2001

A Hierarchical Shape Representation by Convexities and Concavities and Its Application to Shape Matching.

Gustavo Martinez

Louisiana State University and Agricultural & Mechanical College

Follow this and additional works at: https://digitalcommons.lsu.edu/gradschool_disstheses

Recommended Citation
https://digitalcommons.lsu.edu/gradschool_disstheses/244
INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

ProQuest Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600
A HIERARCHICAL SHAPE REPRESENTATION BY CONVEXITIES AND CONCAVITIES AND ITS APPLICATION TO SHAPE MATCHING

A Dissertation

Submitted to the Graduate Faculty of the Louisiana State University and Agricultural and Mechanical College in partial fulfillment of the requirements for the degree of Doctor of Philosophy

in

The Department of Computer Science

by

Gustavo Martinez
B.S., Instituto Politecnico Nacional, 1988
M.Sys.Sc., Louisiana State University, 1994
May, 2001
Acknowledgments

This work is the final step of my graduate studies. There were many people that helped me to get through these unforgettable years, so I would like to thank them here.

I would like to thank my advisor Dr. S. Kundu for his constant criticism and his effort in my formation as a graduate student. I also would like to thank the members of my committee, for reviewing early drafts of this work and their suggestions. Very specially to Dr. S. Kak one of my first professors at LSU and whom always had a minute to talk about life.

Thanks to Dr. Juan Nido for his guidance, advice and friendship. He helped me to clarify key points of this work. To Lucia Giordani who helped me to rewrite several paragraphs to make them more readable. To Guillermo Tonnsman for his friendship and constant support.

To my wife Lidice Garrido, we shared hard days, she always had a smile. To my family, Susy, Raul, Raul Jr., Paty, Sergio and Ricardo. They never lost faith in me.

To my classmates and friends: Rahman Tashakkori and Dr. C.G. Lim. With them I shared the hardships of the life of a graduate student.

To Dr. S. S. Iyengar and Dr. E. Khalaf for their support during my work as a graduate assistant in the Computer Science Department.
Table of Contents

Acknowledgements ................................................................. ii

List of Figures ........................................................................ vi

Abstract ................................................................................... ix

Chapter

1 Introduction .............................................................................. 1
  1.2 Representing Shape ............................................................ 1
  1.3 Shape from Digital Images .................................................. 4
  1.4 Shape Matching ................................................................. 4
  1.5 Overview of the Dissertation ............................................... 5
  1.6 Summary ............................................................................ 5

2 Previous Related Work .......................................................... 7
  2.1 The Human Matching of Shape ........................................... 8
    2.1.1 The Ability of the Human Brain to Complete Information ... 8
    2.1.2 The Human Perception of Shape is Hierarchical .............. 9
    2.1.3 The Information of Shape is Contained in the Boundary .... 9
    2.1.4 The Existence of Dominant Points on the Boundary ......... 10
    2.1.5 The Importance of Shape Orientation ......................... 11
  2.2 The Field of Shape Analysis ................................................ 12
    2.2.1 Transform Techniques ............................................... 13
    2.2.2 Space Domain Techniques .......................................... 14
  2.3 Syntactic Shape Description ............................................... 17
    2.3.1 Hierarchical Representation Based on Grids .................. 18
    2.3.2 Representation of Shape Using Convexity .................... 20
    2.3.3 A Hierarchical Decomposition Using the Convex Hull .... 21
List of Figures

1.1 Euclidean description of the triangle ABC ............................................................. 2
1.2 Analytic representation of the circle ................................................................. 3
1.3 Shape variation by effect of different grid size and shift. ................................. 5
1.4 Natural shapes may not be easy to characterize ............................................... 6
2.1 The human perception of lines. .............................................................................. 9
2.2 The hierarchical perception of shape. ................................................................. 10
2.3 The human perception of shape similarity. ......................................................... 11
2.4 A shape and its skeleton ..................................................................................... 16
2.5 A multi-resolution pyramid. ............................................................................... 19
2.6 Levels of detail in a multi-resolution pyramid. ................................................... 19
2.7 Decomposition of a polygon at concave angles. ............................................... 20
2.8 Hierarchical shape representation by concavities and convex hull. ................. 22
3.1 Interior point $P_A$, boundary point $P_B$ and exterior point $P_C$ of a figure. .... 26
3.2 Approximation of convex shape by $S=AdCeBA$, using $AB$ as the baseline. .... 29
3.3 Approximation of a non-convex shape. ............................................................. 30
3.4 Incremental reconstruction of the shape in (v) from its tree representation. .... 33
3.5 A curve can be represented by many polygonal chains. ..................................... 34
3.6 The error of two polygonal chains representing a curve. ...................... 35
3.7 Extracting the b-sequence from a digital image. ................................. 37
3.8 Directed triangles. ................................................................. 39
3.9 Shape 1 and its CCT representation. .............................................. 42
3.10 Shape 2 and its CCT representation. ............................................ 43
3.11 Shape 3 and its CCT representation. ............................................ 44
3.12 Shape 4 and its CCT representation. ............................................ 45
3.13 Shape 5 and its CCT representation. ............................................ 46
3.14 Same shape with two different trees, obtained by choosing a different base line. 47
3.15 Different shapes with different triangles and the same tree structure. . . 48
3.16 Two different shapes with their corresponding trees. ....................... 49
3.17 A dominantly convex segment formed by the curve from A to B and the base line $AB$. ................................................................. 50
3.18 Representation of a curve segment by a unique hierarchy of directed triangles. 52
3.19 The process of obtaining the CCT of a curve C. .............................. 53
3.20 Finding the largest triangles of the digital shape. ............................. 60
3.21 First few levels of CCT(S) in Figure 3.20. ..................................... 61
3.22 Silver maple leaf. ....................................................................... 62
3.23 Incremental reconstruction of a silver maple leaf. ............................ 64
3.24 Silver maple leaf rotated by 30 degrees. ........................................ 65
3.25 Silver maple partial reconstruction. .............................................. 65
4.1 Six possible directed versions of an ordinary triangle formed by points A,B,C. 75
4.2 The bh-representation of the directed triangles in Figure 4.1. ............ 76
4.3 The $\sigma_C$-similarity of Shape 1 and Shape 2. ............................... 82

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
4.4 The definition of the metric $M_A$ is not enough for shape matching. ........ 83
4.5 Matching two CCT's. ................................................................. 85
4.6 The Matching Trees obtained for the match of shapes $(S_1, S_2)$ and $(S_1, S_3)$. 88
4.7 Computing $M_T(S_1, S_2)$ from the Matching Tree. ....................... 90
4.8 Computing $M_T(S_1, S_3)$ from the Matching Tree. ....................... 91
4.9 Different shapes and their $M_{bh}, M_A$ measure. ........................... 94
4.10 Three skulls to be matched. ..................................................... 95
4.11 Skulls and their $M_A$ measure. ................................................ 96
4.12 Skulls and their $M_A$ measure. ................................................ 98
4.13 Measuring the match for shapes 1 and 2. .................................... 101
4.14 Measuring the match for shapes 1 and 3. .................................... 102
4.15 Measuring the match for shapes 1 and 4. .................................... 103
4.16 Measuring the match for shapes 1 and 5. .................................... 104
Abstract

Shape contains information. The identification and extraction of this information is not straightforward and is the main problem of Shape Analysis. The current trend in extensive manipulation of visual information makes this problem more important. The large volume of published works about shape analysis can be classified into two main categories: statistical shape analysis and structural shape analysis. The structural approach was proposed around thirty years ago by K.S. Fu. The large amount of works published since then proved the difficulty of defining a universal set of primitives for shape characterization.

The structural description of shape is based on the assumption that shape recognition is a hierarchical process. However, no effective general mechanism that captures the hierarchical structure has been found, and the existing representations may be applied to restricted applications.

We propose a new structural representation of shape using convexity. Instead of using a predefined set of primitives, we use two basic components to decompose a shape: convexity and concavity. The decomposition obtained results in a natural hierarchy of these basic components.

We represent the decomposition by a new shape descriptor: the Convexity-Concavity Tree (CCT), which is a binary tree. The CCT-representation is used for matching the shapes of two objects. The matching of two CCTs is also represented by a binary tree, called the Matching Tree. This tree represents the location and magnitude of the mismatch.
between corresponding convexities-concavities of the two shapes. Two shapes match if their corresponding CCTs match.

Some of the advantages of our representation method are: (1) it is information preserving, (2) it has the desired properties of a good description method: invariance, uniqueness and stability, (3) it is economical (4) it is robust in the presence of noise. Our matching method, based on CCT-representation is superior to other methods in terms of simplicity, ability to explain, and measuring mismatches. It may also be used with other well known methods.
Chapter 1

Introduction

Objective

The purpose this chapter is to present the problem which is our object of study, its relevance and the difficulties of such a task.

Overview

The structure of the chapter is the following: section 1.1 defines our problem. Section 1.2 presents some historical steps for representing shape. Section 1.3 presents some difficulties of dealing with discrete shapes. Section 1.4 relates shape representation with its recognition and section 1.5 gives an overview of the dissertation.

1.1 Geometric Shape

The study of shape can be traced back to the early days of geometry. A geometric object is a compact set of points in the Euclidean space to which we associate geometric properties: size, shape and position.

Shape is a very elusive property and there is no general agreement for its definition, however Dryden [17] proposed to define it as the property invariant to translation, rotation and scaling. We will take that definition.

Problem Definition: Given a 2-dimensional object, identify, extract and represent the essential information contained in its shape.
1.2 Representing Shape

The representation of shape has changed with the development of geometry. In Euclidean geometry, shape is described by constructions based on lines, angles, circle segments and arithmetic (see Figure 1.1). The introduction of analytic geometry by Descartes in 1637 marked a revolutionary step in geometry. Descartes introduced a coordinated system as the reference for the description of geometric objects. This reference system allowed the description of geometric shapes (i.e., a circle) by an equation (see Figure 1.2). The description of shape by a closed equation, is very useful since it allows the recognition of similar shapes, by algebraic procedures. Even though analytic geometry provides a very powerful tool for the characterization of shapes, it only investigates certain types of geometric objects, since it is restricted by algebra and elementary geometry [1].

Real world objects generally involve complex shapes, which are not easy to describe. The description of arbitrary curves and surfaces is the object of study of a more recent branch
\[ (x - h)^2 + (y - k)^2 = r^2 \]

Figure 1.2: Analytic representation of the circle
of mathematics, differential geometry by using differential calculus. More recently (1960's), the development of computers and their extended use for scientific applications, has created the need for characterizing shape from digital images, for modeling and recognition.

The recognition of shape is fundamental in many human activities, many of which still pose a challenge for computer science such as recognition of hand-writing, waveform recognition, etc.

Shape analysis is the area of pattern recognition whose main goal is to extract the essential shape information from digital images for its recognition. There are two main approaches for the study of shape: the featured based or statistical approach, and the structural approach. This work presents an improvement on the structural analysis of shape. In particular, we represent shape as a hierarchical structure of convex regions.

1.3 Shape from Digital Images

The description of shapes from digital images is a very important area of research with a myriad of scientific and technical applications. Digital images are discrete approximations obtained from the sampling of real world objects. The sampling process (or digitization) inherently introduces noise due to quantization, adding to the intrinsic difficulty of characterizing shape. Figure 1.3 show how digital shape (shaded) changes by a different choice of grid size or a relative shift between the object and the grid of the digitization.

1.4 Shape Matching

The recognition of shape depends heavily on its representation. In Euclidean geometry, an object is recognized by measuring straight lines and circles. In analytic and differential geometry, a geometric shape is represented by an algebraic or differential equation which is then used to recognize the geometric object.
The shaded region is the discrete shape corresponding to the circle.

Figure 1.3: Shape variation by effect of different grid size and shift.

In many practical applications characterizing a shape may not be an easy task since it is difficult to obtain a nice mathematical expression describing it. Take for example a natural shape such as shown in Figure 1.4. The shape shown corresponds to a sugar maple leaf. Its mathematical characterization may not be so amenable.

1.5 Overview of the Dissertation

This work is organized as follows: Chapter 2 looks at previous work on the representation of shape. Chapter 3 presents the new representation technique: the convexity-concavity tree. Chapter 4 shows the use of this shape representation for shape matching. Chapter 5 summarizes the results obtained, and presents the conclusions of this work.

1.6 Summary

Shape contains information. Shape is the geometric property invariant to scaling, translation and rotation. The representation of shape is essential for its recognition. The representation
Figure 1.4: Natural shapes may not be easy to characterize

of shape has evolved, as geometry has evolved. Simple shapes can be characterized by the traditional methods of elementary geometry or analytic geometry. *Complex* shapes are more difficult to characterize and are commonly encountered in nature. In addition, the digitization of images, increments the difficulty of the problem, since it is an inherently noisy process. Shape analysis is the area of Pattern Recognition that has as main goal to extraction of essential shape information for its recognition. This work presents a new shape descriptor, the Convexity-Concavity Tree, which allows for the representation of shape from binary images, based on the hierarchical description of the concavities and convexities of the shape analyzed.
Chapter 2

Previous Related Work

Objective

The purpose this chapter is to relate our work to the enormous literature on the field of shape analysis.

Overview

This chapter presents in section 2.1 some fundamental facts that are result of the study of the recognition of shape by humans. Section 2.2 presents a landscape of the field of pattern recognition. Section 2.3 describes the structural or syntactic approach to Pattern Recognition. In particular some mechanisms used to represent shape hierarchically (subsection 2.3.1) and works that use convexity to represent shape (subsection 2.3.2), closely related to our method. Section 2.4 summarizes the chapter.

2.1 The Human Matching of Shape

The recognition of forms (shapes) can still be posed as an example to show that computers are unable to compete with humans in tasks that require recognition. Thus if we aim to emulate the cognitive tasks of the brain (still the best cognitive system that we know about), we can not ignore some facts about the human recognition of shape.

The study of the human perception of shape is possibly as old as civilization. Scientists and philosophers alike have been puzzled by the cognitive process of the human recognition
of shape. Even though it is a process not well understood yet, its study has produced important experimental results which may throw some light on the task of automating the recognition of shape. We summarize some of them here, however the interested reader can find the details in more specialized books on the subject such as [46, 55].

- The human brain completes information.
- The information of shape is contained in the boundary of an object [28].
- The recognition of shape is hierarchical.
- There are points on the boundary of the shape that influence the recognition [54].
- The recognition of shape is sensitive to rotation [54].

While some facts about this process that have influenced the development of the methods in shape analysis, others however have been plainly ignored.

2.1.1 The Ability of the Human Brain to Complete Information

The natural ability of man to complete information has been known for some time, however, the explanation of the phenomena still remains a subject of research.

Consider Figure 2.1, where three sets of dots are drawn. Each set of dots is perceived as geometric figure (a triangle, a square and a circle), even though there are no lines connecting the dots. The brain completes the information with an illusory contour. The perception of illusory contours is also a subject of research.

This ability to complete information may prove to be a fundamental factor for the different performance of man and machine when dealing with noisy or incomplete information.
2.1.2 The Human Perception of Shape Is Hierarchical

This result has strongly influenced the methods of shape analysis since it led to the structural description of shape. Consider Figure 2.2, where a set of small characters (H) is shown. They are distributed in such a way that follows the shape of another character (S). The set of characters is first perceived as an S instead of many small characters. This is a typical example to show that the human perception of shape is hierarchical.

2.1.3 The Information of Shape Is Contained in the Boundary

The importance of the boundaries of a plane object was noticed since the early days of geometry: “A figure is that which is contained by any boundary or boundaries” (Euclid’s Elements. book III).

The importance of the edges for human recognition of shape was more recently noted by Wiener: “One of the most remarkable phenomena of vision is our ability to recognize an outline drawing. Clearly, an outline drawing of say, the face of a man, has very little resemblance to the face itself in color, or in the massing of light and shade, yet it may be a most recognizable portrait of its subject” [56].

Figure 2.1: The human perception of lines.
2.1.4 The Existence of Dominant Points on the Boundary

The existence of special points along the boundary that influence the recognition of a shape was proved by Attneave [5]. According to Attneave, visual information is highly redundant. The boundary is specially important since contains more information, due to the high contrast of light (as pointed out by Wiener). Among the points of the boundary there are some of them which are more relevant for the definition of the shape. In particular he used the points of high curvature of the boundary of a cat drawing and then he connected them by line segments to produce a simplified drawing of the cat which still allowed its reasonable recognition.

This experiment was very important for the methods of shape analysis, since it stimulated further research on curve partitioning, dedicated to find these points of high curvature which then can be connected by straight lines.

Figure 2.2: The hierarchical perception of shape.
Shapes (i) and (iii) are perceived as more similar than (i) and (ii) even though (ii) is the rotated version of (i) and (iii) is compressed version of (i).

Figure 2.3: The human perception of shape similarity.

2.1.5 The Importance of Shape Orientation

The human perception of shape is sensitive to rotation. This result is probably the most relevant result of the findings about the human recognition of shape, but surprisingly it has been largely ignored by the methods of shape analysis. This may be due to the supposition that shape recognition must be insensitive to rotation. There are some puzzling questions about the human recognition of shape, in particular the perception of similarity, which we can not ignore, because it may have a deep impact in the recognition of shape:

- Why some rotations of a plane figure influence their recognition and others do not?
- Why some modifications of the figure such as compression affect their perceived similarity less than no modifications of the figure other than rotation? (see Figure 2.3). In our method, shape (i) will be recognized as more similar to shape (iii) than to (ii).
2.2 The Field of Shape Analysis

Since the recognition of plane shapes is fundamental for many problems in Pattern Recognition, there has been an impressive collection of works published since the early days of the discipline.

In order to classify and relate the different methods Pavlidis proposed some criteria to group them [41]. Other criteria may be possible, but we will use these since they are commonly accepted.

These criteria are:

• According to the part of the shape used to characterize it:
  
  – External: the methods that use only the boundary of the shape.
  
  – Internal: the methods that use the whole region of the shape.

• According to the way to characterize the shape:
  
  – Scalar Transform: methods that represent the shape by an array of scalar measurements (features) obtained from it.
  
  – Space Domain: methods that represent the shape as a decomposition of its elements.

• According to the preservation of the shape information:
  
  – Information preserving: methods that allow the reconstruction of the shape from its representation.
  
  – Information non-preserving: methods that do not allow the reconstruction of the shape from its representation.
The way by which the shape is characterized has been established as a fundamental criteria to classify the methods. The scalar transform where vectors of numbers are used, is commonly known as the decision-theoretic or simply the statistical approach to characterize shape. The space domain representation is commonly known as the structural or syntactical approach, where syntactic data structures such as strings, trees or graphs, express the structural relation of the components of the shape, instead of vectors of numbers.

The structural approach is better suited for applications that require to capture the structural relations of the shape components. Our method falls into the structural representation of shape.

2.2.1 Transform Techniques

Most of the techniques in this group appeared in the early days of shape analysis. They consist on the extraction of a scalar measurements from the shape.

External Transform Techniques

The techniques in this group take boundary of the shape and transform it to a characteristic real function such as the tangent angle vs. arc length, proposed by Zhan and Roskies [60], or a complex function such as the techniques proposed by Granlund [27], Person and Fu [18] and Richards and Hammami [45]. Chang [14] obtained the characteristic function of the boundary by measuring the distance from every point in the boundary to the centroid of the shape. Once the boundary function is obtained, the Fourier transform is applied to the function and the coefficients obtained are consider to characterize the shape.

Internal Transform Techniques

Typical of this group is the methods of moments, which was originated in mechanics but was applied to shape analysis by Hu [32]. This method in not popular any more.
2.2.2 Space Domain Techniques

The techniques in this group take the shape and transform it to a graph or a string which describes the relations of the components, which can be spatial, temporal, etc.

External Space Domain Techniques

A significant effort has been made to capture the information contained in the boundary, curvature in particular (remember Attneave's result). Since the points of maximum curvature are the corners, there are many techniques that produce polygonal approximations of the shape.

The simplest method to obtain a polygon that approximates the shape is the Freeman [21] chain code. In Freeman's work every pair of consecutive points along a curve in a conventional direction is represented by a line segment, which is encoded by its slope. Thus the shape is described by a numeric chain.

Freeman and Davis [22] generate a polygon from the chain code. Higher order chain codes are also used by Freeman. Hsu and Mundy obtain a polygon based on the chain code also. Bribiesca and Guzman [13] indirectly use a polygonal approximation, using a differential code.

An obvious limitation of Freeman's chain code is its sensitivity to rotation, i.e. if we rotate the shape by ninety degrees then the chain that represents the shape changes. To solve this problem Bribiesca [12] proposed the vertex chain code (VCC) where the boundary is also represented by line segments but the code of the line is assigned by the kind of vertex that each point represents (straight, convex, concave).

Chain codes have the desirable property of being information preserving and are easy to obtain however they are highly redundant.
Following Atneave's work one expect to eliminate this redundancy by identifying the relevant points of the boundary. Thus using these points, one can define a simpler polygon which still captures the main properties of the shape.

One of the early methods methods to find such a characteristic polygon consists in defining line segments which represent groups of the intermediate points, by minimizing the square error, such as the method proposed by Pavlidis and Horowitz [43]. In this work a line is obtained by grouping points of the boundary under an error threshold. Once the threshold is reached a new line segment is started. An important drawback of this method is that it may produce disconnected line segments.

Davis [16] proposed an alternative method to obtain the characteristic polygon by using curvature maxima and maximal stretches from the boundary to define the line segments.

Yamamoto and Mori [58] use the convex hull of the contour to obtain the distances from the contour of the shape to the convex and use it to identify the lines of the polygon.

Higher order approximations such as splines have been also proposed but they are computationally more expensive. The use of splines is a technique that uses interpolation to approximate curves. They were introduced in computer graphics and computer aided design. They have good properties: they look good to humans, they approximate closely curves found in nature, etc.

Lewis and Graham [35] used damped splines for feature extraction. Other splines are also possible. Cohen [23] presented a technique for shape representation and matching using B-splines. B-splines are piecewise polynomial curves which are related to a guiding polygon.

Generating the characteristic polygon of the shape is just the first part of the task of shape recognition. The second part is the recognition of its elements to be able to
recognize the shape. Thus after a polygon is generated, a recognizing technique is used. Many techniques use syntactic methods to parse the polygon. Horowitz [31] transformed the waveform of an electrocardiogram to a string which is then parsed by a recognizer.

**Internal Space Domain Techniques**

The techniques in this group take the whole shape region and identify its components which are represented in a graph, describing the structure of the shape.

A typical of this group is the Medial Axis Transform (MAT), proposed by Blum [9]. The purpose of the medial axis transform is to obtain a skeleton of the shape, which is then transformed into a graph that represents the structure of the shape. The medial axis transform is very sensitive to variations in the boundary of the shape, thus small changes of the boundary of the shape may change the structure of the graph that represents the shape.

An alternative technique is the decomposition of a complex shape into convex regions, found using the boundary of the shape. These techniques are described into a separate section (section 2.3.2), since they are closely related to our method.
2.3 Syntactic Shape Description

The assumption that the human recognition of shape is a hierarchical process led directly to the structural representation of shape proposed by Fu [25]. In the structural approach, the shape of an object can be decomposed into primitive components or *shape primitives* (primitives), and then the theory of languages can be used to parse the shape [25]. The syntactic representation of shape allows the capability of describing a large number of complex shapes by using a small set of shape primitives, thus this approach seems very attractive, however it has two main drawbacks:

- It is not always clear how to define the primitives.

- The presence of noise may easily complicate the parsing.

The definition of primitives is influenced by the part of the shape that each technique uses as source of information to characterize it.

These techniques can be classified into two large groups: the techniques that decompose the boundary and the techniques that decompose the whole region of the shape into smaller regions. The shape primitives in the first group include: arcs and lines. The primitives in the second group include convex regions and skeletons. The definition of the primitives is crucial because the recognition of the shape relies on them.

In the syntactic approach it is desirable that the shape representation, aside from identifying the structural components, also describes a hierarchical decomposition, that ease the recognition. However, this is difficult to achieve. Most techniques do not provide this advantageous property and they have to rely on a complementary technique: the hierarchical description of shape based on grids, sometimes refered as multiresolution pyramids, which are described next.
2.3.1 Hierarchical Representation Based on Grids

A simple mechanism to implement a hierarchical representation of a region is achieved by a multi-resolution pyramid, where the shape is represented in several layers, at different levels of resolution, being the original image the highest resolution layer and the upper layers derived from this one the levels of lower resolution.

For every layer but the highest resolution one, each pixel is related to a group of four bits in the preceding level and this pixel is black (part of the shape) if all the related pixels in the preceding layer are part of the shape.

Similarly, each pixel in the following layer is related to a group of four pixels in this layer and analogously, that pixel will be in the shape if the related pixels are (under the criterion assumed) in the shape. This is shown in Figure 2.5.

Figure 2.6 shows a shape represented in a multiresolution pyramid. The leftmost shape shows the highest resolution layer (original image). The shape in the middle of the figure shows the second layer. The right-most shape shows the third level of the representation. Note that the in the third level, the shape was broken into two disconnected regions. This is an important inconvenience of this approach. This happens because an arbitrary grid size is chosen. There are other serious drawbacks for this multiresolution representation, i.e., a slight shift between the object and the grid can change the description of the shape.

Another mechanism for representing the region the shape at the pixel level is the representation of a shape by a quad-tree, proposed by Samet [48]. The quad-tree is a tree description of the hierarchy of the pyramid which can be more economical than the pyramid. Since the quad-tree representation is based on the multi-resolution pyramid thus suffers from the same limitations.
Figure 2.5: A multi-resolution pyramid.

Figure 2.6: Levels of detail in a multi-resolution pyramid.
Small variations of the boundary can generate large convex regions

Figure 2.7: Decomposition of a polygon at concave angles.

2.3.2 Representation of Shape Using Convexity

The methods that use convexity assume that the boundary contains enough information to split the shape into its convex components. Many of them first approximate the boundary of the shape by a polygon and then decompose it into convex components.

An early method proposed by Pavlidis, decomposes the polygon at concave angles [41, 24]. This method is very sensitive to noise, since small concavities of the boundary generate large convex pieces (see Figure 2.7). Furthermore, small concavities may introduce additional convex regions which drastically change the graph representing the decomposition.

The decomposition into a set of non-overlapping number of polygons is called a partition. If the polygons can overlap, the decomposition is called a covering. The decomposition of a polygon into a minimal number of polygons has been studied by O'Rourke and others [38, 15].
There are two variants of a minimal decomposition, according to the criteria of the points that are used for the decomposition: (1) Simple decomposition, where the polygon can be decomposed using only points that belong to the boundary; (2) Steinner decomposition if the points of the subpolygons are not in the original polygon.

The complexity of the decomposition of many polygons is still unknown, but Chazelle has found a solution for the minimal decomposition of polygons using Steinner points in polynomial time [15]. Note that even when a decomposition has been obtained, the primary polygons found have to be processed or recognized. Until now the representation of shape using convexity has been more a subject of theoretical interest than of practical application [53].

2.3.3 A Hierarchical Decomposition Using the Convex Hull

A hierarchical decomposition of shape using its concavities was proposed by Batchelor [7]. A recursive application of the convex hull [38], provides a hierarchical description of the shape, see Figure 2.8. The first node \((C_h_1)\) in the tree (which is not necessarily binary) is the convex hull of the whole shape. The children of \(C_h_1\) are the convex hulls of the concavities of the shape \((C_h_2\) and \(C_h_3\)). The next level of the three represent the convex hull of the concavities of \(C_h_3\). This approach is very interesting but it has a serious drawback: it fails to uniquely represent a convex region.

Although not very effective, this work as well as Pavlidis' contain two ideas that are worthwhile exploring, and which guided our research:

- To use boundary information to obtain a hierarchical description based on convex regions.
- To take into account the convex regions that are not in the shape (concavities).
Figure 2.8: Hierarchical shape representation by concavities and convex hull.
Note that the real problem of using convex regions for shape representation is the problem of representing uniquely a convex region.

2.4 Summary

The recognition of shape is a central task in many problems of Pattern Recognition.

Many of these problems can be solved easily by humans, thanks to our natural ability to recognize shape information. Thus the human recognition of shape is a process that has been studied extensively. We know, thanks to several experimental studies the following facts: humans complete information, the perception of shape is hierarchical, the information of shape is contained in the boundary of the object, there are special points on the boundary that influence the recognition and the human recognition of shape is very sensitive to rotation.

The methods of shape analysis have incorporated some of these results from the early works in shape analysis however, other results have been plainly ignored by the methods of shape analysis, such as the sensitivity of the recognition to rotation. This happened perhaps because it conflicts with the mathematical notion that shape is invariant to rotation. Thus there are several facts in the human perception of shape that are not yet explained.

The vast amount of methods in shape analysis could be classified into two main approaches: the statistical approach and the structural approach. Each of these groups can be further divided according to the part of the shape that they process: the boundary of the object (external) or the whole region of the shape (internal).

The structural approach is an interesting approach since it breaks the shape into a number of simpler components. The methods that use the information of the boundary approximate the boundary by a polygon although there are some approaches that use higher
order approximations such as splines. Within the methods that use the whole region of the shape, there are several that use convexity, however, none of them works for practical applications and they are studied more for theoretical purposes.

The fact that shape perception is hierarchical led to the implementation of mechanisms that describe shape hierarchically such as multiresolution pyramids and quad-tress. These approaches have serious drawbacks that make them useless for practical applications, since they use a predefined size of grid. Other attempts to achieve a hierarchical description based in the decomposition of the shape into convex components, using the convex hull have been proposed, however they face a singular problem: there is no proper way of describing convexity.

Our method is a structural method which uses only the information on the boundary of the shape to produce a hierarchical description of the shape by using minimal convex elements as primitives.
Chapter 3

Representing Shape by Convexities and Concavities

Objective

In this chapter we present our representation method. We propose two primitives, common to any general shape: convexity and concavity. We use the notion of a triangle as a primitive component in a shape decomposition which is based on the convexities and concavities of the shape. This is different from the usual decompositions (of say a polygon), where the polygon becomes the union of its decomposed triangles, in our case the polygon may become an “algebraic sum” of the component triangles due to convexities and concavities.

Overview

Section 3.1 presents the basic notation and terminology. Section 3.2 presents the general principle of our method. Section 3.3 presents the formal method to obtain the shape descriptor. Section 3.4 presents the results obtained. Section 3.5 summarizes the chapter.

3.1 Basic Terminology

As pointed out in Chapter 1, geometric objects have three properties: size, shape and position. In this work we will consider only 2-dimensional objects, which are basically connected regions bounded by a closed curve [57]. Since the concept of shape is so subjective, to avoid any ambiguity, we will refer to a 2-dimensional object as a plane figure, following
the Euclidean notion [19]. We consider the figure to be defined by the whole region instead of its boundary. The size of a figure is its area. The shape is the particular set of points that constitute the figure. The position of the figure is the location of these points in the Euclidean space. We can now talk more about the shape of a figure. When we scale a figure $F$, we obtain another figure $F'$, with the same shape. We assume that a figure is bounded (fits entirely within some fixed circle). We condition shapes without any holes in them, by the moment but we believe that our technique can be extended to more cases. The points of the plane can be divided into three classes, with respect to a given figure, interior, exterior and boundary points, (see Figure 3.1). A point is an interior point ($P_A$) if it is the center of
a circle, sufficiently small, which belongs entirely to the figure. A point is an \textit{exterior point} \((P_B)\) if, it is the center of a circle, sufficiently small, that does not contain any point of the shape. A point is a \textit{boundary point} \((P_C)\), if it is the center of a circle that contains both, interior and exterior points. A figure \(F\) is a set of points in the plane with the following properties:

\textbf{Property 1:} If a set \(F\) is a figure, then all the points of its boundary belong to \(F\) as well as the interior points.

\textbf{Property 2:} If \(P\) is a point in the boundary of \(F\), and \(C\) is a circle with center \(P\), then there are interior points and exterior points inside \(C\).

\textbf{Property 3:} The boundary of \(F\), consists of a simple curve (which never crosses itself).

The one-dimensional description of the boundary of \(F\), in either direction, clockwise (CW) or counterclockwise (CCW), is called a \textit{path}. In this work we assume a CCW direction.

\section*{3.2 General Principle for Approximating Shapes}

In order to introduce the general principle of our method, consider the following problem.

\textbf{Problem:} Given a plane figure \(F\), such as shown in Figure 3.2(i), formed by a curve segment and a straight segment, approximate its area.

Consider the following solution: First, consider the straight segment of the boundary of \(F\). Mark the points that define the segment: say A and B. Now, define a segment \(A'B'\) parallel to \(AB\), which is tangent to the boundary of the object and passes through a point \(C\) which has maximum distance from \(AB\), if this point is not unique (e.g. \(C'\)) take the first point found by traversing the boundary of \(F\) in CCW direction.

Now, trace two line segments: \(AC\) and \(CB\). We have now the triangle \(ABC\), (or simply \(\triangle ABC\)) and two smaller versions of the problem. One formed by the curve segment \(AC\)
\( \triangle ABC \) and two smaller versions of the problem. One formed by the curve segment AC and the straight segment \( \overline{AC} \). The other formed by the curve segment CB and the straight segment \( \overline{CB} \).

If we apply the same principle to both subproblems, we obtain \( \triangle ACd \) and \( \triangle CBe \).

We can approximate \( F \), by \( S \), where:

\[
S = \triangle ABC + \triangle ACd + \triangle CBe.
\]

If we continue the process, by bisecting, the curve segments, we will find smaller and smaller triangles, that will add to our approximation. Naturally, the area approximation \( S \) will be eventually, very close to the area of \( F \). This method was used by Archimedes [29], to determine the area of a parabolic segment cut by a straight line.

3.2.1 Representing the Approximation

As we showed in the last problem, we are approximating a shape by a polygon, which is constructed by the addition and subtraction of convex areas (triangles), which become smaller at every step.

We can represent this hierarchy of triangles in a binary tree. In order to make our representation unique, we will establish a conventional direction to direct our segments: we will label the boundary of \( F \), in the CCW direction, starting from the right-most point \( A \), and ending at the leftmost point \( B \), see Figure 3.2(i). The representation of Figure 3.2(i) is shown in Figure 3.2(ii). The root represents the first triangle and each child represents the adjacent triangle along its respective side. Since this tree represents a hierarchy of the convex shapes (triangles), which are added or subtracted to obtain the shape to represent, we call it the Convexity-Concavity Tree of the Shape or simply CCT(S).
C = first boundary point with maximum distance to \( \overline{AB} \)
d = first boundary point with maximum distance to \( \overline{AC} \)
e = first boundary point with maximum distance to \( \overline{CB} \)

(i) Decomposition of the shaded shape by \( S = \triangle ABC + \triangle ACd + \triangle CBe \).

(ii) Tree Representation of \( S \).

Figure 3.2: Approximation of convex shape by \( S = AdCeBA \), using \( \overline{AB} \) as the baseline.
Consider now a general non-convex shape as the shown in Figure 3.3(i), which now contains a convexity and a concavity in the segment ACB. Now, by repeating the general principle, we will find point C first, and then, the point d, by moving the parallel segment $\overline{AC'}$ to the left of $\overline{AC}$. Similarly, we find e. If we continue the process, we find the points f and g.

We represent the shape in Figure 3.3(i) by the Tree shown in Figure 3.3(ii). Note that the tree contains now positive and negative triangles. The signed triangles will be used later to reconstruct the shape from its CCT description.
3.2.3 Shape Reconstruction

Once we obtain the decomposition of a shape and represent it in a binary tree, we can reconstruct it, because our tree contains the elements and position of our decomposition.

An algorithm for the reconstruction of the shape using breadth first search of the tree is presented (Algorithm 1). The algorithm takes as an input the hierarchy of triangles obtained from the decomposition of the shape. The reconstruction starts at the root and proceed in a top-down fashion, until all the elements represented in the nodes of the tree are included. As we proceed and include more nodes of the tree in our reconstruction, we obtain more details of the original shape.

The algorithm uses a queue of triangles contained in the nodes of the tree. The reconstruction of the shape is a figure $S$ which initially is empty and in the first execution of (step 2.a) contains the triangle corresponding to the root of the tree. Every time the loop (2) is executed, a node is dequeued (step 2.a) and its corresponding triangle is added or subtracted (step 2.b) and then the children (if any), of that node are enqueued (steps 2.c and 2.d). The process ends when the queue is empty. Note that some information about the sign of the triangle must be stored in each node.

Figure 3.4 shows the steps of the incremental reconstruction of the shape shown in Figure 3.3, using Algorithm 1.

3.3 Representing Shape as a Hierarchy of Convexities

After presenting the general principle, we now present formally the method to describe an arbitrary shape as a Convexity-Concavity Tree. We first define formally the concepts that we will use to explain our method and then present the formal method to represent a shape. We present the algorithms to implement our method.
Algorithm 1: Breadth First Reconstruction of a Shape from its CCT.

Input: Binary Tree $T$ representing a shape $S$ (as a combination of triangles as in fig. 3.2).
Output: Shape $S$.

1. [Initialize the queue $Q$ with root of $T$ and $S$.]
   
   $Q = \text{enqueue}($root of $T)$;
   
   $S = \emptyset$;

2. while $Q$ not empty do
   
   (a) node = dequeue($Q$);
   
   (b) $S = S + \text{the directed triangle } \triangle(\text{node})$;
   
   (c) if node $\rightarrow \text{leftson}$ exists then $Q = \text{enqueue}(\text{node} \rightarrow \text{leftson})$;
   
   (d) if node $\rightarrow \text{rightson}$ exists then $Q = \text{enqueue}(\text{node} \rightarrow \text{rightson})$;

3.3.1 Definitions

Definition 3.1 A digital image $D$ is a sampled and quantized function of two dimensions, generated by an optical device. The sampling is done on an equally spaced and rectangular grid pattern of width $M$ and height $N$. The value sampled is mapped (quantized) to an integer value (usually in the range 0 to 255) which represents the gray level $I(x,y)$ for $1 \leq x \leq M, 1 \leq y \leq N$.

Definition 3.2 A binary image is a digital image quantized to a binary number i.e. $I(x,y)=0$ or 1.

Definition 3.3 A digital shape $S$ in a binary image is a connected region, without any holes, where $I(x,y)=1$. The regions where $I(x,y)=0$, are called the background $G$. 

Republished with permission of the copyright owner. Further reproduction prohibited without permission.
Figure 3.4: Incremental reconstruction of the shape in (v) from its tree representation.
(i) A directed curve $C$. (ii) Two polygonal chains representing $C$.

Figure 3.5: A curve can be represented by many polygonal chains.

**Definition 3.4** A directed curve $C$ is a simple continuous curve of finite length with a conventional direction assigned. We denote the initial point of $C$ by $P_s$ and the final point of $C$ by $P_e$.

The directed curve $C$ can be represented as a continuous function $C : [a, b] \rightarrow \mathbb{R}^2$ with $C(a) = P_s$ and $C(b) = P_e$.

**Definition 3.5** A polygonal chain of a directed curve $C$ is a finite sequence of points of $C$, $P = (P_1, P_2, \ldots , P_n)$, where $P_1 = P_s$, $P_n = P_e$ and $P_i \neq P_j$ if $i \neq j$ and the points $P_j$'s are the images of a sequence of points $x_1 = a < x_1 < \ldots < x_j < \ldots < x_n = b$ of the interval $[a, b]$.
(i) The polygonal chain A.  (ii) The polygonal chain B.

Figure 3.6: The error of two polygonal chains representing a curve.

A curve C can be represented by many polygonal chains (see Figure 3.5). Two different polygonal chains $A = (P_1, P_2, ..., P_m)$ and $B = (P'_1, P'_2, ..., P'_m)$ may represent curve C (see Figure 3.6).

Note that the areas between the curve and the polygonal chain are the measure of the quality of the representation of C by the polygonal chain. It is generally true that when the chain contains more points of C, which are not co-linear, the chain represents better C, since those areas become smaller.

Definition 3.6 **Boundary sequence**

A boundary sequence or b-sequence $B$ is a polygonal chain of the curve that defines the boundary of a shape $S$. 

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
A polygonal chain can be obtained by a process of sampling or digitizing (see Figure 3.7). The shape $S$ is described by a boundary sequence that includes all its boundary points, obtained by traversing once its boundary in CCW direction.

As we mentioned in Chapter 2, there is experimental evidence showing that there are important points along the boundary that influence the recognition of an object [5]. The identification of the points that define the boundary is not trivial and there have been several works that address that problem [43, 62, 50, 61]. We introduce formally, the concept of dominant point.

**Definition 3.7** Given a $b$-sequence $B = \langle P_1, P_2, ..., P_n \rangle$, $P_i \neq P_j$, $P_d$ is the dominant point of $B$ if it has the largest distance to the straight line defined by $P_s$ and $P_e$ and is the first of such points.

The line defined by $P_1$ and $P_n$ divides the plane into two regions. If we direct the straight line from $P_1$ to $P_n$ ($P_1P_n$), then the region on the right of $P_1P_n$ is called positive and the region on the left of $P_1P_n$ is called negative.

If the dominant point is located on the positive side of $P_sP_e$, then we say that the $b$-sequence is dominantly convex or equivalently, that the $b$-sequence represents a boundary dominantly convex.

If the dominant point is located on the negative side of $P_sP_e$, then we say that the $b$-sequence is dominantly concave or equivalently, that the $b$-sequence represents a boundary dominantly concave.

Note that if $P_i$ is the dominant point for the $b$-sequence $B = \langle P_1, P_2, ..., P_n \rangle$, then $P_i$ may not be a dominant point for the $b$-sequence $B' = \langle P_n, P_{n-1}, ..., P_1 \rangle$. 

36

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
Figure 3.7: Extracting the b-sequence from a digital image.
Definition 3.8 Directed Triangle.

Consider a triangle $\triangle$ with vertices $A$, $B$ and $C$. Let any of the sides of $\triangle$ be the base, say $\overrightarrow{AB}$. Direct $\overrightarrow{AB}$ in either direction, say from $A$ to $B$ ($\overrightarrow{AB}$).

Direct the remaining sides such that they form a directed path from $A$ to $B$ ($\overrightarrow{AC}$ and $\overrightarrow{CB}$). The triangle obtained is called directed. Thus, there are two directed paths from $A$ to $B$. One is formed by the base $\overrightarrow{AB}$. The other is formed by the sides $\overrightarrow{AC}$ and $\overrightarrow{CB}$. The first side $\overrightarrow{AC}$ is called $l_1$ of $\triangle$ and the second side is called $l_2$ of $\triangle$.

We denote a directed triangle by listing its vertices, writing first the vertices that form its base (i.e. in $\triangle ABC$, $\overrightarrow{AB}$ is the directed base). If the region enclosed by a directed triangle $\triangle ABC$, is located to the right hand side of $\overrightarrow{AB}$, then the triangle is called positive and is denoted by $+\triangle ABC$. If the region enclosed by $\triangle ABC$, is located to the left hand side of $\overrightarrow{AB}$ then the triangle is called negative and is denoted by $-\triangle ABC$. The directed triangle $\triangle ABC$ of base $AB=b$, and altitude=0, is called a degenerate triangle of base $b$. We consider the area of a positive directed triangle as positive area and the area of a negative directed triangle as negative area. See Fig. 3.8.

Definition 3.9 A Tree is a digraph with a nonempty set of nodes such that:

- There is exactly one node, called the root of the tree, which has indegree (the number of arcs that terminate at that node) of 0.

- Every node other than the root has indegree of 1.

- For every node $a$ of the tree there is a directed path from the root to $a$.

We represent trees with the root node at the top and all arcs directed downwards, leaving the arrowheads of the arcs implicit.
Definition 3.10 Let \( a \) and \( b \) be two nodes of a tree \( T \). If there is an arc from \( a \) to \( b \), then \( a \) is said to be the father of \( b \) and \( b \) is a son of \( a \).

If there is a directed path from node \( a \) to node \( b \), then the node \( a \) is said to be the ancestor of \( b \) and \( b \) is the descendant of \( a \).

The subdigraph consisting of node \( a \) and all its descendants is a subtree of \( T \) and \( a \) is the root of the subtree.

Definition 3.11 A Binary Tree is a tree in which every node has at most two sons, and every node other than the root is specified to be either the left son or the right son of its father.

Definition 3.12 A Convexity-Concavity Tree of a directed curve \( C \), denoted \( \text{CCT}(C) \) is a binary tree, which represents the decomposition of \( C \) into a hierarchy of directed triangles.
Every node of the CCT(C) is a directed triangle that represents a dominantly convex or dominantly concave b-sequence.

**Definition 3.13** A Convexity-Concavity Tree of a shape S (closed curve), denoted CCT(S) is a binary tree with at most two subtrees: CCT₁(S) and CCT₂(S), with CCT₂(S) = CCT₁(C₁), and CCT₂(S) = CCT₁(C₂), where C₁ and C₂ are respectively, the right and left directed segments of the boundary of S, split over a base line.

Figures 3.9 - 3.13 present five different polygonal shapes and their corresponding CCTs. Figure 3.9 shows shape 1 and its CCT. Shape 1 is a polygon with 7 vertices which are enough to describe the shape boundary. Note that the line $\overline{AC}$ is the base line for the decomposition of the boundary. Thus the boundary is split into two chains $C₁ = \langle A, e, B, f, C' \rangle$ and $C₂ = \langle C, g, D, A \rangle$. CCT(Shape 1) has two subtrees: CCT₁(C₁) and CCT₁(C₂). CCT₁(C₁), the left subtree of CCT(Shape 1), represents the right hand side boundary of shape 1, and CCT₁(C₂), the right subtree of CCT(Shape 1), represents the left hand side boundary of shape 1. The CCT representation for the remaining shapes is obtained in a similar fashion. Thus CCT(Shape 1) contains 5 nodes, each representing a directed triangle obtained from the shape boundary. Observe that only the terminal nodes represent the convex and concave segments of the boundary.

Figure 3.10 shows shape 2 and its CCT representation. Shape 2 also is formed by 7 vertices but the shape looks quite different. CCT(Shape 2), however has the same tree structure of CCT(Shape 1). It also contains 5 nodes and each node in has the same sign. The size of the directed is different though.

Figure 3.11 shows shape 3 and its CCT representation. Observe that the structure of CCT(Shape 3) is the same that CCT(Shape 1), however not all the signs of the directed
triangles are the same. Figure 3.12 shows shape 4 and its CCT representation. Shape 4 is a slight variation of Shape 1, in particular it contains 2 more vertices on the right hand side of the boundary. CCT(Shape 4) contains now 7 nodes. Observe that the structure of CCT(Shape 4) is quite similar to the structure of CCT(Shape 1), except that CCT(Shape 4) contains two more nodes.

Figure 3.13 shows shape 5 and its CCT representation. Shape 5 is also a slight variation of Shape 1. CCT(Shape 5) contains 7 nodes, as CCT(Shape 4) does. Observe that the structure of CCT(Shape 5) almost is the same that CCT(Shape 1), except that CCT(Shape 5) contains two more nodes, in the same way of CCT(Shape 4), but now they are in a different location of the tree, thus producing a variation on the structure of the tree.

Note that the CCT of a shape will change if we choose a different base line as Figure ?? shows. We choose by convention to take the top-most (and left-most if not unique), and bottom-most (and right most if not unique) points to define the base line.

Also note that different shapes may have the same tree structure but different directed triangles as it is shown in Figure 3.15 shows that the structure of the CCT by itself does not reflect the variations of the shape, i.e. the structure of the tree is the same but the sign of the directed triangles is not. Moreover, even when the structure of the CCTs are the same and the signs are the same, the size of the directed triangles can still allow the variation of shape, as in the case of CCT(Shape 1) and CCT(Shape 2).

Figure 3.16 shows how variations of shape are reflected in the CCT: if additional convexities or concavities are added to the boundary, extra nodes are added to the CCT. Moreover the nodes added are well localized within the tree structure, according to the place of the shape where they appear.
Figure 3.9: Shape 1 and its CCT representation.
Figure 3.10: Shape 2 and its CCT representation.
Figure 3.11: Shape 3 and its CCT representation.
Figure 3.12: Shape 4 and its CCT representation.
Figure 3.13: Shape 5 and its CCT representation.
Figure 3.14: Same shape with two different trees, obtained by choosing a different base line.
Figure 3.15: Different shapes with different triangles and the same tree structure.
Figure 3.16: Two different shapes with their corresponding trees.
Lemma 3.3.1 Let $C$ be a directed curve and $P = (P_1, P_2, ..., P_n)$, any polygonal chain describing it, with dominant point $P_d$, then $P_s = P_1$, $P_e = P_n$ and $P_d$ define a unique directed triangle with directed base $P_sP_e$, namely the directed triangle $\Delta P_sP_eP_d$.

Proof: Consider a directed curve with start point $P_s$ and end point $P_e$, as shown in Figure 3.17. The b-sequence $B$ that represents the segment in a conventional direction, contains a dominant point $P_d$, which, by the definition of dominant point, is unique.

Let $\Delta$ be the triangle formed by the points $A$, $B$, $C$ with the segment $\overrightarrow{AB}$, with $A = P_s$ and $B = P_e$ as its base and $C = P_d$. Let its altitude $h$ be the distance from $C$ to the segment $\overrightarrow{AB}$.

The directed triangle $\Delta = \Delta ABC$ is clearly unique.
Theorem 3.1 Any directed curve can be represented by a unique hierarchy of directed triangles.

Proof: Consider a dominantly convex directed curve as the shown in figure 3.18(i). According to lemma 3.3.1, the chain \( B = \{P_3, ..., P_d, ..., P_e\} \), can be represented uniquely by \( \Delta \). The dominant point \( P_d \) divides the chain into two independent subchains: \( B_1 = \{P_3, ..., P_d\} \) and \( B_2 = \{P_d, ..., P_e\} \).

The first subchain \( B_1 \), is represented uniquely (according to lemma 3.3.1), by \( \Delta_1 = \Delta ACd \), where \( d = P'_d \), the dominant point for \( B_1 \). \( \Delta_1 \) shares its base with \( l_1 = \overrightarrow{AC} \) of \( \Delta \). If all the points in the first subchain \( B_1 \), are collinear then \( B_1 \) is represented by the degenerate triangle of base \( \overrightarrow{AC} \) and 0-height.

Similarly, the second subchain \( B_2 \), is represented uniquely by \( \Delta_2 = \Delta CB_e \), where \( e = P''_d \), the dominant point for \( B_2 \). \( \Delta_2 \) shares its base with \( l_2 = \overrightarrow{CB} \) of \( \Delta \). If all the points in the second subchain \( B_2 \), are collinear then \( B_2 \) is represented by the degenerate triangle of base \( \overrightarrow{CB} \) and 0-height.

In summary, the sides of \( \Delta \) are the bases of \( \Delta_1 \) and \( \Delta_2 \). Furthermore, \( \Delta_1 \) is encountered before \( \Delta_2 \). We can represent this dependency in a binary tree, where \( \Delta \) has two children: \( \Delta_1 \), the left child (encountered first), and \( \Delta_2 \) the right child. Thus the hierarchy provides the positional information. See figure 3.18(ii).

The process is repeated for each subsegment, finding a dominant point which defines two independent subsegments represented by their unique left and right triangles. The process will end because at each step either the subsequence is collinear, or a dominant point with absolute distance larger to zero is found and the process is applied again to a shorter subsegment, which will eventually will lead to a straight segment.
Figure 3.18: Representation of a curve segment by a unique hierarchy of directed triangles.
(i) The polygonal chain $A = \langle a, ..., i \rangle$ obtained by a uniform sampling.

(ii) The minimal convex polygon enclosing $A$.

(iii) The directed triangles of $A$.

(iv) The CCT($A$).

Figure 3.19: The process of obtaining the CCT of a curve $C$. 

53
3.3.2 Area as an Information Measure

Figure 3.19 presents the complete process of obtaining the CCT of a curve segment $C$ digitized by a uniform sampling. The curve $C$ is sampled to obtain a polygonal chain $A$ that represents $C$ (shown in Figure 3.19(i)). The minimal convex polygon enclosing $C$ is shown in Figure 3.19(ii).

The directed triangles corresponding to the polygonal chain $A$ are shown in Figure 3.19(iii). Note that some triangles are more important than others, i.e. larger triangles are more important for the decomposition than smaller ones. If we eliminate a large triangle we lose more information than eliminating a small triangle so we can say that larger triangles carry more information than smaller triangles.

We can measure the importance of each triangle by the relation of its area to the area of the minimal polygon enclosing $C$. Thus the directed triangle $-\triangle aie$ has more information than directed triangle $-\triangle ced$.

Thus a good approximation of the shape can be obtained by including the larger triangles of the CCT. In general, one could discard the triangles which are very small when compared to the area of convex hull.

The inclusion of the smallest triangles in the approximation will produce a shape which will be closer to the shape boundary. Thus the areas between the approximation and the original curve will give us smaller error areas.

This implies that even though two different polygonal chains $A = (P_1, P_2, ..., P_m)$ and $B = (P'_1, P'_2, ..., P'_m)$ may represent curve a $C$ as shown in Figure 3.6, even with the same error area, the chain which has the error distributed in a large number of small triangles, represents better the curve $C$. 

54
3.3.3 Method

Our new method, consists of the following steps:

(1) Define a base line.

(2) Extract the two-part boundary of the shapes.

(3) Build the Convexity-Concavity Tree.

(1) Define a Base Segment

In the examples given in section 3.2, the curve and the straight line are given. In our case we have to find the straight line that cuts the shape. If we take two points, say, the top-most point (if not unique, take the leftmost of them \( P_{nw} \)) and the bottom most point (if not unique, take the rightmost of them \( P_{se} \)) of the boundary. These two points define a line that splits the shape into two parts (see Figure 3.20).

(2) Extract the Shape Boundary

The boundary of the shape is computed in two parts, by following the path \( B_1 \) along the boundary in counter-clockwise direction (CCW), from point \( P_{se} \) to \( P_{nw} \), and the path \( B_2 \) from point \( P_{nw} \) to \( P_{se} \).

(3) Building the Convexity-Concavity Tree

After computing \( B_1 \) and \( B_2 \), we will segment each to identify the convexities and concavities of the shape. The shape will be formed by two CCTs, one corresponding to each b-sequence. In order to segment a b-sequence apply recursively the following criteria:

(a) If the points are all collinear then no further segmentation is needed.

(b) If the points are not collinear we will continue the segmentation.

We represent the boundary segments by a binary tree, or Convexity-Concavity Tree (CCT), where each node represents a segment and is labeled according to its convexity.
3.3.4 Implementation

Our method is implemented by algorithm 2. Step (1) selects two points on the boundary of the shape \( P_{se} \) and \( P_{nw} \), which define the base line for the decomposition.

Step (2) extracts the two b-sequences for each side of the base line, walking along the boundary in CCW direction. The computation of the b-sequence is done by algorithm 3. Given two boundary points \( P_s = (P_{se} \text{ or } P_{nw}) \) and \( P_e = (P_{se} \text{ or } P_{nw}) \), the b-sequence is obtained by following the boundary always moving to the right hand side neighboring point of \( P_i \).

Step (3) builds the CCT representation for \( B_1 \) and \( B_2 \). The construction of the CCT of a b-sequence \( B \) or \( \text{CCT}(B) \), is performed by algorithm 4:BuildCCT(B).

Algorithm 2: Representing a Shape by its CCT.

Input: Binary Image \( I \) of an object \( S \).

Output: Convexity-Concavity Description \( \text{CCT}(S) = (\text{CCT}_1(S), \text{CCT}_2(S)) \) of the Shape \( S \) in \( I \).

1. Let \( P_{se} = \text{south-most point of the object (break ties by selecting east-most of such points)} \) and let \( P_{nw} = \text{north-most point of the object (break ties by selecting west-most of such points)} \).

2. [Obtain the two-part boundary \( B_1 \) and \( B_2 \) of \( S \) in counter-clock wise direction.]

Compute \( B_1(S) = (P_1, P_2, \ldots, P_m) \), where \( P_1 = P_{se} \) and \( P_m = P_{nw} \) and

Compute \( B_2(S) = (P'_1, P'_2, \ldots, P'_n) \), where \( P'_1 = P_{nw} \) and \( P'_n = P_{se} \)

3. [Build the CCT Representation of \( S \).]

(a) \( \text{CCT}_1(S) = \text{BuildCCT}(B_1(S)) \);

(b) \( \text{CCT}_2(S) = \text{BuildCCT}(B_2(S)) \);

(c) \( \text{CCT}(S) = (\text{CCT}_1(S), \text{CCT}_2(S)) \);

56

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
Algorithm 3: Obtaining a b-sequence B of shape S.

**Input:** Two boundary points of S: start point $P_s = (P_{se}$ or $P_{nw}$) and end point $P_e = (P_{nw}$ or $P_{se}$).

**Output:** The b-sequence $B = (P_1, P_2, ..., P_n)$ where $P_1 = P_s$, and $P_n = P_e$, in CCW direction.

1. **[Initialize.]**
   
   if $P_s = P_{se}$ then dir = E (east) else dir = W (west);
   
   $i = 1$; $P_1 = P_s$;

2. **[Walk along the boundary in counter clockwise direction.]**
   
   while $P_i \neq P_e$ do
   
   if rhs-neighb($P_i$) $\in S$ then $P_{i+1} =$ rhs-neighb($P_i$) and $\text{dir} = \text{newdir(dir, } -90 \circ \text{)}$;
   
   else if front-neighb($P_i$) $\in S$ then $P_{i+1} =$ front-neighb($P_i$) and $\text{dir} = \text{newdir(dir, } 0 \circ \text{)}$;
   
   else if lhs-neighb($P_i$) $\in S$ then $P_{i+1} =$ lhs-neighb($P_i$) and $\text{dir} = \text{newdir(dir, } 90 \circ \text{)}$;
   
   else $P_{i+1} =$ back-neighb($P_i$) and $\text{dir} = \text{newdir(dir, } 180 \circ \text{)}$;

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
Algorithm 4: BuildCCT(B).

**Input:** A Boundary Sequence $B = (P_1, P_2, ..., P_n)$.

**Output:** Convexity-Concavity Tree $CCT(B)$.

1. [Is $B$ too short?]  
   
   **if** $|B| < 3$ **then** return NULL;

2. [Find the Directed Triangle that represents the $b$-sequence $B$.]
   
   (a) [Define the baseline.]
   
   Let $L$ = the directed line from $P_1$ to $P_n$;

   (b) [Calculate distances for points between $P_1$ and $P_n$.]
   
   i. for $2 \leq i \leq n - 1$ do $d_i$ = distance from $P_i$ to $L$;

   ii. Let $\mu = \max \{|d_i|: 2 \leq i \leq n - 1\}$;

   (c) [Determine the Dominant Point.]
   
   Let $P_d$ = the first point $P_i$ such that $|d_i| = \mu$;

   (d) [Define the directed triangle $\Delta$ formed by $P_1, P_n, P_d$ in that order.]
   
   Let $\Delta = +\Delta(P_1 P_n P_d)$ if $d_d > 0$, otherwise $\Delta = -\Delta(P_1 P_n P_d)$;

3. [Are all points in $B$ collinear?]  
   
   **if** $\mu = 0$ **then** return NULL;

4. [Create root-node of $CCT(B)$.]
   
   root-node = Create node for $\Delta$;

5. [Create subtrees of root-node.]
   
   (a) Let $B_1 = (P_1, P_2, ..., P_d)$, similarly let $B_2 = (P_d, P_{d+1}, ..., P_n)$;

   (b) root-node$\rightarrow$leftson = BuildCCT($B_1$);

   (c) root-node$\rightarrow$rightson = BuildCCT($B_2$);

6. return root-node;
Theorem 3.2 BuildCCT(B) takes $O(n^2)$ steps to construct the CCT of a b-sequence $B = \langle P_1, P_2, ..., P_n \rangle$.

Proof: In algorithm BuildCCT(B): Step (1) takes $O(1)$ steps.

In step (2), steps (c) and (d) take $O(n)$ each, thus step (2) takes $O(n)$.

Step (3) takes $O(1)$ steps. Step (4) takes $O(1)$ steps.

Step (5) fragments the b-sequence $B[1...n]$ into two sub-sequences $B_1[1...d]$ and $B_2[d...n]$.

Thus this recursion generates in the worst case:

$$T(n) = c'n + T(\text{BuildCCT}(B_1)) + T(\text{BuildCCT}(B_2))$$

Assuming our proposition holds

$$\text{BuildCCT}(B_1) \leq cd^2$$

$$\text{BuildCCT}(B_2) \leq c(n - d + 1)^2$$

$$T(n) \leq cn^2$$

$$cd^2 + c(n - d + 1)^2 + c'n \leq cn^2$$

The worst case performance is for a b-sequence in which there is no co-linear subsequence of $B$ of length larger than two. Therefore in the worst case $d=2$:

$$c(4) + c(n - 1)^2 + c'n \leq cn^2$$

$$cn^2 - 2nc + 5c + c'n \leq cn^2$$

$$(c'-2c)n + 5c \leq 0$$

$$(2c-c')n \geq 5c$$

$$2cn \geq 5c$$

$$n \geq 5/2$$

Thus the proposition holds.
Figure 3.20: Finding the largest triangles of the digital shape.
3.4 Results

This section shows the application of our representation method to a shape commonly found in nature: a maple leaf.

3.4.1 Representation of a Maple Leaf

We chose a natural shape to test our representation method for two reasons:

- natural shapes are especially difficult to characterize.
- humans can easily recognize them.

Consider the silver maple leaf shown in Figure 3.22, which we obtained from digitizing the shape of a natural maple leaf. We applied our algorithm to obtain its CCT, which we do not show because of its size.

3.4.2 Incremental Reconstruction

The incremental reconstruction from of the maple leaf in Figure 3.22 is shown in Figure 3.23. The shape shown in Figure 3.23(i) is obtained by the inclusion of the larger triangles in the
CCT. The shape shown in Figure 3.23(ii) is obtained by the addition of smaller triangles in the CCT. We see some additional convexities and concavities added, however the shape still looks artificial. The shape shown in Figure 3.23(iii) includes even smaller triangles. The threshold area used was the triangle shown on upper part the left side of the contour. The reconstruction at this point looks already as a maple leaf, even though there are some fine details missing. Finally the shape shown in Figure 3.23(iv) shows the full reconstruction of the shape.

Consider now Figure 3.24 where the same maple leaf was used but this time rotated to the right by thirty degrees. Once rotated, it was decomposed and represented by its CCT. The result of the reconstruction from the CCT, using the same threshold used in Figure 3.23(iii) is shown in Figure 3.25.
If we compare the two partial reconstructions, we can easily check that the main characteristics of the shape are present in both of them. We can barely observe small differences, produced by the variation of the shape boundary as a product of rotation.

These results show that:

- The CCT representation of a shape preserves the information.
- Larger triangles contain more information of the shape.

3.4.3 Evaluation of the Shape Representation Method

Some criteria have been proposed for the evaluation of a shape representation method by Mokhtarian [37]. We list those criteria:

- **Invariance**: if two shapes have the same shape they should also have the same representation.

- **Uniqueness**: if two shapes do not have the same shape, they should have different representation.

- **Stability**: if two shapes have a small difference their representation should also have a small difference and if two representations have a small difference they should also have a small shape difference.

They also suggest some additional computational properties:

- **Efficiency**.

- **Ease of implementation**.

- **Computation of Shape Properties**.
Figure 3.23: Incremental reconstruction of a silver maple leaf.
Figure 3.24: Silver maple leaf rotated by 30 degrees.

Figure 3.25: Silver maple partial reconstruction.
3.4.4 Evaluation of the CCT Shape Representation

The evaluation of the CCT representation using the criteria listed before:

- **Invariance**: the CCT of a shape is invariant to translation but sensitive to rotation. Regarding to scale, the structure of the CCT is invariant to scale, although the dimensions of the triangles change.

- **Uniqueness**: according to theorem 3.1 every shape is represented by a unique hierarchy of directed triangles.

- **Stability**: any variation of shape will be reflected on its boundary. The variation of the boundary can have two effects:
  
  - change the hierarchy of triangles and thus change the structure of the CCT.
  
  - maintain the structure of the CCT but change the size of a (some) triangles.

Thus if a shape variation is small, it will create or modify a small convexity and thus the structure of the CCT will change slightly or it will maintain the CCT structure but one or more of the triangles will have a small variation in its/their dimensions.

Similarly a large variation (of the boundary) will create or modify a large convexity, so it will reflect in a large change of the CCT structure or the dimensions of its directed triangles. Thus our description method is stable.

Additionally, our method provides the following advantages:

- **Economy**: If we prune small triangles of the CCT, keeping the larger triangles, we eliminate details of the boundary but we keep the essential information of the shape, thus resulting in an economical representation.
• Noise

The presence of noise of small amplitude (low or high frequency) in the boundary of the shape will not have important effect on the larger and more important triangles of the shape, which contain more shape information.

However the noise will be reflected in the smaller triangles, which we can prune of the CCT, easily, since they are in the bottom of the CCT. Thus the CCT representation is robust in the presence of noise.

• Information Preserving

The representation of shape by a CCT is information preserving, since we can exactly reconstruct the original shape.

• Ease of implementation

Obtaining the CCT representation is much easier than obtaining a closed equation or approximation of the shape, specially of complex shapes.

3.5 Summary

In this chapter we introduced a new structural method for shape representation that uses the information contained in the boundary of the object.

The boundary of an object is approximated by a polygon which represents the shape. A shape can be described by many polygons. The quality of the approximation of the shape by a polygon is related to the area between the boundary of the shape and the contour of the shape. As we include more points of the boundary in the polygon, this area becomes smaller thus the description is better. The polygon that approximates the boundary of the shape is represented by a sequence of its vertices or b-sequence, traversing the polygon in
counter clock wise direction. The b-sequence of the shape can be obtained from digitizing the shape.

The polygon is decomposed into a hierarchy of positive and negative triangles defined only by the points along its boundary. A positive triangle represents a triangular region which is present in the shape, whereas a negative triangle represents a triangular region which is not in the shape.

The hierarchy of triangles obtained from a shape $S$ is represented in a binary tree that we call a Convexity-Concavity Tree of $S$ or simply CCT($S$). The original shape can be fully reconstructed from its CCT description by adding positive triangles and subtracting negative triangles from the approximation, while traversing the CCT in a top-down fashion. The final reconstruction of the shape is the result of the addition of all the triangles in the CCT.

In order to identify the points that define the directed triangles, we introduced the concept of dominant point. The dominant point ($P_d$) of a b-sequence is the first point with maximal distance to the base line defined by the first and last point of the sequence. If the dominant point is located to the right-hand side of such a line, then the b-sequence is represented by a positive triangle. If, on the other hand, the dominant point is located to the left-hand side of that line, then the b-sequence is represented by a negative triangle. In case $P_d$ is on the base line, then the sequence is straight and it is represented by a degenerate triangle of 0-height.

The advantages of the CCT representation are:

- A unique CCT($S$) is obtained to represent the b-sequence of an object, by defining a conventional direction to traverse its boundary and a base line.
• A shape can be fully reconstructed from its CCT, thus the CCT is information preserving.

• The CCT representation of the shape can be obtained in polynomial time \(O(n^2)\), where \(n\) is the number of points in the b-sequence.

• The CCT representation can describe very complex shapes which are difficult to characterize by a closed equation.

We applied the CCT representation for the description of a complex shape such as a maple leaf and we found that the CCT representation has the characteristics of a good representation method, according to the criteria proposed in the literature of shape analysis.
Chapter 4
Shape Matching

Objective

The objective of this chapter is to show the use of the CCT representation of shape for matching and recognition. We will show how to match two shapes, $S_1$ and $S_2$ using their CCT representation. Both CCTs will be related by another binary tree, called Matching Tree of $S_1$ and $S_2$.

Overview

Section 4.1 presents the difficulties of matching and recognizing shape. Section 4.2 shows the limitations of the Euclidean concept of similarity of two triangles, then we suggest new concepts of similarity between two triangles, as an improvement to the Euclidean concept, and then we define some metrics to measure the similarity of two triangles. Section 4.3 presents a method to match two shapes using their CCT representation and its application to a sample problem. Section 4.4 presents the results of applying the method to a real world problem and section 4.5 summarizes and concludes the chapter.

4.1 Introduction: Shape Matching and Shape Recognition

Shape matching and shape recognition are central problems of pattern recognition. Shape matching is the process of binary answering yes or no to the identity of an unknown shape as an instance of a known shape (pattern). Shape recognition is the process of identifying
the degree of similarity (*shape similarity*) between the unknown and the known shapes.

The difficulty of measuring shape similarity is reflected in the lack of universal criteria to measure it, even though the subject has been studied for quite some time.

There is a theoretical controversy regarding the nature of shape similarity [55]. There are two main points of view. One point of view, the non-dimensional set theory, establishes that attributes should be considered to be groups of components of the shape, which constitute a larger non-dimensional set and that these attributes may not be reflected by any metric. Thus shapes that are perceived as similar will have the same set of attributes and shapes that are perceived as dissimilar will have different attribute sets [44, 52].

The other point of view is that similarity can be reflected by a distance-like metric, where shapes that are perceived as similar are reflected by a small difference in the value on the metric whereas shapes that are perceived as dissimilar have a large difference in the value on the metric. This point of view was championed by Attneave and others [4, 51, 49].

Shape matching and shape recognition are closely related to shape representation thus face the same problems that representing shape does: the adequate representation where small or large variations with respect to the pattern are reflected proportionally in the representation.

We propose some criteria based on our shape representation. We consider two shapes as similar if: the two shapes contain the same structural elements. Then the similarity of each structural component is measured. The shape similarity between two shapes will be the sum of the similarities for the match of all their structural components.

The criteria for matching shape are by no way universal and depend heavily on the particular application, or even particular objectives of the matching, i.e. the shape of an
airplane and the shape of a plane may have quite different attributes thus a feature based matching may be useful to discriminate between them, however in order to discriminate between different types of airplanes, a more precise matching may be necessary.

Moreover, some applications consider shapes that contain common structural elements (flexible shape) as similar, i.e. the identification of hand-written characters.

We do not intend to give an absolute criterion to recognize shape rather we want to provide with a flexible scheme that allows the implementation of shape matching (flexible or exact).

Since our shape representation method is based on the structural decomposition of the shape into triangles, our matching method is based on matching triangles.

4.2 Matching Two Triangles

Two triangles are said to have the same shape, i.e. they are both triangles. However they may not be the same triangle thus we need to identify a particular triangle. In Euclidean geometry the concept of similarity (equality) of triangles can be based on the angles, on the angles and sides or on their area (their size).

The use of the angles to define the similarity of two triangles has the inconvenience of identifying two triangles with equal angles but different size as the same triangle. If the sides of the triangle are used, it will be difficult to identify triangles with small variations on the lengths of the sides, since one never knows which side had the variation, i.e. thus we do no know which side is the base.

Thus in order to help to identify the correct the orientation of the triangles, we introduce the concept of directed triangle. Using this concept will help us to define different types of similarity.
4.2.1 The Similarity of Two Directed Triangles

Using the concept of a directed triangle, we define some possible concepts of similarity which may help to identify similar directed triangles with different precision. The simplest similarity of two directed triangles is their sign. This similarity may be used in applications the structural elements of the shape are the same and their size is not important.

**Definition 4.1 \(\sigma_C\)-similarity**

Two directed triangles \(\triangle_1(b_1, h_1)\) and \(\triangle_2(b_2, h_2)\), are called \(\sigma_C\)-similar if they are both positive or both negative (i.e. they both represent a dominantly convex or concave segment).

A more restricted similarity of two directed triangles is defined by taking their size (area) into account. This similarity may be used if additionally to the sign of the triangles, their size is also important.

**Definition 4.2 \(\sigma_A\)-similarity**

Two directed triangles \(\triangle_1(b_1, h_1)\) and \(\triangle_2(b_2, h_2)\) are \(\sigma_A\)-similar if \(\triangle_1\) and \(\triangle_2\) are \(\sigma_C\)-similar and \(b_1h_1 = b_2h_2\) (i.e. they have the same area).

Yet a more restrictive similarity can be defined by \(\sigma_{bh}\)-similarity which defines two triangles as similar if they have the same sign (\(\sigma_C\)-similar), the same size (area) (\(\sigma_A\)-similar) and their directed base and altitude are the same.

**Definition 4.3 \(\sigma_{bh}\)-similarity**

Two directed triangles \(\triangle_1(b_1, h_1)\) and \(\triangle_2(b_2, h_2)\), are \(\sigma_{bh}\)-similar if they are \(\sigma_A\)-similar and \(b_1 = b_2, h_1 = h_2\).

The most restrictive similarity is obtained by defining \(\sigma_e\)-similarity, which define two triangles to be similar if they have the same sign (\(\sigma_C\)-similar), the same size (area) (\(\sigma_A\)-
similar), have the same base and altitude \((\sigma_{bh}\)-similar) and additionally they have the same sides.

**Definition 4.4 \(\sigma_e\)-similarity**

Two directed triangles \(\triangle_1(b_1, h_1) = \triangle ABC\) and \(\triangle_2(b_2, h_2) = \triangle A'B'C'\), respectively, are \(\sigma_e\)-similar if \(\triangle_1\) and \(\triangle_2\) are equivalent, i.e. \(AB = A'B', AC = A'C', \) and \(CB = C'B'\) (and thus \(\sigma_e\)-similar in particular).

### 4.2.2 Measuring the Similarity of Two Directed Triangles

As presented in chapter 3, a directed curve is represented by a hierarchy of directed triangles. A dominantly convex curve is represented by a *positive* directed triangle of base \(b\) and altitude \(h\) and a dominantly concave curve is represented by a *negative* directed triangle of base \(b\) and altitude \(-h\).

Using the concept of \(\sigma_{bh}\)-similarity, we can define the following relation:

\[ \triangle_1 \sim \triangle_2 \iff \triangle_1, \triangle_2 \text{ have the same base and altitude} \]

The relation \(\sim\) is an equivalence relation, thus induces a partition of the set of directed triangles into families (the equivalence classes of \(\sim\)). We can establish a one to one relation to the points in the first and fourth quadrant of the plane, by identifying the classes of positive directed triangles with the points in the first quadrant by the association:

\[ +\triangle(b, h) \leftrightarrow (b, h) \]

Similarly we identify the classes of negative directed triangles with the points in the fourth quadrant by the association:

\[ -\triangle(b, h) \leftrightarrow (b, -h) \]

In Figure 4.2 each point in the plane represents a family of directed triangles.
Figure 4.1: Six possible directed versions of an ordinary triangle formed by points A, B, C.
Figure 4.2: The bh-representation of the directed triangles in Figure 4.1.
4.2.3 Defining a Metric for the Similarity of Two Directed Triangles

Based on the notions of similarity for directed triangles just introduced, we aim to define the metrics that will measure such notions. These metrics will be used later to measure the similarity of two general shapes.

In order for a measure to be considered a metric it must have three properties.

- \( \text{dist}(S, S') = 0 \) if and only if \( S = S' \)
- \( \text{dist}(S, S') = \text{dist}(S', S) \)
- \( \text{dist}(S, S'') \leq \text{dist}(S, S') + \text{dist}(S', S'') \)

We will propose three measures \( M_C, M_{bh}, M_A \) that can be used as metric on the set of all directed triangles.

Given two directed triangles, \( \triangle_1(b_1, h_1) \) and \( \triangle_2(b_2, h_2) \) we define the following metrics to determine the mismatch of two directed triangles.

- **\( M_C \) metric:**
  \[
  M_C(\triangle_1, \triangle_2) = \begin{cases} 
  0 & \text{if } \triangle_1 \text{ and } \triangle_2 \text{ have the same sign} \\
  1 & \text{otherwise}
  \end{cases}
  \] (4.1)

- **\( M_{bh} \) metric:**
  \[
  M_{bh}(\triangle_1, \triangle_2) = \sqrt{(b_1 - b_2)^2 + (h_1 - h_2)^2}
  \] (4.2)

- **\( M_A \) metric:**
  \[
  M_A(\triangle_1, \triangle_2) = \begin{cases} 
  \Omega(\triangle_1, \triangle_2) & \text{if } \triangle_1 \text{ and } \triangle_2 \text{ have the same sign} \\
  |A(\triangle_1) + A(\triangle_2)| & \text{otherwise}
  \end{cases}
  \] (4.3)

where \( \Omega(\triangle_1, \triangle_2) \) is the area of the symmetric difference of the maximum overlap of \( \triangle_1 \) and \( \triangle_2 \), aligned by their directed bases and \( A(\triangle) = b_i h_i / 2 \)
Given two shapes $S_1$ and $S_2$, let $t(S_1, S_2)$ denote a translation (or rotation) of $S_1$ that gives a maximum area overlap with $S_2$.

Clearly $t(S_2, S_1)$ is imply the opposite translation applied to $S_2$ that produces exactly the same overlap.

$$t(S_1, S_2) \cap S_2 = t(S_2, S_1) \cap S_1$$

There can be different $t(S_1, S_2)$ which give the maximum area intersection, although the actual area intersection is not the same. Let $S_1 \odot S_2$ denote any of such maximum intersection and let $|S|$ denote the area of $S$.

Define $M(S_1, S_2)$ as the area of the symmetric difference of the overlap of $S_1$ and $S_2$ as the result of $t(S_1, S_2)$.

$$M(S_1, S_2) = |S_1| + |S_2| - |S_1 \odot S_2|$$

**Theorem 3.3** $M_A(S_1, S_2)$ is a metric:

**Proof:** We need to proof that $M_A$ meets the three properties of metric.

- $M_A(S_1, S_1) = 0$ clearly ($|S_1| + |S_1| - 2|S_1 \odot S_1| = 0$)

- $M_A(S_1, S_2) = M_A(S_2, S_1)$ clearly

  $$\left(|S_1| + |S_2| - |S_1 \odot S_2|\right) = \left(|S_2| + |S_1| - |S_2 \odot S_1|\right)$$

- $M(S_1, S_2) + M(S_2, S_3) - M(S_1, S_3) \geq 0$

  $$\left(|S_1| + |S_2| - |S_1 \odot S_2|\right) + \left(|S_2| + |S_3| - |S_2 \odot S_3|\right) - \left(|S_1| + |S_3| - |S_1 \odot S_3|\right) \geq 0$$

  $$2(|S_2| - |S_1 \odot S_2| - |S_2 \odot S_3| + |S_1 \odot S_3|) = 2(|S_2| - |S_1 \cap S_2| + |S_2 \cap S_3| + |S_1 \cap S_3|)$$

78
\[ 2(|S_2| - |S'_1 \cap S_2| + |S'_2 \cap S_3| + |S_1 \odot S_3|) \geq 2(|S_2| - |S'_1 \cap S_2| - |S'_2 \cap S_3| + |S'_1 \cap S'_2|) \]
\[ 2(|S_2| - |S'_1 \cap S_2| + |S'_2 \cap S_3| + |S_1 \odot S_3|) \geq 2(|S_2| - (|S'_1 \cap S_2| + |S'_2 \cap S_3| - |S'_1 \cap S'_2|)) \]
\[ 2(|S_2| - |S'_1 \cap S_2| + |S'_2 \cap S_3| + |S_1 \odot S_3|) \geq 2(|S_2| - (|S'_1 \cup S'_3| \cap S_2|)) \]

*thus our proposition holds.*

### 4.2.4 Which Metric is Better?

Each of the three metrics that we propose has a different purpose.

The metric $M_C$ measures the difference in sign between the two triangles that are matched. The metric $M_{bh}$ is a more precise metric than $M_C$ since it accounts not only for the sign but also for the size of the directed triangles. Yet the metric $M_C$ can not discriminate between two different directed triangles with the same base and altitude. The metric $M_A$ is the most precise since it measures the difference of oriented triangles which are similar in orientation and area.

There is also a cost of computation involved for each metric. The more precise is the metric, the larger its computational cost. The choice of which metric to use is going to be influenced by the particular purpose of the match.

### 4.3 Matching Two Shapes

The complexity of measuring shape similarity for shape matching and shape recognition were mentioned section 4.1. There are other issues involving the recognition of shapes. One of the fundamental questions of shape perception and recognition is the following: is a shape recognized because of its *local features* or its *global properties*?
We do not intend by any means to answer such a question, since it is beyond the scope of this work. However, we show that using our shape representation method, we can implement an automatic recognition of shape, either by identifying key components of it (local match) or by identifying a global property of the shape (global match). The local match is performed by the identification of local features of the shape in the CCT. The global match will be performed by defining a global metric for the match.

Another important issue about shape matching is the identification of different kinds of matching: sometimes the match requires the only the identification of features of the shape, i.e. its convexity or concavity, others it may be required an exact measure of these features, i.e. the sizes of the convexity or concavity.

4.3.1 Exact and Elastic Matching

We consider the process of shape matching as the process of identifying the structural components contained in the shape. In order to understand better this process, we consider convenient to identify two types of shape matching: exact (or rigid) and elastic.

The exact matching of shape exists when we identify exactly the particular components of the shape i.e. the shape of a particular type of airplane, say an F-16. On the other hand, an elastic matching of shape exists when we identify a hand-written character, say the letter “a”. In the first case, the dimensions of each of the identified structural elements of the shape have explicit dimensions. In the second case the shape contains the required structural features, however their dimensions can vary.

Consider the shapes shown in Figure 4.3. The two shapes contain similar components (convexities and concavities), a fact that is reflected in the structure of their CCT. Thus a flexible match will identify them as similar. However an exact match will discriminate
any of the shapes because the components are not the same. We define both types more formally.

**Definition 4.5 Elastic Shape Match** Given two shapes, $S_1$ and $S_2$, represented by $CCT(S_1)$ and $CCT(S_2)$, we say that $S_1$ and $S_2$ make an elastic match if all their pair-wise triangles of $CCT(S_1)$ and $CCT(S_2)$, are $\sigma_C$-similar.

**Definition 4.6 Exact Shape Match** Given two digital shapes, $S_1$ and $S_2$, represented by $CCT(S_1)$ and $CCT(S_2)$, respectively, we say that $S_1$ and $S_2$ make an exact match if the all their pair-wise triangles of $CCT(S_1)$ and $CCT(S_2)$, are $\sigma_e$-similar.

### 4.3.2 The Matching Tree

The definition of a metric such as $M_A$ is not enough to apply it to the shape recognition problem. Consider Figure 4.4 where three simple shapes $S_1$ (square), $S_2$ (triangle) and $S_3$ (rectangle) are presented. The measure $M_A(S_1, S_2) = M_A(S_1, S_3)$, however the shapes $S_1$ and $S_3$ are more similar than $S_1$ and $S_2$. This tells us that even though the metric $M_A$ is nicely defined, it does not reflect the shape information since we expect the square and the rectangle to be somehow more similar than the square and the triangle.

We will show that the match using the metric $M_A$ to the components of the two shapes gives better results. We first identify the structural components of every shape (by obtaining the CCT) and then apply the metric to measure the match of every pair of components.

In order to describe the global match of two shapes, we define a structure which describes the corresponding components to be matched from every shape descriptor: the matching tree.
Figure 4.3: The $\sigma_C$-similarity of Shape 1 and Shape 2.
(i) Tree shapes to be matched.

\begin{align*}
M_A(S_1, S_2) & \quad M_A(S_1, S_3) \\
(ii) M_A(S_1, S_2) = M_A(S_1, S_3).
\end{align*}

Figure 4.4: The definition of the metric $M_A$ is not enough for shape matching.
Definition 4.7 Matching Tree

The Matching Tree (MT) of two shapes, is a binary tree, which represents the matching of two Convexity Trees, CCT(S₁) and CCT(S₂). The MT contains three sets of nodes:

Set 1: The set of nodes present in CCT(S₁) and also present in CCT(S₂).
Set 2: The set of nodes present in CCT(S₁) but not present in CCT(S₂).
Set 3: The set of nodes present in CCT(S₂) but not present in CCT(S₁).

The process of matching involves three sets, according with Tversky's set model [55]. The matching of two shapes by convexities also requires a structure to find/locate the mismatches of two shapes. The matching of two shapes S₁ and S₂, is performed by matching their CCT’s, triangle by triangle (node by node) from the top-down.

The MT represents the match of the two CCT’s. Every node i of the matching tree represents a pair to be matched: (Δ₁i:Δ₂i), where Δ₁i is the triangle of CCT(S₁i) and Δ₂i is the triangle of CCT(S₂).

If a node of CCT(S₁) or CCT(S₂), does not have a triangle to match, then it is labeled φ, otherwise it is labeled with the name of that node. The nodes in the matching tree in which label₁=Δ₁ and label₂=φ form the set of nodes in CCT(S₁) but not in CCT(S₂). The nodes in the matching tree in which label₁=φ and label₂=Δ₂ form the set of nodes in CCT(S₂) but not in CCT(S₁). The nodes in the matching tree in which label₁=Δ₁ and label₂=Δ₂ form the set of nodes in CCT(S₂) and also present in CCT(S₁). Figure 4.5 shows two arbitrary CCTs, CCT(S₁) and CCT(S₂) and the matching tree MT(CCT(1), CCT(2)).

4.3.3 Measuring the Mismatch of Two Shapes

The recognition of a particular shape may involve the identification of a particular characteristic of the shape that may be located in an specific location of the boundary or global
(i) \( \text{CCT}(S_1) \).

(ii) \( \text{CCT}(S_2) \).

(iii) Matching Tree for \( \text{CCT}(S_1) \) and \( \text{CCT}(S_2) \).

Figure 4.5: Matching two CCT's.
property of the shape. Local characteristics have an explicit location on the matching tree and global characteristics are distributed all over the matching tree.

The identification of local characteristics requires to measure the match in particular nodes of the matching tree, that is, a subset of the matching tree and the identification of global characteristics requires to measure the match along the whole tree. We will show how to measure the match for global characteristics.

In order to measure the mismatch between two shapes, say A and B, we measure the pair-wise mismatch between the nodes of their CCTs, proceeding in a top-down fashion.

After we measure the local match for all the nodes in the matching tree, we can measure the global match of the tree.

The following theorem establishes that it is enough to define a metric for the nodes of the tree, to obtain a global metric to measure the match of two shapes.

Given two shapes $S_1$ and $S_2$ represented by CCT($S_1$) and CCT($S_2$) respectively. Let $M(\Delta_1, \Delta_2)$ denote the metric that measures the local match between $\Delta_1$ and $\Delta_2$ for every node in the Matching Tree of $S_1$ and $S_2$.

Define $M_T(S_1, S_2) = \sum_{i=1}^{n} M(\Delta_{1i}, \Delta_{2i})$ as the total measure of the match between $S_1$ and $S_2$, where $n=\text{number of nodes in } MT(S_1, S_2)$.

**Theorem 4.1** If $M(\Delta_1, \Delta_2)$ is a metric then $M_T(S_1, S_2)$ is also a metric.

**Proof:** We need to prove that $M_T$ satisfies the three properties of a metric.

- $M_T(S_1, S_1) = 0$ clearly $M(\Delta_{1i}, \Delta_{1i}) = 0$ for all pairs $(\Delta_{1i}, \Delta_{1i})$, by the definition of $M$ as a metric, thus $M_T(S_1, S_1) = \sum_{i=1}^{n} M(\Delta_{1i}, \Delta_{1i}) = \sum_{i=1}^{n}(0) = 0$
- $M_T(S_1, S_2) = M_T(S_2, S_1)$
clearly $M(\Delta_1, \Delta_2) = M(\Delta_2, \Delta_1)$ by the definition of $M$ as a metric, thus $M_T(S_1, S_2) = \sum_{i=1}^{n} M(\Delta_{1i}, \Delta_{2i}) = \sum_{i=1}^{n} M(\Delta_{2i}, \Delta_{1i}) = M_T(S_2, S_1)$

- $M(S_1, S_3) \leq M(S_1, S_2) + M(S_2, S_3)$ By induction on the number of nodes:

(Basis) $M(\Delta_{1i}, \Delta_{3i}) \leq M(\Delta_{1i}, \Delta_{2i}) + M(\Delta_{2i}, \Delta_{3i})$

(Induction) $\sum_{i=1}^{n} M(\Delta_{1i}, \Delta_{3i}) \leq \sum_{i=1}^{n} M(\Delta_{1i}, \Delta_{2i}) + \sum_{i=1}^{n} M(\Delta_{2i}, \Delta_{3i})$

$\sum_{i=1}^{n} M(\Delta_{1i}, \Delta_{3i}) + M(\Delta_{1i+1}, \Delta_{3i+1}) \leq \sum_{i=1}^{n} M(\Delta_{1i}, \Delta_{2i}) + \sum_{i=1}^{n} M(\Delta_{2i}, \Delta_{3i}) + M(\Delta_{1i+1}, \Delta_{2i+1}) + M(\Delta_{2i+1}, \Delta_{3i+1})$

$\sum_{i=1}^{n+1} M(\Delta_{1i}, \Delta_{3i}) \leq \sum_{i=1}^{n} M(\Delta_{1i}, \Delta_{2i}) + \sum_{i=1}^{n+1} M(\Delta_{2i}, \Delta_{3i})$

$q.e.d.$

Consider the three shapes presented in Figure 4.4(i), $S_1$, $S_2$ and $S_3$. Furthermore consider $|S_2| = |S_3| = 1/2|S_1| = s$.

Then the match of the shapes, without using the CCT representation and the matching tree gives:

$M_A(S_1, S_2) = |S_1| + |S_2| - 2|S_1 \cap S_2| = 1/2s.$

Also $M_A(S_1, S_3) = |S_1| + |S_3| - 2|S_1 \cap S_3| = 1/2s.$

Thus the metric $M_A$ gives the same values for both pairs. This is shown in Figure 4.4(ii).

Now consider the alternative using the CCT representation of each shape to perform the match. The CCT's of the three shapes and the Matching Trees are shown in Figure 4.6.

The metric $M_A$ is applied to every pair in the Matching Tree. We obtain:

$M_A(S_1, S_2) \geq 1/2s$

$M_A(S_1, S_3) = 1/2s$

and $M_A(S_2, S_3) \leq 1/2s$
(ii) $CCT(S_1)$.  (iii) $CCT(S_2)$.  (iv) $CCT(S_3)$.  

(vii) Matching Tree for $S_1$ and $S_2$.  (viii) Matching Tree for $S_1$ and $S_3$.  

Figure 4.6: The Matching Trees obtained for the match of shapes $(S_1, S_2)$ and $(S_1, S_3)$.  

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
Thus $M_A(S_1, S_2) > M_A(S_1, S_3)$, which tells us that $S_1$ and $S_3$ are more similar than $S_1$ and $S_2$.

This last result makes more sense for our purposes.

Observe that the shape similarity is a *global* concept. We consider that two shapes are *similar* if:

- they contain the same structural components.
- the *total distance* produced by the metric that measures the match is within a defined threshold.

In order to decide if a match between the shapes exists, we can define a threshold, which could be defined from previous analysis of the pattern to match.

If the global mismatch of the matching tree is within this threshold, then the shapes match, otherwise they do not. The choice of this threshold as well as the metric used is application dependent.

Next, we point out some properties of obtained by the use of the metrics $M_{bh}$, when they are used for the matching of two shapes.

### 4.3.4 Properties of the $M_{bh}$ Metric

We present some properties of the $M_{bh}$ metric:

**Theorem 4.2** Given two shapes $S_1$ and $S_2$ and their scaled versions $S'_1 = c * S_1$ and $S'_2 = c * S_2$, for some $c > 0$, the following holds: $M_{bh}(S'_1, S'_2) = c * M_{bh}(S_1, S_2)$

**Proof:** From $M_{bh}(S_1, S_2) = \sum_{i=1}^{N} M_{bh}(\Delta_{1i}, \Delta_{2i})$ and $M_{bh}(S'_1, S'_2) = \sum_{i=1}^{N} M_{bh}(\Delta'_{1i}, \Delta'_{2i})$

$\Delta'_{1i} = c * \Delta_{1i} = (c * b_{1i}, c * h_{1i})$ and similarly, $\Delta'_{2i} = c * \Delta_{2i} = (c * b_{2i}, c * h_{2i})$

$M_{bh}(\Delta'_{1i}, \Delta'_{2i}) = \sqrt{(c * b_{1i} - c * b_{2i})^2 + (c * h_{1i} - c * h_{2i})^2} = c * M_{bh}(\Delta_{1i}, \Delta_{2i})$
Figure 4.7: Computing $M_T(S_1, S_2)$ from the Matching Tree.
Figure 4.8: Computing $M_T(S_1, S_3)$ from the Matching Tree.
Thus $M_{bh}(S_1', S_2') = \sum_{i=1}^{N} M_{bh}(\Delta_{1i}', \Delta_{2i}') = c \times \sum_{i=1}^{N} M_{bh}(\Delta_{1i}, \Delta_{2i}) = c \times M_{bh}(S_1, S_2)$

**Theorem 4.3** Assume a shape $S$, represented by $CCT(S)$. Consider the reconstruction of $S$ from $CCT(S)$, denoted $Rec(S)$. The following property holds:

$Rec(S) \times c = Rec(S \times c)$ for each $c > 0$

**Proof:**

Since $Rec(S) = \Delta_1 + \Delta_2 + ... + \Delta_N$ and $Rec(S \times c) = \Delta_1' + \Delta_2' + ... + \Delta_N'$

where $\Delta_i' = c \times \Delta_i = (c \times b_i, c \times h_i)$, $1 \leq i \leq N$, it follows $Rec(S \times c) = c \times Rec(S)$

Thus our proposition holds.

In order to illustrate the use of the metrics proposed to measure of shape similarity, consider the following example.

Given the shapes 1-5 shown in Figures 3.9- 3.13, we want to determine the shape similarity among them, so that we can identify the closest match for every shape. Also we would like to identify the closest match among all the shapes.

Note that even when we can observe certain shape similarity between some of the shapes, it is not straightforward to decide which is the closest match, neither to give a number which reflects the match and justify the decision.

Take for example shape 1, whose right-hand boundary contains two concavities and its left-hand side one concavity in the upper part and a straight segment in the lower part. The $CCT(S_1)$ contains 5 nodes or features.

If we compare with the remaining shapes we find the following: shape 2 contains exactly the same nodes that shape 5 has (perhaps that is why they are perceived as similar), however they are not exactly the same features (perhaps that is why they are perceived as different).
Now shape 3 has different nodes, and perhaps that is why they are perceived as different.

Next, shape 4 has 7 nodes, 5 of them equal to the nodes of shape 1, however the additional 2 nodes introduce some difference with shape 1.

Finally, shape 5 has also 7 nodes, as shape 4 does, 5 of them are equal to the nodes of shape 1, and as shape 4 the additional 2 nodes (in a different position) also introduce some difference with shape 1.

Similar situations will arise while comparing each shape to the remaining ones. Next we apply the metrics defined in this chapter to help the decision. The details of the match between shape 1 and shape 2 - shape 5 are presented at the end of the chapter.

Tables 1 and 2 show the total value of the metrics $M_{bh}$ and $M_A$, respectively for all the shapes.

Looking at these results the decisions seem easier now:

From the table we can see that the best match for shape 1 is shape 5, it has the smallest value of $M_{bh}$ and $M_A$ in the row corresponding to shape 1.

Note that the best match for shape 2 is shape 1, according to $M_{bh}$ however, according to $M_A$ shape 5 is the best match for shape 2. Since $M_A$ is a more precise metric we decide in favor of the last one.

The best match for shape 3 is shape 2 using both metrics. The best match for shape 4 is shape 1 according to both metrics and finally the best match for shape 5 is shape 1. The closest match among all the shapes is between shape 1 and shape 5, according to both metrics.

Note that the results of both metrics are not always the same (i.e. the best match for shape 2 is shape 1 using $M_{bh}$ and shape 5 using $M_A$) and that results that have no difference
### Table 1: $M_{bh}$ measure

<table>
<thead>
<tr>
<th>$M_{bh}$</th>
<th>Shape 1</th>
<th>Shape 2</th>
<th>Shape 3</th>
<th>Shape 4</th>
<th>Shape 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shape 1</td>
<td>0</td>
<td>4.1</td>
<td>9.1</td>
<td>1.9</td>
<td>0.9</td>
</tr>
<tr>
<td>Shape 2</td>
<td>0</td>
<td>10.5</td>
<td>6.3</td>
<td>5.0</td>
<td></td>
</tr>
<tr>
<td>Shape 3</td>
<td>0</td>
<td></td>
<td>11.5</td>
<td>10.5</td>
<td></td>
</tr>
<tr>
<td>Shape 4</td>
<td>0</td>
<td></td>
<td></td>
<td>2.7</td>
<td></td>
</tr>
<tr>
<td>Shape 5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>symmetric</td>
</tr>
</tbody>
</table>

### Table 2: $M_A$ measure

<table>
<thead>
<tr>
<th>$M_A$</th>
<th>Shape 1</th>
<th>Shape 2</th>
<th>Shape 3</th>
<th>Shape 4</th>
<th>Shape 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shape 1</td>
<td>0</td>
<td>13.0</td>
<td>21.4</td>
<td>1.8</td>
<td>1.3</td>
</tr>
<tr>
<td>Shape 2</td>
<td>0</td>
<td>22.9</td>
<td>15.4</td>
<td>11.9</td>
<td></td>
</tr>
<tr>
<td>Shape 3</td>
<td>0</td>
<td></td>
<td>23.2</td>
<td>22.7</td>
<td></td>
</tr>
<tr>
<td>Shape 4</td>
<td>0</td>
<td></td>
<td></td>
<td>3.1</td>
<td></td>
</tr>
<tr>
<td>Shape 5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>symmetric</td>
</tr>
</tbody>
</table>

Figure 4.9: Different shapes and their $M_{bh}, M_A$ measure.

94

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
using $M_{bh}$ (i.e. the entries $M_{bh}(\text{shape 2,shape 3})=M_{bh}(\text{shape 3,shape 5})$) will have difference using $M_A$ (i.e. the entries $M_A(\text{shape 2,shape 3})\neq M_A(\text{shape 3,shape 5})$).

4.4 Results

The following example shows the result of applying our method on some more complicated shapes.

4.4.1 Matching Skulls

Figure 4.10: Three skulls to be matched.

We applied our matching method to the shapes shown in Figure 4.10, which correspond to the silhouettes of a gorilla, a pekin man or homo erectus and a modern man or homo sapiens. The results of the match are summarized in Figure 4.11. The results are shown in three tables (3-5). Table 3 shows the match of the right side, table 4 the match of the left side of the shapes and table 5 summarizes the match. The figures below the tables show the polygons used for the match.

The results of table 5 show that the shapes of the gorilla and the homo sapiens are the most dissimilar and the homo erectus is in between. However the homo erectus is more
Table 3: $M_A(right)$ measure

<table>
<thead>
<tr>
<th>$M_A$(Right)</th>
<th>Sapiens</th>
<th>Erectus</th>
<th>Gorilla</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sapiens</td>
<td>0</td>
<td>5.6</td>
<td>7.2</td>
</tr>
<tr>
<td>Erectus</td>
<td>0</td>
<td>2.8</td>
<td></td>
</tr>
<tr>
<td>Gorilla</td>
<td>symmetric</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

Table 4: $M_A(left)$ measure

<table>
<thead>
<tr>
<th>$M_A$(Left)</th>
<th>Sapiens</th>
<th>Erectus</th>
<th>Gorilla</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sapiens</td>
<td>0</td>
<td>2.6</td>
<td>5.8</td>
</tr>
<tr>
<td>Erectus</td>
<td>0</td>
<td>4.0</td>
<td></td>
</tr>
<tr>
<td>Gorilla</td>
<td>symmetric</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

Table 5: $M_A(Total)$ measure

<table>
<thead>
<tr>
<th>$M_A$(Total)</th>
<th>Sapiens</th>
<th>Erectus</th>
<th>Gorilla</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sapiens</td>
<td>0</td>
<td>8.2</td>
<td>13.1</td>
</tr>
<tr>
<td>Erectus</td>
<td>0</td>
<td>6.8</td>
<td></td>
</tr>
<tr>
<td>Gorilla</td>
<td>symmetric</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

Figure 4.11: Skulls and their $M_A$ measure.

Gorilla   Homo Erectus   Homo Sapiens

96
similar to the shape of the gorilla than to the homo sapiens. This result disagrees with our visual perception.

The explanation of this result is that our $M_A$ metric does not only measures the shape but also the size. In order to prove this, we scaled down the homo sapiens shape and matched the shapes again.

The new results are shown in Figure 4.12. The results are shown in three tables (6-8). Table 6 summarizes the match of the right side of the shapes and table 7 summarizes the match of the left side. Table 8 shows the totals of the match.

These new results show that the shapes of the gorilla and the homo sapiens are the most dissimilar on both sides. The right side of these two shapes is more similar than their left side. This result agrees with our visual intuition.

The homo erectus is somewhere in between of the shapes of the gorilla and homo sapiens. Somehow closer to the homo sapiens than to the gorilla. The right side of the homo erectus is more similar to the homo sapiens than to the gorilla. The left side is also more similar to the homo sapiens than to the gorilla, however the similarity with the homo sapiens is larger on the right side. These new results agree with our visual intuition.

This establishes that the metric $M_A$, captures the size and shape difference and that if we want to measure only the shape, we need to normalize the shapes somehow. Further work needs to be done to establish a normalization criteria that can be applied for general shapes.

Now, for some applications, it may be necessary to include the size in the metric. For those applications the metric $M_A$ may be useful.
Table 6: $M_A(right)$ measure

<table>
<thead>
<tr>
<th>$M_A$ (Right)</th>
<th>Sapiens (2)</th>
<th>Erectus</th>
<th>Gorilla</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sapiens (2)</td>
<td>0</td>
<td>1.7</td>
<td>3.3</td>
</tr>
<tr>
<td>Erectus</td>
<td>0</td>
<td>0</td>
<td>2.8</td>
</tr>
<tr>
<td>Gorilla</td>
<td>symmetric</td>
<td></td>
<td>0</td>
</tr>
</tbody>
</table>

Table 7: $M_A(left)$ measure

<table>
<thead>
<tr>
<th>$M_A$ (Left)</th>
<th>Sapiens (2)</th>
<th>Erectus</th>
<th>Gorilla</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sapiens (2)</td>
<td>0</td>
<td>3.1</td>
<td>5.5</td>
</tr>
<tr>
<td>Erectus</td>
<td>0</td>
<td>0</td>
<td>4.0</td>
</tr>
<tr>
<td>Gorilla</td>
<td>symmetric</td>
<td></td>
<td>0</td>
</tr>
</tbody>
</table>

Table 8: $M_A(Total)$ measure

<table>
<thead>
<tr>
<th>$M_A$ (Total)</th>
<th>Sapiens (2)</th>
<th>Erectus</th>
<th>Gorilla</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sapiens (2)</td>
<td>0</td>
<td>4.8</td>
<td>8.8</td>
</tr>
<tr>
<td>Erectus</td>
<td>0</td>
<td>0</td>
<td>6.8</td>
</tr>
<tr>
<td>Gorilla</td>
<td>symmetric</td>
<td></td>
<td>0</td>
</tr>
</tbody>
</table>

Gorilla   Homo Erectus   Homo Sapiens (2)

Figure 4.12: Skulls and their $M_A$ measure.

98
4.5 Summary

Shape matching and shape recognition are central problems of pattern recognition. This chapter shows the use of the CCT representation for shape matching and recognition.

*Shape matching* is the process of binary answering yes or no to the identity of an *unknown shape* as an instance of a *known shape* (pattern). *Shape recognition* is the process of identifying the degree of similarity (*shape similarity*) between the unknown and the known shapes.

Shape matching and shape recognition are difficult because the criteria for matching shapes are by no way universal and depend heavily on the particular application, or even particular objectives of the matching. In some applications two shapes may be similar if they contain the same structural elements whereas others it may require a more precise match in order to be considered similar.

Furthermore the nature of similarity is not well understood yet and there is theoretical controversy regarding the mathematical representation for shape similarity.

According to one of the two dominant points of view in shape perception, the non-dimensional set theory [52], the attributes of the shape constitute a larger non-dimensional set of attributes and that these attributes may not be reflected by any metric. Thus shapes that are perceived as similar, will have the same set of attributes and shapes that are perceived as dissimilar will different attribute sets.

The alternative point of view is that similarity can be reflected by a distance-like metric, where shapes that are perceived as similar will reflect as a small difference on the metric of their match whereas shapes that are perceived as dissimilar will have a large value on the metric of their match.
One can always find examples to argument in favor of any of these two points of view. One of the factors that makes shape matching difficult is the generality of its scope. In some cases a qualitative identification of the structural elements of the shape is enough, in others a precise measure of their sizes is necessary. We defined two kinds of shape matching: flexible match and exact match. They are not exclusive, the exact match is a particular case of the flexible match.

We do not intend to give an absolute criterion to recognize shape, since it is beyond the scope of this work, rather we provided with the CCT and the Matching Tree a flexible scheme that allows the implementation of shape matching (flexible or exact), featured based or precise, which may be useful for specific cases.

We consider that two shapes are similar if first: they contain the same structural components or features and second: if the total distance produced by the metric that measures the mismatch for every component is within a defined threshold. The threshold could be defined statistically.

The metrics $M_{bh}$ and $M_A$ reflect similarity of shape and size, and they reflect the human intuition as shown in the application examples.
(i) Matching Tree for CCT(Shape 1) and CCT(Shape 2) with the measures $M_{bh}$ and $M_A$ shown next to each node.

$M_{bh}(A_1, B_1) = 0$
$M_A(A_1, B_1) = 0$

$M_{bh}(A_3, B_3) = 2.6$
$M_A(A_3, B_3) = 9.5$

$M_{bh}(A_5, B_5) = 1.5$
$M_A(A_5, B_5) = 2.9$

$M_{bh}(MT) = 4.1$
$M_A(MT) = 13.0$

(ii) $M_{bh}$ Calculation: dotted lines indicate the matching pairs $A_i : B_j$.

Figure 4.13: Measuring the match for shapes 1 and 2.
(i) Matching Tree for CCT(Shape 1) and CCT(Shape 3)
with the measures $M_{bh}$ and $M_A$ shown next to each node.

$M_{bh}(A_1, C_1) = 0$
$M_A(A_1, C_1) = 0$

$M_{bh}(A_3, C_3) = 1.3$
$M_A(A_3, C_3) = 2.0$

$M_{bh}(A_4, C_4) = 1.9$
$M_A(A_4, C_4) = 3.6$

$M_{bh}(A_2, C_2) = 3.4$
$M_A(A_2, C_2) = 10.3$

$M_{bh}(A_5, C_5) = 2.5$
$M_A(A_5, C_5) = 4.5$

$M_{bh}(MT) = 9.1$
$M_A(MT) = 21.8$

(ii) $M_{bh}$ Calculation: dotted lines indicate the matching pairs $A_i : C_j$.

Figure 4.14: Measuring the match for shapes 1 and 3.
(i) Matching Tree for CCT(Shape 1) and CCT(Shape 4), with the measures $M_{bh}$ and $M_{A}$ shown next to each node.

$M_{bh}(A_{1}, D_{1}) = 0$
$M_{A}(A_{1}, D_{1}) = 0$

$M_{bh}(A_{3}, D_{3}) = 0$
$M_{A}(A_{3}, D_{3}) = 0$

$M_{bh}(\phi, D_{6}) = 0.9$
$M_{A}(\phi, D_{6}) = 0.8$

$M_{bh}(MT) = 1.9$
$M_{A}(MT) = 1.8$

(ii) $M_{bh}$ Calculation: dotted lines indicate the matching pairs $A_{i} : D_{j}$.

Figure 4.15: Measuring the match for shapes 1 and 4.
(i) Matching Tree for CCT(Shape 1) and CCT(Shape 5) with the measures $M_{bh}$ and $M_A$ shown next to each node.

(ii) $M_{bh}$ Calculation: dotted lines indicate the matching pairs $A_i : E_j$. 

Figure 4.16: Measuring the match for shapes 1 and 5.
Chapter 5

Summary and Conclusion

Objective

In this chapter we summarize our work and draw our conclusions.

Overview Section 5.1 summarizes the work presented. Section 5.2 lists the contributions of this work and section 5.3 presents the conclusion of this work.

5.1 Summary

The problem of shape representation is central to the recognition of shape which is itself the core of many problems in pattern recognition. The difficulty of the task is evident, judging by the immense amount of work in the field of shape analysis.

On the other hand, the recognition of shape is one of the natural human abilities, and thus can be used as a clear example of the limitations of the computer to mimic human tasks that require some intelligence.

The experimental evidence in the analysis of the human recognition of shape has provided the following results:

- Humans complete information.
- The perception of shape is hierarchical.
- The information of shape is contained in the borders of the object.
• There are points on the boundary that influence the recognition.

• The recognition is influenced by rotation.

While some results were immediately incorporated in the methods of shape analysis, some of them have been plainly ignored thus many of these experimental results have no explanation according to the methods of shape analysis.

We propose an innovative method for the shape representation problem, which improves on the structural representation of shape, in particular, within the existing methods using convexity. The use of convexity is attractive because it is invariant under rotation, scaling and translation, characteristics associated with the concept of shape.

The idea of using convexity has been proposed before, however there is no effective method to implement it. Instead of using predefined structural components from which the shape is built, we represent a shape by flexible structural components contained in every shape.

We perceive the shape as a natural hierarchy of minimum convex components (directed triangles), identified from the information on the boundary and which also describe the incremental reconstruction of the region of the shape.

The hierarchy of directed triangles that represent the shape is described in a binary tree which we call the Convexity-Concavity Tree (CCT) of the shape. Thus the shape can be seen as final result of the incremental addition and subtraction of minimum convex regions, which are described in the CCT.

The method consists of finding the structural components of the shape by the recursive segmentation of the boundary such as the used by Archimedes an the geometers of ancient civilizations, except by the addition of two innovative elements: the description based on an
oriented line (vertical) and the representation of the hierarchy by a binary tree.

We also propose a general method for shape matching based on this representation.

Our method is innovative because it decomposes the shape, by using the boundary information as previous methods do, but considering the convexities which are not on the shape (concavities). Other methods concentrate on the regions that are in the shape, to decompose it. Previous techniques either lacked the expressive power to locate convexities and concavities along the boundary, or they were very sensitive to small variations of the boundary of the shape.

The shape descriptor that we propose, the Convexity-Concavity Tree (CCT), provides a hierarchical representation of shape, which is simple, intuitive and that relates two usually conflicting concepts about shape: region and boundary.

A shape can be reconstructed from the information contained in the CCT by either: (a) reconstructing the boundary, using the vertices of the directed triangles that represent the shape, or (b) reconstructing the region, using the directed triangles. Both concepts are equivalent.

We also obtained a hierarchical shape description which is general, since it is based on the constituting elements of any shape, instead of any other mechanism based on the pixel representation.

We showed the use of the CCT for shape matching and recognition. Since our representation is based on triangles, we introduced new notions of similarity between two triangles which improve on the limitations of the euclidean notion.

We showed that it is enough to define a proper metric for triangles (the nodes), to obtain a global metric for the CCT. We proposed two kinds of matching based on the CCT

107
representation: flexible and exact. The flexible matching consists of the qualitative match between the CCT of the two shapes. The exact matching for two shapes consists in the precise match for every node in their CCT.

We presented the application of our methods to a few shapes and the results that we obtained show that our CCT representation captures the essential shape information. We are also able to characterized some shapes that are difficult to represent by means of analytic equations, such as a maple leaf, a skull, etc.

The representation proposed, provides a flexible matching method, since it allows the flexible matching, where only a qualitative description of the objects is necessary, but also allows a more precise matching, where the exact size of convexities is considered.

The process of matching two shapes is simple, because it is reduced to match the binary trees for the shapes considered. The measure of the match is also straightforward, since it reduces to calculate the similarity between the nodes of the CCT.

The definition of the thresholds for the definition/identification of a pattern, can be obtained from statistical analysis of the shapes which conform the class. Thus providing a solid connection between the structural analysis of shape with the statistical analysis of shape, which are usually considered as competitive approaches.

5.2 Contributions

1. The major contribution of this work is without any doubt, the definition of a simple yet powerful method, which brings into practical application the use of convexity. Convexity has been considered up to now only of theoretical interest, due to the limitations of the methods proposed before.

2. We introduced the following concepts:
• The convexity-concavity tree of a shape.
• The matching tree for two CCTs.
• The concept of directed triangles to represent a convex/concave segment.
• The concept of dominant point to locate the point of segmentation of the boundary.
• A new measure of the similarity between two triangles.

3. We presented a polynomial algorithm to represent the shape.

4. Our method relates two commonly excluding concepts regarding shape: contour and region.

5. The method proposed provides a link between the structural analysis of shape and the statistical analysis of shape.

5.3 Conclusion

We conclude that our representation method produces a natural hierarchy of minimal convex components (triangles) that capture the essence of the shape and overcomes the limitations of other multiresolution schemes.

We defined an adaptive structural component, which provides a qualitative and quantitative description of the structural elements of any shape and this representation can be used for matching and recognition.

Even though our shape representation method is sensitive to rotation, since it relies in the correct orientation of the shape, it models very nicely the human recognition of shape, which is sensitive to orientation. Furthermore, the CCT representation can explain some phenomena of human shape perception, and the similarity between a shape and its skeleton.
The CCT representation can be used with statistical analysis to be able to define some shapes by defining typical sizes (relative or absolute) of its components.

5.4 Future Work

A natural extension of this work is its application to shapes which include holes. We could apply our method also in the inside boundaries (in case of more than one hole), but directing the curve in the clockwise direction. The holes could also be used to help to characterize the shape (i.e. every hole is represented by a CCT).

A more difficult problem to which our method may be applied is the containment of one shape into another one. This problem is more complex because involves the recognition of subparts of the boundary.

An interesting problem is the determination of the definition of a pattern, based on an statistical analysis of the components of a shape, which then can be used for recognition. Thus a "characteristic" maple leaf can be defined after the statistical analysis of its components and thus a grammar and a recognizer can be designed.
Bibliography


Vita

Gustavo Martinez was born in Mexico Distrito Federal, Mexico. He attended the Escuela Superior de Ingeniería Mecánica y Electrónica of the Instituto Politécnico Nacional in Mexico City, where he obtained the degree of Ingeniero en Comunicaciones y Electrónica in 1988. He obtained the degree of Master of Science in Systems Science from Louisiana State University in 1994. He will receive the degree of Doctor of Philosophy in Computer Science in May of 2001 from Louisiana State University.
DOCTORAL EXAMINATION AND DISSERTATION REPORT

Candidate:  Gustavo Martinez

Major Field:  Computer Science

Title of Dissertation:  A Hierarchical Shape Representation by Convexities and Concavities and Its Application to Shape Matching

Approved:

[Signatures]

Major Professor and Chairman

Dean of the Graduate School

EXAMINING COMMITTEE:

[Signatures]

Date of Examination:

1 December 2000