Shape analysis using multiresolution gradient vector field

Goh, Wooi Boon

2006

Goh, W. B. (2006). Shape analysis using multiresolution gradient vector field. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/2637>

<https://doi.org/10.32657/10356/2637>

Nanyang Technological University

Downloaded on 26 Apr 2023 17:51:26 SGT

Shape Analysis Using Multiresolution Gradient Vector Field

Goh Wooi Boon

School of Computer Engineering

A thesis submitted to the Nanyang Technological University
in fulfilment of the requirement for the degree of
Doctor of Philosophy

2006

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research and has not been submitted for a higher degree to any other University or Institution.

.....
Date

_____
Goh Wooi Boon

To
James, Ling, little Josh and Sean (and Jaden)

Acknowledgements

I wish to express my thankful appreciation to my thesis supervisor, Tony Chan for his enduring patience. I am sure he too shares my feeling that this light at the end of the tunnel was a long time coming. To my Deans and Heads of Department, both past and present (Harcharan Singh, Seah Hock Soon, Lau Chiew Tong and Graham Leedham), whom I am sure, had taken some trouble to have my workload lightened and for approving my sabbatical leave. It was during this period that the ideas presented in this thesis took root. To my many colleagues who have helped me with much appreciated expertise, advice, thoughts, encouragement and offer of help. Especially Benjamin Premkumar, Francis Lee, Yeo Chai Kiat, Edmund Lai, Tay Joc Cing and Arlene Bastion. To all my family members, especially my dearest Ling, for their forbearance, understanding and support all these years. My apologies for the many things that I have postponed till '.....when I finish my Ph.D.'. To Jesus, for giving me a reason to keep running this race when my legs grew weary and my spirit wore down.

" .... let us run with endurance the race that is set before us, looking unto Jesus, the author and finisher of our faith, who for the joy that was set before Him endured the cross, despising the shame, and has sat down at the right hand of the throne of God."

Hebrews 12:1-2

Contents

	Page
1. Introduction	1
1.1 Motivations	2
1.2 Objectives	4
1.3 Contributions of Thesis	5
1.4 Thesis Organisation	6
2. Multiresolution Gradient Vector Field	7
2.1 Human Visual System - A Review	9
2.1.1 The Human Visual System	9
2.1.2 Biological Motivations	17
2.2 The Multiresolution Pyramidal Framework	18
2.2.1 Multiresolution Image Pyramids	18
2.2.2 The Gradient Vector Field Pyramid	21
2.2.3 The Smoothing Parameter α	23
2.3 Computational Cost	24
2.3.1 The Gaussian Pyramid	24
2.3.2 The Gradient Pyramid	25
2.3.3 The Gradient Vector Field Pyramid	25
2.4 Summary	27
3. Medial Representation using the Multiresolution Gradient Vector Field	
3.1 Medial Representation - A Review	29
3.1.1 Iterative Skeletonization Algorithms	29
3.1.2 Non-iterative Skeletonization Algorithms	30
3.1.3 Skeletons from Distance Maps and Vector Fields	32
3.1.4 Multiscale Medial Representation	34
3.1.5 Characteristics of Skeleton	35
3.2 Structural and Textural Skeletons	37
3.3 Robust Shape Axes Extraction from Vector Fields	39
3.3.1 Vector Field Disparity Map	39
3.3.2 Shape Axes Extraction from the Disparity Map	44
3.3.3 Maintaining Connectivity of Structural Skeletons	46
3.3.4 Determining Structural and Textural Skeleton End Points	49
3.4 Experimental Results and Discussion	53
3.4.1 Effects of Varying Smoothing Parameter α	53
3.4.2 Scale Invariant Skeletons	56

3.4.3	Rotation-Invariant Skeletons	58
3.4.4	Analysis of Boundary Noise Sensitivity	60
3.4.5	Robustness to Boundary Texture	63
3.4.6	Example Application - Character Recognition	64
3.4.7	Computational Cost	66
3.5	Summary	68
4.	Shape Description using the Multiresolution Gradient Vector Field	69
4.1	Shape Analysis - A Review	70
4.1.1	Shape Representation - Boundary versus Region	70
4.1.2	Shape Similarity and Matching	79
4.2	Shape Description using Gradient Vector Fields	82
4.2.1	The Orientation Histogram	83
4.2.2	Comparing Orientation Histograms	84
4.2.3	Characteristics of the Orientation Histogram	87
4.2.4	Rotation and Mirror Reflection	92
4.3	Experimental Results and Discussion	99
4.3.1	Geometric Invariance	99
4.3.2	Shape Recognition	100
4.3.3	Robustness to Boundary Texture	105
4.3.4	Silhouette Classification	107
4.3.5	Computational Cost	111
4.3.6	Summary of Experimental Results	113
4.4	Limitations of the Orientation Histogram	114
4.5	Summary	117
5.	Part-based Shape Analysis	118
5.1	Part-based Shape Analysis - A Review	118
5.2	MGVF Part-based Shape Analysis	125
5.3	Skeleton Segmentation	131
5.3.1	Skeletal Junction Segmentation	131
5.3.2	Vector Field Disparity Profile Maxima/Minima Segmentation	132
5.3.3	Connectivity End Point Segmentation	135
5.3.4	Skeletal Curvature Maxima/Minima Segmentation	137
5.3.5	Primary Skeletal Segments	139
5.4	Skeleton Merging	140
5.4.1	Merging Criteria	142
5.4.1.1	Orientation Continuity	143
5.4.1.2	Disparity Continuity	143
5.4.2	Merging Process	145
5.4.3	Secondary Skeletal Segments	148
5.5	Part-based Descriptors	150
5.5.1	Source and Sink Nodes	150
5.5.2	Part-based Orientation Histograms	152

5.5.2.1 Segment Histogram	152
5.5.2.2 Connection and Terminus Histograms	153
5.5.3 Skeletal Curvature	157
5.5.4 Limb Width Profile	158
5.6 Visual Saliency of Parts	159
5.6.1 Disparity-based Saliency	159
5.6.2 Length-based Saliency	160
5.6.3 Part Saliency	161
5.6.4 Skeletal Curvature Saliency	162
6. Matching Parts and Shapes	163
6.1 Part Distance	163
6.1.1 Histogram Error	165
6.1.1 Histogram Error	165
6.1.2 Curvature Error	165
6.1.3 Limb Width Error	167
6.2 Global Distance	168
6.2.1 Global Spatial Distances	169
6.2.2 Global Angular Distances	170
6.2.3 Determining Reference Source node Pair	173
6.3 Shape Distance	175
6.4 Bi-directional Shape Distance	179
6.5 Handling Mirror Reflection	181
6.6 Experimental Results and Discussion	182
6.6.1 Geometric Invariance	182
6.6.2 Skeletal Topology	184
6.6.3 Noisy Shape Recognition	185
6.6.4 Robustness to Boundary Texture	187
6.6.5 Occlusion	189
6.6.6 Limb Articulation	192
6.6.7 Silhouette Classification	194
6.6.8 Shape-based Image Database Retrieval	196
6.6.9 MGVF-PBD Parameters	200
6.6.9.1 Part Orientation Histogram Bin Size	200
6.6.9.2 Skeletal Curvature and Limb-width Polynomial Order	201
6.6.9.3 MGVF smoothing parameter α	202
6.6.9.4 Global Distance	203
6.6.10 Human Visual Perception	208
6.6.11 Computational Cost	211
6.7 Summary and Discussion	212
7. Conclusions	216
7.1 Principle Results	216
7.2 Future Work	218
References	221

Appendices

A - Computational Cost - Pyramidal Framework	A-1
A.1 Cost of Building the Gaussian Pyramid	A-1
A.2 Cost of Building the Gradient Pyramid	A-2
B - Additional Experimental Results for Chapter 4	A-4
B.1 Geometric Invariance	A-4
B.2 Robustness to Boundary Texture	A-5
B.3 Silhouette Classification	A-8
C - Query Time Reduction vs. Retrieval Performance	A-9
D - MGVF-PBD Standard Parameter Settings	A-11
E - Shape Databases used in the Experiments	A-14
F - Complete Shape Database Retrieval Results	A-18
G - Data Storage Requirement of the MGVF-PBD	A-21
G.1 Storage Requirement of Part Descriptor	A-21
G.2 Storage Requirement of Shape Database	A-22

Author's Publications

List of Figures

	Page
Fig. 1.1	1
Fig. 1.2	3
Fig. 2.1	7
Fig. 2.2	8
Fig. 2.3	10
Fig. 2.4	11
Fig. 2.5	13
Fig. 2.6	14
Fig. 2.7	15
Fig. 2.8	16
Fig. 2.9	18
Fig. 2.10	19
Fig. 2.11	20
Fig. 2.12	20
Fig. 2.13	21
Fig. 2.14	23
Fig. 3.1	31
Fig. 3.2	32
Fig. 3.3	35
Fig. 3.4	36
Fig. 3.5	37
Fig. 3.6	38
Fig. 3.7	40
Fig. 3.8	41
Fig. 3.9	42
Fig. 3.10	43
Fig. 3.11	43
Fig. 3.12	45
Fig. 3.13	46
Fig. 3.14	47
Fig. 3.15	48
Fig. 3.16	49
Fig. 3.17	49
Fig. 3.18	50
Fig. 3.19	52
Fig. 3.20	52
Fig. 3.21	54
Fig. 3.22	55
Fig. 3.23	56
Fig. 3.24	57
Fig. 3.25	57
Fig. 3.26	58
Fig. 3.27	59
Fig. 3.28	61
Fig. 3.29	62

Fig. 3.30	Skeletons extracted from boundary-textured hare shapes	63
Fig. 3.31	Character recognition test images	64
Fig. 3.32	Extracted skeletons for the clean euro symbol	65
Fig. 3.33	Extracted skeletons for the corrupted euro symbol	66
Fig. 4.1	Fourier descriptor extracted from a radii-based shape signature	72
Fig. 4.2	The computation of the shape context	73
Fig. 4.3	Some geometric properties of a shape	74
Fig. 4.4	Steps in deriving the Zernike moments descriptor	76
Fig. 4.5	Typical trademark images	77
Fig. 4.6	A weakness of boundary-based shape representations	78
Fig. 4.7	The triangle inequality property	80
Fig. 4.8	The continuity property as applied to the curvature of a corner	80
Fig. 4.9	Continuity in boundary discretisation, noise and occlusion	81
Fig. 4.10	Invariance to higher-order transformation and erroneous matches	81
Fig. 4.11	Bin-by-bin and cross-bin histogram comparison	84
Fig. 4.12	The monotonicity characteristics of orientation histograms	87
Fig. 4.13	Curvature and occlusion continuity	88
Fig. 4.14	Comparing a circle with polygonal shapes	88
Fig. 4.15	Scale-normalised orientation histograms and boundary noise	89
Fig. 4.16	Orientation histograms and smoothing parameter α	90
Fig. 4.17	Reducing the effects of boundary discretisation noise using α	91
Fig. 4.18	Effect of rotation and mirror reflection on the orientation histogram	92
Fig. 4.19	Histogram peak normalization is unreliable	93
Fig. 4.20	Some shapes and their respective measures of elongatedness L_G	96
Fig. 4.21	Orientation estimates from moment and gradient-based techniques	96
Fig. 4.22	Influence of occluding object on estimated global orientation	97
Fig. 4.23	Geometric invariance of GGON orientation histogram	99
Fig. 4.24	Shapes used in the shape recognition experiments	100
Fig. 4.25	Shape recognition results using MGVF at differing α values.	101
Fig. 4.26	Shape recognition results using global orientation normalisation	102
Fig. 4.27	AHN gives best shape recognition compared to GGON or FTN	103
Fig. 4.28	Comparative shape recognition results	104
Fig. 4.29	Boundary texture result using the MGVF-GGON shape descriptor	105
Fig. 4.30	25 silhouettes images used in shape classification experiments	107
Fig. 4.31	Shape classification using MGVF-AHN orientation histogram	107
Fig. 4.32	Shape classification using MGVF-GGON and FTN	109
Fig. 4.33	Shape classification for SC, FD and ZMD	110
Fig. 4.34	Different shape topologies but same orientation histograms	114
Fig. 4.35	Inaccurate classification due to part-related shape modifications	116
Fig. 5.1	The two main approaches to shape decomposition	119
Fig. 5.2	Different schemes use to determine part cuts	120
Fig. 5.3	Problems with boundary-based shape decomposition strategies	121
Fig. 5.4	Shape decomposition based on segmenting skeletons	122
Fig. 5.5	Pruning of spurious shock edges	123
Fig. 5.6	The dynamic nature of part decomposition	125
Fig. 5.7	Drawbacks of purely local and purely global shape measures	126
Fig. 5.8	Incorporating local connectivity information at skeletal junction	127
Fig. 5.9	Unstable ligatures and their detection	128
Fig. 5.10	Overview of MGVF part-based shape analysis	129

Fig. 5.11	Stages for extracting secondary skeletal segments from MGVF	130
Fig. 5.12	Process for computing skeletal part descriptors	130
Fig. 5.13	Skeleton segmentation at skeletal junctions	131
Fig. 5.14	Inadequacy of using only skeletal junction segmentation	132
Fig. 5.15	The vector field disparity profile along the skeletal segments	132
Fig. 5.16	Vector field disparity minima/maxima along skeleton	133
Fig. 5.17	Segmentation at vector field disparity profile maxima	134
Fig. 5.18	Smoothness of Gaussian derivative kernel and zero-crossing points	135
Fig. 5.19	Variable skeletal connectivity	136
Fig. 5.20	Part decomposition based on skeletal curvature	137
Fig. 5.21	Detecting sharp skeletal curvature	138
Fig. 5.22	MGVF skeleton segmentation examples	139
Fig. 5.23	Hard segmentation does not reflect continuous shape deformation	140
Fig. 5.24	Characteristics of medial shape representations	141
Fig. 5.25	Extracted medial axis of a lizard object and its substance measure	142
Fig. 5.26	Orientation continuity between a pair of skeletal segments	143
Fig. 5.27	EVF disparity can give substance and connection measures	144
Fig. 5.28	The orientation deviation criterion	146
Fig. 5.29	Merged segments for two different horse shapes	147
Fig. 5.30	Different combination of primary skeletons in a merged segment	148
Fig. 5.31	Set of all possible secondary skeletal segments	149
Fig. 5.32	Tree structure resulting from recursive application of <i>grow[ ]</i>	149
Fig. 5.33	Source node assignment	151
Fig. 5.34	Source node assignment based on centroid proximity and junction	151
Fig. 5.35	Segment, connection and terminus of a skeleton	152
Fig. 5.36	Unreliable orientation histograms at terminus and connection points	154
Fig. 5.37	Visual significance of terminus diminishes as shape elongate	156
Fig. 5.38	Disparity-based saliency	159
Fig. 5.39	Disparity-based saliency and connectivity analysis	161
Fig. 5.40	Saliency of skeletal curvature dependent on the limb width	162
Fig. 6.1	Overview of parts and shapes matching process	164
Fig. 6.2	Computing saliency-weighted curvature error	166
Fig. 6.3	Large within-class variation and global relationship between parts	168
Fig. 6.4	Position of centroid vary due to local shape deformations	169
Fig. 6.5	Why both angular and spatial relationships are needed	171
Fig. 6.6	Steps involved in determining the reference source node pair	174
Fig. 6.7	Over-complete cost matrix $OCM(i,j)$ and cost matrix $CM(i,j)$	175
Fig. 6.8	Getting square weighted cost matrix by adding dummy cost	178
Fig. 6.9	The shape distance supports partial matching	179
Fig. 6.10	The shape distance does not always produce desirable results	180
Fig. 6.11	MGVF-PBD results for geometric invariance experiment	182
Fig. 6.12	Extracted MGVF skeletons for geometric invariance test images	183
Fig. 6.13	Influence of shape size on the reliability of the MGVF-PBD	183
Fig. 6.14	Shapes used in skeletal topology experiment	184
Fig. 6.15	MGVF-PBD results for skeletal topology experiment	184
Fig. 6.16	MGVF skeletons of noisy shape recognition test images	185
Fig. 6.17	Shape recognition performance of the MGVF-PBD	186
Fig. 6.18	Test images for boundary texture robustness experiment	187

Fig. 6.19	Results of experiment to test robustness to boundary texture	188
Fig. 6.20	Viewpoint variation experiment	189
Fig. 6.21	Partial occlusion experiment	190
Fig. 6.22	Shape classification of partial occluded shapes using MGVF-PBD	191
Fig. 6.23	Partial occlusion results for shock graph edit and shape context	191
Fig. 6.24	MGVF skeletons of limb articulation test images	192
Fig. 6.25	Shape classification of articulated shapes using MGVF-PBD	192
Fig. 6.26	Results of articulation experiment using pro-articulation parameters	193
Fig. 6.27	Silhouette classification of 25-shapes dataset using MGVF-PBD	194
Fig. 6.28	The 25-shape database of Pelillo <i>et al.</i>	194
Fig. 6.29	Comparative performance on the dataset of Pelillo <i>et al.</i>	195
Fig. 6.30	MGVF-PBD precision-recall graph for 99-shapes database	198
Fig. 6.31	Representative retrieval results using the 99-shapes database	198
Fig. 6.32	MGVF-PBD precision-recall graph for 216-shapes database	199
Fig. 6.33	Representative retrieval results using the 216-shapes database	200
Fig. 6.34	Precision-recall graphs for histogram bin size experiments	201
Fig. 6.35	Precision-recall graph for polynomial order experiment	201
Fig. 6.36	Precision-recall graph for smoothing parameter experiment	202
Fig. 6.37	Precision-recall graph for global distance experiment	203
Fig. 6.38	Retrieval results with and without the use of global distance	203
Fig. 6.39	Precision-recall graphs for curvature and limb width experiments	204
Fig. 6.40	Retrieval results with and without the use of skeletal curvature	204
Fig. 6.41	Retrieval results with and without the use of limb width profile	205
Fig. 6.42	Retrieval results with and without the use of curvature saliency	206
Fig. 6.43	Precision-recall graph for part saliency experiment	207
Fig. 6.44	Retrieval results with and without the use of part saliency	207
Fig. 6.45	Symmetry saliency experiment	208
Fig. 6.46	Singularity saliency experiment	208
Fig. 6.47	Symmetry saliency matching results	209
Fig. 6.48	Singularity saliency matching results	210
Fig. 7.1	Affine invariant normalisation of shapes	219
Fig. A.1	Gaussian pyramid levels and pixel count	A-1
Fig. A.2	Number of convolutions required to build a Gradient pyramid	A-3
Fig. B.1	Geometric invariance experiment using MGVF-FTN	A-4
Fig. B.2	Boundary texture experiment using MGVF-FTN	A-5
Fig. B.3	Boundary texture experiment using shape context	A-5
Fig. B.4	Boundary texture experiment using ZMD	A-6
Fig. B.5	Boundary texture experiment using FD	A-7
Fig. B.6	Comparative results for silhouette classification experiment	A-8
Fig. C.1	Query time versus K_{max}	A-9
Fig. C.2	Retrieval performance using different K_{max}	A-10
Fig. E.1	Shapes in 99-shapes database	A-14
Fig. E.2	Shapes in 216-shapes database	A-15
Fig. E.3	Extracted MGVF skeletons of 99-shapes database	A-16
Fig. E.4	Extracted MGVF skeletons of 216-shapes database	A-17
Fig. F.1	MGVF-PBD retrieval results for 99-shapes database	A-18
Fig. F.2	MGVF-PBD retrieval results for 216-shapes database	A-19
Fig. G.1	Histogram distribution of part count for 216-shapes database	A-20

List of Tables

Table 3.1	Computational cost in computing Eight and Hare skeletons	67
Table 4.1	Comparative classification performance for textured shapes	106
Table 4.2	Comparative classification performance MGVF descriptors	108
Table 4.3	Shape classification performance of various shape descriptors	109
Table 4.4	Computational costs of various shape descriptors	111
Table 6.1	Computational costs for extracting / matching MGVF-PBD	211
Table G.1	Data storage requirement for MGVF-PBD	A-21

Summary

It is widely held that our perception of visual form is due to the excitation of cells sensitive to orientation and edge information at the boundary of the object. It is hypothesised in this work that the multiscale spatial integration along the hierarchical visual pathway do in some way migrate boundary-based orientation into the interior region of the shape and this dense multiresolution gradient vector field (MGVF) can be exploited for robust representation of shapes. Based on this premise, the thesis proposes a novel computationally efficient multiresolution framework for generating a dense gradient vector field for the interior region of a silhouette shape. Within this framework, the smoothness of the MGVF can be easily adjusted via a single smoothing parameter α .

The thesis subsequently explores the robustness and usefulness of the MGVF in two-dimensional shape analysis. Firstly, a multiresolution algorithm based on the detection of local directional disparity maximas in the gradient vector field is proposed for extracting the medial representation of a shape. It is shown that the MGVF skeleton can be extracted in a noise-robust manner and is invariant to both scaling and arbitrary rotation. Secondly, a global shape descriptor in the form of an orientation histogram is proposed to describe the gradient vector field characteristics within the shape. Comparative evaluation with other popular global descriptors suggests that this descriptor strikes a good compromise between boundary noise robustness, computational efficiency and the ability to classify shapes. Though compact and computationally efficient, the representation of an entire shape with a single global statistical feature made the descriptor sensitive to commonly occurring shape distortions such as occlusion, part articulation, deletion and addition. This finally led to the proposal of a part-based descriptor that incorporates both the MGVF skeleton and the orientation histogram. The decomposition of a shape into parts is based on the segmentation and merging of the MGVF skeletal segments. However, final decomposition is deferred till the shape matching stage as this thesis argues that shape decomposition is a dynamic process, which can only be properly carried out when two shapes are being matched. Comparative evaluation of the proposed MGVF part-based descriptor with those in recent literature shows that its shape retrieval performance is more accurate.

Chapter 1

Introduction

The challenging task of object recognition is often accomplished by analysing the shape of the object. The ease by which humans can perform shape recognition and classification using the projection of these objects on two-dimensional (2D) images belies the fact that this is an extremely difficult task in computer vision. This is due in part to the shape distortion that occurs in the projected image when there are changes in the viewpoint direction, object rotation and illumination. Not to mention the fact that such objects can be subjected to large within-class variations, limb articulation and self-occlusion (see Fig. 1.1a). From a more practical perspective, the process of segmenting the shape from a cluttered background is itself an ill-posed problem that can often introduce noise (see Fig. 1.1b). This thesis focuses on the problem of 2D shape analysis. It addresses issues related to how 2D shapes can be recognised and classified in a robust manner under some of the conditions listed above.

Figure 1.1 – Problems that make 2D shape recognition difficult. (a) Large within-class variation and limb articulation. (b) The extraction of shape regions from natural images via processes such as color-texture [162] and watershed segmentation [120] can result in high boundary noise and partial occlusion. Picture adapted from [124].

1.1 Motivations

The principal issue in shape analysis is to extract a meaningful and robust representation of the shape from its corresponding 2D binary image. The shape in the image is represented by a connected set of pixels that defines an enclosed region. The problem is to transform this set of connected pixels into a representation that is able to describe the shape in a robust manner. The manner by which this representation is derived and the extent it is able to remain invariant to the shape variations highlighted in Fig 1.1, can significantly impact its effectiveness when used for recognition. Shapes have been represented as point sets [18], [40], boundary curves and polygons [14], [65], [79], [124], [149], [190], region-based features [142], [212], [233], medial axis or skeletons [31], [135], [191], [195], [199] among many others. Some of these representations are exclusively boundary-based, exploiting the computational efficiency that comes with dealing with a 1D representation of the shape. This approach often leads to string-like representations, which allow shapes to be matched using well-studied and efficient string matching algorithms [40], [65]. On the other hand, representations such as quad-tree [181], grid-based descriptors [142], the moments-based descriptors [213] are region-based and have been shown to be more resilient to boundary noise [231]. Each approach has its respective strengths and weaknesses as discussed in section 4.1.1.

Like Kimia *et al.* [111], the thesis submits that the shape representation must explicitly and simultaneously exploit both boundary and region information to fully and robustly describe a shape. Such representation also results in the decomposition of shapes that is in close correspondence to human visual perception [198]. The proposed multiresolution gradient vector field (MGVF) presented in chapter 2 also makes explicit use of both boundary and region information since it is created by spatially integrating gradient information at the boundary into the interior region. In this sense, it is conceptually similar to Kimia *et al.*'s lateral wave propagation model [111] and Siddiqi *et al.*'s formulation of the grassfire flow as a Hamilton-Jacobi equation on the Euclidean distance function to the boundary [202]. With the MGVF, the propagation of boundary orientation information into the interior is implemented within a multiresolution pyramidal framework. Biological motivations for the MGVF are given in section 2.1.2. A multiresolution approach is also adopted for practical reasons as it provides a computationally efficient means to generate the MGVF.

Psychophysical experiments on visual phenomena such as luminance modulation and perceptual filling-in [160], [161] are consistent with the propagation-based model. In addition, they also support a medial-based geometry of visual forms [113]. This has motivated the work in chapter 3, which is the development of a robust medial-based representation of shape using the MGVF. The weakness of many medial axes extraction algorithms (see Fig. 1.2a) is its sensitivity to small boundary perturbation or ligatures [11]. Attempts have been made to prune insignificant medial branches [193], but invariably, boundary details have been sacrificed as a result (see Fig. 1.2b). The Hierarchic Voronoi skeleton approach in [156] formulates this pruning process within a more explicit multiscale framework called the skeleton-space. The “core” of Pizer *et al.* [169] is another scale-space approach that describes the medialness function of a shape over a width-proportional scale. The cores describing the interior medialness of shapes are extracted in the wider scales, whilst those of minor boundary protrusions are extracted at finer scales. The cores in the interior of the shape are not necessarily connected to the finer-scaled ones at the boundary. The thesis argues that this disconnected nature is an important feature for boundary noise robust medial-based shape representation (see Fig. 1.2c). This observation motivated the multiresolution approach used to generate the MGVF and has influenced the design of a skeleton extraction algorithm that does not enforce skeletal connectivity.

Figure 1.2 – Extracting medial axes that are robust to boundary noise. (a) Skeleton extracted using the Curve Axis [194] is extremely sensitive to boundary perturbations. (b) The boundary smoothing-based pruning of [193] is able to extract the significant medial axes but those representing boundary details are lost. Adapted from [193]. (c) The proposed MGVF skeleton simultaneously represents both the main shape structure and boundary details using disconnected skeletons.

Indeed, a multiscale approach seems like a natural way to tackle the problem of medial loci extraction but as admitted by Pizer *et al.* [170], the choice of scaling parameter or the amount of smoothing to apply is still an unsolved problem. This has motivated the adoption of a *visual saliency* measure. Acknowledging that ligatures may or may not be formed based on a pre-selected scaling parameter; their impact on subsequent downstream processes such as shape matching can be ameliorated through the use of appropriate weights that quantify their formation stability. Such weights are adopted in the part saliency measure proposed in section 5.6.

A global shape descriptor utilising the MGVF was proposed in chapter 4. The boundary noise robustness of the proposed orientation histogram was observed but further experimentation revealed that robust description of complicated shapes such as those seen in Fig. 1.1a, cannot be realised with a simple global descriptor. This motivated the development of the part-based descriptor in chapter 5, which adopted the MGVF skeleton as the framework for part decomposition and description. The development of the decomposition strategy and shape matching algorithm were motivated by the premise that part decomposition is a dynamic process. In other words, the decomposition of a shape into meaningful constituent subparts is strongly influenced by the shape that it is being compared to. Minor motivations that have influenced this work are discussed in its relevant context within the various chapters.

1.2 Objectives

The main objective of this thesis is to develop robust 2D shape analysis techniques, which can be employed in applications such as content-based image retrieval, automatic shape-based classification and recognition. At the heart of this work is the hypothesis that 2D shapes can be robustly described by a dense multiresolution gradient vector field. To this end, a computationally efficient technique had to be developed for generating this gradient vector field when a binary image of the shape is provided as an input. The motivation for using a medial-based approach for shape representation also required the development of a technique for extracting a medial-based representation that is robust to boundary perturbation. The remaining objectives were to devise appropriate shape descriptors and their accompanying shape matching algorithms. Both global and part-based approaches were explored in this thesis.

1.3 Contributions of Thesis

The contributions of the thesis are listed in this section.

- A computationally efficient multiresolution framework for generating the MGVF was proposed. The MGVF will be shown to be a very useful basis for performing 2D shape analysis.
- A technique for extracting a scale and rotation-invariant skeleton from the MGVF was proposed. The MGVF skeleton is able to simultaneously represent both the gross shape structure and boundary details as shown in Fig. 1.2c. This lead to the introduction of structural and textural skeletons. Comparative studies with several popular skeletonization algorithms suggest that the proposed technique is able to extracted skeletons in a more robust and consistent manner, even under high boundary noise conditions. The disconnected nature of the MGVF skeleton also contributed to the efficiency of the part-based shape matching algorithm.
- A part-based shape analysis technique that utilises the MGVF for skeletal-based part decomposition, part description and visual saliency computation was proposed. It has been shown to be able to handle occlusion, limb articulation, part addition and deletion. It also supports partial matching, which will allow simpler shapes to be matched to complex composite shapes. Comparative studies with other part-based techniques suggest that the proposed part-based descriptor is more accurate when employed in shape-based image database retrieval applications. It also exhibits shape dissimilarity measures that are in close correspondence to the way humans judge shape similarity.
- A minor contribution of the thesis resulting from the initial investigation of the shape describing properties of the MGVF lead to the development of a statistical-based global shape descriptor in the form of the MGVF orientation histogram. Comparative studies with other popular boundary and region-based global descriptors suggest that the proposed global descriptor strikes a good compromise between boundary noise robustness, computational efficiency and the ability to classify shapes. Its limitations lead to the development of the more robust part-based descriptor in chapter 5.

1.4 Thesis Organisation

This thesis is organised into six chapters, each of which is summarised in this section.

Chapter 1 – outlines the motivations for this work and list the main contributions arising from the subsequent investigations.

Chapter 2 – develops the biological motivations behind this work and introduces the MGVF, which forms the key idea on which the remaining work of the thesis is based. The computationally efficient multiresolution pyramidal framework for generating MGVF is also presented.

Chapter 3 – describes how a noise robust medial representation of 2D shapes can be extracted within the proposed multiresolution framework. The concept of structural and textural skeletons was introduced to describe the disconnected nature of the resulting MGVF skeleton. A series of comparative experiments were conducted to study the characteristics of the MGVF skeleton compared to those obtained from other popular skeletonization algorithms. The results show that the proposed algorithm is generally superior in terms of extracting a smooth and consistent skeleton in the presence of boundary noise, arbitrary rotation and scaling.

Chapter 4 – explores the use of the MGVF for global shape description. The idea of the orientation histogram is introduced along with various techniques for achieving scale and rotation invariance. A series of comparative experiments were conducted to study the performance of the MGVF-based descriptor and other well-known boundary and region-based global shape descriptors. The results for the MGVF orientation histogram were mixed. It was superior in some tests but not so in others. Its global nature also made this descriptor susceptible to problems such as occlusions and limb articulation.

Chapter 5 – set out to address these shortcomings with a part-based descriptor. It used the skeletal framework in chapter 3 for part decomposition and incorporated the orientation histogram in chapter 4 for part description. Criteria for skeletal segmentation and merging were described. Various distance and saliency measures were developed for use in the part-based shape matching algorithm. Experimental results suggest that the shape retrieval performance of the proposed descriptor is comparable if not superior to recently published shape matching techniques.

Chapter 6 – concludes with a discussion of the principal results and explores some of the possible future extensions of this work.

Chapter 2

Multiresolution Gradient Vector Field

Consider an object such as the prickly pear in the binary image $I(x,y)$ of Fig. 2.1. The primary visual information available to describe its shape resides mainly in the regions of high contrast, that is at the boundary. One can extract this information by simply applying edge detection filters to the image. Gaussian derivative filters with a narrow spread, such as those in Fig. 2.1a provide detailed boundary information but provide no reliable gradient information within the interior region. In contrast, the filters in Fig. 2.1b that have a broader spread can provide a dense gradient vector field, which describes the gross shape of the object but boundary details are inadvertently lost in the process.

Figure 2.1 – Extracting the gradient information of a prickly pear shape in a binary image. The results of using 1st-order Gaussian derivative filters with (a) a narrow spread of $\sigma = 1.0$ and (b) a broad spread of $\sigma = 8.0$.

In short, convolutions with a single fixed-spread filter cannot yield a dense gradient vector field that retains both the description of the object's general visual form and its detail boundary characteristics. This chapter describes a novel multiresolution pyramidal framework that is able to address this issue. The multiresolution gradient vector field (MGVF) resulting from this framework can be described as a dense vector field whose characteristics at the interior and boundary regions can be regulated using a scaling parameter, α . Selecting an appropriate value of α can produce a gradient vector field that describes both the details near the boundary and the general shape class of the object in the interior region (see Fig. 2.2a). Alternatively, a very smooth vector field describing its gross shape form can be obtained with a low value of α (see Fig. 2.2b).

Figure 2.2 – The MGVF extracted for the prickly pear shape. The results obtained with scaling parameters of (a) $\alpha = 0.9$ and (b) $\alpha = 0.1$.

Also included in this chapter is a brief review of the human visual system, which accompanies a discussion on the biological motivations underlying the proposed MGVF paradigm. The usefulness of scale and orientation-based image representation for form analysis is also discussed.

2.1 Human Visual System – A Review

No machine vision system has yet been able to rival the flexibility and robustness of the human visual system in tasks such as robust object segmentation, recognition and classification. Over the years, many in the field of computer vision have sought the knowledge of neurologist and psychophysics researchers to glean the workings of the human visual system. By incorporating some of the relevant characteristics of this robust organic-based visual system, they hope to improve the performance and reliability of machines that employ computer-based vision systems. This section briefly reviews several aspects of the human visual system with the aim of highlighting some of the biological-related motivations for the work on the MGVF presented in this thesis.

2.1.1 The Human Visual System

The rapid progress of neuroscience research in recent years has provided a clearer picture of how the sensory signals are processed by the human brain. Specialisation, modularity, hierarchical organisation and distributed concurrent processing seem to be common principles adopted in processing the huge amount of sensory information that arrives in the brain from the various sensory modalities. The sensory modality that has received the most attention from neuroscience researchers in the last 50 years has been the visual system. Its importance comes as no surprise, as experiments on the macaque monkey's brain suggest that 50% of the neocortex is devoted to the processing of visual information, while the somatosensory cortex and auditory cortex only occupy 11% and 3% respectively [106].

One of the seminal discoveries about the human visual system was made by the duo, Hubel and Wiesel [97] in the late 50's. Their experiments on the anaesthetised cat seen in Fig. 2.3, revealed that neurons in the striate cortex responded selectively to orientation and direction of movement of an edge. Their subsequent investigations [97] revealed that there were several different kinds of cortical cells that had different receptive field characteristics. They classified them into three types, namely simple, complex and hypercomplex cells.

Figure 2.3 – The experiment of Hubel and Wiesel. Neurons in the visual cortex of the cat respond selectively to oriented edges. The neurons fire vigorously to a bar of light oriented at a particular angle but weakly or not at all to other orientations. Adapted from [215].

Simple cells were so named by Hubel and Wiesel because they observed that their response to complex stimuli could be predicted from their straightforward response to individual spots of light. A spot of light falling on an excitatory region causes the cell to fire more vigorously and one that fell on an inhibitory region reduces the cell's firing rate to that which is below its spontaneous rate. In essence, the output of a simple cell is a linear summation of the complex pattern of stimulation falling on its excitatory and inhibitory regions [159]. There are several different subtypes of simple cells identified and the vast majority has an elongated structure that is tuned to respond vigorously to a line or an edge at a specific retinal position and orientation. Fig. 2.4 shows examples of the various receptive field types of simple cells. Some are constructed to be edge detectors, whilst others respond more strongly

to line-like stimuli. Neurophysiological studies by Young have shown that these receptive profiles have a very close resemblance to the Gaussian derivatives [228]. Koenderink and van Doorn [118] described partial derivatives of the Gaussian function whose corresponding receptive field profiles closely mimic well known simple cells found in the primary visual cortex (see bottom half of Fig. 2.4).

Figure 2.4 – Typical receptive field maps for simple cells and their respective Gaussian derivative models [118]. (a) Light line detector, (b) dark line detector and (c) light-dark edge detector.

Complex cells are the single most common cell type in the striate cortex, making up almost three-quarters of the population [99]. Like the simple cells, the complex cells are orientation-sensitive because of their elongated receptive field, but they are slightly larger in size. Unlike their simpler counterparts, they are highly non-linear and hardly respond to a spot of light falling on their receptive field. However, appropriately oriented lines or edges moving over its receptive field causes the cell to fire vigorously and in a sustained manner. About 10 to 20 percent of cells in the upper layers of the striate cortex seem to exhibit sensitivity to the direction of motion of the stimuli (directional sensitivity) [99], whilst the rest respond well to either direction of motion. Complex cells are not very sensitive to the precise position of the oriented stimuli and will produce a similar response if the bar or slit is slightly displaced within the receptive field but its response diminishes rapidly once the orientation of

the bar or slit is altered. However, adaptation to continuous stimulation (i.e. stationary stimuli) means that the complex cell shows most response during the onset or removal of the stimulus or both [159].

Hypercomplex cells have more selective receptive fields than complex cells. They are often called end-stopped cells because their response is most vigorous when a line or an edge stimulates only a portion of its receptive field. If the line extends into its entire receptive field, the response of the cell diminishes. Most researchers now acknowledge that the hypercomplex cells are not a qualitatively different cell type but are composed of end-stopped simple or complex cells [159].

Multiscale analysis is also an inherent characteristic of the human visual system. Hubel and Wiesel observed that the size of the simple cell's receptive field depends on its position in the retina relative to the fovea, with the smallest fields found in or near the fovea [98], [116]. Typical diameters of monkey foveal receptive fields are a few minutes of arc and those at the periphery can be one degree or more [89]. As illustrated in Fig. 2.5a, the size of the receptive field increases with the retinal eccentricity and the mean receptive field size increases in a linear fashion with distance from the fovea. This corresponds well to a commonly known fact that our visual acuity decreases rapidly with eccentricity from the fovea. Even in a given part of a retina, some variation in size is common due to the substantial overlap in receptive fields [60], [98] and the variance in receptive field size is more pronounced at the periphery, as can be observed in the graph of Fig. 2.5a. In short, these observations suggest that simple cells are tuned not just to orientation but also to the spatial size of the light stimuli [44].

Fig. 2.5b shows the multiscale 'stack' model proposed by Lindeberg [133] to describe the distribution of receptive field sizes over the retina. The multiscale characteristics of the receptive fields in the human visual system have motivated the work of many computational vision researchers. Multiscale, multiresolution and wavelet-based image analysis have been and still are very active areas of research [82], [89], [173], [176].

Figure 2.5 – Variation of the receptive field (RF) size of simple cells with retinal eccentricity. (a) A diagram of the RF size (top) measured for various simple cells spread over various retinal eccentricities and the plot of RF size versus retinal eccentricity (bottom) shows that the mean RF size increases linearly with eccentricity. (b) A corresponding multiscale ‘stack’ model of the distribution of RF size over the retina. Adapted from [90] and [133].

Visual information in humans is processed in a *hierarchical* manner as shown in Fig. 2.6. The spatial arrangement of inputs from the retina (*retinotopic map*) is preserved in most stages along the visual pathways, especially in the early stages. Clinical experiments of Marshall and Talbot confirmed that the striate cortex contains a complete retinotopic map [144], an idea suggested much earlier by Gordon Holmes, a British neurologist who conducted clinical examination of patients with cortical lesions. Later, other researchers such as Allman and Kass [4], Fellman and Van Essen [59] and Zeki [229] identified more specialised retinotopic maps beyond V1 (e.g. V2, V3, MT) that are dedicated to processing more complex features. In essence, as one proceeds up the processing hierarchy, the “features” that cells respond to become more complex and selective. For example, unlike cells in the retina and LGN, those in V1 respond only to complex stimuli such as oriented bars and edges.

Figure 2.6 – The human visual processing hierarchy. (a) Visual input goes from the retinal cells in the eyes to the lateral geniculate nucleus (LGN) and then to the primary visual cortex (V1) that is located at the posterior of the occipital lobe. (b) From the striate cortex (V1) the visual signal is then projected into higher-order visual areas in the extrastriate cortex. Adapted from [171] and [227].

Moreover, Zeki [229] found that the neurons in the middle temporal (MT) area (also known as V5) were selective to the direction of stimulus motion but remain relatively unaffected by its color. In contrast, area V4 was found to be sensitive to color changes but not to the direction of motion. Indeed, as illustrated in Fig. 2.7, information reaching V1 splits into three *concurrent processing streams*. One is the *magno stream* (M-stream), originating from the magnocellular ganglion cells in the LGN. The M stream is highly direction-selective and is generally associated with motion processing. The other two streams (P-streams) originate from the parvocellular ganglion cells and are called *parvo-blob* (P-B) and *parvo-interblob* (P-I). The P-B stream is mainly selective to wavelength, with a preference to low spatial frequency stimuli. It is generally considered the pathway for color processing. Physiological properties of the P-I stream overlap those of the M stream in terms of selectivity to orientation and stereo disparity but it is more wavelength-selective than direction-selective. The P-I stream represents a major source of orientation and disparity information [54],[137].

Figure 2.7 – Functional cortical areas for processing color, motion, depth and form. Adapted from [54].

The P-streams form the ventral pathway, supplying orientation and depth information to the inferotemporal (IT) cortex via V4, whilst the M-stream forms the dorsal pathway leading to MT and the parietal cortex, as shown in Fig. 2.6b. Clinical observations of patients with lesions in the parietal cortex show that their visual orientation is disrupted. They do not respond to objects presented to the visual field contralateral to the parietal lesions. In contrast, patients suffering from lesions in the temporal cortex demonstrated difficulty in discriminating forms, identifying faces and have poor memory of object's shape. These clinical observations suggest that the parietal cortex (dorsal pathway) is responsible for spatial representation (i.e. the 'where') and the temporal cortex (ventral pathway) is responsible for object recognition (i.e. the 'what').

Spatial integration occurs along the human visual pathway as is evident from the increase in receptive field size as one proceeds up the visual processing hierarchy. For example, the receptive field diameter of MT cells is about 10 times that of corresponding V1 cells. This loss of spatial resolution is understandable considering the fact that the two largest areas (V1 and V2) each occupy over 1100 mm^2 while the MT area occupies only about 55 mm^2 [106] and all these areas are found to be retinotopically mapped. In many visual analysis tasks such as shape perception and motion analysis, spatial integration is necessary to disambiguate local visual cues. Therefore, the loss in the spatial resolution of mapping of the visual space moving up the visual processing hierarchy result in a gain in the global understanding of the visual scene. Fig. 2.8 shows how the ambiguous local response to the motion of a complex shape in the V1 area results in a coherent sense of motion of the entire object at the V5 area.

Figure 2.8 – Resolving the aperture problem of motion analysis along the hierarchical visual pathway. Local response to moving edges by cells in V1 is perceived as a coherent global motion of the entire object by cells in V5. Adapted from [229] based on the work in [153].

2.1.2 Biological Motivations

Several aspects of the work presented in this thesis have their underlying motivations derived from the characteristics of the human visual system presented in section 2.1.1. The fact that early visual processing is orientation-selective motivates the fundamental assumption that the gradient vector field derived from the image is a rich source of information for further downstream processing tasks such as form analysis. Orientation of gradient vectors and their directional relationship with nearby vectors provide a useful source of information for describing the shape of an object. At a lower level, the gradient information used in this work is limited to those derived from the 1st-order Gaussian derivative filters, similar to those observed in many of the simple cells in the striate cortex [118].

Progressive spatial integration and increase in receptive field size is typically observed as one moves up the visual pathway. Even though the exact nature and mechanism of this spatial integration is still not well understood by researchers, it does in many ways justify the use of a dense vector field similar to the MGVF described earlier. The dense vector field provides orientation information even in the interior of a shape where local processes would have detected no pertinent visual information (except for color and image intensity). Indeed, one of the propositions of this thesis is that robust description of shape forms must explicitly incorporate both boundary and region-based information. Contour analysis alone will not provide a robust solution, especially in the presence of a large amount of boundary noise or boundary texture. Discussion of this issue and others related to shape analysis are detailed in subsequent chapters.

The multiscale characteristics of the human visual system motivate the adoption of a multiresolution pyramidal framework. The ability to adapt a scaling parameter to the analysis of high-level features such as skeleton and shape is important. It is well known that the detection of image features is implicitly tied to the scale in which the image is analysed [134]. What is generally more difficult, is to decide what scale is appropriate for the analysis of different image features. In any case, the proposed pyramidal framework does provide a free parameter, which can be selectively adjusted to suit the noise conditions or the specific requirements of the shape analysis task at hand.

2.2 The Multiresolution Pyramidal Framework

A novel multiresolution pyramidal framework is proposed for generating the MGVF of shapes in 2D binary images.

2.2.1 Multiresolution Image Pyramids

Multiresolution and multiscale image analysis involves the representation and analysis of images at more than one resolution. The appeal of this approach stems from the fact that features that are hard to detect at one scale may be easily detected at another. As described in the previous section, the multiresolution paradigm is also motivated by the many models of human visual systems that recognise that images are simultaneously processed at multiple scales [118], [174], [226].

Figure 2.9 – The multiresolution image pyramid. (a) A pyramidal image structure consisting of a stack of images, each an octave size apart from the other. (b) Several gray level images from the Gaussian image pyramid.

A very useful and powerful structure for representing images at more than one resolution is the multiresolution image pyramid proposed by Burt and Adelson. They first proposed the image pyramid as an efficient means of performing image compression [41], [42]. As shown in Fig. 2.9a, an image pyramid can be viewed as a collection of decreasing resolution images arranged in the shape of a pyramid. Its base (level 0) contains the original high-resolution representation of the image being processed and its apex (level N) contains the lowest resolution approximation of the

image. The lowest possible resolution at level N will depend on the size of the generating kernel and the original resolution of the image. In any case, resolutions smaller than 8×8 pixels are generally meaningless for most pyramidal-based analysis algorithms due to extensive spatial integration of foreground object and background. A popular pyramidal representation constructed with a Gaussian weighted kernel is the *Gaussian pyramid*. It consists of a sequence of images in which each image is a low-pass filtered and sub-sampled copy of its predecessor, as shown in Fig. 2.9b.

Figure 2.10 – A 1D graphical representation of how the *REDUCE* operation can be used repeatedly to construct the Gaussian pyramid.

There are several useful pyramid operations described by Burt in [42]. The *REDUCE* operator generates a lower resolution image from its predecessor. Consider the task of building a N -level Gaussian pyramid G starting with the original image $I(x,y)$, which represents the highest resolution level G_0 of the pyramid. For $0 < l \leq N$, $G_l = \text{REDUCE}[G_{l-1}]$ is defined as

$$G_l(x, y) = \sum_{m=-k}^k \sum_{n=-k}^k w(m, n) G_{l-1}(2x + m, 2y + n) \quad (2.1)$$

where the weighting function w of size $(2k+1)$ is called the *generating kernel*. In a Gaussian pyramid, the kernel w is a 2D Gaussian function that performs smoothing before the sub-sampling process, as illustrated in Fig. 2.10. For computational efficiency reasons, a separable, normalised and symmetric Gaussian kernel of $k = 2$ is used. An additional ‘equal contribution’ constraint is added so that each node in level l contributes the same total weight to the nodes at level $l+1$. The derivation of the values of the generating kernel w is given in [42].

Figure 2.11 – A 1D graphical representation of how the *EXPAND* operation converts the image G_{l+1} of dimension $M+1$ to image G_l of dimension $2M+1$.

Another pyramid operation, *EXPAND* is used to expand an image of dimension $M+1$ to $2M+1$ by interpolating sample values from a low resolution image, as illustrated in Fig. 2.11. If G_l is derived by expanding its low resolution image G_{l+1} , then for $0 \leq l < N$, $G_l = \text{EXPAND}[G_{l+1}]$ is defined as

$$G_l(x, y) = 4 \sum_{m,n=-k}^k w(m, n) G_{l+1}\left(\frac{x+m}{2}, \frac{y+n}{2}\right) \quad (2.2)$$

and the summation is only carried out when $x+m$ and $y+n$ are even numbers. Fig. 2.12 shows how the repeated application of the *EXPAND* operator to images at four different levels of the Gaussian pyramid can be used to produce full resolution-sized images with varying degrees of low-pass filtering.

Figure 2.12 – The three levels of the Gaussian pyramid in Fig. 2.9b expanded to the size of the original image using repeated application of the *EXPAND* operator.

2.2.2 The Gradient Vector Field Pyramid

It is assumed that the input image is a 2D binary image $I(x,y)$ of size $R \times R$ pixels. The procedure for creating a Gradient Vector Field pyramid is summarised in Fig. 2.13. Firstly, a Gaussian pyramid $G_l(x,y)$ of $N+1$ levels is constructed by applying the *REDUCE* operation defined in (2.1) to the lowest level G_0 of the Gaussian pyramid, which is formed by the input image $I(x,y)$. This results in a new image level G_1 of size $(R/2) \times (R/2)$ pixels. By iteratively applying the *REDUCE* operation N times to every pixel of each resulting image, the Gaussian pyramid will eventually be constructed.

Figure 2.13 – Steps for constructing a Gradient Vector Field pyramid from a binary image $I(x,y)$. Gaussian-based operators are employed to smooth the binary image so that an approximation to a continuous function is reasonable.

A vectorial Gradient pyramid $H_l(x,y)$ is then derived by convoluting the scalar Gaussian pyramid with two derivative filters. The resulting Gradient pyramid consists of two scalar pyramids $H_l^x(x,y)$ and $H_l^y(x,y)$ given by

$$H_l^x(x,y) = g_{\sigma_H}^x(x,y) * G_l(x,y) \quad \text{and}$$

$$H_l^y(x,y) = g_{\sigma_H}^y(x,y) * G_l(x,y) \quad \text{for } 0 \leq l \leq N \quad (2.3)$$

where $*$ is the convolution operator and the convolution kernels $g_{\sigma_H}^x$ and $g_{\sigma_H}^y$ are Gaussian derivatives of the first-order in the x and y directions, respectively and are given by

$$\begin{aligned} g_{\sigma_H}^x(x, y) &= -\frac{x}{\sigma_H^2} \exp\left(-\frac{x^2 + y^2}{2\sigma_H^2}\right) \quad \text{and} \\ g_{\sigma_H}^y(x, y) &= -\frac{y}{\sigma_H^2} \exp\left(-\frac{x^2 + y^2}{2\sigma_H^2}\right) \end{aligned} \quad (2.4)$$

The Gradient pyramid describes the gradient of the image $I(x, y)$ at octave scales as the pyramid is traversed from one level to another. At lower levels of the Gradient pyramid, the interior of the shape often has undefined gradient values (see Fig. 2.13). This is because the size of the Gaussian derivative kernels is intentionally kept small to reduce computational requirements (typically of size 7×7 to 11×11 pixels are used). Only gradient values near the shape's boundary can be computed reliably.

A third step is therefore required to obtain a dense gradient vector field that is properly defined at all localities within the shape and at all image levels. As shown in Fig. 2.13, each level V_l of the Gradient Vector Field pyramid is constructed by iteratively applying *EXPAND* to the proportioned sum of the corresponding level H_l of the Gradient pyramid and *EXPAND* $[V_{l+1}]$, where the image V_{l+1} was computed from the previous iteration. The apex of the pyramid, V_N is first initialised with the value of H_N , the apex of the Gradient pyramid. This is necessary so that the iterative expand and summate process can be subsequently initiated, starting at $l=N-1$ and terminating at $l=0$. More formally, this *expand-summate* process is defined as

$$\begin{aligned} V_l &= H_l & \text{for } l = N \\ V_l &= \alpha H_l + (1 - \alpha) \text{EXPAND}[V_{l+1}] & \text{for } 0 \leq l < N \end{aligned} \quad (2.5)$$

The highest resolution vector field at V_0 of the Gradient Vector Field pyramid is defined as the multiresolution gradient vector field (MGVF). A scaling parameter $\alpha \in [0, 1]$ is introduced to govern the summation ratio between one level and the next and its value determines the smoothness of the gradient vector field. The scaling parameter α is also interchangeably referred to as the *smoothing parameter* as this better reflects its influence on the MGVF.

2.2.3 The Smoothing Parameter α

The MGVPs resulting from various values of α are shown in Fig. 2.14. The orientation histogram[†] for each of the respective vector fields shows that the boundary characteristic of each shape is prominent at high values of α . At $\alpha=0.9$, the long edges of the rectangle in Fig. 2.14a produced two sharp peaks, suggesting a uniform straight boundary whilst that of the wavy object in Fig. 2.14b produced two rounded peaks, suggesting an edge that is irregular (i.e. wavy). In both cases, the presence of two peaks indicates an elongated object. As the smoothing parameter is reduced to $\alpha=0.2$, the orientation histograms of both objects begin to look alike. Since the boundary characteristics of both objects have been de-emphasised, the similarity in their gross shape becomes apparent (i.e. they are both elongated objects with roughly the same height to width ratios). However, when too much smoothing is applied, like in the case of $\alpha=0.1$, the gradient vector field is no longer informative as both orientation histograms have flattened out. The examples in Fig. 2.14 demonstrate the usefulness of the MGVP in describing the various characteristics of an object's shape.

Figure 2.14 – A single parameter α in (2.5) is used to vary the smoothness of the gradient vector field. Elongated objects with (a) straight, (b) wavy boundaries and their respective MGVP (plotted with unit magnitude) and orientation histogram.

[†] Note: Details on the orientation histograms can be found in section 4.2.1

2.3 Computational Cost

The computational cost associated with the extraction of the MGVF from an image is essentially the cost of building the Gradient Vector Field pyramid V . This cost is the sum total of the cost for constructing the Gaussian pyramid G , vectorial Gradient pyramid H and the cost of the iterative expand-summate procedure applied to each level of H to derive V . Each stage within the pyramidal framework is analysed in turn.

2.3.1 The Gaussian Pyramid

The computational cost of constructing the Gaussian pyramid G depends mainly on the number of times the *REDUCE* operation is applied and the cost of each such operation. The overall construction process is highly efficient due to the hierarchical nature of the pyramid, which reduces the image's pixel count by a factor of 4 going up each level. The basic operation is the repeated convolution of the Gaussian generating kernel w with the image at each level l . Direct computation at each image pixel with a kernel of size $n \times n$ typically requires n^2 multiplications and $n^2 - 1$ additions. However, choosing a separable Gaussian kernel, as suggested by Burt [42], allows the 2D convolution to be implemented as two consecutive 1D convolutions in the horizontal and vertical directions. Furthermore, a symmetric kernel with a width n that is odd, reduces the number of multiplications to $(n - 1)/2$ and additions to $(n - 1)$. As a result, the computational requirement per image pixel is given by

$$c_G = (n + 1)M + 2(n - 1)A \quad (2.6)$$

where M and A are the cost of performing a multiplication and an addition, respectively. The value of n is set to 5 as in [42]. Given an input image $I(x, y)$ of size $R \times R$ pixels, the computational cost required to construct a complete Gaussian pyramid of levels $N = \log_2(R) + 1$, as derived in Appendix A.1, is shown to be

$$\text{cost}_G = \frac{R^2 - 1}{3} c_G \quad (2.7)$$

In reality, the computational cost is less than that in (2.7) since the number of useful pyramid levels N is often less than $\log_2(R) + 1$ due to the practical considerations in implementing convolutions when the image is only a few pixels wide.

2.3.2 The Gradient Pyramid

The computational cost of constructing the Gradient pyramid $\mathbf{H}$ is higher than that of the Gaussian pyramid for of two main reasons. Firstly, pyramid $\mathbf{H}$ is a vectorial pyramid consisting of two scalar pyramids H^x and H^y , therefore doubling the number of convolutions required. Secondly, the convolution with the 1st-order Gaussian derivative kernels $g_{\sigma H}$ in (2.4) is applied to all levels of the Gaussian pyramid, including that of the original image resolution of $R \times R$ pixels, resulting in a computational cost (see derivation in Appendix A.2) given by

$$cost_H = 2 \left[\frac{4R^2 - 1}{3} \right] c_H \quad (2.8)$$

where the construction of two pyramids increases the cost by a factor of two and c_H , the cost incurred at each image node in the Gaussian pyramid is given by

$$c_H = (m+1)M + 2(m-1)A \quad (2.9)$$

where m is the odd-sized width of the square, separable and symmetric Gaussian derivative kernels. m is larger than n and is set to 7, 9 or 11 depending on σ_H .

2.3.3 The Gradient Vector Field Pyramid

The Gradient Vector Field pyramid $\mathbf{V}$ is generated from the Gradient pyramid using the expand-summate operations described in (2.5). The computational cost of the *EXPAND* operation is essentially the same as that of *REDUCE* as one is just applied in the reverse pyramidal direction of the other. Since the same separable and symmetric Gaussian generating kernel w of size $n \times n$ pixels is used in the *EXPAND* operation, the computational cost per image node is similar to that in (2.6). The number of *EXPAND* operations required is double that described for the *REDUCE* operation in (2.7) since this operation is carried out on the H^x and H^y pyramids separately. Additionally, due to the summation between image layers based on the proportion dictated by the scale parameter α , an additional 2 multiplications and 1 addition is required. The number of times this operation is carried out is similar to that given in (2.8). Therefore the total computation cost for deriving the Gradient Vector Field pyramid $\mathbf{V}$ from the Gradient pyramid $\mathbf{H}$ can be summarized by

$$cost_V = 2 \left[cost_G + \frac{4R^2 - 1}{3} (2M + A) \right] \quad (2.10)$$

The total computational cost required to generate the multiresolution gradient vector field for an image $I(x, y)$ of size $R \times R$ pixels is given by the sum of the cost incurred at each stage of the pyramidal framework and is given by

$$cost_M = cost_G + cost_H + cost_V \quad (2.11)$$

For example, the computational cost required to derive the MGVF for an image of size 129×129 using kernels of size $n = 5$, $m = 7$, requires 543593 multiplications and 709999 additions. This computational cost does not include other relatively small algorithmic-related costs such as pixel sub-sampling and iterative loop management. As a comparison, the cost of applying straightforward convolutions to the same image with broad separable and symmetric Gaussian derivative filters whose width is say 21×21 pixels, will result in a cost of 732204 multiplications and 1331280 additions. This is about 1.3 and 1.9 times more multiplications and additions respectively. For wide shapes, even such a filter width may not guarantee reliable gradient vectors in the interior of objects. The proposed multiresolution pyramidal framework is a more efficient means of getting a dense gradient vector field. The savings become more apparent as the size of the input image increases or the size of the derivative kernels increase due to the need to increase the spread of the Gaussian derivative kernels.

An alternative means of convolution is to use the Fast Fourier Transform (FFT) method, which requires $O(R^2 \log R)$ operations. In order to implement the convolution, the image of size $R \times R$ undergoes the FFT operation, followed by R^2 multiplications and then the inverse FFT. An efficient implementation of 2D FFT using row-column decomposition with 1-D FFT algorithm requires $0.5 \times R^2 \log(R^2)$ complex multiplications and $R^2 \log(R^2)$ complex additions [132]. This means the entire convolution process using 2 derivative kernels will require about $2R^2 \times (\log(R^2) + 1)$ multiplications and $4R^2 \log(R^2)$ additions. An image of size 128×128 pixels will cost about 491520 multiplications and 917504 additions. Since the complex multiplication and addition typically cost twice that of its real equivalent, the proposed multiresolution method is still more efficient than the FFT-based convolution approach.

2.4 Summary

This chapter introduced the multiresolution gradient vector field (MGVF) and the proposed multiresolution pyramidal framework used for its construction. It started with a review of the human visual system and then developed the biological motivations behind the MGVF. Hubel suggested in [99] that the interior of a form does not itself excite cells in our brain and that our awareness of the interior depends only on cells sensitive to the borders. This thesis supports this notion but it further hypothesises that the representation of visual form does involve its interior at some higher level in visual pathway. Some manner of spatial integration along the visual pathway would effectively migrate boundary-based orientation into the interior region of the object.

A novel multiresolution pyramidal framework is proposed to generate this dense gradient vector field given a silhouette image of an object. This pyramidal framework provides a free parameter α that allows one to control the smoothness of the vector field generated. It will be shown in subsequent chapters that the smoothing parameter α can be used to provide robustness to boundary noise and texture when the MGVF is used in various shape analysis applications.

The computational cost of generating the MGVF within the proposed multiresolution pyramidal framework was also discussed. It was shown that the multiresolution approach for generating a dense gradient vector field is relatively less costly than if straightforward convolution with broad Gaussian derivative filters was employed. More importantly, the MGVF is able to simultaneously retain both the gross visual description of the shape and its boundary characteristics; the former is described by the gradient vector field in the interior of the shape whilst the latter is described by vectors near the boundary.

Subsequent chapters will demonstrate the usefulness and robustness of the MGVF for representing 2D shapes. The next chapter describes how a robust medial representation of shapes can be derived from the proposed MGVF.

Chapter 3

Medial Representation using the Multiresolution Gradient Vector Field

The importance of the skeleton or medial axis was highlighted by the early works of Blum [30], [31], especially in the context of visual perception and shape analysis. Since then, there have been an enormous amount of related research and algorithms published under topics such as “thinning”, “medial axis” and “skeleton”. Many authors have used these terms interchangeably. However, the term “thinning” generally refers to the process of reducing elongated shapes into a line representation, typically appearing in applications related to optical character recognition (OCR) [123]. The term “medial axis” is commonly used to denote the locus of centres of maximal disks contained within a shape [30] or in the case of algorithms related to the distance transform [66], they represent the local maxima of a distance-transformed image of a shape. The term “skeleton” is the most general and it simply refers to the representation of a shape by a collection of thin (or nearly thin) lines, arcs and curves, without regards to the nature of the shape or algorithm used [121]

This chapter describes how the MGVF can be used to provide a medial representation of shapes. It begins with a brief review of various approaches for extracting skeletons and then summaries some useful characteristics skeletons should possess. Section 3.2 introduces two types of skeleton defined in this work, namely the *structural* and *textural* skeletons. Section 3.3 extends the multiresolution pyramidal framework of chapter 2 such that a scale-invariant medial representation can be extracted from the MGVF. Algorithms for local ridge extraction, directional-uphill climbing and end point connectivity analysis are also described. Section 3.4 presents experimental results highlighting the characteristics and comparative advantages of the MGVF skeleton over other skeletonization algorithms.

3.1 Medial Representation – A Review

The medial representation of a shape is useful because it provides a compact representation that preserves many of the general characteristics of the shape, such as topology, size and proportions. For instance, different shape topologies can be distinguished by examining the number and characteristics of the bifurcation joints in the extracted skeletons. Size and proportions can be characterised by examining the relationship of end points in the skeleton. The process of skeletonization often constitutes the preliminary feature extraction stage of many image analysis applications such as shape analysis, handwriting analysis, object recognition, etc. As such, it is not surprising that there are many algorithms proposed for extracting skeletons. This section briefly reviews several common approaches to the problem of skeleton extraction in binary images, including methods that employ multiscale techniques. It rounds off by discussing the various desirable characteristics of a skeleton and highlights the main characteristics emphasised by the approach presented here; which focuses on extracting a boundary noise-robust skeleton from the MGVF.

3.1.1 Iterative Skeletonization Algorithms

The most common way to extract the skeleton of a shape is to implement a thinning procedure that iteratively ‘peels’ away successive layers of pixels on the boundary of the shape while retaining any pixels whose removal would alter connectivity or shorten the branches of the skeleton [104]. Indeed, there are numerous algorithms using this approach and they are well summarised by Lam *et al.* in [121]. These algorithms are classified as either sequential or parallel based on the way they examine the object pixels before erosion or retention.

In *sequential thinning* algorithms, the pixels are examined in a fixed sequence during each iteration and the fate of the current pixel being processed requires the knowledge of the previously processed pixels in the current iteration. The classical sequential thinning algorithm is that proposed by Hilditch [93], which employs a raster scan sequence to complete one entire peel per iteration. An alternative scanning strategy is one that follows the object’s contour [6], [225]. The advantage of the contour-based approach over that of the raster scan is that it only requires the contour pixels to be examined during each iteration.

In *parallel thinning* algorithms, the deletion of pixels in the current iteration would depend only on the result that remains after the previous iteration. This means all pixels can be examined independently and in a parallel manner during the iteration. Parallel thinning algorithms are generally preferred over sequential ones because they are more amenable to computationally efficient implementation on parallel hardware. Some authors [123] claim that results from parallel algorithms tend to produce more isotropic skeletons because they examine all pixels simultaneously using the same set of conditions for pixel deletion. However, results presented in section 3.4.3 will show that arbitrary rotation invariance (i.e. not limited to multiples of $\pi/2$ radians) is difficult to realise with iterative algorithms, be they sequential or parallel. Fully parallel algorithms can have difficulty preserving connectedness in the skeleton if only 3×3 pixels support are considered. Therefore, the usual work around is to divide each iteration into two or four sub-iterations in which only a subset of contour pixels is considered for removal. An example of this approach is the commonly cited parallel thinning algorithm by Zhang and Suen [230], which uses two sub-iterations. Subsequent improvements to this work can be found in [141], [225]. There are also many truly parallel thinning algorithms that use only a single sub-iteration. They typically employ a larger neighbourhood support of about 13 pixels to an optimally small value of 11 pixels [91]. Some of these algorithms with good boundary noise immunity include the AFP3 algorithm by Guo and Hall [86] and the MB2 algorithm by Bernard and Manzanera [23]. An evaluation of several parallel thinning algorithms for OCR application is found in [123].

3.1.2 Non-iterative Skeletonization Algorithms

The non-iterative approach describes non-pixel based algorithms that produce certain medial or centre line of a shape in a single pass without examining each individual pixel and the medial axis transform (MAT) proposed by Blum [30] is perhaps the oldest and most cited example. The medial axis consists of all points within a shape that have two or more equidistant boundary pixels. Montanari [151] gave several equivalent definitions of skeletons in the real plane as shown in Fig. 3.1.

Figure 3.1 – Three equivalent definitions of the skeleton of a shape. (a) The prairie fire model skeleton (white) of a square (gray) obtained as a set of shocks (self-intersection points) cause by the constant velocity propagation of the boundary towards the interior region. Snap shots of the fire front (black) at various times can be seen along with the arrows indicating its direction of travel. (b) The skeleton is the set of centres of maximal disk. (c) The skeleton is the planar projection of the ridges (local maximas) of the distance map from the boundary. Adapted from [110].

Fig. 3.1a shows the prairie fire model definition of a skeleton, such as those obtained from the grassfire transform of Blum and Nagel [32]. Propagation of fire-fronts are generated by setting ‘fire’ to the contours of the shape and the skeleton is obtained at points where the fire is extinguished when two or more fire fronts meet. Other variations of this same principle include the active contour model of Leymarie and Levine [127], Shapiro *et al.* [1] and the Kimia *et al.* [111] approach based on boundary curve evolution using partial differential equations. In [111], shocks or singularities that develop in the course of the boundary curve deformation process are categorised by their nature and time of occurrence. This information is useful for boundary noise removal as well as shape description (i.e. parts, bends, protrusions and nodes). Skeletons can also be extracted from the geometric location of centres of maximal disks [30]. A maximal disk is defined as a disk contained within the shape, for which there exists no other disk in the shape that contains it, as shown in Fig. 3.1b. Other forms of skeletons can be extracted using various definition of local symmetries [38], [62], [68], [129] and [172].

In Fig. 3.1c, the skeleton is defined as a set of ridges on a distance map surface related to the shape. Various distance surfaces may be used but the commonality shared by all these skeletonization algorithms is their additional need to implement some form of local ridge extraction technique. This category of skeletonization algorithms is further discussed in the next section.

3.1.3 Skeletons from Distance Maps and Vector Fields

A common method of obtaining the skeleton is to track the local maxima or ridges within the object after applying the *distance transform* to the image. For a binary image I with a set of object pixels O and background pixels $\bar{O}$, the distance transformation of I will yield a distance map D within the object given by

$$D(p) = \min\{dist_M(p, q), p \in O, q \in \bar{O}\} \quad (3.1)$$

where p are all pixels within the object and q are pixels outside the object (i.e. background). The distance between p and q is determined by the function $dist_M$ using the metric M . The distance metric M may take the following forms:

$$dist_4(p, q) = |p_x - q_x| + |p_y - q_y| \quad (3.2)$$

$$dist_8(p, q) = \max\{|p_x - q_x|, |p_y - q_y|\} \quad (3.3)$$

$$dist_e(p, q) = \sqrt{(p_x - q_x)^2 + (p_y - q_y)^2} \quad (3.4)$$

$$dist_E(p, q) = (p_x - q_x)^2 + (p_y - q_y)^2 \quad (3.5)$$

They are termed city block (3.2), chessboard (3.3), Euclidean (3.4) and squared Euclidean metric (3.5), respectively. The city block and chessboard distance transforms can be computed very quickly but their associated skeleton lacks accuracy and is sensitive to planar rotation [127]. The Euclidean versions $dist_e$ and $dist_E$ are equivalent (see Fig. 3.2) but the latter is preferred as it can be computed using only integer operations.

Figure 3.2 – The squared Euclidean distance transform of (a) the binary image results in (b) the distance map. (c) Sample integer values at corner pixels.

The distance metric M used to compute the distance map is critical as it affects the medialness of the extracted skeleton and its sensitivity to arbitrary rotation. The Euclidean-based metrics are the most accurate but they have not been popular in the past due to their costly computational requirements. This led to many algorithms that used an approximation to the Euclidean metric [34]. For example, Danielsson [52] described a sequential nearest-neighbour transform algorithm that was close to being error-free and it ran in $O(nm)$ time, which is proportional to the size of the $n \times m$ image. Though computationally more efficient, these pseudo-Euclidean distance transforms are sensitive to rotation and their inaccuracy increases with the size of the object [66]. With increasing computational power and faster algorithms for computing Euclidean distance maps [51], it is now more appropriate to compute skeletons directly from true Euclidean distance maps, like in the works of Arcelli and Sanniti Di Baja in [7], [8] and Ge and Fitzpatrick in [66]. Once the Euclidean distance map has been computed, there are many algorithms proposed to extract the skeleton. Algorithms such as those described in [7], [8], [66], [154], [196] detect local maximas in the distance map and these points are then connected by tracking a path along some measure of steepest ascending slope or ‘uphill’ climb.

Vector field-based techniques have also been proposed for extracting skeleton features from images. Recently Siddiqi *et al.* proposed a skeletonization algorithm that is rooted in a physical-based analysis of the gradient vector field of the Euclidean distance function [55], [202]. They first compute the gradient vector field ∇D of the distance-transformed image. The divergence of the vector field within a finite area is then integrated to give the average outward flux of ∇D at every point. A threshold is then applied to the computed divergence map to yield a thick connected skeleton, which is subsequently thinned using a homotopy-preserving thinning algorithm. Shroff and Ben-Arie [197] modeled the boundary gradient of a shape as magnetic dipoles and extracted smooth shape axes at points where the magnetic field interaction resulted in local minima. The local minima are detected using a local valley detection algorithm similar to that presented in section 3.3.1. Cross and Hancock [50] proposed a multiscale framework where the vector field is obtained by computing the curl of vector potential found through volume averaging of the tangential edge gradient vectors. Symmetry axes are represented by locations where the curl of the vector potential vanishes. These symmetry points are extracted by fitting a local quadratic

prism surface to the magnitude of the vector potential in each 3×3 image neighbourhood and then applying a non-minimum suppression test to detect directionally consistent symmetry features. A multiresolution skeletonization algorithm based on Electrostatic Field Theory (EFT) was proposed by Abdel-Hamid *et al.* [1], [83]. The EFT-based method provided skeletal representation across a variety of scales by changing the starting potential value during skeleton generation. They compare their skeletonization results with those obtained using the Euclidean Distance Transform [196] and the Charge Particle Method [10] and showed that the multiresolution EFT-based algorithm is less sensitive to boundary noise [83].

3.1.4 Multiscale Medial Representation

Iterative skeletonization algorithms can provide some limited robustness to boundary noise [121]. By and large, a multiscale framework is the most effective means of providing a structured approach for robust skeleton extraction in noisy conditions. Pizer *et al.* provides a good survey of various multiscale medial representations and their respective properties [170]. For example, skeletons can be computed from Voronoi diagrams of points sampled along the shape boundary [38] and Ogniewicz and Kübler cast this approach within a hierarchical framework they called the Hierarchic Voronoi skeleton [156]. Their axis-pruning process reduces insignificant medial branches and organises relevant ones into a hierarchical skeleton-space based on a prominence measure. Rom and Medioni proposed a method for obtaining a shape's segmented axial description along with a hierarchical decomposition of its respective parts [175]. Parts are segmented by detecting curvature sign changes in the smooth local symmetry and then hierarchically decomposed via a recursive procedure that removes each part starting from the smallest.

Pizer *et al.* [169] proposed a scale-space approach that describes the medialness function (cores) of a shape over a width-proportional scale. The cores describing the interior medialness of shapes can be extracted in the wider scales and cores describing minor boundary protrusions and perturbations can be extracted at finer scales. By selecting cores from appropriate scales, core-based analysis of shapes can be made relatively insensitive to boundary perturbation [152]. One of the major strength of the core-based approach is that it can be applied to grayscale images [170] and the cores in the interior of the shape and those near the boundary are not necessarily connected

(see Fig. 3.3). Violating skeleton connectivity has its advantages (see Fig. 1.2). Non-connected medial representations are typically called axial descriptions [175] and Shroff and Ben-Arie argued that connected axes obtained by most skeletonization algorithms do not represent true shape axes [197]. In this sense, the shape axes extracted from the MGVF share this medial characteristic of cores [169] and [175].

Figure 3.3 – Cores for a sawtoothed rectangle. (a) The sawtoothed rectangle shape. (b) The internal core (white) extracted for the primary elongation, the corners and the teeth. (c) The cores depicted in 3D scale space. Adapted from [152].

3.1.5 Characteristics of Skeleton

The characteristics of the extracted skeleton vary with the algorithm used. The real measure of a skeleton's superiority is ultimately decided by its application (e.g. shape matching [201], character recognition [123], etc.). Some general characteristics [23] that skeletonization algorithms are measured by include:

- *Homotopy* – the skeleton must preserve the topology and connectivity of the original shape. Connectivity means the skeleton has the same number of components as the object, and simple connectivity means it has the same number of holes [154].
- *Reconstructability* – the ability to regenerate the original shape from the skeleton given additional information at each skeletal point, such as the distance to the closest boundary point.
- *Unit-width* – the skeleton consists of curves that are one-pixel-thick at all locations.
- *Mediality* – the skeleton should lie in the centre of the shape, with every point of the skeleton an equidistant from its two closest boundary points.

- *Rotation invariance* – the extracted skeleton does not change when the object undergoes arbitrary rotation about the imaging plane.
- *Noise immunity* – the skeleton should be insensitive to boundary perturbations that may alter the object's boundary texture but not its general shape form.
- *Computational speed* – the computational load of the algorithm and its ability to be efficiently implemented on parallel hardware.

It is not surprising that some of these requirements are contradictory. For example, complete homotopy, mediality and perfect reconstructability are fundamentally at odds with boundary noise immunity [23]. To perfectly reconstruct the discretisation artifacts at the boundary, many hair-like skeletal branches (runners) emanate from the main structural skeleton (see Fig. 3.4a). The structural skeleton itself describes the basic shape topology and not its boundary perturbations. On the other hand, skeletons that are robust to boundary noise should be smooth and not plagued by boundary noise-induced runners (see Fig. 3.4b). The smoothness of the skeleton violates the mediality requirement, which dictates that the skeleton must be jagged to accommodate boundary perturbations. In the final analysis, the geometric characteristics the skeleton should preserve are dependent on the nature of the application. An algorithm such as the MAT [30], which possesses good reconstructive capability would be well suited for applications such as data compression [53]. However, its sensitivity to boundary perturbations means it is ill suited for object recognition where robustness to small shape distortions is required.

Figure 3.4 – The tradeoffs between (a) high mediality and reconstructability (taken from [66]) and (b) robustness to boundary noise (MGVF structural skeleton).

3.2 Structural and Textural Skeletons

The main applications addressed by this thesis are in the area of shape recognition and classification, especially in situations where there are substantial amounts of boundary perturbations and the shape to be analysed may undergo geometric distortions such as scale change and rotation. As such, the extracted skeleton should be mainly characterised by its insensitivity to arbitrary rotation and boundary noise. For this purpose, complete reconstructability in the skeleton such as that shown in Fig. 3.4a is not a desirable property. The myriad of runners resulting from the discretisation artifacts at the boundary will complicate shape analysis and are undesirable. Boundary discretisation noise is a common and typical boundary artifact created when a binary pattern represented on a discrete grid field (e.g. a digital image) is enlarged using a pixel-replication zoom algorithm.

Figure 3.5 – Structural and textural skeletons.

However, in order to preserve the general structure of the object's shape, it is important to ensure the skeleton of the object has the same simple connectivity as its shape, which means the skeleton must remain connected and loops around each hole in the object. In order to accommodate these requirements, two categories of skeletons are proposed in this work, they are the *structural* and *textural* skeletons. As the names imply, the *structural* skeletons describe the overall structure or topology of the object and the *textural* skeletons describe the boundary characteristics such as boundary texture or small protrusions. Fig. 3.5 illustrates these two categories of skeleton for a locker key. Notice the structural skeleton maintains simple connectivity of the shape, including the skeletal loop around the enclosed hole, whilst the textural skeletons capture the prominent boundary artifacts present in the shape, such as the serrated teeth of the key.

The structural and textural skeletons must remain disconnected so as to facilitate independent skeleton-based analysis of the object's gross visual form and its more subtly varying boundary perturbations, respectively. As an example, Fig. 3.6 shows four shapes that are by strict definition, different from each other. However, one would under most circumstances regard these objects as belonging to the same shape class since they share the same triangular form, despite their varying boundary perturbations. Notice how the structural skeleton of each shape is almost identical to the other. This requirement for disconnected structural and textural skeletons implies that the skeletonization process adopted for extracting such skeletons should not seek the preservation of skeletal connectivity at all cost. This criterion differs radically from the majority of the skeletonization algorithms reviewed in sections 3.1.1 and 3.1.2. Sections 3.3.3 and 3.3.4 discuss these issues further and describe algorithms for extracting these two categories of skeletons.

Figure 3.6 – Structural and textural analysis of shapes. (a) Four triangular-shaped objects and (b) their respective structural and textural skeletons. Gross classification of shapes can be made by analysing their structural skeletons. While the analysis of the varying characteristics of their textural skeletons will allow finer comparison to be made between similar looking objects.

3.3 Robust Shape Axes Extraction from Vector Fields

The work presented in chapter 2 describes how the Gradient Vector Field pyramid is obtained within a multiresolution pyramidal framework. This section extends this pyramidal framework further to enable the extraction of a medial representation from the MGVF. The medial loci extracted by the proposed technique are termed *shape axes* to differentiate them from the medial axes. This is to highlight the fact that the medial representation extracted from the MGVF does not always conform to Blum's locus of centres of maximal disk [30]. Further elaboration is given in section 3.4.1. Both the terms *shape axes* and *skeletons* are used to describe the medial representation extracted from the MGVF. The term *skeleton* is used more specifically to describe the medial representation that has been thinned to a unit-width. The term *shape axes* refers to the medial ridges (not necessarily of unit-width) extracted from the MGVF. This section describes how the shape axes can be extracted from the MGVF in a scale and rotation invariant manner. In order to obtain disconnected skeletons (see Fig. 1.2), the proposed skeletonization algorithm does not enforce connectivity but issues of maintaining connectivity in the structural skeleton are discussed in section 3.3.3.

3.3.1 Vector Field Disparity Map

Given the gradient vector field $V_l(x,y)$ in (2.5), the shape axes can be located by detecting locations in the vector field where the local gradient vectors exhibit high directional disparity. This is similar in concept to the technique proposed in [202], where Siddiqi *et al.* used the measure of average outward flux over a very small neighborhood to detect medial points. In this work, the idea of Ben-Arie and Wang in [20] is extended to give an alternative local directional disparity operator Δ given by

$$\Delta(\mathbf{v}(\mathbf{x}), \sigma_D) = \frac{g_{\sigma_D}(\mathbf{x}) * \|\mathbf{v}(\mathbf{x})\| - \|g_{\sigma_D}(\mathbf{x}) * \mathbf{v}(\mathbf{x})\|}{g_{\sigma_D}(\mathbf{x}) * \|\mathbf{v}(\mathbf{x})\|} \quad (3.6)$$

where $\mathbf{x} = (x,y)$ and the normalised local disparity measure $\Delta(\mathbf{v}(\mathbf{x}), \sigma_D) \in [0,1]$ is computed within a weighted locality defined by the Gaussian convolution kernel $g_{\sigma_D}(\mathbf{x})$ given by

$$g_{\sigma_D}(x, y) = \frac{1}{\sqrt{2\pi}\sigma_D} \exp\left(-\frac{x^2 + y^2}{2\sigma_D^2}\right) \quad (3.7)$$

As illustrated in Fig. 3.7a, the local directional disparity operator Δ gives a value close to 1 at the centre of a converging gradient vector field, such as one obtained at the centre of a circle and a value close to 0 in a locality where the gradient vector field is unidirectional, as shown in Fig. 3.7b.

$$\text{Local disparity measure, } \Delta(\mathbf{v}(\mathbf{x}), \sigma_D) = \frac{\overbrace{g_{\sigma_D}(\mathbf{x}) * \|\mathbf{v}(\mathbf{x})\|}^{\text{Scalar magnitude (s)}} - \overbrace{\|g_{\sigma_D}(\mathbf{x}) * \mathbf{v}(\mathbf{x})\|}^{\text{Vector magnitude (v)}}}{g_{\sigma_D}(\mathbf{x}) * \|\mathbf{v}(\mathbf{x})\|}$$

$v \rightarrow 0$, therefore $\Delta \rightarrow 1$

(a)

$v \rightarrow s$, therefore $\Delta \rightarrow 0$

(b)

Figure 3.7 – The behaviour of the local directional disparity operator Δ in different types of gradient vector fields. (a) A converging vector field, $\Delta \rightarrow 1$ and (b) in a unidirectional vector field, $\Delta \rightarrow 0$.

By applying the local directional disparity operator Δ to every level of the Gradient Vector Field pyramid $V(l, x, y)$ we obtain the Vector Field Disparity pyramid $D(l, x, y)$, where each level D_l is formally given by

$$D_l(x, y) = \Delta(V_l(x, y), \sigma_D) \quad \text{for } 0 \leq l \leq N \quad (3.8)$$

Fig. 3.8 shows three shapes and the vector field disparity measures D_0 , D_1 and D_2 computed from the respective levels V_0 , V_1 and V_2 of the Gradient Vector Field pyramid. The size of the Gaussian neighbourhood of the operator Δ , as defined by the value σ_D , remains constant for all pyramid levels.

Figure 3.8 – The gradient vector fields (topmost) of three shapes and their associated local vector field disparity measures computed from three consecutive levels of the Gradient Vector Field pyramid. Results were obtained using $\alpha = 0.8$ and $\sigma_D = 1.5$.

Note: Larger disparity measures are shown in darker shades of gray.

It is obvious from Fig. 3.8 that using a constant value of σ_D at all levels of the pyramid produces varying disparity measures that depend on the size and curvature of the shape boundary. For example, the disparity measures at the medial points of the rabbit's foot are significant at lower levels of the disparity pyramid but become insignificant at higher levels. On the contrary, the disparity measures at the medial points of the rabbit's head become more significant as one moves up the pyramid. This multiscale nature of the vector field disparity measure is similar to the cores proposed by Pizer *et al.* in [169]. In their work, they proposed extracting relevant cores of shapes and their subparts of varying width by tracking optimal scale ridges over a 3D scale-space volume. This approach is computationally costly and requires large amounts of memory if fine scale resolutions are required. The question here is whether it is possible to extract all relevant shape axes from only one level, that is the highest resolution level D_0 of the Vector Field Disparity pyramid?

Fig. 3.9a shows the disparity measure at level D_0 of an image containing three rabbit shapes with sizes spanning scales of two octaves. Fig. 3.9b shows the local maxima extracted from D_0 using a simple local ridge detection algorithm, which flags out a pixel as a ridge pixel when a specified number, N_D of its neighbouring pixels have disparity values less than its own. In this case, a 5×5 square neighbourhood is used and at least $N_D = 14$ pixels must have disparity values less than the centre pixel to classify it as a ridge point. Notice that the extracted shape axes are not scale-invariant.

Figure 3.9 – Inconsistent shape axes extracted from a single level of the vector field disparity pyramid. (a) Disparity measure at level D_0 . (b) The local maxima ridges extracted from the D_0 . Results were obtained using $\alpha = 0.8$, $\sigma_D = 1.5$ and $N_D = 14$.

In order to detect consistent shape axes over different scales of a shape, a full resolution vector field disparity map $M(x,y)$ is obtained by iteratively applying *EXPAND* to the sum of D_l and *EXPAND* [D_{l+1}], starting at $l=N-1$ to $l=0$. This multi-level integration process is illustrated in Fig. 3.10. The final summed output is divided by $N+1$ to re-normalise the disparity map such that $M(x,y) \in [0,1]$. More formally, the vector field disparity map is defined as

$$M(x,y) = \frac{1}{N+1} \sum_{l=0}^N d_l(x,y) \quad (3.9)$$

where $d_l(x,y) = D_l(x,y)$ for $l = N$

$$d_l(x,y) = D_l(x,y) + \text{EXPAND}[d_{l+1}(x,y)] \quad \text{for } 0 \leq l < N \quad (3.10)$$

Figure 3.10 – A multi-level integration technique for generating a vector field disparity map $M(x,y)$.

The integration of all levels of the Vector Field Disparity pyramid $D(l,x,y)$ into one full resolution disparity map $M(x,y)$ allows simultaneous extraction of all relevant shape axes across different scales from a single 2D spatial vector map. The integration of what are essentially multiscale features spread over the discrete pyramidal scale-space into a single 2D spatial image is possible because the medial features extracted, though relevant at different scales, are not spatially co-located. This is true of the local symmetry set defined by the medial locus of Blum [30]. Compared to the results in Fig. 3.9b, the shapes axes in Fig. 3.11b that were extracted from the multi-level integrated disparity map in Fig. 3.11a are significantly more scale invariant.

Figure 3.11 – Scale-invariant shape axes from the disparity map. (a) Disparity map $M(x,y)$. (b) The local maxima ridges extracted from the $M(x,y)$ using $\alpha=0.8$, $\sigma_D=1.5$ and $N_D=14$. Ridges at the boundary of the ($\times 2$) rabbit are due to image zooming discretisation artifacts. Such artifacts can be considered boundary texture.

3.3.2 Shape Axes Extraction from the Disparity Map

Given a full resolution disparity map $M(x,y)$, the shape axes can be extracted by locating the local maximas in the disparity measure using a local ridge detection algorithm. However, as can be observed in Fig. 3.11b, shape axes near bifurcation points were not properly extracted. This is because the ridges in the interior of the shape have been significantly blurred due to the multiple *EXPAND* operations used to derive $M(x,y)$. This poses difficulty in the ridge extraction process. A more robust method of extracting shape axes from the disparity map is proposed. The first step involves computing the derivative of $M(x,y)$ given by

$$\mathbf{M}_V(x,y) = \nabla g_{\sigma_V} * M(x,y) \quad (3.11)$$

where ∇g_{σ_V} produces first-order Gaussian derivative convolution kernels with a standard deviation of σ_V , and the resulting disparity gradient map $\mathbf{M}_V$ is a vector consisting of the x and y derivatives of the disparity map given by M_V^x and M_V^y respectively. The vector directional disparity operator Δ in (3.6) is then applied to $\mathbf{M}_V$ to obtain an enhanced vector field disparity map $M_E(x,y)$ that is shown in Fig. 3.12b and is defined by

$$M_E(x,y) = \Delta(\mathbf{M}_V, \sigma_E) \quad (3.12)$$

where σ_E determines the size of the effective neighbourhood in which the vector disparity is computed. Unfortunately, the vector disparity operator Δ does not differentiate between ridges and valleys. As a result, both are accentuated. Ridge and valley points in the disparity map $M(x,y)$ can be readily differentiated by observing the characteristics of the gradient vectors defined in $\mathbf{M}_V$. Vectors around ridge points are outward flowing and those around valleys are inward flowing. As such, a landscape map $M_L(x,y)$ that highlights ‘ridgeness’ and ‘valleyness’ in the disparity map can be constructed by finding the sum of the resulting two convolutions of each x and y gradient component of the disparity map $M(x,y)$, with their respective x and y first-order Gaussian derivative kernels defined in (2.4), and is given by

$$M_L(x,y) = \sum_{i=x,y} \nabla g_{\sigma_R}^i(x,y) * M_V^i(x,y) \quad (3.13)$$

$M_L(x,y)$ gives large positive values at ridge points and large negative values at valley points. A ridge-enhanced disparity map $M_R(x,y)$ can now be created using

$$M_R(x,y) = M_E(x,y) M_L(x,y) \quad (3.14)$$

The local maximas (see Fig. 3.12c) on the ridge-enhanced map $M_R(x,y)$ are detected using a simple local ridge detection algorithm, which flags out a pixel as a ridge pixel when a specified number, $N_D=14$ of its neighbouring pixels (in a 5×5 neighbourhood) have disparity values less than its own. Observe that this method of ridge detection does not enforce connectivity between the shape axes (e.g. rabbit's leg and body). As discussed in section 3.2, this disconnectivity between structural and textural medial information is desirable and it is a natural outcome of the MGVF-based disparity map and the local maxima ridge extraction process used.

Standard disparity-weighted morphological thinning is applied to thin extracted ridges to unit-width skeletons as shown in Fig. 3.12d. In this process, connectedness within the skeletal segment is maintained while ridge pixels are progressively eroded from outside-in. Removal preference is given to pixels with lower disparity values, thus ensuring the single-width skeletons stay as close to the axes of maximum disparity as discretely possible.

Figure 3.12 – The various stages of extracting MGVF skeleton extraction from the disparity map. (a) Disparity map $M(x,y)$. (b) Enhanced disparity map $M_E(x,y)$. (c) Ridge extracted using a local ridge detection algorithm. (d) Shape axes thinned to unit-width skeletons. Results from three other skeletonization algorithms serve to highlight the comparatively scale-invariant nature of the proposed MGVF-based skeleton extraction process.

As mentioned earlier in section 3.1.5, complete skeletal connectivity is not the goal of the proposed skeletonization process as it seeks to establish disconnectivity between the structural and textural skeletons. However, complete connectivity is

sought within the structural skeleton. These conflicting requirements pose several challenges to the skeletonization process and will be discussed next.

3.3.3 Maintaining Connectivity of Structural Skeletons

The process of ridge detection in the disparity map $M(x,y)$ employs operations that are simple (local maxima detection) and computed within a small local spatial support. This helps reduce the computational requirements of the ridge extraction algorithm. Unfortunately, such simple and local analysis cannot maintain connectivity across saddle and cross ridges such as those illustrated in Fig. 3.13. The cross-shaped skeleton of the plane (F4 Tomcat) forms the structural skeleton and as such, it should be completely connected.

Figure 3.13 – Broken structural skeleton connectivity due to the presence of saddle ridge surfaces. (a) The vector field disparity map $M(x,y)$ and (b) a zoomed-in 3D mesh plot of the disparity surface showing the saddle ridges. (c) Skeleton extracted by detecting local maximas in the disparity surface. Disconnectivity in the skeleton at localities of broad saddle ridges. (d) The connected skeleton after applying the directional-uphill climbing algorithm to end points P_A and P_B .

In order to connect broken skeletal segments within the structural skeleton, a “hill climbing” algorithm is used. There are numerous such algorithms in the literature [7], [66], [154], [196] but they are essentially applied to distance transformed surfaces. The *directional-uphill climbing* (DUC) algorithm proposed here is a modified version of that proposed by Shih and Pu [196], which was originally used in maxima tracking on Euclidean distance transform. Its main characteristics are that it operates on a small 3×3 square neighbourhood, it employs directional information of the current skeletal end points and it adds skeletal pixels to the current end point by climbing up the slope with the steepest gradient.

It is first assumed that all local maxima ridge points within the object have been extracted and thinned to a single-pixel width (see section 3.3.2). All skeletal segment end points are then detected and marked for possible uphill climbing (end points in contact with the shape boundary are excluded). Typical scenarios of skeletal end points are shown in Fig. 3.14. DUC is always initiated at the current end point with the lowest disparity value $M(P)$, where $M()$ is given in (3.9). This ensures skeletal connectivity in situations depicted in Fig. 3.14a, is achieved by growing the skeleton from an end point with a lower disparity value to one with a higher disparity value.

Figure 3.14 – Examples of end points of skeletal segments. (a) An example where there are two end points P_A and P_B that should be connected. Uphill climbing is initiated from the end point with the lower disparity value (i.e. P_A). (b) An example of a single end point P_C that should be connected to a continuous skeletal segment.

The *directional-neighbourhood* of end point P , given by $D_P = \{P_i\}$ is a set of points within the immediate 3×3 pixel neighbourhood N_P of P that are located within $\pm 45^\circ$ slope changes from the current skeletal orientation within N_P . Figs. 3.15b-e show examples of directional-neighbours of various skeletal end point orientations.

The numbering of the neighbourhood pixels adopted here is $P_1, P_2, \dots$, and P_8 starting from the positive x -axis of P and going around clockwise, as shown in Fig. 3.15a.

Figure 3.15 – The directional neighbourhood of end point P . (a) The pixel numbering scheme adopted. (b)-(e) Examples of directional-neighbours D_P for end point neighbourhood with different skeletal orientations.

Fig. 3.16 shows an example of an end point neighbourhood N_P that has three skeletal pixels. In such a neighbourhood, the directional-neighbourhood D_P of P has four elements given by the union of directional-neighbours associated with the two possible skeletal orientations form between end point P and one of two other remaining skeletal pixels in N_P .

Figure 3.16 – Directional-neighbours of an end point neighbourhood that has three skeletal pixels.

The directional-uphill climbing algorithm appends a skeletal point to the current end point P by determining which of the directional-neighbours has the steepest disparity gradient from P . This appended skeletal pixel forms the next end point P_{next} , whose selection is given by

$$P_{next} = \arg \max_{P_i \in D_P} \left\{ \frac{M(P_i) - M(P)}{d_i} \right\} \quad (3.15)$$

where

$$\begin{aligned} d_i &= 1 & \text{for } i &= \text{odd} \\ d_i &= \sqrt{2} & \text{for } i &= \text{even} \end{aligned} \quad (3.16)$$

The value of the inter-pixel Euclidean distance d_i varies due to the use of a square-gridded neighbourhood. If the North, South, East and West pixels in N_P are considered a unit-distance from P , then the other four diagonal neighbours and P are $\sqrt{2}$ apart and must therefore be appropriately compensated when computing the disparity gradient between P and all points in D_P . The DUC algorithm is subsequently applied to P_{next} in an iterative manner until the current neighbourhood N_P contains a skeletal pixel belonging to another skeleton segment. Once this occurs, the two previously disjointed skeletal segments are considered connected. Fig. 3.13d shows the result of applying the DUC algorithm to the two end points P_A and P_B of a plane.

3.3.4 Determining Structural and Textural Skeleton End Points

The connection of broken skeletal segments give rise to another problem. If disconnected segments of structural skeletons were to be reconnected, a good question would be “When is a skeletal segment part of the structural skeleton and when is it a textural skeleton?” This distinction is important because DUC should not be applied to the end points of textural skeletons, as they should remain disconnected from the structural skeleton. In the cases of Fig. 3.17a and 3.17b, this distinction is obvious. However, the protruding flange in Fig. 3.17c looks prominent enough to be considered a part of the overall shape structure but its skeleton remains disconnected after local maxima detection. Should connectivity be applied? This section discusses the criterion used in deciding if a given end point should be connected to the structural skeleton or whether it should remain unconnected.

Figure 3.17 – When is a skeleton a structural or textural skeleton? Shapes in (a) and (b) have obvious structural and textural skeletons. In the case of (b), the narrow protruding flange is considered a “textured” add-on to the gross structure of the object, which is essentially a rectangle. (c) An ambiguous case.

As shown in Fig. 3.18, the object is defined by a set of 2D boundary points $B = \{b_i\}_{i=1}^N$ and there is a set of 2D skeletal end points $P = \{p_j\}_{j=1}^M$ detected within the interior of the object. To determine if the direction-uphill climbing algorithm should be applied to any one of the end point p_j , a circle c_j with radius r_j is first obtained, where r_j is given by

$$r_j = \lambda R_j \quad (3.17)$$

where

$$R_j = \min_{1 \leq i \leq N} \left\{ \left\| b_i - p_j \right\| \right\} \quad (3.18)$$

and $\| \cdot \|$ gives the Euclidean distance between two points.

Figure 3.18 – The λ -maximal disk of connectivity centred on end point p_j . Directional-uphill climbing is applied at p_j if there exist another skeletal segment S_k other than the skeletal segment S_j belonging to end point p_j , within the λ -maximal disk region (shaded) defined by the radius $r_j = \lambda R_j$.

The circle c_j is termed the λ -maximal disk of p_j since its radius r_j is related to R_j (the radius of the maximal disk centred about end point p_j) through the factor λ . The parameter $\lambda \in (0, 1]$ is called the *connectivity factor* and is used to control the size of the λ -maximal disk, which is at its maximum size when $\lambda = 1$. Directional-uphill climbing is only applied to end point p_j if there exist another skeletal segment within the region enclosed by the λ -maximal disk other than its own skeletal segment S_j . Based on this criterion, connectivity factor λ , which governs the disk's diameter, has direct influence over the sensitivity of the process that recruits disjointed skeletal segments to form part of the structural skeleton.

After connectivity has been achieved by the directional-uphill climbing iteration started at end point p_j , a further check is then carried out to confirm that the point of intersection between the two skeletal segments lies within the λ -maximal disk of p_j . If it does, the skeletal segment S_j is now part of the structural skeleton, as shown in Fig. 3.19a. If it lies outside the λ -maximal disk, as depicted in the example in Fig. 3.19b, the skeletal segment S_j of end point p_j will remain unconnected. Such examples do arise because the disjointed skeletal segment S_k found within the λ -maximal disk actually belongs to another nearby textural skeleton. There is no sure way of determining this fact until after directional-uphill climbing has been applied to end point p_j to determine the true location of the connectivity intersection based on the magnitude profile of the vector field disparity map.

Figure 3.19 – A further check is done after connectivity of end point p_j to the structural skeleton. (a) A successful connectivity is obtained when the intersection point p_{jk} between the two skeletal fragments S_j and S_k lie within the λ -maximal disk. (b) A prohibited connectivity occurs in this case because the intersection point p_{jk} lies outside the λ -maximal disk. In both these examples, the value of $\lambda = 1$.

The ability to control the differentiation sensitivity between structural and textural skeletons is useful for applications such as shape classification, where it is the gross shape of the object that determines which class the object belongs to and not characteristics such as boundary texture or minor protrusions. As shown in Fig. 3.20, both the smoothing parameter α in (2.5) and the connectivity factor λ in (3.17) must be used in combination to regulate this sensitivity. Small values of $\alpha = 0.6$ and $\lambda = 0.6$ can prevent even large boundary protrusions such as the broad right flange from being

connected to the structural skeleton (see 3.20d). On the other hand, when both α and λ values are large (i.e. $\alpha = 0.95$ and $\lambda = 1.0$), even the narrow left flange can be made part of the structural skeleton (see Fig. 3.20a). Also notice in the two pre-connected skeletons at the top, that increasing the smoothing parameter α brings the end points of the textural skeletons closer to the structural skeleton. This increases the likelihood of a connection even if a small connectivity factor λ is used. Small values of α results in a poorly defined gradient vector field as most of the high spatial frequency components that enable one to distinguish between a sharp and rounded corner, for example, are significantly attenuated. Any form of shape analysis that uses the information embedded within such a smooth MGVF will lose its discriminative capabilities. It is therefore prudent to use a combination of both the α value and the λ value to obtain the appropriate differentiation sensitivity required to distinguish between structural and textural skeletons. Values for α and λ used in the experiments presented in section 3.4 are obtained empirically.

Figure 3.20 – Controlling the differentiation sensitivity between structural and textural skeletons using the combination of the two parameters α and λ . The top two images show the pixel-width thinned skeletons extracted from MGVFs of $\alpha = 0.95$ and $\alpha = 0.60$. Results of end point connectivity analysis applied to the skeleton of (a) $\alpha = 0.95$, $\lambda = 1.0$; (b) $\alpha = 0.95$, $\lambda = 0.6$; (c) $\alpha = 0.60$, $\lambda = 1.0$ and (d) $\alpha = 0.60$, $\lambda = 0.6$.

3.4 Experimental Results and Discussion

In this section, several experiments are presented to discuss the characteristics of the skeleton extracted from the MGVF. The proposed algorithm for extracting the structural and textural skeletons will be called the MGVF algorithm. In order to compare the performance of the proposed MGVF algorithm, two other skeletonization algorithms were implemented; one uses an iterative approach and the other, a non-iterative approach. For the iterative skeletonization algorithm, the popular two-pass parallel thinning algorithm by Zhang and Suen [230] was implemented with appropriate improvements suggested by Lu and Wang [141] incorporated. This will be called the Zhang & Suen algorithm. For the non-iterative algorithm, the skeleton is extracted from the distance transform of the binary image. A more computationally demanding square Euclidean distance metric defined as $dist_E$ in (3.5) is adopted in the hope of achieving the best results. To reduce the computational requirements, the distance transform computation was confined to pixels representing the binary shape only. The local ridges in the distance-transformed surface are extracted and thinned in the same manner as that described for the proposed MGVF algorithm. In the former, the surface is defined by distance map D given in (3.1) and in the latter by the disparity map M given in (3.9). In some cases, the end point connectivity analysis described in section 3.3.4 is similarly adopted to connect broken ridges extracted from the Euclidean distance transformed (EDT) image. This algorithm will be called the EDT algorithm. In several experiments, skeletonization results obtained from the readily available binary morphological function *bwmorph()* provided by the Matlab® Image Processing Toolbox [145] were also presented to provide additional comparative performance measures and give the readers some sense of the limitations of the native skeletonization function provided by Matlab® version 5.3.

3.4.1 Effects of Varying Smoothing Parameter α

The smoothing parameter α in (2.5) has a significant influence on the characteristics of the extracted skeleton. With a small value of $\alpha = 0.5$, the MGVF generated contains a higher proportion of lower spatial frequency components of the shape. As a result, high frequency boundary artifacts such as the serrations on the key shape in Fig. 3.21a

are de-emphasised (i.e. textural skeletons are fewer and shorter). As α is made larger (e.g. $\alpha = 0.9$), even minor serrations are now noticed.

Figure 3.21 – Skeletons extracted from a key shape using (a) MGVF algorithm with $\alpha = (0.50, 0.90, 0.99)$ and (b) the EDT algorithm.

Fig. 3.21b shows the medial axes extracted from the Euclidean Distance Transform (EDT) image of the key shape. The extracted EDT medial axis is a good approximation to a subset of the medial loci defined by Blum's medial axes [30], where the locus is defined by the centre of all maximal disks in the interior of the shape that contact the boundary in two or more separate positions. As can be observed in Fig. 3.21a when $\alpha = 0.50$ and $\alpha = 0.90$, the MGVF algorithm does not produce skeletons that conform to the medial locus as defined by Blum. This is due to the non-trivial combination of high and low spatial frequency gradient information at the smaller inner boundary (hole in the key) and the larger outer boundary during the generation of the MGVF. The outer boundary, being spatially more extensive, has a more dominant low frequency component than the smaller inner boundary. It is observed that the MGVF's disparity maximas shift towards the smaller inner boundary when low frequencies are emphasised with a low value of α . Conformity to the true medial loci improves as $\alpha \rightarrow 1$, as can be observed when $\alpha = 0.99$.

This α -dependent variability of the structural skeleton's position is less significant in shapes that have no holes since no inner-outer boundary interaction exist. However, if these shapes are sufficiently narrow, like the sine-modulated elongated bars depicted in Fig. 3.22, the structural skeleton can still be significantly influenced by the frequency characteristics of the boundary contour. Notice how in

Fig. 3.22a, the structural skeleton in the interior of the shape begins to undulate along with the boundary modulation when a high value of $\alpha = 0.9$ was used to preserve more of the boundary's high frequency components. Alternatively, low values of $\alpha = 0.5$ allow these structural skeletons to 'straighten out'. Morse *et al.* [152] presented similar experiments to show how the boundary modulation amplitude and the shape's width influenced the 'straightness' of the cores extracted. In the case of the MGVF-based skeleton, an additional smoothing parameter α could be used to regulate the influence the boundary perturbations have on the structural skeleton.

Figure 3.22 – Skeletons extracted from sine-modulated bars using the MGVF algorithm with $\alpha = (0.5 \text{ and } 0.9)$ at sine modulation frequencies of (a) f and (b) $1.3f$.

The characteristics textural skeletons extracted at the boundary depend on both the frequency of the boundary modulation as well as the value of α . It is possible to extract no textural skeleton if both the frequency of the boundary modulation and α is sufficiently low, as can be seen in Fig 3.22a when $\alpha = 0.5$. On the other hand, when both the boundary modulation frequency and α were higher, the textural skeleton extracted were significantly more elongated (see Fig 3.22b when $\alpha = 0.9$). In summary, the experiments presented in this section have highlighted the various influences the smoothing parameter α has on the characteristics of the extracted structural and textural skeletons.

3.4.2 Scale Invariant Skeletons

Pizer *et al.* in [169] proposed describing shapes using cores, which are the set of local maxima in the medialness function of shapes that spans various scales. Computing the cores is costly, as it requires the tracking of optimal scale ridges over a 3-D scale-space volume. Fig. 3.23a shows the cores extracted using the techniques of Pizer *et al.* in [169]. In contrast, by applying the proposed multi-level integration technique described in (3.9) to the Vector Field Disparity pyramid $D(l,x,y)$ in (3.8), the scale-varying local maxima in the vector field disparity can now be extracted from a single full resolution disparity map $M(x,y)$. The resulting medial loci of the MGVF skeletons (see Fig. 3.23b and Fig. 3.23c) are similar to the cores but extracted in a much more computationally efficient manner.

Figure 3.23 – The medial loci of a sawtoothed rectangle. (a) The cores (white) representation taken from [152]. (b) The local disparity maxima (black) obtained with $\alpha = 0.8$ and (c) its single-pixel width thinned skeleton.

The true test of scale invariance for a skeletonization algorithm is its ability to extract consistent skeletons from a shape that varies in size. Fig. 3.24 shows that the MGVF algorithm is able to extract consistent structural skeletons with scale changes spanning two octaves. Textural skeletons, on the other hand, became more numerous with increasing scale. This was due to the increasing prominence of the boundary discretisation noise resulting from the image zooming algorithm. This is not necessarily a shortcoming of the MGVF algorithm as the textural skeletons rightfully represent the increasing visible boundary texture. Such discrepancy does not produce problems for scale invariant analysis since the scale-invariant structural skeleton

remains disconnected from the textural skeletons. This however, cannot be said of the skeletons produced by the Zhang & Suen algorithm in Fig. 3.25a and the EDT algorithm in Fig. 3.25b. Notice as the shapes were scaled up, there were increasing numbers of skeletal branches running from the boundary of the hawk's wings to the main skeleton in the interior. In this case, scale invariant analysis would require some additional processes such as branch pruning.

Figure 3.24 – Scale-invariant skeletons from the MGVF algorithm. (a) Maple leaf shapes, from [1]. (b) Hawk shapes, from [195]. Images are 257×257 pixels and the scaling factors are indicated. Results were obtained with $\alpha = 0.8$ and $\lambda = 0.7$.

Figure 3.25 – Scale-variant skeletons from the (a) Zhang and Suen's parallel thinning algorithm [230] and (b) the Euclidean distance transform algorithm ($\lambda = 0.7$).

3.4.3 Rotation-Invariant Skeletons

In order to test the rotation invariance of the various skeletonization algorithms, three test images shown in Fig. 3.26 were used. The one shown in Fig. 3.26c is rotated by a multiple of $\pi/2$ and the one in Fig. 3.26b is rotated by an odd angle of 30° .

Figure 3.26 – Test images for rotation invariance experiments. (a) Original image, (b) rotated 30° and (c) rotated 45° . Image rotation was done with the help of the *imrotate()* function available in the Matlab® Image Processing Toolbox [145].

The rotation invariance of the MGVF algorithm is clearly observed in the results shown in Fig. 3.27a. Since all convolution operations used to generate the multiresolution gradient vector field and vector field disparity maps are either Gaussian or derivative of the Gaussian in nature, the transformation applied to the shape boundary is completely isotropic. This invariably results in skeletons that are invariant to arbitrary rotation about the image plane.

Good isotropic behaviour or invariance under rotation is difficult to achieve in iterative algorithms such as those reviewed in [121]. In sequential algorithms, the resulting skeleton depends on the order in which the pixels are processed. In the case of parallel algorithms that remove border points in each subiteration, the result is dependent on the order of the subiteration [121]. Parallel thinning algorithms such as the MB2 algorithm [23] claim that rotation invariance can only be guarantee for angles of multiples of $\pi/2$ because they are 4-isotropic, this means the thinning action is exactly the same in four cardinal directions (N, E, S, W). Such limitations, also shared by Zhang & Suen's iterative parallel thinning algorithm are clearly seen in the results of Fig. 3.27b. Notice that the skeleton extracted from the shape that was rotate by an odd 30° is very different from the other two angles that are $\pi/4$ multiples apart.

Skeletons extracted from a distance transform are only rotationally invariant if the distance metric $dist_M$ in (3.1) is truly Euclidean in nature. This means the more computationally efficient city block and chessboard distance are not invariant for arbitrary angles. As for the more computationally demanding squared Euclidean distance transform used in the EDT algorithm shown in Fig. 3.27c, complete rotation invariance is achieved.

Finally, to illustrate the results obtained by a typical commercial skeletonization routine such as that provided by the Matlab® Image Processing Toolbox, the morphological skeletonization operator *bwmorph()* was applied to the images in Fig. 3.26. It produced the least consistent result, as can be seen in Fig. 3.27d.

Figure 3.27 – Results of the rotation invariance experiments. Skeletons extracted from the (a) MGVF algorithm, (b) Zhang and Suen parallel thinning algorithm, (c) Euclidean distance transform algorithm and (d) Matlab® *bwmorph()* function using the ‘skel’ operator.

3.4.4 Analysis of Boundary Noise Sensitivity

Despite the large numbers of skeletonization algorithms developed in the past, there have not been many techniques proposed for the quantitative analysis of skeletons. Lam and Suen in [122] proposed three different measures of distance to evaluate the closeness of an extracted skeleton to a reference or ideal skeleton, namely mean-squared distance based on equispaced points, mean-squared projection distance and mean-squared distance based on string matching and dynamic programming. The experimental setup employed here is similar to that in [1] and the distance measure used is similar to the projection distance employed in [122]. The goal of these experiments is to evaluate the accuracy of the skeleton extracted from the MGVF in the presence of boundary noise. A known reference skeleton with N skeletal points is first extracted from an undistorted shape, which in this case is a square. The boundary of the square is then perturbed in the x and y directions by varying degrees of zero-mean Gaussian noise with varying standard deviation σ_n . For each test image generated, the Euclidean distance between each skeletal pixel in the reference skeleton and the closest skeletal pixel of the newly extracted skeleton is computed, resulting in a distance array x_i , where $1 \leq i \leq N$. The variance of the skeletal discrepancy [1] is given by

$$\text{var}(X) = E(X^2) - E^2(X) \quad (3.19)$$

where

$$E(X) = \frac{1}{N-1} \sum_{i=1}^N x_i \quad (3.20)$$

and

$$E(X^2) = \frac{1}{N-1} \sum_{i=1}^N x_i^2 \quad (3.21)$$

$E(X)$ is the mean Euclidean distance and $E(X^2)$ is the mean-squared Euclidean distance. The standard deviation of the skeletal discrepancy, SD is given by

$$SD = \sqrt{\text{var}(X)} \quad (3.22)$$

Fig. 3.28 shows the variations in the mean value of SD for $0 \leq \sigma_n \leq 4$ for skeletons extracted using the MGVF and the EDT algorithms over 100 runs of the experiment

and for each run, the value of σ_n was incremented in steps of 0.2. The same randomly generated noisy square was used for both the MGVF and EDT algorithms. The results show that the skeletal discrepancy of skeletons extracted using the proposed MGVF algorithm is consistently lower than those extracted using the EDT algorithm. Improvements were more pronounced at higher levels of boundary noise.

Figure 3.28 – The plots of standard deviation (SD) in the skeletal discrepancies of the MGVF and EDT extracted skeletons for a square shape undergoing varying amounts of boundary perturbations. The parameter settings for the MGVF and EDT algorithms were $(\alpha = 0.8, \lambda = 0.7)$ and $(\lambda = 0.7)$, respectively. Error bars drawn at each σ_n value is the standard deviation obtained after 100 runs and they are plotted about their respective mean values.

The boundary noise robustness of the skeletons extracted using the MGVF algorithm is due to the fact that the gradient vector field within the interior of the shape is derived mainly from the low spatial frequency components of the shape's boundary. The skeletal information embedded with the gradient vector field disparity is therefore less susceptible to high frequency distortions present in the noisy and jagged boundaries. This can be clearly observed from the relatively consistent structural skeletons extracted using the MGVF algorithm, even at noise levels of $\sigma_n = 4.0$, as shown in Fig. 3.29.

The Euclidean distance transform on the other hand, faithfully transmits the noisy boundary variations into the interior of the shape as can be seen in the distance maps of Fig. 3.29. This inadvertently distorts the position of the skeletons. Moreover, many extraneous skeletal branches were extracted, which can make subsequent skeletal analysis of the noisy shape complicated. In contrast, the MGVF skeletons have a consistent structural skeleton that represents the basic shape of the square. At the same time, the noisier boundaries are also noted by the increasingly more prominent textural skeletons. Since these two categories of skeletons remain unconnected, this makes the skeletal analysis of noisy shapes relatively straightforward.

Figure 3.29 – Skeletons extracted from the disparity map and distance map using the MGVF and EDT algorithms, respectively. Results shown are sampled for images with boundary noise levels of $\sigma_n = 0, 0.4, 2.0$ and 4.0 . The standard deviations SD of each extracted skeleton from their respective reference skeleton are also indicated. The skeletons at $\sigma_n = 0$ are the reference skeletons. Results presented here were obtained using the same parameter settings as those used in the experiment in Fig. 3.28.

3.4.5 Robustness to Boundary Texture

As discussed earlier, one of the strengths of the proposed MGVF algorithm is its ability to control the proportion of low and high spatial frequency components used to create the gradient vector field. By choosing a lower value of α in (2.5), we can reduce the influence of boundary texture on the gradient vector field. This will facilitate the extraction of relatively consistent structural skeletons that represent the object's gross shape form despite the presence of various types of boundary texture. This can be observed in the boundary textured hares example in Fig. 3.30. What is unique about the MGVF-based skeletonization algorithm is the fact that the textural skeletons near the boundaries are simultaneously extracted together with the structural skeleton. Both these classes of skeletons remain unconnected, which makes possible the simultaneous skeletal analysis of both the object's general shape form and its boundary textural characteristics.

This approach is preferable to Kimia's use of curvature deformation using non-linear diffusion [111]. It is not possible to find one level of smoothing that can allow one to analyse both the gross shape form and the detailed boundary texture. If sufficient shape smoothing has been applied such that the general shape class can be visualised, all notion of boundary texture would have been removed.

Figure 3.30 – Skeletons extracted from four different boundary-textured hare shapes using the proposed MGVF algorithm with parameters $\alpha = 0.6$ and $\lambda = 0.7$. The hairy, wavy and spiky textured images were created from the original smooth hare image.

3.4.6 Example Application - Character Recognition

In optical character recognition (OCR) applications, textual symbols are first isolated and then thinned to a unit-width skeleton before feature extraction and analysis. We compare the performance of the MGVF algorithm with three other skeletonization algorithms when applied to this task. Fig. 3.31a shows a perfectly isolated euro symbol and Fig. 3.31b shows the same symbol but corrupted with typical boundary artifacts that are encountered by OCR systems.

Figure 3.31 – Character recognition test images. (a) A euro symbol with some boundary discretisation noise. (b) The same euro symbol corrupted with simulated line scratch, ink splotch and partial occlusion by an angular object.

Fig. 3.32b shows the skeleton extracted from the uncorrupted euro symbol using the EDT algorithm. Notice how boundary discretisation noise can sometimes result in spurious medial branches. These boundary discretisation artifacts also result in a slightly jagged skeleton when the Zhang and Suen [230] thinning algorithm is employed, as can be observed in the enlarged image segment at the bottom-left corner of Fig. 3.32c. In contrast, the incorporation of some low spatial frequency components into the MGVF resulted in a smoother skeleton (see Fig. 3.32a), much like those of Shroff and Ben-Arie [197]. Also included in the comparison, is the binary morphological function available in the Matlab® Image Processing Toolbox [145], which was applied to the same euro symbol. The results in Fig. 3.32d show that the *bwmorph()* skeletonization operator ('skel') is extremely sensitive to boundary discretisation noise and this results in many false skeletal branches.

Figure 3.32 – Extracted skeletons for the clean euro symbol using various skeletonization algorithms, namely (a) the proposed MGVF algorithm with $\alpha=0.9$ and $\lambda=0.7$, (b) Euclidean Distance Transform, (c) Zhang and Suen parallel thinning and (d) Matlab’s binary thinning algorithm.

Another useful performance measure is to observe how well each skeletonization algorithm is able to consistently replicate the extracted skeleton when the same symbol is now corrupted with various types of boundary artifacts. As expected, the EDT result in Fig. 3.33b shows extra medial branches where the angular occlusion is present and at some locations where the line scratches intersect the symbol. The ink splotch is particularly problematic to the connectivity-seeking thinning algorithm of Zhang & Suen [230] in Fig. 3.33c. The splatter tails around the splotch provide anchor points that lead the thinning algorithm to create many spurious branches in response to the need to maintain skeleton connectivity during the boundary erosion iterations. This same artifact can also be observed along the line scratch. Matlab’s *bwmorph* function suffers from essentially the same problem as can be seen in Fig. 3.33d. Despite the various distortions, the MGVF algorithm is still able to extract a smooth and consistent structural skeleton, as shown in Fig. 3.33a. These results support the case for a medial representation that does not enforce connectedness between medial branches originating at the boundary and those located in the interior of the shape. Minor boundary artifacts, such as the thin line scratch, should generate unconnected textural skeletons that remain close to the boundary. However, the skeletons of the two horizontal segments, which are actually parts of the euro symbol, should be part of the structural skeleton. Since their widths are

comparable with the ‘C’-shaped segment, the two horizontal skeletons do connect at their respective intersections with the ‘C’-shaped skeleton and together, they form a structural skeleton that correctly represents the euro symbol.

Figure 3.33 – Extracted skeletons for the corrupted euro symbol using the same four algorithms and parameter settings described in Fig. 3.32.

3.4.7 Computational Cost

The computation cost of the four skeletonization algorithms used in the previous section were compared using their respective Matlab implementations. Even though the compute times recorded do not necessarily indicate the best performance of each algorithm (since they are not optimised), it does give a rough appreciation of their relative computational efficiency. The only exception in this case is Matlab’s native binary thinning algorithm *bwmorph()* [145], which has been optimised through compilation. Compute time measurements were obtained for the skeletonization of two different 129×129 pixel-sized images and they are tabulated in Table 3.1. The two Eight and Hare images were appropriately selected to demonstrate how the maximum width of the shape affects the computational times of the algorithm.

As expected, the fastest computation times for both images were registered by Matlab’s own optimised *bwmorph()* algorithm. It was 8.5 times and 2.4 times faster than the MGVF-based algorithm for the Eight and Hare images respectively. Its speed advantage dropped significantly as the maximum width of the shape increased. This shape-width variability in skeletonization time is generally undesirable but it is an

inherent characteristic of morphologically-based thinning algorithms such as *bwmorph()* and Zhang and Suen [230] parallel thinning algorithms, which iteratively thin the given shape from the boundary inwards, much like the peeling of the layers of an onion. This variability is also true of the EDT algorithm as the number of distance transform computations depends on the number of pixels in the shape. The Zhang & Suen algorithm exhibited a 11-fold variation in compute time between the two images, compared to the EDT and MGVF algorithms' 2-fold and 1.5-fold variations respectively. In the case of the MGVF algorithm, the variation is mainly due the skeleton-dependent ridge thinning and connectivity analysis processes. The MGVF algorithm was consistently much faster than the EDT algorithm but may be slower than the Zhang & Suen's parallel thinning algorithm for shapes with narrow widths.

Technique	Time (sec.)	Results	Time (sec.)	Results
MGVF				
Generate vector field	1.04		1.04	
Thin and connect ridges	0.82		1.76	
Total compute time	1.86		2.80	
EDT				
Distance transform	3.84		7.52	
Thin ridges	0.49		1.21	
Total compute time	4.33		8.73	
Zhang and Suen				
Total compute time	0.71		7.91	
Matlab – <i>bwmorph</i>				
Total compute time	0.22		1.15	

Table 3.1 – Computational cost incurred in computing the skeletons of two different images (Eight and Hare) using the MGVF, EDT (without connectivity analysis), Zhang & Suen and Matlab® *bwmorph()* skeletonization algorithms. The times shown were measured on a 1.6 GHz Pentium-4 Personal Computer with 255 MB DRAM, running Windows 98® and Matlab® version 5.3. Matlab codes for building the MGVF's Gaussian pyramids were based on Simoncelli's pyramid tools available from [203] and settings of $\alpha = 0.9$ and $\lambda = 0.7$ were used to generate the results.

3.5 Summary

This chapter has described how a medial representation can be recovered from the MGVF. This medial representation is formed by the loci of local directional disparity maximas in the gradient vector field. It was shown how the extracted skeleton is invariant to both scaling and rotation. Since the proposed algorithm does not enforce connectivity during the process of ridge detection, two types of skeleton were obtained. They are the structural skeleton located mainly in the interior of the object and the textural skeletons that describe perturbations at the object's boundary. The advantages of this dual skeletal representation was highlighted by several experimental results which show that the MGVF-based structural skeletons can be extracted in a consistent manner despite the presence of boundary artifacts such as noise, texture, line scratches, etc. Skeletal analysis of shapes is further simplified by the fact that these structural skeletons, which describe the general shape of the object, remained largely disconnected from the numerous textural skeletons at the boundary.

Comparative studies of other popular skeletonization techniques such as the iterative parallel thinning algorithm of Zhang and Suen [230] and Euclidean distance transform-based ridge detection were conducted. Compared to the MGVF algorithm, these algorithms were unable to extract consistent skeletons under conditions where there were significant boundary perturbations. These distortions could arise from a variety of situations such as from boundary discretisation noise created by extensive binary image zooming or from a noisy region segmentation stage of a vision-based recognition system. The iterative thinning algorithm was also observed to be sensitive to rotation. The computational cost of the MGVF algorithm was also shown to be the least shape dimension dependent of all the algorithms evaluated. This is an asset, as any vision application adopting this algorithm will have a more predictable response time. Its total computational cost is lower than that of the EDT algorithm but it may be higher than Zhang & Suen algorithm for shapes with narrow width (e.g. alphanumeric characters).

The medial representation discussed in this chapter will be further exploited in chapter 5, where it will be described how these MGVF-based skeletons can be used as a framework to build a flexible and robust part-based shape descriptor.

Chapter 4

Shape Description using the Multiresolution Gradient Vector Field

Many image processing and analysis systems begin with a gray-scale input image of a scene containing the objects of interest. The recognition of such objects often requires the analysis of its shape. Such analysis can be done using a binary image representing the extent of the object and the shape seen as a silhouette of the object. There are many computer vision applications where the image analysis involves the analysis of shapes. These include content-based image retrieval (CBIR) systems that employ shape-based features such as QVE [94], Photobook [165] and QBIC [61], just to name a few. Other CBIR systems require the analysis of more complex shapes such as those found in trademark images [57], [103]. Shape analysis is also required in many object recognition applications involving handwritten characters, machine parts, traffic signs, etc. Besides recognition, some applications require objects to be classified into different categories based on their shape variations. An example is the classification of microscopic organisms called diatoms [92] into their different species groups as undertaken by the ADIAC project [39].

The main focus of this chapter is to demonstrate and evaluate the shape description capabilities of a global shape descriptor called the *orientation histogram*, which utilises the proposed MGVF as its basis for extracting robust shape-related information. The chapter begins with a review of various approaches to shape representation, and their relative strengths and weaknesses. Section 4.2 describes how the orientation histogram is derived from the MGVF and how it can be used to compute dissimilarity measures between shapes. Section 4.3 presents its performance in several shape analysis experiments and the chapter closes with a discussion of the strengths and limitations of the orientation histogram-based shape descriptor.

4.1 Shape Analysis – A Review

There are many algorithms to perform shape analysis [139]. However, before shapes can be analysed, they must first be reduced to a description that is amenable to efficient storage and analysis. These shape descriptors are usually in the form of a feature vector and can be derived in a variety of ways as reviewed here. Those descriptors used in the experiments to provide a comparative performance analysis are described in greater detail. Also reviewed are issues related to the process of computing the similarity between two shapes.

4.1.1 Shape Representation – Boundary versus Region

Most textbooks on computer vision that describe shape representations will usually categorise them into either boundary or region based approaches [12], [208]. Which then is the best way to represent shapes? Is it by its bounding contour or its interior region? Conceptually the two approaches are equivalent since the interior exists because of the enclosing boundary and vice versa. However, representations that adopt either the boundary or region approach alone can vary significantly in the way they encapsulate information about the shape.

Boundary representation - Representing shapes by its boundary is by far the more popular approach. In part, this could be reinforced by the works of the duo Hubel and Wiesel [99] who suggested that most cells in the visual cortex respond strongly only to orientated edges. This might suggest that pertinent visual information is only encoded in the shape's boundary. From a practical viewpoint, the computational complexity of boundary-based approaches is lower because boundary representations are 1D and therefore less costly to process than 2D region representations. In many ways, this advantage is also its chief downfall as the reduction of dimensionality loses important 2D spatial information about the shape.

Shapes can be described by their geometric boundary properties such as length, curvature, bending energy, signature and chord distribution [208]. A string-based representation called chain codes, first suggested by Freeman in 1961 [64] and its variants such as chain code derivative [217] and the less noise-sensitive smooth chain codes [131] can be used to create a form of grammar, which then transform the shape matching problem to one of string matching. Standard string matching

techniques such as string edit distance computation [40], syntactic matching [65] or dynamic programming [81] have been used to compare shapes. Shape boundaries can also be piecewise approximated using parametric models such as line segments or B-splines [102]. The polyline approximation allows the shape to be represented by 1D signatures such as turning function distance [9] or tangent space [124], which can then be efficiently used for computing distances between shapes [221].

Transform-based representations using shape boundaries such as the generalised Hough Transform (GHT) techniques [3], [49], [138] and [146] are useful because of their invariance to rotation, scaling and in some cases, higher-order deformation such as affine and perspective transformations. However, the major drawback of GHT-based approaches is their high computational requirements.

Another popular boundary-based representation is the Fourier descriptor (FD), first proposed by Granlund in the early 70's [79] and has since seen many variants [158], [179]. In the FD-based method, a shape is first represented by a feature function called the *shape signature*. A discrete Fourier transform is then applied to the signature $f(k)$ to obtain a Fourier descriptor $F(u)$ for the shape, which is given by

$$F(u) = \frac{1}{N} \sum_{k=0}^{N-1} f(k) \exp\left[-\frac{j2\pi uk}{N}\right] \quad (4.1)$$

where N is the number of samples in $f(k)$. Several types of shape signatures can be used and they may be boundary-based, curvature-based or radii-based [142]. Kauppinen *et al.* [108] showed that shape classification performances based on these types of signatures were comparable. However, Zhang and Lu's [232] more recent work on shape-based image retrieval concluded that the radii-based (centroid distance) signature yielded the best performance among the various shape signatures they have implemented.

Given a shape defined by a closed contour, a shape signature based on the centroid distance can be obtained. It consists of N ordered distances computed from the shape's centroid to N boundary points that are distributed over equidistant intervals on the shape boundary, as shown in Fig. 4.1a.

Figure 4.1 – Fourier descriptor extracted from a radii-based shape signature. (a) The shape signature obtained from the ordered distances between 64 equally spaced boundary points and the shape's centroid. (b) The log-plot of the Fourier descriptors for each of the respective shapes. The shape below is obtained from that above by a scale reduction of 70% and a clockwise rotation of 90°.

A rotation invariant FD is obtained by using only the magnitude component of $F(u)$ and scale-invariance is obtained by normalising the descriptor using the $|F(0)|$ component, which represent the overall energy of the shape radii. The final scale and rotation invariant Fourier descriptor, F is given by $|F(u)|/|F(0)|$ and examples of these vectors for two shapes are shown in Fig. 4.1b. Notice that the FDs for both shapes are mostly identical except for the higher frequency components that are clustered around samples 20-40. High frequency errors typically arise from boundary discretisation noise and misalignment in boundary point sampling. The Euclidean distance (L_2 -distance) between the two feature vectors F_1 and F_2 can be used as a dissimilarity measure between the shapes and this is given by

$$d_{L_2}(F_1, F_2) = \sqrt{\sum_{i=1}^N (F_1(i) - F_2(i))^2} \quad (4.2)$$

Another boundary-based shape descriptor that has recently demonstrated good results in shape classification and recognition applications is the *shape context* proposed by Belongie *et al.* [18]. This is a contour-based descriptor that is not restricted to a single closed-contour like the FD. The shape context analysis begins by taking N samples from the boundary of a shape as shown in Fig. 4.2b. Now consider the vector originating from a point in Fig 4.2c to all other points in the shape. These vectors express the appearance of the entire shape relative to the reference point and can be accumulated into a histogram (termed shape context) using a log-polar coordinate system centred about the reference point. A shape with N sample points will have N histograms such as that shown in Fig 4.2d. The similarity between two shapes A and B , each having their respective N shape context histograms can be computed by first computing the χ^2 -distances between every histogram of shape A with every histogram of shape B . The combinatorial matching of histograms between shapes A and B that gives the least total cost is obtained by solving as a weighted bipartite graph matching problem. Issues such as scaling and rotation invariance are detailed in [17].

Figure 4.2 – The computation of the shape context. (a) The original shape. (b) Edges and tangents taken from 50 equally spaced sample points along the boundary. (c) Vectors from a reference sample point (marked +) to all other sample points. Overlaid is the log-polar sampling grid used to derive the histogram shown in (d). This example shows 5 bins for log-space radius bins and 12 equally spaced angle bins. Adapted from [17].

Region representation - Shapes can be described by the geometric properties of the region within its bounding contour such as area, centre of mass (centroid), elongatedness, eccentricity, compactness and others [101], [208]. Fig. 4.3a highlights some of the commonly used geometric shape parameters. However, such geometric descriptors are not very meaningful when the shape becomes more complex, like in the example shown in Fig. 4.3b. More sophisticated region-based descriptors such as a quad-trees representation [181] may be more descriptive for such complex shapes. These representations result in a tree structure, which can be encoded using string symbols based on depth-first traversal of the tree starting at the root. Quad-tree representation of shapes is very compact and has found uses in applications such as image compression [101]. Instead of square grids, shape matrices and shape vectors extract quantitative information about a region using a polar coordinates-based grid centred at the shape's centroid [78], [211], [139].

Lu and Sajjanhar [142] proposed another string-like region-based representation by using a grid overlay technique. However, this grid-based descriptor (GD) does not implicitly handle rotation and thus requires the shape to be first aligned along its major axis. This method of achieving rotation invariance via major-axis alignment rotation can be problematic with non-elongated shapes or elongated ones that undergo bending deformation like the one shown in Fig. 4.3b.

Figure 4.3 – Some geometric properties of a shape. (a) Area, centroid, perimeter, elongatedness, eccentricity, rectangularity and compactness. (b) For more complex shapes such as this, the bounding rectangle does not always give a good representation for elongatedness. It is also difficult to determine an appropriate major axis to define the shape's global orientation. Adapted and summarised from [208].

A popular category of region-based descriptors are those based on moments [96]. Given a shape A with a connected region $\mathfrak{R}$, the (p,q) -moment of A is given by

$$m_{p,q} = \int_{(x,y) \in \mathfrak{R}} x^p y^q dx dy \quad (4.3)$$

For finite point sets, the integral can be substituted by a summation over the two spatial dimensions x and y . The infinite sequence of moments, $p, q = 0, 1, \dots$, uniquely determines the shape. The higher-order the moments, the more detailed is the shape description. However, this is usually at the expense of additional computational cost and increased sensitivity to noise. Moments are popular because of their invariance to many geometric transformations such as translation, rotation and scaling [208]. Affine moment invariants have also been derived by Flusser and Suk [63]. Other variants include geometric moments, Legendre moments, Zernike moments and pseudo Zernike moments [212], [213]. A comparative study on content-based shape retrieval of contour shapes done by Zhang and Lu [231] concluded that Zernike moment descriptors (ZMD) performed slightly better than other shape descriptors such as Mokhtarian's [150] curvature scale space, Lu and Sajjanhar's GD [142] and the Fourier descriptor. In another recent comparative study of three region-based shape descriptors (geometric moments, GD and ZMD), Zhang and Lu [233] again concluded that the ZMD is the most robust of the three. Teague [212] introduced Zernike moments, which allow the construction of independent moment invariants of an arbitrary high order. The complex Zernike moments are derived from Zernike polynomials given by

$$V_{nm}(x, y) = V_{nm}(\rho \cos \theta, \rho \sin \theta) = R_{nm}(\rho) \exp(jm\theta) \quad (4.4)$$

and

$$R_{nm}(\rho) = \sum_{s=0}^{(n-|m|)/2} (-1)^s \frac{(n-s)!}{s! \left(\frac{n+|m|}{2} - s\right)! \left(\frac{n-|m|}{2} - s\right)!} \rho^{n-2s} \quad (4.5)$$

where ρ is the radius from pixel (x, y) to the shape's centroid and θ is the angle between ρ and the x -axis. The integers n and m are subject to $n - |m| = \text{even}$ and $|m| \leq n$. Zernike polynomials are a complete set of complex-valued functions orthogonal over

the unit disk defined by $x^2 + y^2 = 1$. The complex Zernike moments of order n with repetition m can be defined as

$$Z_{nm} = \frac{n+1}{\pi} \sum_x \sum_y I(x, y) V_{nm}^*(x, y) \quad \text{for } x^2 + y^2 \leq 1 \quad (4.6)$$

where $*$ means complex conjugate and $I(x, y)$ is the image of concern. In order to obtain a descriptor that is invariant to translation, scaling and rotation, the input image has to be preprocessed as shown in Fig. 4.4. Since Zernike basis functions are computed within a unit disk, translation invariance can be obtained by shifting the centre of the unit disk to the shape's centroid given by (4.20). Scale invariance is obtained by resizing the shape such that the distance between the shape's centroid and the furthest point on the shape's boundary is always kept to a fixed distance R . In the case of images of size 129×129 pixels, the value of R would be $(width - 1)/2 = 64$ pixels. Rotation invariance is achieved by using only the magnitude of the moments. The magnitude is normalised to $[0, 1]$ by dividing it by the mass of the shape, which is given by the number of pixels in the region $\mathfrak{R}$ that defines the shape.

Figure 4.4 – Steps in deriving the Zernike moments descriptor. (a) The original binary image, where black pixels are of value 1 and white pixels are of value 0. (b) The shape is shifted such that its centroid is at the centre of the image. (c) The final image $I(x, y)$ after the centred-shape is scaled to fit the unit circle. (d) The ZMD consists of a vector comprising the complex Zernike moments's magnitude after it is normalised by the mass of the shape. This example shows the first 36 moments obtained with order n of 0 to 10 and all permitted repetitions m .

The dissimilarity between two shapes described by two Zernike moments vectors $\mathbf{Z}_1$ and $\mathbf{Z}_2$ can be determined by measuring their Euclidean distance as in (4.2) but with $\mathbf{F}_1$ and $\mathbf{F}_2$ replaced by $\mathbf{Z}_1$ and $\mathbf{Z}_2$. The precision of the shape representation depends on the number of moments truncated from the expansion. Being region-based and not limited to closed boundary shapes (like FD), ZMD is often used in shape recognition applications involving complicated trademarks images [103], [109] such as those shown in Fig. 4.5. ZMD can readily describe such complex shapes without the need to introduce preprocessing stages such as extracting and grouping region boundaries into perceptual families of shapes [57] or image filling [103]. Some have tackled these complex shapes by devising global statistical shape descriptors (in the form of histograms) that used local shape information such as edge directions [103] or geometric relationships between edge pairs [100].

Figure 4.5 – Typical trademark images. These shapes are complex because it is not possible to describe them as a single closed contour. Taken from [57].

Boundary or region? – Now, returning to the question, should shape be represented and compared based on its boundary or its enclosed region? Basri *et al.* [14] highlighted an example of the shape *A* shown in Fig. 4.6, which undergoes similar amounts of local distortion to two edges of its boundary to produce two different shapes *B* and *C*. In each case, the extent of the local boundary distortion is identical yet most human observers will testify that shape *A* looks more like *B* than shape *C*. Indeed, Liu and Geiger [136] emphasised the fact that boundary-based representation will not be able to produce shape dissimilarity costs between shapes *A*, *B* and *C* that will match human perception of similarity. This is because human shape perception takes into account not just boundary distortion but also distortions to the 2D region bounded by the contour and other Gestalt properties like symmetry and collinearity [159].

Figure 4.6 – A weakness of boundary-based shape representations. They do not readily account for region information and symmetric relationship between contour segments that are in close spatial proximity though far apart along the 1-D boundary contour. Shape A is distorted into shapes B and C by stretching each of the dashed lines by a factor of 2. Adapted from [14].

Region-based shape descriptors tend to be global in nature as it is often difficult to segment a shape into meaningful subparts based on a region-only representation. On the other hand, many boundary representations such as polyline representations for example, implicitly segment the shape boundary into parts that can be subsequently matched in a partial manner with another shape. This allows the representation to handle problems such as occlusion and partial matching. This issue of global and local shape representation is further discussed in the next section.

Ideally, shape representation should explicitly combine both boundary and region information. Like Kimia *et al.* [111], this thesis argues that the integration of boundary and region information is necessary to realise a shape descriptor that is robust and exhibits closer conformity to the human perception of visual forms. Shape descriptors such as those using medial axis or local symmetry can be said to be just such a representation [111], [131], [188]. The shock graphs of Siddiqi *et al.* [201] for example, are created starting from information at the boundary but are formed within the shape interior region. The proposed multiresolution gradient vector field presented in chapter 2 is conceptually similar since gradient information at the boundary is spatially integrated into the interior region using a pyramidal framework. In this sense, the MGVF describes a shape by making explicit use of both boundary and region information. Additionally, the smoothing parameter α can be viewed as a parameter that allows one to control the amount of boundary detail used in the shape representation.

4.1.2 Shape Similarity and Matching

Shape matching is an important component of many image-based applications such as content-based image retrieval, object recognition and classification, alignment and registration, etc. Within the context of this work, a shape is defined by the geometric characteristics of an enclosed and connected region. Shape matching is the task of transforming a shape into a suitable description, which will allow its resemblance to another shape to be reliably quantified through some form of dissimilarity measure. The goals of a good shape matching technique is to produce results that are in close correspondence to human perception and to do so in a computationally efficient manner, while using a representation that is compact and robust. A comprehensive survey of various shape matching algorithms and dissimilarity measures is given in [221], [222]. This section reviews some common properties that are often considered desirable in a dissimilarity measure. However, it is often the application at hand that determines which properties are desirable as certain combinations of properties are contradictory and cannot be realised within the same distance measure.

Metric - Consider the three shapes A , B and C and the shape distance function d . If $d(A,A)=0$ and $d(A,B)=0$ implying $A=B$, then the function d satisfies both the identity and uniqueness properties, respectively. The function d is said to be a metric if it satisfies the identity, uniqueness and strong triangle inequality properties [14], [221] and a strong form of triangle inequality as defined in [221] is given by

$$d(A,B) + d(A,C) \geq d(B,C) \quad (4.7)$$

The metric property is especially useful in object recognition applications where a poor match between shapes A and B , followed by a good match between shapes B and C will allow us to infer, without further matching, that shapes A and C are not similar. For this reason, most efficient techniques for organising databases of shape features and finding nearest neighbours rely on the metric distances [14], [24]. However, it is well known that human perception of shape similarity does not always obey the metric property, in particular the triangle inequality property [14], [24], [221]. With reference to the shapes in Fig. 4.7, a human observer will perceive the distances between the composite shape A (Greeble-rider) and both shapes B (Rider) and C (Greeble) to be small, since both shapes B and C have a *partial match* in A . However, the perceived distance between shapes B and C is large as they share no similarity.

This is in conflict with the strong form of triangle inequality given in (4.7). One way of overcoming these contradictory requirements is to combine the use of metric and non-metric dissimilarity measures when performing shape matching (see chapter 6).

Figure 4.7 – The triangle inequality property is not valid under situations where partial matching is permitted. For the shapes A , B and C above, human perception would rate $d(A, B)$ as small, $d(A, C)$ as small, even when $d(B, C)$ is large.

Continuity – The property of continuity is often desirable as it permits similar looking shapes to have low dissimilarity measures despite the presence of minor artifacts or distortions such as boundary discretisation noise, smoothing of corners, etc. Continuity implies that if a smoothly evolving shape B_i converges to shape C , then $d(A, B_i)$ converges to $d(A, C)$. This principle is illustrated in Fig. 4.8 for a corner that is smoothly changing from one that is sharp to one that is rounded. Veltkamp in [221] highlighted other aspects of robustness, which he regards as a form of continuity. These include robustness to perturbation, small affine distortions, cracks, blurring, noise and occlusion. Some examples of other continuity properties such as boundary discretisation, boundary noise and occlusion are illustrated in Fig. 4.9.

Figure 4.8 – The continuity property as applied to the curvature of a corner.

Monotonicity – This property is an extension of the continuity property applied to shape changes that are governed by a parametric-based deformation, such as bending, stretching and compression. Monotonicity implies that the dissimilarity measure between a shape and its deformed counterpart should increase monotonically as the deformation applied to the original shape is increased [14]. However, as

pointed out by Basri *et al.* [14], the issue of monotonicity is more complicated in the presence of multiple types of deformations that are simultaneously varying.

Figure 4.9 – The continuity property exhibited in shapes with increasing boundary discretisation, boundary noise and occlusion.

Invariance – It might be desirable for a distance function to have some invariant properties under certain classes of transformation. The most basic class will be the similarity transform, which includes invariance to translation, rotation and scaling. This is generally considered the minimum invariant requirement because these three transformations do not change the shape of the object. Affine and perspective invariance may also be desirable in some object recognition applications. However, not all applications are amenable to a large invariance group. For example, Thomson [214] showed that two hatchetfishes of different genus could be related through an affine transformation (see Fig. 4.10). In such cases, an affine invariant distance function is not discriminating enough and would result in a misclassification.

Figure 4.10 – Two hatchetfishes of different genus related through an affine transformation [214]. Adapted from [221].

4.2 Shape Description using Gradient Vector Fields

Many researchers in the past have proposed the use of vector fields in the analysis of images and shapes. As reviewed in section 3.1.3, works such as those proposed in [1], [50], [197], [202] for example, were mainly focused on the extraction of medial or symmetric image features from vector fields. Subsequent analysis of the shapes was based on these extracted medial primitives. Others like Scharstein [183] and Seitz [192] employed local information in the gradient vector field for point-to-point matching between images. Their work was employed in stereoscopic disparity computation and traffic signs recognition. More closely related to the issue of shape description is the work proposed by Ben-Arie *et al.* [20], [21]. They extracted hierarchical shape descriptors at locations of high vectorial disparity (based on a measure they termed *Cancellation Energy*) when the image gradients at shape boundaries are radially propagated. The issue of scale was later addressed in [22] by their use of a Vectorial Gradual Lattice Pyramid. Their normalised shape feature tokens were shown to be invariant to scaling and rotation but the results presented in [21] were restricted to only 25 distinct shapes.

How can the MGVF associated with a shape be used to formulate a descriptor that exhibits desirable properties such as those listed in section 4.1.2? The next section proposes a statistical shape descriptor based on the histogram of the gradient vector's orientation. Lowe [140] and Ullah *et al.* [219], [220] employed orientation histograms of gradient vectors that are derived from gray-level images. In Lowe's case, the histograms were constructed within the neighbourhood (Gaussian-weighted) of key image points deemed to be of interest. In both works, the orientation histograms were used as template features to locate objects or important localities in an image. The orientation histogram proposed in this chapter is a global shape descriptor, in that it describes the entire shape with one descriptor. It is assumed that the interior region of the object has been segmented out from the scene and subsequently binarised. Issues such as its construction, the incorporation of scale and rotation invariance and shape matching via histogram comparison are discussed. Experimental results highlighting the desirable properties of the orientation histogram, its comparative performance with other global shape descriptors and its limitations are presented.

4.2.1 The Orientation Histogram

Consider a binary image $I(x,y)$ with an object A , whose shape is defined by a connected region $\mathfrak{R}$ and its corresponding MGVF $V_0(x,y)$, which has been extracted from level 0 of the Gradient Vector Field pyramid $V(l,x,y)$ given in (2.5). Each pixel in $I(x,y) \in \mathfrak{R}$ has a corresponding gradient vector $V_0(x,y)$ and this provides the pixel with an orientation $\theta(x,y) \in [0, 2\pi)$ that is defined by

$$\theta(x,y) = \arctan\left(\frac{V_0^y(x,y)}{V_0^x(x,y)}\right) \quad (4.8)$$

where V_0^x and V_0^y are the corresponding x and y components of the gradient vectors. For convenience, the orientation $\theta(x,y)$ is expressed as a positive value between 0 and 2π instead of the more conventional range of $-\pi$ to π . An n -bin orientation histogram H for object A can be built by accumulating the orientation $\theta(x,y)$ of every pixel in $I(x,y) \in \mathfrak{R}$ into its respective discretised bin. The bin $H(i)$ is incremented as follows:

$$H(i) = \begin{cases} H(i) + 1 & \text{if } I(x,y) \in \mathfrak{R} \text{ and } \theta(x,y) \in \theta_i \\ H(i) & \text{otherwise} \end{cases} \quad (4.9)$$

where θ_i is the range of orientation angles spanned by the i^{th} histogram bin. Each of the n bins is of width $2\pi/n$ and they are equally spaced between the angles 0 to 2π and are ordered from 1.. n , starting with the lowest angular range.

A scale invariant descriptor in the form of histogram h can be obtained by normalising the histogram H using the total number of pixels in region $\mathfrak{R}$ as follows:

$$h(i) = \frac{H(i)}{\sum_{j=1}^n H(j)} \quad (4.10)$$

The orientation histogram, being a statistical description of the shape, can be readily compared to yield a metric distance. This is an important property for all practical shape descriptors as outlined in section 4.1.2. The next section discusses the various measures that can be used to compare histograms.

4.2.2 Comparing Orientation Histograms

In order to compare the similarity between two shapes A and B , some measure of dissimilarity between their respective histograms h and k must be used. Rubner *et al.* gives a good summary of many popular histogram dissimilarity measures in [178]. There are broadly two categories of measures that can be adopted [178]. The first is the *bin-by-bin* dissimilarity measure that only compares the contents of corresponding histogram bins $h(i)$ and $k(i)$ for $1 \leq i \leq n$, but not $h(i)$ and $k(j)$ for $i \neq j$. The second is the *cross-bin* measure, which take into account non-corresponding bins during comparison. In order to do this, cross-bin measures make use of the ground distance $g(i, j)$, which defines the distance between representative features in bins i and j .

Figure 4.11 – Bin-by-bin and cross-bin histogram comparison. (a) Single bin-width shift due to bin discretisation errors during the construction of orientation histograms can produce a large discrepancy when using bin-by-bin comparison techniques, especially if the histogram profile is peaky in nature. In this case, the χ^2 -distance wrongly measures h_1 as being more similar to k_2 than k_1 . (b) Quadratic form distances that incorporate cross-bin information can help overcome such problems.

As illustrated by Fig. 4.11a, the bin-by-bin measures are generally more sensitive to the position of bin boundaries and the selection of n , the bin size of the histogram. On the other hand, cross-bin measures such as the quadratic form distance [155] shown in Fig. 4.11b has a tendency to overestimate the mutual similarity between histograms that do not have pronounced peaks [178], [209]. Moreover, the cost of computing cross-bin measures are often much higher than those required by bin-by-bin measures. All experimental results presented are based on these two types of distance measures.

Bin-by-bin measure - The χ^2 -dissimilarity measure is a bin-by-bin measure that has its origins in mathematical statistics. Belongie *et al.* have used it successfully for matching shape context [18], while Schiele and Crowley adopted it for matching multidimensional receptive field (MRF) histograms [186]. The χ^2 -distance between two n -bin normalized histograms h and k is given by

$$d_{\chi}(h, k) = \frac{1}{2} \sum_{i=1}^n \frac{[h(i) - k(i)]^2}{h(i) + k(i)} \quad (4.11)$$

Schiele and Crowley conducted object recognition experiments in [184] with their proposed MRF histograms using three different histogram dissimilarity measures, namely sum of squared difference distance (SSD) given by

$$d_S(h, k) = \sum_{i=1}^n [h(i) - k(i)]^2, \quad (4.12)$$

Swain and Ballard's intersection distance [210] given by

$$d_I(h, k) = \sum_{i=1}^n \min[h(i), k(i)], \quad (4.13)$$

and the χ^2 -distance. Their results showed that the χ^2 -measure was consistently the most reliable form of distance measure for comparing MRF histograms.

Cross-bin measure - The second method of comparing histograms that was used is the cross-bin dissimilarity measure proposed by Niblack *et al.* called the quadratic form distance [155]. The quadratic form distance between two histograms h and k can be given in matrix notation by using a similarity matrix A of weights and is given by

$$d_Q(h, k) = \sqrt{(h - k)^T A (h - k)} \quad (4.14)$$

where the square matrix $A = [a_{ij}]$ defines the cross-bin relationships between different histogram bins. In their work on image database retrieval, Niblack *et al.* [155] proposed a linear relationship between histogram bins given by

$$a_{ij} = 1 - \frac{g_{ij}}{\max(g_{ij})} \quad (4.15)$$

where g_{ij} is the ground distance between bin i and j of the histogram. Due to the cyclical nature of the angular measurement of orientation, the ground distance of an n -bin orientation histogram is given by

$$g_{ij} = |i - j| \quad \text{if } |i - j| < \frac{n+1}{2}$$

$$g_{ij} = n - |i - j| \quad \text{if } |i - j| \geq \frac{n+1}{2}, \quad \text{for } i, j = 1, 2, 3, \dots, n \quad (4.16)$$

A similarity matrix A with a linear roll-off is not suitable for comparing orientation histograms as the cross-bin similarity information encapsulated in matrix A is needed mainly to account for the bin discretisation errors that are typically in the range of ± 1 bin position. A steeper roll-off such as the Gaussian roll-off [88] shown in Fig. 4.11b is preferable and this is given by

$$a_{ij} = \exp\left(-\sigma_Q [g_{ij} / \max(g_{ij})]^2\right) \quad (4.17)$$

where σ_Q controls how steep the Gaussian roll-off should be. Typical values of 8 to 10 were used in the experiments presented.

The Earth Mover's Distance (EMD) is a distance measure based on the minimal cost that must be paid to transform one distribution to another. It has been employed by Rubner *et al.* [178] for image retrieval, where the image descriptors such as color and texture are encoded in the form of signatures as opposed to regular, fixed bin histograms. They claim that EMD, when applied to signatures, produced more robust matching than the histogram-based techniques reviewed earlier. EMD was not used to match orientation histograms as it was suggested in [178] that it does not work well on small histograms (due to coarse ground distances) and is too slow for large ones.

Complexity - The computational requirements of the quadratic-form distance is typically $O(n^2)$, where n is the number of bins in the histograms. The exact complexity depends very much on the choice of matrix A [185]. Hafner *et al.* suggested that complexity of the order $O(n)$ can be achieved with some pre-computation such as diagonalising the quadratic form [88]. On the other hand, the complexity of computing χ^2 distances is strictly $O(n)$. The number of operations required per histogram cell is five, which are two additions, a subtraction, a multiplication and a division. Both of these techniques are significantly more efficient than EMD's computational requirements of $O(n^3 \log n)$.

4.2.3 Characteristics of the Orientation Histogram

Monotonicity – One of the characteristics of orientation histograms that is desirable is its ability to capture the gradual deformation of shapes, thus encapsulating the monotonicity property [14] described in 4.1.2, where a small deformation in the shape should produce a corresponding small monotonic change in the shape dissimilarity measure. Fig. 4.12 shows examples of scale-normalised orientation histograms for a circular shape that is deformed (gradually) to an elliptical shape and then further deformed (rapidly) from an ellipse to a sausage-like shape, through progressive elongation in the horizontal axis. Notice the two histogram peaks representing the orientation angles $\pi/2$ and $3\pi/2$ gradually become more ‘peaky’ as the shape becomes more elongated. The resulting dissimilarity measures based on the χ^2 -distances between the histogram of the reference circle and the deformed shapes exhibit this similar monotonic and proportional change with increasing distortion.

Figure 4.12 – The monotonicity characteristics of orientation histograms. Four shapes undergoing progressive stretching (top) and their respective scale-normalised orientation histograms (bottom). The χ^2 -distances shown compare the dissimilarity of the circle with each of the deformed shapes.

[†] Histogram and MGVF parameters used for all experiments presented in this section are $n = 36$ and $\alpha = 0.9$, unless otherwise stated. So is the $\times 1000$ scaling factor used for all the χ^2 -distances displayed at the top-right hand corner.

Continuity – It is often desirable that the dissimilarity measure between two shape descriptors exhibits some continuity properties [14], [221]. For example, if the curvature of a shape changes smoothly, so should the shape dissimilarity cost. The curvature and occlusion continuity property of the orientation histograms is illustrated in Fig. 4.13. The χ^2 -distances shown compares the dissimilarity of each gradually changing shape with its respective reference shape.

Figure 4.13 – (a) Curvature continuity and (b) occlusion continuity properties of the orientation histogram.

Handling polygons – As highlighted by Basri *et al.* in [14], one of the difficulties of curvature-based shape similarity measures is the fact that curvature is not properly defined at the corners of a polygon. It is therefore not straightforward to measure the shape distance between a circle and a square, without applying some form of regularization, like smoothing of the corners. Yet humans have no problems perceiving that an octagon looks more like a circle than say a square or a triangle. The orientation histogram does not suffer from such problems since it does not rely on the computation of boundary curvature and it provides a similarity measure that matches human perception of curved and polygonal shapes (see Fig. 4.14).

Figure 4.14 – Comparing a circle with polygonal shapes. The χ^2 -distances shown compares the dissimilarity of each polygonal shape and the circle on the left.

Robustness to boundary noise - The orientation histogram is essentially a global shape descriptor. As such, it is relatively robust against boundary perturbations. As long as there is a sufficiently large interior region, such as in Fig. 4.15a-c, the boundary noise will not have a significant influence on the general profile of the orientation histogram. Unfortunately, as the statistical samples reduce with the size of the object, the histogram's profile begins to vary more unpredictably with increasing boundary noise (see Fig. 4.15d-f). An alternative way of looking at the noise robustness of the orientation histogram is to consider the ratio of the shape's interior pixels to its boundary pixels. The lower the ratio of interior to boundary pixels (e.g. a narrow rectangle), the more susceptible the shape's orientation histogram is to boundary noise.

Figure 4.15 – Shapes with varying degrees of boundary noise and their respective scale-normalised orientation histograms. Results for a large square with (a) no noise, (b) zero-mean Gaussian boundary perturbations with $\sigma_n = 2.0$ and (c) $\sigma_n = 4.0$. Results for a small square with (d) no noise, (e) $\sigma_n = 2.0$ and (f) $\sigma_n = 4.0$. The χ^2 -distances shown compare the dissimilarity of each clean square and their respective noisy counterparts.

Smoothing parameter α - The profile of the orientation histogram is affected by the choice of the smoothing parameter α in (2.5), as can be observed in Fig. 4.16. As the value of α reduces, boundary features with high spatial frequency components (e.g. corners and sharp perturbations) are increasingly attenuated. This is sometimes useful as appropriate values of α could be used to reduce the effects of boundary discretisation noise, as can be seen in Fig. 4.17. However, excessive smoothing of the vector field using very small values of α is undesirable since distinguishing features of the shape will also be lost.

Figure 4.16 – The influence of the smoothing parameter α on the MGVF and its corresponding scale-normalised orientation histogram profile. Results at (a) $\alpha = 0.9$, (b) $\alpha = 0.5$ and (c) $\alpha = 0.1$. As α is reduced, the histogram's profile 'flattens out'. This effectively corresponds to the corners of the square being 'rounded-off'.

The values of α used in the shape analysis experiments presented later were selected empirically. Typical values of 0.8 to 0.9 were found to be appropriate for most shape matching experiments. For images with higher boundary noise and boundary texture, values of 0.3 to 0.5 were found to be more suitable. The influence of α on the shape description quality of the orientation histogram is further discussed in the shape recognition experiments found in section 4.3.2.

The gradual smoothing of the shape using decreasing values of α is reminiscent of Kimia's shape smoothing through curvature deformation using non-linear diffusion [111] and Mokhtarian's curvature scale space evolution [150]. In the case of the former, Grayson [80] showed that any embedded curve shrinks to a circle and eventually to a circular point under the geometric heat equation. This is evident in Fig. 4.17 as the shape distance between the triangle and circle at $\alpha = 0.1$ is indeed minimal.

Figure 4.17 – Reducing the effects of boundary discretisation noise using appropriate values of the smoothing parameter α . However, excessive smoothing will eventually cause all MGVF to resemble those of circles or ellipses. This can be seen by the fact the reference shape (triangle) found the circle a better match at $\alpha = 0.1$ than one of the other distorted triangles. The χ^2 -distances shown compare the dissimilarity of the reference triangle to each of the other shapes. For each value of α , the shapes are arranged from left to right in order of the best match.

4.2.4 Rotation and Mirror Reflection

The orientation histogram is not invariant to rotation and mirror reflection, as can be seen in Fig. 4.18. Rotation of a shape will result in a corresponding circular shift in the histogram's bin values (see Fig. 4.18b). On the other hand, a mirror reflection will cause the bin values of the original histogram to be reversed and depending on the axis about which the reflection occurs, this may also be accompanied by a circular shift (see Fig. 4.18c). Since the orientation binning process is discrete, some slight change in the profile of the histogram may also occur due to discretisation errors when sorting individual gradient vectors into their respective bins.

Figure 4.18 – Effect of rotation and mirror reflection on the orientation histogram. (a) Reference shape with its orientation histogram of bin size $n = 36$. (b) Rotation of Θ radians will cause the n -bin histogram to be circular-shifted by $\Theta n / 2\pi$ bins. (c) Mirror reflection causes a reversal of bin order and also a possible circular shift.

There are several possible techniques that can make the orientation histogram insensitive to rotation. One obvious technique, which was used by Lowe in [84], is to determine the bin with the largest value after applying some appropriate amount of Gaussian-weighted smoothing on the histogram. All histograms can be rotation-normalised by circular shifting this peak bin to a common reference position. However, histogram peak normalisation is not very reliable since some shapes have more than one prominent peak. As a result, the positions of highest peaks can be

easily swapped by slight distortions in the shape or by the addition and deletion of small parts. This is illustrated in Fig. 4.19 using two very similar rabbit shapes.

Figure 4.19 – Histogram peak normalisation is unreliable if the orientation histograms have several equally prominent peaks. (a) The desert-tail rabbit and (b) the east-tail rabbit with their respective orientation histogram, which give a low χ^2 -distance of 0.03 before rotation normalisation but (c) a erroneous higher value of 0.14 after histogram peak normalisation.

Moment-based Global Orientation Normalisation (MGON) – If a suitable reference cannot be reliably found in the orientation histogram, then an alternative approach is to directly compute a rotation invariant reference from the shape itself. A technique which works well with elongated objects is to first determine the global orientation Θ of the object with respect to the image x and y -axes. Rotation invariance can then be achieved by adding the value of Θ to the individual orientation $\theta(x, y)$ of each gradient vector before accumulation into the orientation histogram. The global orientation Θ_M of a shape defined by a connected region $\mathfrak{R}$ can be computed using its (p, q) -moment defined in (4.3) and is given by

$$\Theta_M = \frac{1}{2} \tan^{-1} \left(\frac{2m_{11}}{m_{20} - m_{02}} \right) \quad (4.18)$$

where m_{pq} is called the *central moment* and is computed using the shape's centroid (x_c, y_c) as the origin. In a discrete image, the central moment is computed using a summation over the entire connected region $\mathfrak{R}$ and is given by

$$m_{pq} = \sum_{(x,y) \in \mathfrak{R}} (x - x_c)^p (y - y_c)^q \quad (4.19)$$

and the centroid (x_c, y_c) is given by

$$x_c = \frac{1}{N} \sum_{(x,y) \in \mathfrak{R}} x \quad \text{and} \quad y_c = \frac{1}{N} \sum_{(x,y) \in \mathfrak{R}} y \quad (4.20)$$

where N is the number of pixels in the region $\mathfrak{R}$. The 90° ambiguity in (4.18) can be resolved by adding an additional offset of $\pi/2$ to Θ_M if the constraint of $m_{20} < m_{02}$ is met [43]. However, a 180° ambiguity still exists with MGON.

Gradient-based Global Orientation Normalisation (GGON) – Taking advantage of the available MGVF, a novel gradient-based global orientation estimation technique is proposed. Bigün *et al.* [27] presented a method for determining local orientation in the multidimensional Fourier-space by treating the spectral energy as a mass distribution and then reducing the multidimensional tensors to their principal axes. The tensor representation [35] corresponding to such a system in 2D is given by

$$J = \begin{bmatrix} J_{00} & -J_{01} \\ -J_{10} & J_{11} \end{bmatrix} \quad (4.21)$$

where J is a second-order moment of inertia tensor about the origin. By utilising the differentiation properties of the Fourier transform and Parseval's theorem [36], the elements of the moment of inertia tensor matrix J at an image locality $\mathbf{x}_0$ can be described by its first-order partial derivatives in the coordinate system of the spatial domain [27], [101]. The diagonal elements of J are given by

$$J_{ii}(\mathbf{x}_0) = \sum_{j \neq i} \int_{-\infty}^{\infty} g(\mathbf{x} - \mathbf{x}_0) \left[\frac{\partial f}{\partial x_i} \right]^2 d\mathbf{x} \quad (4.22)$$

and the non-diagonal elements are

$$J_{ij}(\mathbf{x}_0) = - \int_{-\infty}^{\infty} g(\mathbf{x} - \mathbf{x}_0) \left[\frac{\partial f}{\partial x_i} \frac{\partial f}{\partial x_j} \right] d\mathbf{x} \quad (4.23)$$

where the weighting function $g(\mathbf{x})$ control the size of the neighbourhood in which the orientation of the local image feature is being determined. The required 2D

orientation associated with the principal axes can be found by solving the quadratic equation given by the characteristic equation of the tensor J in (4.21). The eigenvalue analysis of matrix J will yield two eigenvalues and their associated eigenvectors. The axis about which minimum moment of inertia is obtained is given by the eigenvector associated with the smallest eigenvalue [35]. This eigenvector corresponds to a line through the origin that has an orientation about which the spectral energy distribution has a best fit in the least squared error sense. However, the problem at hand is not the measurement of the local orientation but the global orientation of a connected region $\mathfrak{R}$. The global orientation Θ_G of the shape can be obtained by substituting the partial derivatives in (4.22) and (4.23) with the x and y components of the multiresolution gradient vectors given by V_0^x and V_0^y respectively and summing all vectors within the region $\mathfrak{R}$. The magnitude of the gradient vectors is normalised to unity to provide equal weightage for vectors within the shape. The resulting diagonal elements of the matrix J used to compute the gradient-based global orientation are now given by

$$J_{00} = \sum_{\mathbf{x} \in \mathfrak{R}} \frac{V_0^x(\mathbf{x})V_0^x(\mathbf{x})}{w(\mathbf{x})} \quad \text{and} \quad J_{11} = \sum_{\mathbf{x} \in \mathfrak{R}} \frac{V_0^y(\mathbf{x})V_0^y(\mathbf{x})}{w(\mathbf{x})} \quad (4.24)$$

and the non-diagonal elements are given by

$$J_{01} = J_{10} = \sum_{\mathbf{x} \in \mathfrak{R}} \frac{V_0^x(\mathbf{x})V_0^y(\mathbf{x})}{w(\mathbf{x})} \quad (4.25)$$

The weighting function $w(\mathbf{x})$ required to normalise the gradient vectors to unit magnitude is given by

$$w(\mathbf{x}) = [V_0^x(\mathbf{x})]^2 + [V_0^y(\mathbf{x})]^2 \quad (4.26)$$

If the eigenvector of the smallest eigenvalue of J is $\mathbf{E}_{min}$, then the gradient-based global orientation is given by

$$\Theta_G = \arctan \left(\frac{E_{min}^y}{E_{min}^x} \right) \quad (4.27)$$

where E_{min}^y and E_{min}^x are the y and x components of $\mathbf{E}_{min}$. Since the global orientation of a shape can only be reliably estimated when the shape is elongated, a measure of elongatedness is useful. One such measure is the certainty measure of local

orientation used in [27] and is derived from the maximum eigenvalue, D_{max} of matrix J and is given by

$$L_G = 2D_{max} - 1 \quad (4.28)$$

where $L_G \in [0,1]$ is closer to 1 if the shape is more elongated (see Fig. 4.20). The use of the elongatedness measure, L_G is discussed later.

Figure 4.20 – Shapes and their respective elongatedness measure computed by L_G .

The global orientations estimated using the gradient-based and moment-based techniques are not identical, as illustrated by the example in Fig. 4.21. where the MGVF of the shape consists mainly of vectors in the horizontal directions.

Figure 4.21 – Differing global orientation estimates for the moment-based and gradient-based techniques. (a) Three touching rectangles arranged in a slanted manner. (b) Its corresponding MGVF, which consists mainly of horizontal vectors. (c) The moment-based (dotted) and gradient-based global orientation (solid) estimates.

The resulting global orientation estimated by the gradient-based approach will be in the perpendicular direction (i.e. vertical) since it is dependent on the cumulative orientation of the gradient vectors within the shape and not its spatial distribution. The moment-based technique on the other hand, is dependent on the spatial distribution of the three elongated rectangles and therefore produces a global orientation that takes

into account their slanted arrangement. Since the MGVF is ultimately derived from the spatial topology of the shape, most simple compact shapes will result in very similar values of Θ_G (gradient-based) and Θ_M (moment-based). It is difficult to say which technique is superior but the shape recognition experiments in section 4.3.2 seem to suggest that for matching orientation histograms, rotation normalisation using Θ_G is more accurate than those using Θ_M . Fig. 4.22 also seems to suggest that the moments-based approach is more sensitive to the occlusion and limb articulation, as noted by Cortadellas *et al.* [48].

Figure 4.22 – The influence of (a) an occluding object and (b) limb articulation on the estimated global orientation. In both cases, the gradient-based technique (solid line) resulted in less variation than the moment-based technique (dotted line).

Both the MGON and GGON result in a 180° ambiguity, which can only be resolved by computing two distance measures each time the rotation-normalised orientation histograms h and k of shapes A and B are being compared. The first is between h and k and the second is between h and k^π , where the histogram k^π represents a 180° rotation of the shape B . The histogram k^π can be obtained by

circular shifting bin values in k by $n/2$ bins in either direction. The lower of the two distances is the chosen dissimilarity measure between the orientation histograms of shapes A and B .

In order to handle mirror reflection, a mirrored version of shape B must also be matched to shape A to see which yields a lower dissimilarity measure. If the orientation histogram of shape B is k , then its mirrored version k' is given by

$$k'(i) = k(n - i + 1) \quad (4.29)$$

where n is the total number of bins in histogram k . As before, the lower of the two distances is the chosen dissimilarity measure. Since the mirrored version also has a 180° ambiguity, an additional comparison with k'^π is also required.

Fourier Transform Normalisation (FTN) – The 180° ambiguity and the need to test for mirror reflection means the global orientation normalised histograms require 4 comparisons each time two shapes are matched. An alternative approach that requires only one feature vector comparison per shape matching is to use FTN. This approach is similar to that used by the Fourier descriptors based methods described in section 4.1.1. The orientation histogram can be treated as a shape signature since its profile is also cyclical in nature. The Fourier transform given in (4.1) can then be applied by replacing the shape signature $f(k)$ with values from the orientation histogram $h(k)$. By using only the magnitude of the Fourier transform vector $F(u)$, invariance to both rotation and mirror reflection can be achieved. If the original orientation histogram has n bins, the resulting FTN feature vector will also have n elements and any two such equal-length vectors can be compared using the L_2 -dissimilarity measure given in (4.2).

Adaptive hybrid normalisation (AHN) – The final technique for rotation normalisation of the orientation histogram is a hybrid of two techniques proposed so far. In order to address the shortcoming of GGON, which is basically the inability to estimate reliable global orientation for non-elongated shapes, the elongatedness measure L_G in (4.28) is used to adaptively switch between the GGON and FTN techniques. A threshold λ_{LG} can be set such that FTN is used if $L_G \leq \lambda_{LG}$ and GGON is used when $L_G > \lambda_{LG}$. A suitable threshold value of λ_{LG} was determined empirically to be about 0.1 – 0.2. Experimental results in the next section will compare the relative performance of the various rotation normalisation techniques proposed.

4.3 Experimental Results and Discussion

Several experiments were conducted to study the shape description capabilities of the proposed scale and rotation normalised MGVF-based orientation histogram. Its comparative performance with other global shape descriptors such as the Fourier descriptor [232], shape context [18] and Zernike moments [212] are also presented.

4.3.1 Geometric Invariance

The first experiment demonstrates the MGVF-based GGON orientation histogram invariance to geometric transformations such as scaling and rotation. Twelve 257×257 pixel-sized images comprising three basic shapes (i.e. hare, rabbit and turtle) in various orientations and spanning scales of two octaves were matched each to every other. Fig. 4.23 shows the three best matches to each query image.

GGON (d_i)

	Hare				Rabbit				Turtle			
	×1	×½	×2	×1	×1	×½	×2	×1	×1	×½	×2	×1
Scale:	Original	90°	-30°	mirror	Original	90°	-30°	mirror	Original	90°	-30°	mirror
Rotate:												
Query												
Match												
#1	0	60	6	0	1	56	14	1	0	58	15	0
#2	6	61	6	6	16	56	16	14	15	58	15	15
#3	60	68	68	61	59	59	56	56	58	67	67	58

Figure 4.23 – Results demonstrating GGON orientation histogram ability to handle scaling, rotation and mirror reflection. The χ^2 -distances (top-right corner, ×10000 and rounded to the nearest integer) describe the dissimilarity between query shape and the 3 best matches from the remaining pool of test shapes. #1 indicates the best match, #2 the next best match and so on. Parameters used were $n = 36$, $\alpha = 0.8$ and $\sigma_H = 1.5$.

As expected, all shapes were correctly matched to their respective shape type despite the hare and rabbit having quite similar visual forms. Matching of mirror reflected versions were done as suggested in section 4.2.4. The MGVF-based FTN feature vectors produced similar results (see Appendix B.1) using only one histogram

comparison per shape match compared to the four comparisons per shape match required by the GGON descriptor.

4.3.2 Shape Recognition

The second experiment involves the recognition of the twenty-five 129×129 pixel-sized binary images shown in Fig. 4.24. During each run, the boundary of each of the twenty-five shapes was perturbed in the x and y directions by varying degrees of zero-mean Gaussian noise. The standard deviation σ_n of the noise is progressively increased from 0 to 2.0, in steps of 0.2 for each run. The noisy test image was also randomly rotated (between 0 to 2π radians) to test for rotation invariance. Each generated test shape is matched to all the reference shapes shown in Fig. 4.24a. If the closest matching shape is its base shape, a correct recognition is awarded. A 100% recognition rate during a run requires all twenty-five test shapes, such as those shown in Fig. 4.24b, to be correctly matched to their respective base shape in 4.24a. Each experiment was repeated 20 times with the mean recognition rate plotted together with an error bar showing the standard deviation computed over the 20 trials.

Figure 4.24 – Shapes used in the shape recognition experiments. (a) The 25 base shapes forming the reference shape database used in every run of the experiment. These shapes were also used by Ben-Arie and Wang in [21]. (b) The 25 test shapes with random rotation and simulated boundary noise used in the $\sigma_n = 1.5$ test run.

The smoothing parameter α - The smoothness of the MGVF depends on the value of the smoothing parameter α in (2.5). As such, the shape recognition performance of the MGVF-based orientation histogram is affected by the value of α . Fig. 4.25a shows that the low value of $\alpha=0.3$ gave the best shape recognition performance at high boundary noise levels, while a high value of $\alpha=0.9$ displayed a rapid drop in recognition rates as σ_n is progressively increased. The smoother vector fields produced by lower values of α reduce the influence of boundary noise and therefore provide a more noise robust recognition of shapes. However, the smoothing of the MGVF also diminishes some relevant shape details that allow different shapes to be distinguished. This can be clearly seen in Fig 4.25b, where an extremely smooth MGVF produced by $\alpha=0.1$ has rendered some shapes indistinguishable even when there is hardly any boundary noise except that introduced by the random image rotation. The poor recognition rate remains relatively consistent at different boundary noise levels. This problem of excessive smoothing can also be gleaned in Fig. 4.25a, where the recognition rates at lower boundary noise levels (i.e. $\sigma_n < 1.0$) is generally better for $\alpha=0.5$ compared to a lower value of $\alpha=0.3$. In short, the choice of an appropriate value of α depends on the boundary noise levels of the extracted shape. Typical values of 0.3 to 0.5 seem suitable in noisy situations.

Figure 4.25 – Shape recognition results using MGVF generated from various values of smoothing parameter α . (a) Comparative recognition rates for α values of 0.3, 0.5 and 0.9. (b) Results for α values of 0.1 and 0.5. All results shown were obtained using FTN with the d_{L2} -dissimilarity measure, $n = 36$ and $\sigma_H = 1.5$.

Techniques for rotation invariance - The second set of results evaluates the various techniques proposed in section 4.2.4 for achieving rotation invariance. The graph in Fig. 4.26a shows that the use of gradient-based normalisation (GGON) consistently produces better recognition rates than that based on moments (MGON). This suggests that the use of the gradient vectors in the MGVF provide a more reliable means of making the orientation histogram rotation invariant. This is mainly because the gradient-based approach derives its global orientation estimates from an appropriately smooth MGVF, which is relatively less boundary noise sensitive.

As for the effects of using different histogram comparison measures with the GGON orientation histograms, the results in Fig. 4.26b shows that the quadratic-form distance (d_Q) provides better recognition rates than the χ^2 -distance (d_χ), especially at higher levels of boundary noise. This is because the quadratic-form distance, being a cross-bin dissimilarity measure, is more tolerant to the erroneous shifts in histogram peaks that may occur due to global orientation estimation errors caused by increasing boundary noise.

Figure 4.26 – Shape recognition results using global orientation normalisation. (a) Performance comparison between the gradient-based (GGON) and moments-based (MGON) techniques, with both using the χ^2 -dissimilarity measure. (b) Performance comparison for the GGON orientation histogram using χ^2 -distance (d_χ) and the quadratic-form (d_Q) distance with $\sigma_Q = 8.0$. The first is a bin-by-bin dissimilarity measure and the second is a cross-bin. Parameters used: ($n = 36$, $\alpha = 0.5$ and $\sigma_H = 1.5$).

The performance comparison between the GGON and Fourier transform normalisation (FTN) shown in Fig. 4.27a reveals that each technique has its respective strength and weakness. Due to its inability to estimate the global orientation of non-elongated shapes, GGON does not produce 100% recognition rates even when $\sigma_n = 0$. In this case, the boundary noise is that due to the boundary discretisation caused by the rotation of the test shape images. Global orientation estimation is problematic with some of the 25 shapes as indicated by their weak elongatedness measure L_G in Fig. 4.27b. On the other hand, FTN has no such limitation and is able to produce 100% recognition of all the 25 shapes up to a boundary noise of about $\sigma_n = 0.5$ (see Fig. 4.27a). However, as boundary noise increases, the performance of FTN falls off more rapidly than the GGON technique, highlighting its lower robustness to boundary noise. By selectively adopting either the FTN or GGON based on the computed elongatedness measure of the shape and a pre-selected threshold λ_{LG} , the adaptive hybrid normalisation (AHN) technique is able to produce the best overall recognition rate by combining the strengths of both these techniques.

Figure 4.27 – AHN gives the best overall shape recognition results compared to GGON or FTN. (a) Performance comparison between GGON (with d_χ), FTN (with d_{L2}) and AHN (with d_χ and d_{L2}). The adaptive elongatedness threshold for AHN was set to $\lambda_{LG} = 0.1$. (b) The elongatedness measure L_G in (4.28) for each shape based on the MGVF generated at $\alpha = 0.5$. Other parameters used were $\sigma_H = 1.5$ and $n = 36$.

Other shape descriptors - The performance of the MGVF-based AHN orientation histogram that adaptively switches between GGON (with d_Q) and FTN (with d_{L2}) was compared with the boundary-based Fourier descriptor (FD) and the region-based Zernike moments descriptor (ZMD). Fig. 4.28a shows that the ZMD performed the best and has 100% recognition rates for $\sigma_n \leq 1.3$, compared to the AHN descriptor limit of $\sigma_n \leq 0.6$. The FD's performance was the poorest.

A large number of the recognition errors for the AHN descriptor were caused by the shape pair mismatches highlighted in Fig. 4.28b. The Plane01 shape is composed of many narrow regions and this reduces the ratio of region to boundary pixels, resulting in an orientation histogram profile that is susceptible to boundary noise. Since the general histogram profiles of **Plane01** and **Shape08** are similar (see Fig. 4.28b), the likelihood of mismatch is high when σ_n is high. Using a low value of α to reduce the effects of boundary noise has an unintended side effect, which is the flattening of the orientation histogram's profile. Since the peaks of the **Cross15** shape are now attenuated, there is an increasing likelihood that it would be matched to the **Circle09** shape as the boundary noise increases. To top it all, the elongatedness measure of **Plane01** and **Circle09** are 0.0560 and 0.0005 respectively, which means AHN switches to the less noise robust FTN during shape matching.

Figure 4.28 – Shape recognition results comparing the MGVF-based descriptor with two other shape descriptors. (a) Performance comparison between AHN using $\lambda_{LG}=0.1$ (with d_Q and d_{L2}), ZMD (with $n=10$) and FD (with $N=64$). (b) The two shape pair mismatches (and their respective GGON orientation histograms) that contributed to the majority of the recognition errors incurred by the AHN descriptor.

4.3.3 Robustness to Boundary Texture

The performance of robust shape matching in the presence of boundary texture was demonstrated using twelve 129×129 pixel-sized images comprising four base shapes (i.e. hare, rabbit, pear and candle), each with three different boundary textures (i.e. smooth, wavy and spiky). Each image was matched with every other and the two best matches (excluding the query image itself) obtained using the MGVF-based shape descriptor with GGON are shown in Fig. 4.29. In order to de-emphasise the influence of boundary texture and capture only the gross shape form of each object, a low value of $\alpha = 0.4$ was used when generating the MGVF.

GGON (d_o)

Texture	Hare			Rabbit			Pear			Candle		
	Original	Wavy	Spiky	Original	Wavy	Spiky	Original	Wavy	Spiky	Original	Wavy	Spiky
Query												
Match #1	18	25	18	22	22	22	30	30	37	31	31	34
#2	27	27	25	28	28	22	37	50	50	46	34	46

Figure 4.29 – Matching result for boundary-textured shapes obtained using the MGVF-GGON orientation histogram with the quadratic-form distance ($\sigma_Q = 8.0$). The dissimilarity measures shown in the top-right corner are rounded to the nearest integer after a scaling of $\times 1000$. Parameters used were $\alpha = 0.4$, $\sigma_H = 1.5$ and $n = 36$ bins.

The comparative matching results presented in Table 4.1 are in the form of 1st and 2nd best matches that fall into the correct base shape category of the query image. Measured this way, the matching score for the GGON orientation histogram using the quadratic-form dissimilarity measure produced a perfect score of (12, 12). Rotation normalisation using FTN resulted in a poorer matching score of (12, 8), with misclassifications in the 2nd best matches involving very similar rabbit and hare shapes (see Appendix B - Fig. B.2). As observed earlier in the shape recognition experiments, GGON is generally more robust to boundary noise than FTN when the shapes are elongated. Similar results were observed here with the boundary-textured shapes.

The orientation histogram's robustness to boundary texture compared with other boundary-based shape descriptors such as the Fourier descriptor (FD) and the shape context (SC) of Belongie *et al.* [18] is evident in Table 4.1. FD and SC derive shape information by sampling points on the contour. Since textural features at the boundary can create large variations in the position of the sampled points, these shape descriptors are sensitive to such boundary artifacts.

Since the boundary-based shape descriptors have not proved to be robust against boundary texture, how would a region-based shape descriptor such as the Zernike moments descriptor (ZMD) given in (4.6) fare? The first 36 moments of order $n = 0$ to 10 and all permitted m repetitions were used in this experiment, as in [58], [233]. Table 4.1 shows that a total of 8 misclassifications were obtained with the ZMD, resulting in a matching score of (10, 6). This is an improvement over the boundary-based FD and SC descriptors but not the MGVF-based descriptors. Most of the misclassifications with ZMD were between shape types that shared very similar gross visual form, namely the hare and rabbit (see Fig. B.4a). This result suggests that the MGVF-based descriptor is better than the ZMD in robustly capturing relevant nuances in shape variations. Such subtle shape form variations allow a human observer to tell apart a rabbit from a hare or pear, despite the textural distortions to the shape boundary. Experiments with lower Zernike moments order (i.e. $n = 5$) did not reduce the number of misclassifications as expected, instead the results got slightly worse, from 8 to 9 (see Fig. B.4b). Trading off shape description details by using lower order moments not only reduces the influence of boundary texture but also attenuates the subtle shape variations required to discriminate between the various base shapes.

Shape Descriptor	Shape Similarity Measure	Score (Best match)		Total Mismatch
		1st	2nd	
GGON	d_Q	12	12	0
FTN	d_{L2}	12	8	4
FD	$d_{L2} (N = 64)$	8	1	15
SC	$d_\chi (N = 100)$	7	1	16
ZMD	$d_{L2} (n = 10)$	10	6	8
ZMD	$d_{L2} (n = 5)$	9	6	9

Table 4.1 – Comparative shape classification performance for textured shapes.

4.3.4 Silhouette Classification

In order to test the shape classification capabilities of the orientation histogram, a small database of twenty-five 129×129 pixel-sized silhouette images forming 6 different shape classes were used. This set of images, shown in Fig. 4.30, is similar to that used in the experiments of several other researchers in [18], [65], [195].

Figure 4.30 – The 25 silhouette images used in the shape classification experiments.

The matching results in Fig. 4.31 display the three best matches obtained when each of the 25 shapes were matched each to every other (except with itself) using the AHN orientation histogram. Other researchers have conducted similar experiments and they presented their results in the form of the 1st, 2nd and 3rd best matches that fall into the correct shape category. Measured this way, the shape classification results using the AHN orientation histogram are given as (24, 21, 19), as highlighted in Table 4.2. The best possible score for this experiment would be (25, 25, 25).

AHN (d_x and d_{L2})

Query																									
Match #1																									
#2																									
#3																									

Figure 4.31 – Shape classification results using the MGVF-based AHN orientation histogram. The GGON histograms and FTN feature vectors were compared using the χ^2 and L_2 -dissimilarity measures respectively. Adaptive switching between GGON and FTN was based on the elongatedness threshold of $\lambda_{LG}=0.2$. The parameters used were $\alpha=0.8$, $\sigma_H=1.5$ and $n=36$ bins.

The shape classification results for the various rotation normalisation techniques adopted by the MGVF-based shape descriptors are summarised in Table 4.2. For purposes of comparison, detailed results of the type presented in Fig. 4.31 for all techniques highlighted in Table 4.2 are available in Appendix B.3.

Normalisation Technique	Dissimilarity Measure	Score (Best match)			Total Mismatch
		1st	2nd	3rd	
AHN	d_χ and d_{L2}	24	21	19	11
GGON	d_χ	23	20	17	15
GGON	d_Q	24	19	14	18
MGON	d_χ	22	19	14	20
MGON	d_Q	22	17	13	23
FTN	d_{L2}	24	20	14	17

Table 4.2 – Shape classification performance of MGVF-based shape descriptors using different rotation normalisation techniques and histogram comparison measures.

As expected, the best performance came from the AHN technique, which combines the strengths of the next two best performers, namely GGON and FTN. However, as can be seen in Fig. 4.32a, classification of the planes, fishes and tools were problematic using GGON. In the case of the planes, this was due to the fact that these were not elongated objects. The FTN in Fig. 4.32b was able to address this particular problem but it misclassified shapes sharing similar gross visual forms such as the greebles and rabbits. FTN also had problems with the occluded hand. Many of the misclassifications seen in Fig. 4.32 are due to the inherent global nature of the orientation histogram. Global descriptors usually have problems handling the addition, deletion, distortion of parts and partial matching. Misclassifications between spanners with one/two heads or fishes with/without sharp protrusions are some examples of these problems (see Fig. 4.32a).

From the results of the shape classification experiment in Table 4.2, the quadratic-form (d_Q) dissimilarity measure did not perform better than the χ^2 -measure (d_χ), as was observed in the previous experiments involving boundary perturbations. Cross-bin measures such as the quadratic-form distance has a tendency to overestimate the mutual similarity of histograms that are not distinctively different due to the cross-bin “blurring” introduced by the similarity matrix. As can be seen in Fig. 4.32c, additional misclassifications between the rabbit, greeble and hands were observed. If the boundary noise levels are not expected to be significant, then the more computationally efficient χ^2 -distance measure should be adopted when comparing orientation histograms.

Figure 4.32 – Shape classification results for the MGVF-based descriptor using GGON and FTN. (a) GGON with d_χ , (b) FTN with d_{L2} and (c) GGON with d_O . In all cases, the parameters used were $\alpha = 0.8$, $\sigma_H = 1.5$ and $n = 36$.

Table 4.3 shows the comparative performance of the AHN orientation histogram (parameters used are detailed in Fig. 4.31) and other shape descriptors that were either applied to the 25 shape database shown in Fig. 4.30 or were applied to a similar data set and reported in published literature.

Shape Descriptor	Shape Similarity Measure	Score (Best match)			Total Mismatch
		1st	2nd	3rd	
AHN	d_χ and d_{L2} ($n = 36$)	24	21	19	11
SC	d_χ (WBGM, $N = 100$)	25	22	21	7
FD	d_{L2} ($N = 64$)	22	19	15	18
ZMD	d_{L2} ($n = 10$)	23	20	6	13
Gdalyahu <i>et al.</i>	Syntactic Matching	25	21	19	10
Sharvit <i>et al.</i>	Shock Graph Matching	23	21	20	11

WBGM = Weighted Bipartite Graph Matching

Table 4.3 – Comparative shape classification performance of various other shape descriptors, including FD, SC and ZMD. Other results were obtained from published works and they include the flexible syntactic matching of Gdalyahu *et al.* [65] and the shock graph matching of Sharvit *et al.* [195].

The shape context (SC) result presented in Table 4.3 was obtained by applying the Matlab code from [19] to the image data set using the parameter settings detailed in [17]. The result was slightly worse than the score of (25, 24, 23) reported in [17] but it still presented the best classification performance for the chosen 25 silhouette images. As shown in Fig. 4.33a, the few misclassifications were between shapes sharing similar gross shape such as between greebles/rabbits or tools/fishes. The performance of the proposed MGVF-based AHN orientation histogram was comparable to the other more complicated syntactic matching and shock graph matching techniques proposed in [65] and [195] respectively. However, its shape description capabilities is still found wanting, especially in the fishes and tools categories (see Fig. 4.31). Of all the shape descriptors presented in Table 4.3, the Fourier descriptor (FD) in Fig. 4.33b resulted in the largest number of misclassifications. Even though FD is able to categorize all the elongated objects such as the tools and fishes together, it was unable to describe the subtler shape variations that allow tools and fishes to be distinguished. The Zernike moments descriptor (ZMD) on the other hand, had problems with the hand shapes as they varied by means of part articulation and partial occlusion. Being a region-based global descriptor, such shape variations can result in substantial differences in the shape feature vector.

SC ($N = 100$)

Query																										
Match #1																										
#2																										
#3																										

(a)

FD ($N = 64$)

Query																							
Match #1																							
#2																							
#3																							

(b)

ZMD ($n = 10$)

Query																									
Match #1																									
#2																									
#3																									

(c)

Figure 4.33 – Shape classification results for three other shape descriptors applied to the silhouette image database. (a) SC, (b) FD and (c) ZMD. Settings used by these descriptors were similar to those presented in Table 4.3.

4.3.5 Computational Cost

There are two aspects of computational costs presented in Table 4.4. The first compares the time taken to compute all the required twenty-five shape descriptors for the silhouette image database (see Fig. 4.30) for the three MGVF-based shape descriptors (GGON, FTN and AHN), FD, SC and ZMD. All algorithms were implemented in Matlab®.

Shape Descriptor	Shape Similarity Measure	Computational Cost (seconds)	
		Create database	Query database
GGON	$d_\chi (n = 36)$	18.84	0.22
FTN	$d_{L2} (n = 36)$	18.51	0.06
AHN	d_χ and $d_{L2} (n = 36)$	19.00	0.22
SC	d_χ (WBGM, $N = 100$)	42.38	1182.30
FD	$d_{L2} (N = 64)$	1.87	0.06
ZMD	$d_{L2} (n = 10)$	51.60	0.06

WBGM = Weighted Bipartite Graph Matching

Table 4.4 – Computational costs for computing and matching the various shape descriptors. Parameter settings for the MGVF-based descriptors, SC, FD and ZMD are similar to those used in Table 4.3. The times shown were measured on a 1.6 GHz Pentium-4 Personal Computer with 255 MB DRAM, running Windows 98® and Matlab® version 5.3. Matlab codes for building the Gaussian pyramids were based on Simoncelli's pyramid tools available from [203]. Others such as the shape context and Zernike moments were adapted from Belongie's Matlab code available from [19] and Chang's Matlab code available from [45] respectively.

The second set of computational costs shown in Table 4.4 gives the total time required to compare each of the twenty-five query shapes with all the twenty-five descriptors available in the database. This reflects the type of computational cost incurred in producing shape classification results such as those seen in Fig. 4.31. The timing shown assumes the shape descriptor of the query images and all images in the database are already available. While the first set of computational costs describes the cost of extracting the shape descriptors, the second describes the cost required to make queries to the database. For most large image database applications, the more important concern is usually the cost of querying a database as the shape descriptors

need only be computed once at the time the database is created or at the time a new image is added to the database. On the other hand, the query cost is incurred each time a queried shape is compared to each and every available shape descriptor in the database. In querying large image databases, the multiplier effect of the query cost becomes much more significant than the one-time cost of extracting the query image's shape descriptors.

The cost of computing the various types of MGVF-based shape descriptors does not vary significantly. AHN has the highest cost since it requires both the GGON and FTN descriptors to be computed. However, compared to other shape descriptors such as the shape context (SC) and Zernike moments descriptor (ZMD), the computation of the MGVF-based descriptors is noticeably faster, about 2.2 and 2.7 times faster, respectively. The least costly descriptor to compute is the boundary-based Fourier descriptor (FD), which is about 10 times faster than the AHN orientation histogram.

Computational cost for querying a database varied more noticeably among the various MGVF-based descriptors. This is because the global orientation based GGON requires four histogram comparisons per shape matching in order to resolve both the 180° ambiguity and mirror reflection. On the other hand, the FTN feature vector, being invariant to mirror reflection and rotation, requires only one histogram comparison and is therefore about 4 times lower in terms of query cost. Both the FD and ZMD also required one L_2 -dissimilarity distance computation per shape match and are therefore comparable to FTN. Although the shape context produced the best shape classification results as shown in Fig. 4.33a, it did so by incurring a huge computational cost of about 1182 seconds compared to the fractional 0.22 second required by the MGVF-based AHN orientation histogram. The huge query cost of the SC descriptor is due to the weighted bipartite graph matching (WBGM) algorithm used to determine the lowest combinational matching cost of the $N=100$ shape context histograms of the query shape with another 100 histograms of each of the shape in the database. And WBGM is solved in $O(N^3)$ time using the Hungarian method [17]. Since N is the number of sample points along the contour used to capture the characteristics of the shape, it cannot be set too small or else relevant contour variations of intricate shapes (e.g. the hand shapes in Fig. 4.30) may be lost.

4.3.6 Summary of Experimental Results

The various shape analysis experiments presented have highlighted several characteristics of the MGVF-based shape descriptors. The FTN feature vector is invariant to the similarity transform and mirror reflection. If the shapes are slightly elongated, then the global orientation based GGON and MGON descriptors are also invariant to scaling and rotation.

The smoothing parameter α allows one to control the smoothness of the gradient vector field. This feature facilitates robust recognition and classification of shapes in the presence of boundary perturbations, as shown in the shape recognition and boundary texture experiments. The GGON descriptor was shown to be more robust to boundary noise than the FTN descriptor. In order to overcome GGON's limitation with non-elongated shapes, an adaptive switching between GGON and FTN was implemented using the shape elongatedness measure computed from the MGVF. This AHN descriptor was shown to inherit the combined strength of both the GGON and FTN descriptors.

With regards to the histogram comparison measures, the quadratic-form distance was shown to perform better in the presence of boundary noise but the χ^2 -distance proved more accurate in classifying shapes whose orientation histograms were not distinctively different.

In comparison with other shape descriptors, the region-based ZMD was shown to be more robust than the MGVF-based descriptors in recognising shapes in the presence of high boundary noise. However, when applied to the problem of classifying shapes that shared closer topological similarity, the MGVF-based descriptors performed better. This seems to suggest that the orientation histogram is better able to robustly capture the intra-class shape variations than the ZMD. The SC descriptor demonstrated very good performance in the shape classification experiments. However, the huge computational cost it incurred in querying the image database seems to diminish this advantage somewhat. Although not as computationally efficient as the FD, the MGVF-based AHN descriptor demonstrated much better overall performance than the FD in both shape recognition and classification.

4.4 Limitations of the Orientation Histogram

Even though the orientation histogram possesses many characteristics that make it a reasonably efficient, robust and effective global shape descriptor, it still suffers many limitations. This section highlights some of these weaknesses and provides justifications for the part-based shape descriptor that is proposed in the next chapter.

Non-unique shape description – A serious limitation of the orientation histogram is the possibility of obtaining similar descriptors for different shapes. For example, a solid circle shape (such as shape *B* in Fig. 4.34) and a circle with a circular hole at its centre (i.e. a doughnut-like shape) will both produce identical orientation histograms. Such shape description ambiguity can result in shape dissimilarity measures that do not tally with that of human visual perception. Consider the example in Fig. 4.34. Many observers, when forced to pick a closest matching shape amongst shapes *B*, *C* or *D*, would in most cases judge that reference shape *A* (circular band) is visually most similar to shape *D* (square band). Yet the χ^2 -distances of shape *A*'s orientation histogram compared with that of *B*, *C* and *D* indicate otherwise. In fact, shapes *A* and *D* are measured most dissimilar, even more so than between shapes *A* and *C*!

Figure 4.34 – Two shape pairs (A-B and C-D) that have different shape topologies but share fairly similar orientation histogram profiles. The matching scores of reference shape *A* to shapes *B*, *C* and *D* include the GGON orientation histogram's χ^2 -distances (top-right corner, ×1000) and the ZMD's L_2 -distances (bottom-right corner, ×100).

It is presumed that the topological characteristics of the shape are implicitly encoded within the gradient vector field itself during the generation of the MGVF. However, when the shape is non-compact (e.g. shapes *A* and *D* in Fig. 4.34), gradient vectors from different distal regions of the shape are formed with little or no interaction. As a result, the global accumulation of all the gradient vectors' orientation without further regards for their spatial and topological relationship can result in different shapes having similar orientation histograms. This undesirable shape matching result is not just confined to the orientation histogram. Other global shape descriptors such as the region-based Zernike moments descriptor (see ZMD's L_2 -dissimilarity measures in Fig. 4.34) also suffers such ills, albeit for other reasons.

Another example of this shape description ambiguity was seen earlier in the Plane01 shape shown in Fig. 4.28b. In this case, the orientation information arising from various parts of the plane (e.g. fuselage, wing, elevator, etc.), which may be spatially distant from each other, was collectively accumulated into a single orientation histogram. As before, the topological relationship of these distal subparts are not strongly encoded in the global shape descriptor and the resulting orientation histogram can sometimes take on a profile that is similar to that of another very different looking shape, such as Shape08 in Fig. 4.28b. The crux of this non-unique shape description problem is the global approach employed in deriving the shape descriptor. A single and compact descriptor is obtained at the expense of possible ambiguity in shape description. One way to address this issue is to adopt a part-based approach to shape description.

Part-based modification of shape – It is well known that global shape descriptors are sensitive to part-based shape changes such as articulation of limbs, addition/deletion of parts and rearrangements of parts. As expected, the global-based orientation histogram exhibits similar limitations. Consider an example taken from the shape classification experiment of section 4.3.4. Based on the matching of MGVF-based GGON orientation histograms, the two-headed spanner had difficulty finding a low matching score with the other two single-headed spanners in the shape database. As can be observed in Fig.4.35a, even distinctively different shapes like the hand and Greeble had lower dissimilarity measures. In cases like these, proper shape classification results can only be obtained if the shape descriptor can in some way support partial matching of the shape. Unfortunately, this cannot be easily realised with a global shape descriptor like the proposed MGVF orientation histogram.

Another example is illustrated by the query results shown in Fig.4.35b. The query shape underwent limb articulation in the form of a bending about the mid-section connecting two parts of the shape. Notice how this deformation has deteriorated the matching scores of the two-headed spanners even when the human observer can easily classify the various spanners as belonging to the same category. Like the partial matching problem earlier, these issues can only be addressed using a part-based approach to shape description.

Figure 4.35 – Inaccurate shape classification due to part-related shape modifications. (a) The six best ordered matches obtained with query shapes of (a) a two-headed spanner and (b) the same spanner but with a mid-section bend of about 25°. The shape database queried is given in Fig. 4.30. The d_Q -dissimilarity measures shown at the top-right corners ($\times 1000$) were obtained using the MGVF-based GGON orientation histogram. Parameters used were $\alpha = 0.8$, $\sigma_H = 1.5$, $\sigma_Q = 8.0$ and $n = 64$.

Handling non-elongated shapes – Experimental results in the previous section have demonstrated that the MGVF-based descriptors were most robust to boundary noise and intra-class shape variations when the orientation histogram was used in its raw form and not as feature vectors resulting from Fourier transform normalisation. Unfortunately, reliable rotation invariance using the raw orientation histogram was only possible if the shape exhibits some elongatedness. Even though adaptive switching between FTN and GGON was proposed as a means of overcoming this problem, the examples of mismatched shape pairs illustrated in Fig. 4.28b highlighted the persistence of this problem. Some improved method of rotation normalisation must be employed and this issue is also addressed by the part-based descriptor proposed in the next chapter.

4.5 Summary

This thesis hypothesises that a dense gradient vector field derived from the silhouette of an object can provide a robust, effective and efficient description of its visual form. This chapter describes how such a vector field could be used to build a global shape descriptor in the form of an orientation histogram. Various techniques were proposed to make the orientation histogram invariant to rotation, each meeting various degrees of success. Fourier transform normalisation provided an efficient means of obtaining rotation and mirror reflection invariance but it was global orientation normalisation using the MGVF that proved to be more robust against boundary noise. However, this technique did not work well with non-elongated shapes.

The experimental results revealed that the shape analysis performance of the MGVF-based orientation histogram was a good compromise between the various other shape descriptors evaluated. Even though it was not as noise robust as the Zernike moments descriptor, it was better able to handle nuances in shape variation, especially elongated objects. This allowed the orientation histogram to outperform ZMD in all the shape classification experiments, including that involving boundary texture. From the computational cost perspective, the cost of computing the orientation histogram is lower than all the other descriptors evaluated (i.e. SC and ZMD) except the Fourier descriptor. Its performance in the shape classification experiment was comparable if not better than all the other techniques reviewed except the technique that employed the very computationally costly matching of shape context histograms [18]. The query of a large shape database using the proposed MGVF-based global shape descriptor will scale linearly with increase in population size. Each shape in the database can be stored in its orientation histogram equivalent, which means querying an input shape with a very large database of N shapes will essentially incur the computational cost of computing N histogram dissimilarity measures.

Several limitations of the orientation histogram as an effective shape descriptor were highlighted. Many of these weaknesses stem from the fact that it is a global shape descriptor. The next chapter seeks to address some of these limitations by proposing a part-based shape descriptor that combines the medial representation of Chapter 3 with the MGVF orientation histogram proposed in this chapter.

Chapter 5

Part-based Shape Analysis

Psychophysical evidence suggests that human visual analysis of shapes is carried out by first decomposing the shape into parts [56]. Even more surprising is the fact that this is done preattentively, suggesting that part decomposition may affect not just high-level visual processes such as object recognition but also lower-level processes such as figure/ground separation [16],[166] and the perception of symmetry, repetition [16],[56] and transparency [204]. Such findings continue to motivate the psychologist's interest in shape segmentation research as they provide glimpses into the complicated workings of the human visual system [168]. On the other hand, the computer vision research community's interest in part-based shape representations are motivated by a more practical need for a shape descriptor that is robust in the presence of occlusion and commonly occurring part-related modifications such as limb growth and articulation, part addition and deletion.

This chapter describes how the MGVF medial representation and orientation histograms proposed in chapters 3 and 4 are employed in the development of a part-based shape descriptor. Section 5.1 begins with a brief review of relevant issues related to part-based shape analysis and section 5.2 then outlines the guiding principles that have motivated the various strategies employed in the proposed MGVF part-based descriptor. Sections 5.3 and 5.4 describe how the extracted MGVF skeletons are processed in order to decompose the shape into skeletal parts. These skeletal parts are then associated with meaningful descriptors and the extraction of these part features are discussed in section 5.5. The issue of visual saliency is discussed in section 5.6. Section 5.7 describes how the proposed part descriptors can be used to compute a dissimilarity measure between two parts and how such individual part-to-part matching costs can be subsequently combined to derive a single measure that reflects the dissimilarity between two shapes. A comprehensive set of experimental results is presented in section 5.8 to highlight and discuss the strengths and weaknesses of the proposed part-based descriptor.

5.1 Part-based Shape Analysis – A Review

Unlike the global descriptors discussed in chapter 4, a part-based shape descriptor is local in nature and would therefore require some process in which the complex shape is first decomposed into simpler subparts before each associated part descriptor can be extracted. How does the human visual system perform shape decomposition? There are essentially two main schools of thought with regards to this issue. The first postulates that there is a set of *shape primitives* with which the human visual system searches for in an image. A shape is decomposed into these simpler primitives by means of some matching strategy, as shown in Fig. 5.1a. In essence, shape primitives are responsible for both part decomposition and description. Primitives such as generalised cylinders [28],[143], superquadrics [164] and geons [25] have been proposed. Each set of these primitives works well on a selected class of objects but is unable to capture the variety and complexity of object parts in general.

Figure 5.1 – The two main approaches to shape decomposition. (a) Decomposition based on shape primitives such as Biederman's geons [25] and (b) that based on some geometric partitioning rules, such as the minima rule [95]. Adapted from [26], [198].

The second school of thought postulates that there is some set of basic geometric rules by which human vision computes the part boundaries of a complex shape. The most well known of these is the *minima rule* proposed by Hoffman and Richards [95], which states that the negative minima of curvature provides boundary points for partitioning the shape, as illustrated in Fig. 5.1b. They argued that the mechanism that performs part decomposition is more basic and therefore operates regardless of the shape of the part. In this sense, the rule-based approach is

fundamentally different from the shape primitive approach as part decomposition precedes part description [198].

Singh *et al.* [205] highlighted that the minima rule does not define how shape regions are partitioned as it merely locates boundary point constraints through which the part cuts must occur. They proposed the *short-cut rule*, which required silhouettes to be segmented into parts using the shortest possible cuts that intersect a local symmetry axis within the shape (see Fig. 5.2a). These shortest cuts must be straight lines that join two boundary points of which one or more must be located on a negative curvature. Singh *et al.* argued that Siddiqi and Kimia's [198] scheme of shape partitioning based on *limbs* and *necks* (see Fig. 5.2b) was too restrictive and would fail to partition many shapes properly.

Figure 5.2 – Different schemes use to determine part cuts. (a) The partitioning based on the short-cut rule [205] results in the part cuts shown above as opposed to the alternative shown below. (b) Partitioning based on limbs and necks (for definition of limbs and necks, see [198]). Adapted from [205], [198].

Rosin [177], in turn, argued that the short-cut rule incorporates very limited global shape information and the requirement of having to cross a local axis of symmetry using only straight cuts was unduly restrictive. Instead, Rosin proposed a part decomposition scheme based on maximising the overall shape convexity. This scheme is not without its limitations as optimal cuts are computationally costly to determine and the appropriate number of cuts for a given shape usually needs to be known before hand. Like Rosin's explicit use of convexity, Latecki and Lakämper [124] proposed decomposing the shape boundary into only convex parts, arguing that it is the convex parts of an object that determines visual parts [14], [95], [117]. They proposed identifying convex parts at different stages of a contour evolution process thus providing a hierarchical decomposition of the shape.

Figure 5.3 – Problems with shape decomposition strategies that use only boundary information. (a) Boundary-based part descriptors can often be arranged in a different combination to give a total new shape [24]. (b) The partitioning scheme based on the reaction-diffusion space of [111] produces more natural shape decomposition than that suggested by the minimal rule of [95]. Adapted from [24], [198].

Approaches such as Latecki's relies primarily on boundary curvature information. Indeed, the computational efficiency of processing 1D contours has made such an approach to shape decomposition popular, as can be seen in works such as [24], [65], [84], [167]. However, several researchers [2], [111], [177] have argued that reliable and meaningful shape decomposition and description requires the integration of both boundary and region information. Fig. 5.3a illustrates a case in point, where boundary information alone can result in two very different shapes having a seemingly good part-to-part match. Kimia *et al.* [111] also illustrated an example shown in Fig 5.3b, which suggest that boundary-based partitioning rules such as the minimal rule [95] do not always produce perceptually acceptable results.

Skeleton-based shape representation as popularised by Blum [30] provides a more convenient mechanism for integrating both boundary and region information during shape decomposition. These skeletons are defined by a set of connected points of local symmetry of the region. Like Latecki and Lakämpers' proposition in [124], symmetry-based decomposition also leads to convex part components [139]. Partitioning schemes like Abe *et al.* [2] make explicit use of both boundary and region

information as the skeletons (‘*spines*’) are segmented by taking into account dominant points of negative curvature on the contour. Others like di Baja and Thiel [13] decomposed a shape by breaking their skeletal branches at junctions. They then applied polygonal approximation of each branch so that these continuous segments are further decomposed into skeletal segments that define parts with linearly changing widths and orientation.

Figure 5.4 – Shape decomposition based on segmenting skeletons. (a) Liu and Geiger’s SA-tree representation of parts is derived by connecting discontinuous shape axes extracted via the detection of self-similarity along boundary contours [136]. (b) Kimia and Siddiqi obtained more refined skeletal partitioning and description by using a dynamic interpretation of medial axes as flowing singularities (shocks). The dumb-bell shape has two “seed” parts (4th-order) connected at a “neck” (2nd-order), each part having three protrusions (1st-order) [199]. Adapted from [136], [199].

The process of skeleton extraction is sensitive to boundary perturbations, often resulting in numerous spurious branches. As a result, di Baja and Thiel introduced an additional stage of processing where short and superfluous segments were removed. Unfortunately, the task of determining which branch is visually insignificant and should be deleted is non-trivial and often dependent on some arbitrary threshold setting. Ogniewicz and Kübler [156] used a multiresolution framework of hierarchical Voronoi skeleton to address this issue but the choice of appropriate scale is often just as difficult to determine. Others like Rom and Medioni [175] and Liu and Geiger [135] avoided this problem by computing only a subset of the medial axes as defined by Blum [30]. In their case, only points of local symmetry between boundary segments sharing mirror likeness are considered. As shown in Fig. 5.4a, these skeletal

segments (*'shape axes'*) are disjointed and their significance or saliency can be determined by inspecting their relative length.

Kimia *et al.* [111] proposed a mathematical framework based on the reaction-diffusion equations, where the medial axis is obtained and described by the development of first, second, third and fourth order shocks. The dynamic information associated with shocks allow a more refined partitioning of the skeleton as boundary characteristics such as changes in shape (e.g. at necks [198]) allow a continuous skeletal segment to be further partitioned into more appropriate parts (see Fig 5.4b). The resulting shock graph is a richer descriptor than a medial axis graph, as its topology is more in keeping with our perceptual notion of shapes [200].

Works such as those of Siddiqi *et al.* [201] exploited such skeletal characteristics to provide a rich structural description of the shape. They labeled points on the skeleton using what they called *shock-labels*. Giblin and Kimia [69], on the other hand, adopted a simple structural representation based on just the junctions and terminations of the skeleton. With such representations, shapes can be matched by comparing their differing structural descriptions. Pelillo *et al.* [163], for example, used a sub-tree matching technique. Their shock tree is labeled with the length, radii, velocities and curvatures of the shocks. However, this method is vulnerable to structural variations, which can easily arise due to ligature instability caused by boundary perturbations (see Fig. 5.5b).

Figure 5.5 – Robust matching of shocks in the presence of boundary noise requires the pruning of spurious edges on the graph of the noisy shape. (a) A clean triangle displays the three significant shock branches, which define a triangle. (b) Noise due to boundary discretisation effects, for example, will result in many spurious edges. Adapted from [187].

Siddiqi *et al.* [201] attempted to address this problem by labeling their shock graph with additional attributes such as the time of formation of the shock. The idea here is to increase the hierarchy of shocks (and thereby their significance) the later they are formed. Torsello and Hancock [216] argued that time of formation alone is not an effective measure of branch relevance in the presence of sharp boundary structures (such as the triangle in Fig. 5.5a) and high-curvature features. This is because a skeletal segment that starts at a vertex will always have the earliest time of creation, regardless of its relative significance to the shape. This thesis shares this view and an alternative measure is proposed to quantify the saliency of a skeletal segment (see Fig. 5.9) and this is further discussed in section 5.6.1.

Sebastian *et al.* [191] proposed a more robust technique for tackling this problem during the shape recognition stage. Their shock graph edit-distance algorithm reduces the matching cost contributions of such detractors by applying a *splice* operation to the shock graph during matching. The total splice costs required to transform the shock graph of the noisy triangle in Fig. 5.5b to that of the clean triangle in Fig. 5.5a is low since their corresponding boundary differences are minimal. However, the numerous spurious shock edges that need to be analysed can significantly increase computational cost, as admitted in [187]. They subsequently proposed using a coarse-to-fine matching strategy to match shapes in large shape databases [189], where the coarse matching only computes the deformation cost in terms of the end points of the shock graph path instead of the more accurate optimal alignment (via dynamic programming) between these graphs.

A major drawback of this method is that it requires alignment and explicit comparison of the boundary [216]. This means the shape representation in the database cannot be stored as compact descriptors associated with each skeletal-based part since the shape boundary information must be made explicitly available during shape matching. The part-based descriptor proposed here does not share this problem. Once the shape descriptors associated with each identified skeletal part has been extracted, the original binary shape image, the extracted MGVF skeleton and the MGVF itself can all be discarded. The set of part-based descriptors are all that is required to represent the shape and perform the necessary shape matching during a database query.

5.2 MGVF Part-based Shape Analysis

The medial representation approach to shape analysis has been shown to be effective, as highlighted in the review. A recent paper by Kimia [113] provides a comprehensive discussion of the role of medial geometry in the human visual system. Neurophysiological experiments in [125], [126] observe peak neural activity in the V1 cells at localities of medial axis and psychophysical experiments such those in [119] suggest that we may use medial points in describing shapes. Based on such motivations, the proposed MGVF part-based shape analysis technique also employs a medial representation as the underlying framework for part decomposition and description. Several propositions have guided the strategies adopted in the development of this shape analysis technique and they are discussed here.

Proposition 1 – *The part decomposition of a shape is a dynamic and adaptive process.* As discussed in the review, the traditional view of part decomposition is that it is governed by some fixed rules, such as the minima rule [95], for example. Based on such rules, a shape is decomposed into parts. The dissimilarity between two shapes is then computed by finding the optimal combination of matches between these parts. Shapes, however, can deform in a continuous manner and it is often difficult to decide how a shape should be decomposed at certain transitions in this deformation process (see Fig 5.23). Moreover, Fig 5.6 illustrates how one would intuitively decompose the same shape **A** in different ways when forced to match it to different shapes.

Figure 5.6 – The part decomposition of a shape is often dependent on the other shape that it is being compared to.

In the light of this proposition, the strategy employed in this work is to segment the MGVF skeleton of a shape **A** into basic fragments (termed *primary skeletal segments*) using various segmentation criteria. These skeletal fragments are subsequently

merged back into longer segments using a combination of merging criteria. Since the merged segments can be reconstituted with varying combinations of the basic skeletal fragments (see Fig. 5.30), these different combinations essentially describe all the possible part decomposition options available in shape *A*. The actual decomposition option is not decided until during the part matching stage when information regarding shape *B* is available and incorporated into the decision making process. This strategy is somewhat similar to the edit-distance algorithm of Sebastian *et al.* [191] and other graph topology changing operations employed by works such as [136], [234] to deal with the shape deformation arising from articulation, occlusion and limb growth. The optimisation algorithms employed by these authors essentially modify the graph-based representation of the shape's skeleton through splice and merge operations, which implicitly modifies the part decomposition of the shape.

Proposition 2 – Robust part-based description of shapes should combine both local and global shape measures. Part descriptors employing only local features cannot capture the coherent structure of a shape. Based on local measures alone, the two obviously dissimilar shapes **A** and **B** in Fig 5.7a would be considered a good match as all subparts in one have a local representation in the other.

Figure 5.7 – Drawbacks of purely local and purely global shape measures. (a) Two dissimilar looking shapes that share identical local part features. (b) Global geometry allow us to group shapes sharing similar parts into different object categories. (c) Changes in the shape geometry due to part articulation are problematic for shape descriptors using global features. The global geometry of thin man **A** is closer to fat man **C** than thin man **B**. A human observer has no problem classifying both thin men **A** and **B** as being similar despite their varying arm positions.

Another example is the different pairs of spectacles and dumb-bells shown in Fig 5.7b. A human observer has no problem telling apart these two categories of objects even though they share similar parts. These types of ambiguity can be resolved by incorporating more global measures regarding the inter-part relationship within the shape. However, global measures do have their drawbacks as the resulting descriptor becomes sensitive to partial occlusion and limb articulation, as shown in Fig 5.7c. The proposed part-based descriptor incorporates both local and global characteristics when computing the dissimilarity between two shapes. The former as a measure called *part distance* and the latter as a measure called *global distance*. The relative contribution of these two independent measures to the overall *shape distance* can be adjusted through appropriate weighting. Another strategy that has been employed to help alleviate matching problems such as those seen in Fig. 5.8 is the use of the connection histogram (see section 5.5.2). This local orientation histogram at the end of a skeletal segment captures the connectivity characteristics at skeletal junctions.

Figure 5.8 –As suggested in [191], graph-based techniques such as [201] that do not capture the planar order at nodes will not be able to distinguish between shapes **A** and **B**. The incorporation of local connectivity information at skeletal junctions can improve the dissimilarity detection. Notice the connection histograms at the same skeletal segment in the two shapes are not similar. Shapes were adapted from [191].

Proposition 3 – The presence of ligatures and their effects on the skeletal topology must be explicitly handled by both the decomposition and shape matching processes. Skeletonization algorithms are notorious for their sensitivity to boundary perturbations, which leads to unstable ligatures [11]. Although such ligatures can be reduced in the proposed MGVF skeleton through appropriate choices of smoothing and connectivity parameters, they should not be eliminated altogether. This is because in reality, one cannot always decide if a skeletal segment should be considered an insignificant ligature or a principal component of the structural skeleton (see Fig.

5.9a). To compound this problem further, ligatures can have a significant influence on the skeletal topology. They create branch points that invariably influence the way a shape is decomposed. The proposed part-based shape analysis technique addresses these issues in two ways. Firstly, a measure based on the enhanced vector field directional disparity measure introduced in (3.12) is employed to quantify the saliency of every skeletal segment (see section 5.6). As shown in Fig. 5.9b, unstable ligatures express themselves as segments with low disparity values and these unstable parts are regarded as less salient and thus given little weightage in the shape matching process. Secondly, after the MGVF skeleton has been decomposed into various skeletal-based parts using various segmentation criteria (including segmentation at branch points), a merging process is employed to re-group these segmented fragments into visually continuous segments. This merging process uses this same disparity-based measure to decide how skeletal segments should be merged at branch points (see section 5.4). Such a merging criterion will ensure that the principal components of the structural skeleton are re-grouped despite their initial separation due to the presence of the ligatures (see Fig. 5.9c). The resulting shape decomposition would be more in line with that perceived by the human observer.

Figure 5.9 – Unstable ligatures in (a) may or may not be formed depending on the pre-selected value of the smoothing and connectivity parameters given by α in (2.5) and λ in (3.17), respectively. (b) The enhanced disparity map $M_E(x,y)$ provides a good measure of the formation stability of a skeletal segment. (c) The impact of ligatures on the skeletal topology can be ameliorated using appropriate merging criteria to recombine skeletal fragments segmented by their presences.

The various stages of the proposed MGVF part-based shape analysis process are outlined in Fig. 5.10. Chapters 2 and 3 have addressed the issues of MGVF generation and the robust extraction of the MGVF skeletons respectively. This chapter discusses the process of extracting skeletal parts and the subsequent computation of a set of useful descriptors for each part. Chapter 6 describes the computation of a part-based shape dissimilarity measure using these part-based descriptors.

Figure 5.10 – Overview of the entire MGVF part-based shape analysis process.

The skeletal part extraction process itself consists of various stages as shown in Fig. 5.11. Given the MGVF skeleton of a shape as an input, the output of this stage of the analysis is an exhaustive set of secondary skeletal segments. This set of skeletal parts describes all possible combinations of basic skeletal segments taking into account the visually continuous segment created by the merging process highlighted in Fig. 5.9. This set of descriptor is an over-complete set and the lowest cost combination will be determined by the part-based shape matching algorithm described in chapter 6.

Figure 5.11 – The various stages required to extract a set of secondary skeletal segments from the MGVF skeleton of shape A.

Once a set of secondary skeletal segment has been extracted, a suitable set of shape descriptors for each one of this skeletal part can then be computed. A summary of these descriptors is shown in Fig. 5.12. These descriptors are used in the shape matching process described in chapter 6 and the contribution of each part descriptor to the overall matching cost between two shapes is weighted by each part's visually saliency, which is computed using a weighted combination of several saliency measures.

Figure 5.12 – An overview of the process for computing the descriptors of each skeletal part and their respective visual saliency measures.

5.3 Skeleton Segmentation

Since the proposed part representation is based on the extracted MGVF skeleton, the manner in which the shape is decomposed into its constituent parts correspond directly to the manner in which the skeleton is segmented. This section describes the various segmentation criteria adopted in the skeletal decomposition process.

5.3.1 Skeletal Junction Segmentation

The most obvious means of part decomposition is to segment the skeleton of a shape at every skeletal junction (or branch point) into J continuous skeletal segments S_j , each being described by an ordered set of 2D points given by

$$S_j = \{p_j(i)\}_{i=1}^{N_j} = \{[x_j(i), y_j(i)]\}_{i=1}^{N_j} \quad \text{where } j \in [1, J] \quad (5.1)$$

and N_j is the number of discrete points along the skeletal segment S_j . This segmentation process is depicted in Fig. 5.13. The junction-based segmentation is performed on both the structural and textural skeletons.

Figure 5.13 – Skeleton segmentation at skeletal junctions. Each part consists of a continuous skeletal segment that is segmented at branch points.

Simple junction-based segmentation of the skeleton is sufficient for shapes whose parts are essentially described by extended limbs or protrusions (see Fig. 5.14a). Earlier versions of this work presented in [71],[73],[74] used this simple part decomposition strategy with some limited success when performing shape classification of a small 25-shapes database used by [18],[65],[195] (see Fig. 4.30). However, this method of segmentation is inadequate for shapes or parts whose boundary profile may vary significantly while displaying no branching along their skeleton, as shown in Fig. 5.14b. The segment histogram (discussed in section 5.5.2) describing the shape of each part is computed by accumulating the orientation

histograms along the skeletal segment representing each part. Since no segmentation points are detected in these shapes, the segment histogram of their respective singular skeletal part is effectively a global shape descriptor. Part-based modifications represented by limb width variations like bulges are therefore not properly described.

Figure 5.14 – (a) Shapes whose part decomposition corresponds well with segmentation at skeletal junctions. (b) Shapes that are represented by a single continuous skeletal segment (blue). Segmentation at skeletal junctions will not provide an adequate means of describing the constituent parts of these shapes.

5.3.2 Vector Field Disparity Profile Maxima/Minima Segmentation

Since segmentation at skeletal junctions does not adequately describe limb width variations along a continuous skeletal segment, more refined segmentation must be imposed on each of these segments. Interestingly, the characteristics relating to limb width variations along the skeleton are encoded within the vector field disparity map $M(x,y)$ given in (3.9). Fig. 5.15 shows the disparity map $M(x,y)$ for several shapes and the associated disparity profile along their respective skeletal segments.

Figure 5.15 – The vector field disparity profile along the skeletal segments of four different shapes. The disparity map $M(x,y)$ within each shape is also shown.

The minimas or troughs in the vector field disparity profile characterise skeleton positions where there are transitions in limb width. As highlighted in the sting-ray shape shown in Fig. 5.16a, the disparity profile minima corresponds to the start of a limb extension in the object. On the other hand, maximas or peaks in the disparity profile correspond to the central positions of limbs and bulges. They also correspond to positions of narrow necks in an object (see Fig. 5.16b).

Figure 5.16 – (a) Minimas in the vector field disparity profile along a continuous skeleton provide convenient points to segment limbs from the main body and (b) positions of maxima locate features such as necks and the centre of limb bulges.

Shapes with no obvious limbs such as those shown in Fig. 5.17 will not exhibit any minima in the vector field disparity profile along their structural skeleton. However, a maxima will always exist. Segmentation at this peak ensures that the structural skeleton is partitioned, thus allowing both connection and terminus nodes to exist. These nodes encapsulate a richer description of the shape. Otherwise, the single structural skeleton is only described by terminus information at its two ends. This situation is undesirable as terminus characteristics vary substantially with only minor changes to the shape boundary, as shown in the two heart shapes in Fig. 5.17. In short, the maximas and minimas in the vector field disparity profile provide a meaningful means for more refined segmentation along a continuous skeletal segment.

Figure 5.17 – The segmentation of the skeleton at the maxima of the disparity profile creates a connection point whose orientation histogram provides additional information for differentiating the two heart shapes from that of the face. Terminus information alone makes **Heart B** look more like the **Face** than **Heart A**.

The vector field disparity profile along a continuous segment S_j in (5.1) is given by

$$DS_j(i) = M(S_j) = M(\{p_j(i)\}_{i=1}^{N_j}) \quad (5.2)$$

where the ordered set of discrete points $\{p_j\}$ of S_j provides the set of 2D coordinates needed to extract the 1D disparity profile from the disparity map $M(x,y)$ in (3.9). The positions of disparity maximas and minimas are found by detecting zero-crossing points along the first-order derivative of $DS_j(i)$ and this set of indices is given by

$$ZC_j = \arg \text{zcd}_{1 \leq i \leq N_j} [\nabla g_{\sigma_z}(i) * DS_j(i)] \quad (5.3)$$

where $\text{zcd}[\ ]$ is an operator which returns the set of indices along an ordered set of real values where a sign change occurs between two consecutive values. If there are no maximas or minimas in the disparity profile, an empty set is returned. This operator is more commonly known as the *zero-crossing detector*. The convolution kernel ∇g_{σ_z} is a first-order Gaussian derivative given by

$$\nabla g_{\sigma_z}(i) = -\frac{i}{\sigma_z^2} \exp\left(-\frac{i^2}{2\sigma_z^2}\right) \quad (5.4)$$

A large value of σ_z can provide smoothing along the disparity profile so as to avoid the detection of insignificant zero-crossings due to small peaks and troughs variations caused by boundary perturbations. This problem occurs when the width variations along the limb remain constant except for these boundary perturbations. An example is the teeth serrations on the key shown in Fig. 5.18.

In summary, the skeletal segment S_j is segmented at points where its vector field disparity profile exhibits maximas and minimas. This set of points, Pd_j is obtained by substituting the set of indices obtained in (5.3) into (5.1), and is given by

$$Pd_j = \{p_j(ZC_j)\} \quad \text{and} \quad Pd_j = \phi, \text{ if } ZC_j = \phi \quad (5.5)$$

Figure 5.18 – The zero-crossing points extracted with different amounts of smoothing in the first-order Gaussian derivative kernel. (a) A large value of $\sigma_z = 5.0$ suppresses the detection of insignificant peaks and troughs in the disparity profile. (b) A smaller value of $\sigma_z = 1.0$ may result in over-segmentation of the skeletal segment.

5.3.3 Connectivity End Point Segmentation

The structural and textural distinction of the MGVF skeleton facilitates robust and consistent structural skeleton extraction in the presence of boundary artifacts. However, connectivity is established within the structural skeletons via the connectivity factor $\lambda^\dagger$. Unfortunately, there are situations where a consistent choice of λ and the smoothing parameter α in (2.5) will result in inconsistent structural skeleton connectivity. This problem is illustrated in Fig. 5.19, where two shapes with only minor boundary variations can result in different skeletal connectivity.

[†] see section 3.3.4 for details

Figure 5.19 – Variable skeletal connectivity. Despite using identical values of $\lambda = 1.0$ and $\alpha = 0.9$, different skeletons for **Heart X** and **Heart Y** were obtained after applying the connectivity algorithm[†]. Segmentation at the connectivity end point allows better matching between relevant skeletal segments. The issue of the unmatched connectivity segments is handled using appropriate part saliency (see section 5.6.3).

In the light of this problem, the skeletal-based matching of shapes can be made less sensitive to values of λ and α by introducing additional segmentation points. As shown in Fig. 5.19, any continuous skeletal segment S_j that has skeletal points appended using the directional-uphill climbing algorithm given in (3.15) will be segmented at the respective connectivity end points. A *connectivity end point* is a 2D coordinate on S_j from where a directional-uphill climbing operation was previously initiated in order to establish connectivity in the skeleton. Not all skeletal end points are considered valid connectivity end points since a connectivity test is performed with an appropriate value of λ to determine if directional-uphill climbing should be applied to that particular end point. In summary, the continuous skeletal segment S_j is also segmented at points where there are valid connectivity end points. This set of 2D points Pe_j is given by

$$Pe_j = \{e_j(i)\}_{i=1}^{E_j} \quad \text{if } E_j \geq 1$$

$$Pe_j = \phi \quad \text{otherwise} \quad (5.6)$$

where E_j is the number of elements in the set of valid 2D connectivity end points $\{e_j\}$ that was used in the construction of the continuous skeletal segment S_j in (5.1).

5.3.4 Skeletal Curvature Maxima/Minima Segmentation

Points on a skeleton that exhibit high skeletal curvature maxima or minima are often a good indication of a part transition in the shape, such as in the examples shown in Fig. 5.20a. Sharp boundary curvature associated with these skeletal points can sometimes generate sufficiently significant textural skeletons, which will subsequently segment the skeleton at these curvature maxima or minima, as shown by shapes **Calf 1** and **L 1** in Fig. 5.20a. However, such connectivity is not guaranteed and is too dependent on boundary characteristics and the value of the connectivity factor λ . As such, an additional process is required to analyse the curvature of the skeleton so that points on the skeleton that exhibit absolute local curvature maxima or minima above a predetermined threshold τ_c , will be identified as points at which the skeletal segment S_j will be further segmented, as shown in Fig 5.20b.

Figure 5.20 – Part decomposition based on skeletal curvature. (a) Some shapes exhibit natural part decomposition at points where the skeletal curvature is sharp. Rounded boundary can prevent the formation of skeletal junction at such localities thus resulting in poor segmentation (shapes **Calf 2** and **L 2**). (b) Segmentation of skeleton at points where skeletal curvature exhibit large maxima and minima.

To detect these high curvature points, the local skeletal orientation at each discrete point $p_j(i)$ on skeletal segment S_j is first estimated. This ordered sequence of angular values is then unwrapped to prevent the undesirable 2π step jump when local orientation along S_j crosses the $0 \rightarrow 2\pi$ or $2\pi \rightarrow 0$ boundary. The skeletal curvature $\nabla \theta_j(i)$ is represented by the normalised first-order derivative of the resulting unwrapped angular sequence given by $\theta_j(i)$ and is computed using

$$\nabla \theta_j(i) = \frac{1}{\pi} [\nabla g_{\sigma_c}(i) * \theta_j(i)] \quad (5.7)$$

where ∇g_{σ_c} is a first-order Gaussian derivative convolution kernel given in (5.4) but with a standard deviation of σ_c . Normalisation by a division of π ensures $\nabla \theta_j(i) \in [0,1)$. The index locations of all local curvature maximas and minimas along S_j can be obtained by applying the zero-crossing detector in (5.3) and is given by

$$AC_j = \arg \underset{1 \leq i \leq N_j}{\text{zcd}} [\nabla g_{\sigma_z}(i) * \nabla \theta_j(i)] \quad (5.8)$$

In this case, σ_z can be set to a small value of 0.5 as noise filtering of the local orientation estimates can be achieved via σ_c in (5.7). The subset of these indices where the absolute skeletal curvature maximas and minimas are greater than the threshold τ_c is given by

$$HC_j \subseteq AC_j, \text{ where } |\nabla \theta_j(AC_j)| \geq \tau_c \quad (5.9)$$

Fig. 5.21 summarises the process for detecting skeletal curvature maxima and minima points and this set of 2D points Pc_j , along the skeletal segment S_j is given by

$$Pc_j = \{p_j(HC_j)\} \quad \text{and} \quad Pc_j = \phi, \text{ if } HC_j = \phi \quad (5.10)$$

5.3.5 Primary Skeletal Segments

Four skeleton segmentation criteria have been discussed so far, namely segmentation at skeletal junctions, at connectivity end points and at the maxima/minima points of the vector field disparity profile and skeletal curvature. The complete segmentation process is summarised here. It is first assumed that the MGVF skeleton has been obtained with the technique described in section 3.3 and skeletal connectivity analysis and connectivity has been performed on the raw skeleton with a chosen value of connectivity factor λ . The MGVF structural and textural skeletons are first decomposed into J continuous skeletal segments S_j , by segmenting at all skeletal branch points. For each S_j , its associated set of segmentation points $\{a_j(i)\}$ is given

$$\{a_j(i)\}_{i=1}^{A_j} = Pd_j \cup Pe_j \cup Pc_j \quad (5.11)$$

This is the union of the 2D point sets Pd_j in (5.5), Pe_j in (5.6) and Pc_j in (5.10). The segmentation of skeletal segment S_j results in a set of $A_j + 1$ smaller segments given by

$$\{S_j^p\}_{p=1}^{A_j+1} \quad (5.12)$$

These segmented skeletal fragments are termed *primary skeletal segments*, as they cannot be further fragmented. The segmentation process applied to the entire MGVF skeleton results in a complete set of primary skeletal segments $\{SP_k\}$ given by

$$\{SP_k\}_{k=1}^{Np} = \bigcup_{j=1}^J \bigcup_{p=1}^{A_j+1} S_j^p \quad \text{where } Np = \sum_{j=1}^J A_j + 1 \quad (5.13)$$

Some examples of this segmentation process are shown in Fig. 5.22.

Figure 5.22 – Segmentation of some MGVF skeletons into a set of primary skeletal segments. These results were obtained using the following parameters: $\lambda = 1.0$, $\alpha = 1.0$, $\sigma_z = 5.0$ (for disparity min/max), $\tau_c = 0.1$ and $\sigma_c = 1.5$.

5.4 Skeleton Merging

Segmentation based on the criteria stipulated earlier can be considered a process that is driven by a ‘hard’ threshold. In other words, the detection of a segmentation point based on a maxima or minima in the vector field disparity profile, for example, is either a yes or a no decision. But shapes can be distorted in a continuous manner as shown in Fig. 5.23, thus giving rise to a detectable (or threshold exceeding) minima/maxima at some crossover instants in the deformation. The use of a ‘hard’ segmentation threshold means that the matching of shapes **A** and **B** produces a much better match than that between **B** and **C**. In the case of the former, the primary skeletal segment count in both shapes is the same. Yet most observers would consider the visual differences between the $\mathbf{A} \Rightarrow \mathbf{B}$ and $\mathbf{B} \Rightarrow \mathbf{C}$ to be hardly distinguishable. In order to overcome this problem, segmented primary skeletal segments must be allowed to merge. The subsequent skeletal matching process should make a decision to use the merged or unmerged segments based on whichever produces a lower matching cost. In this way, one is able to get around the ‘hard’ threshold introduced in the skeleton segmentation stage. Merging of skeletons also reduces the permitted combination of primary skeletons (see Fig. 5.31) and this will lead to a more efficient shape matching process as the search over all possible combination of skeletal segment will be greatly reduced. But how should primary skeletal segments be merged? Merging of skeletal segments is straightforward for the shapes in Fig. 5.23 or the **UFO** in Fig. 5.22, where no ‘T’, ‘Y’ or ‘+’ skeletal junctions exist but this is not so for most general shapes.

Figure 5.23 – Hard segmentation criteria or the use of segmentation thresholds do not reflect the fact that shapes can undergo continuous deformation. For example, minimas and maximas in the vector field disparity profile along a skeleton can appear or disappear with only minor variation in the width of the beak-like limb.

This problem of inconsistent skeleton segmentation is indeed an inherent weakness of all medial representation of shapes. Small shape variations due to part rearrangement, deletion and addition (see Fig. 5.24a) or boundary perturbation (see Fig. 5.24b) can have a significant effect on the outcome of the skeletal segmentation process, even one that is based solely on skeletal junction decomposition. Even though the general appearance and classification of the shape is not perceived to have changed much, the number of primary skeletal segments extracted can vary significantly between these shapes. The proposed MGVF-based skeleton extraction algorithm is relatively robust against boundary perturbation compared to other algorithms such as Blum's medial axis transform (MAT) [30], but it is still vulnerable to part rearrangement, addition, deletion and deformation. Moreover, the segmentation results are sometimes dependent on the choice of parameters such as the MGVF smoothing parameter α and the connectivity factor λ . An example of which can be seen in Fig. 3.20.

Figure 5.24 – Characteristics of medial representations. (a) Sensitivity to part deletion and rearrangement. (b) Instability of MAT [30] to boundary perturbations. (c) Medial points with high degree of substance and connection. Adapted from [107].

Indeed, Katz and Pizer attempted to address this very problem in [107]. They argued that medial representations of shapes embody both the substance of each object part and the connections between adjacent parts, as shown in Fig. 5.24c. Small boundary changes result in small changes to its substance but may involve significant changes in its connection characteristics, as highlighted in the example in Fig. 5.24b. Since most skeleton extraction algorithms do not explicitly represent these dual characteristics, it is often difficult to decompose such skeletal structures into a

hierarchy of parts that reflect the natural part decomposition that we perceive. This makes the subsequent matching and recognition of shapes using skeletal information difficult and unstable.

Katz and Pizer devised normalised measures to express the degree of conductance (ω) and substance ($\psi = 1 - \omega$) at every medial point. Neighbouring points that have a substance component that is much greater than their connection component are considered belonging to the same part of the object and are merged into a continuous segment. Since our visual system tends to follow the main medial path of an object while ignoring less significant off shoots, the connection and substance components of axes points in the vicinity of a branch point is computed based on this visual path continuity. They proposed using the continuity in the direction of the medial axes and the continuity of scale along the axes as the two important factors determining *visual conductance*, which they defined as the measure of how likely any two axes at a branch point will be perceived as the single, main perceptual path. An example result produced by Katz and Pizer is shown in Fig. 5.25.

Figure 5.25 – (a) The extracted medial axis of a lizard object and (b) its computed substance measure. The height of each medial point reflects the value of the substance measure, which is reasonably uniform along the lizard's backbone. Taken from [107].

5.4.1 Merging Criteria

In this work, two visual continuity measures are used to decide which pair of primary skeletal segments at a branch point should be merged. They are the orientation continuity and the disparity continuity. Firstly, consider a primary skeletal segment SP_i that is connected to at least two other primary skeletal segments at branch point b . SP_i is defined by an ordered set of 2D points given by

$$SP_i = \{p_{SP_i}(n)\}_{n=1}^{N_{SP_i}} \quad (5.14)$$

where the set contains N_{SP_i} discrete 2D points and the first point in the ordered set, given by $n = 1$, is the branch point $\mathbf{b}$.

5.4.1.1 Orientation Continuity

This measure is similar to that adopted by Katz and Pizer [107] and is based on the continuity of the local skeletal orientation of the two adjoining skeletal segment pairs at the branch point. The orientation continuity between two primary skeletal segments SP_i and SP_j connected at branch point $\mathbf{b}$ is given by

$$CO_{\mathbf{b}}(SP_i, SP_j) = \frac{|\pi - |\theta_i(\mathbf{b}) - \theta_j(\mathbf{b})||}{\pi} \in [0, 1] \quad (5.15)$$

where $||$ gives the absolute value and $\theta_i(\mathbf{b}) \in [0, 2\pi)$ is the local orientation estimated about branch point $\mathbf{b}$ using the subset of consecutive skeletal points on SP_i defined by the set $\{\mathbf{p}_{SP_i}(n)\}_{n=1}^{N_{CO}}$, as shown in Fig. 5.26. The gradient of the straight line fitted to the N_{CO} skeletal points gives a reasonable estimate of the local orientation of SP_i at point $\mathbf{b}$. A minimum of $N_{CO}=2$ is required for a least-squares line fit but a $N_{CO}=4$ was used to provide a more robust estimate.

Figure 5.26 – Orientation continuity between a pair of skeletal segments at the branch point $\mathbf{b}$ is computed using their respective local orientation estimates at $\mathbf{b}$.

5.4.1.2 Disparity Continuity

The second measure of visual continuity makes use of the MGVF and is based on the similarity of the enhanced vector field disparity measure along the pair of primary skeletal segments at branch point $\mathbf{b}$. Fig. 5.27 shows the enhanced vector field (EVF)

disparity map $M_E(x,y)$ given in (3.12) for a distorted rectangle. Notice that the medial points with high degree of substances shown in Fig. 5.24c correspond well with the skeletal points with large EVF disparity values. Unstable ligatures of [11] maps well into skeletal points having high degree of connection and they correspondingly exhibit low disparity values. In order to compute the disparity continuity between two adjoining skeletal segments, their respective local mean EVF disparity at branch point $\mathbf{b}$ must first be computed. A local mean value is used instead of just a single value for two reasons. Firstly, the EVF disparity values at the end of all skeletal segments meeting at branch point $\mathbf{b}$ are identical, that is they all share the same disparity value $M_E(\mathbf{b})$. This renders the disparity continuity measure meaningless since all segment pairs measure a value of 0. Secondly, the EVF disparity characteristics of a skeletal segment as it moves away from the branch point $\mathbf{b}$ is more indicative of its substance or connection measure than a single disparity value at the branch point itself. Fig. 5.27 shows a connection skeletal segment displaying a rapid drop in EVF disparity value moving away from branch point $\mathbf{b}$.

Figure 5.27 – Measuring the local mean EVF disparity value of the skeletal segment at the branch point gives information similar to the substance and connection measures proposed in [107].

Given the ordered set of points in the primary skeletal segment SP_i defined in (5.14), its local mean EVF disparity at branch point $\mathbf{b}$ is given by

$$ED_i(\mathbf{b}) = \frac{1}{N_{ED}} \sum_{n=2}^{N_{ED}+1} M_E(\{\mathbf{p}_{SP_i}(n)\}) \quad (5.16)$$

where $M_E()$ is the enhanced vector field disparity map given in (3.12) and N_{ED} is the number of skeletal points in SP_i about which the local mean value is computed. In this work, a value of $N_{ED} = 10$ was used throughout. Notice that the branch point $\mathbf{b}$ itself (i.e. $n = 1$) is excluded from the mean computation since the disparity value at $\mathbf{b}$ given by $M_E(\mathbf{b})$, is shared by all adjoining skeletal segments. The disparity continuity between two primary skeletal segments SP_i and SP_j connected at branch point $\mathbf{b}$ is therefore given by

$$CD_{\mathbf{b}}(SP_i, SP_j) = \frac{|ED_i(\mathbf{b}) - ED_j(\mathbf{b})|}{ED_i(\mathbf{b}) + ED_j(\mathbf{b})} \in [0,1] \quad (5.17)$$

Since both the orientation continuity and disparity continuity measures give values close to 0 when the visual continuity between the two adjoining skeletal segments is favorable, a combined measure for the visual conductance between two primary skeletal segments SP_i and SP_j connected at branch point $\mathbf{b}$ can be formulated using

$$VC_{\mathbf{b}}(SP_i, SP_j) = \gamma CO_{\mathbf{b}}(SP_i, SP_j) + (1 - \gamma) CD_{\mathbf{b}}(SP_i, SP_j) \in [0,1] \quad (5.18)$$

where the visual continuity emphasis parameter $\gamma \in [0,1]$ is used to determine whether orientation or disparity continuity should be the more dominant merging criterion.

5.4.2 Merging Process

A visual conductance measure based on the orientation and disparity continuity along two adjoining skeletal segments has been proposed in the previous section. This measure is used in deciding how adjoining primary skeletal segments could be merged. The complete merging process is described here. It is assumed that the set of all primary skeletal segments given by $\{SP_k\}_{k=1}^{N_p}$ in (5.13) and their respective end point pairs $\{\mathbf{b}_{k,e}\}_{e=1}^2$ are available. Firstly, the local mean EVF disparity $ED(\mathbf{b}_{k,e})$ at each end point of all N_p primary skeletal segments is computed. Skeleton merging analysis is carried out starting with the primary skeletal segment SP_i whose end point $\mathbf{b}_{i,d}$ has the highest local mean EVF disparity. This is to encourage more salient

skeletal segments (see section 5.6.1 on saliency) to merge first. The point $\mathbf{b}_{i,d}$ may or may not be a branch point. If the current active skeletal segment SP_i does not adjoin any other segment at point $\mathbf{b}_{i,d}$, no merging occurs. If it adjoin only one other primary skeletal segment SP_j , then these two skeletal segments are merged at point $\mathbf{b}_{i,d}$. If point $\mathbf{b}_{i,d}$ is a branch point, the visual conductance given in (5.18) between all adjoining skeletal segments and the current active segment SP_i is computed. The skeletal pair with the lowest visual conductance will be merged at branch point $\mathbf{b}_{i,d}$ on two conditions. Firstly, the adjoining skeletal segment SP_j that is to be merged to SP_i is not already merged to another adjoining segment SP_k . Secondly, the orientation deviation between the skeletal pair is not too large. The orientation deviation criterion can be tested using

$$OD_{\mathbf{b}}(SP_i, SP_j) = |\pi - |\theta_i(\mathbf{b}) - \theta_j(\mathbf{b})|| \leq \theta_{merge} \quad (5.19)$$

where θ_{merge} is a preset angular threshold and no skeleton merging occurs if the orientation deviation exceeds this threshold. If these two criteria are not met, the skeletal pair with the next lowest visual conductance is tested for possible merging until no pairs remain. Once SP_j and SP_i are merged at point $\mathbf{b}_{i,d}$, neither skeletal segments can be merged to another skeletal segment at that same point. Fig. 5.28 shows why the orientation deviation criterion is useful in preventing enforced merging of primary skeletal segments that do not produce good visual continuity, especially at branch points with more than three adjoining skeletal segments.

Figure 5.28 – The orientation deviation criterion ensures that primary skeletal segments that do not exhibit good visual continuity are not forced to merge.

After the merging analysis at point $b_{i,d}$ has been completed on the current active primary skeletal segment, the primary skeletal segment whose end point has the next highest local mean EVF disparity is now made the current active primary skeletal segment and the merging analysis is repeated at this new end point. Complete merging of all primary skeletal segments is achieved once there are no more skeletal end points left to analyse. At this stage, the resulting MGVF skeleton of a shape A is described by a set of N_a merged skeletal segments given by

$$\{SM_m^A\}_{m=1}^{N_a} \quad (5.20)$$

Fig. 5.29 shows the result of the merging process for two horse shapes. Notice the presence of the rider in **Horse 2** did not affect the consistency of the merging outcome as appropriate visual continuity measures were used to guide the merging process. It was observed that the skeletal segment (i.e. backbone) representing the main body of many limb-based biological shapes undergo significant bending. In order to keep this ‘backbone’ skeletal segment as one continuous merged segment, disparity continuity ought to play a larger role than orientation continuity in determining which skeletal segment at a branch point should merge. This is achieved by using a small value of γ when computing the visual conductance in (5.18) as shown in Fig. 5.29.

Figure 5.29 – Merging results of the primary skeletal segments for two different horse shapes. Only one merged segment (blue) representing the skeletal ‘backbone’ is highlighted for clarity. A small value of $\gamma=0.1$ was employed to emphasise disparity over orientation continuity. Other merging parameters used were $\theta_{merge}=4\pi/9$ (80°), $N_{CO}=5$, $N_{ED}=10$ and the images used were of size 129×129 pixels.

5.4.3 Secondary Skeletal Segments

The matching of skeletons between two shapes required a merged segment to be broken down into various combinations of primary skeletal segments in order for the best match to be identified, as shown by the matching of fish shapes in Fig. 5.30.

Figure 5.30 – Different combination of primary skeletal segments in a merged segment is needed to obtain the best skeletal match between different shapes.

These different combinations of primary skeletal segments within a merged segment are called *secondary skeletal segments*. A merged segment SM_m that is composed by linking N_m consecutive primary skeletal segments is defined by the ordered set

$$SM_m = \{SP_{m,k}\}_{k=1}^{N_m} \quad (5.21)$$

It yields an exhaustive set of unique secondary skeletal segments $\{SS_{m,k}\}$ given by

$$\{SS_{m,k}(i, j)\}_{k=1}^{N_s} = \bigcup_{n=0}^{N_m-1} \bigcup_{i=1}^{N_m-n} \bigcup_{j=n+1}^{N_m} \langle SP_{m,i} \Rightarrow SP_{m,j} \rangle \quad \text{and} \quad N_s = \sum_{n=1}^{N_m} n \quad (5.22)$$

where $\langle SP_{m,i} \Rightarrow SP_{m,j} \rangle$ means an ordered concatenation of $(j-i+1)$ primary skeletal segments within the set SM_m given in (5.21), starting from segment $SP_{m,i}$ to $SP_{m,j}$ and $i \leq j$. Fig. 5.31a shows the set of secondary skeletal segments extracted from an example merged segment. Therefore, a shape A with Na merged segments as in (5.20) can be described by an over-complete set of secondary skeletal segments given by

$$\{SS_k^A\}_{k=1}^{NA} \quad \text{where} \quad NA = \sum_{m=1}^{Na} \sum_{n=1}^{N_m} n \quad (5.23)$$

Figure 5.31 – (a) The set of all possible secondary skeletal segments extracted from an ordered set of three merged primary skeletal segments and (b) all the possible combinations of such skeletal segments to reconstitute the merged skeletal segment.

The merged segment SM_m given in (5.21) can be reconstituted in different ways by combining a subset of the secondary skeletal segments given in (5.22) in various permutations (see Fig. 5.31b). These 2^{N_m-1} different permutations can be computed recursively by using the operator $grow[ ]$ defined by

$$grow[SS(i, j)] = \{SS(j+1, j+n)\}_{n=1}^{N_m-j} \quad \text{if } j < N_m \quad (5.24)$$

and it generates a set of $(N_m - j)$ secondary skeletal segments. If the $grow[ ]$ operator is then applied recursively to each of its output segments, a tree similar to that shown in Fig. 5.32 will result. A total of 2^{N_m-1} leaf nodes will be produced and backtracking from each leaf node towards the root of the tree will yield the combination of secondary segments required to reconstitute the merged segment SM_m . The generation of these different permutations is needed for computation of the best match between merged skeletal segments of two shapes and is dealt with in section 6.3.

Figure 5.32 – The tree structure resulting from the recursive application of the $grow[ ]$ operator, starting with a pseudo skeletal segment $SS(0,0)$.

5.5 Part-based Descriptors

Now that the secondary skeletal segments have been extracted, they each have to be associated with some form of descriptor that will allow the dissimilarity measure between two such segments to be computed when shape matching is performed. The descriptors used in this work consist of a combination of various features, such as orientation histograms, skeletal curvature and limb width profile. This section describes how each of these features is extracted from a skeletal segment.

5.5.1 Source and Sink Nodes

The orientation of each skeletal segment with reference to other segments must first be determined before any descriptor can be extracted for each part. In other words, the direction of traversal along the 1D skeletal segment must be determined in a consistent manner across different shapes and these shapes may be reflected, rotated and/or scaled in any arbitrary fashion. Some strategy is required to determine which end point of the skeletal segment is the *source node*; from where the traversal of the skeleton begins and which end point is the *sink node*; at which traversal terminates.

Given a skeletal segment S_j , a simple method of assigning the source node is based on the radii of the maximal disks at the segment's two end points. The end with the larger disk radius is considered the source node and the other end the sink node. This strategy was adopted in [71], [73], [74]. However, it was observed that source node assignment based on maximal disk radial measures did not produce consistent references in shapes such as those shown in Fig 5.33a. This is especially so when vector field disparity maxima/minima segmentation is adopted and skeletal segment end points can be at other locations besides skeletal junctions.

A more reliable strategy is to assign the source node to the end point that has the shortest Euclidean distance to the shape's centroid. The result of this centroid proximity-based assignment strategy is more consistent (see Fig 5.33b).

Figure 5.33 – Source node assignment that is (a) based on the skeletal end with the largest maximal disk radius can be inconsistent since it is dependent on the relative width of limbs, joints and the shape of limb ends. (b) An assignment that is based on centroid proximity is comparatively immune to such part variations.

However, since parts and limbs can bend and articulate, an additional preference is employed to enhance the consistency of the source node assignment. A source node should also be a segmentation point (i.e. the node is in contact with another skeletal segment). The centroid proximity criterion only kicks in if both end points are segmentation points or neither point are. In such cases, the end point closest to the centroid is the source node. Fig. 5.31b illustrates why the combination of this dual constraint is useful in producing more consistent source node assignments compared to using only centroid proximity in Fig 5.31a.

Figure 5.34 – (a) Source node assignment based only on centroid proximity and (b) one that combines preference for a segmentation point (e.g. branch point) and centroid proximity.

5.5.2 Part-based Orientation Histograms

The orientation histogram was introduced in section 4.2.1, where it was used as a global shape descriptor. It is also employed here to describe various aspects of a skeletal-based part. A part can be seen as being characterised by three types of features, namely segment, connection and terminus, as shown in Fig. 5.32a. Each of these features is represented by an orientation histogram. The *segment histogram* describes the general shape of the part. The *connection histogram* describes how the part is linked to its adjoining skeletal segments and the *terminus histogram* describes the characteristics of the limb end, such as whether it is rounded or tapered. Each part has a segment but depending on the overall shape, it can have different combinations of connection and terminus, as shown in Fig. 5.32b.

Figure 5.35 – (a) Segment, connection, terminus and other relevant skeleton-related parameters such as local skeletal orientation θ_j and the maximal disk C_j about the skeletal point p_j . (b) Other possible combinations of connection and terminus.

5.5.2.1 Segment Histogram

The segment histogram is meant to describe the general shape of the part spanned by the skeletal segment. However, this segment may undergo bending such as that shown in Fig 5.35a. In order to capture the curvature-invariant shape of the part, the segment histogram is formed by cumulating rotation-normalised orientation histograms along the skeletal segment. Consider the skeletal segment S_j shown in Fig 5.35a that is

defined by an ordered set of discrete 2D points given by $\{\mathbf{p}_j(i)\}_{i=1}^{N_j}$, where the point $\mathbf{p}_j(1)$ is the source node and the $\mathbf{p}_j(N_j)$ the sink node. In order to extract the orientation histogram at skeletal point $\mathbf{p}_j(i)$, the set C_j^i consisting of all discrete image pixels within the circle circumscribed by the maximal disk radius $r_j(i)$ and centred about $\mathbf{p}_j(i)$, is first determined. The orientation histogram H_j^i consisting of n bins representing the value range $[0, 2\pi)$ is formed by cumulating the quantised orientation associated with the gradient vector of each pixel within C_j^i . The effects of skeletal bending and shape rotation are removed from the orientation histogram by adding the value $\theta_j(i)$ to all orientation values before cumulation. This angle $\theta_j(i)$ is obtained by fitting the set of points given by $\{S_j \cap C_j^i\}$ to a straight line (in the least-squares error sense) and then computing the orientation of this line with respect to the source to sink node traversal direction. An example of this is shown in Fig 5.35a for skeletal point $\mathbf{p}_j(1)$. Since broader sections of a limb are visually more salient than narrower sections, the orientation histogram is weighted by the radius of the maximal disk at point $\mathbf{p}_j(i)$. The resulting limb width-weighted n -bin orientation histogram at $\mathbf{p}_j(i)$ is given by

$$h_j^i(k) = r_j(i) \frac{H_j^i(k)}{\sum_{b=1}^n H_j^k(b)} \quad (5.25)$$

The normalised segment histogram Hsg_j for skeletal segment S_j with N_j points is then obtained by summing all the limb width-weighted orientation histograms computed along each point on the segment and is given by

$$Hsg_j(k) = \frac{\sum_{i=1}^{N_j} h_j^i(k)}{\sum_{i=1}^{N_j} \sum_{b=1}^n h_j^i(b)} \quad \text{where} \quad \sum_{k=1}^n Hsg_j(k) = 1 \quad (5.26)$$

5.5.2.2 Connection and Terminus Histograms

It may seem that the connection and terminus histograms can be obtained by computing the local orientation histograms at the respective end points of the skeletal segment. However, this simplistic approach does not work well in practice. For example, terminus points such as those at the end of a tapering limb part (see Fig. 5.33a) are usually associated with very small maximal disks. The numbers of image

pixels within these small circles are too few to produce a meaningful orientation histogram. Connection points, on the other hand, can sometimes suffer from variability in the skeletal connection angle when structural skeletons are connected as shown in Fig. 5.33c. This variability arises because the directional-uphill climbing algorithm (see section 3.3.4) is applied in regions where the vector field disparity map given by $M(x,y)$ in (3.9) has small values and therefore exhibits a rather flat saddle ridge surface similar to that shown in Fig. 3.13.

Figure 5.36 – Unreliable orientation histograms at terminus and connection points. (a) Few image samples at terminus p_t due to small maximal disk at tapered limb ends. (b) The vector field disparity maps of **Heart** and **Rabbit A**. (c) Skeletal connection angle variations at connection point due to low disparity values.

In both cases, more reliable and consistent terminus and connection histograms can be produced by summing orientation histograms over a larger number of consecutive skeletal points around each of these particular end points. But how many points should this summation process involve? Consider the two terminuses p_t and p_r of the heart shape in Fig. 5.36a. More points about the tapered end p_t should be used compared to the rounded end p_r . This is because the individual orientation histograms associated with points about p_t are less reliable due to their smaller maximal disk radii. By summing more points and thus moving further away from p_t , the respective maximal disk radii increase and so does the quality of their associated histograms.

Since the extent of histogram summation is dependent on the shape of the limb end, it is proposed that the vector field disparity profile along the skeletal segment S_j given by DS_j in (5.2) be employed. The number of skeletal points used to compute the

terminus and connection histograms can be determined by summing the disparity values $DS_j(i)$ along the skeletal segment till a predetermined fraction ρ of the total disparity along S_j has been accumulated. Notice in Fig. 5.36b, the disparity values at the tapered end are much lower than those at the rounded end. This means, for a fixed fraction ρ , more points will be used to compute the terminus histogram at the tapered end than at the rounded end and this is exactly what is desired. So in short, the n -bin orientation histogram obtained from the source node end $p_j(1)$ is given by

$$Hsr_j(k) = \frac{\sum_{i=1}^u h_j^i(k)}{\sum_{i=1}^u \sum_{b=1}^n h_j^i(b)} \quad (5.27)$$

where h_j^i is given in (5.25) and the number of skeletal points involved in computing the source node histogram can be obtained by finding the largest index value of u that still satisfies the condition given by

$$u = \arg \max_{1 \leq u \leq N_j} \left[\sum_{i=1}^u DS_j(i) \leq \rho \sum_{i=1}^{N_j} DS_j(i) \right] \quad (5.28)$$

The n -bin orientation histogram obtained from the sink node end $p_j(N_j)$ is given by

$$Hsn_j(k) = \frac{\sum_{i=v}^{N_j} h_j^i(k)}{\sum_{i=v}^{N_j} \sum_{b=1}^n h_j^i(b)} \quad (5.29)$$

and the index v can be obtained by finding the smallest value of v that still satisfies the condition given by

$$v = \arg \min_{1 \leq v \leq N_j} \left[\sum_{i=v}^{N_j} DS_j(i) \leq \rho \sum_{i=1}^{N_j} DS_j(i) \right] \quad (5.30)$$

In this work, the disparity fraction parameter $\rho \in [0,1]$ for computing a connection histogram was set at $\rho=0.1$. This was found to be sufficient to ensure that the histogram is computed over more skeletal point at connection nodes with low disparity values. However, from experimental investigations, a larger value of $\rho=0.3$ was found to be more appropriate for terminus histograms whose skeletal terminus touches the shape's boundary (e.g. the long and short taper shapes in Fig. 5.34). This is to further increase the reliability of the computed orientation histogram using more maximal disk regions along the skeletal segment whose limb width is usually very

narrow. Orientation histograms at all source and sink nodes are computed with appropriate values of ρ depending on the condition just described.

Another interesting outcome of the proposed strategy employed for computing terminus histograms is related to the human visual perception of shapes. Consider the objects in Fig. 5.34 that have different limb ends, one pair has a rounded terminus and the other a tapered terminus. As the object elongates, the terminus characteristics begin to play a less important role in differentiating the two shapes. Instead other shape features like limb boundary curvature and limb width variation become more critical. In fact, extreme elongation would render their terminus characteristics irrelevant, as the two snake-like shapes would only be compared based on how they are bent and distorted. By employing a fixed disparity fraction ρ in (5.28) and (5.30), a longer skeletal segment means the terminus histogram is constructed using more points and thus captures more segment features. This invariably dilutes the effects of limb end characteristics. This effect can be observed in the terminus dissimilarity measures shown in Fig. 5.34.

Figure 5.37 – Visual significance of terminus characteristics diminishes as the shapes elongate. The χ^2 -dissimilarity measure of the terminus histograms for (a) the two long shapes is less than that for (b) the two shorter shapes. This is in keeping with the perception of reduced terminus saliency as the shape elongates. The number of skeletal points used to compute the terminus histogram is indicated along with the total length of the skeletal segment. They were obtained with disparity fraction values of $\rho=0.1$ or $\rho=0.3$, depending on whether the terminus touches the boundary. Histograms at tapered ends are computed over a larger percentage of segment length.

5.5.3 Skeletal Curvature

Another useful description of the part is its skeletal bending characteristics. This curvature information is readily available in the local skeletal orientation given by the angle values $\theta_j(i)$, computed as described in section 5.5.2.1. The orientation function is then the ordered set of discrete angular values along segment S_j given by

$$\{\Theta_j(i)\}_{i=1}^{N_j} = \text{unwrap}[\{\theta_j(i)\}_{i=1}^{N_j}] \quad (5.31)$$

where the $\text{unwrap}[\ ]$ operator is used to extract a continuous function by maintaining continuity across the $0 \leftrightarrow 2\pi$ cyclical angular transitions, as illustrated in Fig. 5.21.

In order to compare the skeletal curvature of two segments in a scale and rotation-invariant manner, the orientation function Θ_j along segment S_j must be normalised to unit length and its first-order derivative along S_j computed. Another issue that has to be addressed is related to the compactness of the shape descriptor. Since each object has many secondary skeletal segments, curvature information of each segment must be described with a minimal number of real values. This issue is especially pertinent in shape database-related applications where a compact descriptor means lower data storage requirements. In this work, a set of polynomial coefficients is used to describe the normalised skeletal curvature. The set of $(q+1)$ polynomial coefficients representing the normalised orientation function along S_j is given by

$$\{Q_j^k\}_{k=1}^{q+1} = \text{polyfit}\left[\left\{\frac{i}{N_j}\right\}_{i=1}^{N_j}, \{\Theta_j(i)\}_{i=1}^{N_j}\right]^q \quad (5.32)$$

where the operator $\text{polyfit}[\mathbf{x}, \mathbf{y}]^n$ computes the coefficients of a polynomial function $f(\mathbf{x})$ of degree n that fits the data $f(x(i)) \cong y(i)$, in the least-squares sense [29]. The $\mathbf{x}$ values are normalised to unit length by a division of N_j , the number of points on the skeletal segment. The set of q polynomial coefficients $\{dQ_j\}$, representing the scale and rotation-invariant skeletal curvature is obtained by taking the first-order derivative of the orientation function along S_j and it is given by

$$\{dQ_j^k\}_{k=1}^q = \{(i-1)Q_j^i\}_{i=2}^q \quad (5.33)$$

5.5.4 Limb Width Profile

Another feature that has been incorporated into the part-based descriptor is the function describing the limb width of the part as one traverses the skeletal segment S_j from source node to sink node. Like the skeletal curvature information, the limb width information is readily available in the maximal disk radii given by $r_j(i)$, computed in section 5.5.2.1. The set of r polynomial coefficients $\{R_j\}$, representing the normalised limb width profile along S_j is given by

$$\{R_j^k\}_{k=1}^r = \text{polyfit}\left[\left\{\frac{i}{N_j}\right\}_{i=1}^{N_j}, \left\{\frac{r_j(i)}{N_{norm}}\right\}_{i=1}^{N_j}\right]^{r-1} \quad (5.34)$$

where the operator $\text{polyfit}[\mathbf{x}, \mathbf{y}]^n$ is similar to that used in (5.32). The segment length N_j normalises the $\mathbf{x}$ values to unit length and the value N_{norm} normalises the $\mathbf{y}$ values, i.e. the maximal disk radius values, so that the limb width profile will be scale-invariant. The value N_{norm} can be computed in two different ways. If the part descriptor is to be kept strictly local, then N_{norm} is given by

$$N_{norm} = \sum_{i=1}^{N_j} r_j(i) \quad (5.35)$$

In this case, only information within the current skeletal part is used in the normalisation process. However, as discussed in section 5.2, there are some advantages in incorporating global information into a part-based shape descriptor. Weak global shape information can be incorporated into the limb width profile via the normalisation process. In this case, the value of N_{norm} is given by

$$N_{norm} = \sqrt{N_{\mathfrak{R}}} \quad (5.36)$$

where the total area of the shape is given by $N_{\mathfrak{R}}$, the total number of pixels within the binary shape region $\mathfrak{R}$. The square-root operation makes N_{norm} the same dimension as $r_j(i)$. In this work, the global normalisation factor in (5.36) is adopted.

5.6 Visual Saliency of Parts

To quote George Orwell's *Animal Farm* [157], “*All animals are equal but some animals are more equal than others.*” In much the same way, not all parts are equal, some get more attention than others. This implies that an extracted part should be associated with some visual saliency measure that can be used to weigh its significance during shape comparison. In computing shape dissimilarity, matching errors in more salient parts should be given more consideration than less salient ones. There are several characteristics in the skeletal-based part that contributes to its visual saliency. In this work, two measures are adopted, namely the skeletal segment's enhanced vector field (EVF) disparity profile and its length.

5.6.1 Disparity-based Saliency

Since parts are formed by skeletal segments, the saliency of a part must in some measure be related to the *formation stability* of a skeletal segment. Consider that shape transformation of the Greeble object in Fig. 5.35a and its extracted MGVF skeleton. Notice that the formation stability of segment **B** is much weaker compared to that of segment **A**. In other words, the formation of segment **B** vanishes with a slight change to the boundary contour. Such an unstable skeletal segment should have a low saliency measure.

Figure 5.38 – Disparity-based saliency. (a) Formation stability of two different skeletal segments **A** and **B**. (b) Good formation stability is related to large EVF disparity values along the skeletal segment.

How could such formation stability be measured? The EVF disparity values along the skeletal segment could provide just such a measure as can be observed in Fig. 5.38b. Notice that the segment with the weaker formation stability has correspondingly lower disparity values. Consider the EVF disparity profile $DE_j(i)$ along skeletal segment S_j that is given by

$$DE_j(i) = M_E(s_j) = M_E(\{p_j(i)\}_{i=1}^{n_j}) \quad (5.37)$$

where $M_E()$ is the EVF disparity map given in (3.12). The segment s_j consists of an ordered set of n_j 2D points and is obtained by removing from skeletal segment S_j , all skeletal points that were appended using the directional-uphill climbing algorithm given in (3.15). The normalised disparity-based saliency is then given by

$$GD(S_j) = \frac{\sum_{i=1}^{n_j} DE_j(i)}{\sum_{j=1}^J \sum_{i=1}^{n_j} DE_j(i)} \in [0,1] \quad (5.38)$$

where DE_j is defined in (5.37) and J is the total number of continuous skeletal segments S_j given in (5.1). The denominator represents the total EVF disparity of all skeletal points prior to the application of connectivity analysis.

5.6.2 Length-based Saliency

Another measure of a part's saliency is given by the normalised length of its associated skeletal segment. As suggested by Siddiqi and Kimia in [198], a more extended limb is usually considered visually more salient than a shorter limb. The length-based saliency measure is given by

$$GL(S_j) = \frac{n_j}{\sum_{i=1}^J n_j} \in [0,1] \quad (5.39)$$

where J is the total number of continuous skeletal segments S_j given in (5.1) and n_j is the number of points in each segment S_j minus all those skeletal points that were added during connectivity analysis.

5.6.3 Part Saliency

The combined saliency for the part represented by skeletal segment S_j is given by

$$G(S_j) = \frac{1}{2} [GD(S_j) + GL(S_j)] \quad \in [0,1] \quad (5.40)$$

There are several reasons why visual saliency is computed from the set of raw skeletal points extracted from the MGVF instead of the larger set obtained after connectivity analysis. Firstly, as shown in Fig 5.36, when the skeletal segment s_j of a boundary protrusion is artificially connected to the structural backbone segment S_b , its overall disparity value can increase sharply since large disparity values along segment S_b (see EVF disparity map in Fig. 5.36b) is now part of S_j 's end point. Its inclusion in the saliency computation gives the boundary protrusion more visual prominence than it deserves. Secondly, the length of a skeletal segment associated with such boundary protrusions can increase significantly after connectivity analysis, as seen in Fig. 5.36a. This again serves to increase its overall saliency. Based on these observations, the appropriate saliency of a part is better reflected by its pre-connected skeletal segments and that should be used in the saliency computation given in (5.38) and (5.39).

Figure 5.39 – (a) Disparity-based saliency of the skeletal segments before and after connectivity analysis (connectivity factor $\lambda = 1.0$) given by s_j and S_j respectively. Their relative values can vary by as much as 5 times in some cases. Segment length n_j of s_j increased by > 1.5 times after connectivity. (b) Notice the large disparity values in the segment S_b in the EVF disparity map.

5.6.4 Skeletal Curvature Saliency

One aspect of the part-based descriptor involves the curvature of the skeletal-based part (see section 5.5.3). However, it was observed that the mismatch in skeletal curvature does not automatically imply that the two shapes are dissimilar. In other words, the saliency of a part's skeletal curvature is not a constant. Fig 5.37a shows two shapes that most observers will have no problems categorising as string-rays, despite the curvature variation in their tails. However, the curvature variation in the objects in Fig. 5.37b has a significant influence on their perceived shape class.

The variations in skeletal curvature saliency are essentially due to the width of the limbs. This means, when the limb width is narrow, the skeletal curvature error between two parts should be considered with a lower weightage. The saliency weighting that is associated with the skeletal curvature should vary along the skeletal segment since the limb width along the segment varies. Such a saliency measure can be obtained from the limb width profile given in (5.34). A global normalisation factor given in (5.36) should be adopted so as to ensure that the limb width saliency applied to the skeletal curvature at any given skeletal point, is proportional to the relative width of the entire shape and not relative to the average limb width of the part only. The implementation skeletal curvature saliency is detailed in section 6.1.2.

Figure 5.40 – The contribution of the skeletal curvature to shape dissimilarity is dependent of the limb width profile along the skeleton. (a) Two similar shapes with dissimilar skeletal curvature. (b) Two dissimilar shapes with dissimilar skeletal curvature.

Chapter 6

Matching Parts and Shapes

It is widely believed that as far as human perception is concerned, the triangular inequality does not hold in shape similarity comparisons (see section 4.1.2). This is especially so in situations where parts of the shape are occluded or the object is a composite of two or more simpler shapes. As a consequence, the distance between two shapes should not be evaluated based on just a metric distance. As suggested by Berretti *et al.* [24], metric distances are sometimes necessary for efficient organisation of database items through multi-dimensional index structures like M-trees or R-trees [46], [87]. In the context of this work, metric distances are required to rank the similarity between parts in a shape. Similar to the approach of Berretti *et al.* [24], the shape matching strategy proposed here makes use of two different measures called part distance and shape distance. The *part distance* is a semi-metric distance used to measure similarity between two parts. On the other hand, the *shape distance* is a non-metric distance defined as the optimal combination of selected part distances between two shapes that will yield a single dissimilarity measure. This global measure of shape similarity is more consistent with human visual perception and allows for the matching of shapes that are partially occluded.

Another variation in the approach used in this work is that it does not use matching techniques based on a graph representation. Most medial-based shape representations in the literature employ graph or tree-like shape representation such as the *skeletal graphs* of Zhu and Yuille [234], *SA-trees* of Liu and Geiger [136], [67], *rooted shock tree* of Siddiqi *et al.* [201], [163], *shock graphs* of Kimia *et al.* [112] and Sebastian *et al.* [191]. Their respective shape matching algorithms make use of graph-based matching techniques. For example, Liu and Geiger [136] employed an A*-like algorithm to locate the optimal path in their solution tree that consisted of nodes representing edge-to-edge or edge-to-path local matching of two SA-trees. Pelillo *et al.* matched rooted shock trees by finding maximal cliques of an association graph

[163]. Sharvit *et al.* [195] found optimal pairwise assignment of shock graph nodes using the graduated assignment approach [76], while Sebastian *et al.* [191] employed an edit-distance algorithm for shock graphs that could find the optimal deformation path in polynomial time. An advantage of a graph-based shape representation is the embedding of inter-part topological relationships within the link structure of the graph. Exploiting these topological relationships can often be costly as many different graphs can result depending of which part is assigned the graph's root node. The large number of possible permutations leads to high computational cost when finding the optimal match [187]. The notorious instability of medial structures (see Fig. 5.24) only adds to this problem. As a result, edit operations such as splice, merge and contract have to be incorporated to ensure that the distance between two graphs can be computed whilst taking into account the perceptual metrics of similarity [191], [67]. Since the MGVF skeleton is disconnected and therefore cannot be represented as a fully connected graph, an alternative non-graph based approach is proposed. Despite this fact, the local and global topological relationships between parts are still incorporated through the use of connection histograms and global distances.

Figure 6.1 – An overview of the parts and shapes matching process.

This chapter describes the algorithms for computing the dissimilarity measure (i.e. shape distance) between two shapes based on their respective part-based descriptors. An overview of the entire parts and shape matching process is summarized in Fig. 6.1. Two broad categories of distances are used to compute the dissimilarity measure between two shapes, namely the part distances and the global distances. Each of which will be described in their respective sections as indicated in Fig. 6.1.

6.1 Part Distance

The part distance between two skeletal parts S_i and S_j is given by the weighted combination of several error measures that are computed by comparing each of the features in the part descriptor of S_i with corresponding features in S_j .

6.1.1 Histogram Error

The histogram error between two skeletal segments S_i and S_j is the weighted matching cost of their segment, source and sink orientation histograms and is given by

$$d_H(S_i, S_j) = \frac{\eta_1 d_\chi(Hsg_i, Hsg_j) + \eta_2 d_\chi(Hsr_i, Hsr_j) + \eta_3 d_\chi(Hsn_i, Hsn_j)}{\eta_1 + \eta_2 + \eta_3} \quad (6.1)$$

where $d_\chi()$ is the χ^2 -dissimilarity measure given in (4.11) and the histograms Hsg_i , Hsr_i and Hsn_i associated with the skeletal segment S_i are given in (5.26), (5.27) and (5.29) respectively. The weight η_x regulates the contribution of each histogram type to the overall histogram error and the division by the sum of these weights ensures the histogram error $d_H(S_i, S_j) \in [0, 1]$.

6.1.2 Curvature Error

This error is associated with the difference in skeletal curvature between the two segments. Since the skeletal curvature profile along the segment has been normalised to unit length, their dissimilarity can be obtained by computing the difference between these two profiles and then accumulating the area under the resulting curvature difference profile, as shown in Fig. 6.2. However, in order to apply curvature

saliency weighting as described in section 5.6.4, the area under this difference profile is accumulated using the weighting given by the limb width profile along the segment, which incidentally, is also normalised to unit length. Since both the skeletal curvature and limb width profiles are described by the set of q and r polynomial coefficients given by $\{dQ\}$ in (5.33) and $\{R\}$ in (5.34), the operation described above can be applied directly and efficiently to these coefficients.

Figure 6.2 – Summary of the steps required to compute the saliency-weighted curvature error between two skeletal segments.

Firstly, the polynomial coefficients describing the curvature difference polynomial for the skeletal segments S_i and S_j are computed using

$$\{DP_{i,j}^k\}_{k=1}^q = \{dQ_i^k - dQ_j^k\}_{k=1}^q \quad (6.2)$$

Secondly, saliency weighting is applied to the resulting curvature difference polynomial by multiplying the two polynomials representing the curvature difference and the limb width profile. The resulting curve is a $(q+r-2)$ -ordered polynomial whose $(q+r-1)$ coefficients can be obtained using convolution [29] and is given by

$$\{WP_{i,j}^k\}_{k=1}^{q+r-1} = \{DP_{i,j}^k\}_{k=1}^q * \{R_i^k\}_{k=1}^r \quad (6.3)$$

where $*$ is the convolution operator and the limb width profile used when comparing $S_i \Rightarrow S_j$ is that set of polynomial coefficients given by $\{R_i\}$. Finally, the weighted skeletal curvature error, given by the area under the weighted difference profile is obtained by integrating this polynomial function over the unit length profile. The integration of a polynomial function between $[0,1]$ can be obtained simply by summing each of its coefficients with appropriate power-weighting as shown

$$d_C(S_i, S_j) = \left| \sum_{k=1}^{q+r-1} \left(\frac{1}{k} \{WP_{i,j}^k\} \right) \right| \quad (6.4)$$

where $||$ gives the absolute value of the summation. The use of only $\{R_i\}$ and not an equal combination of $\{R_i\}$ and $\{R_j\}$ means that the curvature error given by $d_C(S_i, S_j) \neq d_C(S_j, S_i)$. In this sense, the part distance measure is not a strict metric distance but a semi-metric distance.

6.1.3 Limb Width Error

The limb width error is obtained by first computing the polynomial function of the limb width differences between the two skeletal segments S_i and S_j given by

$$\{DR_{i,j}^k\}_{k=1}^r = \{R_i^k - R_j^k\}_{k=1}^r \quad (6.5)$$

where $\{R_i\}$ is the set of r polynomial coefficients representing the limb width variation along the skeletal segment S_i and is given in (5.34). The limb width error, given by the area under this difference polynomial is then given by

$$d_R(S_i, S_j) = \left| \sum_{k=1}^r \left(\frac{1}{k} \{DR_{i,j}^k\} \right) \right| \quad (6.6)$$

The part distance between two skeletal segments S_i and S_j is the weighted combination of these three errors and is given by

$$d_p(S_i, S_j) = \beta_h d_H(S_i, S_j) + \beta_c d_C(S_i, S_j) + \beta_r d_R(S_i, S_j) \quad (6.7)$$

The weights β_h , β_c and β_r is used to adjust the contribution of each different features to the overall part distance. Only the histogram error has been normalised to the range of $[0,1]$, the curvature and limb width errors are not. In this sense, the relative values of β_h , β_c and β_r do not necessarily indicate their proportional contribution. Suitable values of part descriptors weights were obtained empirically and in most of the experiments presented, the values of $\beta_h = 1.5$, $\beta_c = 0.7$ and $\beta_r = 1.2$ were used.

6.2 Global Distance

The proposed part distance described in section 6.1 does not incorporate much global constraint in the relationships between parts, except for the use of the global normalisation factor N_{norm} (see section 5.5.4). Features such as the connection histogram (see section 5.5.2.2) only enforces local topological constraints regarding how parts are connected at a skeletal joint. However, inter-part relationships of a more global nature are required in shape classification algorithms if large variations are to be permitted within a shape class, such as the ‘animal farm’ shown in Fig 6.3.

Figure 6.3 – The large variation in the parts of a shape does not prevent the human observer from recognising that these shapes belong to the same class of four-footed mammals. The global relationship between parts like the head, neck, legs and body are implicitly used in the categorisation process.

In computing the global constraints, a shape A is viewed as being described by an over-complete set of secondary skeletal segments $\{SS_i^A\}_{i=1}^{NA}$ as given in (5.23). Each of the NA skeletal parts is considered as having two attributes relevant to the computation of global distance. As shown in Fig. 5.35a, the first is the 2D source node coordinate of the secondary skeletal segment given by $p_i(1)$ and the second is the local orientation at the source node given by $\theta_i(1)$. The source node is made the part's reference point since it is generally more stable as it is closer to the centroid compared to the sink node, which can be at the limb terminus and therefore more sensitive to position variability due to limb articulation. An illustration of this can be found in Fig. 5.34. These dual part attributes give rise to two types of global relationship between parts, namely *global spatial distance* and *global angular distance*. Each will be described in turn.

6.2.1 Global Spatial Distances

Centroid-based reference – In order to compute the global spatial relationship between parts, a common reference point on the shape must first be identified. A convenient reference point is the shape's centroid. Given that a binary shape A is described by a centroid given by the 2D point $\mathbf{c}^A$ and a set of source nodes $\{\mathbf{p}_i^A(1)\}_{i=1}^{NA}$, the set of global spatial relationships describing the normalised distances between each source node to the centroid is given by

$$\{TC_i^A\}_{i=1}^{NA} = \left\{ \frac{\|\mathbf{c}^A - \mathbf{p}_i^A(1)\|}{N_{norm}^A} \right\}_{i=1}^{NA} \quad (6.8)$$

where $\|\cdot\|$ gives the Euclidean distance between two points and the scale-invariant normalisation factor N_{norm}^A is computed as in (5.36). The global centroid-based spatial distance between two secondary skeletal segments in shape A and B is then given by

$$g_{tc}(SS_i^A, SS_j^B) = TC_i^A - TC_j^B \quad (6.9)$$

One of the major weakness of using the shape's centroid as a reference is its sensitivity to limb deformation and occlusion [48]. This is highlighted in Fig. 6.4.

Figure 6.4 – The position of the shape's centroid can vary due to different types of local shape deformations. (a) Articulation of limbs such as the elephant's trunk. (b) An addition of an occluding object whose region is segmented along with that of the object. Notice in both cases, the spatial relationship between other non-affected skeletal segments remains relatively constant.

Source node-based reference – In order to extract more robust global spatial constraints, it is proposed that an appropriate source node (i.e. one that is not affected by the local deformation) from a selected secondary skeletal segment be used as a reference point. However, this proposal naturally leads to the question, “Which of the NA source nodes in shape A should we use?” The selection of the most appropriate source node in shape A should be a dynamic decision and dependent on shape B . The reference nodes in shapes A and B are selected such that the overall global distance difference between the two shapes being compared is minimised. This selection process is further discussed in section 6.2.3 and Appendix C.

For now, let us consider that the source node pairing that yields the minimum global distance between shape A and B is given by $\mathbf{p}_u^A(1)$ and $\mathbf{p}_v^B(1)$, respectively. The set of global spatial relationships describing the normalised distances between each source node in shapes A and B to their respective reference source node is given by

$$\{TS_i^A\}_{i=1}^{NA} = \left\{ \frac{\|\mathbf{p}_u^A(1) - \mathbf{p}_i^A(1)\|}{N_{norm}^A} \right\}_{i=1}^{NA} \quad \text{and} \quad \{TS_j^B\}_{j=1}^{NB} = \left\{ \frac{\|\mathbf{p}_v^B(1) - \mathbf{p}_j^B(1)\|}{N_{norm}^B} \right\}_{j=1}^{NB} \quad (6.10)$$

where normalisation factors N_{norm}^A and N_{norm}^B are similar to those used in (6.8). The global source node-based spatial distance between two secondary skeletal segments in shapes A and B is then given by

$$g_{ts}(SS_i^A, SS_j^B) = TS_i^A - TS_j^B \quad (6.11)$$

6.2.2 Global Angular Distances

The global spatial relationship between parts computed in the previous section is unable to disambiguate between shapes such as the two shown in Fig. 6.5a. The difference between the limb relationships is one that is angular in nature and not spatial. As such, additional angle-based constraints should be included in the computation of global distance. Two types of normalised angular measures are proposed for each source node, namely the local *source node skeletal angle* and the *source node link angle*.

Figure 6.5 – (a) Two shapes whose limbs differ only in their angular relationship but not in their spatial relationship. (b) The two angles used to compute the global angular distances are the source node skeletal angle $\theta_i(1)$ and link angle ψ_i .

Normalised source node skeletal angle – this angular measure is obtained directly from the computed local orientation $\theta_i(1)$ at the source node of each secondary skeletal segment SS_i as shown in Fig 6.5b. In order to ensure rotation invariance, all angular measures are normalised by subtracting the value of the local skeletal orientation at the reference source node. The set of normalised source node skeletal angles for shape A is therefore given by

$$\{AS_i^A\}_{i=1}^{NA} = \{posAngle[\theta_u^A(1) - \theta_i^A(1)]\}_{i=1}^{NA} \in [0, 2\pi) \quad (6.12)$$

where $\theta_u^A(1)$ is the local skeletal orientation at the reference source node $p_u^A(1)$ that is selected as discussed in 6.2.3. The $posAngle[]$ operator ensures that the computed angle is represented by a positive value between 0 to 2π . The normalised global source node skeletal angular distance between two secondary skeletal segments in shape A and B is the smaller of the two possible angular differences and is given by

$$g_{as}(SS_i^A, SS_j^B) = \min \left[\frac{|AS_i^A - AS_j^B|}{\pi}, 1 - \frac{|AS_i^A - AS_j^B|}{\pi} \right] \in [0, 1] \quad (6.13)$$

where $||$ gives the absolute value and the division by π normalises the angular distances such that it has a comparable range to that of the spatial distances computed in section 6.2.1.

Normalised source node link angle – this angular measure is computed from the orientation of the line linking a source node to the reference node (see Fig 6.5b). Given that the reference source node in shape A has been determined to be $p_u^A(1) = [x_u^A(1), y_u^A(1)]$, the source node link angle of skeletal segment SS_i is given by

$$\psi_i^A = \arctan\left(\frac{y_u^A(1) - y_i^A(1)}{x_u^A(1) - x_i^A(1)}\right) \in [0, 2\pi) \quad (6.14)$$

As before, in order to ensure rotation invariance, all angular measures are normalised by subtracting the value of the local skeletal orientation at the reference source node. The set of normalised source node link angles for a shape A is therefore given by

$$\{AL_i^A\}_{i=1}^{NA} = \{posAngle[\theta_u^A(1) - \psi_i^A]\}_{i=1}^{NA} \in [0, 2\pi) \quad (6.15)$$

The normalised global source node link angular distance between two secondary skeletal segments in shape A and B is the smaller of the two possible angular differences and is given by

$$g_{al}(SS_i^A, SS_j^B) = \min\left[\frac{|AL_i^A - AL_j^B|}{\pi}, 1 - \frac{|AL_i^A - AL_j^B|}{\pi}\right] \in [0, 1] \quad (6.16)$$

In summary, the global distance between two skeletal segments in shape A and B is derived by the weighted combination of distance measures computed using both the source-node and centroid based references and is given by

$$g_d(S_i^A, S_j^B) = g_s(S_i^A, S_j^B) + \gamma_4 g_{tc}(S_i^A, S_j^B) \quad (6.17)$$

where $g_{tc}()$ is given in (6.9) and the distance measure g_s , computed using a source node-based reference is derived using the weighted combination of both spatial and angular distances given by

$$g_s(S_i^A, S_j^B) = \gamma_1 g_{ts}(S_i^A, S_j^B) + \gamma_2 g_{as}(S_i^A, S_j^B) + \gamma_3 g_{al}(S_i^A, S_j^B) \quad (6.18)$$

The distances $g_{ts}()$, $g_{as}()$ and $g_{al}()$ are given in (6.11), (6.13) and (6.16) respectively. Suitable values of global distance weights were obtained empirically and in most experiments presented, the values of $\gamma_1 = 0.25$, $\gamma_2 = 0.25$, $\gamma_3 = 0.25$, $\gamma_4 = 0.1$, were used.

6.2.3 Determining Reference Source node Pair

In order to compute the global distance described in section 6.2, the reference source node pair given by $(\mathbf{p}_u^A, \mathbf{p}_v^B)$ must first be found. This section describes the procedure to determine this reference source node pair and the basic steps involved are outlined in Fig. 6.6. Consider the situation where the shape distance $s_d(A, B)$ is to be computed between query shape A and shape B . Firstly, the saliency of each of the Na merged segments in shape A as given in (5.20) is computed using the part saliency $G(S_j)$ given in (5.40). The most salient merged segment SM_w^A in shape A is then selected. Since the merged segment SM_w^A can be reconstituted in 2^{Nw-1} possible combinations (see Fig. 5.32), the combination of secondary skeletal segment $\{SS_{w,k}^A\}_{k=1}^{Kw}$ giving the lowest saliency-weighted matching cost is selected as its representative set of skeletal segments. This least cost combination is found using the saliency-weighted matching cost derived in a similar manner as in (6.21).

For each secondary skeletal segment $SS_{w,k}^A$ in the set $\{SS_{w,k}^A\}_{k=1}^{Kw}$, its counterpart secondary skeletal segment $SS_{k \min, k}^B$ in shape B that produces the lowest matching cost is then found with the help of the over complete cost matrix $OCM(i, j)$ given in (6.20). The result is K_w secondary skeletal pairs given by $(SS_{w,k}^A, SS_{k \min, k}^B)_{k=1}^{Kw}$. The set of skeletal pairs are then ordered based on an increasing matching cost. To find $(\mathbf{p}_u^A, \mathbf{p}_v^B)$, each skeletal pair in turn is assumed to contain the reference source nodes and the total global distance error ξ_{gs} given by

$$\xi_{gs} = \sum_{i=1}^{NA} \min_{1 \leq j \leq NB} [g_s(SS_i^A, SS_j^B)] + \sum_{j=1}^{NB} \min_{1 \leq i \leq NA} [g_s(SS_i^A, SS_j^B)] \quad (6.19)$$

is computed based on this assumption. The cost measure $g_s()$ is the source node-based global distance given in (6.18). The skeletal pair that yields the lowest ξ_{gs} contains the reference source node pair $(\mathbf{p}_u^A, \mathbf{p}_v^B)$. A full search begins with the first skeletal pair (i.e. the pair with the lowest matching cost) and proceeds all the way through the ordered list till the last K_w -th pair. After which, the reference skeletal pair with the lowest value of ξ_{gs} can be readily determined.

Figure 6.6 – Summary of the steps involved in determining the reference source node pair between shapes A and B.

6.3 Shape Distance

Consider the problem of matching shape A to shape B , where A is the query shape and B is one of the many shapes in a shape database. The shape distance $s_d(A, B)$ gives the minimal saliency-weighted matching cost that would map all skeletal parts in shape A to those in shape B . The shape distance between two shapes is not a metric distance, which means $s_d(A, B) \neq s_d(B, A)$.

Figure 6.7 – The creation of the over-complete cost matrix $OCM(i,j)$ and the cost matrix $CM(i,j)$ from the set of secondary skeletal segments of two shapes A and B .

In computing the shape distance $s_d(A, B)$, each shape A and B is viewed as being described by an over-complete set of secondary skeletal segments $\{SS_i^A\}_{i=1}^{NA}$ and $\{SS_j^B\}_{j=1}^{NB}$ respectively, as given in (5.23). Firstly, the over-complete cost matrix $OCM(i,j)$ is created (see Fig. 6.7) by computing the sum of the part distances and global distances for every combination of (SS_i^A, SS_j^B) . This results in a matrix of size $NA \times NB$ and its elements representing part-to-part matching costs are given by

$$OCM(i, j) = d_p(SS_i^A, SS_j^B) + g_d(SS_i^A, SS_j^B) \quad (6.20)$$

where $d_p()$ and $g_d()$ are the part and global distance measures given in (6.7) and (6.17) respectively. In order to match every skeletal part in shape A to a corresponding part in shape B in a unique and non-repetitive manner, the over-complete matrix $OCM(i, j)$ must be reduced to its basic form. In other words, every primary skeletal segment in both shapes A and B must be represented in no more than one entry in the basic cost matrix $CM(i, j)$. As a result, all merged segments SM_m given in (5.21) that can be reconstituted in 2^{Nm-1} possible combinations (see Fig. 5.32) must be reduced to a single most suitable combination of secondary skeletal segments as shown in Fig. 6.7.

This can be achieved in the following steps. Firstly, the minimum saliency-weighted matching cost associated with each of the NA secondary skeletal segments of shape A is obtained. This set of matching costs is derived from the over-complete cost matrix and is given by

$$\{MA_i\}_{i=1}^{NA} = \left(\min_{1 \leq j \leq NB} [OCM(i, j)] \right) \times G(SS_i^A) \quad (6.21)$$

where the part saliency $G()$ of a skeletal segment is given in (5.40). For each of the Na merged segments in the set $\{SM_m^A\}_{m=1}^{Na}$ given in (5.20), all possible 2^{Nm-1} combinations of secondary skeletal segments that can reconstitute SM_m^A are derived by using the $grow[]$ operator recursively to generate a binary tree. Backtracking from the leaf nodes is then used to obtain each combination, as described in section 5.4.3. The total matching cost for each combination is then computed by summing the saliency-weighted matching cost given in (6.21) for the respective secondary skeletal segment present in the combination. The combination with the least cost is selected as the most suitable representation of the merged segment SM_m^A . After all Na merged segments have been analysed, shape A is now described by a basic subset of the original over-complete set of NA secondary skeletal segments. Let this resulting basic subset be described by the set

$$\{SR_i^A\}_{i=1}^{nA} \subseteq \{SS_i^A\}_{i=1}^{NA} \quad \text{where } nA \leq NA \quad (6.22)$$

and the set of indices that would map this basic subset to the over-complete set of secondary skeletal segments is given by

$$\{k(i)\}_{i=1}^{nA} \quad \text{where } SR_i^A = SS_{k(i)}^A \quad (6.23)$$

In order to find the corresponding basic subset of secondary skeletal segments for the shape B , the earlier subset extracted from shape A is used to influence the selection process. The minimum saliency-weighted matching cost associated with each of the NB secondary skeletal segments of shape B is obtained over a reduced sample of the over-complete cost matrix and is given by

$$\{MB_j\}_{j=1}^{NB} = \left(\min_{1 \leq i \leq nA} [OCM(k(i), j)] \right) \times G(SS_j^B) \quad (6.24)$$

As for shape A before, each merged segment in shape B given by the set $\{SM_m^B\}_{m=1}^{Nb}$ is analysed in order to determine which combination of secondary skeletal segments would yield the lowest total cost based on the saliency-weighted cost given in (6.24). The combination with the lowest cost is selected to represent the merged segment. When all Nb merged segments have been analysed, shape B is now described by a basic subset of the original over-complete set of NB secondary skeletal segments given by the set

$$\{SR_j^B\}_{j=1}^{nB} \subseteq \{SS_j^B\}_{j=1}^{NB} \quad \text{where } nB \leq NB \quad (6.25)$$

Since both shapes A and B have been reduced to their respective basic subsets, the over-complete cost matrix $OCM(i, j)$ can be reduced to the basic cost matrix $CM(i, j)$ of size $nA \times nB$ as shown in Fig 6.7.

Since more salient parts should carry a higher matching cost penalty, the cost matrix $CM(i, j)$ should be weighted by the respective part's saliency. When computing the shape distance $s_d(A, B)$, only the part saliency for each of the secondary skeletal segments SR_i^A in query shape A is employed. The saliency-weighted cost matrix $WM(i, j)$ is therefore given by

$$WM(i, j) = CM(i, j) \times G(SR_i^A) \quad (6.26)$$

where the part saliency $G()$ of a skeletal segment is given in (5.40).

The shape distance between shapes A and B is the optimal saliency-weighted match between all parts in A and those in B such that the resulting total matching cost is the lowest. In other words, given the weighted cost matrix $WM(i,j)$ in (6.26), the objective is to minimise the total cost of matching subject to the constraint that the matching be one-to-one. If this cost matrix is square, we have a classic case of the square assignment problem (or weighted bipartite matching problem) and this can be solved in $O(N^3)$ time by the Hungarian algorithm [18], [47]. The output from the Hungarian algorithm is a list of skeletal pair matches (SR_i^A, SR_j^B) that results in the lowest overall matching cost given by $s_h(A,B)$. In reality, the differing part count in shapes A and B rarely result in a square cost matrix. This problem can be resolved by padding the matrix $WM(i,j)$ with dummy costs such that it results in a square matrix before the Hungarian algorithm is applied. The addition of dummy cost is also useful as it serves to penalize a query shape that has additional parts, especially if these additional parts are visually salient as saliency weighting can also be applied to the dummy costs to moderate its influence on the overall cost of matching two shapes (see Fig. 6.8b). However, the way the dummy costs are employed is dependent on two possible scenarios as shown in Fig. 6.8.

Figure 6.8 – The method of producing a square weighted cost matrix by adding dummy costs depends on the number of parts count in query shape A and shape B . (a) When shape A has less parts than B . (b) When shape A has more parts than B .

The first scenario is when $nA < nB$. In this case, there are less parts in query shape A than there are in shape B , which means only a subset of the parts in shape B will have matches. The weighted cost matrix $WM(i,j)$ is padded with a large dummy cost of say $DC_{A<B} = 10$ as shown in Fig 6.8a. The extra dummy costs $DC_{A<B}$, which are larger than any of the entries in $WM(i,j)$ can be subsequently removed from the

matching cost $s_h(A, B)$ computed by the Hungarian algorithm. The final shape distance in this case is given by

$$s_d(A, B) = s_h(A, B) - [(nB - nA) \times DC_{A < B}] \quad (6.27)$$

The second scenario is when $nA > nB$. In this case, there are more parts in query shape A than there are in shape B , which means some parts in shape A must be matched to dummy parts in shape B . In order to achieve this, dummy cost of say $DC_{A > B} = 1.5$ is padded to the cost matrix $CM(i, j)$ to make it a square matrix as shown in Fig 6.8b. If both shapes A and B share a similar gross shape, the extra skeletal parts in A are typically low saliency textural skeletons. By applying saliency weighting to this padded cost matrix, the eventual dummy cost of non-salient parts will be reduced drastically and not contribute significantly to the overall matching cost, despite the addition of dummy parts. From the padded cost matrix, the saliency-weighted matrix $WM'(i, j)$ shown in Fig 6.8b is computed as in (6.26). The Hungarian algorithm is then applied to the square matrix $WM'(i, j)$ and the final shape distance is simply

$$s_d(A, B) = d_h(A, B) \quad (6.28)$$

6.4 Bi-directional Shape Distance

The shape distance $s_d(A, B)$ is not a metric distance and does not obey the triangle inequality property described in (4.7). This characteristic is useful as it supports the human perception of shape similarity where partial matching of shape parts are often used in associating simpler shapes to more complex composite ones (see Fig. 6.9). This is achievable because the shape distance $s_d(A, B)$ given in (6.27) does not penalise the additional parts in shape B that are not present in A .

Figure 6.9 – The shape distance supports partial matching. The matching cost between query Greeble **A** and the composite shape #1 is lower than those with the Greeble variants (**B**, **C** and **D**), since Greeble **A** is present in both the query and shape #1. Shape distances between query and each shape is given in the bottom-left corner.

However, there are instances when partial matching will work against us. Consider the example shown in Fig 6.10a, where a bent double-headed spanner shape is queried against a set of spanner and fish variants. Notice that the matching of the double-headed spanner to the various fish shapes yielded better match scores than with **Spanner #5**. This does not reflect the similarity perception of the human observer. The additional head in the query spanner had no part match in the single-headed **Spanner #5** and was therefore duly penalised by the addition of dummy cost when computing the shape distance $s_d(A,B)$ given in (6.28). In order to overcome such problems, the *bi-directional shape distance* $s_b(A,B)$ is proposed and is given by

$$s_b(A,B) = \frac{1}{2}[s_d(A,B) + s_d(B,A)] \quad (6.29)$$

The idea behind bi-directional shape distance is to ensure that a good match is only obtained if the shape distances obtained when matching shape A to B and for matching shapes B to A are both low. In this way, partial matching of a simple shape to a composite shape may yield a low shape distance one way but would be penalised when the matching direction is reversed. The combined score better reflects the similarity between the two shapes (see Fig. 6.10b). Bi-directional shape distance $s_b(A,B)$ was employed in most of the shape classification experiments.

Figure 6.10 – (a) The shape distance does not always produce desirable results when the query shape has more parts than another shape in the same shape category. (b) Bi-directional shape distance provides a similarity measure that is better able to account for this disparity in the number of parts between shapes. The respective distances between query and each shape are given at the bottom-left corner.

6.5 Handling Mirror Reflection

Since the proposed MGVF part-based descriptor is not invariant to mirror reflection, the shape A must be matched to both shape B and B' , its mirrored version. The smaller of the two shape distances will be the effective shape distance between shape A and B . In order to convert the part descriptors of shape B to those of B' , several additional computations are required and they are listed here.

Firstly, the part-based orientation histograms described in section 5.5.2 must be converted to their mirror-reflected equivalents. As illustrated in Fig. 4.18, mirror reflection causes a reversal of bin order. For each skeletal segment S_i , its associated histograms Hsg_i , Hsr_i and Hsn_i given in (5.26), (5.27) and (5.29) respectively, are re-ordered to their corresponding mirrored equivalents using the histogram bin re-indexing operation given in (4.29). The mirrored equivalent histogram error $d_H'()$ is then obtained by substituting the re-order histograms into (6.1).

Secondly, the skeletal curvature of skeletal segment S_i described by the set of polynomial coefficients $\{dQ_i\}$ given in (5.33) is sensitive to the mirror transform. Since these coefficients are first-order derivatives of the orientation function along S_i , mirror reflection will merely change their sign and their mirrored versions are given by

$$\{dQ_i'\} = \{-dQ_i\} \quad (6.30)$$

The mirrored equivalent curvature error $d_C'()$ given in (6.4) can then be computed by substituting these modified coefficients. The part distances $d_p'()$ associated with the mirror reflected shape B' can now be obtained by substituting the mirrored equivalent histogram and curvature errors into (6.7).

Finally, the two global angular distances can be converted to their respective mirrored equivalent by subtracting from the value of 2π . The sets of normalised source node skeletal and link angles given in (6.12) and (6.15) respectively, can be transformed to their mirrored equivalent by

$$\{AS_i'\} = \{\text{posAngle}[2\pi - AS_i]\} \quad \text{and} \quad \{AL_i'\} = \{\text{posAngle}[2\pi - AL_i]\} \quad (6.31)$$

With these sets of modified angular distances, a mirror reflected global distance $g_s'()$ can be computed and along with $d_p'()$, the shape distance $s_d(A, B')$ can now be calculated.

6.6 Experimental Results and Discussion

Several experiments were conducted to study the robustness and shape description capabilities of the proposed MGVF part-based descriptor (MGVF-PBD). Some of these experiments are similar to those used to study the MVGF-based global descriptor in section 4.3. The parameter settings used in all the experiments presented are detailed in Appendix D unless explicitly indicated otherwise.

6.6.1 Geometric Invariance

The first experiment demonstrates the MGVF-PBD invariance to geometric transformations such as scaling and rotation. Twelve 257×257 pixel-sized images comprising three basic shapes (i.e. hare, rabbit and turtle) in various orientations and spanning scales of two octaves were matched each to every other. Bi-directional shape distances were computed for both the normal and mirrored versions[†]. The lower of the two was selected and the resulting matched shapes were ordered based on closest match as shown in Fig. 6.11.

	Hare				Rabbit				Turtle			
	Scale: $\times 1$	Scale: $\times \frac{1}{2}$	Scale: $\times 2$	Scale: $\times 1$	Scale: $\times 1$	Scale: $\times \frac{1}{2}$	Scale: $\times 2$	Scale: $\times 1$	Scale: $\times 1$	Scale: $\times \frac{1}{2}$	Scale: $\times 2$	Scale: $\times 1$
	Rotate: Original	Rotate: 90°	Rotate: -30°	Rotate: mirror	Rotate: Original	Rotate: 90°	Rotate: -30°	Rotate: mirror	Rotate: Original	Rotate: 90°	Rotate: -30°	Rotate: mirror
Query												
Match #1	5	19	11	5	6	23	16	6	1	15	11	1
#2	11	19	12	12	16	23	17	17	11	15	12	12
#3	19	22	22	19	23	30	23	23	15	20	20	15

Figure 6.11 – Results demonstrating the MGVF-PBD ability to handle scaling, rotation and mirror reflection[†]. A cross (X) at the bottom-left corner indicates a misclassification. The bi-directional shape distance measures shown in the top-right corner are rounded to the nearest integer and scaled by $\times 100$.

[†] Shape distance computation of mirrored version is discussed in section 6.5

All shapes were correctly matched to their respective shape type except for the two rabbit shapes, which are at their respective scale extremes. The smallest-scaled

rabbit ($\times \frac{1}{2}$) found the smallest-scaled hare a better match than its largest-scaled counterpart ($\times 2$). This is not surprising as the hare and rabbit shapes share very similar skeletal topology (see Fig. 6.12). This can easily lead to mismatches when scale differences as large as ($\times 4$) exist between the shapes. When the shape in an image gets very small, it is difficult to form statistically reliable orientation histograms due to the reduced number of discrete gradient vectors within each part. This same problem is observed in Fig. 6.13. Notice that the bi-directional shape distances obtained when matching the small hand ($\times \frac{1}{2}$ scale) to all the other sized hands in the database deteriorate much more rapidly and significantly compared to using the large hand ($\times 2$ scale).

Figure 6.12 – The extracted MGVF skeletons for the twelve shapes used in the geometric invariance experiment.

Figure 6.13 – Influence of shape size on the reliability of the MGVF part-based descriptors. The MGVF-PBD extracted from very small shapes are not as reliable as those obtained from larger-sized shape.

6.6.2 Skeletal Topology

The objective of this experiment is to assess if the MGVF-PBD is able to distinguish between structurally similar shapes. Many shapes that are perceptually different may give rise to the same skeleton topology, as shown by the series of morphed shapes in Fig. 6.14. It is therefore important that the proposed part-based shape descriptor encapsulates more than just the structural characteristics of the shape.

Figure 6.14 – Shapes used in skeletal topology experiment. (a) Ten shapes from a morphed sequence of a donkey to a rabbit. Taken from a similar experiment by Torsello and Hancock [216]. Shown below are their corresponding MGVF skeletons.

Fig. 6.15 shows the bi-directional shape distances obtained with the MGVF-PBD when the original donkey and rabbit images were used as queries. The near monotonic increase in matching cost is observed when moving across the ordered morphed sequence. These results prove that the proposed part-based descriptor and matching algorithm is able to disambiguate between shapes with topologically similar skeletons and the desirable monotonicity property discussed in section 4.1.2 is evident in the MGVF-PBD's shape dissimilarity measure.

Figure 6.15 – Result for the skeletal topology experiment using the MGVF-PBD.

6.6.3 Noisy Shape Recognition

This experiment involves the recognition of a set of twenty-five 129×129 pixel-sized binary images (see Fig. 6.16a) that were randomly rotated and whose boundaries were perturbed with increasingly more Gaussian noise with each test run. The detail of the experiment set up has been described in section 4.3.2. Even at a boundary noise level of $\sigma_n=2.0$, only two recognition errors were obtained using the MGVF-PBD, as highlighted in Fig. 6.16b. Their respective mismatches were shapes whose skeletal topology is very similar.

Figure 6.16 – The extracted MGVF skeletons for the shapes used in the noisy shape recognition experiment. (a) Original set of clean images similar to that used by Ben-Arie and Wang in [21]. (b) The set of noisy images in the $\sigma_n=2.0$ test run. Shapes recognised wrongly are marked with a (X) and their respective mismatch indicated.

The shape recognition performance of the MGVF-PBD over different boundary noise levels is generally better than the MGVF global descriptor proposed earlier in section 4.2. The results in Fig. 6.17a show that at high levels of noise, the MGVF-PBD consistently outperforms the adaptive hybrid normalisation (AHN) orientation histogram descriptor. The problem of non-distinct global orientation histograms (highlighted in Fig. 4.28) for some shapes (e.g. circle and cross) is not an issue for the part-based descriptor. This is because the MGVF-PBD approach decomposes a complex shape into simpler subparts and the shape of each of these parts is encoded

using separate orientation histograms unlike the single global descriptor. The richer variety of features allows different shapes to be described in a more distinctive manner. As a result, these descriptors are more tolerant to noise-induced variations.

However, since each individual part descriptor describes only a small local region of the shape, they by themselves, will inadvertently be more severely affected by the presence of boundary noise. This is especially so if the parts are narrow, like the wings and fuselage of an aircraft (observable in Fig 6.16b). Unfortunately, this is an inherent weakness of part-based shape descriptors. It can be readily observed in Fig. 6.17b, that many of the shape recognition errors do indeed occur for the aircraft shapes. As a result of this, the comparative results in Fig. 6.17a show that the performance of the Zernike moments descriptor (ZMD), a global descriptor, is better than the MGVF-PBD at low boundary noise levels. Surprisingly, as the boundary noise exceeds $\sigma_n = 1.5$, the noise resilience of the MGVF-PBD was still holding up as the recognition errors of the ZMD began to show [†]. Shapes with more extensive interior regions and distinctive skeletal topology continued to be robustly described by the MGVF-PBD.

Figure 6.17 – Shape recognition performance of the MGVF-PBD. (a) Performance comparison between MGVF-PBD ($\alpha = 0.9$) and the global shape descriptors MGVF-AHN ($\lambda_{LG} = 0.1$, $\alpha = 0.5$) and the ZMD ($n = 10$). (b) The number of recognition errors associated with each of the 25 shapes when using the MGVF-PBD.

[†] Being a region-based shape descriptor, the ZMD mismatches at high noise levels were mainly between the circle and hexagon shapes

6.6.4 Robustness to Boundary Texture

The shape classification performance of the MGVF-PBD in the presence of boundary texture was studied using the set of four base shapes and their respective textured variants shown in Fig. 6.18. The objective of the experiment is to retrieve all textured versions of the base shape when the base shape is used as a query image. Perfect retrieval is achieved when the closest three matches consist of all the textured shapes derived from the query base shape.

Figure 6.18 – (a) The set of 16 binary images (129×129 pixels) used to test the MGVF-PBD robustness to boundary texture. Four base shapes (hare, rabbit, pear and candle), each with three different textures (wavy, spiky and hairy) form the test shape database. (b) The corresponding extracted MGVF skeletons.

The retrieval results using the MGVF-PBD in Fig. 6.19a show that all associated textured shapes of each of the four query based shapes were correctly retrieved within the best three matches. This perfect retrieval result was attainable partly due to the ability of the shape distance measure to perform partial matching. The additional skeletal fragments resulting from the boundary texture (see Fig. 6.18b) are not considered in the computation of the shape distance $s_d(A, B)$ once all parts in the query shape A have located their corresponding part match in the comparison shape B . In this way, the textural skeletons did not significantly affect matching performance.

Additionally, the MGVF-PBD is reasonably robust against boundary noise as observed in the noisy shape recognition experiment in section 6.6.3. The disconnected nature of the MGVF skeleton, which resulted in structural and textural skeletons, did

increase reliability and computational speed of performing part-to-part matching of skeletal segments between the two shapes.

As a comparison, shape classification results using the popular Zernike moments descriptor were also obtained. Earlier results in section 4.3.3 showed this to be the next best performing global shape descriptor in the boundary texture experiment after the MGVF orientation histogram. Region-based descriptors such as the ZMD are generally more robust to boundary-based distortions. The results in Fig. 6.19b shows that the ZMD was unable to correctly retrieve all textured versions of the hare and rabbit base shape within the best three matches. Part of the weakness of the ZMD descriptor is the need for centroid centering and scaling to a unit circle (see Fig. 4.4) before the moments can be computed in a scale and rotation-invariant manner. This process is often prone to error when the shape is highly textured, especially with long spiky protrusions. It is therefore not surprising that the ZMD resulted in mismatches between the hare and rabbit shapes, which are actually very similar visual forms.

Figure 6.19 – Results of the experiment to test robustness to boundary texture. (a) The four closest matches using the MGVF-PBD for each query shape, which consisted of the base shape. The shape distance, $s_d(A, B)$ shown in the top-right corner is rounded to the nearest integer and scaled by $\times 100$. (b) The four closest matches using the Zernike moments descriptor. Centroid centering and scaling to a unit circle was applied to the shape before computing the ZMD. The first 36 moments of order up to $n = 10$ was compared using the L_2 -distance measure. Shapes that were misclassified within the first three closest matches are marked with a (X).

6.6.5 Occlusion

Any useful shape matching technique must be robust to commonly occurring visual distortions such as occlusions. Occlusions arise due to several reasons. Since 2D shapes are obtained as projection of 3D objects, their 2D silhouettes can change with variation in the viewpoint or through self-occlusion when the object is rotated. Additionally, an object could be partially occluded by another object that is in front of it. Under these circumstances, if the extent of the occlusion is not too severe, a good shape descriptor should enable such partially occluded shapes to be recognised or matched to an unoccluded example in the shape database. Part-based shape descriptors are generally more robust to occlusion than global shape descriptors as they describe shapes as a composition of several parts. If a sufficient number of parts in the shape are not occluded, a good matching score can still be obtained.

Figure 6.20 – Viewpoint variation experiment. The bi-directional shape distances increase almost monotonically as the viewing angle of the 3D object varies away from a reference norm given by the query image. Experiment follows that in [191].

The result in Fig. 6.20 demonstrates the continuity property of the MGVF-PBD to the viewpoint variation of a 3D shape. As the viewpoint is changed gradually, the respective 2D projected parts of the shape are also deformed gradually. This led to a corresponding change in the overall bi-directional shape distances that was nearly monotonic. This was expected since features in the MGVF-PBD such as the orientation histograms have been shown to exhibit the occlusion and curvature continuity properties (see Fig. 4.13). Viewpoint variation may also cause parts in the 2D projected shape to appear and disappear at certain viewing angles. Such shape modification can be handled by the partial matching capabilities of the proposed shape distance measure as discussed in section 6.4. However, the addition of parts and

the deletion of parts from a shape have a different influence on the computed shape distance. Fig. 6.21 shows the results of comparing a shape A to its partially occluded versions. In this example, the occluding object blends with the background and thus has the effect of deleting an increasing number of parts from shape n as the amount of occlusion increases. With occlusions of this nature, a higher shape distance $s_d(A, n)$ would result if the unoccluded query shape A is compared to the occluded shapes n than if the shapes were matched in the opposite direction. This is because the unoccluded shape A , which has more parts, will be penalised by the additional cost incurred in assigning pseudo matches to dummy parts in the occluded shape n (see section 6.3).

On the other hand, occlusion can occur in such a way that the occluding object blends with the shape (e.g. the shape **Hand 3** in Fig. 6.22). In this case, the occluded shapes n will now have more parts than the unoccluded query shape A . Under such circumstances, the shape distance $s_d(A, n) < s_d(n, A)$. Since in practice, the nature of the occlusion may not be known a priori, a good compromise is to adopt the bi-directional shape distance measure, which computes an average of the two shape distances $s_d(A, n)$ and $s_d(n, A)$, as shown in Fig. 6.21.

Figure 6.21 – Partial occlusion experiment. When matching partially occluded shapes, the direction in which the shape distance is computed has an influence on the computed distance measure. Bi-directional shape distance should be employed if the nature of the occlusion is not known before hand.

A small dataset of six shapes similar to that employed by Sebastian *et al.* in [191] was used to compare the robustness of the MGVF part-based descriptor and the MGVF global descriptor (AHN) to partial occlusion. Fig. 6.22a shows that the

MGVF-PBD produced perfect shape classification results, whereas the global nature of the MGVF-AHN shape descriptor proposed in chapter 4 made it more susceptible to partial occlusion (see Fig. 6.22b). Additional comparative performance using this same dataset is presented in Fig. 6.23. Sebastian’s edit distance algorithm [191] applied to shock graphs (Fig. 6.23a) was similarly able to produce perfect classification results but not the shape context of Belongie *et al.* [18].

Figure 6.22 – Shape classification performance with a partially occluded shape using (a) the MGVF-PBD and (b) the global shape descriptor MGVF-AHN ($\lambda_{LG}=0.2$, $\alpha=0.8$) described in chapter 4. Shapes that were misclassified within the first two closest matches are marked with a (X).

Figure 6.23 – Shape classification performance with a partially occluded shape using (a) shock graph edit distance [191] and (b) the shape context [18]. The former is a part-based descriptor but the latter is a global one. Results adapted from [191].

6.6.6 Limb Articulation

Limb articulation is another shape deformation that is commonly encountered in practical shape recognition applications, especially those involving objects such as human figures or articulated tools and machinery such as pliers, cranes, etc. Shape descriptors like the MGVF-PBD that are derived from the decomposition of a complex shape into basic subparts, are generally more robust to such deformation.

Figure 6.24 – 12 binary images (129×129 pixels) used to study limb articulation and their extracted MGVF skeletons obtained with the parameters listed in Appendix D.

	Pliers			Chopsticks			Fat Man			Thin Man		
Query												
Match #1												
#2												
#3												
#4												
#5												

Figure 6.25 – Shape classification performance with articulated shapes using the general MGVF-PBD parameter settings given in Appendix D. For each of the query shape, the five matched shapes in the database that gave the lowest bi-directional shape distance are shown, in increasing value from match #1 to #5. Shapes that were misclassified within the first two closest matches are marked with a (X).

Using the four sets of three articulated pliers, chopsticks, fat and thin man shapes (see Fig. 6.24), the shape classification results seen in Fig. 6.25 were obtained for the MGVF-PBD. Notice that all shape categories except the thin man had at least two misclassifications within the two best matches. These misclassifications are the result of certain shape features incorporated into the MGVF-PBD. Firstly, consider the three types of part-based orientation histograms (see section 5.5.2). The segment histogram

describes the general curvature-normalised shape of the limb part along the skeletal segment and remains relatively unchanged with limb articulation. However, the source and sink histograms, which may be connection histograms, are influenced by the intersection angle of skeletal axes at the skeletal junction and this characteristic changes as the limb articulates (see Fig. 6.24). If the situation requires the shape descriptor to be relatively immune to limb articulation and more sensitive to the general shape of the part, then the segment histogram weight η_1 given in (6.1) should be increased relative to η_2 and η_3 , the source and sink histogram weights respectively.

Secondly, limb articulation also affects the shape's global geometry. In most cases, both the spatial and angular relationships between parts are affected. Since the global distance (see section 6.2) measures such inter-part relationships, its contribution to the cost matrix given in (6.20) must be de-emphasised in order to reduce sensitivity to limb articulation. This can be done by reducing the global distance weights γ_1 , γ_2 , γ_3 and γ_4 , given in (6.17) and (6.18). Fig. 6.26 shows perfect shape classification results obtained when the parameters given in Appendix D are modified as suggested: the segment histogram weight η_1 is increased by a factor of 2.0 and the global distance weights γ_1 , γ_2 , γ_3 and γ_4 are all reduced a factor of 0.3. This increase in emphasis of part shape form and reduction in emphasis of inter-part geometric relationship do come with their associated side-effects and this issue is further discussed in the silhouette classification experiments (see Fig. 6.29).

	Pliers			Chopsticks			Fat Men			Thin Men		
Query												
Match #1												
#2												
#3												
#4												
#5												

Figure 6.26 – Perfect shape classification results for the articulated shape dataset using MGVF-PBD with increased emphasis on the segment histogram (relative to the connection and terminus histograms) and reduced emphasis on the global distance.

6.6.7 Silhouette Classification

The shape classification capability of the MGVF-PBD was tested using several silhouette databases. In order to carry out comparative performance evaluation with the various global shape descriptors featured in section 4.3.4, the same dataset of 25 images (6 categories) shown in Fig. 4.30 was used. The matching results shown in Fig. 6.27 were obtained with the MGVF-PBD. Employing the same scoring method detailed in section 4.3.4, the score is (25, 25, 24), which is very close to the perfect score of (25, 25, 25). This is by far the best score compared to those descriptors summarised in Table 4.3. The best reported score for this test set using a global descriptor was that of Belongie *et al.*'s shape context in [18] and it was (25, 24, 23). Klein *et al.* [115] reported a score of (25, 25, 22) by applying edit-distance to shock graphs. Their part-based approach assigned cost to skeletal-based splice/merge operations in the search for an optimal match, thus providing a means to account for matching discrepancies due to occlusion, limb articulation, part addition and deletion. These types of deformations are present to varying extents in the 25-image dataset and the good classification results obtained with the proposed MGVF-PBD demonstrate its robustness to such shape changing processes as well.

MGVF PBD	Planes				Bunnies				Hands				Spanners				Greeble				Fishes			
Query																								
Match #1																								
#2																								
#3																								

Figure 6.27 – Shape classification results for Sharvit *et al.*'s [195] dataset using the MGVF-PBD. Only one misclassification was observed within the three best matches and it is marked with a red circle (○)

The shape classification performance of the MGVF-PBD was also compared with two other medial-based shape matching techniques using the database of Pelillo *et al.* [163]. It consists of 25 shapes, organised into 8 categories as shown in Fig. 6.28.

Brush			Hammer			Pliers				Driver		Wrench		Hand			Head			Horse	

Figure 6.28 – The 25-shape database of Pelillo *et al.* [163].

Figure 6.29 – Comparative performance of the MGVF-PBD to other medial-based shape matching techniques on the dataset of Pelillo *et al.* [163]. (a) MGVF-PBD with parameters in Appendix D. (b) MGVF-PBD with parameters used in limb articulation experiment shown Fig. 6.26. (c) Shock graph edit distance [191]. (d) Shock tree matching using association graphs [163]. Correct within-category matches and erroneous within-category matches are highlighted in yellow (✗) and red (✗) respectively. Results for (c) and (d) were adapted from [191].

Firstly, the comparison of results between the two MGVF-PBD with different parameter settings highlights the trade-off between being robust to relative part size changes (e.g. fat and slim screwdrivers) as in Fig. 6.29a and being robust to limb articulation (e.g. pliers) as in Fig. 6.29b. By de-emphasising the global geometry descriptors in the MGVF-PBD as suggested in section 6.6.6, the classification errors in the articulated pliers were reduced but intuitively similar looking shapes like the fat screwdriver and brush were wrongly matched because their relative part sizes were more similar than those between the variety of screwdrivers. By enforcing greater consistency in global geometry (all the screwdrivers have limbs protruding from the centre of the handle but not the brushes), the classification errors in the screwdrivers

were reduced at that the expense of the articulated pliers. Such conflicting requirements are difficult to resolve unless high-level knowledge is brought to bear on the shape classification process. For example, the human observer probably knows from experience, what configuration of tools can articulate (e.g. pliers) and what cannot (screwdrivers) and would apply such knowledge to adapt the constraints employed in judging shape similarity.

As a comparison, both the shock-based methods of Sebastian *et al.*'s edit distance shock-graph matching [191] and Pelillo *et al.*'s rooted shock-tree matching [163] have five within-category mismatches versus the MGVF-PBD's four. Closer examination shows that the rooted shock-tree matching erroneously matched shapes that are perceptually very distant, like the male head profile and fat screwdriver being matched to the hands. From the point of view of intuitive shape similarity, the edit distance shock-graph matching did better than the shock-tree but not as well as the MGVF-PBD. Categories like the hands and horses, though taxonomically distinct, are related by the fact that they both have several elongated sequentially-placed limbs protruding from a larger body. The MGVF-PBD results in Fig. 6.29a and 6.29b do reflect this fact, especially when the pro-articulation parameter settings are employed. These results are better than the shock graph edit distance in Fig. 6.29c. This seems to suggest that the MGVF-PBD recalls intuitively similar shapes better than both the shock-based and graph matching approaches of Sebastian *et al.* and Pelillo *et al.*

6.6.8 Shape-based Image Database Retrieval

Content-based image retrieval relies on the inherent visual features of the query image to retrieve all similarly relevant images within the image database. One very effective feature is the shape of the object [24], [147]. In this set of experiments, the effectiveness of the MGVF-PBD as a feature for shape-based image retrieval was studied. *Precision* and *recall* are commonly used measures of the quality for the retrieved images at any given point in the retrieval process [149], [188], [206]. In the context of this work, precision is defined as

$$\text{Precision} = \frac{\text{Number of shapes in the correct category retrieved so far}}{\text{Number of shapes retrieved so far}}$$

And recall is defined as

$$\text{Recall} = \frac{\text{Number of shapes in the correct category retrieved so far}}{\text{Number of shapes in this category in the database}}$$

Precision and recall values are plotted against each other to give a graph that would allow one to visualize the quality of the retrieval result. They both have an upper bound of 1. With precision, a value of 1 means all the shapes retrieved so far are relevant and with recall, it means all the relevant shapes in the current category have been retrieved. A perfect retrieval will feature a precision-recall graph that is a horizontal line, with the precision value at 1.0 for all recall values until it makes a sharp drop to 0.0 when the recall value reaches 1.0. In the results presented here, each image in the database is used as a query image and all the individual precision-recall graphs are averaged to form a single graph that summarises the overall performance of the shape descriptor's retrieval quality for that particular database. The two test databases used by Sebastian *et al.* in [191] were employed in these experiments. The first features 99 shapes (9 categories of 11 shapes each) and the second features 216 shapes (18 categories of 12 shapes each). The complete databases and their respective MGVF skeletons can be found in Appendix E.

The first set of results using the 99-shapes database was obtained by matching each shape in the database with all others. Recognition rates in percentage by this measure are (100, 100, 100, 100, 100, 100, 100, 99, 91, 88). In other words, the top 7 best matches are always correct when any shape is queried against the remaining 98 shapes in the database. This performance is better compared to that of the shock graph edit-distance results reported in [191], which was only (100, 100, 100, 99, 99, 99, 97, 96, 95, 87). An alternative presentation of this result is shown in the precision-recall graph in Fig. 6.30, which compares the retrieval performance on the 99-shapes database for three shape matching techniques, namely the MGVF-PBD, the shock graph edit-distance of Sebastian *et al.* [191] and the shape context of Belongie *et al.* [18]. Data for the latter two techniques were reproduced from [191]. It is clear from the graph that shape context, a global shape descriptor, did not perform as well on the 99-shapes database compared to the other two part-based descriptors. Of these two part-based descriptors, the MGVF-PBD outperforms the shock graph edit-distance. Fig. 6.31 shows several representative 99-shapes database retrieval results for these two descriptors. Notice the incorporation of global distance in the MGVF-PBD, which encodes inter-part spatial and angular relationships to help avoid the mismatches between the dog and human-figure evident in the shock graph edit

approach (see Fig. 6.31b), which is strictly local in its description of parts. However, the side-effect of using global geometric constraints is the reduced robustness to occlusion, as is evident in the fact that one occluded rabbit (subtractive occlusion) could not match the other occluded rabbit (additive occlusion) within the 10 closest matches (see Fig. 6.31a).

Figure 6.30 – The precision-recall graph demonstrating the comparative retrieval quality of the MGVF-PBD, shock graph edit-distance and shape context using Sebastian *et al.*'s 99-shapes database. For comparison purposes, the MGVF-PBD result was incorporated into the precision-recall graph presented in [191].

Figure 6.31 – A few representative retrieval results using Sebastian *et al.*'s 99-shapes database [191]. (a) 12 closest matches using the MGVF-PBD, ordered by their bi-directional shape distance. (b) 12 closest matches using shock graph edit distance, ordered by their normalized edit distance. Reproduced from Sebastian *et al.* results in [191]. Erroneous within-category matches are highlighted in red (✖).

The second set of MGVF-PBD results using the larger 216-shapes database resulted in recognition rates of (100, 100, 100, 100, 100, 99, 98, 97, 95, 87, 74). In other words, the top 5 best matches are always correct when any shape is queried against the remaining 215 shapes in the database. This performance is again better than that of the shock graph edit-distance reported in [191], which only had 100% error-free retrieval for the top 3 best matches, with recognition rates of (100, 100, 100, 99, 97, 99, 96, 96, 95, 91, 80). The more accurate retrieval performance of the MGVF-PBD is also reflected in the precision-recall graph in Fig. 6.32. It is far better than shape context and marginally better than the shock graph edit. However, for this database, the shock graph edit exhibited more graceful drop-off in performance as the remaining same-category shapes in the database became more difficult to retrieve. One or two shapes in several categories of the 216-shapes database did pose problems for both the MGVF-PBD and shock graph edit as highlighted in several representative comparative retrieval results shown in Fig. 6.33. For example, MGVF-PBD had some problems with the bird and elephant categories (see Fig. 6.33a) whereas the stingray category (see Fig. 6.33b) was more problematic for shock graph edit-distance. Detailed results for both 99-shapes and 216-shapes database are given in Appendix F.

Figure 6.32 – The precision-recall graph demonstrating the comparative retrieval quality of the MGVF-PBD, shock graph edit-distance and shape context using Sebastian *et al.*'s 216-shapes database. For comparison purposes, the MGVF-PBD result was incorporated into the precision-recall graph presented in [191].

Figure 6.33 – A few representative retrieval results using Sebastian *et al.*'s 216-shapes database [191]. (a) 13 closest matches using the MGVF-PBD, ordered by their bi-directional shape distance. (b) 13 closest matches using shock graph edit distance, ordered by their normalised edit distance. Reproduced from Sebastian *et al.* [191].

6.6.9 MGVF-PBD Parameters

In this series of experiments, several parameters of the MGVF-PBD were altered to study how their values affected the shape classification performance of the descriptor.

6.6.9.1 Part Orientation Histogram Bin Size

These bin sizes given by n , have a significant influence on the data storage requirement of the shape descriptor (see Appendix G). Fig. 6.34a shows the precision-recall graphs for four increasing values of n when the MGVF-PBD was used to perform shape retrieval on the 99-shapes database. Except for n , all other parameter settings were similar to those in Appendix D. It can be observed that as the number of bins increases, the performance of the shape descriptor also increases. This improvement is due to the increased resolution used to encode the gradient vector field characteristics of both the skeletal part's shape and its connectivity to other skeletons. However, Fig. 6.34b shows that this performance improvement does not go on indefinitely and at some point, it levels off. In fact, with higher values of n , a slight drop in performance was observed. When the binning is too fine, small variations in orientation can change the bin assignment during histogram construction. Since the chi-squared bin-by-bin dissimilarity measure is employed in comparing orientation histograms, a lower matching score could result due to bin misalignment [178]. Experimental results suggest that a good histogram bin size value is $n = 36$.

Figure 6.34 – The precision-recall graphs for the MGVF-PBD histogram bin size experiments. (a) Results for $n = 6$ to $n = 36$. (b) Results for $n = 36$ to $n = 60$.

6.6.9.2 Skeletal Curvature and Limb-width Polynomial Order

Figure 6.35 – The precision-recall graphs for the MGVF-PBD polynomial order experiment. Results were obtained using the 99-shapes database and parameter settings in Appendix D (except for q and r).

The skeletal curvature and limb width descriptors for each skeletal segment are encoded as coefficients of q -order and r -order polynomials respectively (see sections 5.5.3 and 5.5.4). A series of test runs were conducted to study if the order of these polynomials has an influence on the shape classification performance of the MGVF-PBD. Results in Fig. 6.35 show that moving from an order of 3 to 6 gave a slight improvement in performance. However, increasing the polynomial order from 6 to 9 did not yield further improvement. A high-order polynomial with a large number of coefficients is undesirable as it translates to a higher data storage requirement for the

part descriptors. Experimental results suggest that a good polynomial order for describing the skeletal curvature and limb width profile is $q = r = 6$.

6.6.9.3 MGVF smoothing parameter α

The smoothing parameter α given in (2.5) has a direct influence on the smoothness of the gradient vector field generated within the shape's interior. As such, it affects both the formation of the MGVF skeleton as depicted in Fig. 3.20 and the characteristic of the extracted part-based orientation histograms. Fig. 6.36 shows that the best performance was obtained when $\alpha = 0.9$. A lower value of $\alpha = 0.8$ may have smoothen the gradient vector field a little too much such that part shape and connectivity details may have been inadvertently blurred, leading to high misclassification. On the other hand, when the value of α was set higher at 0.95, the increased influence of boundary noise and artifacts within the orientation histogram may have also lead to more misclassifications. Results from these experiments suggest that an optimal value for the smoothing parameter is $\alpha = 0.9$.

Figure 6.36 – The precision-recall graphs for the MGVF-PBD smoothing parameter experiment. Results were obtained using the 99-shapes database and parameter settings in Appendix D (except for α).

6.6.9.4 Global Distance

This experiment explores the effect of removing the proposed global distance measure from the MGVF-PBD by setting the γ -weights given in (6.17) and (6.18) to zero. Fig. 6.37 shows a drastic drop in retrieval performance when inter-part spatial and angular relationships were not taken into consideration during shape matching. The nature of the mismatch is clearly observed in Fig. 6.38b when only local part constraints were employed in the matching. Many **Mammal** shapes were incorrectly matched to the hands because their legs found good local part-to-part matches with fingers of the hand. This result does lend support for the use of the global distance in the part-based descriptor, especially if there is a large within-class variation in the shapes.

Figure 6.37 – The precision-recall graphs for the MGVF-PBD global measure experiment. Results were obtained using the 99-shapes database and parameter settings in Appendix D (except for $\gamma_1 = \gamma_2 = \gamma_3 = \gamma_4 = 0$ for that without global distance).

Figure 6.38 – Shape retrieval results for the **Mammal** category when global distance was (a) employed and (b) when it was not. Within-category mismatches are shaded.

6.6.9.5 Skeletal Curvature and Limb Width Profile

This experiment explores the effectiveness of the two polynomial-based features that describe the curvature and the limb width profile along the skeleton. Fig. 6.39 shows a drop in retrieval accuracy when either feature was removed from the proposed descriptor. The sharper fall-off in retrieval accuracy in Fig. 6.39a, suggest that the skeletal curvature feature has a more significant effect on robust part-based description than the limb width profile shown in Fig. 6.39b. A closer look at some selected retrieval results in Fig. 6.40 and Fig. 6.41 could explain why this is so.

Figure 6.39 – The 216-shapes database precision-recall graphs for the MGVF-PBD. (a) Results for the skeletal curvature and (b) limb width profile experiments. Appendix D settings used except $\beta_c = 0$ in (a) and $\beta_r = 0$ in (b) for d_p given in (6.7)

Figure 6.40 – Selected shape retrieval results from the 216-shapes database when the skeletal curvature was (a) employed and (b) when it was not. Mismatches are shaded.

Mismatches such as those for the **Heart** and **Camel** shapes shown in Fig. 6.40b suggest that explicit skeletal curvature information is important in distinguishing

shape classes. This is especially so since the segment histogram describing the shape of the part has been curvature-normalised to accommodate limb articulation. However, skeletal curvature saliency must also be employed alongside in order to distinguish when the skeletal curvature significantly alters the perceived class of the shape or when it is merely expressing the articulation of narrow limb parts.

In the case of limb width profile, the mismatches observed in Fig. 6.41b suggest that on the whole, this information does improve the database retrieval accuracy of the descriptor slightly. The improvement is not as significant as that seen with the skeletal curvature because limb width variations along the skeletal segment are also encoded in the segment histogram described in section 5.5.2.1, albeit not as robustly. This is especially so if the skeletal segment has been merged and contain multiple necks and limbs. In fact, in some cases, such as with the ‘fat-handle’ **Fork** shape, the exclusion of limb width profile allowed the part-based descriptor to be less sensitive to variation in limb thickness. As a result, the other ‘thin-handle’ **Fork** shapes in the database could be retrieved with better accuracy compared to the results shown in Fig. 6.41a. Like the case of global distance and limb articulation, the selection of appropriate weightages for the limb width profile and skeletal curvature given in (6.7) is a tricky problem. In some sense, it is dependent on a priori knowledge about the nature of the within-class shape variations. However, there can arise situations where a complicated shape may require lax limb width constraints for some parts (e.g. legs) but greater limb width constraints for others (e.g. body). Such shapes would probably be difficult to handle and best served with the compromise weight setting given in Appendix D.

Figure 6.41 – Selected shape retrieval results from the 216-shapes database when the limb width profile was (a) employed and (b) when it was not. Mismatches are shaded.

6.6.9.6 Skeletal Curvature Saliency

This experiment explores the effectiveness of employing skeletal curvature saliency in the computation of part distance. It was argued that if skeletal curvature is to be incorporated into the part-based descriptor, the curvature error between the two skeletal parts must be moderated with a measure that reflects how significantly this error affects the perceived overall shape similarity. Fig. 6.42a shows the best retrieval accuracy for a selected number of shapes when skeletal curvature was employed together with the suggested curvature saliency proposed in section 5.6.4. Fig. 6.42b shows the result when no skeletal curvature is employed at all. Shapes such as the **Heart** and **Camel** had many misclassifications. These classes of shapes are distinctly defined by unique curvature of their backbone structural skeletons. When the weightage of the skeletal curvature error in computing part distance is increased by a little (see Fig. 6.42c), the mismatches in the **Heart** and **Camel** were reduced by a little. Further increase in curvature weightage (see Fig. 6.42d), further improved their retrieval accuracy. However, not moderated by the proposed skeletal curvature saliency, the large curvature variations in the trunk of the **Elephant** and tail of the **Ray** significantly increase the mismatches in these two categories. Such results support the use of skeletal curvature saliency.

Figure 6.42 – Selected shape retrieval results from the 216-shapes database when the (a) skeletal curvature saliency was employed. (b) Results when no skeletal curvature

saliency is used. Result when the skeletal curvature weightage given in (6.7) is set at (c) $\beta_c = 0.05$ and (d) $\beta_c = 0.2$.

6.6.9.7 Part Saliency

This experiment explores the effectiveness of employing part saliency in the shape matching process. Fig. 6.42 shows a very quick fall-off in retrieval accuracy when every skeletal segment was given equal weightage during computation of shape distance regardless of its formation stability or length. Mismatches observed in Fig. 6.38b suggest that the structural backbone of the **Greeble**, which is straight, was not given enough weightage such that it could significantly affect the overall shape distance when matched to the slightly curved structural backbone of the rabbits. This result shows why it is important to employ part saliency and that the proposed measure is reasonably effective in representing visual saliency.

Figure 6.43 – The precision-recall graphs for the MGVF-PBD part saliency experiment on the 99-shapes database, with parameter settings in Appendix D, except $G(S_j)$ given in (5.40) was set to unity for the test run that did not employ part saliency.

Figure 6.44 – Shape retrieval results for the **Greeble** category when part saliency was (a) employed and (b) when it was not. Within-category mismatches are shaded.

6.6.10 Human Visual Perception

Goldmeier [77] performed some studies on human perception of visual similarity in geometric shapes. He discovered that some visual properties featured more prominently in people's perception of similarity than others. For example, Fig. 6.45 shows the shapes used in a typical experiment in which human subjects were forced to decide which of the two shapes **2** or **3** is more similar to the reference shape **1**. It was discovered that the majority of subjects chose shape **3** despite the fact that one side of shape **2** is identical to shape **1**. This is because symmetry is a perceptually significant attribute that is used in judging shape similarity.

Figure 6.45 – Symmetry saliency experiment. Is shape **2** or **3** more similar to **1**?

Goldmeier's work also suggested that the human visual system attends to geometric properties he termed *singularities*. Line features that are vertical, horizontal or parallel for example, are considered singular because a small change in such properties makes a significance perceptual difference. A slanted line that is tilted slightly remains slanted. But the same angular tilt applied to a vertical line quickly destroy its perception of being vertical. By the same virtue, a shape varying within its nonsingular bound is usually perceived to have only changed slightly.

Figure 6.46 – Singularity saliency experiment. Is shape **2** or **3** more similar to **1**?

Consider the shapes in Fig. 6.46. In both cases, subjects considered reference shape **1** to be more similar to shape **3** than **2**. Consider the first case in Fig. 6.46a, despite the fact the overall thickness of shapes **1** and **2** were closer, subjects preferred shape **3**

because the straight line is preserved. The preservation of this singularity was perceptually more important in judging similarity than the thickness of the shape. In the case of Fig. 6.46b, the larger curvature variation between shapes **1** and **3** did not stop subjects from judging these two shapes to be more similar. This is because the large variation did not break the nonsingular bound unlike in shape **2**, which was at a singularity point since its base was straight. Such observations have been employed in the perceptually based relevance ranking of shapes [223], [224].

Several experiments were conducted using the MGVF-PBD and other shape descriptors to study if their respective matching cost reflects this perceptual ranking. Fig. 6.47 shows the results using the shapes of the symmetry saliency experiment. Only MGVF-PBD produced matching scores that correspond to the human perceptual ranking suggested by Goldmeier [77]. This seems to suggest that a global approach to shape description may not be how the human visual system perceives visual form.

Figure 6.47 – Symmetry saliency matching results for (a) MGVF-PBD (setting in Appendix D), (b) MGVF-AHN ($\lambda_{LG}=0.1$, $\alpha=0.9$), (c) shape context ($N=100$) [18], (d) Zernike moment descriptor ($n=10$) and (e) Fourier descriptor ($N=64$). Matching scores ($\times 1000$) given at the bottom-right corner show that only MGVF-PBD returns a matching score that corresponds to the human perception of shape similarity. Erroneous matches are marked with a (X).

The results for shapes used in the singularity saliency experiment were a little more mixed. Only MGVF-PBD and shape context produced results that were in total correspondence with the observed perceptual ranking. Other descriptors such as the global orientation histogram (MGVF-AHN), Zernike moments descriptor and Fourier

descriptor had a correct match for only one of the two reference shapes. In summary, this series of experiments seems to suggest that the MGVF-PBD is able to emulate the way humans employ visual features such as symmetry and geometric singularities to bias their similarity perception of geometric shapes. This characteristic is probably achieved through the combined use of a skeletal representation and the orientation histogram accumulated along the skeletal segments.

Figure 6.48 – Singularity saliency matching results for (a) MGVF-PBD, (b) MGVF-AHN, (c) shape context, (d) Zernike moment descriptor and (e) Fourier descriptor. Parameters for all descriptors and display are same as in Fig. 6.47. Only the MGVF-PBD and shape context matches correspond to the perceptual ranking for both reference shapes. Erroneous matches are marked with a (✗).

6.6.11 Computational Cost

Two aspects of the computational costs related to the MGVF-PBD were examined, namely the time taken to create and query a database. For comparative reasons, the same 25-shapes database in Fig. 4.30 was used and results from Table 4.4 are summarised together with those of the MGVF-PBD in Table 5.1. As expected, the time taken to create the database of descriptors for all 25 shapes is significantly higher than for the other descriptors. Unlike the global descriptors, the MGVF-PBD breaks a shape down into many parts and multiple features describing each part have to be individually computed. In typical database applications, the computation of shape descriptors need only be done once when a new shape is first entered into the database. Of more concern is the time to query the database. From Table 5.1, it can be seen that the MGVF-PBD performance is much poorer than all the other global descriptors except shape context. The total query time using the MGVF-PBD for the 25 shapes in the database is at least 10 times less than the shape context. Though both the matching algorithms uses the $O(N^3)$ Hungarian algorithm to solve a square assignment problem, the dimension of the MGVF-PBD's cost matrix was only between the sizes of 8×8 to 20×20 (for the 25-shapes database) compared to the shape context's 100×100 ($N=100$). Computational performance comparison with other part-based descriptors was not done due to the unavailability of public domain codes. Further discussion on improving MGVF-PBD's query time is in Appendix E.

Shape Descriptor	Shape Similarity Measure	Computational Cost (seconds)	
		Create database	Query database
MGVF-PBD	see section 5.7	390.80	95.13
AHN	d_χ and d_{L2} ($n=36$)	19.00	0.22
SC	d_χ ($N=100$)	42.38	1182.30
FD	d_{L2} ($N=64$)	1.87	0.06
ZMD	d_{L2} ($n=10$)	51.60	0.06

Table 6.1 – Comparative computational costs for extracting/matching the MGVF-PBD and other shape descriptors. Parameter settings for the MGVF-PBD are given in Appendix D. The times shown were measured on a 1.6 GHz Pentium-4 Personal Computer with 255 MB DRAM, running Windows 98® and Matlab® version 5.3 (see Table 4.4 for further details of the other descriptors).

6.7 Summary and Discussion

This chapter started out with the goal of addressing some of the shortcomings of the MGVF global shape descriptor, namely its inability to handle occlusion, limb articulation, part addition and deletion. A part-based descriptor that incorporates both the MGVF medial representation and the orientation histograms was proposed.

The MGVF skeleton provided the framework by which a 2D shape can be decomposed into skeletal-based parts. An earlier version of the part-based descriptor segmented the skeleton into continuous segments at skeletal junctions [71], [74]. However, it soon became clear that such skeletal decomposition is inadequate as many shapes can be perceived as having multiple neck and limb parts but can still be constituted using only a single continuous skeleton. This led to the introduction of a more comprehensive set of segmentation criteria, which also included segmentation at the maxima/minima points of the vector field disparity profile and the skeletal curvature, and also at connectivity end points. The motivation for each of the proposed criteria was discussed in the respective sections.

It was observed that shapes can deform in a continuous manner and segmentation that adopts a hard-threshold is not amenable to such variability. In order to avoid over-segmentation, the segmented skeletal fragments should be allowed to merge into longer continuous segments that would most likely represent single coherent parts. Two merging criteria were adopted. The first was motivated by the notion of visual path continuity suggested by Katz and Pizer [107] and is based on the orientation continuity between two skeleton axes at a branch point. The second is based on their disparity continuity. Since ligatures exhibit very small vector field disparity measures compared to the main structural skeleton that they inadvertently fragment, this measure can ensure ligatures and principal components of the structural skeleton do not merge at branch points (see Fig. 5.9). The end result of the segmentation and merging stage is a set of secondary skeletal segments and the possible permutations that can be employed to reconstitute merged segments using different combinations of these skeletal elements. The selection of the optimal combination is deferred till the shape matching stage as this work is motivated by the view that part decomposition is a dynamic process, which can only be determined when the characteristics of the shape being compared is known.

Various features were proposed for endowing these skeletal segments with a rich set of quantitative shape-describing attributes. The monotonicity characteristic of this combination of features is evident from the skeletal topology experiment in Fig. 6.15. The continuity property was also exhibited by the descriptors in the presence of increasing shape distortion due to partial occlusion and viewpoint variations, as seen in the results of the experiments in section 6.6.5. Two of these features, the skeletal curvature and limb width profile are described along the length of the skeletal segment. In order to make the descriptor compact and independent of the skeletal length, they were encoded as coefficients of an n^{th} -order polynomial. Experiments in section 6.6.9.2 suggest that a small order of 6 is sufficient to effectively describe such parametric variations along the skeletal part, at least for shapes such as those in the 99-shapes database [191]. The importance of the curvature and limb-width features are demonstrated by the deterioration in the shape retrieval accuracy when they were excluded from the MGVF-PBD, as highlighted by the experiments in section 6.6.9.5.

Other part-based features employed included a set of three orientation histograms that described various important characteristics of the skeletal part. Firstly, the terminus of the part is described with a terminus histogram, which is made robust through the accumulation of orientation histograms along an appropriate length of the skeletal segment. This strategy allowed tapered ends to build statistically significant orientation histograms and it also produced terminus descriptors that are in conformance to the manner in which humans perceive the terminus characteristics of limbs as they elongate. Secondly, skeletal ends at branch points are described by a connection histogram. As illustrated in Fig. 5.8, this descriptor is effective in distinguishing the manner in which disparate parts are connected as the nature of this connectivity can significantly change the perception of the shape. Thirdly, the segment histogram described the curvature-normalised shape of the part, which is necessary if the part descriptor is to handle limb articulation and bending. The effectiveness of curvature-normalisation along the skeletal segment can be observed in the reasonably good retrieval results obtained for the elephant (see Fig. F.2a) and ray (see Fig. F.2b) categories of the 216-shapes database, despite the elephants' trunk and rays' tail exhibiting high bending variability.

The robustness to limb bending can also be attributed to the use of skeletal curvature saliency (see section 5.6.4). Since skeletal curvature is employed in the part descriptor, its contribution to the overall shape distance should be selectively reduced

in limbs that are narrow, like the tails of the ray. The results in Fig. 6.42 show the effectiveness of employing the proposed skeletal curvature saliency. Two other measures of visual saliency were also proposed. Disparity-based saliency is employed to reduce the influence of unstable ligatures and length-based saliency is employed to ensure that longer limbs are given more influence on the shape distance computation than shorter limbs. The effectiveness of this combined part saliency is clearly observed in the precision-recall graphs in Fig. 6.43.

Several part-based dissimilarity measures were proposed for computing part-to-part matching cost. The first is the part distance, which measures the local dissimilarity between two parts based on the weighted combination of distances between their respective part-related features. The second is the global distance, which measures the dissimilarity between the spatial and angular inter-part relationships of each part within a shape and those of another shape. These inter-part relationships are computed with reference to some global attributes within the shape, namely its centroid and an appropriately selected source node. It was argued that such global characteristics must be employed if large local distortion is permitted within shapes of the same category. The experimental result observed in Fig. 6.37 does support this claim. When the global distance was excluded from the computation of the cost matrix, a significant drop in the shape retrieval accuracy for the 99-shapes database was observed. The pros and cons of incorporating global features into the part-based descriptor were highlighted in several experiments. For example, in the limb articulation experiments of section 6.6.6, it was demonstrated that the global distance emphasis in the MGVF-PBD must be reduced in order to increase its robustness to limb articulation. In that example, the change in limb position, which invariably affected the shape's global geometry was more significant than the subtle differences separating the shape classes (e.g. **Thin man** vs. **Fat man**).

The shape-based dissimilarity measure proposed was the shape distance. This non-metric measure employs the part distance, global distance and part saliency measure to decide which of the many combinations of skeletal segments in the shape should be adopted in the final computation of shape dissimilarity. This selection process is guided by using the minimum saliency-weighted part-to-part matching cost but it cannot be considered an optimal search. An exhaustive search for the combination of skeletal segments that would give the true lowest shape matching cost would be too computationally costly for such an algorithm to be usable in practical

shape database retrieval applications. Once the sub-optimal part decomposition of the shape has been determined, a single matching score is computed by finding the optimal one-to-one part correspondence between the two shapes. Appropriate dummy parts are employed when the part count between the two shapes is not the same. A bi-directional shape distance was proposed to handle situations where the partial matching capability inherent in the shape distance, does not produce appropriate shape dissimilarity scores. This occurs when the query shape has more parts than the shape it is being compared with.

A comprehensive series of experiments were conducted to study the characteristics of the proposed MGVF-PBD. Its invariance to basic geometric transforms such as scaling and arbitrary rotation was demonstrated. Problems were observed when the shape was too small as extracting statistically reliable orientation histograms was difficult due to the lack of gradient samples within the shape. Mirror reflection could be handled through appropriate transformation of certain features sensitive to angular variations such as the orientation histograms and global distance (angular aspects). It was also shown that the proposed descriptor is remarkably resilient to boundary noise and texture, which is usually the strength of region-based global descriptors and not part-based descriptors. The boundary texture experiments in section 6.6.4 showed that the partial matching capabilities of the proposed shape distance could be exploited when clean template shapes have to be matched to poorly segmented regions when performing object recognition in natural, noisy or cluttered scenes. Experiments on occlusion and limb articulation demonstrated the goal of devising a shape descriptor that is able to overcome the limitations of the MGVF global descriptors has been achieved with the MGVF-PBD, albeit at an increase in computational cost, both for descriptor extraction and comparison. The data storage requirements of the descriptor has also increased and is discussed in Appendix G. Results from the shape classification experiments on several mid-sized databases showed that the shape retrieval accuracy of the MGVF-PBD is comparable if not better than the state-of-the-art part-based shape matching techniques currently available in literature. Some experiments on geometric attributes such as symmetry and singularities showed that the matching scores using the MGVF-PBD is in congruence with human shape similarity perception. Similarly consistent results were not achievable with several other popular shape descriptors.

Chapter 7

Conclusions

This thesis started out with the hypothesis that the vector field generated by migrating the boundary gradient information into the interior region would provide a useful basis for 2D shape description. It aims to propose a suitable framework for generating the gradient vector field and develop techniques that would exploit the embedded information for robust description of 2D shapes. To this end, several principle results and conclusions were reached and they are summarised here.

7.1 Principle Results

A computationally efficient multiresolution pyramidal framework has been proposed for generating the MGVF. A single free parameter α can be used to determine the smoothness of the vector field. Selection of an appropriate value of α would allow both the detailed boundary and gross shape characteristics to be simultaneously represented. Subsequent investigations have proven that the MGVF is indeed useful as a basis for robust 2D shape description.

The multi-level integration of the Vector Field Disparity pyramid to form the disparity map $M(x,y)$ was proposed as a way for extracting scale invariant shape axes from the MGVF. The subsequent ridge extraction and thinning algorithm was designed to recover a disconnected MGVF skeleton. As suggested in section 1.1, a disconnected skeleton is most appropriate for representing shapes with high boundary noise. The results obtained demonstrate that the extracted MGVF skeleton is more consistent under varying scale and boundary noise conditions compared to other skeletonization algorithms (see section 3.4.2). The application of the same ridge extraction and thinning process on the Euclidean Distance Transform (EDT) of the shape did not produce similarly robust results. This suggests that the characteristics of the MGVF itself, is what permits the robust separation of structural skeleton and textural skeleton. The former describing the gross shape of the object whilst the latter

its boundary characteristics. The MGVF skeleton's invariance to arbitrary rotation was also observed and this desirable characteristic could not be obtained with popular iterative thinning algorithms such as Zhang and Suen's [230].

The main contribution of this thesis is the proposed part-based descriptor. Its characteristics and performance are well summarised in section 6.. However, what should be highlighted are several aspects of the proposed descriptor that have contributed significantly to its superior shape retrieval performance over state-of-the-art techniques such as the shock graph edit distance [191]. Firstly, breaking away from the traditional restriction of keeping part-based descriptors strictly local in nature has definitely help improve performance (see Fig. 6.37). The use of global distance, which describes how parts within a shape are related to selected global references, has embedded some sense of global geometry into the MGVF-PBD. This allowed it to tolerate large within-class variations at the risk of increasing sensitivity to occlusion and limb articulation. However, sensitivity to limb articulation was ameliorated by defining spatial and angular relationship at the locality of the source node, which tends to be the limb end that is closer to the body and therefore not significantly affected by articulation. The search technique for selecting the reference source node was also designed to minimise the effect occlusion has on locating consistent global references, albeit at a high computational cost. Compromise between query time and retrieval performance is discussed in Appendix C.

Secondly, the incorporation of various saliency measures such as the skeletal curvature and part saliency (see Fig. 6.43) have also contributed significantly to the performance of the MGVF-PBD. One reason for their effectiveness is the fact that the proposed disparity-based saliency is intricately related to the stability and significance of different segments of the MGVF skeleton. Since these very segments are used to defined parts of the shape, the proposed part saliency appropriately measures the influence each part should have on the overall shape distance. In so doing, the perennial medial representation concern about ligatures [11] has also been addressed.

On the whole, the proposed MGVF-PBD has demonstrated its ability to handle many common shape variations such as occlusion, limb articulation, part addition and deletion. It is also robust to boundary noise and has demonstrated its effectiveness as a descriptor for shape-based image database retrieval. Another interesting observation is its conformance to some of Goldmeier's [77] observations of human perception of visual similarity such as symmetry and singularity.

7.2 Future Work

In order to make the proposed MGVF-PBD usable in practical content-based image retrieval (CBIR) applications, the shape database retrieval experiments must be conducted using a much larger database of several thousand images. Unless this method of shape analysis can be demonstrated to scale up well, it will not be of much interest to the CBIR community. From this point of view, several issues require further examination. The computational speed of shape query must be improved. The current Matlab® (ver. 5.3) code running uncompiled at an average of 95 seconds per query is not indicative of the best performance of MGVF-PBD but other optimisation strategies besides porting the code to a compiled language like C++ are needed. Several strategies have yet to be explored and evaluated. One would be to combine both the global and part-based descriptors. The computationally more efficient global descriptor could be used to reduce the search space before the better performing MGVF-PBD is then used to rank the retrieved images within this subspace. Another is to employ a distinctive separation between each shape's structural and textural skeletons. Coarse search is first performed using only the structural skeleton and then more refined search on say, the top 50 best matches is then performed using all the skeletal parts. Data storage requirement for the part-based descriptor is another issue relevant to this scaling up problem. Some possible improvements in this area are discussed in Appendix G.

Another area of future work is to develop a framework for full affine invariant recognition that includes both skew and stretch. So far the MGVF-PBD descriptor has been shown to be able to handle translation, scaling and rotation. As for skew and stretch, some preliminary experimentation has been done along the approach used by Baumberg [15] and Mikolajczyk and Schmid [148] for addressing problems related to wide baseline stereo matching and affine invariant interest point detection. They proposed using the second moment matrix computed at an image locality to normalise a region in an affine invariant way. In the examples shown in Fig. 7.1, the MGVF within the entire region was used to construct the second moment matrix. An iterative algorithm was then implemented to progressively normalise the shape to an affine invariant version, stopping when the second moment matrix is close to the unity (i.e. identity matrix). If all shapes in the database were similarly normalised, then full affine invariant shape query could take place. Further studies have to be conducted to

analyse the shape retrieval performance for such a scheme. The example in Fig. 4.10 suggests that full affine invariance may not be sufficiently discriminating for some classes of shapes.

Figure 7.1 – Affine invariant normalisation of shapes using the MGVF-based second moment matrix. (a) Examples of binary shapes and their respective normalised version. The skeletons can be easily matched since the MGVF-PBD is invariant to scaling and rotation. (b) Examples of vectorised handwritten test digits [5] undergoing normalisation based on the MGVF generated within the enclosing circle. The slant and stretch variations have been reduced after normalisation.

A serious issue that has only been partially addressed in a tedious empirical fashion is the selection of appropriate values for the many MGVF-PBD parameters listed in Appendix D. Fine tuning these large number of variables in order to obtain good shape retrieval performances for different databases has not been straightforward and involved many hours of trial and error experimentation. As discussed in sections 6.6.7 and 6.6.9.5, the pre-selected weightages for features such as global distance, segment histogram and limb width profile are dependent on the characteristics of the shape and its within-class variations. A priori information such

as whether the class of objects can articulate or the extent of their within-class limb width variation and bending can be useful in implementing class-dependent adaptive weightage. In other words, the settings given in Appendix D should not be made fixed for all queries. When a new shape is queried with say a **Human** shape, then values closer to the pro-articulation settings (see section 6.6.6) should be adopted. Such adaptive MGVF-PBD parameters could be taught to the shape retrieval system using some user-interactive relevance feedback framework [85], [180]. Based on users' feedback on what constitute good matches, the standard MGVF-PBD parameters in Appendix D can be systematically adjusted to reduce the computed shape distance. The new settings are incrementally learned by the system and associated with the registered shape class. With sufficient training, the parameters for each shape class

References

- [1] G.J. Abdel-Hamid and Y.H. Yang, "Multiresolution Skeletonisation: An Electrostatic Field-Based Approach" in the *Proceedings of IEEE International Conference on Image Processing (ICIP'94)*, vol. 1, pp. 949-953, 1994.
- [2] K. Abe, C. Arcelli, T. Hisajima and T. Ibaraki, "Parts of Planar Shapes," *Pattern Recognition*, vol. 29, no. 10, pp. 1703-1711, 1996.
- [3] A.S. Aguado, E. Montiel and M.S. Nixon, "Invariant Characterisation of the Hough Transform for Pose Estimation of Arbitrary Shapes," *Pattern Recognition*, vol. 35, pp. 1083-1097, 2002.
- [4] J.M. Allman and J.H. Kaas, "Representation of the Visual Field in Striate and Adjoining Cortex of the Owl Monkey," *Brain Research*, vol. 35, pp. 89-106, 1971.
- [5] E. Alpaydinlman and F. Alimoglu, "Pen-Based Recognition of Handwritten Digits Database," Department of Computer Engineering, Bogazici University, 80815 Istanbul, Turkey, available at <http://www.inf.fu-berlin.de/lehre/SS01/NeuronaleNetze/pendigits/>, July 1998.
- [6] C. Arcelli, "Pattern Thinning by Contour Tracing," *Computer Graphics and Image Processing*, vol. 17, pp. 130-144, 1981.
- [7] C. Arcelli and G. Sanniti Di Baja, "Ridge Points in Euclidean Distance Maps," *Pattern Recognition Letters*, vol. 13, pp. 237-243, 1992.
- [8] C. Arcelli and G. Sanniti Di Baja, "Euclidean Skeleton vis Centre-of-Maximal-Disc Extraction," *Image and Vision Computing*, vol. 11, no. 3, pp. 163-173, 1993.
- [9] E.M. Arkin, L.P. Chew, D.P. Huttenlocher, K. Kedem and J.S.B. Mitchell, "An Efficiently Computable Metric for Comparing Polygonal Shapes," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 13, no. 3, pp. 209-216, 1991.
- [10] A. Arumugan, T. Radhakrishnan and C.Y. Suen, "A Thinning Algorithm based on the Force Between Charged Particles," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 16, no.1, pp. 69-83, 2002.
- [11] J. August, K.S. Siddiqi and S.W. Zucker, "Ligature Instabilities in the Perceptual Organization of Shape," in the *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR'99)*, vol. 2, pp. 42-48, 1999.
- [12] D.H. Ballard and C.M. Brown, *Computer Vision*, New Jersey: Prentice Hall, 1982.
- [13] G.S. di Baja and E. Thiel, "(3,4)-weighted Skeleton Decomposition for Pattern Representation and Description," *Pattern Recognition*, vol. 27, no. 8, pp. 1039-1049, 1994.
- [14] R. Basri, L. Costa, D. Geiger and D. Jacobs, "Determining the Similarity of Deformable Shapes," *Vision Research*, vol. 38, pp. 2365-2385, 1998.

- [15] A. Baumberg, "Reliable Feature Matching Across Widely Separated Views," in the *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR'00)*, vol. 1, pp. 774-781, 2000.
- [16] G.C. Baylis and J. Driver, "One-sided Edge Assignment in Vision. 1. Figure-ground Segmentation and Attention to Objects," *Current Directions in Psychological Science*, vol. 4, no. 5, pp. 140-146, 1995.
- [17] S. Belongie and J. Malik, "Matching with Shape Context," in the *Proceedings of IEEE Workshop on Content-based Access of Image and Video Libraries (CBAIVL'00)*, pp. 20-26, 2000.
- [18] S. Belongie, J. Malik and J. Puzicha, "Shape Matching and Object Recognition using Shape Contexts," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, no. 24, pp.509-522, 2002.
- [19] S. Belongie, J. Malik and J. Puzicha, "Matlab Demo Code for Shape Context Matching," available from Computer Science Division, University of California, Berkeley at http://www.cs.berkeley.edu/~sjb/shape/sc_demo/, 2003.
- [20] J. Ben-Arie, K.R. Rao and Z. Wang, "Robust Shape Description and Recognition by Gradient Propagation," in the *Proceedings of the International Conference on Image Processing (ICIP'95)*, vol. 3, pp. 368-371, 1995.
- [21] J. Ben-Arie and Z. Wang, "Hierarchical Shape Description and Similarity-Invariant Recognition using Gradient Propagation," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 15, no.8, pp. 1251-1261, 2001.
- [22] J. Ben-Arie and Z. Wang, "Shape Description and Invariant Recognition Employing Connectionist Approach," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 16, no.1, pp. 69-83, 2002.
- [23] T.M. Bernard, A. Manzanera, "Improved Low Complexity Fully Parallel Thinning Algorithm," in the *Proceedings of the 10th International Conference on Image Analysis and Processing (ICIAP'99)*, Venice, Italy, pp. 215-220, 1999.
- [24] S. Berretti, A.D. Bimbo and P. Pala, "Retrieval by Shape Similarity with Perceptual Distance and Effective Indexing," *IEEE Trans. on Multimedia*, vol. 2, no. 4, pp. 225-239, 2000.
- [25] I. Biederman, "Recognition-by-Components: A Theory of Human Image Understanding," *Psychological Review*, vol. 94, no.2, pp. 115-147, 1987.
- [26] I. Biederman, "High-level Vision," in D.N. Osherson, S. Kosslyn and J. Hollerbach, Eds., *An Invitation to Cognitive Science: Visual Cognition and Action*, Cambridge, MA: MIT Press, pp. 41-72, 1990.
- [27] J. Bigün, G.H. Granlund and J. Wiklund, "Multidimensional Orientation Estimation with Application to Texture Analysis and Optical Flow," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 13, no. 8, pp. 775-790, 1991.
- [28] T.O. Binford, "Visual Perception by Computer," in the *Proceedings of IEEE Conference on Systems and Control*, Miami, Florida, 1971.

- [29] A. Biran and M. Breiner, *Matlab for Engineers*, Wokingham, England: Addison-Wesley, 1995.
- [30] H. Blum, "A transformation for Extracting new Descriptors of Shape," in W. Wathen-Dunn, Ed., *Models for the Perception of Speech and Visual Form*, Cambridge, MA: MIT Press, pp. 363-380, 1967.
- [31] H. Blum, "Biological Shape and Visual Science (Part I)," *Journal of Theoretical Biology*, vol. 38, pp. 205-287, 1973.
- [32] H. Blum and R.N. Nagel, "Shape Description using Weighted Symmetric Axis Features," *Pattern Recognition*, vol. 10, pp. 167-180, 1978.
- [33] M.L. Boas, *Mathematical Methods in the Physical Sciences*, New York: John Wiley & Sons, 1966.
- [34] G. Borgefors, "Distance Transformation in Digital Images," *Computer Vision, Graphics, and Image Processing*, vol. 34, pp. 344-371, 1986.
- [35] A.I. Borisenko and I.E. Tarapov, *Vector and Tensor Analysis*, New York: Dover Publication, 1968.
- [36] R.N. Bracewell, *The Fourier Transform and its Applications*, New York: McGraw Hill, 1978.
- [37] H. Bunke and U. Bühler, "Applications of Approximate String Matching to 2D Shape Recognition," *Pattern Recognition*, vol. 26, no. 12, pp. 1797-1812, 1993.
- [38] M. Brady and H. Asada, "Smoothed Local Symmetries and their Implementation," *International Journal of Robotics Research*, no. 3, pp. 33-61, 1984.
- [39] H. du Buf and M.M. Bayers (eds.), "Automatic Diatom Identification," in *Series in Machine Perception and Artificial Intelligence*, vol. 51, Singapore: World Scientific Publishing Co., 2002.
- [40] H. Bunke and U. Bühler, "Applications of Approximate String Matching to 2D Shape Recognition," *Pattern Recognition*, vol. 26, no. 12, pp. 1797-1812, 1993.
- [41] P.J. Burt and E.H. Adelson, , "The Laplacian Pyramid as a Compact Image Code," *IEEE Trans. On Communication*, vol. 31, no. 4, pp. 532-540, 1983.
- [42] P.J. Burt, "The Pyramid as a Structure for Efficient Computation," in A. Rosenfeld, Ed., *Multiresolution Image Processing & Analysis*, Berlin Heidelberg New York: Springer-Verlag, pp. 6-35, 1984.
- [43] K.R. Castleman, *Digital Image Processing*, Englewood Cliffs, New Jersey: Prentice Hall, 1996.
- [44] P. Cavanagh, "Pathways in Early Vision," in Z.W. Pylyshyn, Ed., *Computational Processes in Human Vision: An Interdisciplinary Perspective*, Norwood, New Jersey: Ablex Publishing Corporation, pp. 239-261, 1988.
- [45] K.Y Chang, "Lans Pattern Recognition Toolbox," available from Laboratory of Artificial Neural Systems, Department of Electrical and Computer Engineering, University of Texas at <http://www.lans.ece.texas.edu/~lans/lans/>, 2003.

- [46] P. Ciaccia, M.F. Patella, P. Zezula, "M-tree: An Efficient Access Method for Similarity Search in Metric Spaces," in the *Proceedings of the Conference on Very Large Databases*, pp. 522-525, 1997.
- [47] J. Clark and D.A. Holton, *A First Look at Graph Theory*, Singapore: World Scientific Publishing Co., 1991.
- [48] J. Cortadellas, J. Amat and F. de la Torre, "Robust Normalization of Silhouettes for Recognition Applications," *Pattern Recognition Letters*, vol. 25, pp. 591-601, 2004.
- [49] J.R. Czar, N. Guil and E.L. Zapata, "Detection of Arbitrary Planar Shapes with 3D Pose," *Image and Vision Computing*, vol. 19, pp. 1057-1070, 2001.
- [50] A.D.J. Cross and E.R. Hancock, "Scale Space Vector Fields for Symmetry Detection," *Image and Vision Computing*, vol. 17, pp. 337-345, 1999.
- [51] O. Cuisenaire, "Distance transformations: fast algorithms and applications to medical image processing," *Ph.D Thesis*, UCL/TELE, Louvain-la-Neuve, 1999.
- [52] P.E. Danielsson, "Euclidean Distance Mapping," *Computer Graphics and Image Processing*, vol. 14, no. 1, pp. 227-248, 1980.
- [53] E.R. Davies and A.P.N. Plummer, "A New Method for the Compression of Binary Picture Data," in the *Proceedings of the 5th International Conference on Pattern Recognition*, pp. 1150-1152, 1980.
- [54] E.A. DeYoe and D.C. van Essen, "Concurrent Processing Streams in Monkey Visual Cortex," *Trends in Neuroscience*, vol. 11, no. 5, pp. 219-226, 1988.
- [55] P. Dimitrov, C. Philips and K. Siddiqi, "Robust and Efficient Skeletal Graphs," in the *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR'00)*, Hilton Head, South Carolina, vol. 1, pp. 417-423, 2000.
- [56] J. Driver and G.C. Baylis, "One-sided Edge Assignment in Vision. 2. Part Decomposition, Shape Description, and Attention to Objects," *Current Directions in Psychological Science*, vol. 4, no. 5, pp. 201-206, 1995.
- [57] J.P. Eakins, J.M. Boardman and M.E. Graham, "Similarity Retrieval of Trademark Images," *IEEE Multimedia*, vol. 5, no. 2, pp. 53-63, 1998.
- [58] S.J.. Ed, "MPEG-7 Visual Part of Experimentation Model Version 5.0," *ISO/IEC JTC1/SC29/WG11/N3321*, Nordwijkerhout, March, 2000.
- [59] D.J. Fellman and D.C. van Essen, "Distributed Hierarchical Processing in the Primate Cerebral Cortex," *Cerebral Cortex*, vol. 1, pp. 1-47, 1991.
- [60] B. Fischer, "Overlap of Receptive Field Centers and Representation of the Visual Field in the Cat's Optic Tract," *Vision Research*, vol. 13, pp. 2113-2120, 1970.
- [61] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang and B. Dom, et al. "Query by Image and Video Content: ABIC System," *IEEE Computers*, vol. 28, no. 9, pp. 23-32, 1995.
- [62] M.M. Fleck, "Local Rotational Symmetries," in the *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 332-337, 1986.

- [63] J. Flusser and T. Suk, "Pattern Recognition by Affine Moment Invariants," *Pattern Recognition*, vol. 26, pp. 167-174, 1993.
- [64] H. Freeman, "On the Encoding of Arbitrary Geometric Configurations," *IRE Trans. on Electronic Computers*, vol. EC-10, pp. 260-268, 1961.
- [65] Y. Gdalyahu and D. Wienshall, "Flexible Syntactic Matching of Curves and its Application to Automatic Hierarchical Classification of Silhouettes," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 21, no. 12, pp. 1312-1328, 1999.
- [66] Y. Ge and J.M. Fitzpatrick, "On the Generation of Skeletons from Discrete Euclidean Distance Maps," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 18, no. 11, pp. 1055-1066, 1996.
- [67] D. Geiger, T.L. Liu and R.V. Kohn, "Representation and Self-Similarity of Shapes," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 25, no. 1, pp. 86-99, 2003.
- [68] P.J. Giblin and S.A. Brassett, "Local Symmetry of Plane Curves," *American Mathematical Monthly*, vol. 92, pp. 689-707, 1985.
- [69] P.J. Giblin and B.B. Kimia, "On the Local Form and Transitions of Symmetry Sets, Medial Axes and Shocks," in the *Proceedings of the 7th International Conference on Computer Vision (ICCV'99)*, pp. 385-391, 1999.
- [70] P.J. Giblin and B.B. Kimia, "A Formal Classification of 3D Medial Axis Points and Their Local Geometry," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 26, no. 2, pp. 238-251, 2004.
- [71] W.B. Goh and K.Y. Chan, "Shape Description Using Gradient Vector Field Histogram," in the *Proceedings of the 4th International Conference on Scale Space*, pp. 713-728, 2003.
- [72] W.B. Goh and K.Y. Chan, "Robust Shape Axes Extraction Using Multiresolution Gradient Vector Fields," in the *Proceedings of the IASTED International Conference on Signal Processing, Pattern Recognition, & Applications (SPPRA03)*, pp. 313-318, 2003.
- [73] W.B. Goh and K.Y. Chan, "Part-based Shape Recognition using Gradient Vector Field Histograms," in the *Proceedings of the 10th International Conference on Computer Analysis of Images and Patterns (CAIP 2003)*, pp. 402-409, 2003.
- [74] W.B. Goh and K.Y. Chan, "A Shape Descriptor for Shapes with Boundary Noise and Texture," in the *Proceedings of the 14th British Machine Vision Conference (BMVC 2003)*, vol. 1, pp. 201-210, 2003.
- [75] W.B. Goh and K.Y. Chan, "A Global Shape Descriptor based on Gradient Vector Field," in the *Proceedings of the International Workshop on Advanced Image Technology (IWAIT 2005)*, pp. 163-168, 2005.
- [76] S. Gold and A. Rangarajan, "A Graduated Assignment Algorithm for Graph Matching," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 18, no. 4, pp. 377-388, 1996.
- [77] E. Goldmeier, "Similarity in Visually Perceived Forms," *Psychological Issues*, vol. 8, no. 1, 1972.

- [78] A. Goshtasby, "Description and Discrimination of Planar Shapes using Shape Matrices," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 7, pp. 738-743, 1985.
- [79] G.H. Granlund, "Fourier Preprocessing for Hand Print Character Recognition," *IEEE Trans. on Computer*, vol. 21, pp. 195-201, 1972.
- [80] M.A. Grayson, "Shortening Embedded Curves," *Anal. of Mathematics*, vol. 129, pp. 71-111, 1989.
- [81] J. Gregor and M.G. Thomason, "Dynamic Programming Alignment of Sequences Representing Cyclic Patterns," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 15, no. 2, pp. 129-135, 1993.
- [82] L.D. Griffin and M. Lillholm, *Scale Space Methods in Computer Vision*, Berlin Heidelberg: Springer-Verlag, 2003.
- [83] T. Grigorishin and Y.H. Yang, "Skeletonization: An Electrostatic Field-Based Approach," *Pattern Analysis and Applications*, vol. 1, pp. 163-177, 1998.
- [84] W.I. Grosky and R. Mehrotra, "A Pictorial Index Mechanism for Model-based Matching," *Computer Vision, Graphics and Image Processing*, vol. 52, pp. 416-436, 1990.
- [85] G.D. Guo, A.K. Jain, W.Y. Ma and H.J. Zhang, "Learning similarity measure for natural image retrieval with relevance feedback," *IEEE Trans. on Neural Networks*, vol. 13, no. 4, pp. 811-820, 2002.
- [86] Z. Guo and R.W. Hall, "Fast Fully Parallel Thinning Algorithms," *Computer Vision, Graphics, and Image Processing: Image Understanding*, vol. 55, no. 3, pp. 317-328, 1992.
- [87] A. Guttman, "R-trees: A Dynamic Index Structure for Spatial Searching," in the *Proceedings of the ACM SIGMOD Conference on Management of Data*, pp. 47-57, 1984.
- [88] J. Hafner, H.S. Sawhney, W. Equitz, M. Flickner and W. Niblack, "Efficient Color Histogram Indexing for Quadratic Form Distance Function," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 17, no. 7, pp. 729-736, 1995.
- [89] B.M. ter Harr Romeny and L. Florack, "A Multiscale Geometric Model of Human Vision," in W.R. Hendee and P.N.T. Welss, Eds., *The Perception of Visual Information*, 2nd Edition, New York Berlin Heidelberg: Springer, pp. 87-126, 1997.
- [90] B.M. ter Harr Romeny, "Front-end Visual System 1," in a course offered to PhD Research Schools, *An Introduction to Scale-Space Theory in Computer Vision*, Faculty of Biomedical Engineering, Technical University Eindhoven and University of Maastricht, currently available at <http://www.bmt.tue.nl/imaging/FEV/index2002.htm>, 2002.
- [91] R.W. Hall, "Optimally Small Operator Supports for Fully Parallel Thinning Algorithms," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 15, no. 8, pp. 828-833, 1993.
- [92] Y.A. Hicks, A.D. Marshall, P.L. Rosin, R.R. Martin, M.M. Bayer and D.G. Mann, "Modelling Life Cycle Related and Individual Shape Variation in

- Biological Specimens," in the *Proceedings of the British Machine Vision Conference (BMVC'00)*, Cardiff, Wales, U.K., pp. 323-332, 2000.
- [93] C.J. Hilditch, "Linear Skeletons from Square Cupboards," in B. Meltzer and D. Michie, Eds., *Machine Intelligence*, New York: Elsevier, vol. 4, pp. 403-420, 1969.
  - [94] K. Hirata and T. Kato, "Query by Visual Examples: Content-based Image Retrieval," in the *Proceedings of the 3rd International Conference on Extending Database Technology (EDBT'92)*, vol. 580, pp. 56-71, 1992.
  - [95] D.D. Hoffman and W.A. Richards, "Parts of Recognition," *Cognition*, vol. 18, pp. 65-96, 1984.
  - [96] M.K. Hu, "Visual Pattern Recognition by Moment Invariants," *IRE Trans. on Information Theory*, vol. IT-8, pp. 179-187, 1962.
  - [97] D.H. Hubel and T.N. Wiesel, "Receptive Fields of Single Neurons in the Cat's Striate Cortex," *Journal of Physiology (London)*, vol. 148, pp. 574-591, 1959.
  - [98] D.H. Hubel and T.N. Wiesel, "Uniformity of Monkey Striate Cortex: A Parallel Relationship between Field Size, Scatter and Magnification," *Journal of Computational Neurology*, vol. 158, pp. 296-306, 1974.
  - [99] D.H. Hubel, *Eye, Brain, and Vision*, New York: Scientific American Library Series No. 22, 1988.
  - [100] B. Huet and E.R. Hancock, "Line Pattern Retrieval using Relational Histograms," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 21, no. 12, pp. 1363-1370, 1999.
  - [101] B. Jähne, *Digital Image Processing*, 5th Edition, Berlin Heidelberg New York: Springer-Verlag, 2002.
  - [102] A.K. Jain, *Fundamentals of Digital Image Processing*, Englewood Cliffs, New Jersey: Prentice-Hall International, Inc., 1989.
  - [103] A.K. Jain and A. Vailaya, "Shape-based Retrieval: A Case Study with Trademark Image Databases," *Pattern Recognition*, vol. 31, no.9, pp. 1369-1390, 1998.
  - [104] B.K. Jang and R.T. Chin, "Analysis of Thinning Algorithms using Mathematical Morphology," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 12, no. 6, pp. 541-551, 1990.
  - [105] J.M. Jolion and A. Rosenfeld, *A Pyramid Framework for Early Vision*, Dordrecht, The Netherlands: Kluwer Academic Publishers, 1994.
  - [106] E.R. Kandel, J.H. Schwartz and T.M. Jessel, *Principle of Neural Science*, 4th Edition, New York: McGraw Hill, 2000.
  - [107] R.A. Katz and S.M. Pizer, "Untangling the Blum Medial Axis Transform," *International Journal of Computer Vision*, vol. 55, no. 2/3, pp. 139-153, 2003.
  - [108] H. Kauppinen, T. Seppänen and M. Pietikäinen, "An Experimental Comparison of Autoregressive and Fourier-based Descriptors in 2D Shape Classification," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 17, no. 2, pp. 201-207, 1995.

- [109] H.K. Kim and J.D. Kim, "Region-based Shape Descriptor Invariant to Rotation, Scale and Translation" *Signal Processing: ImageCommunication*, vol. 16, pp. 87-93, 2000.
- [110] R. Kimmel, D. Shaked and N. Kiryati, "Skeletonization via Distance Maps and Level Sets," *Computer Vision and Image Understanding*, vol. 62, no. 3, pp. 382-391, 1995.
- [111] B.B. Kimia, A. Tannenbaum and S.W. Zucker, "Shape, Shocks and Deformation, I: The Components of 2-D Shape and the Reaction-Diffusion Space," *International Journal of Computer Vision*, vol. 15, pp. 189-224, 1995.
- [112] B.B. Kimia, J. Chan, D. Bertrand, S. Coe, Z. Roadhouse and H. Tek, "A Shock-based Approach for Indexing Image Databases using Shapes," *SPIE: Multimedia Storage and Archiving Systems II*, vol. 3229, pp. 288-302, 1997.
- [113] B.B. Kimia, "On the Role of Medial Geometry in Human Vision," to appear in *Journal of Physiology*, currently available online at www.lems.brown.edu/vision/publications/Kimia's_Publication/Journal/journal.s.htm.
- [114] B.B. Kimia, "Silhouette Databases Downloads", Laboratory for Engineering Man/Machine Systems (LEMS), Brown University, currently available at <http://www.lems.brown.edu/vision/software/index.htm>, 2005.
- [115] P. Klein, T. Sebastian and B. Kimia, "Shape Matching using Edit-Distance: An Implementation," in the *Proceedings of the ACM-SIAM Symposium for Discrete Algorithms (SODA'01)*, pp. 781-790, 2001.
- [116] J.J. Koenderink and A.J. van Doorn, "Visual Detection of Spatial Contrast; Influence of Location in the Visual Field, Target Extent and Illuminance Level," *Biological Cybernetics*, vol. 30, pp. 157-167, 1978.
- [117] J.J. Koenderink and A.J. van Doorn, "The Shape of Smooth Objects and the Way Contours End," *Perception*, Vol. 11, pp. 129-137, 1981.
- [118] J.J. Koenderink and A.J. van Doorn, "Representation of Local Geometry in the Visual System," *Biological Cybernetics*, vol. 55, pp. 367-375, 1987.
- [119] I. Kovacs, A. Feher and B. Julesz, "Medial-point description of shape: a representation for action coding and its psychophysical correlates," *Vision Research*, vol. 38, pp. 2323-2333, 1998.
- [120] R. Lakämper and F. Seytter, "Manipulation objektbasiert codierter Bilder als Anwendungsbeispiel neuer Videostandards," in the *Proc. of 19th DAGM Conf. on Pattern Recognition*, pp. 427-434, 1997.
- [121] L. Lam, S.W. Lee and C.Y. Suen, "Thinning Methodologies – A Comprehensive Survey," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 14, no. 9, pp. 869-885, 1992.
- [122] L. Lam and C.Y. Suen, "Automatic Evaluation of Skeleton Shapes," in the *Proceedings of the 11th International Conference on Pattern Recognition*, vol. 2, pp. 342-345, 1992.
- [123] L. Lam and C.Y. Suen, "Evaluation of Parallel Thinning Algorithms for Character Recognition," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 17, no. 9, pp. 914-919, 1995.

- [124] L.J. Latecki and R. Lakämper, "Convexity Rule for Shape Decomposition based on Discrete Contour Evolution," *Computer Vision and Image Understanding*, vol. 73, no. 3, pp. 441-451, 1999.
- [125] T. Lee, "Neurophysiological evidence for image segmentation and medial axis computation in primate V1," in the *Proceedings of the 4th Annual Computational Neuroscience Conference*, Kluwer Academic Publishing, The Netherlands 1996.
- [126] T.S. Lee, D. Mumford, R. Romero and V.A. Lamme, "The role of Primary Visual Cortex in Higher Level Vision," *Vision Research*, vol. 38, pp. 2429-2454, 1998.
- [127] F. Leymarie and M.D. Levine, "Simulating the Grassfire Transform using and Active Contour Model," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 14, no. 1, pp. 56-75, 1992.
- [128] F. F. Leymarie, "Three-Dimensional Shape Representation via Shock Flows," *Ph.D Thesis*, Brown University, May 2003.
- [129] M. Leyton, "A process-Grammar for Shape," *Artificial Intelligence*, vol. 32, no. 2, pp. 213-247, 1988.
- [130] LEMS, "3D Shock Transitions in Carpal Bones Dataset," Brown University at http://www.lems.brown.edu/vision/researchAreas/shock3d_transitions/carpal_bone.html , 2005.
- [131] X. Li and Z. Zhiying, "Group Direction Difference Chain Codes for the Representation of the Border," in the *Proceedings of SPIE Conference on Digital and Optical Shape, Representation and Pattern Recognition*, pp. 372-376, 1988.
- [132] J.S. Lim, *Two-dimensional Signal and Image Processing*, New Jersey: Prentice-Hall, 1990.
- [133] T. Lindeberg and L. Florack, "Foveal Scale-Space and Linear Increase of Receptive Field Size as a Function of Eccentricity," Technical Report ISRN KTH NA/P-94/27-SE, Department of Numerical Analysis and Computing Science, *Royal Institute of Technology*, August, 1994.
- [134] T. Lindeberg, "Feature Detection with Automatic Scale Selection," *International Journal of Computer Vision*, vol. 30, no. 2, pp. 77-116, 1998.
- [135] T.L. Liu and D. Geiger, "Representation and Self-Similarity of Shapes," in the *Proceedings of the International Conference on Computer Vision (ICCV'98)*, pp. 1129-1138, 1998.
- [136] T.L. Liu and D. Geiger, "Approximate Tree Matching and Shape Similarity," in the *Proceedings of the International Conference on Computer Vision (ICCV'99)*, pp. 456-462, 1999.
- [137] M.S. Livingstone and D.H. Hubel, "Psychophysical Evidence for Separate Channels for the Perception of Form, Color, Movement and Depth," *The Journal of Neuroscience*, vol. 7, no. 11, pp. 3416-3468, 1987.
- [138] R.C. Lo and W.H. Tsai, "Perspective-Transformation-Invariant Generalized Hough Transform for Perspective Planar Shape Detection and Matching," *Pattern Recognition*, vol. 30, no. 3, pp. 383-396, 1997.

- [139] S. Loncaric, "A Survey of Shape Analysis Techniques," *Pattern Recognition*, vol. 31, no. 8, pp. 983-1001, 1998.
- [140] D.G. Lowe, "Object Recognition from Local Scale-Invariant Features," in the *Proceedings of the 7th IEEE International Conference on Computer Vision (ICCV'99)*, vol. 2, pp. 1150-1157, 1999.
- [141] H.E. Lu and S.P. Wang, "A Comment on 'A Fast Parallel Algorithm for Thinning Digital Patterns'," *Communications of the ACM*, vol. 29, no. 3, pp. 239-242, 1984.
- [142] G. Lu and A. Sajjanhar, "Region-based Representation and Similarity Measure Suitable for Content-based Image Retrieval," *Multimedia Systems*, vol. 7, pp. 164-174, 1999.
- [143] D. Marr and K.H. Nishihara, "Representation and Recognition of the Spatial Organization of Three Dimensional Structure," *Proceedings of the Royal Society of London, Series B*, vol. 200, pp. 269-294, 1978.
- [144] W.H. Marshall and S.A. Talbot, "Recent Evidence for Neural Mechanism in Vision leading to a General Theory of Sensory Acuity," in H. Kluver, Ed., *Visual Mechanism*, Lancaster, PA: Cattell, pp. 117-164, 1942.
- [145] Matlab User Guide – Image Processing Toolbox (version 2), *The MathsWork Inc.*, 1999.
- [146] D.S. McKenzie and S.R. Prothero, "Curve Description using the Inverse Hough Transform," *Pattern Recognition*, vol. 23, no. 3-4, pp. 283-290, 1990.
- [147] R. Mehrotra and J.E. Gary, "Similar-Shape Retrieval in Shape Data Managment," *IEEE Computer*, vol. 28, no. 9, pp. 57-62, 1995.
- [148] K. Mikolajczyk and C. Schmid, "Scale and Affine Invariant Interest Point Detector," *International Journal of Computer Vision*, vol. 60, no. 1, pp. 64-86, 2004.
- [149] E. Milios and E. Petrakis, "Shape Retrieval base on Dynamic Programming," *IEEE Trans. on Image Processing*, vol. 9, no. 1, pp. 141-146, 2000.
- [150] F. Mokhtarian and A.K. Mackworth, "A Theory of Multiscale, Curvature-based Shape Representation for Planar Curves," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 14, no. 8, pp. 57-62, 1995.
- [151] U. Montanari, "a Method for Obtaining Skeletons using a Quasi-Euclidean Distance," *Journal of the Association for Computing Machinery*, vol. 15, no. 4, pp. 600-624, 1968.
- [152] B.S. Morse, S.M. Pizer, D.T. Puff and C. Gu, "Zoom-invariant Vision of Figural Shape: Effects on Cores of Image Disturbances," *Computer Vision and Image Understanding*, vol. 69, no. 1, pp. 72-86, 1998.
- [153] J.A. Movshon, E.H. Adelson, M.S. Gizzi and W.T. Newsome, "The Analysis of Moving Visual Pattern," in C. Chagas, R. Gattass and C. Gross, Eds., *Pattern Recognition Mechanism*, New York: Springer-Verlag, 1986.
- [154] C.W. Niblack, P.B. Gibbons and D.W. Capson, "Generating Skeletons and Centrelines from the Distance Transform," *Computer Vision, Graphics, and Image Processing: Graphical Models and Image Processing*, vol. 54, no. 5, pp. 420-437, 1992.

- [155] W. Niblack, R. Barber, W. Equitz, M.D. Flickner, E.H. Glasman, D. Petkovic, P. Yanker, C. Faloutsos, G. Taubin and Y. Heights, "Querying Images by Content, using Color, Texture and Shape," in the *Proceedings of SPIE Conference on Storage and Retrieval for Image and Video Databases*, vol. 1908, pp. 173-187, 1993.
- [156] R.L. Ogniewicz and O. Kübler, "Hierarchic Voronoi Skeletons," *Pattern Recognition*, vol. 28, no. 3, pp. 342-359, 1995.
- [157] G. Orwell, *Animal Farm*, Oxford: Heinemann New Windmills, 1994.
- [158] P.J. van Otterloo, *A Contour-Oriented Approach to Shape Analysis*, U.K.: Prentice Hall International Ltd., 1991.
- [159] S.E. Palmer, *Vision Science, Photons to Phenomenology*, Cambridge, Massachusetts: The MIT Press, 1999.
- [160] M.A. Paradiso and K. Nakayama, "Brightness Perception and Filling In," *Vision Research*, vol. 31, pp. 1221-1236, 1991.
- [161] M.A. Paradiso and S. Hahn, "Filling-in Percepts Produced by Luminance Modulation," *Vision Research*, vol. 36, pp. 2657-2663, 1996.
- [162] E. J. Pauwels, P. Fiddelaers, and F. Mindru, "Fully unsupervised clustering using center-surround receptive fields with applications to colour segmentation," in the *Proceedings of 7th Int. Conf. on Computer Analysis of Images and Patterns*, pp. 17-24, 1997.
- [163] M. Pelillo, K. Siddiqi and S. Zucker, "Matching Hierarchical Structures using Association Graphs," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, no. 11, pp. 1105-1120, 1999.
- [164] A. Pentland, "Recognition by Parts," in the *Proceedings of the First International Conference on Computer Vision (ICCV'87)*, pp. 612-620, 1987.
- [165] A. Pentland, R.W. Picard and S. Sclaroff, "Photobook: Content-based Manipulation of Image Databases," *International Journal of Computer Vision*, vol. 18, no. 3, pp. 233-254, 1996.
- [166] M.A. Peterson and B.S. Gibson, "Shape Recognition Inputs to Figure Ground Organization in Three Dimensional Displays," *Cognitive Psychology*, vol. 25, pp. 383-429, 1993.
- [167] E. Petrakis and E. Milios, "Shape Retrieval based on Dynamic Programming," *IEEE Trans. on Image Processing*, vol. 9, pp. 141-147, 2000.
- [168] S. Pinker, "Visual Cognition: An Introduction," *Cognition*, vol. 18, pp. 1-63, 1984.
- [169] S.M. Pizer, D. Eberly, D.S. Fritsch and B.S. Morse, "Zoom-invariant Vision of Figural Shape: The Mathematics of Cores," *Computer Vision and Image Understanding*, vol. 69, no. 1, pp. 55-71, 1998.
- [170] S.M. Pizer, K. Siddiqi, G. Székely, J. N. Damon and S.W. Zucker, "Multiscale Medial Loci and Their Properties," *International Journal of Computer Vision*, vol. 5, no.2-3, pp. 155-179, 2003.
- [171] S. Polyak, *The Vertebrate Visual System*, Chicago, Illinois: University of Chicago Press, 1957.

- [172] J. Ponce, "On Characterising ribbons and finding skewed symmetries," *Computer Vision, Graphics, and Image Processing*, vol. 52, no. 3, pp. 328-340, 1990.
- [173] L. Prasad and S.S. Iyengar, *Wavelet Analysis with Applications to Image Processing*, Orlando, Florida: CRC Press, 1997.
- [174] J. Robson, "Frequency Domain Visual Processing," in O.J. Braddick and A.C. Sleight, Ed., *Physical and Biological Processing of Images*, Springer-Verlag, 1983.
- [175] H. Rom and G. Medioni, "Hierarchical Decomposition and Axial Shape Description," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 15, no. 10, pp. 973-981, 1993.
- [176] A. Rosenfeld, *Multiresolution Image Processing & Analysis*, Berlin Heidelberg New York: Springer-Verlag, 1984.
- [177] P.L. Rosin, "Shape Partitioning by Convexity," *IEEE Trans. on Systems, Man and Cybernetics – Part A: Systems and Humans*, vol. 30, no. 2, pp. 202-210, 2000.
- [178] Y. Rubner, C. Tomasi and L.J. Guibas, "The Earth Mover's Distance as a Metric for Image Retrieval," *International Journal of Computer Vision*, vol. 40, no. 2, pp. 99-121, 2000.
- [179] Y. Rui, A.C. She and T.S. Huang, "Modified Fourier Descriptor for Shape Representation – A Practical Approach " in the *Proceedings of the 1st International Workshop on Image Database and Multi Media Search*, 1996.
- [180] Y. Rui, T.S. Huang, M. Ortega and S. Mehrotra "Relevance Feedback: A Power Tool for Interactive Content-Based Image Retrieval," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 8, no. 5, pp. 644-655, 1998.
- [181] H. Samet, "Region Representation: Quadrees from Boundary Codes," *Communications of ACM*, vol. 23, pp. 163-170, 1980.
- [182] G. Sapiro and A. Tannenbaum, "Area and Length Preserving Geometric Invariant Scale-Spaces," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 17, no. 1, pp.67-72, 1995.
- [183] D. Scharstein, "Matching Images by Comparing their Gradient Fields," in the *Proceedings of the 12th International Conference on Pattern Recognition*, vol. 1, pp. 572-575, 1994.
- [184] B. Schiele and J.L. Crowley, "Object Recognition using Multidimensional Receptive Field Histograms," in the *Proceedings of the 4th European Conference on Computer Vision (ECCV'96)*, vol. 1, pp. 610-619, 1996.
- [185] B. Schiele, "Object Recognition using Multidimensional Receptive Field Histograms," *Ph.D Thesis*, I.N.P.Grenoble, English Translation, 1997.
- [186] B. Schiele and J.L. Crowley, "Recognition without Correspondence using Multidimensional Receptive Field Histograms," *International Journal of Computer Vision*, vol. 36, no. 1, pp. 31-50, 2000.
- [187] T.B. Sebastian, P.N. Klein and B.B. Kimia, "Recognition of Shapes by Editing Shock Graphs," in the *Proceedings of the International Conference on Computer Vision (ICCV'01)*, pp. 755-762, 2001.

- [188] T.B. Sebastian and B.B. Kimia, "Curves versus Skeletons in Object Recognition," in the *Proceedings of the International Conference on Image Processing (ICIP'01)*, vol. 3, pp. 22-25, 2001.
- [189] T.B. Sebastian, P.N. Klein and B.B. Kimia, "Shock-based Indexing into Large Shape Databases," in the *Proceedings of the European Conference on Computer Vision (ECCV'02)*, pp. 731-746, 2002.
- [190] T.B. Sebastian, P.N. Klein and B.B. Kimia, "On Aligning Curves," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 25, no. 1, pp. 116-124, 2003.
- [191] T.B. Sebastian, P.N. Klein and B.B. Kimia, "Recognition of Shapes by Editing Shock Graphs," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 26, no. 5, pp. 550-571, 2004.
- [192] P. Seitz, "Using Local Orientation Information as Image Primitives for Robust Object Recognition," in the *Proceedings of SPIE*, vol. 1199, pp. 1630-1639, 1989.
- [193] D. Shaked and A.M. Bruckstein, "Pruning Medial Axes," *Computer Vision and Image Understanding*, vol. 69, no. 2, pp. 156-169, 1998.
- [194] D. Shaked and A.M. Bruckstein, "The Curve Axis," *Computer Vision and Image Understanding*, vol. 63, pp. 367-379, 1996.
- [195] D. Sharvit, J. Chan, H. Tek and B.B. Kimia, "Symmetry-based Indexing of Image Databases," *Journal of Visual Communication and Image Representation*, vol. 9, no. 4, pp. 366-380, 1998.
- [196] F.Y. Shih and C.C. Pu, "A Skeletonization Algorithm by Maxim Tracking on Euclidean Distance Transform," *Pattern Recognition*, vol. 28, no. 3, pp. 331-341, 1995.
- [197] H. Shroff and J. Ben-Arie, "Finding Shape Axes using Magnetic Field," *IEEE Trans. on Image Processing*, vol. 8, no. 10, pp. 1388-1394, 1999.
- [198] K.S. Siddiqi and B.B. Kimia, "Parts of Visual Form: Computational Aspects," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 17, pp. 239-251, 1995.
- [199] K.S. Siddiqi and B.B. Kimia, "A Shock Grammar for Recognition," in the *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR'96)*, pp. 506-513, 1996.
- [200] K.S. Siddiqi, B.B. Kimia, A. Tannenbaum, and S. W. Zucker, "Shapes, Shocks and Wiggles," *Image and Vision Computing*, vol. 17, no. 5-6, pp. 365-373, 1999.
- [201] K.S. Siddiqi, A. Shokoufandeh, S. Dickinson and S. Zucker, "Shock Graphs and Shape Matching," *International Journal of Computer Vision*, vol. 35, no. 1, pp. 13-32, 1999.
- [202] K.S. Siddiqi, S. Bouix, A. Tannenbaum, and S. W. Zucker, "Hamilton-Jacobi Skeletons," *International Journal of Computer Vision*, vol. 48, no. 3, pp. 215-231, 2002.

- [203] E. Simoncelli, "Steerable Pyramid Toolbox," available from Department of Computer and Information Science, Penn Engineering, University of Pennsylvania at <http://www.cis.upenn.edu/~eero/steerpyr.html>, 2003.
- [204] M. Singh and D.D. Hoffman, "Part Boundaries Alter the Perception of Transparency," *Psychological Science*, vol. 9, pp. 370-378, 1998.
- [205] M. Singh, G.D. Seyranian and D.D. Hoffman, "Parsing Silhouettes: The Short-cut Rule," *Perception and Psychophysics*, vol. 61, no.4, pp. 636-660, 1999.
- [206] T. Sødning and A.F. Smeaton, "Evaluating a Musical Information Retrieval System – TREC Style," in *the Papers presented at the Workshop on the Creation of Standardized Test Collections, Tasks, and Metrics for Music Information Retrieval (MIR) and Music Digital Library (MDL) Evaluation*, pp. 71-78, 2002.
- [207] L. Sola, "Biology 182 course at Glendale Community College, Glendale, Arizona, USA", *Web resource at* <http://www.gecko.maricopa.edu/~Isola/bio182/labreview/diatom.jpg>.
- [208] M. Sonka, V. Hlavac and R. Boyle, *Image Processing, Analysis, and Machine Vision*, 2nd Edition, Pacific Grove, CA: PWS Publishing Corporation, 1999.
- [209] M. Stricker and M. Orengo, "Similarity of Color Images," in the *Proceedings of SPIE Conference on Storage and Retrieval for Image and Video Database III*, vol. 2420, pp. 381-392, 1995.
- [210] M.J. Swain and D.H. Ballard, "Color Indexing," *International Journal of Computer Vision*, vol. 7, no. 1, pp. 11-32, 1991.
- [211] A. Taza and C. Suen, "Discrimination of Planar Shapes using Shape Matrices," *IEEE Trans. on Systems, Man and Cybernetics*, vol. 19, pp. 1281-1289, 1989.
- [212] M.R. Teague, "Image Analysis via the General Theory of Moments," *Journal of Optical Society of America*, vol. 70, no. 8, pp. 920-930, 1980.
- [213] C.H. Teh and R.T. Chin, "On Image Analysis by Methods of Moments," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 10, no. 4, pp. 496-513, 1988.
- [214] D.W. Thomson, "Morphology and Mathematics," *Trans. of the Royal Society of Edinburgh*, vol. L, part IV, no. 27, pp. 857-895, 1915.
- [215] T. Thompson, "Integrative Neuroscience course at University of Texas, Dallas", *Web resource at* http://www.utdallas.edu/~tres/integ/neuroII_read.html.
- [216] A. Torsello and E.R. Hancock, "A Skeletal Measure of 2D Shape Similarity," *Computer Vision and Image Understanding*, vol. 95, no. 1, pp. 1-29, 2004.
- [217] W.H. Tsai and S.S. Yu, "Attributed String Matching with Merging for Shape Recognition," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 7, no. 4, pp. 453-462, 1985.
- [218] J.K. Tsotsos, "How Does Human Vision Beat the Computational Complexity of Visual Perception," in Z. Pylyshyn, Ed., *Computational Processes in*

- Human Vision: An Interdisciplinary Perspective*, Norwood, New Jersey: Ablex Publishing Corporation, pp. 286-338, 1988.
- [219] F. Ullah, S. Kaneko and S. Igarashi, "Orientation Code Matching for Robust Object Search," *IEICE Transaction on Information and Systems*, vol. E84-D, no. 8, pp. 999-1006, 2001.
  - [220] F. Ullah and S. Kaneko, "Using Orientation Codes for Rotation-invariant Template Matching," *Pattern Recognition*, vol. 37, no. 2, pp. 201-209, 2004.
  - [221] R.C. Veltkamp, "Shape Matching: Similarity Measures and Algorithms," Technical Report UU-CS-2001-3, Department of Information and Computing Science, *Utrecht University*, Utrecht, The Netherlands, January, 2001.
  - [222] R.C. Veltkamp, "Shape Matching: Similarity Measures and Algorithms," in the *Proceedings of Shape Modelling International 2001*, pp. 188-197, 2001.
  - [223] O. Veselova, "Perceptually Based Learning of Shape Descriptions," *Master's Thesis*, Massachusetts Institute of Technology, Cambridge, MA, 2003.
  - [224] O. Veselova and R. Davis, "Perceptually Based Learning of Shape Descriptions," in the *Proceedings of the 19th National Conference on Artificial Intelligence (AAAI'04)*, 2004.
  - [225] P.S.P. Wang and Y.Y. Zhang, "A Fast and Flexible Thinning Algorithm," *IEEE Trans. on Computer*, vol. 38, no. 5, pp. 741-745, 1989.
  - [226] H. Wilson and J. Bergen, "A Four-Mechanism Model for Threshold Spatial Vision," *Vision Research*, vol. 19, pp. 19-32, 1979.
  - [227] Univ. of Wisconsin-Madison, Dept. of Physiology, "Neuroscience 524", *Web resource at* <http://www.physiology.wisc.edu/neuro524/highervisual02.htm>.
  - [228] R.A. Young, "The Gaussian Derivative Model for Spatial Vision: I. Retinal Mechanism," *Spatial Vision*, vol. 2, pp. 273-293, 1987.
  - [229] S. Zeki, *A Vision of the Brain*, Oxford: Blackwell Scientific Publication, 1993.
  - [230] T.Y. Zhang and C.Y. Suen, "A fast parallel algorithm for thinning digital patterns," *Communications of the ACM*, vol. 27, no. 3, pp. 236-239, 1984.
  - [231] D. Zhang and G. Lu, "Content-Based Shape Retrieval using Different Shape Descriptors: A Comparative Study," in the *Proceedings of IEEE International Conference on Multimedia and Expo (ICME'01)*, pp. 317-320, 2001.
  - [232] D. Zhang and G. Lu, "A Comparative Study on Shape Retrieval using Fourier Descriptors with Different Shape Signatures," in the *Proceedings of the 5th Asian Conference on Computer Vision (ACCV'02)*, pp. 646-651, 2002.
  - [233] D. Zhang and G. Lu, "A Comparative Study of Three Region Shape Descriptors," in the *Proceedings of the Conference on Digital Image Computing Techniques and Applications (DICTA'02)*, pp. 86-91, 2002.
  - [234] S.C. Zhu and A.L. Yuille, "FORMS: A Flexible Object Recognition and Modeling System," *International Journal of Computer Vision*, vol. 20, no. 3, pp. 187-212, 1996.

Appendices

Appendix A

Computational Cost – Pyramidal Framework

A.1 Cost of Building the Gaussian Pyramid

Given the cost of the *REDUCE* operation c_G as defined in (2.6), the total cost required to build a Gaussian pyramid can be found by determining the number of times *REDUCE* is applied and this in turn, is determined by K_G , the number of new image pixels created from the original image $I(x,y)$ of size $R \times R$ pixels, as shown in Fig A.1.

Figure A.1 – The number of new image pixel created during the construction of the Gaussian pyramid.

The value of K_G is given by

$$K_G = \left(\frac{R}{2}\right)^2 \left[\sum_{i=0}^{\log_2 R - 1} \left(\frac{1}{2}\right)^{2i} \right] = \left(\frac{R}{2}\right)^2 S_i \quad (\text{A.1})$$

where S_i is a geometric series of even powers that contain $\log_2 R$ items (see Fig. A.1). The series S_i can be described using as a sum of two simple geometric series given by

$$S_i = \frac{1}{2} \left[\sum_{i=0}^{2 \log_2 R - 1} \left(\frac{1}{2}\right)^i + \sum_{i=0}^{2 \log_2 R - 1} \left(-\frac{1}{2}\right)^i \right] \quad (\text{A.2})$$

It is well known that a geometric progression with $2\log_2 R$ terms such as the two given in (A.2) can be expressed in a closed form [33]. This gives

$$S_i = \frac{1}{2} \left[\frac{1 - (1/2)^{2\log_2 R}}{1 - (1/2)} + \frac{1 - (-1/2)^{2\log_2 R}}{1 - (-1/2)} \right] \quad (\text{A.3})$$

which then simplifies to

$$S_i = \frac{4}{3} \left(1 - \frac{1}{R^2} \right) \quad (\text{A.4})$$

substituting S_i in (A.4) into (A.1), gives a expression of K_G in terms of the original size of the image

$$K_G = \frac{R^2 - 1}{3} \quad (\text{A.5})$$

Given the cost of the *REDUCE* operation c_G as defined in (2.6), the total cost incurred in constructing the Gaussian pyramid is given by

$$\text{cost}_G = K_G c_G = \frac{R^2 - 1}{3} c_G \quad (\text{A.6})$$

A.2 Cost of Building the Gradient Pyramid

The cost of building the Gradient pyramid is dependent on the number of pixels in the Gaussian pyramid since the two Gaussian derivative kernels in (2.4) are applied to every pixel, including that of the original input image at level 0. The pixels count K_H of a Gaussian pyramid constructed from image $I(x, y)$ of size $R \times R$ pixels is given by

$$K_H = R^2 \left[\sum_{j=0}^{\log_2 R} \left(\frac{1}{2} \right)^{2j} \right] = R^2 S_j \quad (\text{A.7})$$

where S_j is a geometric series of even powers of $\log_2 R + 1$ items (see Fig. A.2). The series S_j can be described using as a sum of two simple geometric series given by

$$S_j = \frac{1}{2} \left[\sum_{j=0}^{2\log_2 R} \left(\frac{1}{2} \right)^j + \sum_{j=0}^{2\log_2 R} \left(-\frac{1}{2} \right)^j \right] \quad (\text{A.8})$$

The geometric series with $2\log_2 R + 1$ terms can be rewritten as

$$S_j = \frac{1}{2} \left[\frac{1 - (1/2)^{2 \log_2 R + 1}}{1 - (1/2)} + \frac{1 - (-1/2)^{2 \log_2 R + 1} + 1}{1 - (-1/2)} \right] \quad (\text{A.9})$$

which then simplifies to

$$S_j = \frac{4}{3} - \frac{1}{3R^2} \quad (\text{A.10})$$

substituting S_j in (A.10) into (A.7), gives an expression of K_H in terms of the original size of the image

$$K_H = \frac{4R^2 - 1}{3} \quad (\text{A.11})$$

If the cost of each convolution operation is c_H , then the total cost incurred in constructing the vectorial Gradient pyramid is given by

$$\text{cost}_H = 2K_H c_H = 2 \left[\frac{4R^2 - 1}{3} \right] c_G \quad (\text{A.12})$$

Figure A.2 – The number of convolutions required to build a Gradient pyramid, which consist of the two scalar pyramids H^x and H^y .

Appendix B

Additional Experimental Results for Chapter 4

B.1 Geometric Invariance

The MGVF-based orientation histogram can be made invariant to rotation and mirror reflection by using the magnitude of the Fourier transformed vector of the orientation histogram. This is illustrated by the matching results shown in Fig. B.1, where every query image was correctly matched to images of its respective shape type in the best three matches.

	Hare				Rabbit				Turtle			
	×1 Original	×½ 90°	×2 -30°	×1 mirror	×1 Original	×½ 90°	×2 -30°	×1 mirror	×1 Original	×½ 90°	×2 -30°	×1 mirror
Query												
Match #1	0	13	4	0	1	10	6	1	1	14	6	1
#2	4	13	4	4	6	10	6	6	6	14	6	6
#3	13	13	13	13	10	10	10	10	14	16	16	14

Figure B.1 - Results demonstrating scaling, rotation and mirror reflection invariant shape matching using the MGVF-based FTN feature vectors. The L_2 -distances (top-right corners, $\times 100$ and rounded to the nearest integer) describe the dissimilarity of each query shape with the three best matches from the remaining pool of test shapes. Parameters used in this experiment were $n = 36$, $\alpha = 0.8$ and $\sigma_H = 1.5$.

B.2 Robustness to Boundary Texture

The detailed results for the boundary texture experiment for the various descriptors are presented here.

FTN (d_{l_2})

Texture	Hare			Rabbit			Pear			Candle		
	Original	Wavy	Spiky	Original	Wavy	Spiky	Original	Wavy	Spiky	Original	Wavy	Spiky
Query												
Match #1	14	8	8	8	13	8	10	10	12	15	7	7
#2	14	10	11	10	15	11	12	14	14	17	15	17

Figure B.2 - Matching result for boundary-textured shapes obtained using the MGVF-based descriptor with FTN with L_2 -distance. The dissimilarity measures shown in the top-right corner are rounded to the nearest integer, with scaling of $\times 100$. Parameters used were $\alpha = 0.4$, $\sigma_H = 1.5$ and $n = 36$ bins. A cross (✗) in the bottom-left corner indicates a misclassification.

SC ($N = 100$)

Texture	Hare			Rabbit			Pear			Candle		
	Original	Wavy	Spiky	Original	Wavy	Spiky	Original	Wavy	Spiky	Original	Wavy	Spiky
Query												
Match #1	18	18	18	16	16	18	16	16	21	17	17	21
#2	18	18	22	18	18	21	26	26	21	26	26	22

Figure B.3 - Matching results for various boundary textures using the shape context with parameters settings similar to that in [17], namely ($N = 100$, 12 equally-spaced angle bins, 5 log-spaced radius bins and $\beta = 0$). The matching costs shown are $\times 100$ and rounded to the nearest integer..

Shape descriptors like the Zernike moments descriptor (ZMD) can have their order n truncated to reduce their sensitivity to boundary noise. This tradeoff does not always produce the desired results, especially if the variations between the different shapes are not very significant. This can be observed from the matching results shown in Fig. B.4. Trading off shape description details by using lower order moments not only reduces the influence of boundary texture but also attenuates the subtle shape variations required to discriminate between the various base shapes.

ZMD ($n = 10$)

Texture	Hare			Rabbit			Pear			Candle		
	Original	Wavy	Spiky	Original	Wavy	Spiky	Original	Wavy	Spiky	Original	Wavy	Spiky
Query												
Match #1	4	4	14	5	5	14	12	12	21	10	10	13
#2	12	11	17	13	11	19	21	27	22	13	13	13

(a)

ZMD ($n = 5$)

Texture	Hare			Rabbit			Pear			Candle		
	Original	Wavy	Spiky	Original	Wavy	Spiky	Original	Wavy	Spiky	Original	Wavy	Spiky
Query												
Match #1	1	1	7	1	1	8	3	3	15	6	6	8
#2	8	7	8	6	7	13	18	20	17	6	7	9

(b)

Figure B.4 - Results using the ZMD with order up to (a) $n = 10$ and (b) $n = 5$. Centroid centering and scaling to unit circle (see Fig. 4.4) was applied to shape before computing the ZMD. Shape dissimilarity measures shown are $\times 100$ and computed using L_2 -distance. The Matlab code used to compute the ZMD was modified from that of Chang at [45].

In the case of the Fourier descriptor (FD), a similar reduction in sensitivity to boundary perturbations can be attained by using only the lower frequency components of the feature vector. A reduction in the number of misclassifications from 15 to 9 was observed when only the lowest 6 FD components were used as the descriptor instead of the entire feature vector.

FD ($N = 64$)

Texture	Hare			Rabbit			Pear			Candle		
	Original	Wavy	Spiky	Original	Wavy	Spiky	Original	Wavy	Spiky	Original	Wavy	Spiky
Query												
Match #1	4	4	8	4	4	8	4	4	11	4	4	11
#2	6	5	10	5	6	9	15	16	12	14	13	13

(a)

FD ($N = 64$) – lowest 6 components

Texture	Hare			Rabbit			Pear			Candle		
	Original	Wavy	Spiky	Original	Wavy	Spiky	Original	Wavy	Spiky	Original	Wavy	Spiky
Query												
Match #1	2	2	4	3	3	3	5	5	13	9	9	13
#2	7	5	4	4	3	4	34	39	15	20	14	14

(b)

Figure B.5- Results using the radii-based Fourier descriptor ($N=64$) with the L_2 -distance measure. (a) All 64 components used and (b) only the lowest 6 components used. The matching costs shown are $\times 100$ and rounded to the nearest integer.

B.3 Silhouette Classification

The detailed shape classification results for the various MGVF-based shape descriptors summarised in Table 4.1 are shown in Fig. B.6. The parameters used to generate these results were $\alpha=0.8$, $\sigma_H=1.5$ and $n=36$.

Figure B.6 – Shape classification results for (a) AHN with d_χ and d_{L2} , (b) GGON with d_γ , (c) GGON with d_θ , (d) MGON with d_γ , (e) MGON with d_θ and (f) FTN with d_{L2} .

Appendix C

Query Time Reduction vs. Retrieval Performance

The computation of the total global distance error ξ_{gs} given in (6.19) is very costly as it involves $2 \times NA \times NB$ arctangent computations, where NA and NB are the total number of secondary skeletal segments in shapes A and B respectively. One way to reduce the computational requirements is to avoid doing a full search through all the possible K_w skeletal pairs. Since the set of skeletal pairs $(SS_{w,k}^A, SS_{k \min, k}^B)_{k=1}^{K_w}$ can be ordered based on increasing matching cost, the likelihood of finding the reference skeletal pair is higher at the starting end of the ordered list where the matching cost between the skeletal pairs is lower. A preset limit of $Kmax$ could be set such that the search begins at the start of the ordered list and stops when the $Kmax$ limit or K_w is reached, whichever is the smaller for the particular query shape A . Fig. C.1 shows the variable query time for querying all 99 shapes in the 99-shapes database when the value of $Kmax$ is changed. By assuming the lowest matching cost pair is the reference skeletal pair (i.e. $Kmax=1$), the total query time can be reduced by more than a half compared to a more rigorous search through all K_w skeletal pairs. Such figures confirm the claim that a large fraction of the total computational cost for matching two shapes is devoted to the task of determining the reference source nodes.

Figure C.1 – The reduction in MGVF-PBD query time for querying all 99 shapes in the 99-shapes database (see Appendix E) using different values of $Kmax$.

The next obvious question is whether the suggested sub-optimal search does affect the retrieval performance? Fig. C.2a shows that as K_{max} moves from 1 to 10, the precision-recall graph clearly shows a drastic improvement in retrieval performance. This suggests that the appropriate reference skeletal pair for computing global distance is not necessarily the skeletal pair with the lowest matching cost. However, Fig. C.2b shows that as K_{max} increases beyond 15, not much further improvement in retrieval performance can be obtained. This does suggest that for databases like the 99-shapes database, the appropriate skeletal pair from which the reference source nodes can be determined can generally be found within the first 15 skeletal pairs if the set of all skeletal pairs are ordered with an increasing matching cost. Since computational time is not a major consideration in the experiments presented in section 6.6, a value of $K_{max} = 100$ was used. Since all the shapes have less than 100 secondary skeletal segments, a full search was effectively employed.

Figure C.2 – The retrieval performance using different values of K_{max} . (a) Significant improvement in performance with variation at low K_{max} values. (b) Insignificant performance improvements with values of $K_{max} > 15$.

Appendix D

MGVF-PBD Standard Parameter Settings

The parameter settings listed in this appendix were used to generate the MGVF, extract the MGVF skeleton, decompose the skeleton into skeletal-based parts and derive the part-based descriptors. These standard settings were used in all the experiments described in Chapter 5 unless explicitly mentioned otherwise.

Parameters for generating MGVF

No.	Description	Notation	Equation Number	Value
1.	Width of Gaussian generating kernel for generating Gaussian pyramid	k	(2.1)	2
2.	Standard deviation of Gaussian derivative kernel for generating Gradient pyramid	σ_H	(2.4)	1.5
3.	Smoothing parameter	α	(2.5)	0.9

Parameters for extracting MGVF skeleton

No.	Description	Notation	Equation Number	Value
4.	Standard deviation of Gaussian convolution kernel to compute local directional disparity	σ_D	(3.7)	1.5
5.	Local ridge detection threshold	N_D	Sec. 3.3.2	16
6.	Standard deviation of 1 st -order Gaussian derivative convolution kernel to compute derivative of disparity map	σ_V	(3.11)	0.5
7.	Standard deviation of Gaussian convolution kernel to compute local directional disparity in derivative disparity map	σ_E	(3.12)	0.5
8.	Standard deviation of Gaussian convolution kernel used to compute ‘ridgeness’ and ‘valleyiness’ in derivative disparity map	σ_R	(3.13)	1.0
9.	Connectivity factor	λ	(3.17)	1.0

Parameters for segmenting MGVF skeletal segments

No.	Description	Notation	Equation Number	Value
10.	Standard deviation of 1 st -order Gaussian derivative kernel used in the zero-crossing detector for disparity minima/maxima	σ_z	(5.4)	5.0
11.	Standard deviation of 1 st -order Gaussian derivative kernel for computing skeletal curvature	σ_c	(5.7)	1.5
12.	Standard deviation of 1 st -order Gaussian derivative kernel used in the zero-crossing detector for curvature minima/maxima	σ_z	(5.8)	0.5
13.	Skeletal curvature maxima/minima threshold	τ_c	(5.9)	0.1

Parameters for merging MGVF skeletal segments

No.	Description	Notation	Equation Number	Value
14.	Number of skeletal points used to estimate local orientation	N_{CO}	Fig. 5.26	4
15.	Number of skeletal points used to estimate local mean disparity	N_{ED}	(5.16)	10
16.	Visual continuity emphasis parameter	γ	(5.18)	0.5
17.	Angular threshold for merging	θ_{merge}	(5.9)	80°

Parameters for estimating MGVF part-based descriptor features and matching

No.	Description	Notation	Equation Number	Value
18.	Part-based orientation histogram bin size	n	Sec. 5.5.2	4
19.	Disparity fraction parameter when terminus is not touching boundary	ρ	(5.28) (5.30)	0.1
20.	Disparity fraction parameter when terminus is touching boundary	ρ	(5.28) (5.30)	0.3
21.	Polynomial order for skeletal curvature	q	(5.32)	6
22.	Polynomial order for limb width profile	r	(5.34)	6
23.	Segment histogram weight	η_1	(6.1)	1.5
24.	Source histogram weight	η_2	(6.1)	1.0
25.	Sink histogram weight	η_3	(6.1)	1.0
26.	Part distance orientation histogram weight	β_h	(6.7)	1.5
27.	Part distance curvature error weight	β_c	(6.7)	0.7
28.	Part distance limb width error weight	β_r	(6.7)	1.2
29.	Global distance source node-based spatial distance weight	γ_1	(6.18)	0.25
30.	Global distance source node-based skeletal angle weight	γ_2	(6.18)	0.25
31.	Global distance source node-based link angle weight	γ_3	(6.18)	0.25
32.	Global distance centroid-based spatial distance weight	γ_4	(6.17)	0.1
33.	Search limit for finding reference source node pair	K_{max}	Sec. 6.2.3	100

Appendix E

Shape Databases used in the Experiments

The 99-shapes and 216-shapes databases were used by Sebastian *et al.* in [191] and were downloaded from [114]. The binary images of varying sizes were resized to a uniform size of 129×129 pixels. Fig. E.1 shows images from the 99-shapes database, which consists of 9 categories, each with 11 different shapes. Fig. E.2 shows images from the 216-shapes database, which consists of 18 categories, each with 12 different shapes. Fig. E.3 and Fig. E.4 show the corresponding MGVF skeletons extracted from each of the respective databases using the parameters given in Appendix D.

Figure E.1 – Binary images in the 99-shapes database

Figure E.2 – Binary images in the 216-shapes database

Figure E.3 – The extracted MGVF skeletons for shapes in the 99-shapes database

Figure E.4 – The extracted MGVF skeletons for shapes in the 216-shapes database

Appendix F

Complete Shape Database Retrieval Results

This appendix presents the complete retrieval results for the shape-based image database retrieval results obtained for the experiments on the 99-shapes database (see Fig. F.1) and 216-shapes (see Fig. F.2) database. These results were obtained using the MGVF-PBD parameter settings given in Appendix D.

Figure F.1 – MGVF-PBD retrieval results ordered according to best matching score for the 99-shapes database. Erroneous within-category matches are in pink ( ).

Figure F.2a – The first 12 categories of the MGVF-PBD retrieval results ordered according to best matching score for the 216-shapes database. Erroneous within-category matches are in pink (.

Figure F.2b – The remaining MGVSF-PBD retrieval results ordered according to best matching score for the 216-shapes database. Erroneous within-category matches are in pink ( )

Appendix G

Data Storage Requirement of the MGVF-PBD

This appendix presents a summary of the data storage requirement of the MGVF-PBD and discusses a possible strategy for reducing the storage requirement.

G.1 Storage Requirement of Part Descriptor

For purposes of computing the storage requirement in number of bytes, the following assumptions are made regarding the various data types: unsigned integers (uint) are 2 bytes and floats are 4 bytes. The storage requirement is listed in Table G.1 and based on the MGVF-PBD settings given in Appendix D, a total of 284 bytes are required for each part descriptor associated with secondary skeletal segments.

No.	Description	Data Type	Equation Number	Storage (bytes)
1.	Secondary skeleton number, SS_j	uint	(5.23)	2
2.	Merged skeleton number, SM_m	uint	(5.20)	2
3.	Skeletal segment end points, $p_j(1), p_j(N_j)$	uint		2×2
4.	Skeletal source node angle, $\theta_j(1)$	float		4
5.	Segment histogram, Hsg_j	uint	(5.26)	36×2
6.	Source node histogram, Hsr_j	uint	(5.27)	36×2
7.	Sink node histogram, Hsn_j	uint	(5.29)	36×2
8.	Curvature polynomial coefficients, $\{dQ_j^k\}_{k=1}^q$	float	(5.33)	6×4
9.	Limb width polynomial coefficients, $\{R_j^k\}_{k=1}^r$	float	(5.34)	6×4
10.	Disparity-based saliency, GD	float	(5.38)	4
11.	Length-based saliency, GL	uint	(5.39)	2

Table G.1 – Data storage requirement for each secondary skeletal segment descriptor

G.2 Storage Requirement of Shape Database

The storage requirement of the shape database will depend on the number of shapes it contains and the number of secondary skeletal segments within each shape. Fig. G.1 shows the characteristics of the part count within the shapes of 216-shapes database given in Appendix E. The average number of primary skeletal segments is 13.7 with a median of 9 and the average number of secondary skeletal segments is 30.9 with a median of 18.

Figure G.1 – Histograms depicting the distribution of (a) primary and (b) secondary skeletal segment counts for all the shapes in the 216-shapes database.

The current implementation stores all the secondary skeletal segments of each shape. This means the 216-shapes database would have a storage requirement of about 1895530 bytes, which is slightly less than 2MB. Since the secondary segments are derived from the various combinations of primary segments, it is theoretically possible to store only the primary skeletal segments and compute the secondary segments during the actual query. If this is done, the storage requirement falls to about 840413 bytes, or below 1 MB. The obvious side effect of this data storage reduction strategy is the increase in query time. The actual increase in time has not been experimentally quantified and requires further investigation. However, several possible descriptor accuracy compromises must also be evaluated. Firstly, the computation of the combined skeletal curvature and limb width profile polynomial coefficients from the separate sets of polynomial coefficients of two or more primary

skeletal segments may not be as accurate as those estimated directly from the skeletal points of the equivalent secondary segment. Secondly, new segment histograms have to be computed from those belonging to two or more primary skeletal segments. Some of these require their bin order to be reversed because of the change in skeletal traversal direction resulting from the combination of several primary segments and the change in source node position. The accuracy of the reversed order histogram is often compromised by bin quantisation bias, especially if the number of bins in the orientation histogram is small. Further research is required to explore the combined impact these errors have on the overall shape retrieval performance.

Glossary

Acronyms

(In order of appearance)

1. MGVF	Multiresolution gradient vector field
2. 2D or 1D	Two-dimensional or one-dimensional
3. RF	Receptive field
4. LGN	Lateral geniculate nucleus
5. MT	Middle temporal
6. IT	Inferotemporal
7. OCR	Optical character recognition
8. MAT	Medial axis transform
9. EFT	Electrostatic field theory
10. EDT	Euclidean distance transform
11. DUC	Direction-uphill climbing
12. CBIR	Content-based image retrieval
13. GHT	Generalised Hough Transform
14. GD	Grid-based descriptor
15. ZMD	Zernike moment descriptor
16. FD	Fourier descriptor
17. MGON	Moment-based global orientation normalization
18. GGON	Gradient-based global orientation normalization
19. FTN	Fourier transform normalization
20. AHN	Adaptive hybrid normalization
21. SC	Shape context
22. WBGM	Weighted bipartite graph matching
23. EVF	Enhanced vector field
24. MGVF-PBD	Multiresolution gradient vector field part-based descriptor

Symbols

Symbols introduced in chapter 2

1. $I(x, y)$ Binary input image
2. $G_l(x, y)$ Gaussian pyramid image at level l
3. $H_l(x, y)$ Vectorial gradient pyramid image at level l
4. σ_H Standard deviation (SD) of Gaussian derivative kernel for $G_l(x, y)$
5. $V_l(x, y)$ Gradient vector field pyramid image at level l
6. α Smoothing parameter for generating $V_l(x, y)$

Symbols introduced in chapter 3

7. $\Delta(\cdot)$ Local directional disparity operator
8. σ_D Standard deviation of Gaussian convolution kernel for $\Delta(\cdot)$
9. N_D Local ridge detection threshold
10. $D_l(x, y)$ Vector field disparity pyramid image at level l
11. $M(x, y)$ Disparity map
12. $M_V(x, y)$ Disparity gradient map
13. σ_V SD of 1st-order Gaussian derivative convolution kernel to compute derivative of disparity map
14. $M_E(x, y)$ Enhanced disparity gradient map
15. σ_E SD of Gaussian convolution kernel to compute local directional disparity in derivative disparity map
16. $M_L(x, y)$ Disparity landscape map
17. σ_R SD of Gaussian convolution kernel used to compute ‘ridgeness’ and ‘valleyiness’ in derivative disparity map
18. $M_R(x, y)$ Ridge-enhanced disparity landscape map
19. λ Connectivity factor

Symbols introduced in chapter 4

20. $F(u)$ Fourier descriptor
21. $\theta(x, y)$ Orientation of a gradient vector
22. d_{L_2} Euclidean distance
23. d_χ or χ^2 Chi-squared distance or dissimilarity measure
24. d_Q Quadratic form distance

25. Θ_M	Moment-based global orientation
26. m_{pq}	Central moment
27. J	Second-order moment of inertia tensor matrix
28. Θ_G	Gradient-based global orientation
29. L_G	Elongatedness measure

Symbols introduced in chapter 5

30. S_j	Continuous skeletal segment j
31. DS_j	First-order derivative of continuous skeletal segment j
32. σ_z	SD of 1 st -order Gaussian derivative kernel to use in the zero-crossing detector for disparity minima/maxima
33. Pd_j	Set of points of vector field disparity profile maxima and minima
34. Pe_j	Set of points of valid connectivity end points
35. Pc_j	Set of points of displaying high curvature maxima and minima
36. τ_c	Skeletal curvature maxima/minima threshold
37. N_{CO}	Number of skeletal points used to estimate local orientation a
38. N_{ED}	Number of skeletal points used to estimate local mean disparity
39. γ	Visual continuity emphasis parameter
40. θ_{merge}	Angular threshold for merging
41. $\{SP_k\}$	Set of primary skeletal segments
42. SM_m	Merged skeletal segment m
43. $\{SS_k\}$	Set of secondary skeletal segments
44. Hsg_j	Normalised segment histogram for skeletal segment j
45. Hsr_j	Normalised histogram at source node of skeletal segment j
46. Hsn_j	Normalised histogram at sink node of skeletal segment j
47. η_1	Segment histogram weight
48. η_2	Source histogram weight
49. η_3	Sink histogram weight
50. $\{dQ_j\}$	Set of polynomial coefficient describing curvature derivative along skeletal segment j
51. $\{R_j\}$	Set of polynomial coefficient describing the limb width profile along skeletal segment j
52. $GD(S_j)$	The normalized disparity-based saliency of skeletal segment j
53. $GL(S_j)$	The length-based saliency of skeletal segment j

54. $G(S_j)$	The overall (combined) saliency of skeletal part j
55. $d_H(S_i, S_j)$	Histogram error between skeletal segments i and j
56. $d_C(S_i, S_j)$	Curvature error between skeletal segments i and j
57. $d_R(S_i, S_j)$	Limb width error between skeletal segments i and j
58. β_h	Part distance orientation histogram weight
59. β_c	Part distance curvature error weight
60. β_r	Part distance limb width error weight
61. $g_{tc}(S_i^A, S_j^B)$	Global centroid-based spatial distance between segment i of shape A and segment j of shape B
62. $g_{ts}(S_i^A, S_j^B)$	Global source node-based spatial distance between segment i of shape A and segment j of shape B
63. $g_{as}(S_i^A, S_j^B)$	Normalised source node skeletal angle between segment i of shape A and segment j of shape B
64. $g_{al}(S_i^A, S_j^B)$	Normalised source link skeletal angle between segment i of shape A and segment j of shape B
65. $g_d(S_i^A, S_j^B)$	Global distance between segment i of shape A and segment j of shape B
66. γ_1	Global distance source node-based spatial distance weight
67. γ_2	Global distance source node-based skeletal angle weight
68. γ_3	Global distance source node-based link angle weight
69. γ_4	Global distance centroid-based spatial distance weight
70. K_{max}	Search limit for finding reference source node pair
71. $s_d(A, B)$	Shape distance between shape A and B
72. $OCM(i, j)$	Two-dimensional over-complete matrix
73. $CM(i, j)$	Cost matrix
74. $WM(i, j)$	Saliency-weighted cost matrix