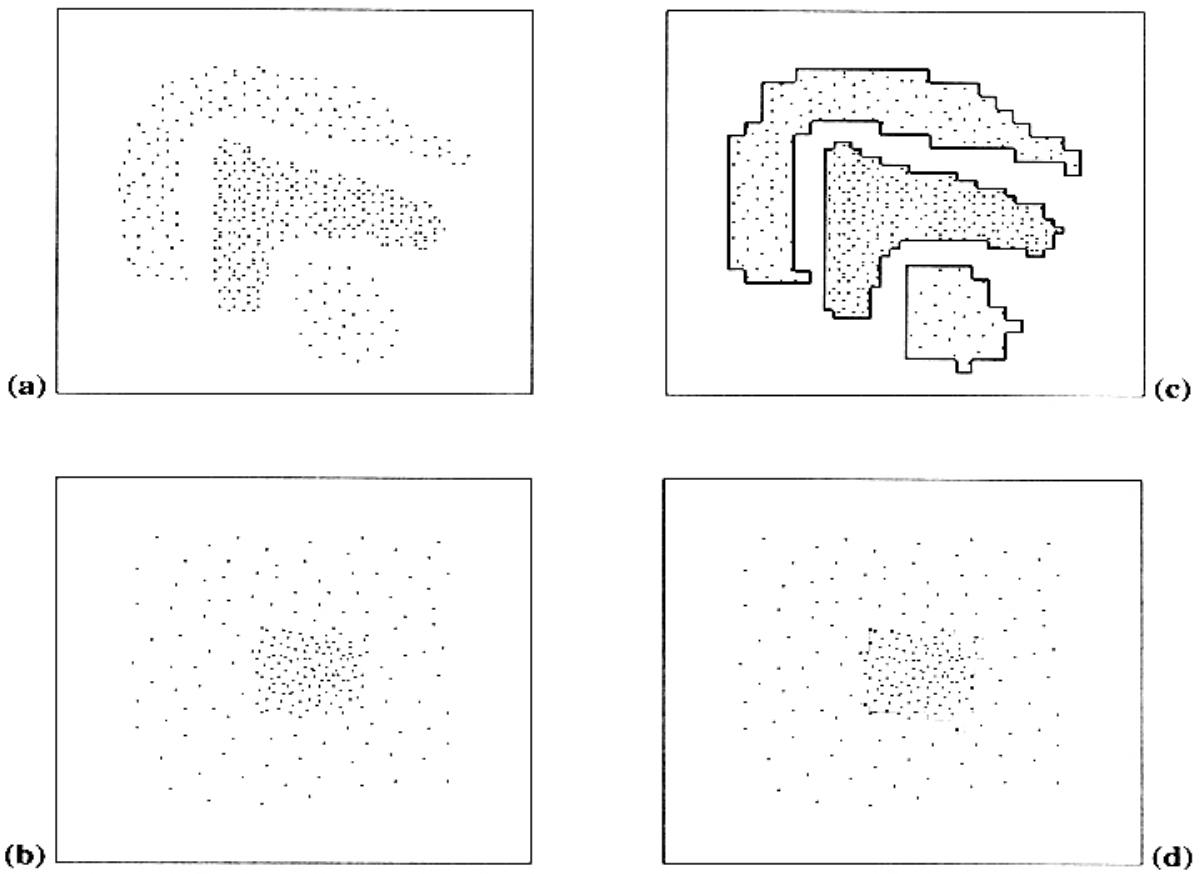


Figure 2.10 *Two complex dot patterns with their s-shape based borders*

This approach should be modified when the patterns are not connected (e.g. when the distance between two dot patterns is less than the sum of their respective structuring grid-lengths). At the overlapped zone, the border of the denser pattern may be considered as the common separator and the border of the less dense pattern should be modified accordingly. Two cases of such complex dot patterns are shown in Figure 2.10(a) and Figure 2.10(b). In the first case (Figure 2.10(a)) patterns are disconnected but in the latter one (Figure 2.10(b)) denser pattern is embedded in another DP. The results are shown in Figure 2.10(c) and Figure 2.10(d), respectively. A more effective method is provided in Chapter 5 which can handle complex patterns where the density changes locally within a component.

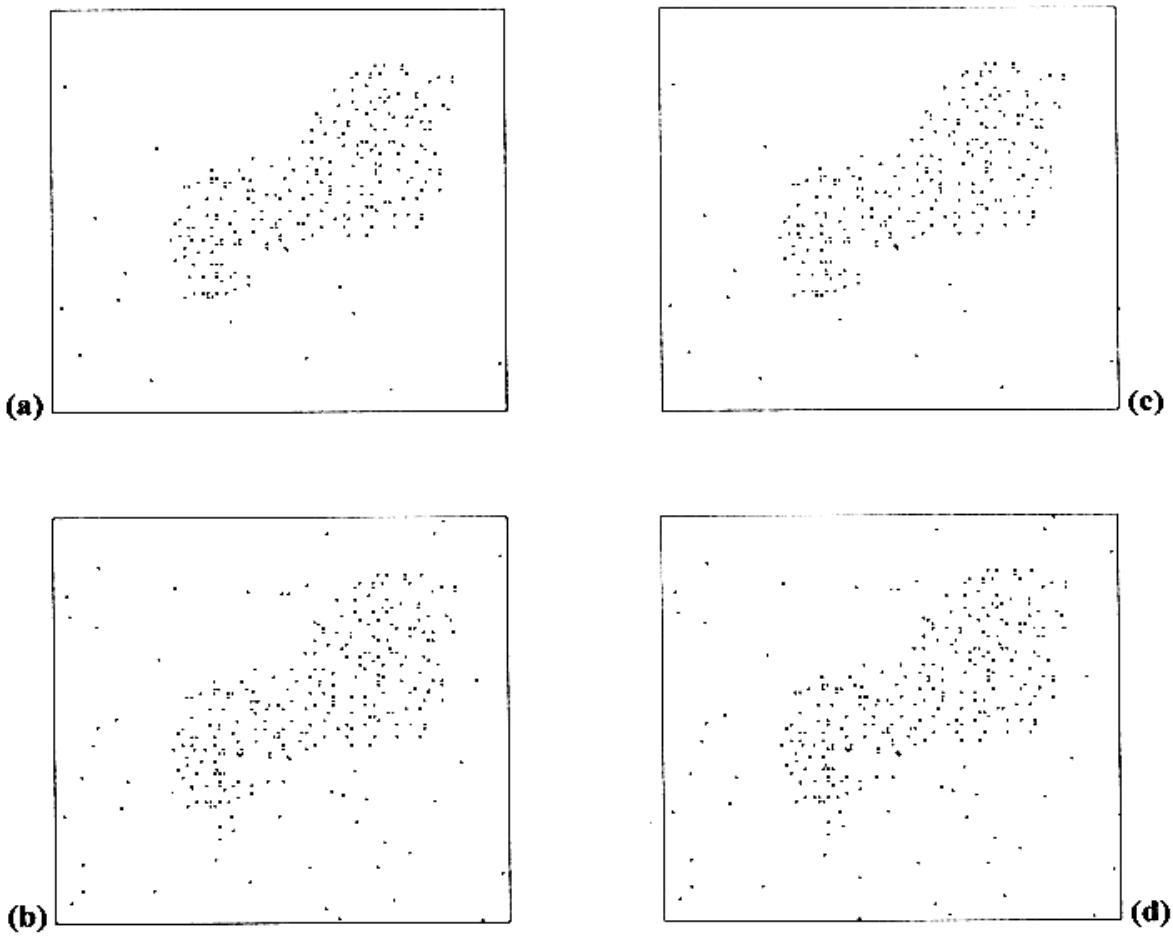


Figure 2.11 *Robustness in presence of noise*

If a dot pattern is degraded by noise then by applying the alternating filter [Serra 88], [Schonfeld & Goutsias 91] on the $DMAT_{prj}$ a considerably good result could be achieved in the following way. After binary closing of $DMAT_{prj}$ by unit digital disk, the output is opened by the same structuring element. For example, in Figure 2.11(a) and Figure 2.11(b) the signal to noise ratio, SNR is 10 db and 5 db [Zhou & Gordon 91]. The resulting borders (Figure 2.11(c) and Figure 2.11(d) respectively) are quite satisfactory.

Note that our method can be easily implemented on hardware where the proposed border extractor can be treated as a low-level vision operator. The proposed approach on external shape computation can be extended to data in higher dimensions. This can also be applied for data clustering where the final output in the form of a set of edges will give the boundary of a cluster. Regarding the consistency of s-shape some nice theoretical results are established in Chapter 6.

Chapter 3.

The *r*-Shape : Towards Perceptual Shape Recovery

3.1 Introduction

This chapter presents a computational approach to extracting the border of the dot pattern that is found reasonably close to the perceptual structure of the dot pattern. It is not the intention here to model the grouping mechanisms used by the human visual system [Zusne 70], [Lowe 85]. Rather, the goal is to present a shape descriptor and an efficient algorithm for border generation of (digital) dot pattern with a single appropriate parameter based on that descriptor.

The outline of a dot pattern generated by *s*-shape has edges only in horizontal and vertical directions, thus making stair-case like appearance of border. Here we propose an external shape descriptor called *r*-shape, that creates a smooth border closer to the perceptual shape of the dot pattern. In this polygonal border representation, edges are not restricted in horizontal and vertical directions.

The idea behind this descriptor is as follows. Here, subject to proper selection of the radius *r*, called structuring radius, by a data driven procedure, the union of disks with radius *r* and center at the dots is considered as a representation of the underlying region of the pattern. The *r*-shape is constructed by connecting the respective centers of each pair of intersecting disks which are partially exposed to the background. The structuring radius for an *r*-shape is selected from the same sequence of positive numbers that is used for *s*-shape spectrum (Definition 2.2). After partial or complete deletion of inconsistent loops (characterized in Section 3.2.1) from the *r*-shape, we obtain the border of the dot pattern that is compatible with the perceived shape.

In Section 3.2, the *r*-shape and its related terminologies are formally defined. Among the existing external shapes, the result of *r*-shape has close similarity with that of the α-shape proposed by Edelsbrunner et. al. [Edelsbrunner et.al.83]. The relationship of *r*-shape and α-shape is presented in this section. Also, the inconsistent edges in *r*-shape are characterized.

In Section 3.3, the *digital r-shape* is defined and an algorithm to compute it with known structuring radius is presented. For efficient implementation and faster processing of *r*-shape, a successor listing type of data structure is proposed in Section 3.4. Based on that data structure, an algorithm of inconsistent edge detection as well as extraction of the perceived shape is presented in detail. Finally, in Section 3.5, the evaluation of our *r*-shape based border extraction is done. The computational complexity of the algorithm is analyzed and some results are presented in this section. A 'real dot pattern' resulting from representative point set of pigment granules in a squamous epithelium cell is presented and a method of demarcating the nucleus border is also illustrated .

3.2 The r-Shape

Let $S\{P_1, P_2, \ldots P_i, \ldots, P_n\}$ be a set of n points in $\Re^2$ and let r be a positive quantity. Let $D_r(Q)$ be the closed disk with radius r and center Q and let $C_r(Q)$ be the boundary of $D_r(Q)$. The boundary and the interior of a closed set A will be denoted by ∂A and *int* A respectively.

Consider the union of all disks centering at some points of S with fixed radius r. If the perimeter of the disk centering the point P_k is at least partially exposed to the background then its an *r-extreme point*. Otherwise, it is an *r-interior point*. For example, in Figure3.1(a) P_k is an *r*-interior point. Formally,

Definition 3.1 : A point P_k in S is said to be an r-interior point of S if and only if

$$C_r(P_i) \subset int\left(\bigcup_{j=1}^{n} D_r(P_i)\right). \tag{3.1}$$

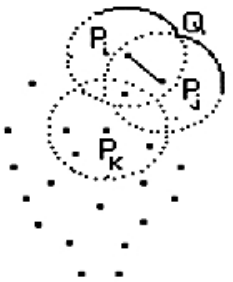
Let

$$E_r(P_i) = C_r(P_i) - int\left(\bigcup_{j=1}^{n} D_r(P_j)\right) \tag{3.2}$$

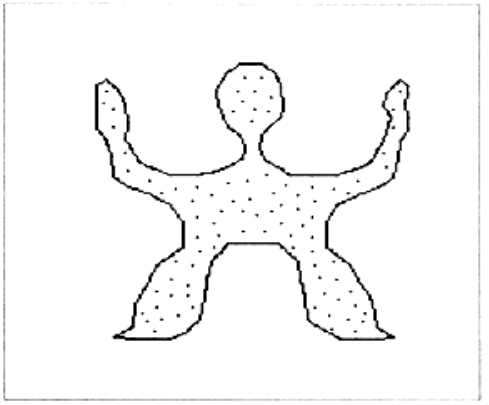
It is easy to verify the following proposition –

Proposition 3.1 : $E_r(P_i) = \phi$ if only if P_i is an *r*-interior point of S.

Definition 3.2 : A point P_i is said to be r-adjacent point of P_j and $\overline{P_iP_j}$ is said to be an r-edge if and only if $E_r(P_i) \cap E_r(P_j) \neq \phi$.



(a) *Illustration of r-interior, r-edge and r-adjacents (r-extreme points)*



(b) *The r-shape (closed curve). (Input pattern is given in Figure 2.1)*

Figure 3.1

The above definitions are illustrated in Figure 3.1(a). Note that $\overline{P_iP_j}$ is an r-edge because the disks centering P_i and P_j respectively, have a common point (say, Q) that is exposed to the background with respect to the union of all disks centering points of S. (See Figure 3.1(a); the shaded region represents the union of disks and the portion bounded by dotted line in Figure 3.1 (b) is magnified in Figure 3.1(a) where P_k is an r-interior point.)

Definition 3.3 : Any r-extreme point P_i being an end point of an r-edge is said to be an *r-vertex* of S.

So, the set of r-vertices consists of non-isolated r-extreme points. The number of r-edges passing through an r-vertex P_i is said to be the *degree* of that vertex. If P_i has more than two r-adjacents then P_i is said to be an *r-branch* vertex of S.

Definition 3.4 : The r-shape of the planar point set S, denoted by $rsh(S)_r$ is the planar straight line graph whose vertices are the r-vertices and whose edges are comprised of r-edges.

An *r-path* (or, simply a *path*) in r-shape graph is defined as a maximal alternating sequence of r-vertices and r-edges, beginning and ending with r-vertices (referred to as *end vertices*), such that each r-edge is incident with the r-vertices preceding and succeeding it. No r-vertex appears more than once and all other r-vertices except the end

vertices on the path is of degree 2. A path can be expressed by a string of r-vertices. In the diagram of r-shape, a path represents a simple curve.

Let us recall the existing definitions of α-shape [Edelsbrunner et. al.83]. Let α be an arbitrary real number. A generalized disk of radius $1/\alpha$ is defined as a closed disk of radius $1/\alpha$ if $\alpha > 0$, the closed complement of a disk of radius $-1/\alpha$ if $\alpha < 0$, and a closed half-plane if $\alpha = 0$. For a set S of 2-D points, the α-hull of S is the intersection of all generalized disks of radius $1/\alpha$ that contain S. A point P_i in S is α-extreme in S if there exists a generalized disk of radius $1/\alpha$ containing S such that P_i lies on its boundary. Two α-extreme points P_i and P_j of S are α-neighbors if there exists a generalized disk of radius $1/\alpha$ containing S such that both P_i and P_j lie on its boundary. The α-shape of S is the planar straight line graph whose vertices are the α-extreme points and whose edges connect the respective α-neighbors. As α approaches to zero, the α-shape becomes the convex hull of S.

The relation between r-shape and α-shape is as follows. In all terminologies (e.g. r-interior point) and transforms (e.g. E_r) defined to find the r-shape, at first the union of disks are taken and then the interior of the union is considered. On the other hand, if the interior of each disk is taken and then their union is considered then it can be proved that the edge joining the points P_i and P_j satisfying $E_r(P_i) \cap E_r(P_j) \neq \phi$ is an edge of the α-shape ($\alpha = -1/r > 0$) [RayChaudhuri et.al. 94], [RayChaudhuri et.al. 97].

Using the properties of α-shape and r-shape it can be shown that r-shape is a subgraph of its respective α-shape. The r-shape graph does not contain any point not having a neighbor within its r-distance but such an isolated point is a vertex of the respective α-shape. In addition to the isolated points, other vertices may exist in the α-shape ($\alpha = -1/r$) that do not occur in the respective r-shape. For example, consider an α-extreme point P_i such that $C_r(P_i) \not\subset \bigcup_{j=1}^{n} int\, D_r(P_j)$ but $C_r(P_i) \subset int\left(\bigcup_{j=1}^{n} D_r(P_j)\right)$. Then there exists a point Q on $C_r(P_i)$ such that if it belongs to any disk $D_r(P_j)$, its boundary $C_r(P_j)$, passes through Q. Since P_i is an r-interior point, at least the boundary of three disks intersect at Q making Q under interior of union of disks. All edges in the α-shape with one end at P_i are *weak edges* and such edges do not occur in the respective r-shape. Here, by a weak edge

we mean an edge that disappears when the value of the graph parameter is slightly changed. In this case if α is slightly decreased from $-1/r$, these weak edges disappear. Thus, the r-shape is more *stable* than the respective α-shape (that is when $\alpha = -1/r$).

It is clear that the r-shape is different for different values of r. To get a perceptually acceptable shape, a suitable value of r should be chosen and there is no closed form solution to this problem. We have not come across any published literature on proper selection of r. In principle, one should be able to get an appropriate value of r, called *structuring radius* from the dot pattern itself. (The disk centered at origin and with radius of length of the appropriate r, is said to be the structuring disk.) Note that for each s_i of the sequence $\prec s_i \succ$, we can find an r-shape where $r = s_i\sqrt{2}$. However, we shall consider the r-shape corresponding to the structuring grid length $\bar{s}$. The selection of structuring radius is justified by the possibility of existence of single point in diagonally opposite corners of two 8-*connected* grids. (We use 8-connectivity as described in digital image processing literature.)

Definition 3.5 : The quantity $\bar{r} = \bar{s}\sqrt{2}$ is called the structuring radius of the dot pattern.

If we fix a value of ε from a suitable range we obtain an acceptable s-shape (as discussed in the previous chapter) and the r-shape. For more or less evenly distributed dot pattern, ε in the range 0.3–0.5 (typically 0.4) is a good choice for r-shape also. (See Figure 3.6(a)-(b).)

3.2.1 Characterization of inconsistency in the r-shape

Due to local irregularity in distribution on the pattern, some edges in r-shape (even for an optimal structuring radius r,) are redundant and this is inconsistent with the perceptual structure. This inconsistency can be characterized by :

I. Existence of one or more simple closed contours of negligible length, (e.g. A in Figure3.7(a))

II. Existence of multiple paths between two branch vertices some of them of negligible length. (For example, multiple paths exist between B and C in Figure 3.7.(a)).

A successor listing of a graph is a data structure used for the graph representation. In r-shape graph the ratio of number of edges (e), to the number of vertices (v), is very low. Thus successor listing is very convenient for storage, retrieval, and manipulation of the graph. In general, the structure of successor listing is as follows. After assigning the vertices distinct numbers 1, 2, ... v, a linear array is constructed for each vertex k whose elements are the vertices adjacent to k [Deo 90].

A successor listing type of data structure, referred to as *r-shape Listing* is proposed for cleaning the border of DP generated by r-shape by tracing the route(s) from one r-vertex to another. Its composition and the way of tracing is described in Section 3.4.

A simple closed curve in an r-shape of length less than the perimeter of the structuring disk ($\approx 6 \times \bar{r}$, where 6 is an approximation of 2π) is considered as an inconsistent loop because of its negligible length. However, since the distribution of points in a DP is more or less even, average length of r-edges is approximated by $\bar{r}$. Thus, a simple closed loop in digital implementation is considered as inconsistent if the total number of r-edges on the loop is less than 6.

Let more than one path between two branch vertices be detected. Among them, a path if not the shortest one, is deleted if the total Euclidean length of that path and the smallest route between those branch vertices is smaller than the perimeter of the structuring disk. In digital implementation, as in the above case, for determining the validity of a path between the two branch vertices we consider only the number of edges in the closed loop. A path in between two branch vertices is considered inconsistent if the total number of r-edges on this path as well as on the corresponding shortest path is less than 6.

3.3 Digital r-Shape and its Computation

Let $S = \{P_1, P_2, \ldots P_i, \ldots P_n\}$ be a digital dot pattern consisting of n discrete points. Under a metric 'd' (d is the Euclidean metric unless otherwise specified), the discrete (digital) disk with radius r around P_i is a set $DD_r(P_i)$ of pixels, $P = (x, y)$ so that the distance of every P from P_i satisfies the inequality $d(P, P_i) < r + 0.5$. Moreover, the label of each pixel P in $DD_r(P_i)$ is i. Let τ_r be the set of ordered pairs defined as follows –

$$\tau_r = \left\{ (P, L(P)) \Big|_{\max_{P \in DD_r(P_i)} i} \right\}. \tag{3.3}$$

The *border* or *boundary* of τ_r, denoted by $\partial\tau_r$, is defined as

$$\partial\tau_r = \left\{ (P, L(P)) \quad \text{at least one of 4 - neighbors of } P \text{ has label } 0 \right\} \tag{3.4}$$

The *interior* of τ_r, denoted by $int\,\tau_r$ is defined as

$$int\,\tau_r = \tau_r - \partial\tau_r. \tag{3.5}$$

Let the set of labels occurring in $\partial\tau_r$ be $\lambda(\partial\tau_r) = \left\{ L(P) \middle| (P, L(P)) \in \partial\tau_r \right\}$. Then P_i is said to be an *r-extreme pixel* in S if and only if $i \in \lambda(\partial\tau_r)$.

Consider 8-connected components of $\partial\tau_r$, each having the same label. The number of such components is more than or equal to $\#\lambda(\partial\tau_r)$. In general, a component with two or more pixels will have exactly two end pixels, viz. the pixels having exactly one 8-neighbor in the component.

Let ς be the set of *r*-extreme pixels in *S*. For P_i and $P_j \in \varsigma$, $\overline{P_iP_j}$ will be an *r*-edge of *S* if an end pixel of a connected component with label i in $\partial\tau_r$ and an end pixel of another connected component with label j ($i \neq j$) are 8-connected. Then P_j is an *r-adjacent pixel* of the *r-vertex* P_i. The degree of *r*-vertex P_i is then the total number of *r*-adjacents of P_i.

The digital *r*-shape of S is expressed by a diagram consisting of *r*-edges where the length of the structuring radius, r is equal to $\bar{r} = \lfloor \hat{s}\sqrt{2} \rfloor$ ($\hat{s}$ is the structuring grid-length of the dot pattern).

3.3.1 Computation of the r-shape

Step 1 : Create the discrete structuring disk with structuring radius $r = \bar{r}$.

Step 2 : τ_r is computed by placing the structuring disk as a mask at every point $P_i(x_i, y_i)$ as center (i = 1, 2, .., n) as follows. Each pixel Q in the square neighborhood (of side-length $2r+1$) of P_i is considered. If Q is under the disk $DD_r(P_i)$ then the label i is assigned to Q.

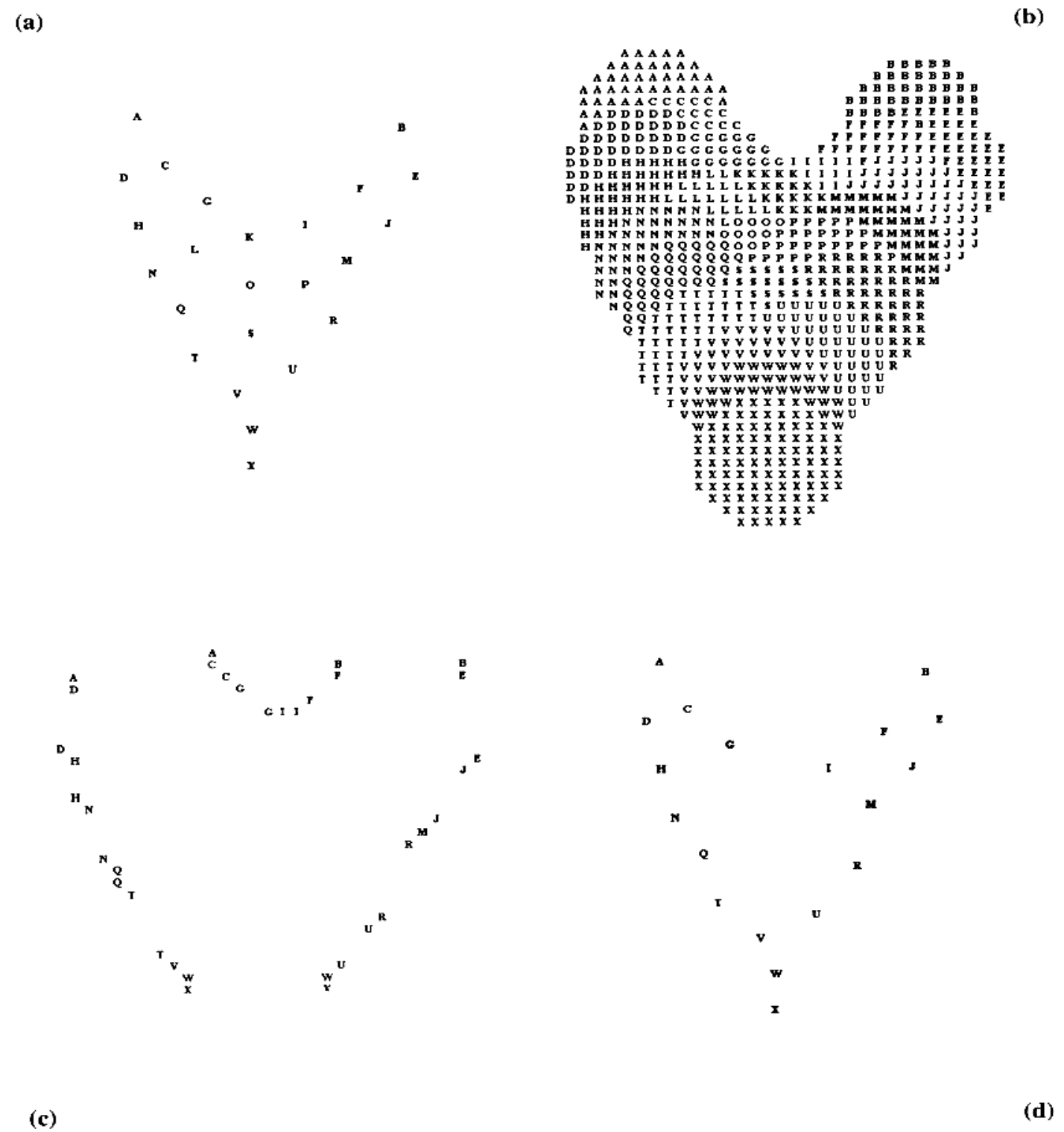


Figure 3.2 *Computation of the digital r-shape.*

(a) *The labeling of pixels by a raster scan;* **(b)** τ_r, *the union of labeled digital disks;* **(c)** *The label of detected r-adjacent vertices;* **(d)** *The resultant r-shape.*

Step 3 : The square neighborhood (of side-length $2r+1$) of P_i in τ_r is scanned for each i. If in that neighborhood, a pixel P has label i such that one of its 4-neighbors is a background pixel then P_i is an r-extreme pixel. Also, if such a P has a 8-neighbor Q with distinct label j ($i \neq j > 0$) such that Q has also a 4-neighbor background pixel then P_i is an r-vertex and $\overline{P_i P_j}$ is an r-edge of S. But if such a Q does not exist for any P, then that r-extreme pixel is considered as an isolated one.

For P_i, all the mutually distinct labels of the Q s' with properties stated above are found and the i-th row of successor listing is constructed. Let the stored labels be i_j ($j = 1, 2, ..., k$) where $i_j \in \{1, 2, ..., i\text{-}1, i\text{+}1, ..., n\}$. Then the r-adjacents of the r-vertex P_i are $P_{i_1}, P_{i_2}, ..., P_{i_k}$. The r-shape is a combination of r-edges which can be drawn using the successor listing. In this case, the successor listing is composed of n linear arrays where the i-th array contains the label of r-adjacent(s) of P_i. If the i-th array of the successor listing contains j ($> i$), then the r-edge $\overline{P_i P_j}$ is drawn. (Thus, each edge in the r-shape diagram is drawn only once.)

The r-shape computation is illustrated in Figure 3.2 For easy understanding and representation, English alphabets (A - X) are used on the 24 pixels present in the pattern. The pixels labeled by a raster scan are shown in Figure 3.2(a). The output of Step 2, namely τ_r is shown in Figure 3.2(b). The label of r-adjacent vertices detected in Step 3 are shown in Figure 3.2(c). The resultant r-shape is shown in the diagram of Figure 3.2(d).

3.4 Extraction of Perceived Border from the r-Shape

To get the border from the r-shape, at first, the structuring radius is estimated. With the estimated value of $r = \bar{r}$, the r-shape is computed.

3.4.1 The r-shape listing

As stated earlier, due to local irregularities, even under structuring radius r, some edges in an r-shape may appear as inconsistent with respect to the perceived boundary of the DP. A modified successor listing, referred to as r-shape listing is constructed to trace these edges and is denoted as $List_r(S)$.

label	↦	*deg*	*deg′*	*string* →		
A	↦	2	2	C	D	
B	↦	2	2	F	E	
C	↦	2	2	A	G	
D	↦	2	2	A	H	
E	↦	2	2	B	J	
F	↦	2	2	B	I	
G	↦	2	2	C	I	
H	↦	2	2	D	N	
I	↦	2	2	G	F	
J	↦	2	2	E	M	
K	↦	0	0			
L	↦	0	0			
M	↦	2	2	J	R	
N	↦	2	2	H	Q	
O	↦	0	0			
P	↦	0	0			
Q	↦	2	2	N	T	
R	↦	2	2	M	U	
S	↦	0	0			
T	↦	2	2	Q	V	
U	↦	2	2	R	W	
V	↦	2	2	T	W	
W	↦	3	3	V	X	U
X	↦	1	1	W		

r-shape listing $\left(list_r(S)\right)$

linear struc. →	X	W							
loop →	W	V	T	Q	N	H	D	A	C
									G
	W	U	R	M	J	E	B	F	I

Extracted border

Figure 3.3 *The r-shape listing and the extracted border in the form of strings.*

This list can be stored in a matrix (say, L) of order $n \times \lceil 2\pi(r+1) \rceil$ where each of the first two entries of i-th ($i = 1, 2,..., n$) row i.e. $L[i,1]$ and $L[i, 2]$ contains the degree of P_i and the remaining entries are the labels of its r-adjacent vertices. The first two entries of each row is henceforth respectively referred as *deg*, *deg'* and the remaining entries constitute the *string of r-adjacents* (or simply *adjacent string*).

The initial choice of $L[i, 1] = L[i, 2] = deg = deg'$ for each r-vertex of the pattern is justified in the following algorithm on border cleaning.

3.4.2 Finding the consistent edges in r-shape

After constructing the $List_r(S)$, the algorithm for selecting the consistent r-edges and removing the inconsistent ones from r-shape is executed through structural analysis using the following three steps: (1) tracing of *primitive linear structures* (2) selection of path(s) from multi-route in between branch vertices, and (3) removal of simple closed loops of negligible length. Note that from this r-shape listing all possible paths from an r-vertex can be found out.

In step (1), a primitive linear structure means a path of r-vertices whose one end is free i.e. a vertex of degree one while the other end is either another vertex of degree one (means a simple linear structure) or a branch vertex (means a tail-like structure). A linear structure can be found starting from an r-vertex of degree one and including the successor r-adjacent vertices till a vertex of degree one or a branch vertex is found. Let this output string be denoted by *lin-str*. An efficient *lin-str* computational approach is described below.

At first, one free end of a linear structure is detected (provided it exists) by checking the first column of L. Let P_i be such an end vertex. Then, both $deg(i)$ and $deg'(i)$ are assigned the value 0 and the first entry of *lin-str* is set to i. Now, the label (say i_1,) of the adjacent vertex, P_{i_1} of P_i on the linear structure is the value of $L[i, 3]$ (which is the first and only non-zero value of i-th string of r-adjacents). So, $lin\text{-}str[2] = i_1$ is taken. Obtain the position with value i, in i_1-th string of r-adjacent vertices. If $L[i_1, 2+l]$ (for $l > 0$) is equal to i then replace its value by 0, and swap it with the content of $L[i_1, 2 + deg'(i_1)]$. This processing has been done to avoid redundant search because with this operation a null

entry occurs only at the end of an adjacent string. The above way of removal of traced r-vertex label from the string of r-adjacents is referred to as *conditional popping*.

After the conditional popping, $deg'(i_1)$ is reduced by 1. If $deg(i_1)$ is equal to 2, both $deg(i_1)$ and $deg'(i_1)$ are set to 0 and if $L[i_1, 3] = i_2$, then we have to set $lin\text{-}str[3] = i_2$.

This process is continued until a label, say i_k, is found whose deg is not equal to 2. Then P_{i_k} is either a branch vertex (if $deg(i_k) > 2$) (which means the linear structure is tail-like), or another free end (if $deg(i_k) = 1$) of the simple linear structure under consideration. Note that the *length* of the $lin\text{-}str[i, i_1, i_2, ..., i_k]$ minus 1 is the number of r-edges in the linear structure (which is k here). The diagram of the traced primitive linear structure (which is a part of the r-shape), can be drawn by following the successor labels in $lin\text{-}str\ [i, i_1, i_2, .., i_k]$ and joining the corresponding vertices.

When all the edges lying on the primitive linear structures are traced and removed, the $List_r(S)$ contains only the edges on paths in between branch vertices and the edges in the simple closed loops. Note that any remaining path from a branch vertex always ends at another branch vertex. (A loop may be formed at the same branch vertex.)

Since the perimeter of a discrete disk of radius r is of order $\boldsymbol{O}(\lceil 2\pi r \rceil)$, the degree of an r-vertex is also of order $\boldsymbol{O}(\lceil 2\pi r \rceil)$. Thus, all paths from a branch vertex which are not (already traced) tail-like linear structure, can be stored in a matrix (say,) M of size $k \times n$, where $k > \lceil 2\pi r \rceil$. For a dense pattern, we have $k << n$.

Except at the starting vertex, the tracing procedure of multi-paths in between branch vertices is exactly the same as that for primitive linear structure. The deg column of $List_r(S)$ is scanned. If for b-th row we have $deg(b) > 2$ and $deg'(b) > 0$ then the vertex P_b is considered as the starting branch vertex and is tagged by a pointer (let us call it *branch vertex pointer*).

The label of end branch vertices to which P_b is path connected, the number of r-edges on the path and their row position in M are stored in three linear arrays, (say,) X, Y and Z, respectively.

From X, the existence of multi-routes between two branch vertices can be easily found. If b' is the label of an end branch vertex in X such that it repeats t-times then it signifies

that there exist t number of paths between P_b and $P_{b'}$ $(b < b')$. The path of minimum length between P_b and $P_{b'}$ can be found by checking the values in Y and corresponding Z. Consider the rest of the paths between P_b and $P_{b'}$. Any one such path and the shortest path make a closed loop. If the total number of r-edges on this loop is less than 6 then the path is considered as inconsistent.

The consistent paths from the branch vertex (which is at present tagged by the pointer) are identified by the labels of Z. Now, each consistent path can be drawn by following the successor labels of r-adjacent vertices ordered in the row of M identified by Z.

In this way, all branch vertices and their connecting paths are detected and r-edges on these paths are removed from $List_r(S)$. Note that if there are multi-routes between two branch vertices, the tracing over the path never repeats. For example, if for path connected branch vertices P_b and $P_{b'}$, the label b is less than the label b' then all paths between P_b and $P_{b'}$ are processed when the branch vertex processing pointer is at b-th row of $List_r(S)$.

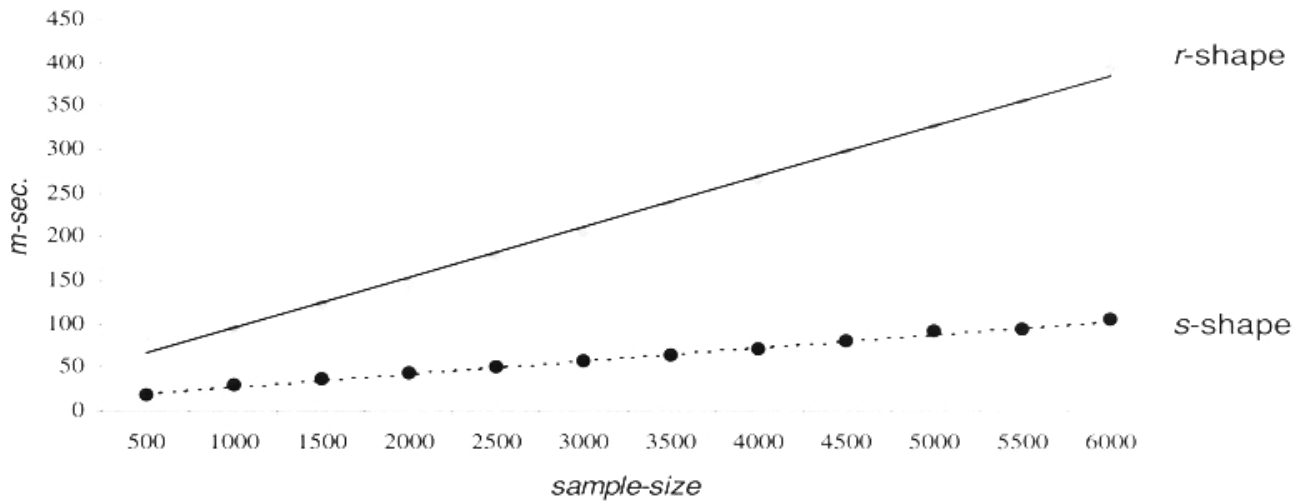
The rest in the $List_r(S)$ correspond only to those edges that lie on simple closed loop where each r-vertex is of degree 2. Now, to search an r-vertex on such a closed loop, deg' column is scanned from the top. If P_i is the lowest labeled r-vertex on a simple loop then $deg'(i) = 2$. This label i is placed on the first position of the output string. Both $deg(i)$ and $deg'(i)$ are set to zero. The first member of string of adjacents, say i' is tagged as the label of the end r-vertex. If the label of the second adjacent vertex of P_i is i_1, then i_1 is stored in the second position of the output string. After conditional popping at the i_1-th adjacent string, the other adjacent vertex of P_{i_1} (say, P_{i_2}) is found. Until an r-vertex of label i' is found, the same procedure i.e. conditional popping and storing at the output string is repeated. If the number of r-edges is less than 6, the r-edges on the loop are considered as inconsistent. Otherwise, by tracing the successive labels in the output string, the corresponding r-edges can be joined by line segments. In this way, all existing simple closed loops are traced out.

The resulting diagram consisting of remaining r-edges, are considered as the perceived border of S. The r-shape listing, the extracted linear structure as well as the (valid) loop existing in the r-shape of the dot pattern of Figure 3.2(d) (at a branch vertex labeled by 'W') are presented in Figure 3.3. (In this figure isolated linear structure and simple closed loop are absent.)

Note that for the re-generation purpose of the *r*-shape, only the *r*-connected points with respective connectivity information (including the edge-consistency) are stored.

Test case	No of Samples	Required time for *s*-shape computation in μ–sec.	Required time for *r*-shape computation in μ–sec.
1	500	19241	85101
2	1000	28885	94010
3	1500	36389	123187
4	2000	43898	145301
5	2500	51599	177875
6	3000	58329	206100
7	3500	65693	235870
8	4000	72650	264710
9	4500	79645	299390
10	5000	92126	329962
11	5500	94112	356237
12	6000	106238	394275

(a) *The table representing s-shape and r-shape computation time in micro second for samples randomly taken from the Figure 6.1*



(b) *The linear regressions corresponding to values in the above table.*

Figure 3.4 *A comparative run-time evaluation between s-shape and r-shape*

3.5 Evaluation of Proposed Approaches

3.5.1 Computational complexity analysis

To get an idea about the actual run-time required for patterns and how it changes with varying sample sizes for the *s*-shape and the *r*-shape, dots are randomly selected from Figure 6.1 (Some patterns with different sample sizes are shown in Figure 6.2). The table in Figure 3.4(a) shows the *s*-shape and the *r*-shape computation time (for a given structuring grid length and its corresponding structuring radius, respectively) in micro-seconds for sample sizes in the range of 500 to 6000. Naturally, the *s*-shape computation time is lower than that for *r*-shape, as reflected in the graph shown in Figure 3.4(b).

Next, We shall analytically establish that the computation of *r*-shape as well as the extraction of the perceived border, as a whole, is possible in linear time. At first, the complexity of *r*-shape computation is examined.

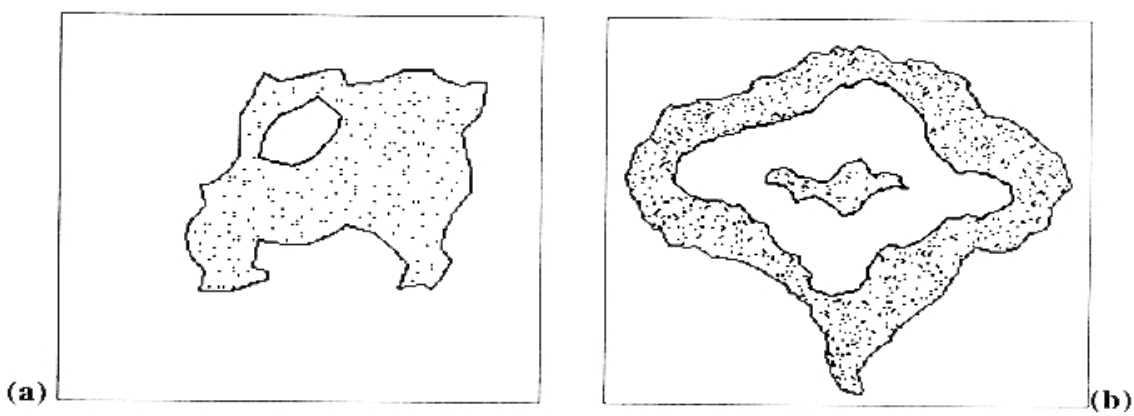


Figure 3.5 *r-shapes without any inconsistent edges*

3.5.1.1 The r-shape computational complexity

Since generation of a discrete disk with radius r at each point of DP requires a constant time $O(r^2)$, τ_r can be computed in $O(n)$ time. As described in the algorithm on *r*-shape computation, for each point P_i, pixels in its square neighborhood with sides of length $2r+1$ are scanned and processed. If P is a pixel in τ_r with label i in the square neighborhood of P_i then its 8-neighbors are also traced. If in the 8-neighborhood of P, there exist a background pixel as well as a pixel Q with a label distinct from i, then the 4-neighborhood of Q is scanned to detect whether Q has any background pixel. So, the worst case

computation for each pixel in square neighborhood of P_i requires 32 (= 8× 4) comparisons. Thus, to check whether a pixel is an r-interior, or an isolated r-extreme pixel or an r-vertex, as well as finding its r-adjacents requires a constant time. Therefore, the r-edges of r-shape can be detected in $O(n)$ time.

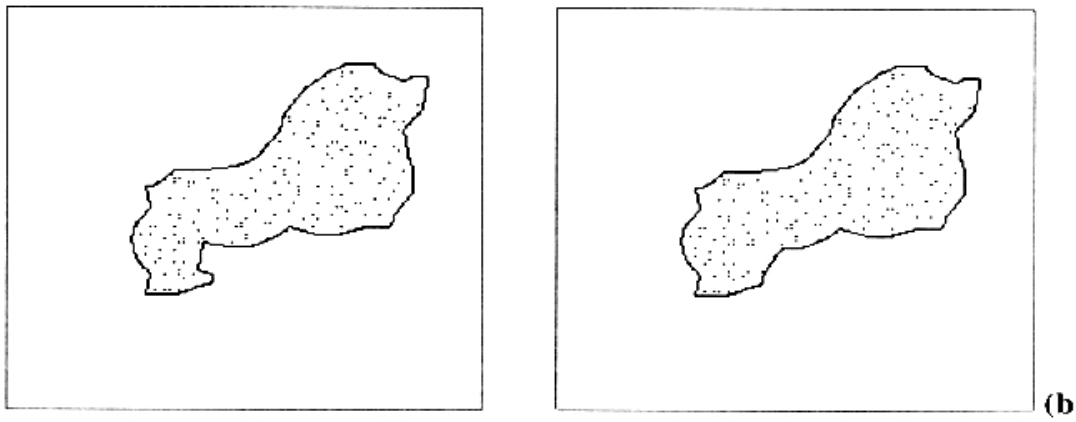


Figure 3.6 *Compatibility of r-shape with the perceived border.*
(a) $\varepsilon = 0.3$ (0.4) **(b)** $\varepsilon = 0.5$.

3.5.1.2 Perceived border extraction complexity

As discussed earlier in Section 2.3.3, the structuring radius can be estimated in linear time with respect to n.

The required time for computing the listing, $List_r(S)$ is also linear with n. Since degree of an r-vertex is $O(\lceil 2\pi r \rceil)$, in the worst case situation, the total number of r-edges is $O(\lceil 2\pi r \rceil \times n) \approx O(n)$. Now, recall that each r-edge of r-shape is traced over $List_r(S)$ only once. So this can be easily verified from the detailed description of the algorithm that each of the 3 modules namely tracing over the linear structure, selection of path(s) from multi-route between two branch vertices and finally the selection from simple closed loops, require only $O(n)$ computation. Thus, the computation of finding the consistent edges of r-shape requires $O(n)$ time. From the above discussion it is concluded that the time complexity of border extraction is linear with respect to the cardinality of the pattern.

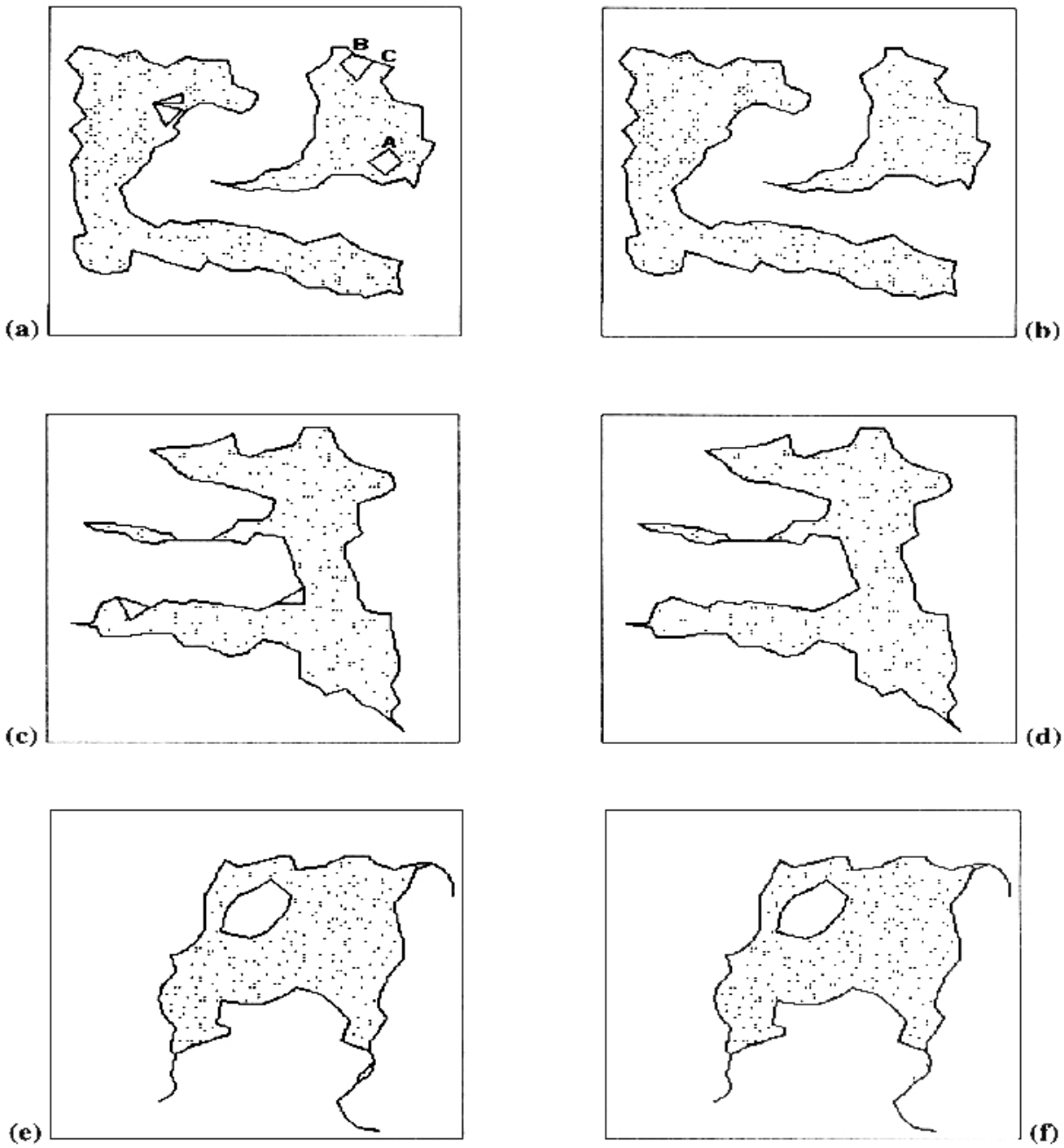


Figure 3.7 *Few examples where r-shapes are not free from inconsistent edges* **

** In each row, the left figure has inconsistent edge(s); the right figures are obtained by edge prunning described in Section 3.4.2

3.5.2 Experimental results and discussion

We have tested the proposed algorithms on several synthetic dot patterns. The results are quite satisfactory. As discussed in Section 2.4, we have experimentally found that if ε lies in the range 0.3 to 0.5, the extracted border is compatible with the perceptual border of the dot pattern. For a given DP (Figure 2.7) the *r*-shape based extracted borders for ε = 0.3, 0.4, 0.5 (The shapes for ε = 0.3, 0.4 are the same) are shown in Figure 3.6(a) and 3.6(b), respectively (The input patterns are presented in Chapter 2). Over a few other typical dot patterns where the extracted borders are proper subgraphs of respective *r*-shapes, are presented in Figure 3.7. The borders resulting from *s*-shape and *r*-shape are comparable.

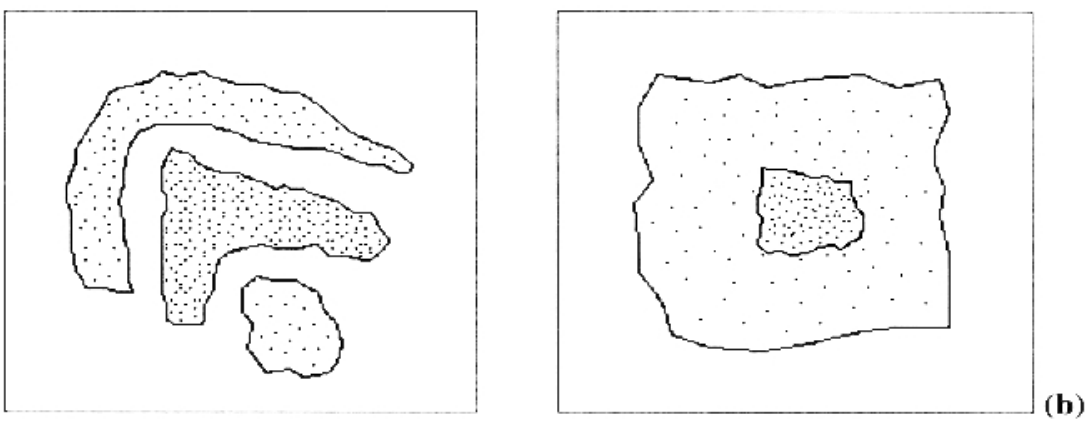


Figure 3.8 *Two complex dot patterns with their r-shape based borders.*

For a given dense dot pattern, our proposed *r*-shape based border generator is on average a $O(n)$ time algorithm. On the other hand, any algorithm for border extraction based on Voronoi diagram [Ahuja 82], Delaunay triangulation [Edelsbrunner et.al.83] or Sphere of influence graph [Toussaint 88] needs at least $O(n \log n)$ computation [Prepareta & Shamos 85].

In digital case, the number of iterations required for getting the structuring radius is very small. For example, in all the above listed figures of *r*-shape, the number of iterations required is less than 4. Except for a very few operations, the computation is based on integer addition, subtraction and comparison. Thus, our method of border extraction of dot pattern is very fast and efficient.

The proposed algorithms on structuring radius estimation and r-shape (r-shape listing) are locally computable and can be directly implemented on parallel machine. In [RayChaudhuri et.al. 94], where the concept of r-shape was first introduced, we proposed an algorithm on r-shape computation that could be implemented in Single Instruction Multiple Data computers viz., on CRCW-SM (Concurrent Read, Concurrent Write, Shared Memory) computer model.

The r-shape based boundary extraction algorithm is directly applicable for disconnected dot patterns as in case of s-shape. For example, in Figure 3.6(b) and Figure 3.7(a), the number of such disconnected dot patterns is 2. However, if pattern set contains components of different ε-dispersion measures i.e. component wise their structuring radii are different, the cut one out approach as discussed in the previous chapter, is adopted (Figure 3.8). This approach is also recommended for separating components if the pattern is degraded by (pepper) noise.

3.5.2.1 Application to cell nucleus shape detection

As mentioned in the Introduction, dot pattern shape analysis has possible applications in various real-life problems. Here, we illustrate one in the area of Cytology.

Cellular organelles are components located inside the cell cytoplasm having defined shape, size and function. Among these the nucleus is the most important one. The phospholipid bi-layer of a membrane that circumscribe the nucleus is a flexible structure that responds to a variety of environmental or internal cues and consequently changes its shape and size. In Figure 3.9(a) part of the human cell of squamous epithelium is shown. Note that the nucleus shape as well as the distribution of pigment granules in squamous epithelium is a determinant factor in some skin diseases including cancer. In a malignant cell the nucleus boundary appears distorted and even splits into small nuclei. The order of deformation gives clue to the level of degradation caused.

The slide that is photographed is prepared by immuno fluorescent method. In the figure the pigment granules (dark spots) and the nucleolus (gray region) within the nucleus are clearly visible. But due to poor immuno affinity at the nucleus membrane, it is hard to differentiate the nucleus from the cytoplasm. However, as dark spotted pigment granules are densely surrounding the nucleus, its border can be efficiently estimated as follows.

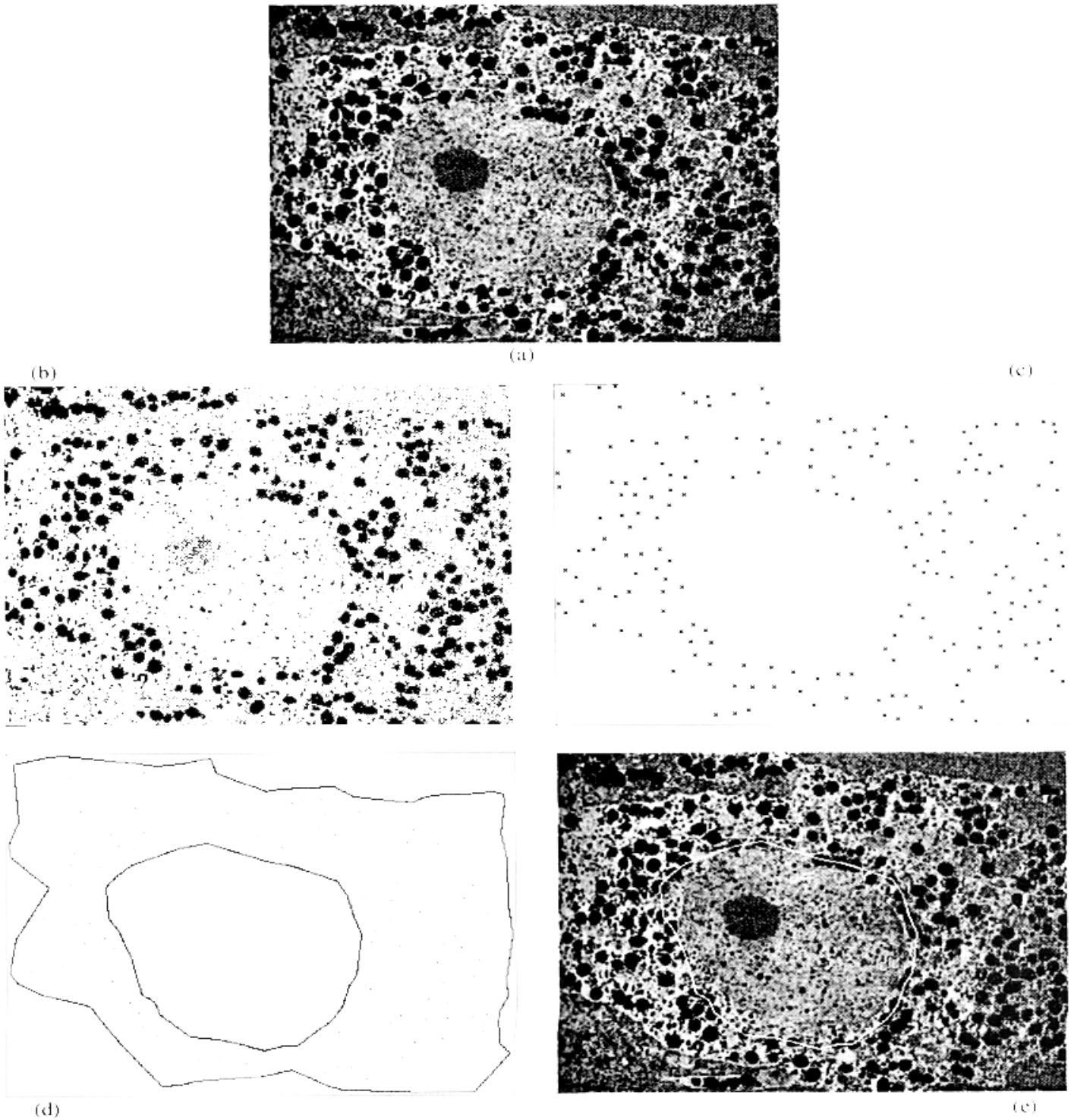


Figure 3.9 *An example where r-shape has been used to find out the nucleus border. (a) Original cell-image (b) Thresholded binary image. (c) The representative dot pattern over the filtered image (d) The r-shape (e) The estimated border of the nucleus*

Step 1: Convert the gray-scale image to binary by a simple thresholding (Figure 3.9(b)). As some part of the nucleolus with some noise are still as object pixels in the image, it is further filtered by an morphological opening (The structuring disk of

radius 2 is applied in this image and the gray region in the Figure 3.9(c) represents the filtered version).

Step 2: Find representative points (pixels) for each component. The output of this step is a digital dot pattern surrounding the nucleus (each dot is represented by a cross for better visibility in Figure 3.9(c)).

Step 3: The value of r is estimated and the smooth r-shape of the pattern is generated (As the distribution of granules is not very uniform, the closeness factor ε is set to 0.5. The result is shown in Figure 3.9(d)).

Step 4: Finally, the closed border that does not surround any single dot of the pattern is taken and superposed over the original pattern. The result is shown in Figure 3.9(e).

Chapter 4.

A Morphological Approach to Shape Description of Dot Patterns

4.1 Introduction

In this chapter, we consider (mathematical) morphology based approaches to recover the underlying structure of dot pattern. The approach is based on the concept of r-shape and the dispersion matrix of the dot pattern, proposed in the earlier chapters.

Because of its theoretical potential and implementation ease in parallel system, mathematical morphology plays a major role in image processing and pattern recognition problems [Serra 82], [Sternberg 86], [Haralick et.al. 87], [Dougherty 94]. The fundamentals of mathematical morphology are provided in the Appendix. Recently theory of mathematical morphology is also extended in analysis of graphs [Vincent 89], [Bertin et.al. 92]. However, we have not come across any published article on morphological algorithms for shape computation of dot patterns except [Worring & Smeulders 94]. In this paper, the α-hull of a input pattern X is expressed as the binary morphological closing of X by the generalized disk (Section 3.2).

Among two aspects of shape of a regular dot pattern, the external shape is considered in the previous two chapters. But the other aspect, internal shape is the main concern of this chapter, although external shape(border) extraction is also considered.

The internal shape descriptor tries to capture the core of that underlying region in the form of line-like skeletal representation. Some related works are [Zahn 71], [Parui et.al. 95], [Ogniewicz 91] and [Faugeras 93]. (The last two works mentioned have been presented for binary images but can be applicable for dot patterns also with little modifications.) Here, an efficient skeletonization approach based on projection of dispersion matrix is proposed.

This chapter is organized as follows. In Section 4.2, the r-shape is interpreted by binary morphological operators (Section 4.2.1), and an algorithm on border extraction is

presented in digital case (Section 4.2.2). A simple but effective extreme point extractor is developed by binary morphology. With structuring radius, $r = \bar{r}$ (see Definition 3.5) the r-shape is formed by joining these extreme points under certain constraints. To represent the border of the dot pattern the border of the bounded region of r-shape is smoothed by morphological filters. It is observed that this border is compatible with the perceptual structure of the pattern. A new approach to find the skeletal representation of dot pattern is proposed in Section 4.3. The binary projection of the dispersion matrix which is closely related to the notion of s-shape is used for that purpose. It is found that the binary digital image resulting from the dispersion matrix, if smoothed by morphological filters (as described in Section 4.3.2), is a scale reduced digital image of the smoothed s-shape. At first, this image is morphologically thinned (the algorithm is presented in Section. 4.3.1) and then stretched (described in Section 4.3.3) over the pattern. In Section 4.4, presentation of experimental results on few typical data sets is followed by a discussion.

4.2 External Shape (Border) Computation of Dot Pattern

In Section 2.3.2, for s-shape based border extraction we applied morphological operator namely binary closing. To obtain a morphology based algorithm for smoother polygonal border of dot pattern, at first the r-shape is interpreted by morphological operators.

4.2.1 The r-shape interpretation by mathematical morphology

Let $S = \{P_1, P_2, ..., P_i, ..., P_n\}$ be a set of n points in $\Re^2$ and let r be a positive quantity. As mentioned earlier, $D_r(Q) \subset \Re^2$ is the closed disk with radius r and center Q; whereas, $C_r(Q)$ and $intD_r(Q)$ denote the boundary and interior of $D_r(Q)$. When Q is (0,0), $D_r(Q)$ and $C_r(Q)$ are simply denoted by D_r and C_r respectively. Consider the isothetic rectangle W with minimum area covering S.

By a transform each point P_i is assigned a unique integer (gray) label say, i. Considering the disk, D_r as the region of support, a constant function assigns a (gray) label (say, $z \geq 0$) at each point on the disk. Dilate S by D_r in gray level [Sternberg 86]. Then, the existence of a pair of adjacent arcs (say, $Q'Q$, QQ'') having color $i+z$ and $j+z$ respectively, in the border of $S \oplus D_r$ will identify the r-edge $\overline{P_iP_j}$ (See Figure 3.2). However, the r-extreme points of S are more efficiently extracted by binary morphological transforms as follows.

Let

$$S_r^{\oplus} \cong S \oplus int\, D_r \tag{4.1}$$

$$S_r^{\bullet} \cong S_r^{\oplus} \ominus D_r \tag{4.2}$$

$$\partial S_r \cong S / S_r^{\bullet} \tag{4.3}$$

where the binary dilation and erosion operator are respectively denoted by $\oplus$ and $\ominus$. For visual understanding of $S_r^{\oplus}$, $S_r^{\bullet}$ and ∂S_r Figure 4.1 is referred.

Both the following Propositions can be easily verified.

Proposition 4.1 : ∂S_r contains all the r-extreme points of S. In fact, its the set of α-extreme points of S where $\alpha = -\frac{1}{r}$.

Let $\overline{S} \cong S \oplus D_r = \bigcup_{i=1}^{n} D_r(P_i)$ and $\partial\overline{S}$ denote the border of the region $\overline{S}$.

Proposition 4.2 : $\overline{S} = S_r^{\bullet} \cup (\partial S_r \oplus D_r)$.

Note that a point P_i of S is r-adjacent to another point P_j if $C_r(P_i) \cap C_r(P_j) \cap \partial\overline{S}$ is non-empty. Thus, the r-shape, $rsh(S)_r$ can be generated by joining pair wise r-adjacent points. Let $PS(S)_r \cong \#\partial S_r$, $r \in \Re^+$.

Definition 4.1 : The plot $\prec PS(S)_r, r \succ$ is said to be the dot pattern spectrum over S.

Definition 4.2 : The region bounded by the r-shape is called the r-hull and is denoted by $rhull(S)_r$.

Note that any bounded region (component) in the complement of $rhull(S)_r$ is considered as hole if it does not intersect the border of W, that is ∂W. Otherwise, it is just a part of the background of the dot pattern. The $rhull(S)_r$ can be found by growing $S_r^{\bullet}$ in the r-shape. For a 'bear' shaped dot pattern (Figure 4.1(a)), the $S_r^{\oplus}$, $S_r^{\bullet}$, $b(S)_r$, $rsh(S)_r$ and $rhull(S)_r$ are, respectively, illustrated in Figure 4.1(b-f).

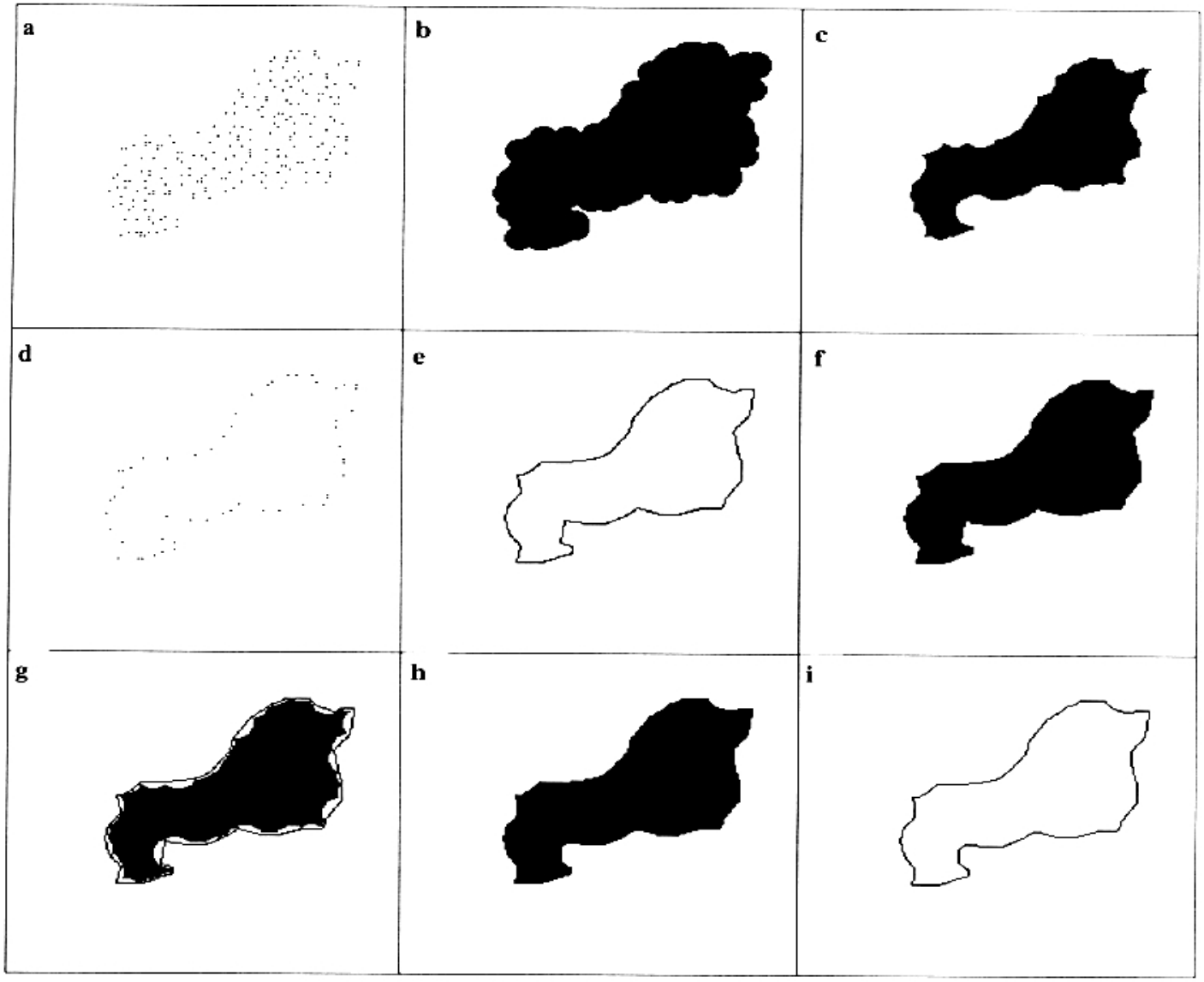


Figure 4.1 *Proposed approach of border extraction of dot pattern*

(a) *The bear shaped dot pattern*; **(b)** $S_r^{\oplus}$; **(c)** $S_r^{\bullet}$; **(d)** ∂S_r; **(e)** *the r-shape*; **(f)** *the r-hull*; **(g)** S'; **(h)** $S'^{\bullet}$; **(i)** *border of the dot pattern.*

Recall that due to local irregularity in the distribution of dots, some edges in r-shape (even for a structuring radius $r = \tilde{r}$) are redundant for border detection. When such edges are present, *inconsistent* holes of negligible size may appear in r-hull and the topology of the pattern may be changed. Here, a hole of r-hull is characterized as inconsistent in contrast with perceived shape, if the structuring disk of the pattern can not be contained in it. If the r-hull is morphologically operated by a binary closing where the structuring element is the disk of radius half of that structuring disk, all inconsistent hole are removed.

Let $r' = \left\lceil \frac{\hat{r}}{2} \right\rceil$.

Definition 4.3 : The output of r-hull by the binary closing with structuring element $D_{r'}$ is called the smoothed r-hull and is denoted by $\overline{rhull}(S)_r$.

Finally, the border of the smoothed r-hull is referred to be the border of the dot pattern.

4.2.2 Morphology based DP border computation in digital case

Let S be a binary image where almost all object pixels, $P_i = (x_i, y_i) \in \Im^2$ are isolated but densely and more or less evenly distributed in a subregion. Then S can be considered as an image of a dot pattern. Note that, the task of extracting the perceptual boundary of a image of dot pattern in digital domain can be divided into three major modules namely :

(*i.*) Estimation of structuring radius, $r = \hat{r}$.

(*ii.*) Computation of r-shape by the estimated r.

(*iii.*) Removal of inconsistency from border.

An algorithm on structuring radius estimation is given in Section 2.3.1. Thus, we omit the first module.

4.2.2.1 Dot pattern's border extraction algorithm

Let $\overline{DD_r} = DD_r \oplus DD_{0.5}$.

Step 1 $S_r^{\oplus} \leftarrow S \oplus DD_r$. (See Figure 4.1(b))

Step 2 $S_r^{\bullet} \leftarrow S \ominus \overline{DD_r}$. (See Figure 4.1(c))

Step 3 $\partial S_r \leftarrow S / S_r^{\bullet}$. (See Figure 4.1(d))

Here, ∂S_r is the set of *r-extreme pixels*.

Let $P_i, P_j \in \partial S_r$.

Step 4 For each r-extreme pixel $P_{i_k} \in DD_{2r}(P_i) \cap S$ with $(i_k > i)$,

if $\left(DC_r(P_i) \cap b\left(S_r^{\oplus}\right) \right) \cap DC_r\left(P_{i_k}\right) \neq \Phi$ then *digital r-edge* $\overline{P_i P_{i_k}}$ is drawn.

(See Figure 4.1(e))

After Step 4 we found the digital r-shape $rsh(S)_r$ the diagram consists of digital r-edges.

Note that the *digital r-hull* (see Figure 1(f)) is the set of pixels bounded by $rsh(S)_r$. As the thickness of the void region in between $rsh(S)_r$ and $S_r^{\bullet}$ is less than r, a smooth version of the digital r-hull (even without computing r-hull) can be easily found as follows.

Step 5 $S' \leftarrow rsh(S)_r \cup S_r^{\bullet}$. (See Figure 4.1(g))

Step 6 $S'^{\bullet} \leftarrow S' \bullet D_{r'}$; where $r' = \left\lceil \frac{r}{2} \right\rceil$; (See Figure 4.1(h))

Step 7 Find the border of $S'^{\bullet}$ that consist pixels each having a 4-neighbor in the complement of $S'^{\bullet}$. (See Figure 4.1(i))

Finally, the output of Step 7 is considered as the border of the dot pattern S.

4.3 Internal Shape (Skeleton) of Dot Pattern

The process of locally symmetrical width reduction of a pattern to line-like structures without altering connectivity or without appreciable distortion of shape in the input image, is called skeletonization or thinning. For image analysis (not dot pattern), thinning is a widely used technique. For a survey on conventional approaches see [Lam et.al. 92] and in particular morphology based algorithms [Jang & Chin 90] is referred.

As in continuous object, if the dot pattern is elongated, skeleton is a meaningful representation of the shape of the pattern. Delaunay triangulation [Faugeras 93] or Voronoi diagram [Ogniewicz 91] based skeleton representation techniques are computed over border (corner) points. Since the border points of a dot pattern, namely, r-extreme points can be

very efficiently extracted by ∂S_r, one way of computing the skeleton of a dot pattern is to apply one of these skeleton representation techniques on the set of r-extreme points.

If the underlying structure of a dot pattern is correctly recovered then the skeleton of this recovered structure can be considered as an internal shape representation of dot pattern as well. So, another approach of computing the internal shape of the DP would be the exploitation of existing smart morphological or conventional thinning algorithms over the smoothed r-hull of DP. (See Figure 4.8 where smoothed r-hull is thinned by the algorithm presented in [Rosenfeld & Kak 82]; the input pattern is shown in Figure 4.7).

In the following, we have proposed an alternative efficient way of thinning the dot pattern based on projection of dispersion matrix.

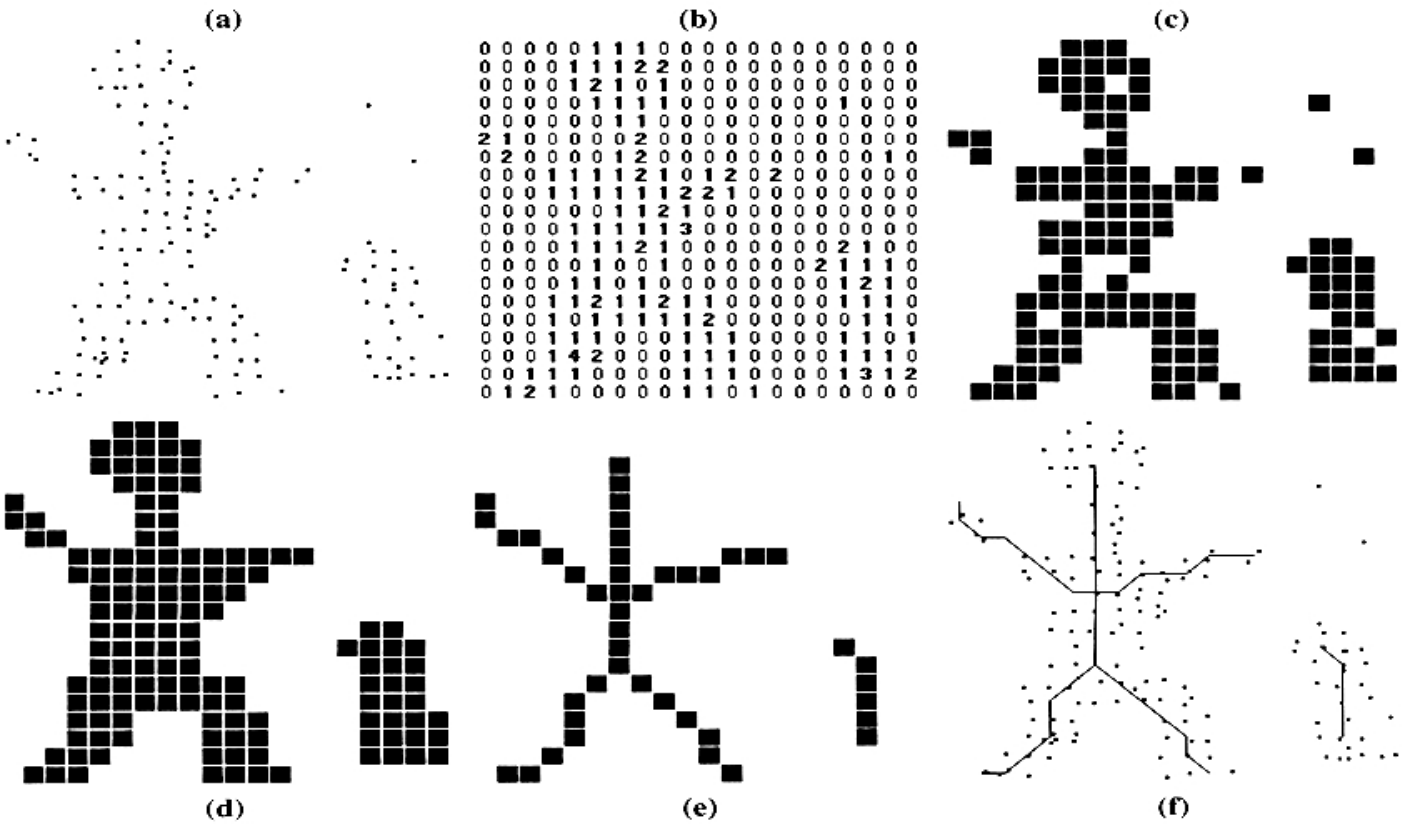


Figure 4.2 *Illustration of the proposed skeletonization of dot pattern.*
(a) *Dot pattern under the lattice with structuring grids;* **(b)** *dispersion matrix, DMAT(S);*
(c) *$DMAT_{prj}$ the binary projection of DMAT(S),each square grid represents a pixel in $DMAT_{prj}$;*
(d) *morphological smoothing of $DMAT_{prj}$;* **(e)** *thinning of (d);*
(f) *stretching and overlaying of the skeleton(e) on the dot pattern.*

4.3.1 Proposed approach of skeletonization of dot pattern

The dispersion matrix $DMAT(S)$ which is invariant under translation and scaling can be used as a structural basis of the dot pattern S. The binary projection (see Figure 4.2(c)) of dispersion matrix, $DMAT_{prj}$ (see Figure 4.2(b), where input dot pattern is shown Figure 4.2(a)) is considered as a binary image where each grid represents a pixel. The $DMAT_{prj}$ may be porous and cracked due to local irregularity of distribution of the pattern. In case of digital dot pattern (Section 2.3), it is a scaled (reduced) version of the hull induced by structuring grid where the scaling factor is the square of the structuring grid-length. It has been observed that the smoothed version (as described in Section 4.3.2) of the said projection can successfully capture the underlying structure of DP. Since in general, the topology of r-hull is well preserved in the smooth version, the skeleton of DP can be extracted by $DMAT_{prj}$ without actually computing the r-hull in the following manner. After taking care of pores and border-cracks in $DMAT_{prj}$ the resulting image is morphologically thinned. The proposed algorithm is discussed in 4.3.2. The skeleton is then overlaid by stretching (as described in Section 4.3.4) and considered as the skeleton of the dot pattern.

This approach has three-fold advantage:

i. Any smart binary image-thinning algorithm can be used on the $DMAT_{prj}$.

ii. Due to search space reduction, it is extremely fast.

iii. Extraneous spurs (parasitic branches) generally caused by thinning is minimized.

Note that, the domain of binary projection of dispersion matrix is $\Im^2$ and henceforth unless otherwise mentioned, we will consider thinning only over binary digital domain. Finally in Section 4.3.5 the dot pattern-thinning algorithm is presented.

4.3.2 A morphology based binary image thinning algorithm

Let A be a binary digital image. A boundary pixel P of A is not *removable*: (*i*) if its removal alters the connectivity, (in that case the pixel is referred to as a *critical pixel*), or (*ii*) if the removal shortens any leg of the skeleton (In that case the respective pixel has exactly one 8-connected neighbor and is referred to as a *end pixel*). Pixels that can be removed are called *simple pixels*.

Consider Figure 4.3 where the positions of (horizontal-vertical) 4-neighbors and of the rest (diagonal) 8-neighbors of P are respectively represented by {a,c,b,d} and {e,f,h,g} (in clock-wise direction).

e	c	f
a	P	b
g	d	h

Figure 4.3 *The 8-Neighboring positions of a pixel P*

Proposition 4.3 : If the neighborhood of P matches any of the four templates T_1 - T_4 (Figure 4.4(a)) (centering at P), then the following statements are true.

(a) (For template T_1) P is critical iff (c = 0 and f = 1) or (d = 0 and h = 1).

(b) (For template T_2) P is critical iff (a = 0 and e = 1) or (b = 0 and f = 1).

(c) (For template T_3) P is critical iff (c = 0 and e = 1) or (d = 0 and g = 1).

(d) (For template T_4) P is critical iff (a = 0 and g = 1) or (b = 0 and h = 1).

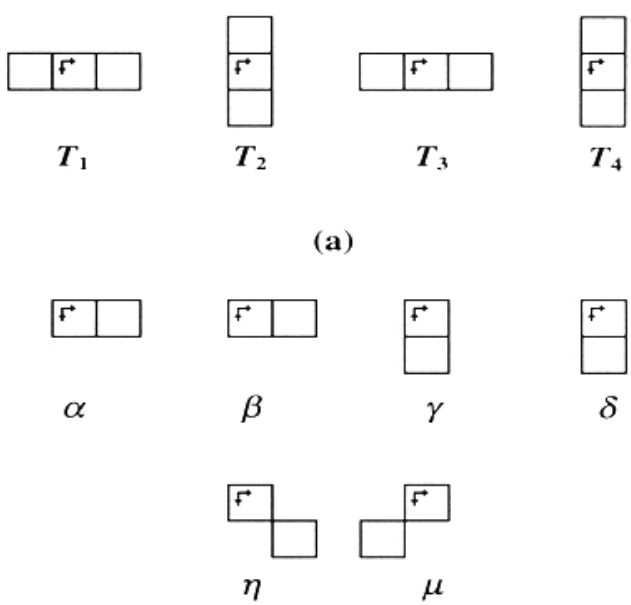


Figure 4.4 *The set of* **(a)** *templates and*
(b) *structuring elements used for thinning and stretching.*

Proposition 4.4 : If P matches template T_1 or T_3 then P is an end pixel iff e, c, f, g, d and h are zero. On the other hand, if P matches template T_2 or T_4 then P is an end pixel iff e, a, g, f, b and h are zero.

Our morphology based algorithm is based on the above two propositions [Datta & Parui 94].

Consider the six templates $(\alpha,\beta,\gamma,\delta,\eta,\mu)$ illustrated in Figure 4.4(b). A few composite transforms are defined as follows.

$$\Upsilon^{(\alpha,\beta)}_{(\gamma,\delta),(\eta,\mu)} = A \otimes (\alpha,\beta) \cap \left(A \otimes (\gamma,\delta) \cup A \otimes (\eta,\mu)\right) \tag{4.4}$$

$$\Omega_{\alpha,\beta} = A^c \ \Theta \ \left(DD_1 / (\alpha \cup \beta)\right) \tag{4.5}$$

$$\Psi^{(\alpha,\beta)}_{(\gamma,\delta),(\eta,\mu)} = A / \left(A \otimes (\alpha,\beta) / \left(\Upsilon^{(\alpha,\beta)}_{(\gamma,\delta),(\eta,\mu)}(A) \cup \Omega_{\alpha,\beta}(A)\right)\right) \tag{4.6}$$

$$\Sigma(A) = \Psi^{(\tilde{\gamma},\delta)}_{(\eta,\beta),(\mu,\tilde{\beta})}\left(\Psi^{(\alpha,\tilde{\beta})}_{(\tilde{\eta},\tilde{\delta}),(\mu,\delta)}\left(\Psi^{(\gamma,\tilde{\delta})}_{(\tilde{\eta},\tilde{\beta}),(\tilde{\mu},\beta)}\left(\Psi^{(\tilde{\alpha},\beta)}_{(\eta,\delta),(\tilde{\mu},\tilde{\delta})}\right)\right)\right) \tag{4.7}$$

('~' denotes the reflection operator; e.g. $\tilde{\gamma} = \{-x | x \in \gamma\}$)

$$\Gamma(A) = \Sigma\left(\Sigma\left(\ldots(\Sigma(A))\right)\right) \tag{4.8}$$

It can be shown that $\Psi^{(\tilde{\alpha},\beta)}_{(\eta,\delta),(\tilde{\mu},\tilde{\delta})}$, $\Psi^{(\gamma,\tilde{\delta})}_{(\tilde{\eta},\tilde{\beta}),(\tilde{\mu},\beta)}$, $\Psi^{(\alpha,\tilde{\beta})}_{(\tilde{\eta},\tilde{\delta}),(\mu,\delta)}$ and $\Psi^{(\tilde{\gamma},\delta)}_{(\eta,\beta),(\mu,\tilde{\beta})}$ remove all those simple border pixels of the input image matching templates T_1- T_4, respectively.

Using Proposition 4.3 and Proposition 4.4 following result can be established.

Proposition 4.5 : $\Gamma(A)$ is a unit pixel thick skeleton except at critical junction and preserves the connectivity as well as skeletal legs.

Each pixel in the skeleton has two neighbors unless it is an end pixel or a critical junction pixel. The advantage of the proposed thinning is that it does not require any post trimming operation of the skeleton like several existing parallel algorithms. (Often skeletal legs are not preserved due to such trimming [Jang & Chin 90].) Also, the closeness factor of the skeleton to the medial axis is quite high. However, as mentioned earlier any alternate efficient thinning of binary images may be applicable also.

4.3.3 Smoothing of dispersion matrix projection

As mentioned earlier, some inconsistent holes and border cracks with respect to the underlying structure generally appear in $DMAT_{prj}$ due to local irregularity in distribution of dots. If the size of a hole is less than 3×3 then it is considered as inconsistent. If these holes and border cracks exist, they can be removed by a closing (say, $DMAT_{prj}^{\bullet 1}$) with the unit-structuring disk, DD_1 (See Figure 2.6).

Many thinning algorithms [Datta & Parui 94] fail to tackle the object pixel that has only one diagonal neighbor which is 4-connected to other object pixel and often produce erroneous limb in the skeleton. Here, this problem is tackled by removing such pixels as follows.

$DMAT_{prj}^{\bullet 1}$ is opened by DD_1. The 8-isolated pixels of the residue $DMAT_{prj}^{\bullet 1}$ that do not lie in $DMAT_{prj}^{\bullet 1} \circ DD_1$ are considered. If for such a isolated pixel P, $DD_1(P)$ contains less than 2 pixels of $DMAT_{prj}$, P is removed from $DMAT_{prj}^{\bullet 1}$. Let the remaining of $DMAT_{prj}^{\bullet 1}$ be denoted by $smth\left(DMAT_{prj}\right)$. Note that $smth\left(DMAT_{prj}\right)$ is also free from 8-isolated pixel of $DMAT_{prj}$.

4.3.4 Stretching and overlaying of DMAT skeleton on DP

Let the skeleton of the $smth(DMAT_{prj})$, $\Gamma(\ smth(DMAT_{prj}))$ be henceforth simply denoted by Γ. An injective mapping from the Γ to the digital dot pattern space can be defined in the following manner. Consider a pixel P in Γ. Let its corresponding element in the matrix $DMAT$ be g_{lm}; where g_{lm} represents a grid say, g in $G(\hat{s})$. Let the mid-point of g be denoted as h_{lm}. To detect critical junctions, few composite transforms are defined as follows.

$$\Pi_{(\alpha,\eta)}(A) = \left((A \Theta \alpha) \cap (A \Theta \eta)\right) \oplus \eta \tag{4.9}$$

$$\Delta_1(A) = \Pi_{(\alpha,\eta)}(A) \cup \Pi_{(\tilde{\alpha},\tilde{\eta})}(A) \cup \Pi_{(\alpha,\tilde{\mu})}(A) \cup \Pi_{(\tilde{\alpha},\mu)}(A) \tag{4.10}$$

$$\Delta_2(A) = A \circ (\alpha \cup \gamma \cup \eta) \tag{4.11}$$

Note that $\Delta_2(A) \subset \Delta_1(A)$. It can be shown that $\Delta_1(\Gamma)$ contains all the pair wise diagonal 8-neighboring pixels of Γ that have also a critical junction as a common 4-neighbor

whereas $\Delta_2(\Gamma)$ contains the union of 2×2 sized squares. (In general, presence of such squares in the skeleton are very rare.)

For each pixel P and any of its 8-neighbor(s), say P' in Γ, consider their respective mid-pixels (say,) h_{lm}, $h_{l'm'}$. Now, the following actions are taken. (*i.*) h_{lm} and $h_{l'm'}$ are joined by line segment if both P, P' do not belong to $\Delta_1(\Gamma)$. (*ii.*) If both P, P' belong to $\Delta_1(\Gamma)$ but not to $\Delta_2(\Gamma)$, the corresponding h_{lm} and $h_{l'm'}$ are not connected. But if (*iii.*) P, P' belong also to $\Delta_2(\Gamma)$ where P and P' are diagonal neighbors, then h_{lm} and $h_{l'm'}$ are connected in the dot pattern space.

If h_{lm} and $h_{l'm'}$ are joined without any criteria like (*i.*)-(*iii.*) then at each junction of the skeleton of the DP, local loops will appear which are inconsistent with the topology of the skeleton Γ derived from dispersion matrix. Note that action (*ii.*) suppresses diagonal connection to maintain the 'L' shaped structure of the skeleton whereas action (*iii.*) suppresses the connection of midpoints of 4-neighbors in Γ but directs to connect the corresponding mid-points of the diagonal 8-neighbors in Γ at a 2×2 junction. The resulting diagram is considered as the skeleton representation of the dot pattern S. In the following, the outline of the proposed dot pattern skeleton algorithm is sketched.

4.3.5 DP thinning algorithm

Step1. Compute $DMAT_{prj}$ of the pattern. (See Figure 4.2(c))

Step2. Compute $smth\left(DMAT_{prj}\right)$ from $DMAT_{prj}$. (See Figure 4.2(d))

Step3. Compute the skeleton of $smth\left(DMAT_{prj}\right)$ by Γ transform. (See Figure 4.2(e))

Step4. Stretch the Γ and overlay the skeleton on the dot pattern. (See Figure 4.2(f))

4.4 Results and Discussion

A few typical examples of digital dot patterns (having holes, multi-components etc.) with their r-shapes and borders are shown in Figure 4.5. Except at concave corners, which are become blunt, these results are exactly same with corresponding results of Chapter 3.

Consider a 'Amaeba' shaped digital dot pattern having elongated limbs and two holes in Figure 4.6. The skeleton and the outputs at the different levels of our proposed algorithm is provided in Figure 4.7. The skeleton of the smoothed *r*-hull for the same DP presented in Figure 4.8 by a standard algorithm [Rosenfeld & Kak 82], are comparable.

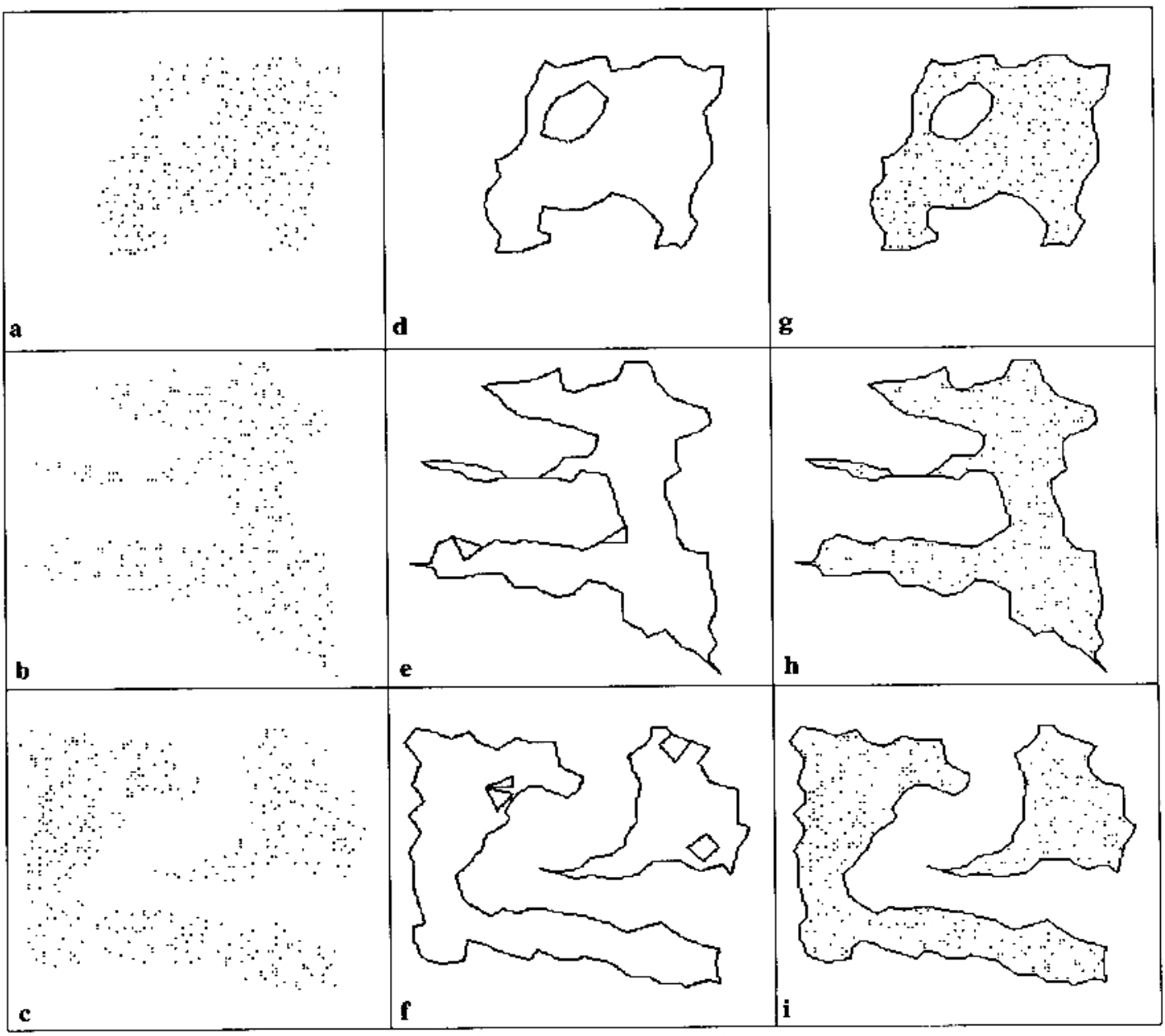


Figure 4.5 **(a)-(c)** *Examples of dot patterns; their respective* **(d)-(f)** *r-shape and* **(g)-(i)** *extracted border.*

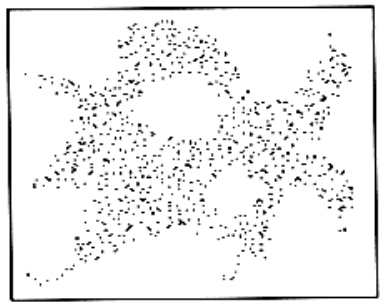


Figure 4.6 *A Ameba shaped pattern having elongated limbs and two holes.*

Figure 4.7 *Step by Step skeletonization of a digital dot pattern.*
(a) *Result of Step* 1; **(b)** *result of Step* 2; **(c)** *result of Step* 3; **(d)** *the final out put.*

Our proposed internal shape (skeleton) extractor of dot pattern is robust under noise. Subject to the proposed approach, skeletons obtained through our unified method in noisy environment (with 15 db SNR (Figure 4.9(a)) and 10 db SNR (Figure 4.9(b)) respectively) are quite satisfactory.

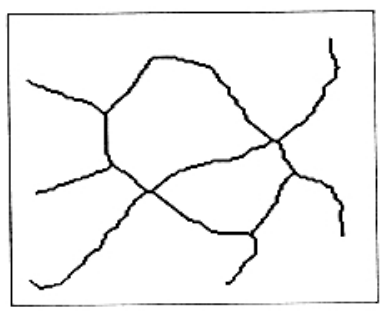


Figure 4.8 *Thinning of r-hull by a standard algorithm.*

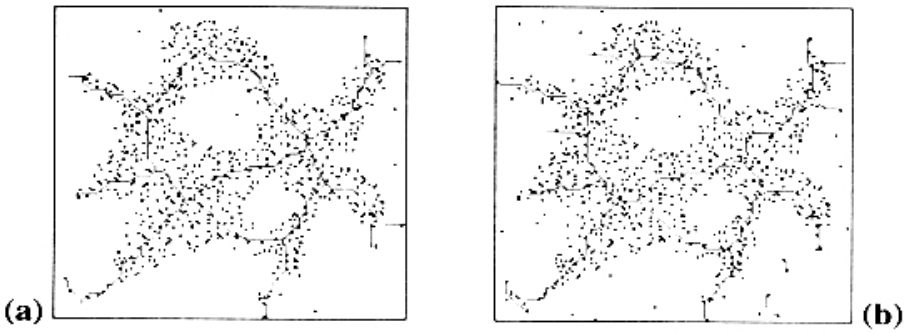


Figure 4.9 *Extracted skeletons on noisy patterns.*

Suppose some points of a dot pattern are locally displaced. The effectiveness of dispersion matrix as a structural basis is established on such cases of (locally) distorted dot patterns. For example, 50% and 95% of points of the original dot pattern have been randomly displaced in the local neighborhood of radius r $(= \bar{r})$ = 4, respectively. The distorted patterns with their respective skeletons are shown in Figure 4.10.

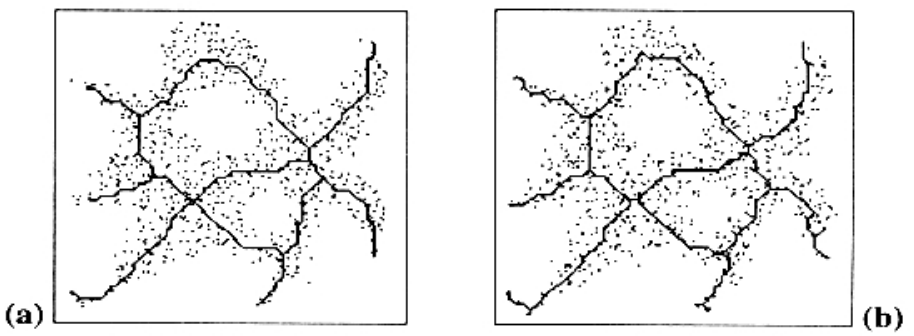


Figure 4.10 *Resulting skeletons over distorted patterns.*

Skeletons of a few other digital dot patterns having elongated structures are shown in Figure 4.11.

As discussed earlier, the number of iterations required for generating the projection of the dispersion matrix is very low. In all illustrated examples, the number is less than 4. It is apparent that the proposed border extraction of DP, especially using parallel processor, require moderate computation. As the underlying structure (*DMAT*'s projection) is a reduced size image of the underlying region (its approximation is the smoothed r-hull), the thinning process is considerably fast.

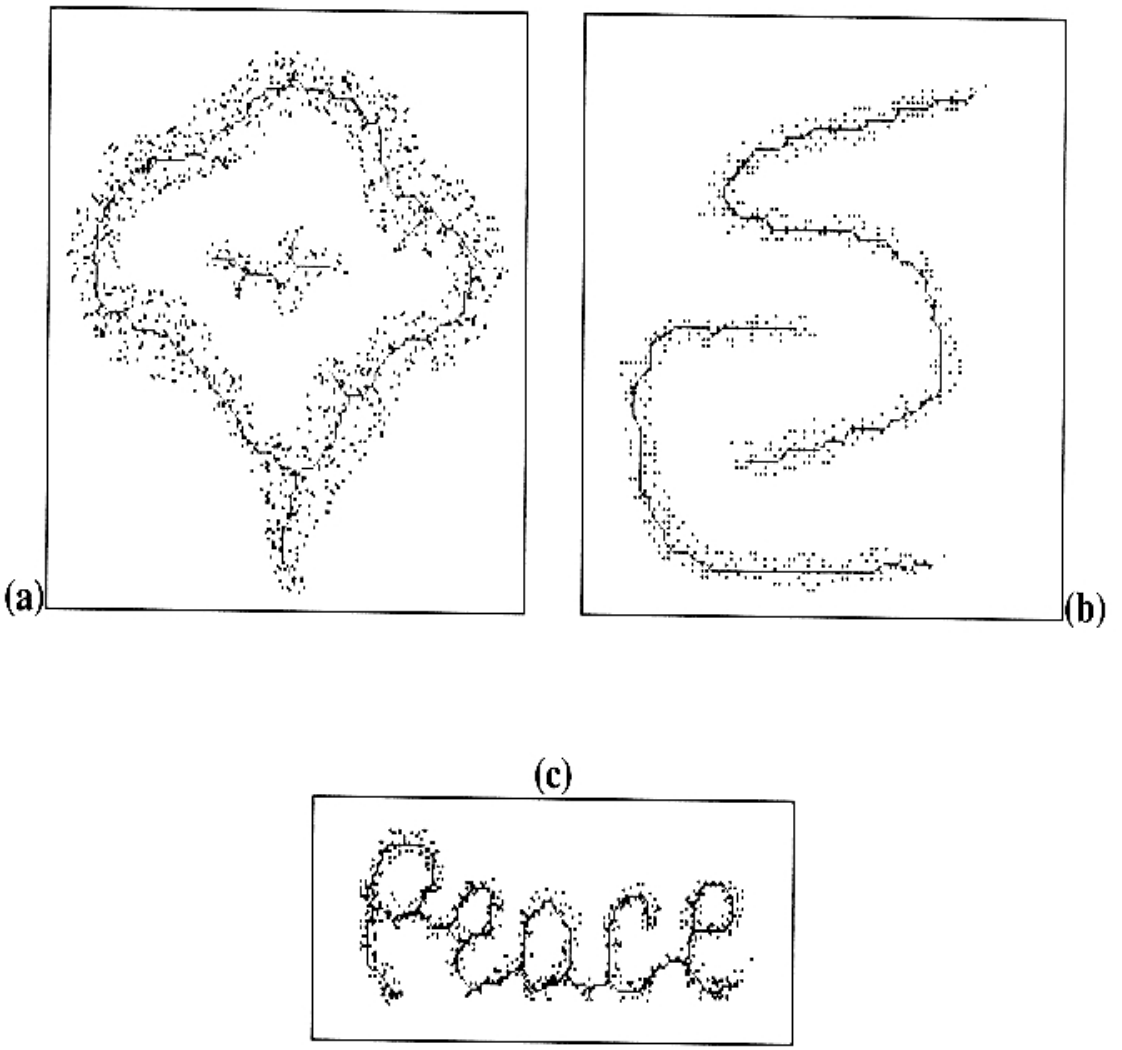


Figure 4.11 *A few examples of elongated dot patterns with their respective skeletons.*

Our approach can be extended to data in higher dimensions. As skeletons of dispersion matrices of dot patterns which are evolved by translation, scaling, addition of noise or local distortion of a dot pattern are remain unaffected, thinning of dispersion matrices may be effectively used for dot pattern classification and pattern matching problem.

Chapter 5.

A New *K*-Nearest Neighbor Based Approach on Clustering and Shape Computation of Complex Dot Patterns

5.1 Introduction

In this chapter, finding of borders of nonlinear components with variable density, as well as arcs belonging to linear parts of dot pattern is of concern. Recall that a component of a dot pattern is curve-like or linear if the shape looks like a line (technically, the component has locally one degree of freedom). On the other hand, if the component covers a region with finite area then it is a non-linear component. Consider Figure 5.1 where C_1 to C_4 are simple non-linear components while A, B are end points of a simple linear structure. Among more complex components the curve between C and D represents a tail-like extension from a globular shaped nonlinear component C_1 and the curve EF having no free end is connected on both sides to two non-linear components, C_5 and C_6 respectively. The rest of the dots with no perceptual significance in G to K appear as noise. Among these, only I contain two dots and rests are singletons.

In the previous three chapters, the shape recovery of regular dot patterns is discussed. In this chapter the r-shape is generalized to (*i*) variable density dot pattern, (*ii*) complex pattern having linear components and (*iii*) several distinct components of dots in a single scene.

The proposed approach is based on *k-nearest neighbors* (k-NN) of points. The average distance of k-NNs (with appropriate value of k) of a point can be taken as structuring radius at that point. The resulting r-shape, called *generalized r-shape* provides a border of the dot pattern. Note that the generalized r-shape constitutes disks of variable radii instead of constant radii considered in our previous approach.

An arbitrary dot pattern may contain several components or clusters as shown in Figure 5.1. If the clusters of smoothly varying density can be identified then the borders of generalized r-shapes of these clusters give rise to the shape of the overall dot pattern. Moreover, a dot pattern may contain perceivable linear components as well. Our approach is to detect and trace the linear patterns and then process the remaining dots as possible parts of non-linear component.

The generalized r-shape is defined at the beginning of Section 5.2. Thereafter, clustering is described as a *relativistic* phenomenon (Section 5.2.1). The proposed clustering approach of finding multiple non-linear components is described at the end of this section (Section 5.2.2).

The detection of the linear component is considered in Section 5.3 (although in actual implementation the linear components are treated at first). Section 5.4 contains the digital implementation. Results on some dot patterns are presented and the time complexity of the algorithm is discussed in Section 5.5.

5.2 The Generalized r-Shape and Proposed Clustering

Consider a single non-linear component $S\{P_1, P_2, ..., P_i, ... P_j, ...P_n\}$. Let the average distances of k-nearest neighbors of P_i be denoted by r_i. A point P_i is considered as an interior point in the component if it is 'surrounded' by its k-neighbors. Formally, it can be defined as follows.

Definition 5.1 : A point P_i is said to be a k-r-interior point of S if

$$C_{r_i}(P_i) \subset int \bigcup_{j=1}^{n} D_{r_j}(P_j). \tag{5.1}$$

Otherwise, P_i is said to be a k-r-extreme point of S.

Let

$$E_{r_i}(P_i) = C_{r_i}(P_i) - int \bigcup_{j=1}^{n} D_{r_j}(P_j). \tag{5.2}$$

Definition 5.2 : A point P_i is said to be k-r-adjacent of P_j and $\overline{P_iP_j}$ is said to be a k-r-edge if

$$E_{r_i}(P_i) \cap E_{r_j}(P_j) \neq \phi. \tag{5.3}$$

Any k-r-extreme point having k-r-adjacent(s) is a *non-isolated* point and the set of non-isolated points is considered as *border points* of the dot pattern.

Definition 5.3 : The k-NN based generalized r-shape of S denoted by $k\text{-}rsh_r(S)$, is the planar straight line graph whose vertices are the k-r-extreme points and whose edges are k-r-edges.

If local continuity exist in the distribution of dot pattern then non-linear components of the dot pattern can be identified reasonably well in terms of k-NNs. However, k should be selected properly. Note that Voronoi neighbors are good representation of neighboring points for an interior point and (assuming Poisson distribution of points) on an average the number of Voronoi neighbors is 6 [Okabe et. al. 92] and hence we take $k = 6$.

Let r_i denote the average of distances of k-NNs of P_i.

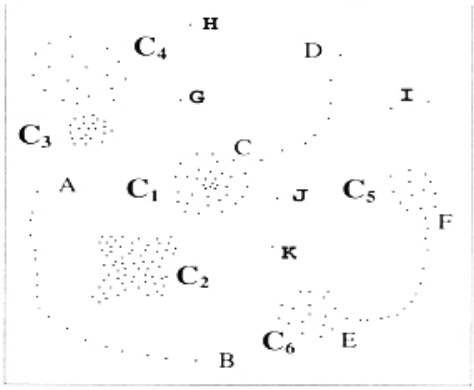


Figure 5.1. *A complex dot pattern with labeled components*

5.2.1 Clustering : a relativistic phenomenon

A close look at how data appear to be clustered in the Euclidean space will reveal that clustering is a relativistic phenomenon. Here, some observations are presented those may be useful for designing clustering algorithms in general [Chaudhuri 93].

I. Whether a set of data (points/samples or features) S will appear clustered depend on the window under observation in which the data set is defined. If S is the only data set in the window and if S appears as clustered then the window size is much bigger than the underlying region of S.

II. If a set of data S appears clustered then the density of data will show a single plateau or a single top and no prominent valley.

III. If there exist another cluster T in the window under observation and if S appears clustered then either one or more of the following should be true:

(a) Number of data in S is comparable with that in T.

(b) Average spacing (density) of data in S is comparable with that in T.

(c) Distance between S and T is large compared to the average spacing of data in S and T.

If the number of data in S is very small compared to that in T, it may not be wise to consider S as a cluster. Similarly, if the average spacing of data in S is very large compared to T the data of S may appear as stray unclustered data. On the other hand, if the distance between S and T is not large compared to the average spacing of data in S and T then they do not appear as distinct clusters. Then either a part or the whole of S gets absorbed in T (i.e. appears as a part of T).

There is interplay of characteristics (a)-(c) stated above. Depending on the situation, one characteristic predominates or aids or obstructs the other in the cluster perception.

IV. Subclusters may be seen in S when 'viewed closely'. But when viewed from a distance, S may appear as a single cluster.

5.2.2 Proposed clustering for non-linear component isolation

This clustering criterion is motivated by the observations discussed above. Because of constant value of k, we are 'viewing the data' from a fixed distance. In other words, only single observation of the data is taken. Thus, IV-th condition is not reflected in our proposed clustering. However, by varying values of k the IV-th condition can be incorporated.

Definition 5.4 : The point-wise consistency measure (PCM) of a neighboring point P_j with respect to P_i is defined as

$$PCM(P_i, P_j) = \frac{|r_i - r_j|}{r_j}. \tag{5.4}$$

A k-neighbor P_j is considered as *consistent* with respect to P_i if $PCM(P_i,P_j) < \varepsilon$ where ε ($<< 1$) is a positive constant.

The approach is based on the concept of *pedigree* using *PCM* as follows. Consider a seed point and find all consistent neighbors among its k-NNs. These consistent neighbors are referred to as the *children of 1st generation*. Now, the children of the 2nd generation for each child of the first generation can be detected. The cluster will continue to grow by appending the children of the previous generation that are not already labeled at the earlier generation. Thus, two members say, P_i and P_j in the pattern set are of same *pedigree* (or cluster) only if one of them (say, P_i) is consistent with the ancestor of the other (say, P_j) at a certain generation.

To choose the seed point, we observe that the point-wise consistency is not a symmetric relation and hence this approach of growing cluster is somewhat biased towards absorbing relatively dispersed points than the ones with higher density.

For a better understanding of this effect, consider two mutually k-neighboring points P_i and P_j. Let $r_i < r_j$ such that

$$PCM\left(P_i, P_j\right) = \frac{\left|r_i - r_j\right|}{r_j} < \varepsilon < \frac{\left|r_j - r_i\right|}{r_i} = PCM\left(P_j, P_i\right). \tag{5.5}$$

In this situation if a cluster grows from P_i, it absorbs P_j irrespective of the distribution of the pattern set. On the other hand if the cluster grows from P_j, unless there already exists some point in the cluster from which a generation-wise consistency relation could be established to P_i, then P_i will never be absorbed in the cluster. Thus, for proper clustering the seed point should be chosen at the densest part of the cluster.

This biased criterion in fact fits well with the human perceptual process. Imagine that you are sitting at the center of a bi-variate Gaussian distributed point cluster. If you look towards the outer dots, they appear to grow out of the center and the cluster contiguity remains intact. But the perception of cluster growth becomes weak if the observer is placed on the periphery of the cluster. The central zone (denser part) appears more distinct and the site of the observer seems to be isolated.

In implementation, to capture the border of the component in a better way, the average of distances of k-NNs is not always taken as the structuring radius (for details, see the algorithm in Section 5.4.1) Also, by introducing additional measures, some corrections (or smoothing) on the border are performed.

5.3 Detection and Tracing of Linear Components

Since the value of average neighboring distance for a point over a linear pattern is much higher than the structuring radius, it is better to distinguish points over linear pattern before the clustering. It is assumed that points on linear structure in dot pattern are more or less evenly spaced. A point P_i that does not lie between two neighboring points of the dotted curve is an *end point*. On the other hand, if there exist two neighboring points at nearly opposite directions about P_i then P_i is an *interim point* of the linear structure.

Definition 5.5 : The local linearity measure (*LLM*) at a point P_i of a dot pattern is defined as the smallest angle at P_i subtended by the two nearest neighbors.

Let θ ($< 45°$) be a small threshold angle and for $i = 1,2, ..., n$ let the point P_i be labeled as '*e*' or '*i*' according to its *LLM* $\leq \theta$ or $> (\pi-\theta)$ respectively. Now, consider 2-NNs of these labeled P_i. If any of these neighbors is not labeled by '*e*' or '*i*', then it is labeled by '*x*'. Let the rest of the points in the dot pattern be labeled by '*z*'. A *e*-labeled point has a very low *LLM* and hence it is very likely to be an end point of a linear structure. Similarly, a *i*-labeled point is very likely to be an interim point in the linear structure. Points labeled by '*x*' may belong to linear structure depending on its surroundings while the point labeled '*z*' is most likely to belong to non-linear structure. In an unsupervised environment where the structure of the underlying pattern is unknown, labeling of points by '*e*', '*i*', '*x*' and '*z*' is not sufficient to trace/detect the linear pattern. But the labeling certainly gives clue about linearity. If in the vicinity of the point P_i having label '*e*' no other point exists with label '*e*' or '*z*' then P_i is accepted as a *strong free end vertex* of a linear structure. In Figure 5.1 A , B, C are the strong free end vertices.

Definition 5.6 : If none of the *k*-NNs of a point P_i with label '*e*' is labeled by '*e*' or '*z*' then P_i is called a *k*-strong free end vertex.

In case of a linear pattern having no free end, the possibility of labeling a point P_i by '*x*' as well as the existence of its 2-NNs with label '*z*' and '*i*' are very high at the vicinity of the junctions between linear and nonlinear structures. If a linear structure with substantial length could be traced out from such a P_i then P_i is considered as a *strong closed end vertex*.

Definition 5.7 : If a point P_i labeled 'x' with its 2-NNs labeled 'z' and 'i' is found such that from P_i a linear path of t ($\geq k$) points having labels other than 'z' could be established and no k-NN of the $\lceil 0.5t \rceil$-th vertex on that path is labeled by 'z' then P_i is said to be a *k-strong closed end vertex*.

It is accepted that a strong end vertex implies presence of a true linear structure. Taking into account the 2-NNs, from a strong end vertex (intuitively from the end vertex) and its successors along the curve, the linear structure are traced out. If the shape of the linear structure is tail-like then usually the label of the other end is 'z' where the tracing is terminated. On the other hand, if it is a simple linear structure then the other end is normally labeled by 'e'. Note that for a simple linear structure, the existence of three successive vertices with label 'x' is impossible. This constraint is also accounted at the time of tracing. If two consecutive predecessors as well as the successor are labeled by 'x' then the tracing is terminated at the successor. If a vertex is found with label 'e' at the time of tracing then that vertex is included in the linear structure. Furthermore, if one of its untagged 2-NNs is 'i', the predecessor with label 'e' is considered as a interim point and the tracing of linear path is continued by changing its label to 'i'.

5.4 Digital Implementation

Let S be a digital dot pattern consisting of n discrete points $P_i = (x_i, y_i)$, $(i = 1,2, ..n)$. Among these points (pixels) suppose that a few are scattered in the image and appear as noise whereas rest of the points form clusters (of uniform or variable densities) or dotted curves.

5.4.1 Algorithm

The algorithm has the following 4 basic components :

I. The computation of k-NN matrix and the corresponding distance matrix;

II. (a) Labeling of points according to its *LLM* by 'e', 'i', 'x' and 'z'.

(b) Detection of all k-strong free end vertices as well as tracing of the linear structures.

(c) Detection of all k-strong closed end vertices as well as tracing of the linear structures,

III. (*a*) Ordering of remaining points from highest density;

(*b*) Non-linear component isolation by clustering.

IV. Computation of generalized *r*-shape based graph of each cluster whose cardinality $\geq k$, as well as their smoothing and correction.

Now, the algorithm is described as follows.

Step 1 : Form an integer matrix (say,) $\boldsymbol{K}(i, j)$ with n rows and $(k+1)$ columns. The first entry ($j = 0$) in each row indicates the candidate point. Set up the corresponding distance matrix (say,) Δ where (i, j)-th entry for $(j > 0)$ is the Euclidean distance between dots P_i and P_j. The $(i,0)$-th entry of Δ represents the average distance of k-NN of P_i viz. r_i. In fact, $\boldsymbol{K}$ and Δ can be grown simultaneously.

Step 2 : Compute the uni-dimensional array Λ of size n where i-th entry stores the *LLM* at P_i. Find the set of k-strong free end vertices using $\boldsymbol{K}$ and Λ. Taking into account the 2-NNs from such a strong free end vertex the linear structure is traced out. For example, let P_i be a k-strong free end vertex with its 1-NN being P_{i_1}, $i_1 \in \{1,2, .., i\text{-}1, i\text{+}1, \ldots n\}$. Both are tagged at Λ. Now, among 2-NNs of P_{i_1}, find the untagged neighbor say, P_{i2} which is different from P_i; tag that vertex also. Continue the recursive search and tagging. The tracing is terminated when a vertex is confronted with only untagged neighbor of label '*z*'. If after tracing two successive vertices with label '*x*' an untagged vertex with label '*x*' is found then too the tracing is terminated. After tracing all linear structures from k-strong free end vertices, k-strong closed end vertices as well as their corresponding linear paths are traced out. The successive vertices along the dotted curve(s) are joined by digital line segments in the output image $\boldsymbol{I}$. Next, the set S' containing the remaining points of S that do not belong to linear structures is processed.

Step 3 : Checking the average k-neighboring distances in Δ, points of S' are ordered in decreasing density and maintained in a uni-dimensional array (say) Γ. Without loss of generality, let $\Gamma = \{P_1, P_2, \ldots P_i, \ldots P_m\}$ where $m \leq n$. Initialize the cluster counter t by 0. P_1 is taken as the seed of the first cluster.

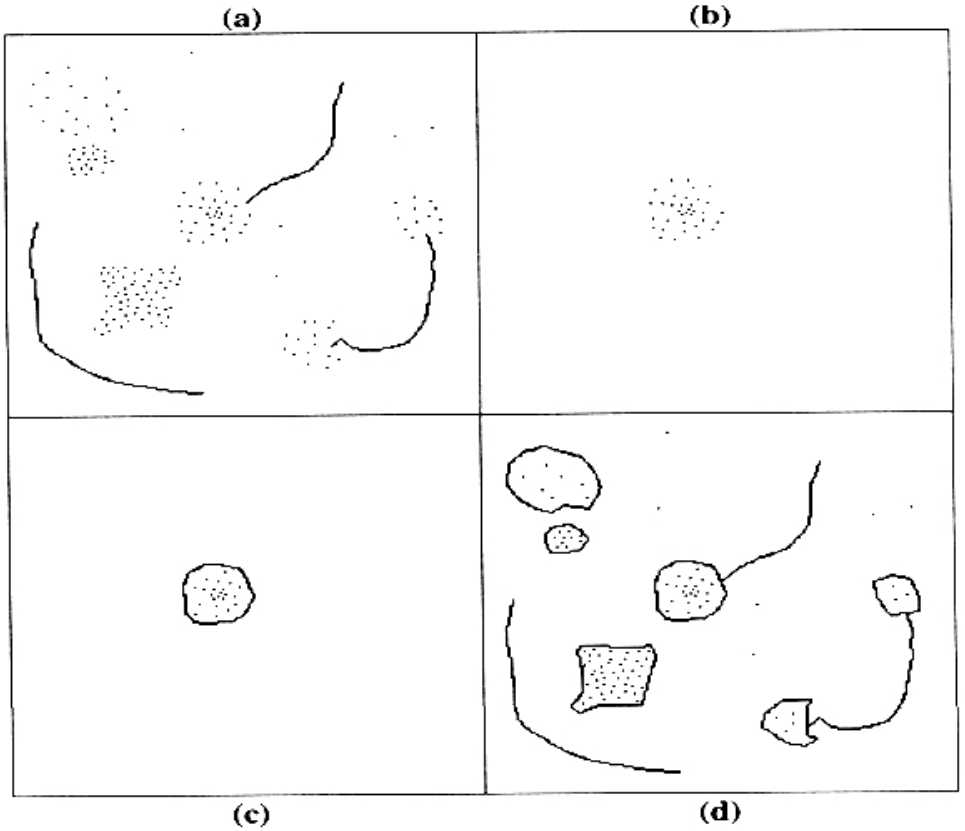


Figure 5.2 *Output at some major steps of the algorithm.*

Step 4 : Suppose the seed is denoted as P'. The cluster around P' is grown. By checking PCM the children of 1st generation among the k-NNs of P' are detected. Let the set be denoted by $\{P'_1, ..., P'_{k'}\} \subseteq \Gamma$ $(k' \leq k)$. Next, the children of 2nd generation viz. the consistent neighbors of P'_1, P'_2, ..., $P'_{k'}$ are detected. Redundant search of vertices is avoided by tagging already traced vertices. When all generation wise consistent neighbors are traced, the growth of cluster from P' is completed. The current cluster is treated as noise if its cardinality is less than k. Otherwise, the cluster is accepted as valid and t is updated by t +1. Without loss of generality, let t-th valid cluster be denoted by $C_t = \{P_1, P_2, \ldots, P_i, \ldots, P_T\}$ where $T \leq m \leq n$. The border of C_t is extracted as follows.

Step 5 : The structuring radii at points of C_t are computed. If the linearity label of the candidate point as well as of its 2-NN are other than 'e' then the average distance of its consistent neighbors among the k-NNs is taken as the structuring radius of that point.

Otherwise, the distance from its nearest neighbor is taken as its structuring radius. Let $\tilde{r}_i$ denote the structuring radius at P_i and Ψ be initialized as an empty image frame. Digital disks $DD_{\tilde{r}_i}(P_i)$ having radii $\tilde{r}_i$ and centers at P_i are successively superposed on Ψ for $i = 1,2, ..., T$. The label of a pixel P in Ψ is assigned by the maximum label i of the disk $DD_{\tilde{r}_i}(P_i)$ which contains P. For each $i = 1,2, ...T$, all the discrete points on $DC_{\tilde{r}_i}(P_i)$ are scanned over Ψ. If there exists a pixel P of label i, having background pixels in its 4-neighborhood then P_i is considered as an *extreme pixel* of the dot pattern. Further, if such a pixel P has a 8-neighbor Q with label j ($i \neq j > 0$) such that Q has also background pixel in its 4-neighborhood then P_j is an *extreme adjacent pixel* of P_i. In that case, digital line segment $\overline{P_iP_j}$ is a possible edge on the border of the cluster. The nodes and edges of the corresponding (undirected) graph of the resulting diagram are stored in a linked list like r-shape listing (Figure 3.3) as described in Section 3.4.1, for further processing.

Step 6 : Some correction or smoothing at the border of the cluster C_l is done through the linked list. If the number of vertices of a closed loop in the list is very small ($\leq k$) then that loop is removed. If multiple paths are detected between two vertices and if one of the paths appears with single edge (see Figure 5.4(b)) then that path is also removed (see Figure 5.4(c)).

It is to be noted that the *degree* (the total number of incident edges) of almost all vertices in the generalized r-shape of non-linear structures are 2. However, if there is a 'neck'-like joint at a vertex then the degree of that vertex is ≥ 4 (see Figure 5.4(e)). If more than 1 self-loop paths are found at a branch point of degree ≥ 4 then the loop making maximum acute angle at the branch point is modified. The branch point is skipped from the loop by joining the second and last but one vertices (see Figure 5.4(f)). The details of path tracing of graphs are skipped. An efficient approach for this purpose is already provided in Section 3.4 [RayChaudhuri et.al. 97]. The consistent edges in the form of digital line segments are drawn in $\boldsymbol{I}$.

Step 7 : The existence of next cluster is checked. If P_j is at the top of Γ that is not tagged by any cluster label then P_j is taken as the seed and the process of border extraction is repeated from Step 4. If no such P_j is found then the algorithm is terminated.

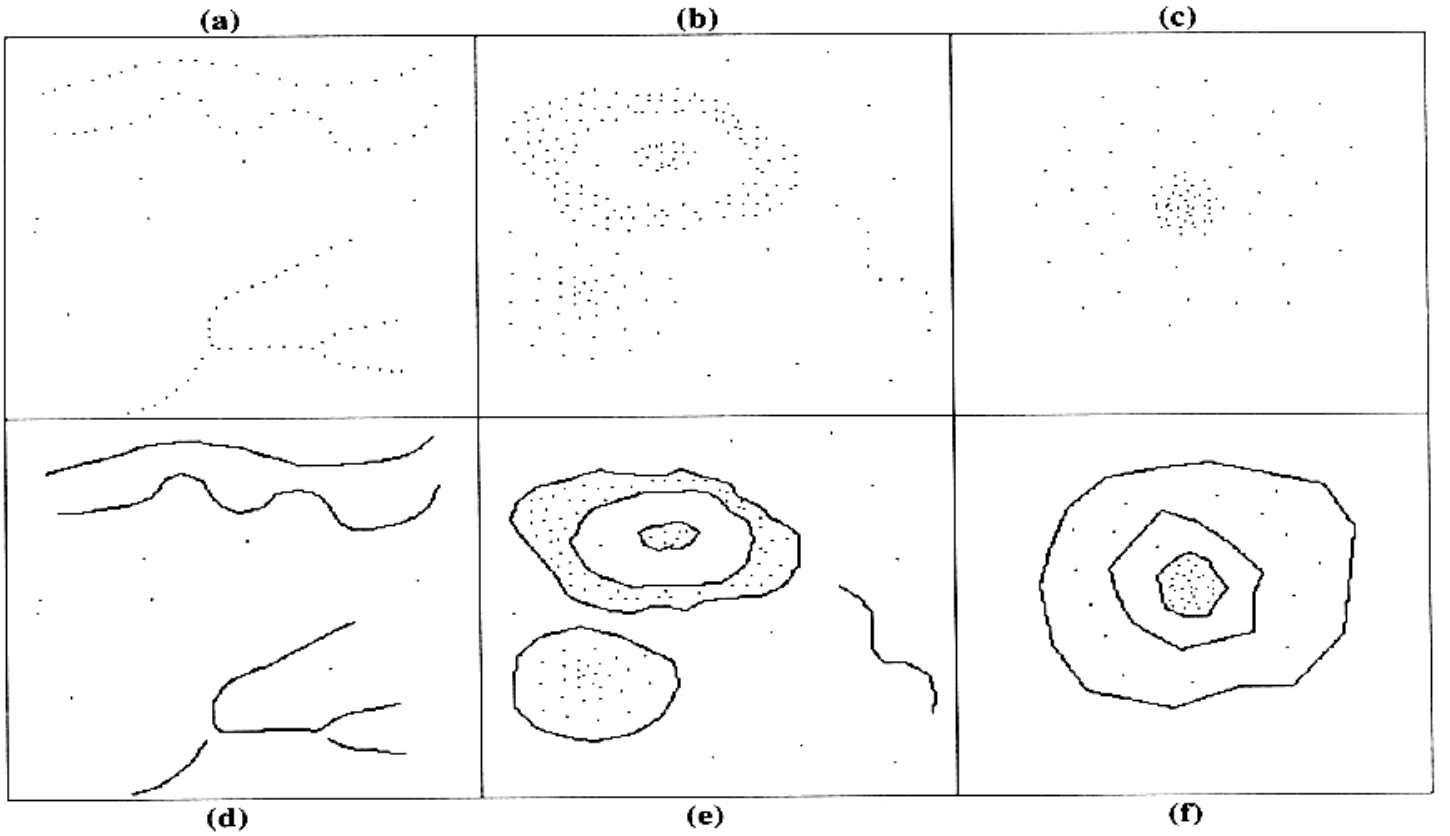


Figure 5.3 *A few dot patterns with various shapes and their respective extracted borders.*

5.5 Results and Discussion

The proposed algorithms are successfully tested on several images of dot patterns. In Figure 5.2(a), all valid extracted linear structures of Figure 5.1 are shown (Results of Step 2). The initially detected non-linear cluster is shown in Figure5.2(b) (Result of Step 4). Its generalized *r*-shape based border is shown in Figure 5.2(c) (Result of Step 5). The complete set of extracted shape from the scene is presented in Figure 5.2(d) (Result of Step 7). Note that because of small size, the 5 clusters *G* to *K* are not considered for the border generation.

A few other typical dot patterns are shown in Figures 5.3(a)-(c). The first one contains linear structures only. One non-linear component surrounded by another non-linear Gaussian-like pattern and a simple linear structure exist in Figure 5.3(b). Although, both

these figures are corrupted by isolated points that may be considered as noise. In Figure 5.3(c) one non-linear denser component is embedded in a dispersed one. In all these images the borders of non-linear components are correctly recovered without any inconsistent edge and the linear patterns are traced as shown in Figures 5.3(d)-(f). However, in Figure 5.3(d) one of the linear structures becomes disconnected at branch points. In Figure 5.4(a) and Figure 5.4(d) two patterns are shown. The first one is rather evenly distributed. In the other figure, five closely spaced Gaussian-type clusters are shown. The two types of inconsistent edges that are discussed in Step 6 of the algorithm are present in the generalized *r*-shapes shown in Figure 5.4(b) and Figure 5.4(e). The corrected borders are shown, respectively, in Figure 5.4(c) and Figure 5.4(f).

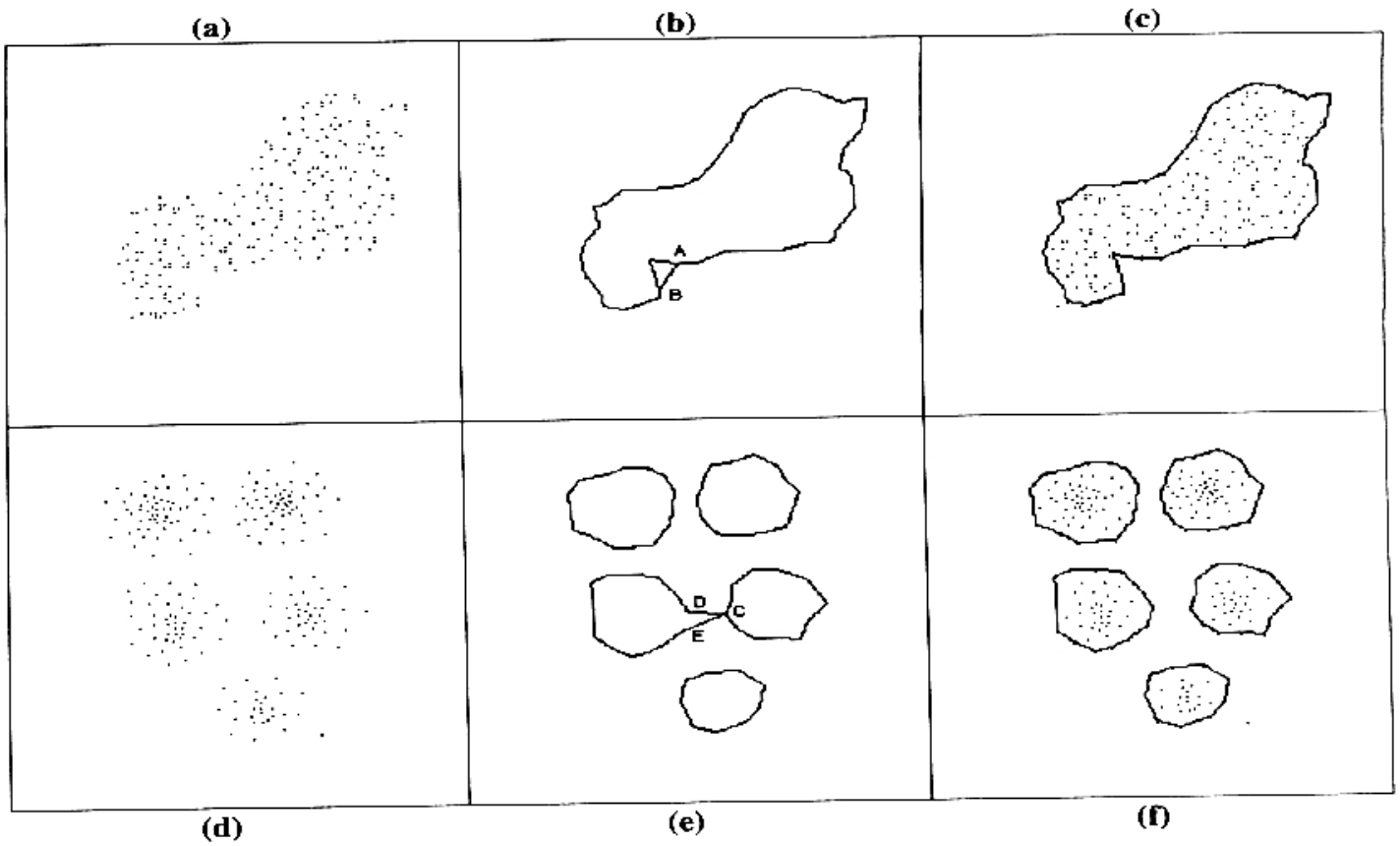


Figure 5.4 *Two cases where correction of borders is necessary*

Our proposed algorithm is robust. In all the figures the isolated noise-like points are correctly detected. However, two points which are not perceived as isolated, got separated from their respective components as shown in Figure 5.3(f) and Figure 5.4(c).

To analyze the time complexity, the proposed algorithm is grouped into 4 parts: (*i*) the computation of the *k*-NN matrix $\boldsymbol{K}$, distance matrix Δ and ordering of points from highest density in Γ; (*ii*) detection and tracing of linear structures; (*iii*) non-linear component isolation by clustering; and (*iv*) computation of generalized *r*-shapes and their smoothing. For straight forward methods, the average case time complexity to obtain $\boldsymbol{K}$, Δ and Γ is $\boldsymbol{O}(n^2)$. However, there exist faster algorithms [Border 90], [Zakara & Ozard 96] for accomplishing this task in $\boldsymbol{O}(n \log n)$. Therefore, it is assumed that the first part can be done in $\boldsymbol{O}(n \log n)$. It can be easily verified that (*ii*) can be computed in linear time. Note that being non-iterative, our proposed clustering algorithm performs a breath first search only once and for each node, there are atmost k times computation on point-wise consistency measure. Thus, the average time complexity of (*iii*) viz., clusters generation is also linear.

Note that there exists algorithm to compute the *k*-SIG [††] in expected $\boldsymbol{O}(n \log n)$ time [Miller et. al. 91]. Since the generalized *r*-shape is a subgraph of *k*-SIG and partially exposed disks to the background can also be detected in $\boldsymbol{O}(n \log n)$ time [Imai et. al. 85], the expected time complexity of generalized *r*-shape computation is bounded by $\boldsymbol{O}(n \log n)$.

Our proposed clustering is not based on a similarity measure and symmetric type relationship among data points as reflected in other k-NN based clustering [Jarvis & Patrick 71], [Gowda & Krishna 78]. Rather, the criterion of clustering is based on dominating points and their consistent zone of influence. It seems more natural that denser points and their neighbors, up to which some contiguity can be established, dominate over the other points, in forming the clusters. This is the justification behind our usage of non-symmetric relation.

[††] The *k*-SIG is the generalization of the sphere of influnece graph [Toussiant 88] by considering the influence zone of each point as the smallest disk containing *k*-NN rather than the 1-NN [Guibas et. al. 91].

Chapter 6.

Consistent Set Estimation in k-Dimension: An Efficient Approach

6.1 Introduction

Shape description may be viewed as an associated problem of the more basic question of set estimation from a finite number of sample points drawn from the set. External boundary of the estimated set may then be used as shape descriptor. One major problem of interest in set estimation is *consistency*. A set estimator is consistent if it converges (in an appropriate sense defined in the following subsection) to the original set α as the number of points drawn from α tends to infinity. Moreover, a good set estimator (a) should be computationally efficient; and (b) should be automatic, in the sense that the method should be able to detect the number of independent disjoint components in the true region even when this number is unknown. None of the estimators [Grenander 75], [Murthy 88], [Mandal et. al. 92], [Worring & Smeulders 94] known to us combine all these properties. Our aim is to obtain a set estimator that possesses all these desirable qualities.

In this chapter, the notion of s-shape in 2-D is extended and a class of set estimators in higher dimension are derived. The theoretical properties related to consistency of these estimators are studied. The spirit of the procedure is non-parametric in nature. At first, we formally define the consistency. Some important results existing in literature on this topic are also mentioned which is followed by the layout of this chapter.

6.1.1 Consistent estimation and existing results

Let $X_1, X_2, \ldots, X_n$ be k-dimensional (k-D) independent and identically distributed (*i.i.d.*) observations from a distribution $\wp$ which is supported on a set $\alpha \subset \Re^k$. Let $cls(\alpha)$, $int(\alpha)$, and $\partial(\alpha)$, respectively denote the closure, interior and boundary of α; whereas λ, Δ and E denote respectively the k-dimensional Lebesgue measure, symmetric difference operator and the expectation of a random variable.

Definition 6.1 : Let $\alpha^*_n \subset \Re^k$ be a set estimator of α based on $X_1, X_2, \ldots, X_i \ldots, X_n$. Then α^*_n is said to be a consistent estimator of α (denoted as $\alpha^*_n \rightarrow \alpha$) if

$$\lim_{n\to\infty} E\left[\lambda\left(\alpha_n^* \Delta\, \alpha\right)\right] = 0 \qquad (6.1)$$

Theorem 6.1 : Let $\alpha \subset \Re^2$ be a bounded Borel set whose boundary has Lebesgue measure 0. Let $<\varepsilon_n>$ be a sequence of positive numbers such that as $n \to \infty$, $\varepsilon_n \to 0$ implies $n\,\varepsilon^2{}_n \to \infty$. Let $\alpha_n^* = \cup\{X \mid \|X_i - X\| \le \varepsilon_n\}$. Then $\alpha^*{}_n$ is a consistent estimator of α under the assumption that $\wp$ is uniform.

The above theorem is due to Grenander [Grenander 75]. The same result is considered in [Mandal et. al. 92] where the circular disk surrounding each point X_i is replaced by rectangular neighborhood.

It is possible to define infinitely many different sequences of ε_n's with the property that as $n \to \infty$, $\varepsilon_n \to 0$ and $n\varepsilon^2{}_n \to \infty$. Since the choice of ε_n does not depend on $X_1, X_2, \ldots, X_n$, Grenander's class of estimators does not have the scale equivariance property.

Another consistent set estimator based on the Minimum Spanning Tree (MST) is due to Murthy [Murthy 88]. In this case, the radii ε_n's are made functions of $X_1, X_2, \ldots, X_n$ in the context of *compact regions* [Apostol 71]. A set α is said to be compact region if α is path-connected, compact, $cls(int(\alpha)) = \alpha$, and $\partial(\alpha)$ consists of finitely many rectifiable curves. Let φ_n denote the MST of $(X_1, X_2, \ldots, X_n)$ and the length of φ_n is l_n.

Theorem 6.2 : Let $\alpha_n^* = \cup\{X \mid \|Y - X\| \le h_n,\ Y \in \varphi_n\}$ where $h_n = \sqrt{\frac{l_n}{n}}$. Then $\alpha^*{}_n$ is a consistent estimator of α.

The above result is also true for any continuous distribution. However, the result can not be extended to the case of union of multiple compact regions unless the number of disjoint components is known. The above two theorems, established only on $\Re^2$, basically take the union of certain circular neighborhoods centering every sample point (in Theorem 6.1) or points over the MST of sample points (in Theorem 6.2) as an estimate of the original set α.

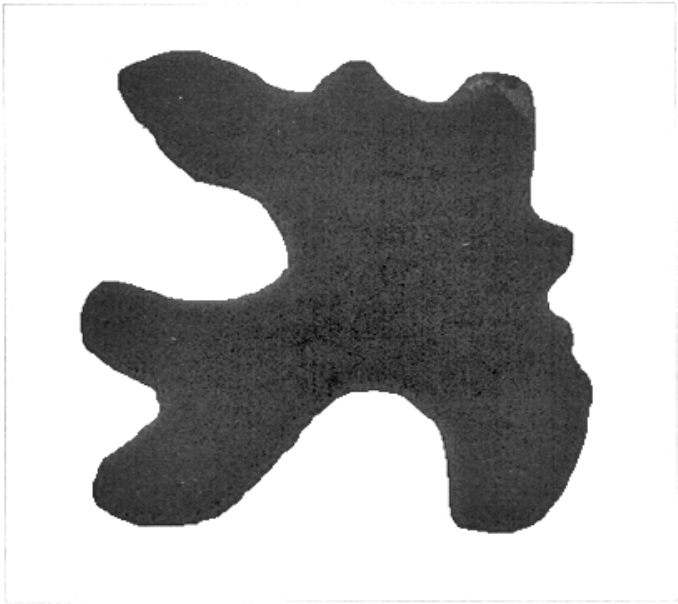


Figure 6.1 *A 'fish shaped' region*

In Section 6.2, the *s*-shape in *k*-D and its derivative, a smoother version, are formally defined. In Section 6.2.1, we prove the consistency of *s*-shape when the point set is generated by a uniform distribution on α. This result is extended to general continuous distributions under appropriate conditions via another theorem. In Section 6.2.2, the error rate in estimation is analyzed. One important derived result is that the order of error is independent of the dimensionality. The consistency of smooth *s*-shape is also established. The notion of *s*-shape spectrum is extended in Section 6.2.3. Coupled with binary opening these class of estimators are highly robust in the presence of noise. Two more results on consistency of the *clopen* (closing followed by binary opening by the same structuring element) version of *s*-shape as well as the *s*-shape spectrum as set estimators, are established.

In Section 6.3, algorithms of linear order complexity that are readily derivable from the definition of the proposed set estimators are applied to dot patterns and some results in digital domain are illustrated. The role of δ, the parameter controlling the structure of the estimators as well as its values which appear to be intuitively and experimentally justifiable are analyzed. Finally, a general discussion is presented in Section 6.4.

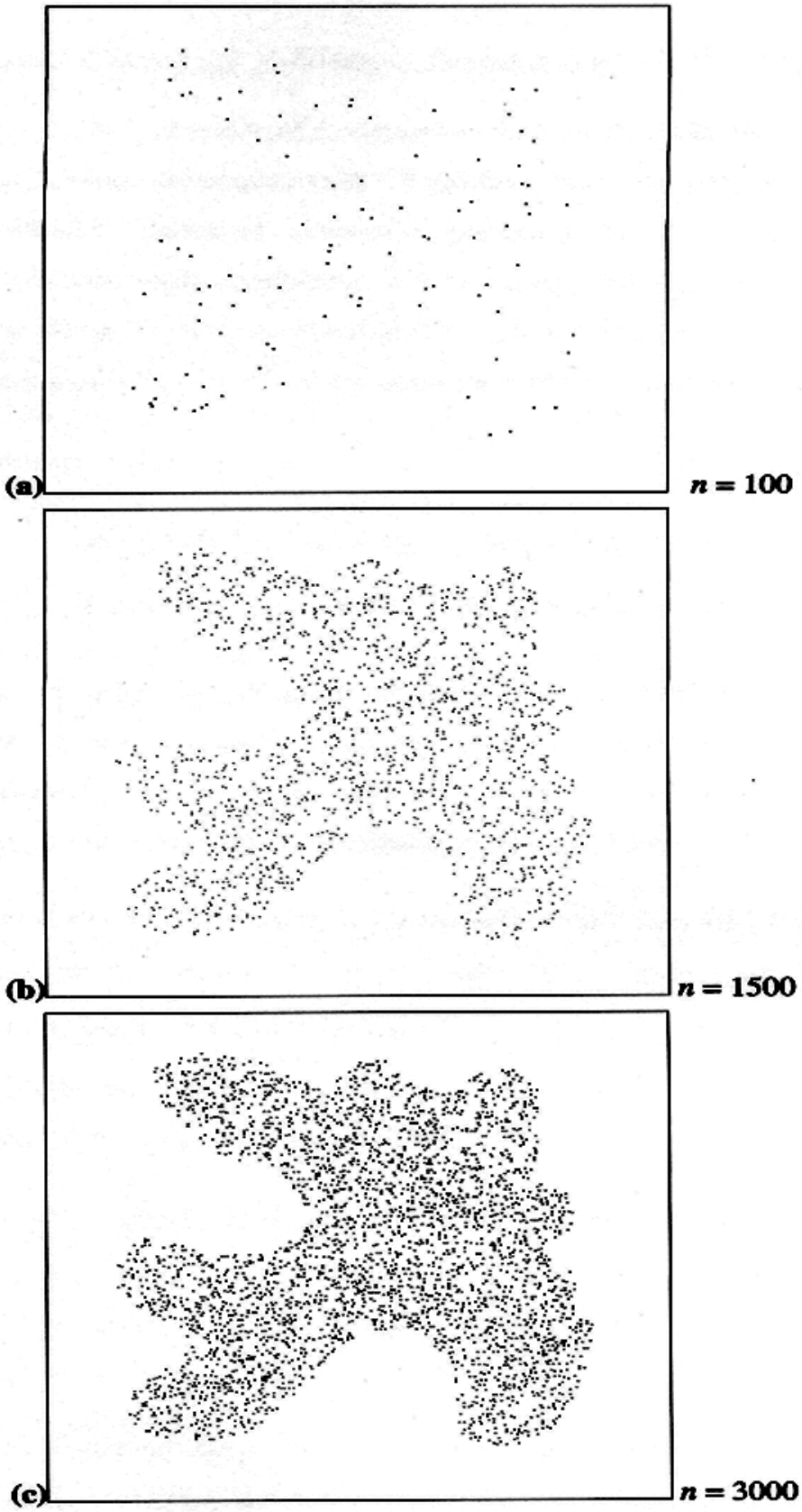


Figure 6.2 *Random samples from* α *of Figure* 6.1.

6.2 Proposed Class of s-Shape Based Set Estimators

Let $S_n \{X_1, X_2,.., X_i,.., X_n\}$ be a set of n sample points in $\Re^k$. Let W_n be the *optimal* (with smallest k-volume) hyper-rectangle, with boundary surfaces perpendicular to the $(k-1)$ dimensional coordinate planes of reference, covering S_n, i.e. $S_n \subset W_n \subset \Re^k$. For the remainder of the chapter, such a rectangle will be simply referred to as the optimal hyper-rectangle covering the point pattern. For a given hyper-cube (grid) of side-length s_n, let $\boldsymbol{F}(s_n)$ denote a lattice of grids on $\Re^k$, with sides parallel to the coordinate axes.

For any such lattice, define

$$\boldsymbol{G}(s_n) = \{g \mid g \cap W_n \neq \phi\}; \qquad G(s_n) = \cup\{g \mid g \in \boldsymbol{G}(s_n)\} \tag{6.2}$$

$$\boldsymbol{H}(s_n) = \{g \mid g \cap S_n \neq \phi\}; \qquad H(s_n) = \cup\{g \mid g \in \boldsymbol{H}(s_n)\} \tag{6.3}$$

Note that $G(s_n)$ denotes the set-union of grids over W_n while $H(s_n)$, the s-shape in k-dimensional space, is the subset of $G(s_n)$ by joining the grids each of which contains at least one point from S_n. Let $\#H(s_n)$ denote the number of grids in $H(s_n)$. Then the Lebesgue measure of $H(s_n)$ in $\Re^k$ is $\lambda(H(s_n)) = \#H(s_n) \times (s_n)^k$.

As discussed in Section 2.2, starting from the grid nearest to the center of reference of the coordinate axes, let the grids of $\boldsymbol{G}(s_n)$ be ordered (like raster scan) in a k-dimensional array. Then $\boldsymbol{G}(s_n)$ induces a k-dimensional array (say,) $\prec z_{t_1,t_2,\ldots,t_k} \succ$. In 2-D, as discussed earlier, it is the dispersion matrix whose (t_1, t_2)-th element denotes the number of dots in the grid situated at t_1-th row, t_2-th column position.

Definition 6.2 : The <u>binary projection</u> of the above array is the array $\prec b_{t_1,t_2,\ldots,t_k} \succ$, defined as

$$\left.\begin{aligned} b_{t_1,t_2,\ldots,t_k} &= 1 \quad \text{if } z_{t_1,t_2,\ldots,t_k} > 0 \\ &= 0 \quad \text{otherwise.} \end{aligned}\right\} \tag{6.4}$$

A set in k-D composed of the positions of non-zero entries in the binary projection, is referred to as an *object* whereas the rest is considered as *background*. Note that there is a one-to-one correspondence between the object and $\boldsymbol{H}(s_n)$. The object in the projection is also denoted by $\boldsymbol{H}(s_n)$.

By generalizing the definitions of 8-neighbors for black and 4-neighbors for white pixels of two-tone image, [Rosenfeld 70] the connectivity in grids of $\boldsymbol{F}(s_n)$ can be defined. Any two grids in the object are *neighbors* if they meet at a point, line, a plane or a hyper-plane of dimension less than k. But in case of background, two grids are neighbors only if they meet at a hyper-plane of dimension k-1.

Two grids g_1, g_l of same type (both empty or non-empty) in $\boldsymbol{F}(s_n)$ are *connected by a path* of grids (in object or background exclusively) in the form of a sequence, say, g_1, g_2, ..., g_i,..., g_l so that g_{i-1} is a neighbor of g_i. A (*maximally connected*) *component* in $\boldsymbol{H}(s_n)$ (or in $\boldsymbol{F}(s_n)\cap\boldsymbol{H}(s_n)^c$) is the subset of $\boldsymbol{H}(s_n)$ ($\boldsymbol{F}(s_n)\cap \boldsymbol{H}(s_n)^c$) so that each grid in it is connected by a path to any other grid in the set. A *hole* is a component of finite measure in the background, $\boldsymbol{F}(s_n)\cap\boldsymbol{H}(s_n)^c$ [Voss 91].

Consider a 3×3×...×3 (k-tuple) array, (say,) τ having all entries equal to 1 as a *k-D structuring element* where the center of reference is located at the middle position of the array. As discussed earlier, the binary closing is a well known morphological filter which is defined as dilation followed by a erosion with a structuring element [Haralick et. al. 87]. It is an increasing, extensive and idempotent operator. Let the object be morphologically operated by a binary closing with τ in k-D integer space $\Im^k$. Let the resulting morphologically closed version of the object in the projection be denoted by $\overline{\boldsymbol{H}}(s_n)$. Then,

$$\overline{\boldsymbol{H}}(s_n)=\left\{Z\in\Im^k \,\middle|\, t\in\Im^k,\ Z\in\tau_t \Rightarrow \boldsymbol{H}(s_n)\cap\tau_t\neq\Phi\right\} \tag{6.5}$$

where τ_t denotes the translation of τ to t.

Let $\overline{H}(s_n)$ denote the union of all grids whose corresponding reference positions in the projection, namely, $Z = (z_1, z_2,..., z_k)$ belong to $\overline{\boldsymbol{H}}(s_n)$. We refer the $\overline{H}(s_n)$ as the *smoothed induced hull of $H(s_n)$ in k-D*. The closing 'smoothes' the set from outside (as discussed in Section 2.2) so that all holes of 'negligible size' (less than size of τ) become part of the object. The superset $\overline{\boldsymbol{H}}(s_n)$ is considered as a smooth version of $\boldsymbol{H}(s_n)$ in the projection. For example in 2-D, an empty grid, g having 5 non-empty grids in its 8-neighbors becomes a part of the closed version $\overline{H}(s_n)$.

The consistency of $H(s_n)$ is first analyzed under a uniform distribution. Regarding the choice of s_n, a data driven procedure is proposed and the range over which $H(s_n)$ remains consistent is established. Then it is generalized to the case of any continuous distribution.

6.2.1 Consistency of the s-shape

Consider any set α, a finite union of connected subregions in $\Re^k$, each of which is bounded by a closed hyper-surface of finite area. Assume that the interior of α has a positive k-D volume (Lebesgue measure) but the boundary is of Lebesgue measure zero. Also let W be an optimal hyper-rectangle such that α lies in the interior of W. Without loss of generality, let the volume of α be p, $(0 \le p \le 1)$ and that of W be 1. Let S_n $\{X_1, X_2, \ldots X_i, \ldots, X_n\}$ be the set of n points chosen at random under the uniform distribution over α. Let the k-D volume of the optimal hyper-rectangle W_n covering S_n be $A_n \le 1$. Let the side-length of hyper-cube grids in lattice $\mathcal{F}(s_n)$ on $\Re^k$ be

$$s_n = n^{-\delta}\left(\sqrt[k]{A_n}\right), \qquad 0 < \delta < 1. \tag{6.6}$$

For clarity of presentation, the following notations are used. Let $\#G_n$ denote the number of grids in the union, G_n. Let T_n, I_n, B_n and H_n denote, respectively, the union of grids in $\mathcal{F}(s_n)$ intersecting α (some of them may not contain points of S_n), completely in the interior of α, intersecting the boundary of α and the grids containing points of S_n, i.e. the k-D s-shape. Clearly, $T_n = I_n \cup B_n$. Let $\#T_n$, $\#I_n$, $\#B_n$ and $\#H_n$ denote the number of grids in the respective unions. Moreover, let n_I and n_B represent the number of points of S_n in I_n and in B_n, respectively. If s_n is chosen as above, we shall show that the Lebesgue measure of the symmetric difference of H_n and α goes to 0 in probability as n tends to ∞ for the appropriate choices of δ.

By the Strong Law of Large Numbers (SLLN) any subregion of α with positive Lebesgue measure eventually has a point chosen from it in S_n with probability 1. Thus $W_n \to W$ in the sense of (6.1) as $n \to \infty$ and with probability 1, $A_n \to 1$. We also have,

$$\lim_{n\to\infty} \lambda(B_n) = 0. \tag{6.7}$$

In fact,

$$\lim_{n\to\infty} \frac{\lambda(I_n)}{\lambda(T_n)} = 1 \text{ and } \lim_{n\to\infty} \frac{\lambda(B_n)}{\lambda(T_n)} = 0. \tag{6.8}$$

Similarly,

$$\lim_{n\to\infty} \frac{n_I}{n} = 1 \text{ with probability 1.} \tag{6.9}$$

Note that $\#G_n$ is approximately $n^{k\delta}$, and $\lim_{n\to\infty} \frac{n^{k\delta}}{\# G_n} = 1$.

Let $n_I \cong n\, a_n$ where $\lim_{n\to\infty} a_n = 1$. By the above results, it is easily established that

$$\lim_{n\to\infty} \frac{\# I_n}{\# G_n} = p\,. \tag{6.10}$$

That is,

$$\lim_{n\to\infty} \frac{\# I_n}{n^{k\delta}} = p\,. \tag{6.11}$$

Suppose that n balls are thrown at random in $P_n = \frac{n^\theta}{a}$ boxes, $0 < \theta < k$, where a is a finite positive constant. Then the probability that any box remains empty is

$$\left(\frac{P_n - 1}{P_n}\right)^n = \left(1 - \frac{1}{P_n}\right)^n = \left(1 - \frac{a}{n^\theta}\right)^n, \tag{6.12}$$

which goes to zero in the limit for $\theta < 1$. Now, the expected proportion of empty cells among the $\#I_n$ cells in the interior of the region α is

$$\left(1 - \frac{1}{\# I_n}\right)^{n a_n} = \left(1 - \frac{n^{k\delta}}{\# I_n} \frac{1}{n^{k\delta}}\right)^{n a_n} \tag{6.13}$$

Since $\lim_{n\to\infty} \frac{n^{k\delta}}{\# I_n} = \frac{1}{p}$, the limit of the expression in the right hand side of the above equation becomes $e^{-\frac{1}{p}}$ if $\delta = \frac{1}{k}$; equals to 0 if $\delta < \frac{1}{k}$; and equals to 1 if $\frac{1}{k} < \delta < 1$.

Thus, for any choice of $\delta < \frac{1}{k}$, the expected proportion of empty cells among the squares completely in the interior of the region α goes to 0 for a sufficiently large n.

Since the proportion of empty grids is a non negative random variable, the proportion of empty grids among those completely in the interior of α also goes to 0 in probability by Markov's inequality.

Also, by equation (6.8), as $\lim_{n\to\infty} \frac{\# B_n}{\# T_n} = 0$, the proportion of empty cells among $\#T_n$ is 0 in the limit. Hence, $\lim_{n\to\infty} \frac{\# H_n}{\# T_n} = 1$ and H_n eventually covers α in probability. That is,

$$\lambda\left(H_n^c \cap \alpha\right) \to 0 \text{ in probability as } n \text{ goes to infinity.} \quad (6.14)$$

Conversely,

$$\begin{aligned} \lambda\left(H_n \cap \alpha^c\right) &\le \lambda\left(\left(B_n \cup I_n\right) \cap \alpha^c\right) \\ &= \lambda\left(B_n \cap \alpha^c\right) + \lambda\left(I_n \cap \alpha^c\right) \le \lambda\left(B_n\right). \end{aligned} \quad (6.15)$$

Taking limit on both sides,

$$\lim_{n\to\infty} \lambda\left(H_n \cap \alpha^c\right) \le \lim_{n\to\infty} \lambda\left(B_n\right) = 0. \quad (6.16)$$

Combining (6.14) and (6.16) it is established that $\lambda\left(H_n \Delta \alpha\right) \to 0$ in probability. ❐

Since the above symmetric difference is a positive bounded random variable, the following theorem is established :

Theorem 6.3 : Let S_n $\{X_1, X_2, .., X_i, .., X_n\}$ be *i.i.d.* observations from a uniform distribution supported on a region α, a finite union of connected subregions in $\Re^k$, where each subregion is bounded by a closed hyper-surface of finite area. Let W_n be an optimal hyper-rectangle covering S_n with hyper-volume A_n. If $s_n = n^{-\delta}\left(\sqrt[k]{A_n}\right)$, $0 < \delta < \frac{1}{k}$, then the s-shape $H(s_n)$ is a consistent estimator of α in k-dimension.

6.2.1.1 Points from continuous distribution

Let f (>0) be the continuous density function of the random variable over the k-dimensional region α which is defined as in the beginning of Section 6.2.1. Without loss of generality let the k-volume (Lebesgue measure) of α be p', $(0 < p' < 1)$ and that of W be 1.

Let $\wp(Q)$ be the probability of any subregion Q of α under the density f. Let n points be chosen from the region α at random under $\wp$ and the set of points be denoted by S_n. Consider any $\varepsilon > 0$. Then one can choose a large number m such that

$$\lambda\left(\alpha_m\right) > p' - \frac{\varepsilon}{2} \quad \text{where} \quad \alpha_m = \left\{x \mid x \in \alpha, \frac{1}{m} < f\left(x\right) < m\right\}. \quad (6.17)$$

Let $\wp(\alpha_m) = p$. We assume that the boundary of the set α_m has Lebesgue measure 0. Let $T_{m,n}$, $I_{m,n}$, $H_{m,n}$, denote, respectively, the union of grids in $F(s_n)$ intersecting α_m, completely in the interior of α_m, the grids in α_m containing points of S_n. Also, let $\#T_{m,n}$, $\#I_{m,n}$, and $\#H_{m,n}$ denote the number of grids in the respective unions. The other notations and terms are identical with those of the previous theorem.

As justified in the previous theorem, in the sense of expression (6.1), as $n \to \infty$,

$$W_n \to W \tag{6.18}$$

while with probability 1,

$$A_n \to 1 \tag{6.19}$$

and

$$\lambda\left(\alpha_m \cap T_{m,n}^c\right) = 0\,. \tag{6.20}$$

Let a (sample) point Z be drawn from α. Then

$$\wp\left(Z \in I_{m,n}\right) = \int_{I_{m,n}} f(x)dx < m\lambda\left(I_{m,n}\right). \tag{6.21}$$

For a given grid g among $\#I_{m,n}$,

$$\wp\left(Z \in g \middle| Z \in I_{m,n}\right) = \frac{\wp(Z \in g)}{\wp\left(Z \in I_{m,n}\right)} > \frac{\frac{1}{m}\lambda(g)}{m\ \#I_{m,n}\,\lambda(g)} = \frac{1}{m^2 \#I_{m,n}}. \tag{6.22}$$

If m_I points are drawn from $I_{m,n}$ then an upper bound of the probability that g is empty can be found from the following equation.

$$\wp\left(g \cap S_n = \Phi \middle| g \subset I_{m,n};\ \#\left(I_{m,n} \cap S_n\right) = m_I\right) < \left(1 - \frac{1}{m^2 \#I_{m,n}}\right)^{m_I} \tag{6.23}$$

where $\#\left(I_{m,n} \cap S_n\right)$ denotes the cardinality of $I_{m,n} \cap S_n$.

By the SLLN,

$$\lim_{n\to\infty} \frac{m_I}{n} = p\,. \tag{6.24}$$

Let $m_I \cong na_n$, where $\lim_{n\to\infty} a_n = p$.

For $i = 1, 2, \ldots, \#I_{m,n}$ let the characteristic function χ_i be defined as

$$\chi_i = \left.\begin{array}{l} 1 \ \ if\ \ i\,th\ \ grid\ \ is\ \ empty \\ 0 \ \ otherwise. \end{array}\right\} \tag{6.25}$$

As proportion of empty grids among $\# I_{m,n}$ is $\frac{\sum_{i=1}^{\# I_{m,n}} \chi_i}{\# I_{m,n}}$, the *expected* proportion of empty grids among $\# I_{m,n}$ is

$$E\left(\frac{\sum_{i=1}^{\# I_{m,n}} \chi_i}{\# I_{m,n}}\right) = \frac{\sum_{i=1}^{\# I_{m,n}} E(\chi_i)}{\# I_{m,n}} < \left(1 - \frac{1}{m^2 \# I_{m,n}}\right)^{m_l} = \left(1 - \frac{1}{m^2} \frac{n^{k\delta}}{\# I_{m,n}} \frac{1}{n^{k\delta}}\right)^{n a_n} \qquad (6.26)$$

Following same arguments as in (6.13) and (6.24) the limit of the expression in the right hand side of the above equation becomes $e^{\frac{-p}{m^2 \lambda(\alpha_m)}}$ if $\delta = \frac{1}{k}$; equals to 0 if $\delta < \frac{1}{k}$; and equals to 1 if $\frac{1}{k} < \delta < 1$. Thus, for any choice of $\delta < \frac{1}{k}$, the proportion of empty grids completely in the interior of the region α_m goes to 0 for a sufficiently large n.

Following same arguments as for the expression (6.14),

$$\lambda\left(H_{m,n}^c \cap \alpha_m\right) \to 0, \qquad \text{in probability.} \qquad (6.27)$$

For any given $\varepsilon > 0$ and $0 < t < 1$, by (6.16), (6.17) and (6.27) we can choose M and N such that whenever $m \geq M$, $n \geq N$ and $\delta < \frac{1}{k}$, the probability

$$P\left(\left|\lambda\left(H_n^c \cap \alpha\right)\right| < \varepsilon\right) \geq P\left(\left|\lambda\left(H_{m,n}^c \cap \alpha_m\right)\right| < \varepsilon\right) \geq 1 - t. \qquad (6.28)$$

As (6.28) is true for arbitrary ε and t, and since

$$\lambda\left(H_n \cap \alpha^c\right) \leq \lambda\left(B_n\right) \to 0, \qquad (6.29)$$

we have, $\lambda\left(H_n \Delta \alpha\right) \to 0$ in probability. ❐

Thus, from the above result the following theorem is established.

Theorem 6.4 : Let $X_1, X_2, \ldots, X_n$ be *i.i.d.* observations from any continuous distribution $\wp$ having support α on which its density function f is positive. Then the *s*-shape, $H(s_n)$, is a consistent estimator of α in *k*-dimension when the other conditions are the same as in Theorem 6.3.

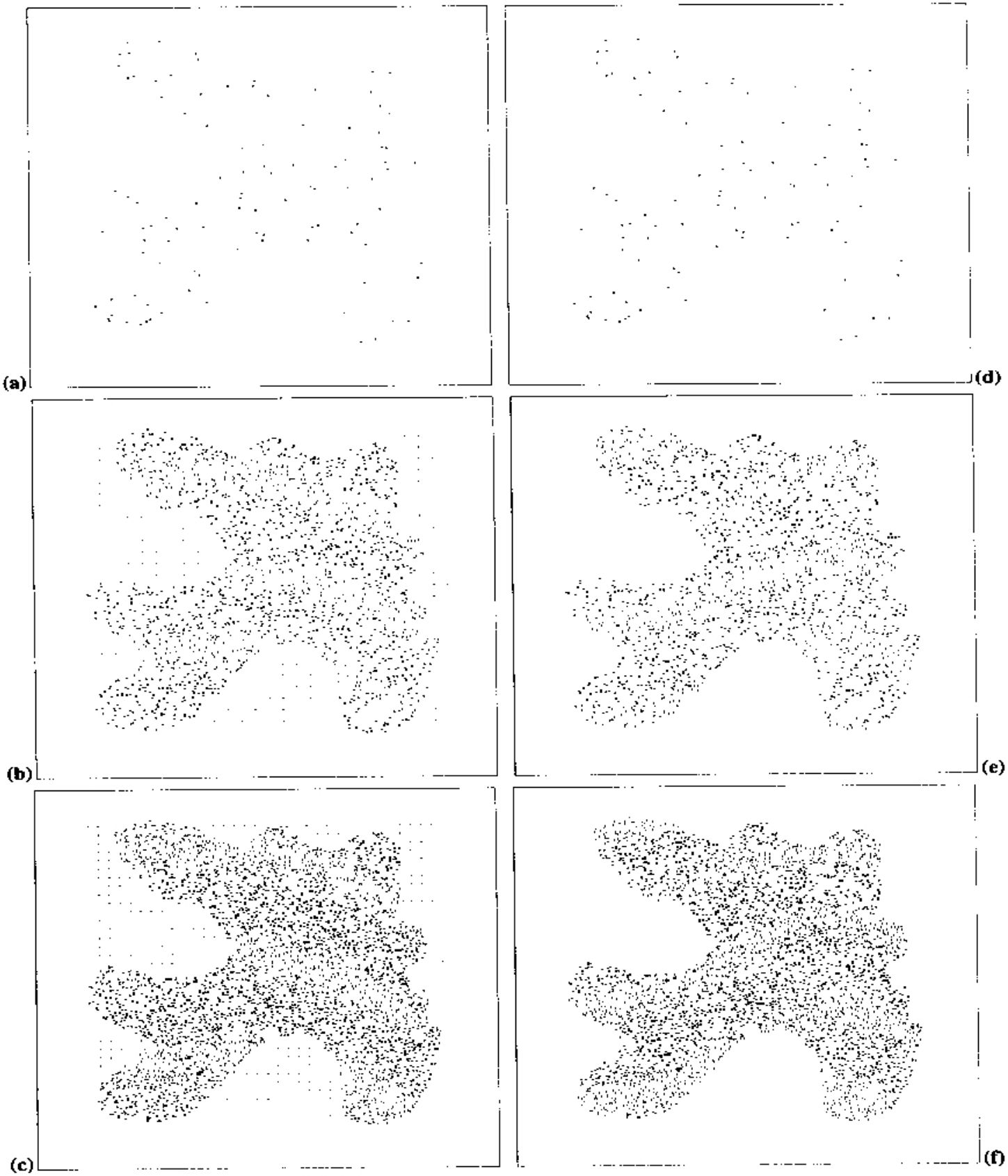


Figure 6.3 *The asymptotic convergence for* $\delta = 0.45$.

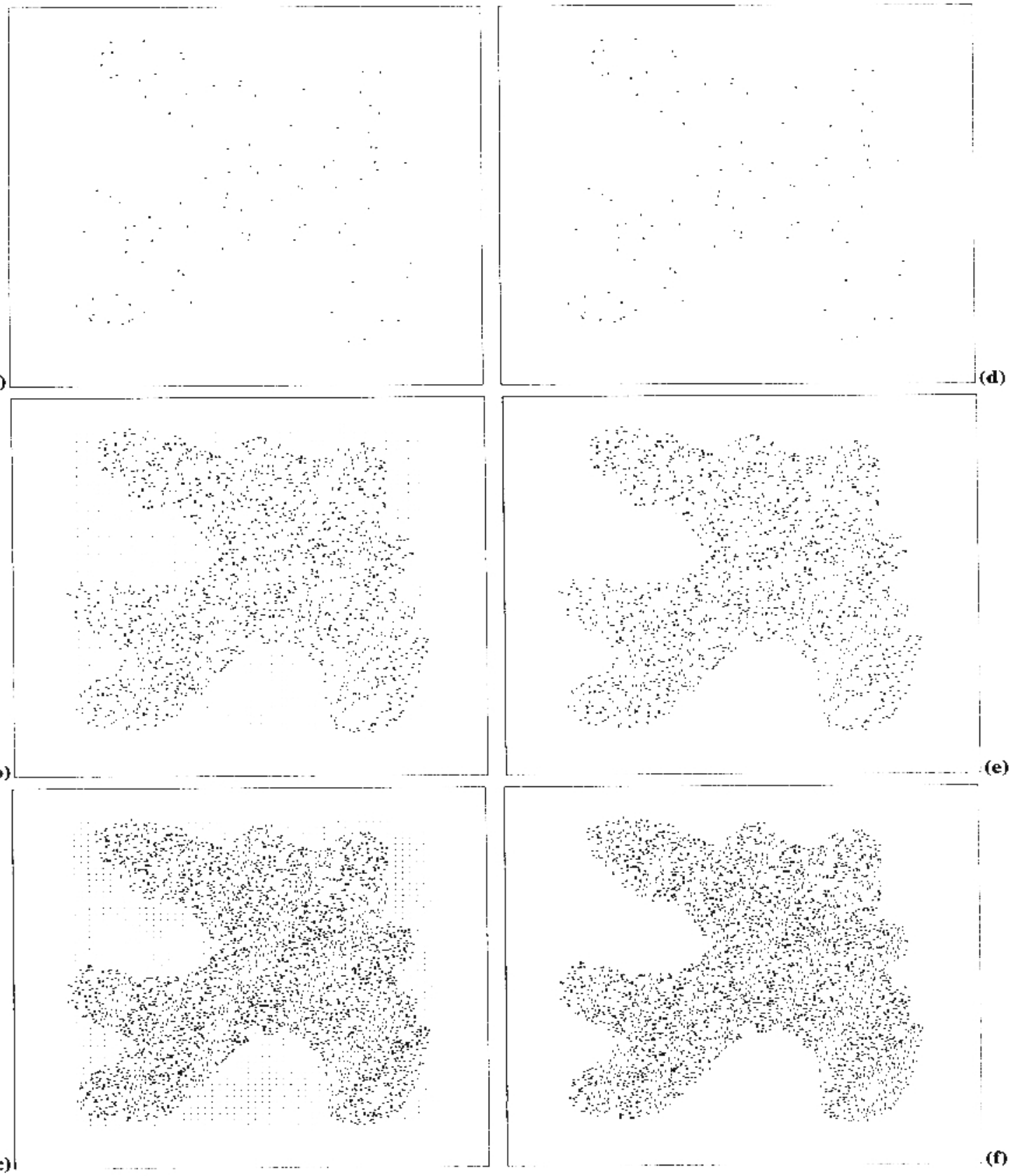
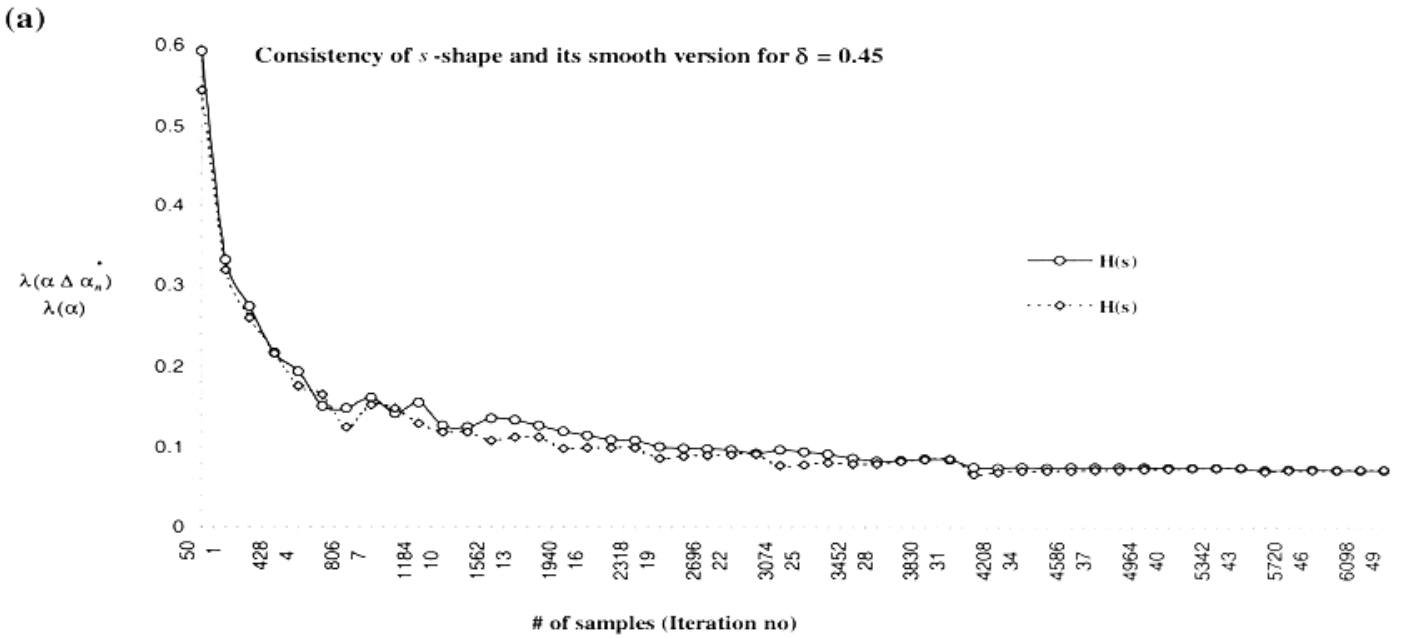


Figure 6.4 *The asymptotic convergence for* $\delta = 0.49$.



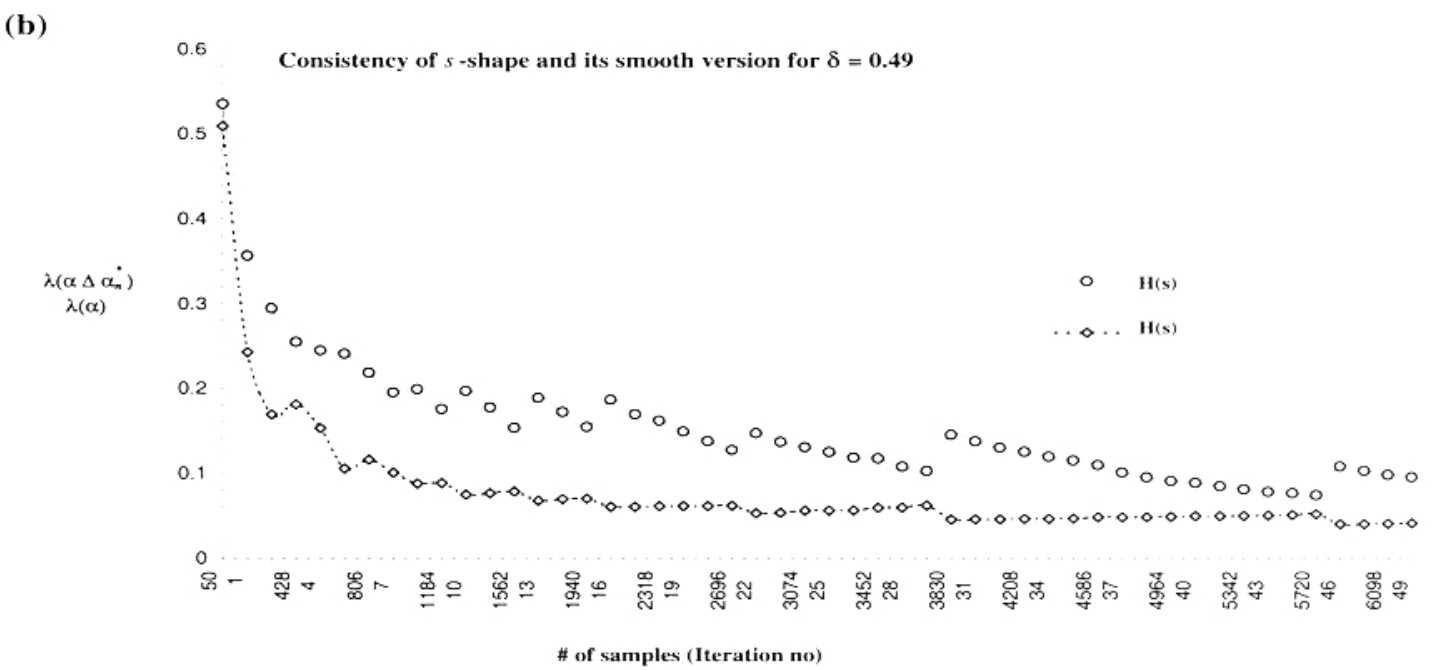


Figure 6.5 *Plots showing asymptotic convergence of proposed estimator in 2-D*

Note that henceforth, for convenience we assume that the distribution of sample points is uniform. The minor difficulties that arise in case of a continuous distribution can be treated by more elaborated definitions and considerations, which would only tend to detract from our presentation.

6.2.2 Error

It is crucial that the experimenter has an idea of the order of error (in terms of the Lebesgue measure of the symmetric difference between α and α^*_n) when the procedure is terminated at a particular value of n and the corresponding estimate α^*_n has been determined. We provide an upper bound to this error when the points are drawn under a uniform distribution. We consider the hyper-cubes in the interior and the boundary of α separately.

The error in the interior E_I, related to the proportion of empty grids, is equal to

$$E_I = \lambda(I_n) \times \left(1 - c_1 \frac{1}{n^{k\delta}}\right)^{c_2 n} \tag{6.30}$$

where c_1 and c_2 are constants greater than 0.

The logarithm of the R.H.S. of (6.30) is taken. Expanding $log\left(1 - c_1 \frac{1}{n^{k\delta}}\right)^{c_2 n}$ and then exponentiating back, the leading term of E_I is found to be $c_3 e^{-n^{(1-k\delta)}}$ (where $c_3 > 0$).

Let $\zeta(\partial(\alpha))$ denotes the k-D surface area of α (in k-1 dimension). Then the error in the boundary E_B satisfies

$$\begin{aligned} E_B &\leq \#B_n \times n^{-k\delta} \\ &\leq \left(\frac{\zeta(\partial(\alpha))}{n^{(k-1)\delta}}\right) \times n^{-k\delta} \\ &\leq c_4 n^{-\delta}, \quad \text{where } c_4 \text{ is a positive constant.} \end{aligned} \tag{6.31}$$

Note that the error in the boundary dominates that in the interior. Thus, the error in estimation is at most of order $O(n^{-\delta})$. One important point to note is that the error is independent of the dimensionality.

6.2.2.1 Consistency of the smooth s-shape, $\overline{H}(s_n)$

The smoothed induced hull $\overline{H}(s_n)$, in general, is a better representation of the shape of a dot pattern than s-shape [RayChaudhuri et.al. 97]. The consistency of $\overline{H}_n$ (which is an abbreviation of $\overline{H}(s_n)$) is analyzed.

It can be easily verified that

$$\lambda(H_n) \le \lambda(\overline{H}_n) \le 3^k \lambda(H_n). \tag{6.32}$$

As $\overline{H}_n$ is a superset of $H(s_n)$, we get by (6.14),

$$\lambda(\overline{H}_n^c \cap \alpha) \to 0 \qquad \text{in probability.} \tag{6.33}$$

The boundary error may increase in case of $\overline{H}(s_n)$. But as

$$\lambda(\overline{H}_n \cap \alpha^c) < 3^k \lambda(H_n \cap \alpha^c),$$

by (6.16),

$$\lambda(\overline{H}_n \cap \alpha^c) \to 0 \quad \text{in probability.} \tag{6.34}$$

The above two equations result in the following theorem. ❐

Theorem 6.5 : The smooth induced hull $\overline{H}(s_n)$ is a consistent estimator of α under the same condition imposed on Theorem 6.3.

6.2.3 Robust estimators in presence of noise

Noise is a very common phenomenon in any system of observation and inference. The input data may be corrupted at various level of processing like sampling, transmission etc. Perhaps because of the scarcity of implementable consistent estimators very little attention has been paid to this problem. If the reference position of a point is outside the region under consideration then that point should be treated as noise. Due to inherent connectivity in the MST, the consistent estimator based on MST [Murthy 88] is highly noise sensitive. The class of s-shape based estimators proposed above are *partially robust* in the following sense : if the noise originates from within the optimal hyper-rectangle then its influence is little but otherwise it may adversely affect the result.

In the presence of noise, there is a possibility of abrupt increase in the size of the optimal hyper-rectangle, the grid-size becomes erroneously larger than the structuring

size. Moreover, since $H(s_n)$ and $\overline{H}(s_n)$ are unable to remove grids upon scattered noisy points, in constrained situation, these result in wrong estimation with deformed shape and topology (considering the set estimator as shape descriptor).

To make the class of *s*-shape based estimators more robust two steps are taken. First, to find a proper grid size in noisy data, as discussed in Chapter 2, the notion of *s*-shape spectrum is extended in k-D. Thereafter, grids upon scattered noisy points in the *s*-shape are taken care by the clopen transform.

6.2.3.1 *The s-shape spectrum in k-D and its consistency*

Let

$$\left.\begin{aligned} s_n^t = s_n &= n^{-\delta}\sqrt[k]{A_n} && when \quad t=1 \\ &= n^{-\delta}\sqrt[k]{A\left(H\left(s_n^{t-1}\right)\right)} && >1 \end{aligned}\right\} \tag{6.35}$$

where $A\left(H(s_n)\right)$ denotes the *k*-dimensional volume of the hull $H(s_n)$.

Then $\left\langle H\left(s_n^t\right)\right\rangle$ is the *s*-shape spectrum in *k*-dimension.

We will analyse whether $H\left(s_n^t\right)$ are consistent set estimators for finite t's. To prove the consistency, we will only show that the expected proportion of empty grids completely in the interior of α goes to 0 for sufficiently large n. The rest of the proof is similar to that of Theorem 6.3. We do this for the uniform distribution case, but the result goes through for the general distribution with minor modifications.

Let for simpler notation $H\left(s_n^t\right)$ be also denoted by H_n^t. Also, let $\#I_n^t$ denote the number of grids completely in the interior of α. Similar simplified notations are also used for other related terms.

Note that,

$$s_n^{t+1} = n^{-\delta}\sqrt[k]{A_n}\ \prod_{i=1}^{t}\sqrt[k]{\frac{A\left(H_n^i\right)}{A\left(H_n^{i-1}\right)}}, \quad t \geq 1 \tag{6.36}$$

where $A\left(H_n^0\right) = A_n$.

We want to show that H_n^{t+1} is a consistent estimator of α when t is a finite positive integer.

With the initial grid length s_n^1 the optimal isothetic window W_n covering S_n is partitioned into $\#G_n^1$ ($\approx n^{k\delta}$) grids and the consistency of H_n^1 i.e. $H(s_n)$ has already been established. Now consider, the case for $t = 1$.

$$s_n^2 = n^{-\delta}\sqrt[k]{A\left(H_n^1\right)}$$
$$= n^{-\delta}\sqrt[k]{A_n}\sqrt[k]{\frac{A\left(H_n^1\right)}{A_n}}$$

By the consistency of the s-shape generated with grid length s_n^1,

$$\sqrt[k]{\frac{A\left(H_n^1\right)}{A_n}} \to \sqrt[k]{p}\,. \tag{6.37}$$

and
$$\#G_n^2 \approx \frac{A_n}{n^{-k\delta}A\left(H_n^1\right)} \approx \frac{n^{k\delta}}{p} \qquad \text{for large } n. \tag{6.38}$$

Also $\lim_{n\to\infty} \frac{\#I_n^2}{\#G_n^2} = p$ implies
$$\lim_{n\to\infty} \frac{\#I_n^2}{n^{k\delta}} = 1\,. \tag{6.39}$$

Thus, the expected proportion of empty grids among the $\#I_n^2$ in the interior of the region of α is

$$\left(1-\frac{1}{\#I_n^2}\right)^{na_n} = \left(1-\frac{n^{k\delta}}{\#I_n^2}\frac{1}{n^{k\delta}}\right)^{na_n} \quad \text{where} \quad \lim_{n\to\infty} a_n = 1 \tag{6.40}$$

For arbitrary large n, the above relation tends to e^{-1} if $\delta = \frac{1}{k}$; equals to 0 if $\delta < \frac{1}{k}$; and equals to 1 if $\frac{1}{k} < \delta < 1$.

Thus, for $t = 1$, H_n^{t+1} is a consistent set estimator under the conditions of theorem 6.3.

Note that for $t = 2$,

$$\#G_n^3 = n^{k\delta}\left[\frac{A_n}{A\left(H_n^1\right)}\right]\times\left[\frac{A\left(H_n^1\right)}{A\left(H_n^2\right)}\right]. \tag{6.41}$$

Now, $\frac{A_n}{A\left(H_n^1\right)}$ and $\frac{A\left(H_n^1\right)}{A\left(H_n^2\right)}$ tend to $\frac{1}{p}$ and 1 respectively for large n.

Thus, for large n

$$\#G_n^3 \approx \#G_n^2 \approx \frac{n^{k\delta}}{p} \tag{6.42}$$

Subsequently, for $t = 2$, H_n^{t+1} is also a consistent set estimator under the same conditions. Thus, by induction, it is established that –

Theorem 6.6 : The s-shape spectrum $\langle H(s_n^t) \rangle$ is a consistent estimator of α under the same conditions imposed on Theorem 6.3.

One interesting observation from (6.42) is that for large n there is no significant change of the s-shapes in the spectrum for t ≥ 1. Thus, in noisy data $H(s_n^2)$, the result after second iteration may be taken as the s-shape of the sample set for large n.

6.2.3.2 Consistency of the clopen version of the s-shape

Let $\overline{H}(s_n)$be morphologically filtered by the binary opening with the structuring element, τ. In contrast with the binary closing, opening is an increasing, anti-extensive and idempotent transform. Let the output be denoted by $\mathring{\overline{H}}(s_n)$ and the union of all grids, whose corresponding (l,m)-th reference positions belong to $\mathring{\overline{H}}(s_n)$, be denoted by $\mathring{\overline{H}}(s_n)$. Also, let $\mathcal{A}$ denote the event that any non-zero position in $\overline{H}(s_n)$, whose corresponding grid lies in the interior of α, remains non-zero after opening. Now, to show the consistency of $\mathring{\overline{H}}(s_n)$, it is sufficient to establish that the probability of $\mathcal{A}$, $P(\mathcal{A}) \rightarrow 1$.

Consider a window, w of size 5×5×...×5 (k-tuple). Let b be a non-zero (black) position of $\overline{H}(s_n)$. It can be verified that to convert the value of b to 0 (white) by opening with the structuring element τ, at least two 0 valued positions should exist in the w centering at b. (The number of required white positions in w increases with the dimensionality). Let Y = number of white grids-positions in w around the non-zero position, b in $\overline{H}(s_n)$.

By Markov's inequality,

$$P(Y \geq 2) \leq \frac{E(Y)}{2} = 5^k \frac{E(\chi_i)}{2} \rightarrow 0 \tag{6.43}$$

In other words, $P(\mathcal{A}) \rightarrow 1$. Thus,

Theorem 6.7 : The clopen version of the s-shape, $\mathring{\overline{H}}(s_n)$ is also a consistent estimator of α under the same condition imposed on Theorem 6.3.

6.3 Implementation in Digital Domain

For visualizing the effectiveness of our proposed consistent set estimators in practice, we have implemented the *s*-shape in two dimensional digital domain. The foreground in a (binary) digital image might be considered as α. Random samples of n object pixels are picked. The *area* of α, $\lambda(\alpha)$ is measured by the total number of object pixels in α. Following Section 2.3, algorithms of linear order time complexity for computing *s*-shapes and their smooth versions as set estimators are readily derived from the definition of binary projection.

The cardinality of the foreground in Figure 6.1 is 63903. We randomly take sample points shown in Figure 6.2 (a)-(c) with $n = 100$ ($\approx 0.001 \times \#\,\alpha$), 1500 ($\approx 0.02 \times \#\,\alpha$) and 3000 ($\approx 0.04 \times \#\,\alpha$), respectively. For $\delta = 0.45$, their $H(s_n)$ and $\overline{H}(s_n)$ are presented in Figure 6.3 (a)-(c) and Figure 6.3 (d)-(f), respectively. The corresponding figures for $\delta = 0.49$ are shown in Figure 6.4. The ratio of $\lambda(\alpha^*_n \Delta \alpha)$ by $\lambda(\alpha)$ are plotted for $\alpha^*_n = H(s_n)$ and $\alpha^*_n = \overline{H}(s_n)$ against the sample size for $\delta = 0.45$ and 0.49 in Figure 6.5 (a) and (b), respectively. The asymptotic convergence of α^*_n is readily understood despite the limitation due to quantization. As far as set estimation is concerned, the smoothing leads to a substantial improvement for the case $\delta = 0.49$, but not for $\delta = 0.45$.

Figure 6.6 presents two more examples where our estimator is applied as shape descriptor as well. The hole (in input Figure 6.6(a)) and disconnected components (Figure 6.6(d)) are correctly recovered. Two sample sizes of 300 and 1500 are taken in both sets. All output figures represent smoothed *s*-shapes due to binary closing with δ fixed at 0.49.

The robustness of the s-shape based class of set estimators is reflected in Figure 6.7. The input patterns shown in Figure 6.2 are degraded by noise [Zhou & Gordon 91]. In all cases signal to noise ratio, SNR, are fixed to 10 db and the used estimator is $\overline{H}(s_n^2)$ (with δ fixed at 0.49).

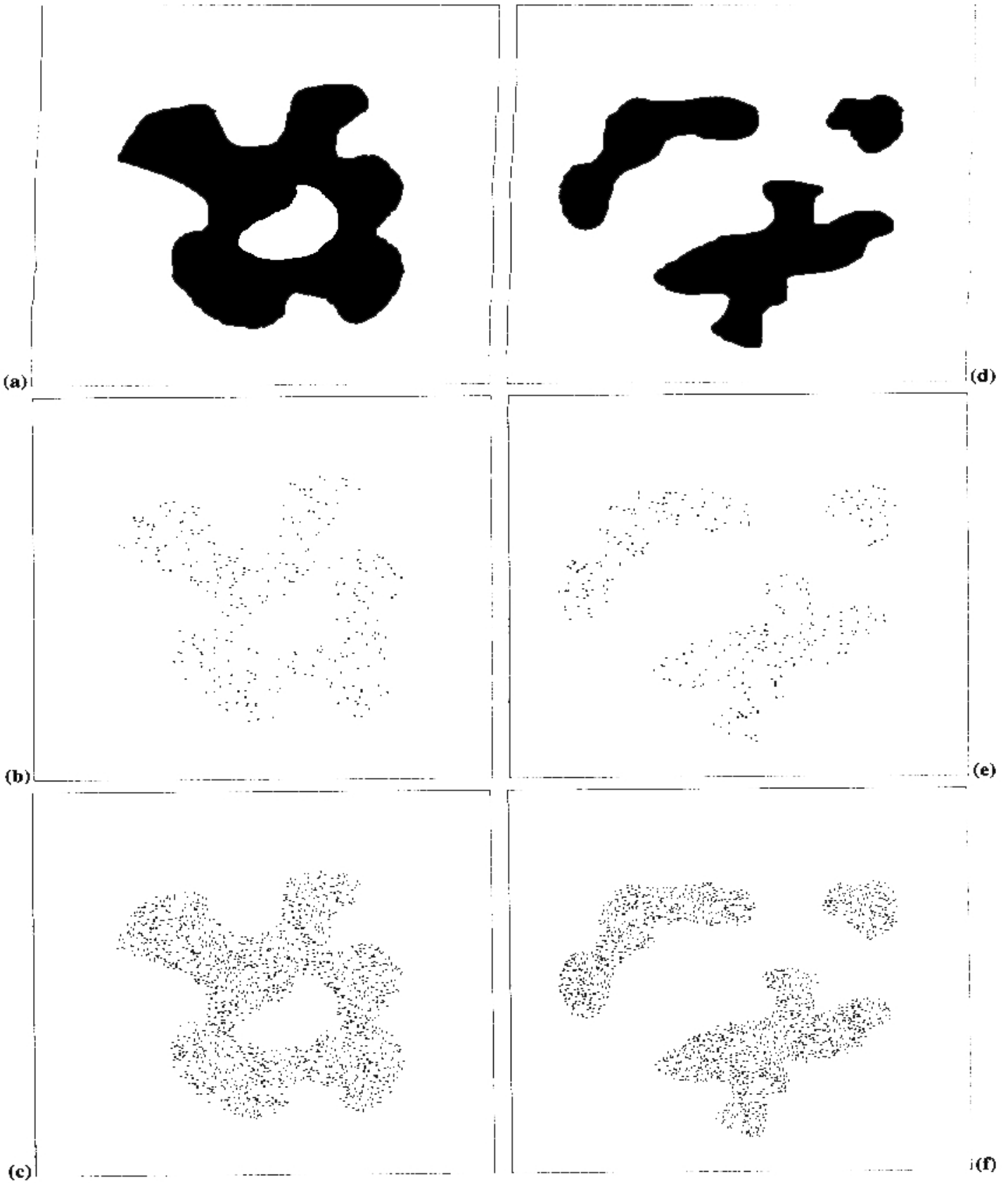


Figure 6.6 *Two more examples where smoothes s-shape based set estimators extracts pattern shapes.*

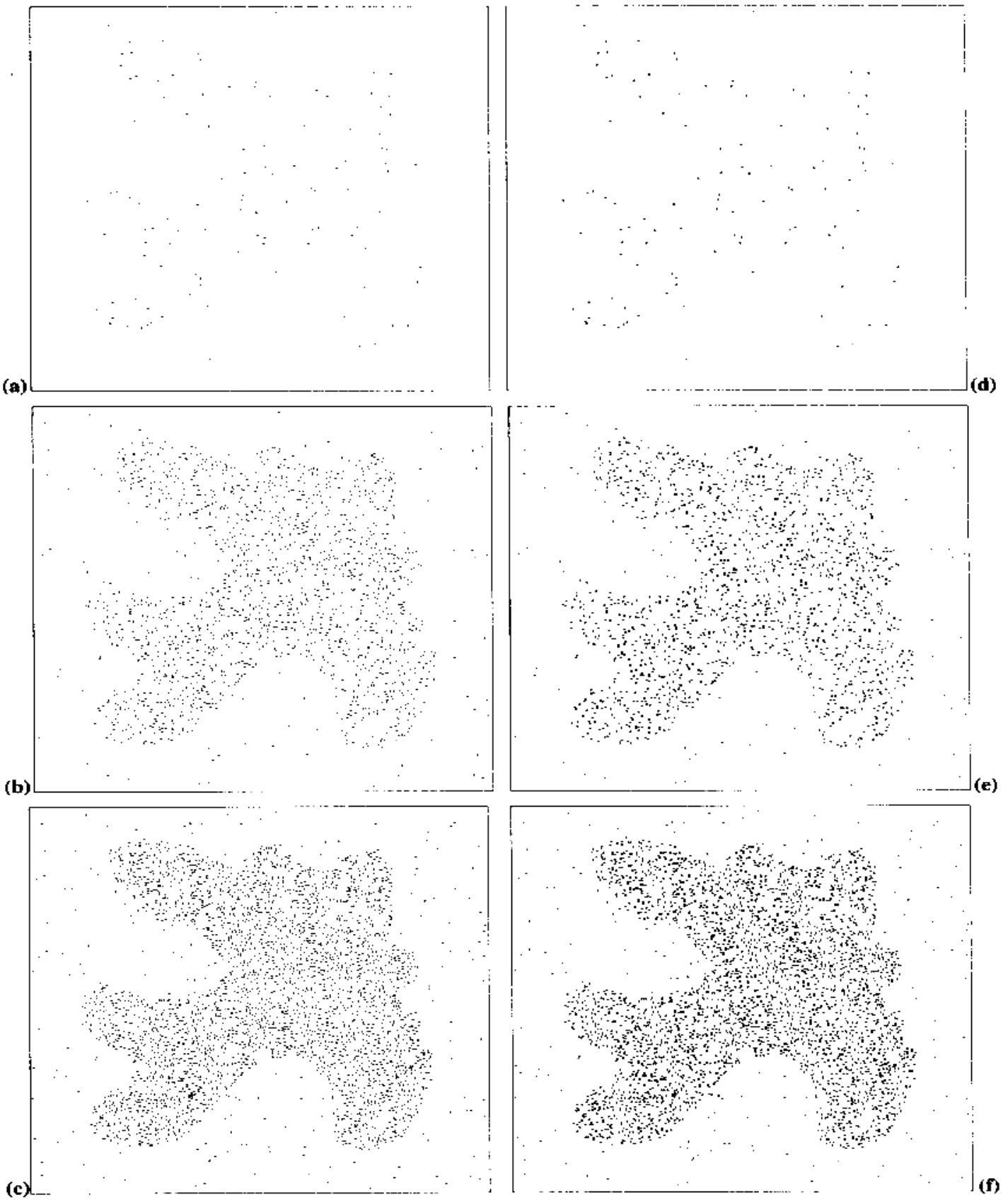


Figure 6.7 *Robustness of the class of proposed set estimators.*(SNR = 10 db)

6.3.1 Choice of δ

It is clear that the choice of δ has considerable impact on the resulting *s*-shape. For smaller values of δ, the boundaries of $\alpha^*_n = H(s_n)$ are cruder – so much so that the *s*-shapes for δ in the range $(0, 0.45)$ appear to be of little practical utility. For larger values of δ, on the other hand, the figure exhibits larger number of inconsistent holes (compare Figure 6.3 and Figure 6.4). In the particular case $\delta = 0.5$, the proportion of the area formed by the union of these holes with respect to the area of the region under estimation converges to a fixed non-zero constant so that consistency fails to hold. This suggests that 'smoothing' may be more useful for *s*-shapes obtained with values of δ close to 0.5. For a given n, larger values of δ lead to small values of $\lambda(\alpha^*_n \cap \alpha^c)$ and smaller values of δ lead to small values of $\lambda((\alpha^*_n)^c \cap \alpha)$ i.e. values of δ near opposite ends of the allowable range are more efficient in reducing complementary components of the symmetric difference.

On the whole, it appears that when single values of δ have to be recommended, then values of δ around the center of the allowable range or closer to 0.5 should be chosen (as larger values of δ reduce the dominant boundary error). When coupled with smoothing, including the case of noisy data, the values of δ close to 0.5 should be chosen.

6.4 Discussion

The effectiveness of *s*-shape as a set estimator has been illustrated in the above examples. From these examples it can be seen that the class of *s*-shape based estimators can also be viewed as descriptors of the shape of the underlying region. These aspects have separately been discussed in Chapter 2 though the treatment is slightly different.

Our proposed consistent set estimator is totally different from the consistent estimators reported earlier. The consistent estimators proposed by others are constructed by dilating either sample points themselves or the edges of MST of the samples by a certain structuring disk. In our case, the optimal zone covering the sample set is partitioned by a lattice and the union of non empty grids is taken as α^*.

From the error rate analysis one interesting observation in connection to the *s*-shape spectrum is that for dense patterns there is no significant change of the *s*-shapes after the second iteration.

The major advantage of our estimators is that it is fully unsupervised. The disconnected components are correctly detected and estimated as n increased. Whereas, the MST based estimator [Murthy 88] needs prior knowledge about the number of disconnected components in α. As we have shown, the *s*-shape could be easily modified to become robust in presence of noise without violating its consistency which is another important advantage.

In practical implementation, one needs to have an idea of the order of error (in terms of the Lebesgue measure of the symmetric difference) when the procedure is terminated. We have provided an upper bound to this error (which is independent of the dimensionality) that may be used by practitioners as a guiding measure in determining the *stopping criterion*. Note that the stopping criterion is unavailable for other existing estimators.

Chapter 7.

Conclusion

7.1 Summary and Contributions of the Thesis

This thesis concerns the recovery of dot pattern shape. Two aspects of the shape are considered here. The first aspect, called external shape, tries to capture the perceived border of the dot pattern. The second aspect, called internal shape, tries to capture the core of the underlying region in the form of skeletons. It is to be noted that all the shape study in this thesis has been done for 2-D patterns. While extension to high dimension is possible, it has not been done in the thesis.(Further, we do not know the run-time complexity of the these shape descriptors in more than 2 dimensions)

The shape detection of dot pattern may also be viewed as a set estimation problem. In that case validation of consistency criterion becomes an important issue. In this thesis we have shown that our shape descriptors can be used for set estimation as well.

The problem of partitional clustering of points arises if several point patterns exist in the space under consideration. So this type of clustering is treated as a sub-problem of shape detection in case of complex point patterns. A set of connected edges gives the boundary of a cluster. This type of clustering is also considered in the thesis.

The contribution of the thesis can be described in the following points.

I. A new shape descriptor, s-shape is defined based on proper selection of the grid-length s of the lattice laid over the pattern plane. The s-shape that is closer to the perceptual shape of the dot pattern can be selected from the s-shape spectrum, which is a finite sequence of s-shapes. Because of its inherent simplicity, the descriptor is easily implemented with linear (in terms of cardinality of dot pattern) time complexity in digital domain.

II. Another external shape descriptor called r-shape is proposed. It produces polygonal border smoother than s-shape. The structuring radius for the r-shape is selected from the same sequence of positive numbers that is generated from the s-shape spectrum. Like s-shape, the time complexity of r-shape based perceptual border extraction is

linear. In their original form, the *s*-shape and *r*-shape can be applied on dot patterns of even distribution only.

III. Note that the spatial density measure of certain cell-organelles in a tissue section is an essential part in pathological test. Moreover, shape and size of important organelles have to be studied individually. In some case the boundary of such an organelle is not distinct from the background but is surrounded by tiny organelles. We have illustrated one in the case of nucleus border segmentation and this process may be used as a general technique in the above problem.

IV. A new structural basis of dot pattern called dispersion matrix is generated from the *s*-shape. It corresponds to a natural partition of the data. The dispersion matrix is a useful tool for shape analysis and shape recovery of dot patterns. It has been shown that the binary image resulting from this dispersion matrix, which we call the binary projection is similar to a scale-reduced version of the underlying region of the dot pattern. Moreover, the number of object pixels in it is less than the cardinality of the dot pattern. This projection is efficiently used for thinning (skeletonization) of the dot pattern and thus extracting its internal shape.

V. The internal shape of dot pattern resulting from the morphology based thinning of the dispersion matrix projection, is a compact representation of the structure of the DP. This can be used as vector representation, from which shape understanding is possible, by a syntactic approach.

VI. Our approach of shape computation of regular dot patterns is extended to complex cases where the data density may vary in a single component, or overlapped pattern or a denser pattern embedded in dispersed dots. We have also developed the approach for detecting patterns where the dots are arranged like arc or line. Such curvilinear patterns are at first detected and separated out. Then a new clustering algorithm is used to tackle the rest of the pattern.

VII. The utility of our approach in the problem of set estimation has been examined. We develop a *s*-shape-based class of set estimators that is unsupervised, consistent and even robust in presence of additive noise. It is proved that the *s*-shape is a consistent estimator not just under the uniform distribution, but also when the points are drawn

according to any continuous distribution. The order of error in estimation is independent of the dimensionality. The proposed class of set estimators outperforms other set estimators like [Murthy 88], [Grenander 75], [Mandal et. al. 92].

7.2 Scope of the Future Work

I. This thesis is oriented towards developing methodologies for dot pattern shape analysis and set estimation rather than their applications to practical problems. (We have presented only one such problem in cytology). The future work should be directed towards using these approaches to different data of practical interest. One of the possible applications is in Astronomy namely, automatic detection of the shape of constellation of stars where a star can be considered as a point in space. The shape of different galaxies and nebulae may also be found by our approach [Thonnat 88]. Another possible application may be visual texture analysis in images where the texture consists of grains of different densities [Toriwaki & Yokoi 88]. The centroid of each grain may be considered as a point and textures of different grain densities can be separated using the approach of Section 5.2.

II. In this thesis most studies are done on points in 2-D space although extension and generalization in 3 and higher dimension have been indicated and described. More extensive work should be done on 3-D space. For example, in 3-D the dots may be arranged as a plane or a curved surface, besides being on a line or arc [Charan & Ahuja95]. So, detection of planar or surface structure of dots in 3-D should be worked out. The perceived boundary of the dot pattern in 3-D also should be a polyhedra whose planes have some dots as vertices [Edelsbrunner & Mücke 94]. It is necessary to extend our approach for the detection of polyhedral faces.

III. In order to detect multiple components in a two dimensional dot pattern, we have devised a clustering. As there is no dimensional constraint in the proposed algorithm, it will be interesting to study its performance in higher dimensional feature space.

IV. Since the binary projection of dispersion matrix is invariant to translation and scaling of the dot pattern and is not sensitive to addition of noise or distortion (locally) to the

pattern, it may be used for dot pattern classification and pattern matching problems. These potentials of binary projection may be investigated in further detail.

V. In this thesis we have worked on the consistency of *s*-shape and its smooth version. It will be interesting to work on the consistency of the *r*-shape as well.

Appendix

Basics of Binary Mathematical Morphology

There are two basic operations in binary Mathematical Morphology : dilation and erosion. Other operators are derived from these two operations. Dilation and erosion can be constructed from even lower level set-operations like translation, set-union and set-intersections. The morphological operators used in the thesis, are defined below.

Let X and T be any two bounded compact subsets in a norm space I and let t be a point in I. (In image processing application, I is the discrete digital plane). Here X is treated as object and T as structuring element by which X is morphologically operated.

The dilation ($\oplus$) is a morphological operator that locates all points (pixels in digital case) where the structuring element can be positioned in the pattern space (binary image) with respect to its center of reference and can intersect with at least one point (pixel) of the object (foreground) of the input. Formally,

$$X \oplus T = \bigcup_{x \in X} T_x . \tag{A.1}$$

Here, X denotes the object (binary image) and T_x denotes translation of the structuring element T by a distance x.

On the other hand, erosion (Θ) locates all points (pixels in digital case) in X where T can be positioned and contained completely within X. Formally,

$$X \Theta T = \{x | T_x \subseteq X\}. \tag{A.2}$$

The other morphological operators are derived from these two basic transforms. Among them, two most important operators namely closing ($\bullet$) and opening ($\circ$) are defined respectively by

$$X \bullet T = (X \oplus T) \Theta T \tag{A.3}$$

and

$$X \circ T = (X \Theta T) \oplus T. \tag{A.4}$$

In image processing, filters are used to clean the noise in the input. A transformation $\Psi(*)$ is said to be a morphological filter if for any two object sets (images) X and X' in the domain of transformation,

$$X \subseteq X' \Rightarrow \Psi(X) \subseteq \Psi(X') \qquad (Increasing) \tag{A.4}$$

$$\Psi(\Psi(X)) = \Psi(X) \qquad (Idempotence) \tag{A.5}$$

The closing and opening are both morphological filters. More specifically, they are respectively, extensive ($X \subseteq X \bullet T$) and anti-extensive ($X \supseteq X \circ T$) . The opening 'smoothes' the set from inside : all or parts of foreground structures that cannot contain the structuring element are removed by the closing. On the other hand, closing 'smoothes' the set from outside: all background structures that can not contain the structuring element are added to the foreground by closing.

Another important morphological operator is the hit-and-miss transform ($\otimes$). It is used to select points with certain geometric properties of the input. In case of digital image, the hit-and-miss transform is used to detect corner pixels, isolated pixels, or border pixels and it performs template matching, thinning etc. Let T and T' be two mutually disjoint structuring elements. The hit-and-miss transform of X by (T, T') is defined by

$$X \otimes (T, T') = (X \ominus T) \cap \left(X^c \ominus T\right) \tag{A.6}$$

where X^c denotes complement of the set X.

The morphological 'interiorical' and 'boundary' of a set are proposed as follows.

Definition A-1 : The *interior* of a set $X \in \boldsymbol{I}$ is the union of all disks within A with some positive radius, denoted as $int(X)$ so that

$$int(X) = \bigcup_{\varepsilon > 0} X \circ T_\varepsilon \, . \tag{A.7}$$

Definition A-2 : The *(inner) boundary* of a set $X \in \boldsymbol{I}$, $b(X)$ is the residue of X that does not belong to $int(X)$. In other words

$$b(X) = X \,/\, int(X) \, . \tag{A.8}$$

Definition A-3 : The *outer boundary* of a set $X \in \boldsymbol{I}$, $b'(X)$ is the residue of complement of X, that is always contained in the dilated part A for any structuring disk of arbitrary small radius. That is,

$$b'(X) = \left(\bigcap_{\varepsilon > 0} X \oplus T_\varepsilon \right) \cap X^c \tag{A.9}$$

The Bibliography

1. [Ahuja 82] **N. Ahuja**, 'Dot Pattern Processing Using Voronoi neighborhoods', *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 4, No. 3, 1982, pp.336-343.

2. [Ahuja & Schachter 83] **N. Ahuja** and **B. J. Schachter**, *Pattern Models*, John Wiley & Sons, New York, 1983.

3. [Ahuja & Tuceryan 89] **N. Ahuja** and **M. Tuceryan**, 'Extraction of early perceptual structure in dot patterns: integrating region, boundary, and component Gestalt', *Computer Vision, Graphics, and Image Processing*, Vol.48, 1989, pp. 304-356.

4. [Ahuja & Charan 95] **N. Ahuja** and **R. Charan,** 'Pixel matching and motion segmentation in image sequences' *Proc. of the 2nd Asian Conference on Computer Vision, ACCV'95*, Vol. I, The National Technological University Press, Singapore, 1995, pp. 310-314.

5. [Akl 85] **S. G. Akl**, 'Optimal parallel algorithms for selection, sorting and computing convex hulls', *Computational Geometry*, Ed., G. T. Toussaint, North-Holland, Amsterdam, 1985, pp. 1-22.

6. [Anderberg 73] **M. R. Anderberg**, *Cluster analysis for application*, Academic Press, New York, 1973.

7. [Apostol 71] **T. M. Apostol**, *Mathematical analysis*, Addision Wesley (1971).

8. [Attneave 59] **F. Attneave**, *Application of information theory to psychology : a summary of basic concepts, methods, and results*, New York : Holt, Rinehart and Winston, , 1959.

9. [Aurenham 91] **F. Aurenhammer**, 'Voronoi diagrams – A survey of fundamental geometric data structure', *ACM Comuting Suveys*, Vol. 23, 1991, pp. 345-406.

10. [Ballard & Brown 82] **D. H. Ballard** and **C. M. Brown**, *Computer Vision*, Pretice-Hall, Englewood Cliffs, N.J., 1982.

11. [Banerjee et. al. 91] **S. Banerjee, D. Mount**, and **A. Rosenfeld**, 'Pyramid computation of neighbor distance statistics in dot patterns', *CVGIP : Graphics Models and Image Processing*, Vol. 53, No. 4, 1991, pp. 373-381.

12. [Barrow 92] **J. D. Barrow**, 'Some statistical problems in Cosmology', *Statistical Challenges in Modern Astronomy*, Eds. E. D. Feigelson and G. J. Babu, Springer-Verlag, New York, 1992.

13. [Bently & Ottman 79] **J. L. Bently and T. Ottman**, 'Algorithms for reporting and counting', *IEEE Trans. Computers*, Vol. 28, 1979, pp.643-647.

14. [Bertin et. al. 92] **E. Bertin**, **R. Marcelpoil**, **J. M. Chassery**, 'Morphological algorithms based on Voronoi and Delaunay graphs : microscopic and medical applications', *Image algebra and morphological image processing III*, SPIE Ed., San Diego, 1992, pp. 356-367.

15. [Bhasker et. al. 88] **S. K. Bhaskar** and **A. Rosenfeld** and **A.Wu** 'Models for neighbor dependency in planar point patterns', *Tech. Report* CAR-TR-351, Center for Automation Research, University of Maryland, College Park, Mary land 20742, 1988.

16. [Blum 64] **H. Blum**, 'A transformation for extracting new descriptor of shape', *Proc. Symp. Models for the perception of speech and visual form*, Cambridge, MA : MIT Press, 1964.

17. [Border 90] **A. J. Border**, 'Strategies for efficient incremental nearest neighbor search', *Pattern Recognition*, Vol. 23, 1990, pp. 171-178.

18. [Carpenter & Grossberg 92] **G. A. Carpenter** and **S. Grossberg** (Eds), *Neural networks for vision and image processing*, MIT Press, Cambridge, MA, 1992.

19. [Charan & Ahuja 95] **R. Charan** and **N. Ahuja**, 'Curve detection in 3-D dot patterns' *Proc. of the 2nd Asian Conference on Computer Vision, ACCV'95*, Vol. I, The National Technological University Press, Singapore, 1995, pp. 199-203.

20. [Chaudhuri 85] **B. B. Chaudhuri**, 'Application of quadtree, octree, and binary tree decomposition techniques to shape analysis and pattern recognition', *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 7, No. 6, 1985, pp. 652-661.

21. [Chaudhuri 85a] **B. B. Chaudhuri**, 'An efficient hierarchical clustering technique', *Pattern Recognition Letters*, Vol. 3, 1985, pp.179-183.

22. [Chaudhuri 91] **B. B. Chaudhuri**, 'Some shape definitions in fuzzy geometry of space', *Pattern Recognition Letters*, Vol. 12, pp. 531-535, 1991.

23. [Chaudhuri & Majumder 93] **B. B. Chaudhuri** and **D. Datta Mahumder**, *Two-tone image processing*,Wiley Eastern Limited, New Delhi, 1993.

24. [Chaudhuri 93] **B. B. Chaudhuri**, 'Dynamic clustering for time imcremental data', *Pattren Recognition Letters*, Vol. 15, 1993, pp. 27-34.

25. [Chaudhuri 95] **B. B. Chaudhuri**, 'A new definition of neighborhood of a point in multi-dimensional space', *Pattren Recognition Letters*, Vol. 17, 1995, pp. 11-17.

26. [Chaudhuri 95a] **D. Chaudhuri**, *Some studies on density estimation and data clustering techniques*, Ph. D. Thesis, Jadavpur University, 1995.

27. [Chaudhuri & Chaudhuri 97] **D. Chaudhuri** and **B. B. Chaudhuri**, 'A novel multiseed nonhierarchical data clustering technique', *IEEE Trans. Syst. Man Cybernetics*, Vol. 27, No. 5, 1997.

28. [Choi 96] **H. K. Choi**, *New methods for image analysis of tissue sections*, Ph. D. Thesis, Uppsala University, 1996.

29. [Cox & Hinkley 74] **D. R. Cox** and **D. V. Hinkley**, *Theoretical statistics*, Chapman and Hall, London., 1974.

30. [Cover & Hart 67] **T. M. Cover** and **P. E. Hart**, 'Nearest neighbor pattern classification', *IEEE. Trans. Inform. Theory*, Vol. 13, 1967, pp. 21-27.

31. [Cucka 96] **P. Cucka**, **N. S. Netanyahm** and **A. Rosenfeld**, 'Learning in navigation : Goal finding in graph', *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 10, No. 6, 1996 pp.429-446.

32. [Datta & Parui 94] **A. Datta** and **S. K. Parui**, 'A robust Parallel thinning algorithm for binary images', *Pattern Recognition*, Vol. 27, No.8, 1994, pp. 1181-1192.

33. [Datta & Parui 97] **A. Datta** and **S. K. Parui**, 'Skeletons from dot patterns : a neural netwrok approach', *Pattern Recognition Letters*, Vol. 18, 1997, pp. 335-342.

34. [Datta et. al. 97] **A. Datta, T. Pal** and **S. K. Parui**, 'A modified self-organizing neural netwrok for shape extraction, *Neuro Compuintg*, Vol. 14, 1997, pp. 3-14.

35. [Dearholt 91] **D. W. Dearholt** and **F. Harary**, (Eds), *Proceedings of the frst workshop on proximity graphs*, (New Mexico State University, Las Cruces), Memoranda in Computer and Cognitive Science, 1991, MCCCS-91-224.

36. [Deo 90] **N. Deo**, *Graph theory with applications to engineering and computer science*, Prentice-Hall International, 1990.

37. [Di Gesu & Maccarone 98] **V. Di Gesu** & **M. C. Maccarone**, 'Solving image analysis problems with fuzzy sets', *Proc. of the Int. Workshop on Softcomputing and Intelligent Systems*, Indian Statistical Institute, Calcutta, 1998, pp. 156-169.

38. [Di Gesu & Maccarone 84] **V. Di Gesu** & **M. C. Maccarone**, 'A method to classify spread shapes based on possibility theory', *Proc. of the 7th Int. Conf. on Pattern Recognition*, IEEE Computer Soc., Montreal, Canada, 1984.

39. [Diggle 83] **P. J. Diggle**, *Statistical analysis of spatial point pattern*, Academic Press, New York, 1983.

40. [Dougherty 94] **E. R. Dougherty** (Ed), *Digital image processing methods*, Marcel Dekker, Inc., New York, 1994.

41. [Duda & Hart 73] **R. O. Duda** and **P. E. Hart**, *Pattern Classification and Scene Analysis*, Wiley, New York, 1973.

42. [Dubes & Jain 79] **R. Dubes**, and **A. K. Jain**, 'Validity studies in clustering methodologies', *Pattern Recognition*, Vol. 11, 1979, pp.235-254.

43. [Edelsbrunner et. al. 83] **H. Edelsbrunner**, **D. G. Kirkpatrick**, and **R. Seidel**, 'On the shape of a set of points in the plane', *IEEE Trans. Infor. Theory*, IT-29, 1983, pp.551-559.

44. [Edelsbrunner 87] **H. Edelsbrunner**, *Algorithms in Combinatorial Geometry*, New York, Springer-Verlog, 1983.

45. [Edelsbrunner 92] **H. Edelsbrunner**, 'Weighted α-shape',*Technical Report UIUCDCS-R-92-1760*, 1992.

46. [Edelsbrunner & Mücke 94] **H. Edelsbrunner**, and **E. P. Mücke,** 'Three-dimensional alpha shapes', *ACM Trans. on Graphics*, Vol. 13, No. 1, 1994, pp.43-72.

47. [Fairfield 79] **J. Fairfield**, 'Contoured shape generation : forms that people see in dot patterns, *Proc. IEEE Conf. Cybernatics and Society*, 1979, pp. 60-64.

48. [Fairfield 83] **J. Fairfield**, 'Segmenting dot patterns by Voronoi diagram', *IEEE Trans. Pattern Analysis and Machine Intelligence*, 1983, pp. 104-110.

49. [Faugeras 93] **O. Faugeras**, *Three-Dimensional Computer Vision : A Geometric Viewpoint*, The MIT Press, Cambridge, London, 1993, Chapt10., pp.403-482.

50. [Foley et. al. 90] **J. D. Foley, A. van Dam, S. K. Feiner** and **J. H. Huges**, *Computer graphis : principles and practice*, Addision-Wesley Publishing House, Massachusetts, 1990.

51. [Freeman 61] **H. Freeman**, 'On the encoding of arbitrary geometric configurations', *Computer Surveys*, Vol. 6, 1962, pp. 260-268.

52. [Frigui & Krishnapuram 96] **H. Frigui** and **R. Krishnapuram**, 'A robust algorithm for automatic extraction of a unknoen number of clusters for noisy data', *Pattern Recognition Letter* , Vol. 17, 1996, pp. 1223-1232.

53. [Fu 74] **K. S. Fu**, *Syntactic methods in pattern recognition*, Academic Press, New York, 1974.

54. [Gabriel & Sokal 69] **K. R. Gabriel** and **R. R. Sokal**, 'A new statistical approach to geographic variation analysis', *Systematic Zoology*, Vol. 18, 1969, pp. 259-278.

55. [Garai & Chaudhuri 88] **G. Garai** and **B. B. Chaudhuri**, 'A split and merge procedure for polygonal border detection', *Image Vision and Computing*. (In press).

56. [Geller & Huchra 89] **M.J. Geller** and **J. P. Huchra**, 'Mapping the universe', *Science*, Vol. 246, pp.897-903.

57. [Glass 69] **L. Glass**, 'Moire effect from random dots', *Nature*, Vol. 223, No. 9, 1969, pp.578-580.

58. [Gonzalez & Woods 93] **R. C. Gonzalez** and **R. E. Woods**, *Digital image processing*, Addision-Wesley Publishing Company, 1993.

59. [Gowda & Krishna 78] **K. C. Gowda** and **G. Krishna**, 'Aggolomerative clustering using the concept of mutual nearest neighborhood', *Pattern Recognition*, Vol. 10, 1978, pp. 105-112.

60. [Grenander 75] **U. Grenander**, *Abstract inference*, John Wiley, New York, 1975.

61. [Guibas et. al. 91] **L. Guibas, J. Pach** and **M. Sharir**, 'Sphere-of-influence graphs in higher dimensions', *Colloquia Mathematica Societatis Ja'nos Bolyai* 63. *Intuitive Geometry*, Szeged (Hungary) 1991, pp.131-137.

62. [Haralick et.al. 87] **R. M. Haralick, S. R. Sternberg**, and **X. Zhuang**, 'Image Analysis Using Mathematical Morphology', *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 9, 1987, pp. 523 - 550.

63. [Haralick & Shapiro 92] **R. M. Haralick,** and **L. G. Shapiro,** *Computer and Robot Vision,* Addision-Wesley Publishing Company, New York, 1992.

64. [Herbin et. al. 96] **M. Herbin, N. Bonnet** and **P. Vautrot**, 'A clustering method based on estimation of the probability density function and on the skeleton by the influence zone : Application to Image Processing', *Pattern Recognition Letter,* Vol. 17, No. 11, 1996, pp. 1141-1150.

65. [Hill & Taylor 92] **A. Hill** and **C. J. Taylor**, 'Model-based image interpretation using genetic algorithm', *Image and Vision Computing*, Vol. 10, 1992, pp295-300.

66. [Hough 62] **P. V. C. Hough**, 'Methods amd means for recognizing complex patterns', *U. S. Pattent* 3,069,654.

67. [Hu 62] **M. K. Hu**, 'Visual pattern recognition by momemnt invariants', *IRE Trans. Info. Theory*, Vol. 8, 1962, pp.179-187.

68. [Huang & Shieh 90] **J. S. Huang**, and **W. R. Shieh,** 'A heuristic method for separating clusters from noisy background', *Pattern Recognition*, Vol.23, No. 12. pp.147-157, 1990.

69. [Huang et. al. 94] **J. Huang, S. M. Dunn, S. M. Wienner, P. Decosta**, 'A method for detecting correspondence in stereo pairs of electron micrographs of networks', *J. Comput. Assist. Microsc.*, Vol. 6, No. 3, 1994.

70. [Ichino & Sklansky 85] **M. Ichino** and **J. Sklansky**, 'The relative neighborhood graph for mixed feature variables', *Pattern Recognition*, Vol. 18, 1985, pp. 161-167.

71. [Imai et. al. 85] **H. Imai, M. Iri,** and **K. Murota,** 'Voronoi diagram in the Laguerre geometry and its application', *SIAM Journal of Computing*, Vol. 14, No. 1, 1985, pp. 93-105.

72. [Jain & Dubes 88] **A. K. Jain**, and **R. Dubes**, *Algorithms for clustering data*, Prentice Hall, New Jersy, 1988.

73. [Jain 89] **A. K. Jain,** *Fundamental of digital image processing*, Prentice-Hall, Inc. Englewood Cliffs, N. J. USA, 1989.

74. [Jang & Chin 90] **B. K. Jang**, and **R. T. Chin**, 'Analysis of thinning algorithms using mathematical morphology', *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 12, 1990, pp. 541-551.

75. [Jaromczik & Toussiant 92] **J. W. Jaromczik** and **G.T. Toussiant**, 'Relative neighborhood graphs and their relatives', *Proc. IEEE*, Vol. 80., No. 9, 1992. pp. 1502-1517.

76. [Jarvis & Patrick 71] **R. A. Jarvis** and **E. A. Patrick**, 'Clustering using a similarity measure based on shared near neighbors', *IEEE, Trans. Computers.*, Vol. 11, 1971, pp.1025-1073.

77. [Jarvis 77] **R. A. Jarvis**, 'Computing the shape hull of points in the plane', *Proc. IEEE Comp. Soc. Conf. on Pattern Recog. and Image Proc.*, 1977, pp. 231-241.

78. [Kanade 81] **T. Kanade**, 'Recovery of three dimensional shape of an object from a single view', *Artificial Intelligence*, Vol. 17, 1981, pp. 409-460.

79. [Kennedy & Ware 78] **J. M. Kennedy** and **C. Ware**, 'Illusory contours can arise in dot figures', *Perception*, 1978, pp. 191-194.

80. [Koffka 35] **K. Koffka**, *Principles of Gestalt Psychology*, Harcourt Brace, New York, 1935.

81. [Kohonen 89] **T. Kohonen**, *Self-organizing and associative memory*, Springer, Berlin, 1989.

82. [Kumar et. al. 96] **M. A. Kumar**, **B. N. Chatterjee**, **J. Mukherjee** and **P. P. Das**, 'Representation of 2-D and 3-D binary images using medial circles and spheres', *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 10, No. 4, 1996, pp.365-387.

83. [Krikpatrick & Radke 85] **D. G. Krikpatrick** and **J. D. Radke**, 'A framework for Computational Morphology', *Computational Morphology*, Ed., G. T. Toussaint, North-Holland, Amsterdam, 1988, pp. 217-248.

84. [Lam et.al. 92] **L. Lam, S. W. Whan,** and **C. Y. Suen**, 'Thinning methodologies - a comprehensive survey' *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 14, No.9, 1992, pp. 869-885.

85. [Laurini & Thompson 92] **R. Laurini** and **D. Thompson**, *Fundamental of Spatial Information Systems*, The A.P.I.C. Series No. 37, Academic Press, London, 1992.

86. [Link & Zucker 88] **N. K. Link** and **S. W. Zucker**, 'Corner detection in curvilinear dot pattern', *Biological Cybernatics*, Vol. 59, 1988, pp. 247-256.

87. [Lowe 85] **D. G. Lowe**, *Perceptual Organization and Visual Recognition*, Kluwer Academic Publishers, 1985.

88. [Mandal et. al. 92] **D. P. Mandal**, **C. A. Murthy** and **S. K. Pal**, 'Determining the shape of a pattern class from sampled points in $\Re^2$', Int. J. General Systems, Vol. 20, 1992, pp.307-339.

89. [Mardia 72] **K. V . Mardia**, *Statistics of directional data*, Academic Press, London, 1972.

90. [Marshall 89] **S. Marshall**, 'Review of shape coding techniques' *Image and Vision Computing*, Vol. 7, No. 4, 1989, pp. 281-294.

91. [Marr 82] **D. Marr**, *Vision*, Freeman, San Francisco, 1982.

92. [Matheron 75] **G. Matheron**, *Random Sets and Integral Geometry*, Wiley, New York 1975.

93. [Matula & Sokal 84] **D. W. Matula** and **R. R. Sokal**, 'Properties of Grabriel graphs relevant to geographical variation research and the clustering of points in the plane', *Geographical Analysis*, Vol. 12, 1984, pp. 205-222.

94. [Medek 81] **V. Medek**, 'On the boundary of a finite set of points in the plane', *Computer Graphics and Image Processing*, Vol. 15, pp. 93-99, 1981.

95. [Miller et. al. 91] **G. Miller, S. H. Tenh** and **S. Vavasis**, 'A unified geometric approach to graph separators', *Proc. 32nd. IEEE Symp. on Foundation of Computer Science'*, 1991, pp.538-547.

96. [Mizo & Shimura 80] **R. Mizoguchi**, and **M. Shimura**, 'A nonparametric algorithm for detecting clusters using hierarhical structure', *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 2, 1980, pp. 292-300.

97. [Mori et. al. 92] **S. Mori**, **C. Y. Suen** and **K. Yamamoto**, 'Historical review of OCR research and development', *Proc. IEEE*, Vol. 80, 1992, pp. 1029-1058.

98. [Moss 67] **W. W. Moss**, 'Some new analytic and graphic approaches to numerical taxonomy, with an example from dermanyssidae (acari)', *Systematic Zoology*, Vol. 16, 1967, pp. 177-207.

99. [Murthy 88] **C. A. Murthy**, *On consistent estimation of classes in $\Re^2$ in the context of cluster analysis*, Ph. D. Thesis, Indian Statistical Institute, Calcutta, 1988.

100.[O'Callaghan 74] **J. F. O'Callaghan,** 'Computing the perceptual boundaries of dot patterns', *Computer Graphics and Image Processing*, Vol. 3, 1974, pp.141-162.

101.[Okabe et. al. 92] **A. Okabe, B. Boots** and **K. Sugihara**, *Spatial tessellations : Concepts and Applications of Voronoi Diagrams*, John Wiley and Sons, 1992.

102.[Ogniewicz 91] **R. Ogniewicz**, 'Regularization of the Voronoi medial axis', *Tech. Rep.* 127, IKT/Image Science Lab, ETH, Zurich, Switzerland, 1991.

103.[Pal & Majumder 86] **S. K. Pal** and **D. Dutta Majumder**, *Fuzzy mathematical approach to pattern recognition*, Wiley (Halsted), N. Y., 1986.

104.[Parui 84] **S. K. Parui,** *Some studies in analysis and recognition of 2-dimensional shape*, Ph. D. Thesis, Indian Statistical Institute, Calcutta, 1984.

105.[Parui et.al. 93] **S. K. Parui, S. Sarkar**, and **B. B. Chaudhuri**, 'Computing the shape of a point set in digital images', *Pattern Recognition Letters*, Vol. 14, 1993, pp. 89-94.

106.[Parui at. al. 95] **S. K. Parui, A. Datta**, and **T. Pal**, 'Shape approximation of arc patterns using dynamic neural networks', *Signal Processing*, Vol. 42, 1995, pp. 221-225.

107.[Pavlidis 78] **T. Pavlidis**, 'A review of algorithms for shape analysis', Comput. Graph. Image Proc., Vol. 7, 197, pp.243-258.

108.[Pavlidis 80] **T. Pavlidis**, 'Algorithms for shape analysis of contours and waveforms', *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 2, No. 4, 1980, pp.301-312.

109.[Pernus 88] **F. Pernus**, 'The Delaunay traingulation and the shape hull as tools in muscle fibre analysis', *Pattern Recognition Lettres*, Vol. 8, 1988, pp. 197-202.

110.[Petersen 56] **S. Petersen**, *Weather analysis and forecasting Vol 1 : Motion and motion sytems*, McGraw Hill, New York, 1956.

111.[Pfalz & Rosenfeld 69] **J. L. Pfalz** and **A. Rosenfeld**, 'Web grammers', *Proc. 1st. Int. Joint Conf. Artificial Intelligence*, pp. 609-619.

112.[Postaire et. al. 93] **J.G. Postaire**, **R.D. Zhang** and **C.L. Botte**, 'Cluster Analysis by binary morphology', *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 15, No. 2, 1993, pp.170-180.

113.[Prazdny 86] **K. Prazdny**, 'Psychophysical and computational studies of random-dot Moire patterns', *Spatial Vision*, Vol. 1, No. 3, 1986, pp.231-242.

114.[Preparata & Shamos 85] **F. P. Preparata** and **M. I. Shamos**, *Computational Geometry : An introduction*, Springer-Verlag, 1985.

115.[Radke 88] **J. D. Radke**, 'On the shape of a set of points', *Computational Morphology*, Ed., G. T. Toussaint, North-Holland, Amsterdam, 1988, pp. 105-136.

116.[Ragnemalm 93] **I. Ragnemalm**, The *Euclidean Distance Transform,* Dissertations, No. 304., Linköping Studies in Science and Technology, 1993, pp.7-11.

117.[Ranade & Rosenfeld 80] **S. Ranade** and **A.Rosenfeld**, 'Point pattern matching by relaxation', *Pattern Recognition*, Vol. 12, 1980, pp. 269-275.

118.[Rao 73] **C. R. Rao**, *Linear statistical inference and its applications*, John Wiely and Sons, 2nd Edition, New York, 1973.

119.[RayChaudhuri et.al. 94] **A. Ray Chaudhuri, S. K. Parui**, and **B. B. Chaudhuri**, 'Computing the shape of a dot pattern in digital images in parallel environment', *ICARCV'94 : Proc. of the 3rd. International Conf. on Automation, Robotics and Computer Vision*, Singapore Vol. 1, 1994, pp. 474-478.

120.[RayChaudhuri et.al. 95] **A. Ray Chaudhuri, B. Chanda**, and **B. B. Chaudhuri**, 'Detection of occluded circular objects by morphological operators', *Signal Processing*, Vol. 46, 1995, pp 233-242.

121.[RayChaudhuri et.al. 95a] **A. Ray Chaudhuri, S. K. Parui**, and **B. B. Chaudhuri**, 'Extraction of perceptual border of dot pattern', *Proc. of the 2nd Asian Conference on Computer Vision, ACCV'95*, Vol. III, The National Technological University Press, Singapore, 1995, pp. 62-65.

122.[RayChaudhuri et.al. 95b] **A. Ray Chaudhuri, S. K. Parui**, and **B. B. Chaudhuri**,, 'An Efficient Approach of Computing the Shape of Dot Pattern and Extraction of its Perceptual Border', *Pattern recognition, Image processing and Computer vision : Recent Advances*, P. P. Das and B. N. Chatterjee (Eds.), Narosa Publication, India, 1995, pp. 203-208.

123.[RayChaudhuri & Chaudhuri 96] **A. Ray Chaudhuri**, and **B. B. Chaudhuri**, 'An efficient morphological approach to computing external and internal shape of dot patterns', *Tech. Rep.*, CVPR/2/96 August, 1996, Computer Vision & Pattern Recognition Unit, Indian Statistical Institute, Calcutta, India.

124.[RayChaudhuri & Chaudhuri 97] **A. Ray Chaudhuri**, and **B. B. Chaudhuri**, 'A new *K*-nearest neighbor based approach on clustering and shape computation of complex dot patterns', *Tech. Rep.*, CVPR/1/97 April, 1997, Computer Vision & Pattern Recognition Unit, Indian Statistical Institute, Calcutta, India. Also communicated to the *IEEE Trans. on System Man Cybernatics Part B* (under revision). Extended abstract of this paper is published in 16th Int. CODATA Conference Proc. 8-12 November, 1988, New Delhi, India.

125.[RayChaudhuri et.al. 97] **A. Ray Chaudhuri, B. B. Chaudhuri** and **S. K. Parui**, 'A novel approach to computation of the shape of dot pattern and extraction of its perceptual border', *CVGIP : Computer Vision and Image Understanding*, Vol. 68, No. 3, 1997, pp. 257-275.

126.[RayChaudhuri et.al. 97a] **A. Ray Chaudhuri, A. Basu, S. Bhandari** and **B. B. Chaudhuri,** 'An efficient approach to consistent set estimation', *Sankhya –B* (In Press).

127.[RayChaudhuri et.al. 98] **A. Ray Chaudhuri, A. Basu, S. Bhandari** and **B. B. Chaudhuri**, 'Consistent set estimation in k-dimensions : an efficient approach', in the International Workshop on Statistical Techniques in Pattren Recognition, Sydney Australia and appeared in the Lecture Notes in Computer Science Series, Springer Verlag, NY 1998.

128.[Richards 86] **J. A. Richards**, *Remote senssing digital image analysis : an introduction*, Springer Verlag, London, 1986.

129.[Richards 91] **F. M. Richards**, 'The protien folding problem', *Scientific American*, Vol. 264, No. 1, 1992, pp.54-63.

130.[Riplay 81] **B. Ripley**, *Spatial statistics*, Wiely, New York, 1981.

131.[Rock & Palmer 90] **I. Rock**, and **S. Palmer** 'The legacy of Gestalt psychology', *Scientific American*, December, 1990, pp. 84-90

132.[Ronse 89] **C. Ronse,** 'A bibliography on digital and computational convexsity (1961-1988)', *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 11, No. 2, 1989, pp.181-190.

133.[Rosenfeld 69] **A. Rosenfeld,** 'Picture processing by computers' , *Computing Surveys*, Vol. 1, 1969, pp. 147-174.

134.[Rosenfeld 70] **A. Rosenfeld,** 'Connectivity in digital pictures' *J. Assoc. Comput. Mach.*, Vol. 17, 1970, pp. 146-160.

135.[Rosenfeld et. al. 76] **A. Rosenfeld, R. Hummel** and **S. Zucker**, 'Scene labeling by relaxation operation', *IEEE Trans. Systems Man Cybernet.*,Vol. 6, 1976, pp. 420-433.

136.[Rosenfeld 78] **A. Rosenfeld**, 'Clusters in digital pictures', *Inf. and Control*, Vol.39, 1978, pp. 19-34.

137.[Rosenfeld & Kak 82] **A. Rosenfeld** and **A. C. Kak**, *Digital picture processing*, Academic Press, New York, 1982.

138.[Rosenfeld 84] **A. Rosenfeld**, 'Fuzzy geometry of image subsets' *Pattern Recognition Letter*, Vol. 2, 1984, 311-317.

139.[Rosenfeld & Jolion 88] **A. Rosenfeld** and **J. M. Jolion,** 'Local operations on labelled dot patterns', *Tech. Report* CAR-TR-379, CV Lab., Center for Automation Research, University of Maryland, College Park, Mary land 20742, 1988.

140.[Rubin 15] **E. Rubin**, *Synsoplevede figurer. Studier i psykologisk analyse. Ko'benhavn og Kristiania.* 1915, (German ed. 1921, Visuell wahrgenommene Figuren.)

141.[Russ 95] **J. C. Russ,** *The image processing handbook*, C.R.C Press Ann Arbor, 1995.

142.[Sabourin & Mitiche 93] **M. Sabourin** and **A. Mitiche**, 'Modelling and classification of shape using Kohonen asociative memory with selective multiresolution', *Neural Networks*, Vol. 6, 1993 pp. 275-283.

143.[Satio et. al. 91] **T. Satio**, **J. I. Toriwaki** and **S. Yokoi**, 'Properties of extended digital α-hull with its applications to shape feature analysis of a fugure set', *Forma*, Vol. 6, 1991, pp. 9-25.

144.[Serra 82] **J. Serra,** *Image Analysis and Mathematical Morphology*, Academic Press Inc., New York, NY,1982.

145.[Serra 88] **J. Serra** (Ed), *Image Analysis and Mathematical Morphology Volume 2: Theoretical Advances*, Academic Press Inc., New York, NY,1988.

146.[Schonfeld & Goutsias 91] **D. Schonfeld** and **J. Goutsias**, 'Optimal morpological pattern restoration from noisy binary images', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 13, No. 1, 1991, pp. 14-28.

147.[Sternberg 86] **S. R. Sternberg**, 'Grayscale morphology', *Computer Vision Graphics and Image Processing*, Vol. 35, 1986, pp.333-355.

148.[Su & Chang 91] **T. H. Su** and **R. Ch. Chang**, 'Computing the constrained relative neighborhood graphs and constrained Gabriel graphs in Euclidean plane', *Pattern Recognition*, Vol. 24, 1991, pp. 221-230.

149.[Schikuta 96] **E. Schikuta**, 'Grid-clustering : An efficient hierarchical clustering method for very large data set', *Proc. of ICPR* , IEEE Press, 1996, pp. 1015-4651.

150.[Smith & Jain 84] **S. P. Smith** and **A. K. Jain**, 'Testing for uniformity in multi-dimensional data', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 6, No. 1, 1984, pp. 73-81.

151.[Stoyan et. al. 87] **D. Stoyan**, **W. S. Kendall** and **J. Mecke**, *Stochastic geometry and its application*, John Wiley and Sons, New York, 1987.

152.[Taylor 77] **P. J. Taylor**, *Quantitative Methods in Geography : An Introduction to Spatial Analysis*, Houghton Mifflin Company, Boston, 1977.

153.[Teh 88] **C. H. Teh**, 'On image analysis by the methods of moments', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 10, No. 4, 1988, pp. 496-513.

154.[Thonnat 88] **M. Thonnat**, 'Toward an automated classification of galaxies', in *Le monde des galaxies, Physics*, Springer Verlag, New York, 1988.

155.[Toussaint 80] **G. T. Toussaint**, 'The relative neighborhood graph of a finite palanar set', *Pattern Recognition*, Vol. 12, 1980, pp.261-268.

156.[Toussaint 88] **G. T. Toussaint**, 'A graph-theoretical primal sketch', *Computational Morphology*, Ed., G. T. Toussaint, North-Holland, Amsterdam, 1988, pp. 229-260.

157.[Toriwaki & Yokoi 88] **J. I. Toriwaki** and **S. Yokoi**, 'Voronoi and related neighbors on digitized two dimensional space with application to texture analysis', *Computational Morphology*, Ed., G. T. Toussaint, North-Holland, Amsterdam, 1988, pp. 207-228.

158.[Tuceryan et. al. 92] **M. Tuceryan**, **A. K. Jain** and **N. Ahuja**, 'Supervised classification of early perceptual structure in dot patterns', *Proc. 11-th IAPR International Conference on Pattern Recognition*, The Hague, The Netherlands, August 1992, Vol. 2, pp. 88-91.

159.[Uttal et.al. 70] **W. R. Uttal**, **L. M. Bunnel**, and **S. Corwin**, 'On the detectability of straight lines in the visual noise : An extension of French's paradigm into the milliseccond domain', *Perceptual Psychophisics*, Vol. 8. No. 6, 1970, pp. 385-388.

160.[Urquhart 82] **R. B. Urquhart**, 'Graph theoretical clustering based on limited neighborhood sets', *Pattern Recognition*, Vol. 15, 1982, pp.173-187.

161.[Veltkamp 89] **R. C. Veltkamp**, '2-D and 3-D computational morphology on the γ-neighborhood graph', *ACTA STEREOL*, Vol. 8, No. 2, 1989, pp. 595-600.

162.[Verveer & Duin 95] **P. J. Verveer** and **R. P. W. Duin**, An evaluation of intrinsic dimensionality, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 17, No. 1, 1995, pp. 81-86.

163.[Vincent 89] **L. Vincent**, 'Mathematical morphology on graphs' , *Signal Procesing*, Vol. 16, No. 4, 1989, pp.365-388.

164.[Vistnes 85] **R. Vistnes**, 'Detecting structure in random-dot patterns', Proc. Workshop on Image Understanding, DARPA, 1985, pp.350-362.

165.[Voss 91] **K. Voss**, 'Images, objects and surfaces in •'', *Int. J. Pattern Recog. Artif. Intell*, Vol. 5, 1991, pp. 797-808.

166.[Warnekar & Krishna 79] **C. S. Warnekar**, and **G. Krishna**, 'A heuristic clustering algorithm using union of overlapping pattern-cells', *Pattern Recognition*, Vol. 11, 1979, pp.85-93.

167.[Watabe & Okino 93] **H. Watabe** and **N. Okino**, 'A study on genetic shape design', *Proc. th. ICGA*, 1993, pp. 445-450.

168.[Wertheimer 38] **M. Wertheimer**, 'Laws of organization in perceptual forms', in *A Source Book of Gestalt Psychology*, Ed., W. D. Ellis, Harcourt Brace, New York, 1938. pp. 71-88.

169.[Worring & Smeulders 94] **M. Worring** and **A. W. M. Smeulders**, 'Shape of an arbitrary finite point set in $\Re^2$', *Journal of Mathematical Imaging and Vision*, Vol.4, No. 2, 1994 pp.151-170.

170.[Yahil & Brown 76] **A. Yahil**, and **M. B. Brown**, 'On separating clusters from background', *Technometrics*, Vol. 18, No. 1. 1976, pp. 55-58.

171.[Zadeh 65] **L. A. Zadeh**, 'Fuzzy sets', *Information and Control*, Vol. 8, 1965, pp.338-353.

172.[Zahn 71] **C. T. Zahn**, 'Graph theoretical methods for detecting and describing Gestalt clusters', *IEEE. Trans. Computer* Vol. 20, 1971, pp. 68-86.

173.[Zakara & Ozard 96] **P. Zakarauskas**, and **J. M. Ozard**, 'Complexity analysis for partitioning nearest neighbor searching algorithms', *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 18, No. 6, 1996, pp. 663-668.

174.[Zhou & Gordon 91] **X. Zhou** and **R. Gordon**, 'Generation of noise in binary images', *CVGIP: Graphical Models and Image Processing* Vol. 53, No. 5, 1991, pp. 476-478.

175.[Zucker & Hummel 79] **S. W. Zucker** and **R. A. Hummel**, 'Towards a low-level description of dot clusters : Labeling edge, interior and noise point, *Computer Graphics Image Processing*, Vol. 9, 1979, pp. 213-233.

176.[Zucker & Davis 88] **S. W. Zucker** and **S. Davis**, 'Points and endpoints : a size/spacing constraints for dot grouping', *Perception*, Vol. 17, 1988, pp. 229-247.

177.[Zusne 70] **L. Zusne**, *Visual Perception of form*, Academic Press, New York, 1970.

INDIAN STATISTICAL INSTITUTE LIBRARY

Processing slip

Source.....P/E/G date.23.12.05 Acc. no.T 156

Author.Ray chaudhuri, Amirban

Title.Some studies on shape of dot patterns

.....

CHECKING : Call no.

1 In library/earlier ed./later ed.

2 Earlier volumes/nos. of the series in lib.

3 New title ✓

Suggested/Approved by

Note :

23.12.05
Signature & date

CLASSIFICATION :

Class no.....x-ref.....

Scrutiny report : Signature & date

Class no.....x-ref.....

Note :

Signature & date

CATALOGUING ;

1 Main Card

2 Supp. Cards (Subject/Title/Series/Added entries/Shelf List) [Total no. of cards].....

3 Mechanical processing completed on.....

4 Cards filed

Note :

Signature & date