University College London
Department of Computer Science

Calculating the Curvature Shape Characteristics of the Human Body from 3D Scanner Data

by: Ioannis Douros

a thesis submitted for the degree of
Doctor of Philosophy in Computer Science
University of London

June 2004

ABSTRACT

In the recent years, there have been significant advances in the development and manufacturing of 3D scanners capable of capturing detailed (external) images of whole human bodies. Such hardware offers the opportunity to collect information that could be used to describe, interpret and analyse the shape of the human body for a variety of applications where shape information plays a vital role (e.g. apparel sizing and customisation; medical research in fields such as nutrition, obesity/anorexia and perceptive psychology; ergonomics for vehicle and furniture design). However, the representations delivered by such hardware typically consist of unstructured or partially structured point clouds, whereas it would be desirable to have models that allow shape-related information to be more immediately accessible.

This thesis describes a method of extracting the differential geometry properties of the body surface from unorganized point cloud datasets. In effect, this is a way of constructing curvature maps that allows the detection on the surface of features that are *deformable* (such as ridges) rather than *reformable* under certain transformations. Such features could subsequently be used to interpret the topology of a human body and to enable classification according to its *shape*, rather than its *size* (as is currently the standard practice for many of the applications concerned).

The background, motivation and significance of this research are presented in chapter one. Chapter two is a literature review describing the previous and current attempts to model 3D objects in general and human bodies in particular, as well as the mathematical and technical issues associated with the modelling. Chapter three presents an overview of: the methodology employed throughout the research; the assumptions regarding the data to be processed; and the strategy for evaluating the results for each stage of the methodology.

Chapter four describes an algorithm (and some variations) for approximating the local surface geometry around a given point of the input data set by means of a least-squares minimization. The output of such an algorithm is a surface patch described in an analytic (implicit) form. This is necessary for the next step described below. The case is made for using implicit surfaces rather than more popular 3D surface representations such as parametric forms or height functions.

Chapter five describes the processing needed for calculating curvature-related characteristics for each point of the input surface. This utilises the implicit surface patches generated by the algorithm described in the previous chapter, and enables the construction of a "curvature map" of the original surface, which incorporates rich information such as the principal curvatures, shape indices and curvature directions.

Chapter six describes a family of algorithms for calculating features such as ridges and umbilic points on the surface from the curvature map, in a manner that bypasses the problem of separating a vector field (i.e. the principal curvature directions) across the entire surface of an object. An alternative approach, using the focal surface information, is also considered briefly in comparison.

The concluding chapter summarises the results from all steps of the processing and evaluates them in relation to the requirements set in chapter one. Directions for further research are also proposed.

Acknowledgements

I would like to thank my supervisor Professor Bernard Buxton for his help, support, enthusiasm and meticulous feedback, without which this work would be impossible to complete. I would also like to thank Professor Philip Treleaven for enabling me to do this work and supporting me throughout, and my assessor, Professor Simon Arridge for his detailed and constructive remarks that have been essential in determining the directions of the research (especially the comment that "real world data is not all that it is cracked up to be").

Special thanks are due to Dr. Laura Dekker, whose human body modelling work at UCL has been the inspiration and starting point for my research; to Dr. Ian Porteous, whose help and support provided valuable and essential insight into the world of 3D surfaces and their features; and to Hamamatsu Photonics for providing the hardware that made this work possible.

I gratefully acknowledge the help and expertise of many other people: Jeni Bougourd, Jon Wells, Avy Tahan, Tim Hutton, Conny Ruiz, Bernhard Spanlang, Tzvetomir Vassilev, João Oliveira, George Kartsounis, Kostas Leftheriotis, Nikolaos Sapidis, Hein Daanen and Yiorgos Chrysanthou.

Finally, I would like to thank Sue Baggett (for reasons that are too many to list), all my colleagues at UCL, my family and friends for their encouragement and support, and all the anonymous volunteers who offered to have their bodies scanned for this research.

TABLE OF CONTENTS

ABSTRACT *1*

Acknowledgements *2*

Index of Figures *6*

Index of Tables *10*

1 Introduction *11*

1.1 Background 11

1.2 Motivation 12

1.3 Significance of the Research 13

1.4 Hypothesis 16

1.5 Objectives 16

1.6 Definitions of Basic Concepts 17

1.7 Contributions 19

1.8 Structure of the Thesis 21

2 Existing Work *22*

2.1 Application Domains 22

2.2 Human Body Scanning Systems 25

2.3 Human Body Modelling Work 26

2.4 Surface Reconstruction and Parameterisation 27

2.5 Curvature Estimation 31

2.6 Local Surface Fitting 32

2.7 Feature Detection 33

2.8 Matching and Registration 34

2.9 Deformable Shape Modelling and Analysis 35

2.10 Multiple Levels of Detail 39

3 Design and Methodology *40*

3.1 Local Surface Reconstruction 40

3.2 Issues to be Addressed 42

3.3 Evaluation of the Results 49

4 Local Surface Reconstruction *55*

4.1 Approximation of the Local Surface Shape 55

4.2 Minimization Problem 62

4.3 Closest point selection 66

4.4 Results 69

4.5 Conclusions 81

5 Calculation of Curvature Characteristics *82*

5.1 Principal Curvatures and their Directions 82

5.2 Gaussian and Mean Curvature 89

5.3 Curvedness and Shape Index 91

5.4 Curvature Labelling and Tracking 93

5.5 Results 95

5.6 Conclusions 119

6 Feature Detection *121*

6.1 Umbilic and Minimal Points 121

6.2 Parabolic Lines 123

6.3 Ridges 125

6.4 Flexcords 130

6.5 An alternative approach: using the Focal Surface 131

6.6 Results 133

6.7 Conclusions 160

7 Conclusions and Future Work *161*

7.1 Overview of the Results 161

7.2 Conclusions 162

7.3 Investigation of the Hypothesis 163

7.4 Further Work 165

7.5 The Broader Context 169

APPENDIX A : Bibliography – References *171*

APPENDIX B : Mathematical Background *191*

B1: Local Fitting of Quadric Surface Patches **191**

B2: Curvature of Implicit Surfaces using the Fundamental Forms **193**

B3: Curvature of Implicit Surfaces using Minimization **194**

B4: Solution of the Non-linear Eigenproblem for Prinicipal Curvatures **199**

APPENDIX C : Implementation and Platform Issues *202*

Index of Figures

Figure 1: Examples of 3D body surface scanners 11

Figure 2: Examples of 3D data obtained by body surface scanners 14

Figure 3: Fitting conic sections to body scanner data using the 3D extension of the method described in [Book79] 33

Figure 4: Building a surface model of a real (scanned) human body using a deformable model 35

Figure 5: Implicit Functions vs. Height Functions. 41

Figure 6: Sufficient vs. Insufficient Sampling. 42

Figure 7: Anatomical Landmarks, detected using the software developed by L. Dekker [Dekk00] 43

Figure 8: Modelling of the leg and armpit regions of a whole body scan using B-spline patches that intersect [DoDb99b] 46

Figure 9: Modelling of the armpit region of a whole body scan using triangulated meshes with vertex replication [Dekk00] 46

Figure 10: Examples of whole body surface reconstruction where interpolation has been used to cover gaps in the original data 48

Figure 11: Diagram of the overall evaluation strategy 50

Figure 12: The test objects used for evaluation of the results. Left to right: Ellipsoid, Tooth, Rounded Octahedron, Bumpy Sphere 50

Figure 13: Ridges on a bumpy sphere (reproduced from [Port94]) 52

Figure 14: Forms of synclastic,developable and anticlastic shapes, reproduced from [Gass83] 56

Figure 15: Neighbour collection using a simple Euclidean distance threshold. 58

Figure 16: Problems with neighbour collection when the radius is too small 59

Figure 17: Problems with neighbour collection when the radius is too large. 59

Figure 18: Neighbour collection using a path length threshold. 60

Figure 19: Improvement of neighbourhood collection using PCA. 62

Figure 20: Accuracy of surface fit by minimizing: the height error the algebraic error; and the orthogonal distance 66

Figure 21: Illustration of the closest point calculation.. 67

Figure 22: Results of the quadric patch fitting for test data representing surfaces of known types 69

Figure 23: Application of the proposed patch fitting algorithm to real 3D scanner data.

....70
Figure 24: Histograms for the values of the evaluation metrics of quadric patch fitting. 74
Figure 25: Graphs for the mean values of the evaluation metrics for the test ellipsoid with increasing levels of noise...79
Figure 26: Graphs of the mean values of the evaluation metrics for a noisy ellipsoid with increasing levels of smoothing...80
Figure 27: Lack of scale-invariance of curvature properties. 92
Figure 28: Interpretation of the values of the shape index...93
Figure 29: Propagation of red and blue curvature labels based on the directions of principal curvature. 94
Figure 30: Interpretation of the colour mapping to the results..96
Figure 31: Curvature mapping results for the test ellipsoid..97
Figure 32: Curvature mapping results for the test tooth surface..98
Figure 33: Curvature mapping results for the test rounded octahedron..........................100
Figure 34: Curvature mapping results for the bumpy sphere..101
Figure 35: Results for scans of human torsos (front view). 103
Figure 36: Results for scans of human torsos (rear view)...104
Figure 37: Curvature directions: Thin male, fat male, thin female, fat female.............105
Figure 38: Changes to the Gaussian curvature map as a female torso deforms from a thin to a fat body type...105
Figure 39: Changes to the shape index map as a female torso deforms from a thin to a fat body type..105
Figure 40: Results for a whole body Hamamatsu scan of a male subject.....................106
Figure 41: Results for a whole body Hamamatsu scan of a female subject..................107
Figure 42 (part 1 of 2):Histograms of differences between the actual values of curvature properties on the test objects, and the values obtained via the fitted quadric patches. 112
Figure 43: Behaviour of the curvature calculation algorithm on an ellipsoid with varying levels of uniform noise...114
Figure 44: Graphs for the mean values of the absolute differences between calculated and actual values of the Gaussian and mean curvatures for the test ellipsoid with increasing levels of noise. 116
Figure 45: Graphs for the mean values of the absolute differences between calculated and actual values of the curvedness and shape index for the test ellipsoid with

increasing levels of noise. 117
Figure 46: Graphs for the mean values of the absolute differences between calculated and actual values of the Gaussian and mean curvatures for a noisy ellipsoid with increasing levels of smoothing. 118
Figure 47: Graphs for the mean values of the absolute differences between calculated and actual values of the curvedness and shape index for a noisy ellipsoid with increasing levels of smoothing. 119
Figure 48: Types of ordinary umbilics, and the behaviour of lines of curvature around them 121
Figure 49: Detection of possible near-parabolic points by searching for zero-crossings of the Gaussian curvatures. 124
Figure 50: Illustration of the definition of ridges. 126
Figure 51: Vertices on a curve. 127
Figure 52: Ridge detection on the discretised data. 128
Figure 53: Flexcord detection on the discretised data 131
Figure 54: Results on the trivial example of a sphere 134
Figure 55: Ridges and umbilics on the test ellipsoid. 135
Figure 56: Ridge detection results on a tooth surface 137
Figure 57: Ridges and umbilics on a rounded octahedron 138
Figure 58: Ridges and umbilics on a bumpy sphere 139
Figure 59: Sensitivity to shape. 140
Figure 60:Detection of small ridge features 141
Figure 61: Results for low-resolution torsos (shown along with the Gaussian curvature map) 142
Figure 62: Results for low-resolution torsos (shown along with the shape index map). 142
Figure 63: Maximal ridges for high-resolution Hamamatsu whole body scans, shown along with the Gaussian curvature map (no thresholding) 143
Figure 64: Maximal ridges for high-resolution Hamamatsu whole body scans, shown along with the Gaussian curvature map (including hysteresis thresholding) 144
Figure 65: Planes of symmetry for the test objects. 147
Figure 66: Ridge detection response diagram for the case of the test ellipsoid 152
Figure 67: Ridge detection response diagram for the case of the test tooth surface. 153
Figure 68: Ridge detection response diagram for the case of the test rounded octahedron 154

Figure 69: Ridge detection response diagram for the case of the test bumpy sphere ... 155

Figure 70: Summary of response results for various values of the distance threshold in the case of a bumpy sphere. ... 156

Figure 71: Ridge Points on a deforming torso, with the Gaussian curvature map rendered on the surface. ... 157

Figure 72: Distances d_r between each deformation frame and the initial shape. ... 159

Figure 73: Distances d_r between successive deformation frames. ... 159

Index of Tables

Table 1: Curvature values returned for the test patches of Figure 22 69

Table 2: Summary of statistics for the evaluation metrics of quadric patch fitting. 72

Table 3: Summary of statistics for the evaluation metrics of quadric patch fitting following the a posteriori thresholding on D. 73

Table 4: Mean values of the evaluation metrics for the test ellipsoid with increasing levels of noise 78

Table 5: Mean values of the evaluation metrics for a noisy ellipsoid with increasing levels of smoothing. 80

Table 6: Summary of statistics for the evaluation of curvature calculation 110

Table 7: Ranges of the actual (analytically calculated) values of the curvature properties for the test objects. 111

Table 8: Mean values of the absolute differences between calculated and actual curvature property values for the test ellipsoid with increasing levels of noise. .. 115

Table 9: Mean values of the absolute differences between calculated and actual curvature property values for a noisy ellipsoid with increasing levels of smoothing. 118

Table 10: True and false positives for the test ellipsoid 152

Table 11: True and false positives for the test tooth surface (detection includes only maximal ridges) 153

Table 12: False and true positives for the test rounded octahedron (detection includes only maximal ridges) 154

Table 13: False and true positives for the test bumpy sphere (detection includes minimal ridges, and the distance threshold set was 15mm for a typical sampling resolution of 12mm) 155

Table 14: Summary of response results for various values of the distance threshold (in tenths of millimetres) in the case of a bumpy sphere 156

Table 15: Distances d_r between each deformation frame and the initial shape. 158

Table 16: Distances d_r between successive deformation frames. 159

1 Introduction

In the recent years, there have been significant advances in the development and manufacturing of 3D scanners capable of capturing detailed (external) images of whole human bodies. Such hardware (Figure 1) offers the opportunity to collect information that could be used to describe, interpret and analyse the shape of the human body for a variety of applications where shape information plays a vital role (e.g. apparel sizing and customisation; medical research in fields such as nutrition, obesity/anorexia and perceptive psychology; ergonomics for vehicle and furniture design). However, the representations delivered by such hardware typically consist of unstructured or partially structured point clouds, whereas it would be desirable to have models that allow shape-related information to be more immediately accessible.

Figure 1: Examples of 3D body surface scanners. Left-to-Right: Hamamatsu, Cyberware, Wicks&Wilson.

This thesis describes a method of extracting the differential geometry properties of the body surface from unorganized point cloud datasets. In effect, this is a way of constructing curvature maps that allows the detection on the surface of features that are *deformable* (such as ridges) rather than *reformable* under certain transformations. Such features can subsequently be used to interpret the topology of a human body and enable classification according to its *qualitative shape*, rather than its *quantitative size* as is currently the standard practice for many of the applications concerned.

1.1 Background

The problem of building robust and accurate models to represent bodies of real humans is not a new one. Work started primarily in the middle and late nineties [Thal90][LiJo94][JoBW95][Nurr97][BFHS98][HiGe98][OlBu99], when the first 3D scanning devices emerged on the market [DeHW97][DaJW98][Hori98][Dour99]. Many approaches have been followed, according to the hardware available and the application domain of interest. As a result, there are now available some systems that are optimised either for extracting accurate measurements from parts of the body, or for realistic visualisation and animation for use in games, virtual environments and, lately, even e-commerce applications.

1.2 Motivation

What has been missing so far, is a way to model the whole body, using a completely automatic approach. Although the 3D data capturing hardware has reached a mature stage, and there are many systems that can capture 3D surface images of the whole human body with sufficient speed, accuracy, and reliability [Dour99], these systems generally are not complemented by sufficient modelling software built specifically for human bodies. The significance of this research lies in the fact that, if successful, it will enable the creation of affordable, high-performance human body modelling systems that can be used without requiring an expert operator. Mesh-based approaches[1] have been used extensively, but as the demands of modelling increase, these approaches are beginning to show their limitations, namely their local nature and their lack of "intelligence" (since in most cases they are unable to handle imperfect data sets).

The new trend seems to be the use of models that incorporate information on the variation in the shape of the class of objects that they represent. Such models, (for example: parametric models [CDTL95][MITe95b][NFSK95][FuBr96][StBa98a] [RuBu01]; statistical models such as Active Shape Models [Gowe71][Good91] [CTCG95][BaHo95][SmTA96][KoTa98][LuHa99]; or even models based on genetic algorithms [BrSt96]) have been used successfully for parts of the body such as faces and heads [KLML96][LaTC99][VWVG99][HaFr99][BlVe99][HuHD99][HuBH01].

However, the use of such methods is still embryonic in the case of modelling the whole body [Hilt99][OnGo99]. This is because human faces and heads have some nice properties, most importantly a simple, "deformed sheet" topology (faces) or "deformed sphere" topology (heads) with which it is relatively easy to work. A whole human body, however, is characterised by more complicated geometrical and topological characteristics, such as branchings and self-occlusions that: (i) make the capturing of sufficient data problematic and, (ii) impose a significantly large number of constraints

---

[1] (see e.g. [LoCl87][LiJo94][Hopp94][JoBW95][BrGK95][HSIW96][QiTe97][StBa98][Cust98][AmBK98][SeSS98][HiGe98] [DoDB99][OlBu99][Dekk00][KaSK00][Floa00])

on the modelling process. The latter is a very important problem: constraints have to be placed that incorporate assumptions about the object that is to be modelled, and they have to be placed *prior* to the modelling. However, these assumptions can only be made reliably when there is sufficient knowledge about the shape of the object, i.e. *after* the modelling. This "chicken-and-egg" situation appears to be a typical problem in most 3D human body modelling applications [LiJo94][Nurr97][NeTh98][Hilt99][PlFA99][DoDB99a][DoDB99b][Dekk00][RuBu01].

Finally, an important note must be made here regarding the use of 3D imaging for human body shape analysis: there has been a lot of work on the analysis of human body shape in 2D. It is expected that the use of three dimensions will increase the robustness of any technique. This is because the shape of a 2D image is littered with 3D orientation components [HaFR99]. Ideally those components should be identified and removed, because all they do is distort the appearance of the object. However, in 3D the components of shape variation constitute useful information rather than undesirable artifacts.

1.3 Significance of the Research

This research is primarily concerned with the problem of converting *data* from 3D scanners to useful surface and shape *information* about the human body. It is expected to extend methods established in 2D (e.g. for height functions) into 3D in order to produce a technique for analysing human body shape in a way that can be useful in a number of application domains including medical research and e-commerce.

Since the early days of 3D whole body scanning technology, there has been a problem with how best to represent the information acquired by the hardware. The problem of modelling has always been essentially the conversion of *data* to a representation of *3D objects* from the real world.

There are a number of variants of the technologies used for acquiring 3D human body data, but in all cases the primary output of such a scanner is a set of points in space. This point cloud is a "lowest common denominator" between outputs from different body scanners, and will be referred to from now on as our *original data* (Figure 2b), and the assumptions regarding its properties are discussed in detail in section 3.1.1. Furthermore, this 3D point cloud might be unorganized, semi-organized (i.e.

accompanied by partial information about the connectivity of the points - Figure 2c) or fully organized (i.e. triangulated)[2], and it might be accompanied by appearance information in the form of colour values at each data point or texture maps [Dour99].

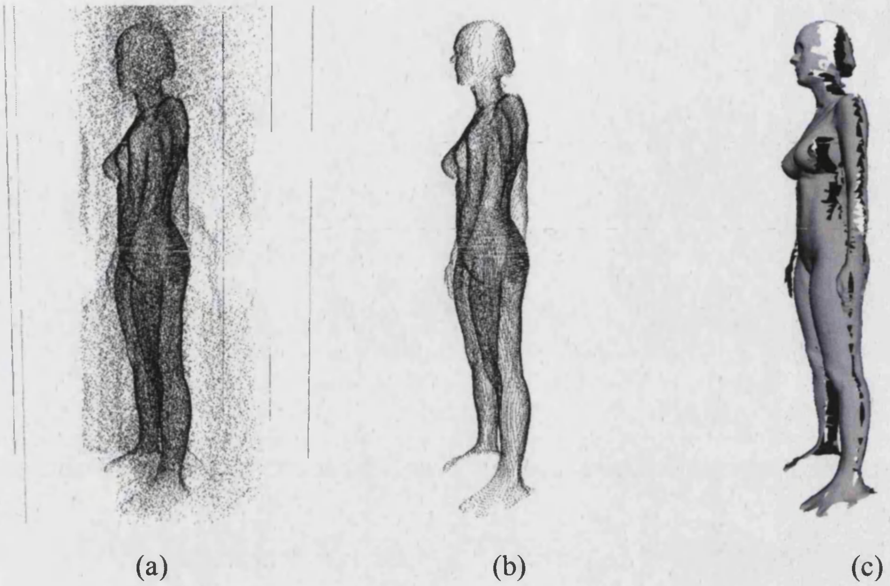


Figure 2: Examples of 3D data obtained by body surface scanners. (a) Raw point cloud, as delivered by the scanner, including outliers corresponding to the scanner walls and to secondary reflections. (b) Point cloud with the outliers removed, and only those points left that are deemed to represent the body surface ("original data"). (c) Partial triangulation, inferred by utilising implicit knowledge about the scanner structure and the data format used by the hardware.

In any case, the data set lacks a number of properties that are often necessary in applications. These properties are:

- A robust and consistent representation of the scanned object's *geometry* (i.e. its *curvature characteristics*).
- A description of the object's *topology*.
- A description and classification of the object's *shape*.
- A collection of *semantic information* associated with the above (i.e. *higher order features*: information that relates to the *object*, rather than to the *data* representing it).

All this information and properties are inherent in the data, and can be extracted if the data is of sufficient quality and a number of safe assumptions can be made regarding the object's topology (in our case, human bodies). However, as pointed out by Dekker [Dekker00] there is no known standardised way of extracting this symbolic information

---

[2] The issue of whether any such connectivity information should be used is discussed in detail in section 3.2.7.

automatically. In most applications to date there are stages at which human intervention is involved, whereby a human operator must observe and scrutinise the data in order, for example, to remove artefacts, place landmarks, fill in missing data or take measurements. These processes are normally slow, tedious, and, potentially, of questionable accuracy and reproducibility.

For instance, in the course of the research we have occasionally had the opportunity to experiment with a number of scanners, acquire data and use the accompanying software. Such scanners were the Hamamatsu Body Lines Scanner, the Telmat Symcad, the Wicks and Wilson TriForm, and the [TC]^2 scanner. All of these systems offered automated surface reconstruction and 'automated' landmarking measurement extraction software for apparel design applications (i.e. measuring tape simulators). With most subjects, the software did operate reliably enough to produce a full set of body landmarks and measurements that at least *appeared* to be correct[3]. However, all of these systems presented cases where the results were obviously wrong. For example, the body scans were mis-segmented, body landmarks could not be located or there were obvious outliers that were affecting the values of the measurements. Every time such an error occurred, manual interventions such as those described in the previous paragraph were necessary in order to correct the data. It is worth noting that in many of these cases, even after manual interventions the results have been so poor that the data had to be discarded and the scanning process repeated, in the hope that similar errors would not re-occur.

This research is an attempt to address the problem of extracting such information (geometry, topology, shape and features) from 3D scanner data in a robust and *fully automated* manner, in the case where the object of interest is a human body. The contribution to existing knowledge and research is a set of implemented, tested and evaluated techniques and algorithms. Together these form a hardware-independent (within the types of hardware indicated in section 1.2) method of processing a 3D point cloud in order to extract higher-level information (i.e. surface features) that can be fed into a number of useful applications. The research is inspired by similar work with 2D data (i.e. Image Processing) and is expected to lead towards the development of the technology that can make 3D scanning as mature and usable as 2D imaging, with the

---

[3] A rigorous benchmarking exercise involving these and a number of other systems is known to have been carried out as part of the SizeUK project. Unfortunately, because of the obvious commercial sensitivity of the results, there is no access to them so far.

additional advantage of the much richer information that is implicitly contained in 3D data.

Finally, the research is addressing the issue of invariance and evolution of surface features as the surface deforms. This is to be addressed in the context of classification of shapes using their curvature properties. This issue is discussed in greater detail in section 7.4.4.

1.4 Hypothesis

Our hypothesis is that:

- we can *reliably* construct *curvature maps* of human body surfaces from *3D scans*, such that:
- we can *reliably* extract *point features*,
- we can *reliably* extract *line features*,
- we can *classify* the surface.

Definitions of the terms in italics (within the context of this thesis) are given in section 1.6 below.

1.4.1 Scope

In particular, the work presented here has been concerned with:

- Justifying and defending the first three points of the hypothesis by implementing algorithms that construct the curvature maps and extract the features, and by providing a quantitative analysis of the results that shows that the maps and feature extraction are reliable.
- Attempting to illustrate a scheme that could use the results of feature extraction in order to classify bodies into topological categories (more details are included in section 7.4.4).

Note that the complete classification of human body types in a mathematical or quantitative sense is beyond the scope of this research. Its implementation has therefore been neither expected nor intended.

1.5 Objectives

Given the hypothesis stated above, the following objectives have been pursued throughout the duration of the research:

- To show that it is possible to build a model that describes accurately the surface geometry of a real human body, using data taken from several types of whole body 3D (surface) scanner.
- To show that the method to be developed is *reliable*, i.e. that it works for almost any human body within a well defined and reasonably wide range of body shapes.
- To show that the method to be developed is *robust*, i.e. tolerant to a reasonable level of noise, missing data, and errors occuring after each intermediate step of processing.
- To show that the method developed can be *automatic*, i.e. with no user intervention at any stage between scanning and extraction of the surface geometry.
- To address, and attempt to resolve, the theoretical and technical problems that are introduced in the process of achieving the aforementioned objectives.
- To provide a quantitative assessment of the method's *accuracy* (i.e. how true the calculated curvature characteristics and shape features are to those of the actual object scanned), and its reliability and robustness.
- To illustrate the possibility of using the curvature properties obtained to classify body types. Such a classification could be useful, for example, in application domains such as clothing design and medical research.

1.6 Definitions of Basic Concepts

In the interest of clarity, it is essential at this point to mention that this research is concerned solely with the study of human bodies, represented as 3D scans. More specifically, throughout this report:

- "3D Modelling" refers exclusively to the modelling of human body 3D scan data. In particular, we do not attempt to produce generic modelling techniques that would work with *any* 3D object.
- "Scanning" refers to the use of surface scanners (photonic or photogrammetric) that generate point cloud representations of the surfaces of human bodies. Such representations will be hereafter referred to as "3D Scans". Note, in particular, that in the context of this research the term "3D Scan" does not refer to voxel images of the inside of the human body, such as those obtained using a CT or MRI scanner. It is assumed that the point clouds are sufficiently dense in relation to the density of

identifiable features on the surface. It is also assumed that the point cloud is either structured (triangulated), or in a form such that a reasonable neighbourhood relationship between its points can be inferred (for more details on the issues surrounding this please see section 3.2.7).

Also, in order to clarify the hypothesis that is at the core of this research, it is important to define a number of important concepts related to surface[4] processing:

- Point features: are points on a surface that have unique properties in comparison to other, nearby points. These properties depend on the application domain, but in any case allow us to identify and locate the points. Also, when we have a set of 3D objects of the same class, these points can be found on the surface of any object in the set. Typical examples include: local minima and maxima of Gaussian curvature, minimal saddles and ordinary umbilics (see, e.g. [Port94], especially p.184).
- Line features: are contour curves on a surface that have unique properties in comparison to other nearby curves. These properties depend on the application domain, but in any case allow us to identify and locate the curves Also, when we have a set of 3D objects of the same class, these curves can be found on the surface of any object in the set. Typical examples include: ridge curves, bitangent curves, parabolic curves and subparabolic curves (flexcords).
- Curvature Profile: of a point on a surface *S*, we define to be a record holding information related to the curvature properties of the surface. This includes both differential geometry and curvature information that is independent of the chosen representation and frame of reference (e.g. Gaussian curvature, shape index), as well as related information that is representation dependent (e.g principal curvatures, tangent plane, normal vector, the fundamental forms and the curvature tensor for that point).
- Curvature Map: is a mapping from the 2D parametric space (u,v), given a surface $S(u,v)$ in R^2, to a set P of curvature profiles, such that $P(u,v)$ is the curvature profile of point $S(u,v)$ on the surface.
- Object Shape Classification: is the process whereby the members of a set of 3D objects (represented as surfaces) of the same conceptual class or type are labelled as members of conceptual subclasses. One of the aims of this research is to produce (or

---

[4] Rergarding the concept of "surface", the reader must note that occasionally different research communities may understand different definitions. There is an important note on this at the final paragraph of section 2.4.1.

at least to work towards producing) a classification of body shape in terms of the curvature profiles of a surface and their properties. At present, the classification of human body shape is either quantitative (by means of key measurements, such as clothing "size"), or descriptive by terms such as "endomorphic", "mesomorphic", "ectomorphic", etc. [Shel54].

- Anatomical Landmarks: are features (usually points or curves) on the surface of a human body that indicate the location of a body organ or component. These features do not necessarily have a robust definition in terms of the surface geometry. However, their definitions are often more intuitive and, since they are anatomically meaningful, they can easily be located by a human expert or semi-expert. Such landmarks are typically used in the clothing industry [Nurr97][Dour99] [Dekk00][RuBu01] and also in anatomy and anthropology [Book91][MITe95a] [TiLe00][HHNH01]. Typical examples include: nape of the neck (7th cervical vertebra), top of the head, iliac crests, underbust and mid-clavicle point.
- Geometric/Computational Landmarks: are features on a surface that can be indentified and located by algorithmic means [Book91][GoAy93][Port94] [HiBT97][VWVG99][HuCH00][PoPu00]. In some cases these landmarks may correspond to anatomical features and organs of the body, but in general they do not.
- Surface Matching: Is the process whereby a transformation is calculated and applied to one of the surfaces in order to minimize a distance function defined on the two surfaces or their features [Good71][Gowe75][Besl92][GoAy93][PaVG95][LoEF95] [BaSh97][FlBH96][YaFa98][LuHa99][Fitz01].

1.7 Contributions

The work presented in this thesis makes four main research contributions:

1. In the course of pursuing the objectives arising from the hypothesis, the research has prompted (and achieved) a synthesis of implicit function methods for surface fitting and computation of surface curvature properties.

2. The thesis describes the design rationale, development, testing and assessment of a system for the computation of curvature properties and characteristics of the human body surface from 3D optoelectronic scanner data. The implementation of this system was enabled by the above synthesis of implicit function methods. This system constitutes, in fact, the primary and most important contribution of this

research. It is reliable, robust, modular, fully automated, works on a variety of 3D scanner data for human bodies, is independent of the co-ordinate system in which the data are represented, is capable of processing representations of organised or unorganised point clouds, and can handle noisy data. In particular, the system includes modules for:

- Fitting quadric surface patches to 3D scanner data.
- Direct calculation of the surface curvature properties from the implicit form.
- Detection of deformable surface features (in particular ridges), via a process similar to the Canny edge detector [Cann86].

All the above have been validated on simple geometric objects, and evaluated on both geometric and real (body) objects.

3. In the course of the research, a number of publications were generated ([DoDB99a] [DoDB99b] [DDBT99] [WDFD00] [BDDV00] [DoBu02] [RBDT02] [KaDT03] and [Dour04]) that are related to the work presented in this thesis and demonstrate its relevance to applications related to medical research and the clothing industry. These are all papers appearing in conference proceedings, however a journal publication is on the way at the time of writing this thesis.

4. Parts of this research have been presented as part of the work on the following projects:

   - Wellcome Trust BodyScan
   - ALSPAC
   - 3D Centre
   - SizeUK
   - e-T Cluster
   - e-Tailor
   - Eurasia-Tex

   Of these projects, the first two on the list were concerned with the medical applications of 3D body surface scanning and modelling, the others with the clothing industry and e-commerce applications. In all cases it has been possible to utilise the opportunities for exposure of the outcome and benefits of this work to the relevant research communities.

1.8 Structure of the Thesis

The rest of this thesis is structured as follows: Chapter two is an overview of the previous and current attempts to model 3D objects in general and human bodies in particular. It involves an overview of the mathematical and technical issues associated with the modelling. Chapter three presents an overview of the methodology employed throughout the research, the assumptions regarding the data to be processed, and the strategy for evaluating the results for each stage of the methodology. The latter is necessary in order to establish that the steps of the proposed method are robust and, if possible, reliable.

Chapter four describes an algorithm together with some variations of it for approximating the local surface geometry around a given point of the input data set by means of a least-squares minimization. The output of such an algorithm is a surface patch described in an analytic (implicit) form. This is necessary for the next step described below. The case is made for using implicit surfaces rather than more popular 3D surface representations such as parametric forms or height functions.

Chapter five describes the processing needed for calculating curvature-related characteristics for each point of the input surface. This utilises the implicit surface patches generated by the algorithm described in the previous chapter, and enables the construction of a "curvature map" of the original surface, which incorporates rich information such as the principal curvatures, shape indices and principal curvature directions.

Chapter six describes a family of algorithms for calculating features such as ridges and umbilic points on the surface from the curvature map, in a manner that bypasses the problem of separating a vector field (i.e. the principal curvature directions) across the entire surface of an object. An alternative approach, using the focal surface information, is also considered for comparison.

The concluding chapter summarises the results from all steps of processing and evaluates them in relation to the requirements set out in this first chapter. Directions for further research are also proposed.

2 Existing Work

This chapter is an overview of previous and current attempts to model 3D objects in general and human bodies in particular, as well as of the mathematical and technical issues associated with the modelling. Section 2.1 focuses on the application domains where the potential for using 3D human body modelling techniques has already been identified, and some attempts to utilise such techniques have already been carried out. The remaining sections of this chapter present work that has been carried out on issues that are related (albeit not always exclusively) to the problem of building a surface model of a whole human body from real data.

2.1 Application Domains

The list of applications that could benefit from the use of 3D human body modelling technologies is potentially limitless. However, we present here those that are the most immediately relevant and perhaps also currently the most significant: medical research; clothing design and manufacturing; and media/entertainment.

2.1.1 Medical Applications

Given that the subject of medical science is the human body itself, there is a significant number of opportunities created by 3D body metrics and modelling for medical applications. These opportunities arise from the 3D optical whole body scanners capable of rapidly capturing accurate data on human body size and shape which are now beginning to appear on the market. Examples include the Cyberware, Hamamatsu and Telmat scanners [Dour99]. However, 3D applications software in the medical research area is still embryonic and there is an urgent need to rectify this in order to maximize the benefits of the new technology. Once sophisticated software is developed, such scanners will have a major impact on areas that are related to health and fitness research such as:

- Evaluation of body composition (through the automatic calculation of body volume, and therefore body density, etc),
- Study of nutritional disorders (e.g. obesity and anorexia),
- Monitoring and prediction of body growth,

- Analysis and treatment of posture and skeletal problems (osteopathy),
- Treatment of burns and injuries,
- Prosthetic and cosmetic surgery,
- Body shape analysis, body kinetics and performance analysis.

Examples of such applications are given in [Book91][MITe95a][BaCA96][SmTA96] [FlBH96][LaTC99][HaFr99][BlVe99][HuHD99] and [WDFD00]. In comparison to alternative techniques, photonic methods present many advantages including:

- Use of *safe*, *non-invasive* visible or infrared light, with little intrusion on the subject.
- *High speed*, allowing rapid subject throughput and low running costs.
- High *accuracy* and *resolution*.
- Easy integration with digital computer technology for data storage, communication, manipulation and analysis.
- Multiple functions of a single scan.
- Complete measurement *automation*, allowing *standardisation* within and across organizations.

These advantages are particularly important, for example, in paediatric and geriatric medicine, since alternative technologies are frequently inappropriate for the patient groups involved [WDFD00]. There are also, of course, a number of disadvantages of using surface imaging techniques, most notably:

- Most techniques are posture-specific, and a patient's condition may prohibit the adoption and maintenance of the appropriate posture for data capture.
- The reliability of modelling techniques is not yet sufficiently studied to assess its suitability for critical medical applications (although this research is hoping to contribute towards this direction.
- Surface imaging techniques are solely based on the outer body shape, and do not convey any visual information about the inside of the body (i.e. in a way that CT or MRI imaging do), and there is still a great deal of work to be done in order to establish relioable associations between body shape and physiology.

However, all these disadvantages represent challenges that warrant ongoing research effort in the areas of both 3D imaging hardware and 3D surface modelling.

2.1.2 The Clothing Industry

Clothing designers, manufacturers and retailers represent an important industrial sector that is interested in human body analysis and visualisation technologies, owing to the enormous potential of such technologies in, for example, clothing design and manufacturing, and especially in e-commerce. The clothing industry is on its way to adopting virtual shopping systems and technologies. These will enable the customers to visualise themselves virtually in a garment before purchasing it, thus helping ensure customer satisfaction and reducing return rates on purchases[5]. Projects for the development of such technologies (such as the Centre for 3D Electronic Commerce, pioneered by UCL) involve the development of shape analysis and visualisation tools, as well as very large-scale surveys of the market population [RBDT02]. This is in order to identify the major body types and sizes and accordingly adjust retailers' and manufacturers' sizing scales so that they match the market population well.

When 3D scanning hardware first became available [JoBW95], research focused on 1D measurement techniques (automatic tape measurements) [Dekk00]. Subsequently, attention turned to 2D surface techniques [DoDB99a][DoDB99b]. It is now appropriate to investigate and exploit data analysis and manipulation methods that are based entirely on 3D mathematical models. There are particular advantages to the use of such models, namely their robustness, accuracy, and maximum usage of geometrical and structural information, that make them optimal for applications where stability and reliability are as important as accuracy [Gowe75][Good91][CTCG95].

2.1.3 Media and Entertainment

Digital representations of human bodies are currently in constant demand. From the use of realistic avatars in virtual environments to the innumerable applications in the gaming industry, the interest is focused on enhancing the user's feeling of participation in simulated action. An application example that has been much discussed is the replacement of an actor in a motion picture by a visual representation of another actor, which might even be the user herself or himself. Use of the appropriate human body modelling and animation techniques in combination with recently established data streaming and compression standards such as MPEG-4 will enable the development of such applications in the very near future [Thal90][BFHS98][KMMA98][OlBu99]

---

[5] Such a system has been recently implemented by BodyMetrics Ltd. and is being tested at the Selfridges flagship store in London at the time of writing.

[PlFA99][APFG99][HBGS99][SCSA02]. Another application is the user interaction with human avatars, as presented, for example by Lee et al[LCRH02].

2.2 Human Body Scanning Systems

Three dimensional image capture is still a developing technology. Various systems constantly emerge in the market, whose detailed description is beyond the scope of this thesis (a list of a number of systems with comparisons between them has been compiled by Daanen [DaJW98], and another collection of information on systems and technologies has been compiled within the framework of the UCL 3D Centre project [Dour99], whereas a more up-to date overview can be found in [Dour04]). However, a description of the two major imaging technologies employed in this system is given in the sections that follow.

2.2.1 Range (Depth Sensor) Scanners

These scanners are based on an array of depth sensors such as the one developed by Horiguchi [Hori95] - currently used in the Hamamatsu Body Line Scanner range [Hori98]. These sensors fire a laser beam at the object and, by measuring the direction of the reflected ray, calculate the distance between the sensor lens and the incident surface. With the location and orientation of the sensor known from the scanner's construction and calibration, it is easy to calculate the Cartesian co-ordinates of the point in space. There are some later descendants of this technology that are of particular interest, because they have led to the development of systems that are portable or even handheld, such as those by Konica Minolta (VI-900) and N-Vision (ModelMaker). The major disadvantage of this technology is that it involves scanning of a surface on a line-by-line or point-by-point basis, which effectively means slow scanning speeds (order of a few seconds) that make it impossible to scan objects in real time or in motion. On the other hand, systems built using depth sensors can operate at light frequencies outside the visible spctrum, which makes them very robust with regard to lighting conditions.

2.2.2 Photogrammetric Scanners

These scanners are based on conventional 2D cameras. Typical examples include the systems developed by Human Solutions (Germany), Telmat Industrie (France), [TC]^2 (USA), Wicks&Wilson (UK) and, more recently, SurfIM (UK). These systems typically consist of usually several pairs of cameras, the images of which are sent to a computer that calculates the binocular disparities between corresponding points in each pair of images. Thus the 3D geometry is inferred using stereoscopy [DeHW97]. In some models, a sinusoidal pattern [DeHW97] is projected on the object in order to generate a rich texture that assists the identification of feature points. This photogrammetric technology relies heavily on the quality of 2D photographs taken, the lighting conditions, and the efficiency of the stereo matching algorithm employed. This is a potential disadvantage. However, photogrammetric scanners are capable of much higher scanning speeds (typically the order of millimetres) which makes them ideal for frame-by-frame (real time) 3D capture, which can also be relatively posture-independent as long as there is suffient coverage of the subject. They are also usually capable of delivering point clouds of much higher density (less than 1mm, i.e. up to 10 times better than depth sensor systems can typically manage), and with full colour and texture information.

2.3 Human Body Modelling Work

Pioneering work on an integrated approach to Human Body Modelling (i.e. use of hardware and construction of a high-level application software system) has been carried out by the HUMAG group at Loughborough University, led by Peter Jones. The early stages of this work included the use of the LASS scanner [LiJo94] to capture data that was subsequently cleaned, segmented, and represented (torso only) as a series of B-spline curves [JoBW95]. This work has shown that it is possible to represent a human torso very efficiently by parameterising key slices of the body.

Additionally, Hilton et al. [HiGe98][Hilt99][HBGS99] at Surrey University have been working for a number of years on the construction of realistic human avatars from real data. Their most recent work involves the extraction of body silhouettes from a number of 2D views (front and side) and the subsequent deformation of a 3D template to fit the silhouettes. The 2D views are then mapped as texture onto the deformed model to enhance realism. Also, in Switzerland, an enormous amount of work has been carried out jointly by the LIG/EPFL group (D. Thalmann) and the MIRAlab group (N. Magnenat-Thalmann) [Thal90][BaTh99]. Their work covers many areas including: surface reconstruction, articulation, animation and rendering, with an emphasis on realism and the representation of real people (i.e. not CAD models).

In the Computer Science Department at UCL, work on human body modelling has also been going on for a number of years. Dekker [Dekk00] has used a series of meaningful anatomical assumptions in order to optimise, clean and segment data from a Hamamatsu whole body scanner in order to generate quadmesh representations of human bodies and build applications for the clothing industry. There is currently a system capable of identifying up to 90 significant landmarks on the body surface and of using them to calculate sets of anthropometric measurements. Research has also been carried out on the generation of B-spline whole body representations [DoDB99] from the scanner data.

Other examples include the work of Ong and Gong [OnGo99] for the generation of a dynamic human model using hybrid representations in hierarchical PCA space, as well as the work of Fua et al. [PlFA99] on automated body modelling from video sequences.

These are only some examples on the work being carried out for the development of technologies specifically for human body modelling. It must be noted however, that since this has recently become a very popular area, the list is far from complete.

2.4 Surface Reconstruction and Parameterisation

Conversion of a point-cloud surface representation to a parametric form is desirable, not only for smooth visualisation, but also because it facilitates the extraction of measurements in order to describe surface features. It must be noted here that, especially in the area of computer graphics, the term "surface" tends to mean a polygonal representation of the object. Such representations, however, tend to lack the flexibility required for shape analysis. It must therefore be made clear that the surfaces of interest in this section are *parametric surfaces*. Nevertheless, there are some well-established methods for triangulating point sets, such as the one based on Voronoi diagrams by Amenta et al. [AmBK98], or the Marching Cubes algorithm [LoCl87] for generating triangulated surfaces from voxels. Therefore, the problem of inferring a parameterised representation from a triangulated surface will be discussed here as well as the problem of inferring a parameterised surface representation from a point cloud. Triangulated surfaces are discussed in section 2.4.1.

Perhaps the most well known algorithm for converting unorganized point sets to B-splines is the one developed by Hoppe [Hopp94]. This is a very elegant technique that requires very few or no assumptions about the object's topology. Theoretically it can handle any object, as long as it is sufficiently sampled. It can also provide optimal results in terms of minimization of the error between data points and the surface because it involves iterative refinement. However, it has proven not to work very well with the usually imperfect, noisy and incomplete, data sets that are produced by most 3D scanners.

In an attempt to handle arbitrary topology in a controlled manner, Stoddart et al. have introduced an algorithm for 'Slime' surfaces [StBa98b][StBa98c][HSIW96]. These are G^1 continuous surfaces made up from a number of smoothly joining spline patches that are not necessarily quadrilateral. This is achieved by defining a set of voxels around the data set, and each node in the voxel set generates a spline patch with a number of sides equal to the node's valence. These patches are then iteratively refined to match the surface. Subsequently, this modelling method has evolved into Progressive Splines [StBa98a], whereby the algorithm has been modified to produce successive multiresolution surfaces. This reconstruction scheme has been combined with a self-calibration algorithm [CuSt98] for use in an integrated system.

A different approach that has also been followed is the representation of whole data sets by implicit functions[Taub94]. It appears the most popular primitives for this approach have been the radial basis functions, although existing techniques [TuOB99][BBCS99] have so far been unsuitable for large data set because of their computational demands. A fast implementation has been proposed relatively recently by Carr et al. [CBCM01].

Another approach for handling arbitrary topologies is based on the concept of active contour models (Snakes). McInerney and Terzopoulos have presented a variant of the snake algorithm for contour description that is topologically adaptable [MITe95b], and then extended it to surfaces to demonstrate its application in medical imaging.

When generating a smooth surface from a triangular mesh, the main obstacle is that the usual, otherwise easy-to-use quadrilateral splines are not very well suited to the problem. This is because there can be no straightforward 1-1 correspondence between a triangle and a quadrilateral surface without singularities. Ueshiba and Roth [UeRo99] have tried to resolve this difficulty by devising an algorithm that calculates for each triangle a set of three quadrilateral splines, smoothly joined in such a manner that a curved triangle is formed. The main disadvantage of this approach is that it compromises on accuracy, as it requires the shifting of the original vertices of the triangular mesh in order to make the spline patches join properly. The other approach is to use triangular NURBS patches as described by Qin and Terzopoulos [QiTe97]. The main disadvantage then is that these primitives are not as simple and usable as quadrilateral NURBS.

A different approach to the problem of the topological restrictions on the shape of objects that can be efficiently represented by quadrilateral splines is the use of implicit reconstruction schemes like those introduced by Doo-Sabin [Doo78][DoSa78] and Catmull-Clark [CaCl78]. This group of algorithms does not output a set of parametric surface patches. Instead, they use the NURBS interpolation scheme [SeSS98] or one of its variants in order to generate points that should belong to the complete surface. These points are generated by subdividing each triangle of the original mesh into smaller triangles, and using the result of the interpolation as an extra vertex for the mesh. The result is still a mesh, but this subdivision can be iterated to generate arbitrarily smooth surfaces. The obvious advantage of such techniques is that they can handle non-uniformly sampled surfaces. There is generally no parametric representation for these surfaces, although Stam [Stam78] has shown that there can be a closed form for the case of Catmull-Clark subdivision surfaces, so that points can be evaluated directly without subdivision. Furthermore, Blane et al. [BLCC00] have recently demonstrated a least-squares approach for fitting implicit polynomial surfaces of very high degree to point data.

The problem of parameterisation itself (regardless of whether a parametric surface is reconstructed from the data or not) is defined as the calculation of a mapping from the actual surface to a parametric domain, so that parametric co-ordinates can be defined across the whole surface. Brechbuechler et al. [BrGK95] achieve this for closed surfaces by defining a harmonic mapping of the surface onto a unit sphere whilst Timsari and Leahy [TiLe00] use a semi-isometric approach for creating flat maps of the cerebral cortex. In each case, parameterisation is formulated as an optimisation problem where, in the former, they are trying to map smoothly and uniformly onto a sphere and, in the latter, are trying to reduce the 'convexity' of the surface. In any case, the problem with parameterisation is that, even for point sets or meshes with a very regular point or vertex distribution respectively, the mapped points or vertices tend to become very dense around the centre of the parametric domain. This problem is even more serious if the connectivity of the vertices is irregular. However, Walder and Lovell [WaLo03] have recently presented an improved method of fitting algebraic curves to vector data by using an additional set of vectors that are labelled as 'interior' or 'exterior' to the curve. The method allows the use of Kernel functions and seems to provide topologically stable solutions. Schroeder et al. [LSSC98] have tried to overcome this problem by following a hierarchical approach. Finally, Floater [Floa00] has proposed a technique for meshless parameterisation, in which the parameters are determined prior to a Delaunay triangulation carried out in the parametric space.

Surface reconstruction can be made easier if assumptions can be made on the shape of the object being represented by the data. Werghi et al. [WFAR99] demonstrate a method for recovering descriptions of quadric surfaces from 3D range data that can be used efficiently to generate CAD models of mechanical parts. Similarly, Dekker [Dekk00] has used a series of meaningful anatomical assumptions in order to generate quadmesh representations of human bodies from 3D scans. This method has subsequently been expanded to generate B-spline whole body representations [DoDB99].

Finally, if the purpose of reconstruction is purely the visualisation of the data set, Schaufler [ScJe00] has demonstrated that there is a way around the problem, by using a ray-tracing algorithm that works on single points and requires neither a faceted nor a parametric surface representation.

2.4.1 Triangulated surface models

Triangulated meshes are a very standard way of representing 3D surface sets. This is both because they can be represented by manageable and indefinitely expandable data structures, but also because there are standard ways of generating them from real sensor data such as points, for example by using Voronoi diagrams (Amenta et al. [AmBK98]), or voxels via the Marching Cubes algorithm [LoCl87]. It is even possible (although not in every case) to construct from them smooth spline surfaces [UeRo99]. Identification of landmarks on such a surface representation is not the most accurate (owing to the discretisation error introduced), but it is quite easy. Available techniques range from simple measurements of angles and distances on the mesh to the more elaborate identification algorithms via LoD reduction as proposed by Taylor et al. [BrTa98], where 'non-significant' vertices are iteratively removed and all remaining ones are eventually taken as landmarks. Landmark correspondence is an issue to be addressed, although Thalmann et al. [HoTT88] have shown that some operations such as interpolation between two triangular meshes can be carried out even without identifying any landmarks. Similarly, it is possible to merge partially overlapping meshes using a zippering algorithm such as the one by Turk and Levoy [TuLe94]. This is very useful in the case of human bodies, because most body scanners typically consist of a number of sensors that cover multiple and partially overlapping views. It is even possible to carry out operations such as Gaussian smoothing on a triangulated mesh, e.g. by using a geodesic transform (Mokhtarian [MoKY98]), diffusion and curvature flow (Schroeder et al. [DMSB99]), or level-set methods (Faugeras [HeFG99]).

It is important to note here that, in the literature generated by the computer graphics community (which is by default concerned primarily with the *representation* and *visualisation* of physical entities) the term "surface" is often somewhat narrowly understood as "a set of triangulated facets". This may occasionally cause some confusion in other research communities, where a "surface" is either a physical part of a real object, or an abstract entity described by an equation. In our research, we are concerned with the mathematical representation of the actual human body. Therefore, whenever we use the term "(body) surface", we *always* refer to the latter (i.e. a physical entity and/or a non-discrete mathematical representation, rather than a polygonal set), unless specifically stipulated otherwise.

2.5 Curvature Estimation

Curvature estimation is a very important tool for the analysis of 3D surface data and the extraction of geometric features (Dekker [Dekk00], Lee et al. [LeKi95]) as well as surface registration (Nguyen et. Al [NgNS99], Gourdon et al. [GoAy93]) and shape classification (DaCosta et al. [CRAA04]). By analysing the differential geometry of the surface (see Kreyszig [Krey59] and Gasson [Gass83] for a complete overview) it is possible to classify the neighbourhood of each point as parabolic, hyperbolic or elliptic. This enables us to identify interesting parts of the surface such as recesses, bumps, ridges, creases and saddles. In the last couple of decades, there has been therefore a renewed interest in the differential geometry of surfaces, especially in the context of computation. Typical leading examples are the relevant works by Porteous [Port94][Popu00], Koenderink [Koen90][KoVD94][KoVD98], Giblin [BrGT96], Morris [Morr95][Morr96] and Belyaev [BePK98][OBAT03][OhBS02][OhBS03]. However, in the chapters that follow, we will explain how none of the known methods proposed in the literature is directly applicable or appropriate for the type of data with which we are dealing (although, of course, they have all played an important role in the design of the algorithms proposed here).

When we have a representation of a 3D surface in its analytic/parametric form, calculating the curvature is a straightforward differentiation task. However, if we are working with real data (point sets in the majority of cases) there is the additional difficulty of the surface "not being there", i.e. we need to estimate the curvature of the surface that *should* be there. This means that we first need to guess the surface of the shape locally before estimating. The presence of noise in these cases can make curvature estimation an unusually difficult task.

2.6 Local Surface Fitting

A popular approach for curvature estimation is to fit a local surface patch around the point whose curvature profile we need to evaluate, and then to calculate principal curvatures and other information directly from the analytic representation of the patch. Bookstein [Book79] has introduced a method for least-squares (LS) fitting of conic sections to scattered point data. This could be used for 3D point data and it makes immediate sense since we want to calculate and analyse the surface curvature properties and distinguish between parabolic, hyperbolic or elliptic surface patches. Bajaj et al. [BaIW93] have also studied the approximation and interpolation of 3D point clouds using implicit surfaces. However our experimentation has shown that, when extended to 3D point data, these methods can occasionally be noise sensitive, and that they may produce poor results for a surface that has a locally elliptic profile (Figure 3). Fitzgibbon [FiPF99] has tried to overcome this problem by devising an algorithm for specific fitting of an ellipse rather than a generic conic section.

Unfortunately, this is a 2D technique that apparently cannot be extended to 3D because it uses in 2D a quadric constraint expressed in a matrix form peculiar to this dimensionality. Moreover, we, of course, do not know in advance whether the surface patch should be elliptic, hyperbolic or parabolic.

Figure 3: Fitting conic sections to body scanner data using the 3D extension of the method described in [Book79] (green: quadratic patch, blue: cubic patch – red lines illustrate the local surface normals calculated from the patches.). The poor quality of the result is evident when the data set corresponds to a synclastic (elliptic) shape.

Other authors have attempted to work around the local surface reconstruction problem using various approaches such as geodesic sampling (Mokhtarian [YuKM99]), region growing (for triangulated surfaces - Saachi et al. [SPTH99]), and tensor voting with no local patch fitting (Tang and Medioni [TaMe99]).

Each one of these approaches seems to work well for particular types of data such as manufactured mechanical parts with well defined regions, or fully triangulated closed surfaces, that do not necessarily meet our assumptions about human body data as set out in section 3.1.1. A typical difficulty when range data are used is the so-called 'chess-board' effect, whereby the magnitude of the principal curvatures is estimated accurately but the sign is wrong for particular regions of the surface, thus producing curvature maps that are only partially useful. This typically occurs when the directions of surface normals (whether these are given or estimated) are not consistent throughout the surface. Such a situation is unfortunately quite likely if the initial data set is unorganized.

2.7 Feature Detection

One of the key stages in building shape models of any kind is the identification of landmarks[6]. In the majority of cases, these are points on the surface of the 3D object in question. However, this is not absolutely necessary. Any quantifiable entity such as curves or lines may constitute a landmark, as long as it is well defined and can be repeatably placed, either manually or automatically (although only the latter case is of interest for this research). Again, if the surface has not been reconstructed, then the landmarks have to be placed on the assumed surface. This introduces the additional issue of estimation of the surface shape for those regions of the object that were not visible to the scanner hardware at the time of acquisition.

There are numerous approaches for identifying landmarks on a 3D surface that is represented as a point set or as a triangulated model. Brett, Hill et al [BrHT97] have proposed a method of non-rigid correspondence, whereby they iteratively reduce the level of detail of a triangulated surface by removing non-significant vertices, and take all remaining vertices as important landmarks. Another approach by Van Gool et al. [VWVG99] uses bitangent curves as landmarks. These are detected by mapping (transforming) the surface into tangent space and then detecting the self-intersection of the transformed surface. Lee et al. [LeKi95] use a topographic feature extraction method, more akin to image processing, that is based on template matching. This is a 2D method used, for example, for character recognition. Frantz et al. [FrRS98] have also addressed the problem of localisation of point landmarks, in order to achieve subpixel/subvoxel accuracy when using discretised images.

A slightly different approach that is quite popular when trying to model the human body or parts thereof is to use both differential geometry and a set of assumptions based on knowledge of body morphology in order to identify landmarks that correspond to anatomical features. Typical examples of this approach include Hutton's work [HuCH00] on cranial data for dental structures, as well as the modelling systems developed by Nurre [Nurr97] and later by Dekker [Dekk00] for applications related to the clothing industry.

2.8 Matching and Registration

The issues of matching and registration are important for the construction and use of high quality models. It is essential, in order to compare data and models, to register

---

[6] A very good overview of the concept of landmarking and the historical evolution of landmarking techniques can be found at http://www.qc.edu/Biology/fac_stf/marcus/multisyl/Quarto.htm

multiple views of the same 3D object so that the whole point cloud (or model) exists in the same frame of reference. A very popular algorithm in this category is Besl's ICP [Besl92], which has the advantage of being representation independent. Used in combination with Van Gool's method of initial (rough) registration using bitangent curves [VWVG99] it can produce very good results and constitutes a computationally efficient solution. The use of genetic algorithms for surface matching has also been proposed (Brunnstroem and Stoddart [BrSt96]), as well as variants of the well-known Procrustes alignment [Good91] and applications of the EM (maximum likelihood) algorithm (Luo and Hancock [LuHa99]). There are also a group of relatively recently developed algorithms that use the curvature information in order to establish criteria that allow surface matching. Typical examples are the works by Sapidis [SaDM02] and Thodberg [ThOl03].

2.9 Deformable Shape Modelling and Analysis

Another important issue regarding shape modelling is the construction of Deformable Models. Work in that area has illustrated the possibility of calculating a surface model that describes a 3D point data set by deforming an existing model (reference shape) so that it matches the data (Figure 4).

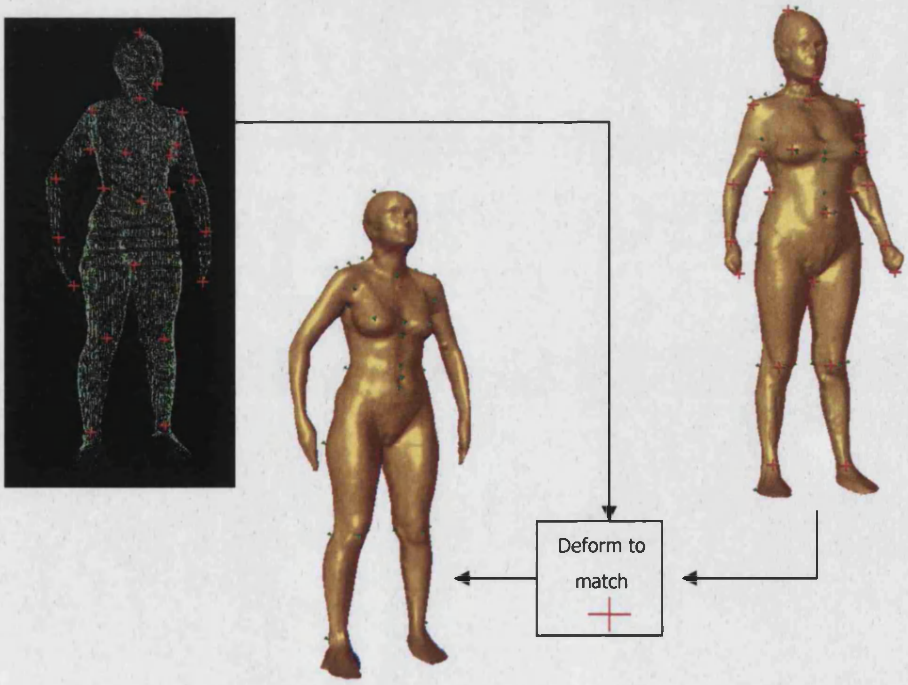


Figure 4: Building a surface model of a real (scanned) human body using a deformable model. A surface model (right) is deformed so that its geometric features (shown as red crosses) match those of a scan (green point cloud, left). The result is a surface model that matches the scan, with the anatomic landmarks (green cones) falling into place.

Deformable Modelling can be generally described as a two-phase process:

- Expressing (quantifying) a change in the object's shape (either over time or with respect to a reference shape), and
- Visualising the change (morphing).

The idea is therefore to represent the change in an object's shape simply by codifying the changes in the locations of a number of landmark points placed on the object's surface. The use of deformable models is focused on quantifying the magnitude and significance of such changes. It is, however, also important to calculate the changes that occur at other points on the surface. This is normally done by means of interpolation. A very popular tool is the use of thin-plate splines as presented by Bookstein [Book89][Book91]. The idea behind thin-plate splines is to simulate the physical behaviour of a very thin metal plate. Landmarks on the surface represent points on the plate at which forces are applied, whereas landmarks on the target surface (with a 1-1 correspondence) define the force vectors. The model allows the surface to deform so that source landmarks match the target landmarks, and the rest of the surface is accordingly 'dragged-along' without introducing artefacts. The model was originally developed for use in 2D but, owing to its mathematical simplicity, extension to 3D is simple and straightforward.

In the case of triangulated 3D data, there has been some work on smooth warping without feature detection. Kanai et al. [KaSK00] have shown that such warping is possible for open surfaces with a single closed boundary by means of generating a harmonic mapping, then carrying out the warp in 2D parameter space. Kuebler et al. [NFSK95] have presented a snake-oriented approach called 'velcro-surfaces', where they focus on morphing an initial template onto a data set in a manner that is not affected by the initial state of the model.

With respect to human bodies, there has also been significant work. Thalmann and Thalmann [NeTh98] have been working on morphing surfaces (B-splines, metaballs, etc.) to animate human models for a number of years. Blanz and Vetter [BlVe99] have shown that it is possible to deform a 3D template of a human face in order to reconstruct the 3D face from a single 2D photograph. This is an interesting application that, unfortunately, cannot be easily extended for use with whole bodies owing to the very high level of self-occlusion of the whole body. However Ong et al. [OnGo99] have developed a technique for reconstructing the posture of a 3D skeleton by analysing the deformations in the shape of a human silhouette, provided that certain constraints are met.

The concept of Active Shape Models (ASMs) has originally been developed by Cootes and Taylor [CTCG95], in order to overcome the problem of lack of model specificity when using Flexible Templates, the then current method for modelling objects whose appearance can vary. Their work has been inspired by Active Contour Models ('snakes' [MITe95b]), but they have introduced the key difference that an ASM can only deform to fit the data in ways consistent with a training set. It has taken a few years to iron out the original design [BrTa98] and finally produce an integrated work [CoTa98] including a full description of the underlying mathematics as well as ways to resolve caveats related to non-linear dependencies. Since the mid-1990s, 2D Active Shape Models have become a very popular tool (see below). Originally, ASMs were based on the identification of point landmarks on the data, but work has been carried out by Baumberg and Hogg [BaHo95] for the development of adaptive eigenshape models, whereby the model is iteratively built for shapes where there are no landmarks available. This work has been further improved by Kotcheff and Taylor [KoTa98], who introduced a direct optimisation technique for varying the pose and parameterisation of each shape in the training set in order to produce the best possible eigenshapes.

The initial stage of building an ASM is the alignment of shapes (each expressed as a set of landmarks). There are various alignment algorithms, but the most popular (owing to its robustness and intuitiveness) is Procrustes alignment, first presented by Gower as a method of comparing different multivariate analyses of the same data [Gowe71] [Gowe75] and then as a method of statistical shape analysis by Goodall [Good91]. The robustness of the Procrustes alignment procedure (named after a villain of the Greek mythology who enjoyed employing perverse methods of making his victims 'fit' the shape and size of his torture bench) is due to an iterative step. During that step, all shapes are first aligned (in a least-squares manner) to a reference frame, usually chosen to be one of the examples of the training set, picked at random, then all aligned shapes are re-aligned to their mean, and the mean, in turn, to the reference. This re-alignment step makes the whole alignment problem mathematically well defined. In addition, closed-form (non-iterative) solutions to the orientation alignment (often required at each step of the Procrustes procedure) have been developed, such as the one by Horn [Horn87]. This solution is especially useful in 3D when obtaining an estimate of the rotation that best aligns two sets of points in a least-squares sense is not entirely trivial.

Given a set of aligned training examples, the next step in building an ASM involves a principal components analysis (PCA) [Joll02] in order to produce a point distribution model (PDM). The average shape is calculated and subsequently the covariance or scatter matrix and its eigenvalues and eigenvectors. Then, by sorting the eigenvalues of the covariance matrix in descending order it is possible to identify the most significant modes of variation in shape [CoTa98] and express each shape in terms of its variation from the mean. However, complications can arise from non-linearities in the shape variation, which tend to produce unreliable and unstable PDMs that do not represent the data accurately and lack specificity. One way to resolve this is to use a hierarchical approach. This approach has been demonstrated by Heap and Hogg [HeHo97], who have shown that the distribution space can be hierarchically segmented so that in each partition a linear point distribution can be found. Also, Ong et al. [OnGo99] are using a hierarchical segmentation of PCA space in order to achieve 2D-3D dynamic human model representations.

There are in addition numerous recent examples of applications of ASMs. Smyth et al. [SmTA96] are using this approach for the measurement of vertebral shape from DXA (dual x-ray absorbtiometry) images. They produce a very robust recognition system using images of low spatial resolution containing significant random and structural noise. Lanitis et al. [LaTC99] used the shape information as well as appearance templates in a combined Active Appearance Model (AAM) to predict facial ageing from 2D photographs. Blanz et al [BlVe99] have used a similar approach in order to fit successfully a morphable 3D model to 2D facial image data. In fact, it has been argued [CoET99] that AAMs perform better than ASMs. This is definitely true [CoET99] for the 2D case, although no known work seems to exist for the performance comparison between 2D AAMs and 3D ASMs. The work of Hutton et al. [HuCH00] on the modelling of dental structures from cephalograms, X-rays and 3D MRI scans for the design of dental prostheses is another good example of the robustness and accuracy of active shape models. Finally, ASMs have been used in the SizeUK project (the UK National Sizing Survey) in order to assess and analyse the size and shape of the UK population [TBRB03].

2.10 Multiple Levels of Detail

In order to achieve computational efficiency as well as satisfactory visualisation of human body models, it is important to consider working with multiple levels of detail. Schroeder [ScZL92] has presented one of the first efficient algorithms for reducing the level of detail of triangle meshes. This has been improved by Hoppe [HDDM93] and further optimised by Heckbert and Garland [HeGa94]. A recent algorithm has also been proposed by Oliveira [OlBu99], specifically with human models in mind. It is based on the work of Hoppe and Garland, but incorporates constraints designed to preserve a good impression of the body's shape and posture even at very low levels of detail. There is also the approach by Brett, Hill and Taylor [BrHT97], who have used LoD reduction as a means of identifying significant feature points on a triangulated surface, and the research by Schroeder et al. [LSSC98] for the development of adaptive parameterisations of a surface at various levels of detail. Finally, Heap and Hogg [HeHo97] have used the LoD approach for the creation of a hierarchical PDM model of shape analysis whereby the point distribution space is segmented so that the variation within each segment may be well approximated by a linear model.

3 Design and Methodology

This chapter describes the proposed algorithm and the reasoning behind its methodology. More specifically, Section 3.1 contains an overview of: (i) the local surface reconstruction and why it is necessary for curvature extraction; (ii) the assumptions regarding the input data and the pre-processing that must be carried out on it for the algorithm to work; and (iii) the step-by-step structure of the proposed algorithm at the highest level. Section 3.2 discusses the issues that need to be addressed throughout the algorithm steps, and a number of crucial decisions that had to be made regarding the design of the method. Finally, section 3.3. illustrates the strategy followed in order to evaluate the quality of the results through each step of the algorithm.

3.1 Local Surface Reconstruction

The idea behind the proposed algorithm is quite simple. To assess the curvature at each point in the given point cloud, the surface is *locally* approximated by an analytic representation or 'surface patch'. The curvature properties of the patch at the point of interest are then calculated analytically and assigned back to the point. The process is repeated for each point in the point cloud. The analytic representation of choice is an implicit surface of the form $F(x,y,z) = 0$. The reason for choosing this representation is that the fitting of such a surface is easy to formulate as a least squares minimization that, for an algebraic distance expression, can be solved as an eigenproblem. This makes the implementation straightforward and also generates additional information that can be used to assess the method's reliability and stability. Moreover, this type of representation allows the method to work on any type of point cloud, regardless of the orientation of the underlying surface. In particular, it may therefore be used on point clouds made up of registered subsets of points from a group of sensors placed all the way around the body. This is an advantage that representations of the form $z = f(x,y)$ do not have, since they cannot represent surfaces where a line of the form $(x=x_c, y = y_c)$ intersects the surface at more than one point (e.g. spheres, ellipsoids, and most other conic sections - Figure 5).

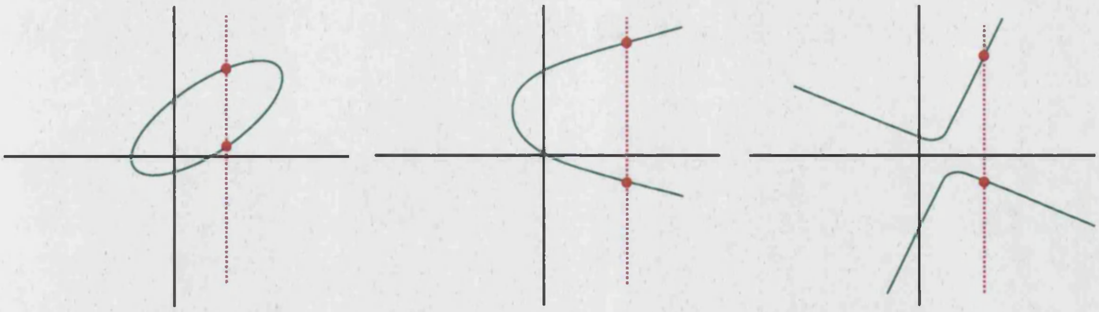


Figure 5: Implicit Functions vs. Height Functions. All three shapes illustrated here in green (From left to right: Ellipse, Parabola, Hyperbola) are conic sections and therefore representable as implicit functions. However, none of them can be represented as a height funtion because there are cases where a line parallel to the height axis (shown here in red in all three cases) intersects the surface at more than one point (i.e. there is more than one "height").

3.1.1 Assumptions

For the proposed algorithm to work, the following assumptions must be valid:

- The input data set is a *Point Cloud*. It is not necessary for the point cloud to be connected, although the connectivity information may be used whenever it is available to speed up the implementation and improve its performance, as described in section 3.1.2 below.
- The surface represented is, at each of its points (locally), either *synclastic*, or *anticlastic*, or *developable* (as defined and discussed in section 4.1).
- The given point cloud is *Reasonably Clean*. It has already been pre-processed to remove points that do not correspond to the original surface (see e.g. Figure 2 – this pre-processing is the step from 2a to 2b). This means that any pre-processing such as outlier removal and segmentation are not within the scope of this project. Such issues are already covered adequately by the design of a good scanner – see e.g. [Hori98][JoBW95][Dekk00]. Noise in the location of the surface points, as long as it is an order of magnitude smaller than the size of the object represented, is accepted. It is assumed that such noise is zero mean, and approximately normally distributed. It is important, however, since the first step is a least-squares fitting procedure that there are no (or very few) outliers in the data.
- The surface must be *Sufficiently Sampled*. The point cloud must be sufficiently dense (i.e. at a density of an order compared to the maximum surface curvature) so that a "small" sphere centred on any point on the sampled surface contains only neighbours of the central point, and does not contain any points that do not belong to the local surface manifold around the point (Figure 6). If the surface is triangulated or otherwise connected, the sphere should contain at least first and second nearest neighbouring points on the surface.
- The point cloud represents a *Significant Coverage* of the original surface. This

affects only the use of the results. No curvature is calculated in areas of the object surface not covered by the point cloud, so if the results are to be used for a matching application, the set of features detected will only exist for the covered areas.

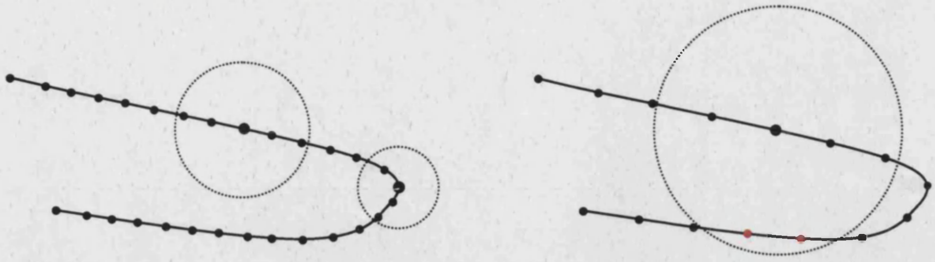


Figure 6: Sufficient vs. Insufficient Sampling. Left: The shape is sufficiently sampled. A circle centered on a sample and large enough to contain the sample's immediate neighbours contains no other samples. Right: The shape is insufficiently sampled. A circle centered on a sample and large enough to contain the sample's immediate neighbours contains also points that do not belong to the neighbourhood (shown in red). These can be filtered out if the ordering of the samples is known (i.e. if they arose from the sampling of a parametric curve or surface). This is impossible if the ordering is unknown (i.e. if they are obtained by scanning a real object).

3.1.2 Structure of the Proposed Algorithm

The steps of the algorithm are as follows:

Step 1: For every point *P* in the set:

Step 1.1: A neighbourhood of points of size *N* around *P* is selected. At this stage, connectivity information, if available, may be utilised in order to reduce the search space and improve computational efficiency.

Step 1.2: An implicit surface patch $F(x,y,z) = 0$ is fitted onto the neighbourhood of points.

Step 1.3: A projection P_0 of point *P* onto the patch is calculated, such that $F(P_0) = 0$.

Step 1.4: The curvature properties of *F* are calculated at P_0 and assigned to *P*.

Step 2: Every point P is appropriately labelled as a ridge, flexcord or umbilic according to the curvature properties assigned to it.

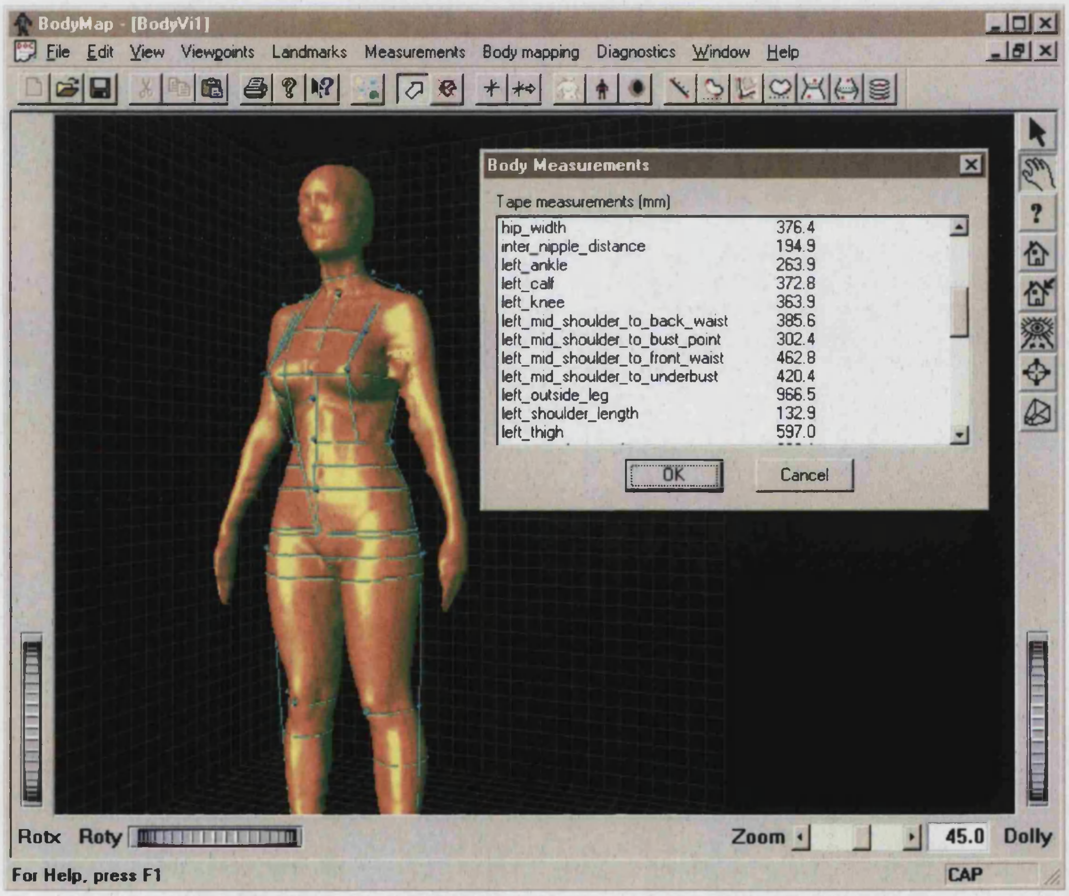
The steps of the algorithm are discussed in detail in the chapters that follow. More specifically: Steps 1.1 and 1.2 are the subject of chapter 4; steps 1.3 and 1.4 are presented in chapter 5; and step 2 is dealt with in chapter 6.

3.2 Issues to be Addressed

There are a number of issues that do not appear directly in the hypothesis, but need however to be addressed in order to devise practicable modelling techniques [BDDV00]. These issues are related both to the pre-processing required for the data to be brought into a form that makes our proposed modelling feasible, and also to the peculiarities introduced as a result of the fact that our data represent real-world, free-form objects. All these issues are discussed in detail in the sections that follow.

3.2.1 Landmark identification

We need to determine what kinds of surface feature need to be detected, and how to detect them. Previous work by Dekker [Dekk00] on the identification of anatomical landmarks as surface features indicates that there are robust ways to find about 100 anatomical landmarks on a human body (Figure 7), at least for specific categories (e.g. adult women of medium build). It is possible to use these landmarks for a classification of body types, for example by extracting contour measurements on the surface, and using their values to determine "sizing range", as is the current standard practice in the clothing industry[7].



Tape measurements (mm)

hip_width	376.4
inter_nipple_distance	194.9
left_ankle	263.9
left_calf	372.8
left_knee	363.9
left_mid_shoulder_to_back_waist	385.6
left_mid_shoulder_to_bust_point	302.4
left_mid_shoulder_to_front_waist	462.8
left_mid_shoulder_to_underbust	420.4
left_outside_leg	966.5
left_shoulder_length	132.9
left_thigh	597.0

Figure 7: Anatomical Landmarks, detected using the software developed by L. Dekker [Dekk00]

---

[7] There are many such sizing range specifications in the clothing industry, and it is practically impossible to single out only a few references to insert here. A fairly comprehensive list of relevant references may be obtained from: http://www.human.cornell.edu/txa/faculty/SizingSystems/references.html

However, these landmarks have a set of significant problems, namely:

(i) They are not particularly robust under surface deformations and, as a result, their successful detection depends heavily on the subject's posture and also on the subject's size and shape (hence the example of "adult women of medium build" above);
(ii) The "sizing range" subclasses described above are historically dictated and therefore, from our point of view, largely arbitrary. In view of this arbitrariness, these anatomical landmarks are not really adequate for shape modelling and classification. They only work under the assumption that there is a quantitative one-to-one relationship between the landmarks, in a manner somewhat akin or equivalent to the notion of body "size". However, recent work (such as SizeUK – the recent UK National Sizing Survey[8]) supports the notion that such a quantitative one-to-one relationship does not exist (i.e. it is possible to have people whose bodies can be characterised by similar body measurements and locations of landmarks in 3D space, yet their body shape types are radically different).

3.2.2 *Level of detail*

The level of modelling detail will be determined in each application by the needs and resources of the application itself. In general, a human torso requires less detail than the head/face areas [OlBu99]. Also, other attributes of the modelling such as number of landmarks, surface deformations, number of iterations for convergence of algorithms, etc.) will depend on the level of detail. It is therefore necessary to determine, with respect to particular application domains, what kind of detail can be considered sufficient. A reasonable approach would be to use a modular model that consists of different sub-models for different parts of the body. Otherwise, at low levels of detail certain features such as, for example, the fingers, become indistinguishable. This is well adapted to intuitive notions of body parts. Another approach would be to incorporate multiple levels of detail and therefore consider a hierarchical modelling structure [OlBu99]. This would be useful in computer graphics applications (see section 2.10 for further details).

3.2.3 *Automatic vs. manual processing:*

It is desirable for any modelling technique to be fully automatic and, in particular, if it involves the use of landmarks, these must be reproducible. Previous works such as that by Hutton [HuCH00] has illustrated that the placement of landmarks on a large number of 3D objects is both tedious and unreliable. Therefore, manual processing of 3D surfaces should be avoided. Some manual stages in the early phases of proposed implementations may be allowed, in order to bootstrap the model and allow quick evaluation. However, those stages should be removed from the final versions of the modelling techniques.

3.2.4 Peculiarities of Human Bodies

Throughout this work, the choices of models, primitives and methods have always had to take into account the non-trivial geometry and topology of the class of objects in question. Human bodies are non-rigid, free-form natural objects that cannot be represented by simple geometric primitives. Moreover, the type of primitives used to describe their shape needs to be carefully considered because of the following two problems.

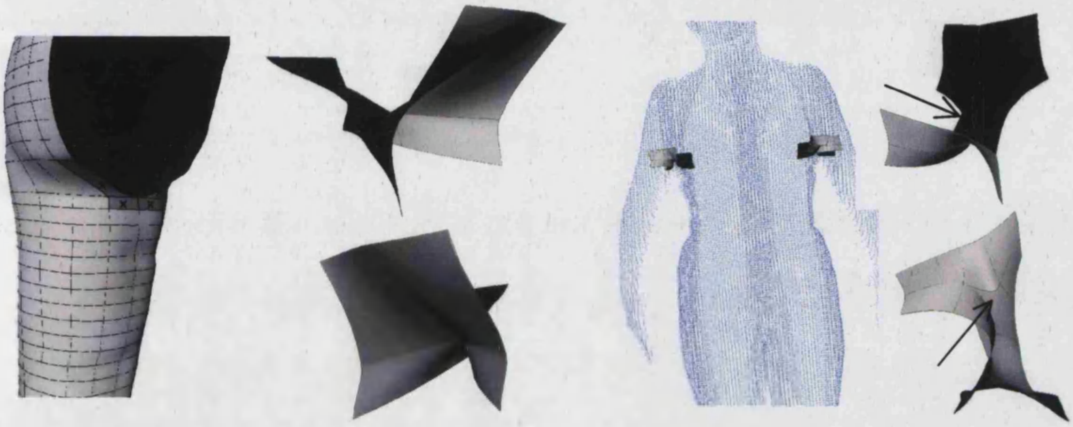
Firstly, the use of height functions has to be ruled out. This is because a human body surface is closed, and its normals can have any orientation. It is therefore essential to choose surface representations that do not have any particular restrictions or preferences regarding the choice and, in particular, the orientation of the frame of reference. This also means that any surface modelling methods based on height functions and used for modelling parts of the body that have the topology of a deformed sheet (such as faces [KLML96][HuBH01]), are not suitable for whole bodies.

Secondly, although it is possible to model individual parts of the body (such as torsos or legs [JoBW95]) as deformed cylinders, this is not the case for a whole body. This is because the topology of the body contains branches, which are in practice also mutually occluded in many scanner systems. Moreover, at the branching points where the deformed cylindrical segments meet, the local geometry becomes even more complicated. It includes singularities such as cusps and folds and, quite often, 'unknown' features that are not represented in the scan data, simply because they are not visible to any of the scanner's sensors (e.g. in the armpits). It is possible to use

---

[8] At the time of writing, no results of this Survey have been made available to the general public. They are, however, available to UCL because of its involvement in the exercise.

relatively simple geometric primitives to model these regions sufficiently for visualisation purposes, as shown by [Dekk00] and [DoDB99b]. However, this normally incurs the cost of sacrificing the global integrity of the surface model. It may require either self-intersections of surface primitives (which, strictly speaking, is incorrect since the human skin is a surface that does not self-intersect, Figure 8), or arbitrary placement of primitives that overlap in order to give the visual impression of a continuous surface (but do not conform to any acceptable definition [OlBu99] of consistent surface modelling, Figure 9). In both cases, the result is a whole body surface model that cannot reliably be used by subsequent applications, e.g. for measurement extraction or accurate landmark identification.



(a) (b) (c) (d)

Figure 8: Modelling of the leg and armpit regions of a whole body scan using B-spline patches that intersect [DoDb99b]. (a) Detail of the left leg and left half of the torso to show the connectivity. The four marked patches are those immediately adjacent to the groin branching point. (b) Two different views of the set of patches that are immediately adjacent to the groin branching point. The set consists of four patches from each side of the body and is viewed from a front-right (top) and a front-left (bottom) angle. (c) The point cloud of a scan together with the two patch sets at the armpit. (d) The patch set for the right armpit viewed from above (top) and from below (bottom). The patch indicated by the arrow on the top right and bottom right pictures is a special patch that extends from the front side of the armpit to the rear in a lateral manner, and is used to close the visual gap that would otherwise occur.

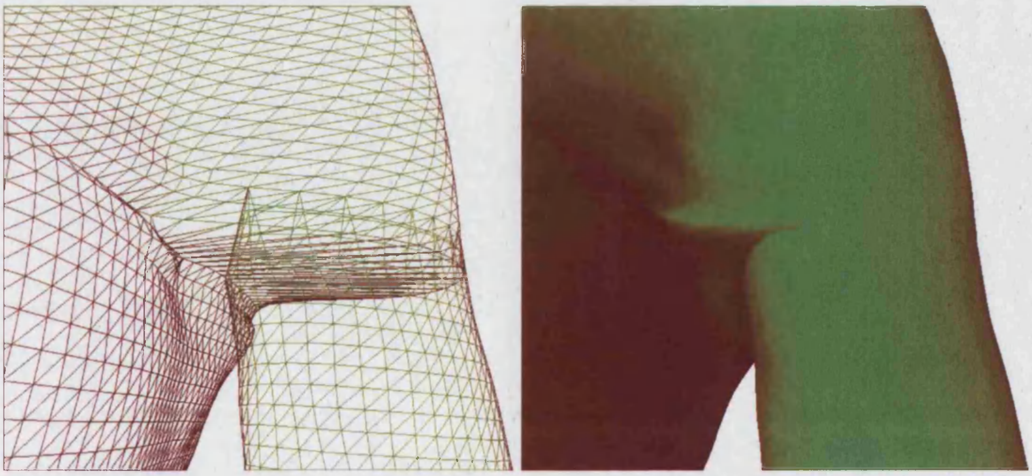


Figure 9: Modelling of the armpit region of a whole body scan using triangulated meshes with vertex replication [Dekk00].Left: wireframe view. Right: solid triangle rendering using Phong shading.

Finally, a note must be made here regarding body surface cusps (i.e. skin folds). In this work, cusps are features that we do not attempt to model. The primary reason is that we assume that we do not always have the connectivity information that would allow us to detect them reliably (e.g. by thresholding the angles between neighbouring polygons). The second reason is that the early stages of the algorithm involve noise reduction by Gaussian smoothing, whereby such sharp features are anyway lost.

3.2.5 Invariance

At the early stages of the research, it was deemed desirable to develop techniques that would detect on a 3D body surface features that would be invariant to a number of deformations. However, since we primarily envisage application of the features detected to the matching of body scans, this notion can be refined somewhat.

Instead of distinguishing between invariant and non-invariant features, it makes more sense to distinguish between features that are *reformable*, and features that are *deformable* under deformations of the surface. Previous work by Porteous [Port88][Port94][PoPu00], and Koenderink [Koen09][KoVD98] has shown that features such as lines of curvature and geodesics have no individual identity if the surface on which they lie deforms in a continuous manner. By this, they mean that we cannot mark any such line and then study its development in time as the surface deforms - at each moment one has to reintegrate the differential equations that these lines satisfy, and obtain a completely new configuration. On the other hand, parabolic lines, as well as ridges and flexcords, do deform when the surface is deformed and, as Porteous notes [PoPu00, also pers. comm.], "...suffer perestroikas, and even births and deaths. Umbilic points also exhibit the same type of behaviour".

Of these features just mentioned, umbilics and parabolic lines are well covered in the literature (e.g. by Porteous[Port94], Giblin [BrGT96], Gasson [Gass83], Tricoche [TrSH00], Csakany [CzWa00], Fisher [TrFi95], Sander [SaZu92], Gordon [Gord91], Koenderink [Koen90][KoVD94][KoVD98], Linney [KLMK96] and Ponce [PaPo01] [FoPo01]), but ridges and especially flexcords have not been studied very widely in comparison [Port88][Koen90][Port94][Morr96][BePK98][PoPu00], although they codify important features of the geometry of the deforming surface.

It is therefore desirable to achieve a robust and reliable method of detecting these *deformable* features. The easy detection of such features would facilitate the study of their behaviour on a deforming surface or on a set of surfaces, thus constituting an additional application of this research to the detection itself.

3.2.6 Local vs. Whole Surface Reconstruction

This work is concerned with extracting meaningful information from available data. In this framework, no attempt is made to reconstruct complete whole body surfaces from body scans. The local surface reconstruction method described in chapter 4 is used strictly for the assessment of the geometry of the surface around each point in the original data set. We are trying to follow the general principle of not attempting to interpolate in order to generate (assume) points for the parts of body surface that were not delivered by the scanner. We try to work only with the parts of body surface that are included in the data set, rather than introduce a set of arbitrary assumptions in order to generate additional points of questionable reliability. Such points could, for example, later cause the introduction of spurious small-scale features which may themselves be unreliable (Figure 10).

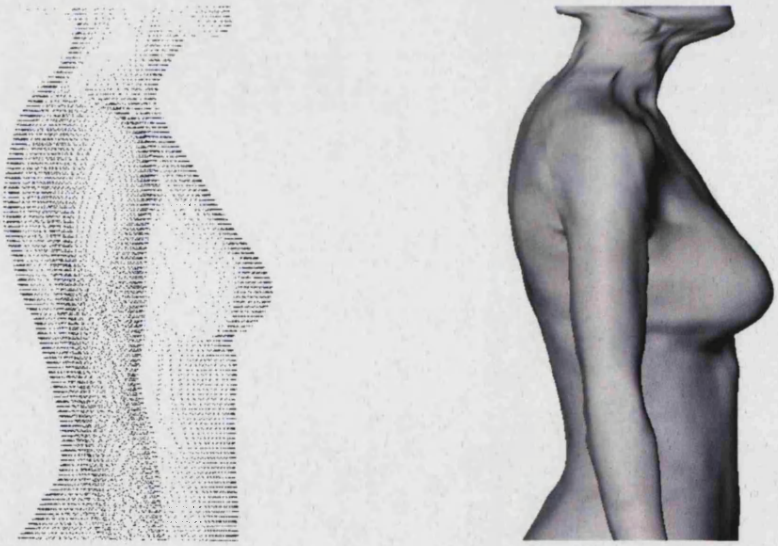


Figure 10: Examples of whole body surface reconstruction where interpolation has been used to cover gaps in the original data. The gaps occur at the side of the Torso.Left: Original point cloud (detail view of the torso, upper arm and neck segments), shown from a side view that illustrates the gaps in the sampling. Right. The same segment with a B-Spline surface fitted to it. Note the spurious features (ripples) arising at the gaps where there is insufficient data to drive the spline interpolation effectively.

Note, however, that it is possible to combine patches resulting from a local surface reconstruction step in order to reconstruct larger surface segments covering ‘continuous’ subsets of the original data (i.e. regions of the surface that are adequately sampled throughout). This blending method, recently presented by Ohtake et al. [OBAT03] has not been implemented within the context of this research as it is well beyond the scope of this thesis. However, it implies that if a body scan provides sufficient coverage (in the sense that there is sufficient overlap between the locally reconstructed patches of all points in the original data), it may then be possible to construct a closed implicit surface representation that covers the whole body.

3.2.7 Use of Connectivity Information

One of our assumptions (section 3.1.1) is that connectivity information is not always available in our original point cloud. In fact, our proposed algorithm in its general formulation does not use such connectivity information directly. Explicit connectivity is sacrificed for the sake of generality, and with little loss in performance.

It is shown later (section 4.1.1), that it is more important to establish a *neighbourhood* of points around each member of the original data set, rather than a strict *connectivity* for the algorithm to work. In this context it becomes obvious that the use of any organizational information (full triangulation or even partial polygonization of some form, e.g as in Figure 2c) is beneficial for the performance of the algorithm, as it facilitates efficient search for neighbours. It is also beneficial for the reliability of the results. Noise and errors in data capture may mean point-to-point distance may not always be sufficient for the purposes of estimating neightbouhoods

3.3 Evaluation of the Results

This section describes the strategies that will be followed in order to evaluate the results. More specifically, the choice of test objects is explained, followed by an overview of the criteria for evaluating the results on scan data from real human bodies.

3.3.1 Test Objects

Throughout the steps of the algorithm, we use a set of test objects in order to evaluate the *accuracy* of the results (Figure 11). These are relatively simple objects and have a known analytic (implicit function) representation and, in many cases, known locations of the differential geometry features that we are trying to detect, especially the ridges and ordinary umbilics.

It is therefore easy and straightforward to generate sampled point sets of these objects with a structure resembling that of our available body scans (i.e. sliced, and not necessarily uniform). We then process these virtual 'scans' of the test objects using the implementation of our proposed algorithm and obtain the curvature properties and differential geometry features. These are compared with the curvature properties (section 5.5.1) and features (section 6.6.3) obtained from the analytic forms of the test objects. These comparisons will comprise the ultimate *accuracy* tests of our proposed method.

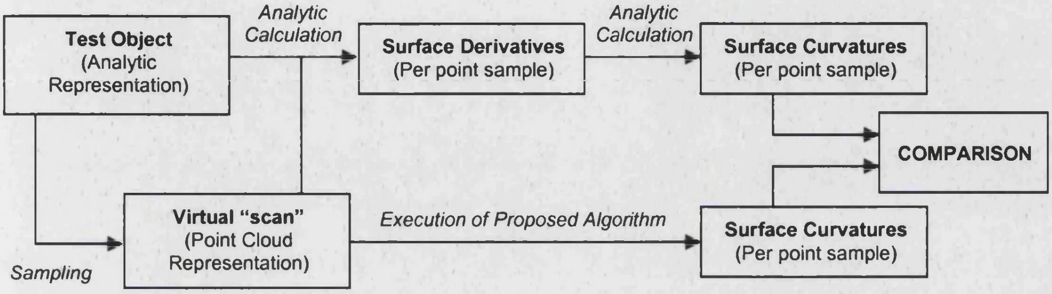


Figure 11: Diagram of the overall evaluation strategy.

In addition, tests are presented that evaluate the accuracy of the local surface reconstruction, the performance of the ridge detection technique, and the behaviour of the proposed scheme with respect to noise and smoothing.

The simple geometric test objects that we will use in the evaluation of the results (Figure 12), all have a common property: they have planar symmetries, therefore the locations of at least some of their ridges are immediately known, since they occur at the intersection of the object with its planes of symmetry [Port94]. In addition, each of those objects has some special properties that are important for our evaluations.

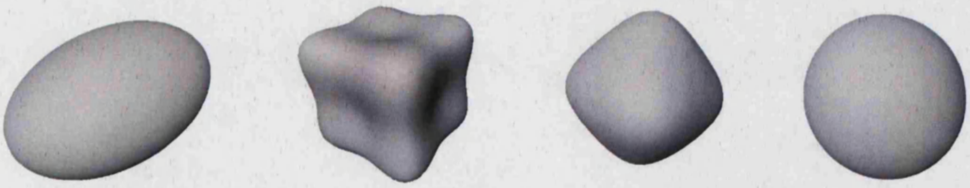


Figure 12: The test objects used for evaluation of the results. Left to right: Ellipsoid, Tooth, Rounded Octahedron, Bumpy Sphere.

The definitions of the test objects, as well as the reasons for choosing each of them, are described below:

- *Ellipsoid*: $$\frac{x^2}{a^2}+\frac{y^2}{b^2}+\frac{z^2}{c^2}-1=0$$

This is a somewhat trivial object, but it is one of the simplest to use as a starting point. It has the important property of belonging to the type of implicit representation we will use, so all errors are observation, sampling and numerical accuracy. There is no approximation error. Moreover, it can be resampled in a manner enabling us to know the exact locations of the points that should be classified as ridges. We can therefore use this type of object in order to check that the evaluation procedure behaves as expected. Also, the locations of the umbilic points on that objects can be analytically calculated.

- *Tooth*: $$x^4+y^4+z^4-x^2-y^2-z^2=0$$

This is a special case of a Goursat surface [Gour97]. It has known planes of symmetry, on which we expect ridges to be found. We are not aware of the existence of ridges elsewhere on the surface. It is believed [Ian Porteous, priv.comm.] that there aren't any, but no proof is known to us. Moreover, this is the only test object with concavities and clearly defined saddles. This is important for assessing the performance of the curvature calculation algorithm, both directly (i.e. by assessing the results for regions of negative as well as positive curvature) and indirectly (i.e. by evaluating the results of the ridge detection, which in turn relies on correct curvature calculation).

- *Rounded octahedron*: $$x^4+y^4+z^4+5x^2y^2+5y^2z^2+5x^2z^2-1=0$$

For this particular surface, the locations of all ridges are known [BePK98]. Moreover, all the maximal ridges (i.e. the ones of most interest, as explained below) are on the intersections of the object with its planes of symmetry, making the evaluation very straightforward.

- *Bumpy sphere*[9]: $$x^2+y^2+z^2+\frac{1}{3}\varepsilon xyz-1=0$$

This is essentially a sphere with an additional term that introduces bump features on the

---

[9] This particular case of bumpy sphere is often referred to (mistakenly) as a "bumpy cube", but actually it has tetrahedral symmetry [Port94].

surface. The intensity of the bumps is controlled by the value of parameter ε, which is positive and typically very small (hence the image of the bumpy sphere in Figure 12 does not appear to differ much from an ordinary sphere). For this object, the locations of all ridges are known, and given in [Port94] (Figure 13). Some of them are small closed loops around its umbilic points and we have no feasible means of evaluating detection and locations of them (this particular feature on a bumpy sphere, and its implications for our results are discussed in detail in section 6.6). However, all remaining ridges are on the object's planes of symmetry, again making evaluation straightforward.

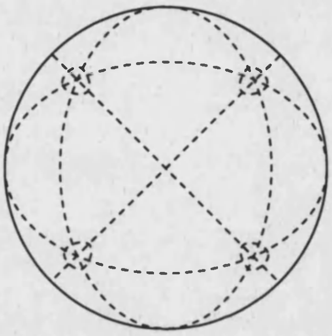


Figure 13: Ridges on a bumpy sphere (reproduced from [Port94]), shown here as dashed lines.

The manner in which these test shapes were used was by means of sampling the implicit functions presented above in order to produce point clouds that resembled, both in density and structure, the body scans available for our research. Therefore, for each of the test shapes, these steps were followed:

Step 1: The implicit function was expressed in polar co-ordinates (more specifically, cylindrical for the ellipsoid and the rounded octahedron, and spherical for the tooth and the bumpy sphere). This produced a parameterisation of the form $r = (\theta, z)$ for the former two, and $r = (\theta, \varphi)$ for the latter two.

Step 2: A set of samples were taken at regular grid intervals of the parametric space, so as to generate a quadmesh of points expressed in polar co-ordinates.

Step 3: The polar co-ordinates of the sample were translated back to Cartesian co-ordinates.

Step 4: The set of Cartesian co-ordinates was uniformly scaled so that the set would represent an object of 1m height.

An important note regarding Step 2 above is that the density of the sampling in the parametric space was chosen so that the samples would have an average resolution of 5-10mm *after* the scaling in Step 4.

This method of sampling generated four "virtual scans" (point clouds) with a resolution comparable to that of the real human body scans available for this research, with a similar "horizontally sliced" space arrangement, and with similarly convenient and rich (quadmesh) connectivity that would allow them to be processed by our implemented software just as if they were human body scans.

In addition to the objects described above, we also use some sampled quadric surfaces of various types (hyperboloid, paraboloid, ellipsoid), in order to carry out the early assessment of our local surface reconstruction results. These tests on quadrics are discussed and illustrated in section 4.4.

3.3.2 Reliability Tests for Human Body Scans

The previous section described the use of test objects for the purpose of measuring the *accuracy* of our proposed method. However, it is also essential to show that the method demonstrates sufficient *reliability* when used on scans of human bodies.

Unfortunately, human bodies are free-form surfaces. Moreover, it is not practical within the scope of this research to use real human subjects as validation objects, i.e. to measure their surface curvature at, say, a few thousand sample points by an independent method and then compare against the results of our implementation. Although it is possible to use a phantom instead, e.g. a mannequin such as the ones used for garment display at retail outlets, the implementation of any of the known techniques for measuring the curvature on the phantom surface (such as Chebyshev Nets, first introduced in 1878 [Tsch78] and recently re-introduced by Koenderink in a computer vision context [KoVD98b]) is beyond the scope and resources available for this research.

Therefore, it is not possible within the scope of this work to assess the accuracy of our method in a *numerical* and *quantitative* manner, such as for the test objects. Instead, the criteria will have to be mostly visual and qualitative, although section 7.4.4 indicates some directions that could be taken for evaluating reliability quantitatively within the scope of further work such as surface matching.

The criteria for evaluating reliability may then be formulated, as follows:

(i) The method must produce results from scans of subjects of *both genders* and of *all ages*, as long as the scans conform to the assumptions set out in section 3.1.1.
(ii) The method must be able to *detect*, wherever possible, any *invalid results* and notify the user accordingly.
(iii) The method must be *suitable for matching* and shape *classification*, i.e. it must produce results that are:
    a. *similar*, for scans of subjects of the same gender, body type, age and posture, and
    b. *distinctly different* otherwise.

The above criteria are qualitative, and will be illustrated by visual examples wherever appropriate throughout this thesis. It must be noted, however, that there is some scope in them for quantitative evaluation in the future. For instance, criterion (i) could be evaluated by tests on a few thousand scans, should they become available, rather than on only the approximately 100 scans on which the method was tested throughout this work. Alternatively, measures of similarity for criterion (iii)-a could be defined if an appropriate matching strategy becomes available in the future.

4 Local Surface Reconstruction

This chapter presents details of the design, implementation and results of steps 1.1, 1.2 and 1.3 of the proposed algorithm (as defined in section 3.1.2). More specifically, section 4.1 provides a description of the local surface reconstruction problem and of the surface primitives used and section 4.2 presents the formulation and solution of a least-squares error minimization problem for implicit surface fitting. The results are given in section 4.3.

4.1 Approximation of the Local Surface Shape

The scanner delivers a set of data points $(x(i),y(i),z(i))$ for $i=1,\ldots,N$, say, in a cloud of data points sampled from the body surface. Our aim is to find the form of the surface within some small neighbourhood $N(P)$ of a point P on the surface and hence, for example, to estimate the curvature there. To do so, we choose a local analytic approximation to the surface and fit the data points in $N(P)$.

It is important that the representation of the analytic approximation used does *not* single out any coordinate axis. There are thus two choices (see, for example, [Krey59]):

i. An implicit function: $F(x,y,z) = 0$ (4.1)

ii. A parametric representation:

$$\begin{aligned} x &= x(u,v) \\ y &= y(u,v) \\ z &= z(u,v) \end{aligned} \tag{4.2}$$

Although the parametric form (4.2) is usually preferred in differential geometry, here it presents the additional problem of finding the surface parameters $(u(i), v(i))$ of each point i in $N(P)$ [Wats00].

There is no such problem with the implicit function F in (4.1) and, indeed, this type of representation seems to have been used quite often in fitting to range data from manufactured objects. In fact, quite often, quadric, superquadric and hyperquadric forms were used in such works [Book79][Samp82][GaGS94][KHGB95][CoCo96][FiFE97][WeFR98][LUMM98][BLCC00]. Additionally, implicit functions of this form are ideally suited to our problem, since their form provides a direct error metric that allows the formulation of a minimization problem.

At this point, the next decision to be made is on the type of implicit function to choose for describing local shape. The factors to take into consideration here are:

- The types of local shape that we need to describe: The functions chosen need to be able to cover the sorts of shape locally found on a human body. More specifically, we need a function that can describe local surface shape that can be elliptic (synclastic), parabolic (developable), or hyperbolic (anticlastic) [Gass83]. Moreover, we do not know in advance which of these three forms (Figure 14) corresponds to each region of our initial data, therefore they must all be covered by the same function so that the type can be inferred by the solution of our local reconstruction problem.
- The surface properties and features that we eventually wish to detect: We are going to attempt to calculate curvature properties and detect features (such as ridges, flexcords and umbilic points) on the locally fitted patches, and then assign these to the original surface. Therefore, *F* must be chosen such that it can exhibit these types of feature.

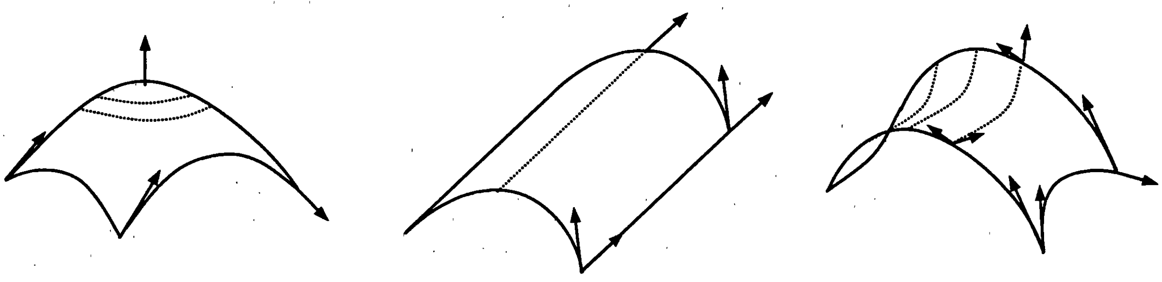


Figure 14: Forms of synclastic,developable and anticlastic shapes, reproduced from [Gass83] (ordered from left to right).

It is clear from the requirements above that a polynomial form of second degree is most suited to our problem. Though they are not generic, regular surfaces as defined by Porteous [Port94], second order polynomials are most convenient as they exhibit the umbilic points and ridge lines that are of most interest to us. Any lower degree would make *F* unsuitable as it would only allow fitting of planar regions. Any higher degree, and the variety of surface forms that could be generated by a general third order form becomes too great for the fitting of a patch over a small neighbourhood of points to be likely to be reliable. Use of the Monge form would circumvent this difficulty [Krey59], but would entail us inventing a type of height function, even if only locally [CaPo03]. It is interesting to note [CaPo03] that a variety of quadric surfaces can have the same curvature at the same point, so provided that we obtain accurate values for the curvature properties, obtaining the "correct" type of quadric is not an issue.

It is shown, in fact, by Cazals and Pouget [CaPo03], that if the surface primitives used are quadric (i.e. of second degree), then there is no loss of information compared to the use of Monge patches (height functions): all possible shape forms described by using a quadric height function can also be represented by an implicit quadric. Also, implicit functions have the additional advantage of coordinate independence over height functions. By choosing implicit quadrics we lose the 3rd-order derivatives typically used for calculating ridges and similar differential geometry properties and features from the Monge form. However, this is a reasonable trade-off because we can calculate these properties using different methods, as illustrated in the next chapter.

Our requirements are thus met by the quadric function of the form:

$$F(x,y,z) = ax^2 + by^2 + cz^2 + 2exy + 2fyz + 2gzx + 2lx + 2my + 2nz + h = 0 \qquad (4.3)$$

and the local surface reconstruction problem is formulated as follows:

Given: a continuous surface B_s represented as a (sample) point cloud P_s,
Then: for every point P in the point cloud we need to find the values of parameters a, b, c, e, f, g, l, m, n and h of a quadric surface function F that best describes the shape of B_s within a small neigbourhood $N(P)$ of P.

4.1.1 Selection of Neighbourhood

In the above formulation of the problem, it is important to define carefully the *neighbourhood* $N(P)$ around P. What we essentially need is a subset of the point cloud P_s that complies with the following requirements:

- Every point P_i in $N(P)$ is within a close distance from P.
- There are sufficient points in $N(P)$ for the surface reconstruction problem to be well-

defined in a mathematical sense (i.e. not underdetermined).

- The configuration of the members of *N(P)* allows a quadric surface to be fitted to the data that describes the local surface shape *reliably*, i.e. it is not adversely affected by those points furthest away from the surface that are potentially less significant regarding the local shape around *P*, nor is the configuration degenerate in any way that would leave the fit underdetermined, or ill-posed, or ill-defined.

These requirements are essential, since the method for choosing the neighbours for fitting purposes affects both the *reliability* (and therefore the *accuracy*) of the fit, and the *computational complexity* of the overall implementation. Both these factors are important and strongly linked, since we are seeking a solution that is both practicable and useful. There are a number of issues arising from these requirements that affect the choices of implementation of the algorithm, especially with regard to the use of any connectivity information available. The first requirement implies that the collection of neighbours around *P* could be carried out using a distance threshold, say d_S such that:

$$|P_i - P| < d_S \quad . \tag{4.4}$$

In that case, the *N(P)* would consist of the points that fall within a sphere of radius d_S centered at *P* (Figure 15).

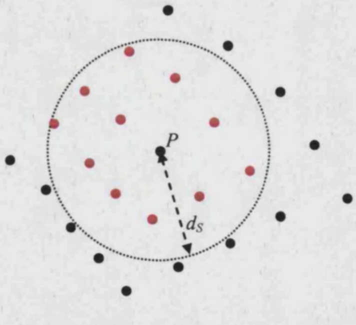


Figure 15: Neighbour collection using a simple Euclidean distance threshold. Collected neighbours are shown in red.

The value of d_S, however, is dictated by the other two requirements. If d_S is too small, then too few points will fall within the catchment area of the sphere, and the problem becomes under-determined (there are 10 unknowns, namely the coefficients of the quadric in a homogeneous function, so any neighbourhood with less than 9 members is unlikely to return well-defined results). Additionally, too few neighbours may produce unstable results where there are significant levels of noise, especially when the region around *P* is almost planar.

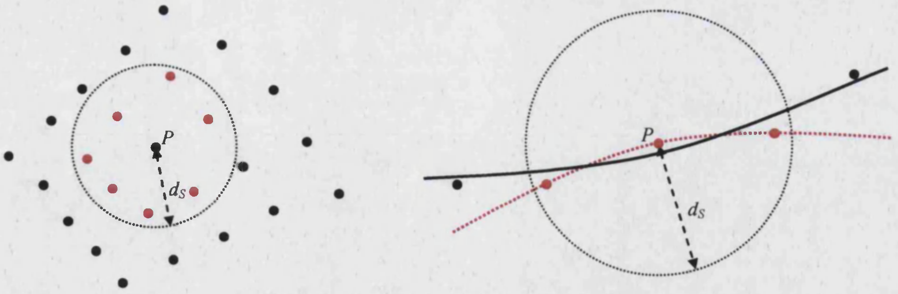


Figure 16: Problems with neighbour collection when the radius is too small. Left: Too few neighbours cause the surface fitting problem to be underdetermined. Right: On a sampled surface where there is noise in the samples, the neighbours collected are insufficient for the purposes of describing the local surface shape accurately (black solid line: original surface from which the -noisy- samples were generated;red dotted line: the surface eventually fitted based on the neighbours collected.

On the other hand, if d_S is too large, a number of errors may occur. Firstly, it is likely that points may be collected that belong to a different sheet of B_s and are not really neighbours of P conceptually as they do not contribute to the local shape around it (recall Figure 6). For instance, if P is a point on the side of the lower torso, and d_S is comparable to the distance between hands and torso for the particular scan, points may be thus collected from the hand (Figure 17, left). In this case, it is easy to filter out such points if there is some connectivity information provided along with P: if the shortest path length value from P_i to P (i.e. the minimum number of edges that need to be traversed to get from one point to the other) is too high, then P_i can be excluded from the collection even if (4.4) holds true.

Most importantly, however, if d_S is too large, it is possible for neighbours of the same sheet to be included in *N(P)* that deviate from the strictly local shape profile, as shown in Figure 17, right. The way around this problem depends largely on the presence of connectivity information in the original data.

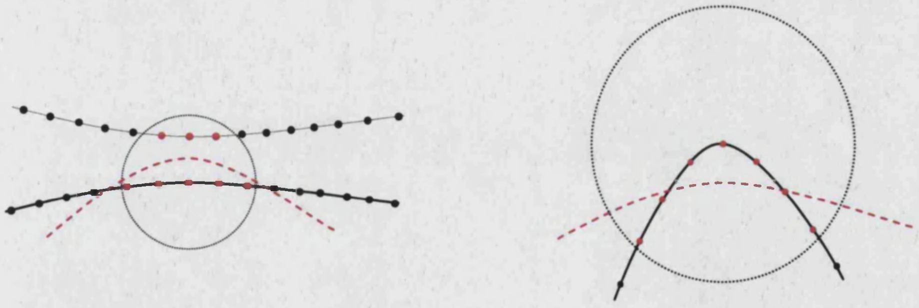


Figure 17: Problems with neighbour collection when the radius is too large. Left: collection of a point from a different sheet; Right: collection of too many neighbours from the same sheet, which do not contribute to the local shape. In both cases: Red dashed line: Fitted surface. Black solid line: Desired result. Collected neighbours are shown in red.

If the connectivity between the points in P_s is given, we can then impose an additional threshold, say l_P, this time on the path length between P_i and P. Experimentation with various values of l_P has indicated that a value of 2 is usually the most meaningful: it allows the collection of only the “immediate neighbours” of P and the “immediate neighbours of the immediate neighbours” of P. These are normally sufficient for capturing the second degree shape profile around P. Figure 18 illustrates the concept on the assumption that P is locally a quadmesh with 8-connectivity. Note that even if a path length threshold l_P is used, it makes sense to keep using the distance threshold d_S as well. There are cases of “jagged” polygons (see, for example, the bottom right hand side of Figure 18) where an “immediate” neighbour is too far from P, therefore unlikely to contribute to its local surface profile. In that case, a value of:

$$d_S = l_P \cdot \overline{d_P} \tag{4.5}$$

where, as explained, $l_P = 2$ and $\overline{d_P}$ is the average edge length, is usually sufficient (because it implies a maximum of 24 neighbours to be collected). Note that the average path length can be calculated globally, i.e. over the whole of P, or somewhat more adaptively over the points that fall within an initial estimated value of d_S.

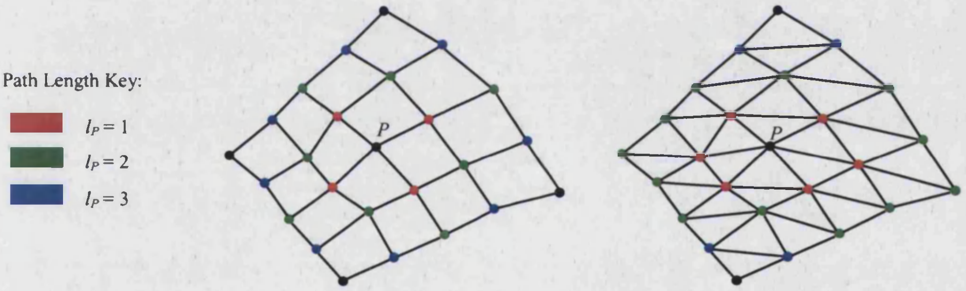


Figure 18: Neighbour collection using a path length threshold. Neighbours are coloured according to their path length from P. The same mesh is shown with two different connectivity structures, to illustrate how connectivity affects the result of neighbour selection.

If the connectivity between the points in P_s is not given, it is not possible to calculate $\overline{d_P}$ directly as there are no “edges” as such. However, it is possible to work out a rough estimate. For instance, if the resolution of the scanner used for obtaining the point cloud is known, it is reasonable to use this as an estimated value of $\overline{d_P}$. This is because the resolution is normally the distance between two *successive* samples of the surface as obtained by the hardware. It is therefore conceptually equivalent to edge length.

Estimating $\overline{d_P}$ introduces a set of issues related to *computational complexity*: Even if the resolution of the scanner is not known, and cannot be easily guessed by inspecting the point cloud, providing an estimate for $\overline{d_P}$ may still be possible albeit neither straightforward nor computationally efficient. It involves calculation of all distances between all[10] members of the point cloud P_s, and generation of a histogram. On this histogram, the first (leftmost) peak should correspond to the distances between successive samples (any other peaks would correspond to distances that are longer and therefore unlikely in turn to correspond to successive sample points). Assuming the most simple (albeit naïve) implementation, without any attempts to optimise performance by using some suitable data structure, this operation would have a complexity of $O(N^2)$. Given that N is typically in the order of a few thousands for a typical whole body scan, it becomes clear that such an implementation should not be attempted. Instead, it is appropriate in that case to implement one of the more sophisticated and efficient space and point set partitioning methods such as octrees [JaTa80][Glas84] or KD trees[Sam90a][Sam90b].

Therefore, the use of any available organizational information is beneficial to the overall computational performance. Firstly, it enables us to avoid an exhaustive search for neighbours of *P* throughout the entire point cloud. This implies that, in case of nothing else being available, even a once-only (O(N)) clustering of the points of the cloud according to the value of one co-ordinate would accelerate the search significantly by narrowing the search to neighbouring clusters, thus, bringing the overall complexity down to O(Nlog(N)). Secondly, utilisation of organizational information enables us to optimise the performance of the search for an estimated value for the "average edge length" mentioned above.

However, we emphasize that it is not necessary for connectivity or other organizational information to be present for the proposed algorithm to work. As long as a neighbourhood has been established around every point *P* in the initial point set, the subsequent steps are independent of the method employed for establishing that neighbourhood.

Finally, it is possible further to optimise the reliability (accuracy) of the search by limiting it to an elliptic, rather than spherical, region around *P*. This is carried out as an additional step on the selected neighbourhood. Principal Components Analysis is carried out on the initially selected points, and the resulting principal axes are used to

---

[10] In fact, not necessarily all of them, as we will shortly show.

determine the size and orientation of the ellipsoid (as shown in Figure 19). The purpose of such an improvement would be twofold: (i) to provide an initial estimate for the normal vector and tangent plane at P, and (ii) to assess whether the spatial distribution of the neighbours indeed resembles a sampled-surface arrangement. The latter may be assessed by checking that the lowest of the three eigenvalues returned by the PCA is significantly smaller than the other two, which are in turn fairly similar [Pear01].

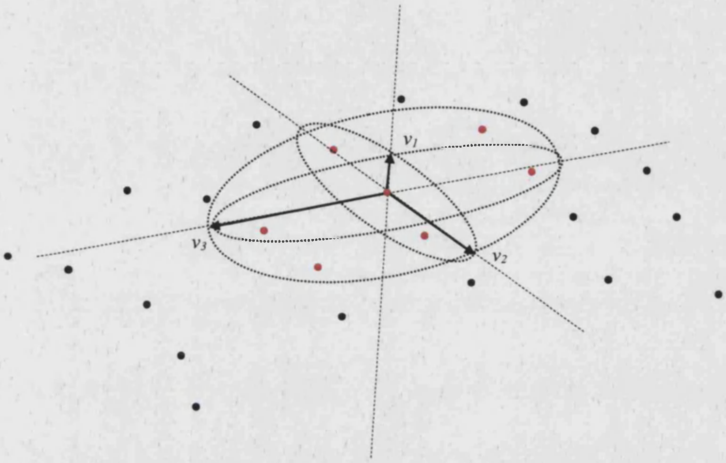


Figure 19: Improvement of neighbourhood collection using PCA.The smallest eigenvector (v_1) provides an estimate of the surface normal at P, whilst the other two define an estimate for the tangent plane.

However, for adequately sampled scans, such as the majority of those used throughout this research, this improvement step makes little difference and appears not to be worth the additional computation. Moreover, in regions of higher curvature such an operation is likely to remove neighbours that implicitly contain that information, thus resulting in the fitting of a flatter quadric patch than if a spherical neighbourhood were used, and in a subsequent underestimation of the local curvature value.

4.2 Minimization Problem

If there are enough points in the neighbourhood $N(P)$, then each datum (x,y,z) in $N(P)$ will not lie perfectly on the surface F, i.e. we shall have:

$$ax^2 + by^2 + cz^2 + 2exy + 2fyz + 2gzx + 2lx + 2my + 2nz + h = \varepsilon \quad . \tag{4.6}$$

Although this is *not* a measure of (Euclidean) distance from the surface (the choice of algebraic vs. geometric error is discussed in detail in section 4.2.1), if we have enough data points (x,y,z) in N then we want to minimize the error ε (known as the algebraic error):

$$\min\left\{\sum_{x,y,z\in N}\varepsilon^2\right\} \quad , \tag{4.7}$$

i.e:

$$\min\left\{\sum_{x,y,z\in N}F^2(x,y,z)\right\} \quad . \tag{4.8}$$

By using representation (4.3), it is possible to express this minimization as a 10x10 eigenproblem. This is particularly straightforward, as shown in appendix B1, if we set a constraint on the coefficients of the homogeneous equation (4.3) of the form:

$$a^2 + b^2 + c^2 + h^2 + (2e)^2 + (2f)^2 + (2g)^2 + (2l)^2 + (2m)^2 + (2n)^2 = 1 \ . \tag{4.9}$$

The approach is similar to that detailed in many articles on the use of implicit functions in surface fitting and interpolation (see, for example, [BaIW93]), and may be traced back to the work of Pearson (1901 - [Pear01]) on the fitting of planar surfaces. A similar approach was presented by Fitzgibbon et al [FiPF99] for ellipse-specific fitting, which is in turn based on the work of Bookstein [Book79]. Since (4.3) is homogeneous, a constraint is essential in order to to make the problem well-defined and avoid the trivial solution $F = 0$. The choice of normalisation (4.9) is convenient and, as noted in [BaIW93] does not introduce any singularities. Yet, it may not be the best. For example, for a plane of the form:

$$2lx + 2my + 2nz + h = \varepsilon \tag{4.10}$$

it is known that the normalisation condition:

$$(2l)^2 + (2m)^2 + (2n)^2 = 1 \tag{4.11}$$

leads to a minimization problem that is unbiased and equivalent to the minimization of the Euclidean or orthogonal error [Pear01]. This, for example, is frequently used in computer graphics for Level-of-Detail problems [HeGa94].

The same result may be obtained by use of a constraint on all the coefficients as in (4.9) if we shift the data points $N(P)$ so that the origin of reference coincides with their centroid, which for a plane leads to:

$$h^2 + (2l)^2 + (2m)^2 + (2n)^2 = 1 \tag{4.12}$$

This is in fact the approach used in our implementation. Unfortunately, there is no normalisation for higher order fitting which has similar statistical and geometric properties to that used by Pearson for a plane and, as noted by Bajaj et al. [BaIW93], the problem of determining a quadratic normalisation with "good properties" remains open. However, provided that a few standard [HaZi00], simple precautions are taken such as choosing a frame of reference with origin at the centroid of each point neighbourhood, use of (4.9) has proven satisfactory in practice, as shown later, in chapter 5.

Given normalisation (4.9), the minimization of (4.8) with quadric form (4.6) reduces to a 10x10 real symmetric eigenproblem of the form:

$$A\underline{a} = \lambda\underline{a} \tag{4.13}$$

In (4.13), A is the *scatter matrix*:

$$A = D^T D \tag{4.14}$$

and D is the nx10 *design matrix* (n is the number of input points, i.e. the number of points in the neighbourhood), each row of which is built up by the following elements:

$$\begin{bmatrix} x_i^2 & y_i^2 & z_i^2 & x_i y_i & y_i z_i & x_i z_i & x_i & y_i & z_i & 1 \end{bmatrix}, \text{ where } i = 1,\ldots,\text{n} \tag{4.15}$$

Matrix A is real and symmetric. It therefore has 10 real eigenvalues, and 10 corresponding mutually orthogonal eigenvectors may always be found. Furthermore, since the error (4.8) is positive semidefinite, the eigenvalues of A are non-negative and the eigensolutions may be labelled in increasing order commencing with the smallest. The coefficients of the eigenvector corresponding to the *smallest* eigenvalue λ_1 are then the coefficients *a, b, c, e, f, g, l, m, n, h,* of the quadric surface that best approximates the point cloud locally around P. This is because the value of λ_1 is itself the value of the sum of squared errors (4.7). The value of λ_1 can be used to assess the quality of fit and, additionally, by sorting the eigenvectors of A according to the magnitude of the corresponding eigenvalue, we also acquire the nine next best fits to the point cloud (for other types of quadric surface).

4.2.1 Orthogonal vs. Algebraic Error

The solution described above is based on minimizing the *algebraic* distance ε between a quadric surface and a set of fixed points. It must be noted that this distance does not have an intrinsic geometric significance [ARCW02]. Also, it is not invariant to coordinate transformations [SuSP94]. This is why it is always important to use the centroid of the data points as the origin of the local co-ordinate system in order to reduce bias.

The obvious alternative would be to minimize instead the mean-*squared geometric (Euclidean) distance* between the points and the surface. However, minimization of the algebraic error is convenient and computationally straightforward, whereas the geometric error has singularities. These singularities are illustrated, for example, by the behaviour of the offset curves of a simple quadratic such as a parabola (see, e.g. [Port94], chapter 1) and are precisely related to the curvature properties that we wish to calculate [BaIW93].

Moreover, some attempts have been made to formulate and solve a minimization problem based on the geometric distance (i.e. minimization of the *orthogonal* error [SuSP94][ARCW02]), but they appear to constitute a minority in the relevant literature on surface fitting. The reason is that, although such methods are deemed to be more appropriate and more accurate – and they certainly produce "intrinsically meaningful results" [ARCW02], their implementations are not free of problems. Most importantly, the geometric distance cannot be computed in a closed form [SuSP94]. This means that there is no simple solution to the orthogonal distance minimization problem. Indeed, it can be seen that the geometric properties of the surface, and the singularities of the Euclidean distance functions at rib and ridge lines preclude a simple solution. For a good discussion of these effects see [Port94], in particular chapter 1 where the relevant phenomena are explained in detail in 2D for curves. Thus, the only known ways to solve the problem is by means of iterative refinement that introduces a considerable computational burden into the implementation and, because of the aforementioned singularities, doubts as to their convergence. In our case, this iteration cycle would have to be repeated for every single point in the initial data set. Given that, the increase in complexity (and the effort required to implement a far more complicated technique) do not seem to constitute a reasonable trade-off against a gain in accuracy (of questionable reliability). In fact, it can be argued that because we are only trying to fit surface patches locally, and the demands for accuracy are high only in the proximity of each point of interest, we can afford to use algebraic distance without significant sacrifices in accuracy. The results presented in this and subsequent chapters seem to support this argument: within the scope of this research we are ultimately interested in the curvature values of the surface, not in the fitted patches themselves.

Figure 20 illustrates this argument. The bias introduced by the use of the algebraic error is expected to be higher [Zhan95] as we move away from the centroid of the data points. The nature of our data (i.e. samples taken using a scanner) implies that in most cases this will be close to the point of interest *P*, which is in the centre of the neighbourhood catchment area. We can therefore argue that, for the purposes of our intended applications, the bias around the point of interest is minimal, and a further improvement is not significant unless it is of similar computational efficiency. The results of curvature calculation and ridge detection given in chapters 5 and 6 respectively appear to support this argument.

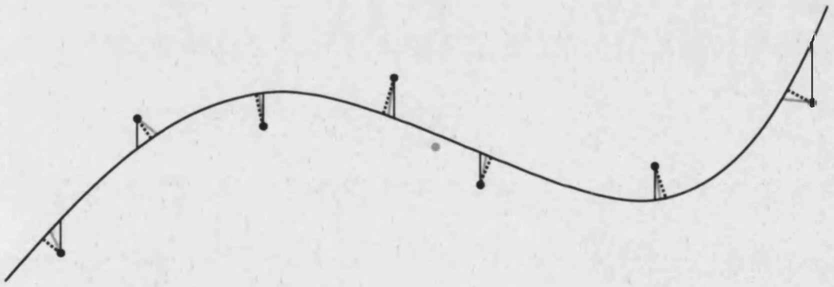


Figure 20: Accuracy of surface fit by minimizing: the height error (solid line); the algebraic error (grey line), the orthogonal distance (dotted line). The direction of the algebraic error is biased towards the centroid of the points (shown here as a grey dot) with respect to the direction of the orthogonal error[FiFi95][Zhan95]. In most cases the algebraic distance is greater than the orthogonal distance but smaller than the height distance, especially when the surface tends to become locally parallel to the vertical axis.

4.3 Closest point selection

As a result of the nature of the least-squares fitting, there is no constraint to guarantee that $F(P) = 0$ i.e. that the point of interest lies exactly on the quadric patch fitted around its neighbourhood. In general, however, it is safe to assume that $F(P)$ is very close to zero (generally proportional to the square root of λ_1). It is therefore sufficient to find a point P_0 on F that is as close as possible to P. It would be ideal to locate the point on F

that is *exactly the closest* to P. However, it is not convenient to do this in practice, as it would require calculation of the normal n_0 to F, that passes through P and then calculation of P_0 as the intersection between n_0 and F.

Unfortunately, the resulting formulation of the problem is a 6th-degree equation with no closed-form solution. Solving this sextic would, in fact, be equivalent to finding the orthogonal distance from a point to a surface, a problem of some considerable subtlety as noted above. Again, an iterative approach would be computationally expensive, so we chose instead to calculate a direction n_0 that *approximates* the surface normal. In the current implementation we do so as follows (Figure 21):

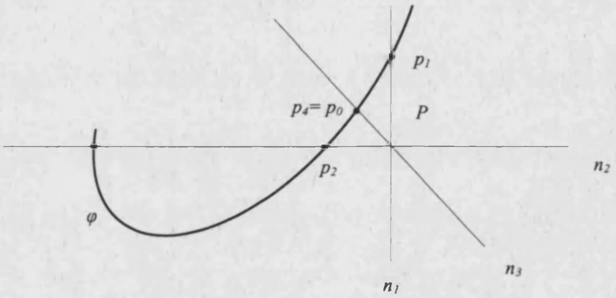


Figure 21: Illustration of the closest point calculation. In the case illustrated, p4 is the one closest to p so it is selected as the best estimate.

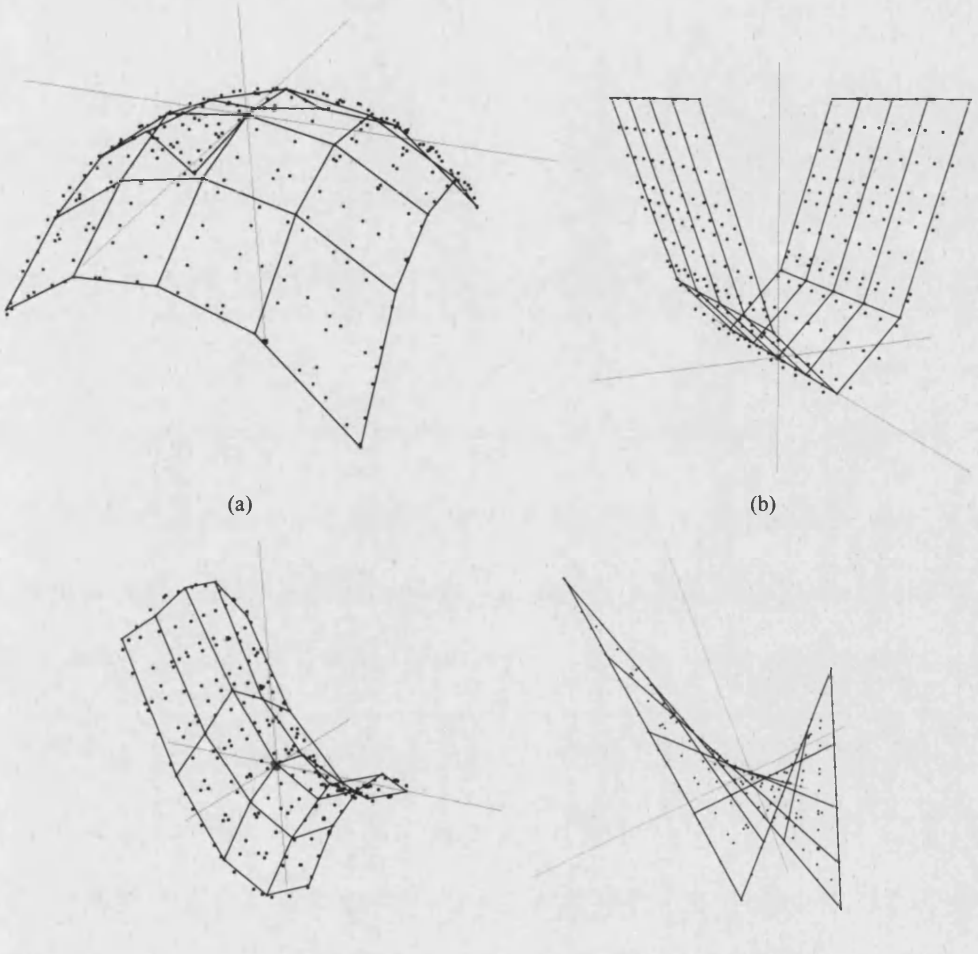
We select three lines n_1, n_2 and n_3 that pass through P. For best results we choose them to be orthogonal to each other, and in order to simplify the implementation we can choose them, without sacrificing generality, to be parallel to the axes of the orthogonal coordinate system used. For each of them we calculate the intersections P_i with F. This problem reduces to a second degree equation that is easy to solve and has either zero, one, or two solutions. In the case of zero solutions, we reject n_i from our set of options. In the case of two solutions, we select the one that corresponds to the smaller distance d_i between P and P_i. In addition, we set a new line n_4 to be in the mean direction of the remaining lines and calculate its intersection P_4 and its distance d_4 from F. We now have up to four points P_i that are likely approximations to the ideal point P_0. We select the one (P_{min}) that corresponds to the smallest $d_i = d_{min}$, and set $P_0 = P_{min}$.

Alternatively, it is possible to further refine the result by setting a new line n_{i+1} to be in the mean direction of n_{min} (i.e. the direction from P that intersects F at P_{min}) and n_{next} (i.e. the one of the previously set directions n_i from P that intersects F with the second smallest distance). Note that this iteration is not the same as the one necessary for minimizing the orthogonal error as presented by [BaIW93], [SuSP94] or [ARCW02]. It is a much simpler affair because it occurs after fitting, i.e. F is known, P is known, and each iteration involves the solution of a quadratic rather than a sextic equation thus removing the need for *nested* iterations. In any case, it has been shown that in practice the process converges very quickly (typically after 2-3 iterations). Our experiments have indicated that, in practice, the further refinement described in this paragraph only needs to be employed in less than 0.5% of the cases.

We thus obtain a point P_0 that is a sufficiently accurate approximation to the point on F closest to P, in that the curvature properties of the surface at P_0 are sufficiently close to those at the closest point to P on F.

4.4 Results

Figure 22 illustrates some results of our implementation of the quadric patch fitting on test data representing surfaces of known type [Gass83].



(a) (b)

(c) (d)

Figure 22: Results of the quadric patch fitting for test data representing surfaces of known types: (a) synclastic, (b) developable, (c) soft anticlastic, (d) steep anticlastic. In each case, the 25 test data points are organized as 5x5 quadmeshes (as shown here as by the black lines, with the data points themselves located on the vertices), black dots are sampled points of the resulting fitted quadrics.

Table 1 below shows the values of the Gaussian and (absolute) mean curvatures that were returned for the test patches illustrated in Figure 22. It is immediately obvious that the figures correspond to the definition of surface geometry according to curvature values.

Curvature	*(a) synclastic*	*(b) developable*	*(c) soft anticlastic*	*(d) steep anticlastic*
Gaussian (K)	0.12	0.00	-0.12	-0.12
Mean (|H|)	0.39	0.99	2.44	4.32

Table 1: Curvature values returned for the test patches of Figure 22

It must be noted that the above results are only illustrations (hence the use of only trivial data sets at this point) and are not intended to constitute a thorough evaluation of our approach and its implementation. This is presented in the remaining sections of this chapter, where we present a detailed numerical evaluation of the surface fitting using a number of test objects. The accuracy of the curvature evaluation will be discussed in chapter 5.

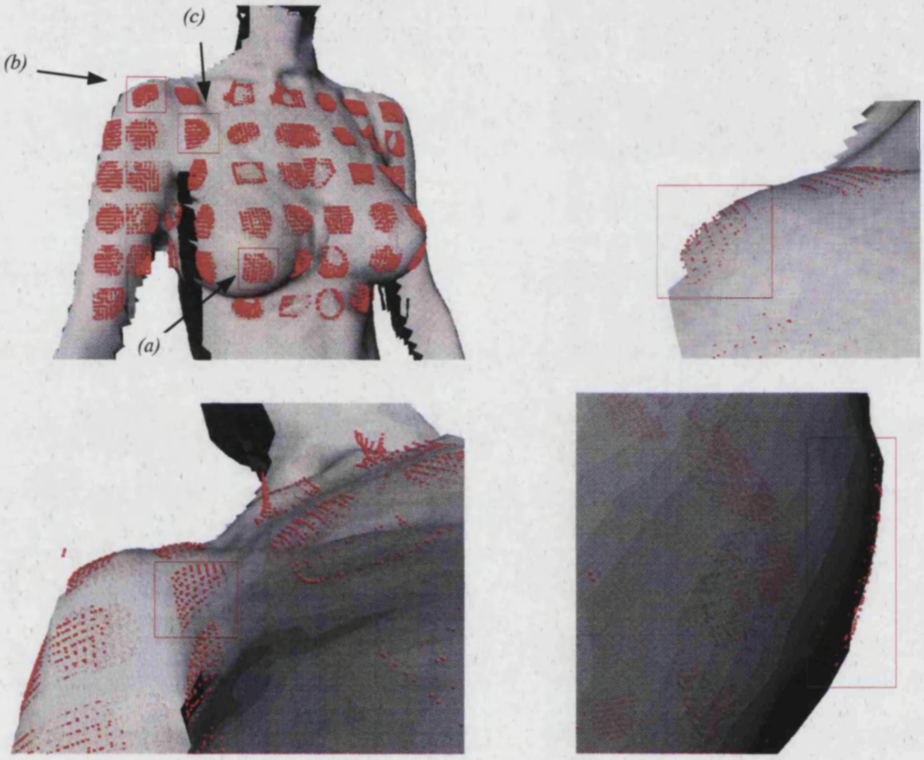


Figure 23: Application of the proposed patch fitting algorithm to real 3D scanner data. Top left: upper torso with selected fitted patches. The three patches marked are(a) synclastic,(b) developable and (c) anticlastic. Bottom right, top right and bottom left: close-up views of the marked patches in the same order.

Finally, Figure 23 shows the application of the proposed quadric patch fitting algorithm to real human body data acquired using 3D scanning hardware. The cases illustrated are the three usual ones: synclastic, developable, and anticlastic. In can be seen that in all cases there is a satisfactory approximation of the surface around the point of interest.

4.4.1 Evaluation Strategy

In order to evaluate the accuracy of our method, we have used the sampled test objects described in section 3.3.1. More specifically, for each test object the steps below were followed:

- For every point P in the point cloud describing the object:
  - A quadric patch was fitted using the minimization described in 4.2.
  - A 'closest' point P_0, on the patch, to P was found using the method described in 4.3.
  - The following metrics were recorded: the lowest eigenvalue λ_1 of the scatter matrix (4.14), the second lowest eigenvalue λ_2, the ratio of λ_1/λ_2, and the distance D between P and P_0.
- The mean, standard deviation, minimum and maximum of each of the metrics were calculated, and histograms of the values were built.

In the metrics above, λ_1 represents the algebraic error of fit. It is a measure of the quality of fit across the neighbourhood of points. The ratio λ_1/λ_2 is a measure of stability of the fit – lower values indicate a better separation of the values of λ_1 and λ_2 and therefore more stable (i.e. potentially more reliable) results. Finally, D is an estimate of the orthogonal error between P and the fitted surface F. It provides a more localised measure of the quality of fit at P, rather than across $N(P)$ as provided by λ_1. The next section illustrates the results of this procedure.

4.4.2 Analysis of the Results

The summary of statistics for the metrics defined above is given in Table 2, both for each individual test object and overall for the complete set of them. The units of measurement in which the distance figures (D) are given are 10^{ths} of a millimetre, whereas the ratios are pure numbers. The eigenvalues correspond to algebraic distances and cannot be expressed in the same units as those used for geometric distances. The distance values are significantly small compared to the size of the objects, which were designed to have an average radius of 0.5m, A this stage, it would thus seem that the quality of fit is very satisfactory, at least in the majority of cases.

The only apparent exception seems to be the case of the tooth, where there are approximately 30 (over a total of 4,963) sample points for which the value of D seems to be inappropriately high. Although the reason for these outlying 0.6% of the points is not clear, it is easy to remove them by setting an *a posteriori* threshold for the values of D. In fact, it can be argued that such thresholding is necessary, since they contaminate the statistics and compromise the reliability of the results.

Ellipsoid	**Lowest Eigenvalue**	**2nd Lowest**	**Ratio (stability)**	***D***
Average:	3.05742E-05	16110.92475	0.00021268	0.001332752
Standard Deviation:	2.83572E-05	10660.14037	0.006267672	0.007944858
Minimum:	4.49038E-10	9.34936E-05	2.74666E-13	0
Maximum:	0.000194906	50088	0.249425629	0.10004499
Tooth				
Average:	0.465408973	1517.769065	0.009958386	1.018129475
Standard Deviation:	1.250563973	2045.16721	0.040317276	6.856167633
Minimum:	1.84468E-07	0.00205797	6.20141E-11	0
Maximum:	8.99739	12588	0.400731758	95.43370317
Octahedron				
Average:	0.052893416	10270.04213	0.000151273	0.281431566
Standard Deviation:	0.075643784	6757.162969	0.003120395	0.258899973
Minimum:	1.34676E-08	3.27967E-05	3.38165E-09	0
Maximum:	0.484073	42628.6	0.15448414	1.41
Bumpy Sphere				
Average:	0.000259255	6834.897987	3.99731E-08	0.012930371
Standard Deviation:	0.000601341	1874.26086	1.03275E-07	0.017778042
Minimum:	0	0	5.16157E-13	0
Maximum:	0.032962	12151.5	5.21821E-06	0.106301458
OVERALL				
Average:	0.129648055	8683.408482	0.002580595	0.328456041
Standard Deviation:	0.331709364	5334.182851	0.012426361	1.785197627
Minimum:	0	0	2.74666E-13	0
Maximum:	8.99739	50088	0.400731758	95.43370317
	Lowest Eigenvalue	**2nd Lowest**	**Ratio (stability)**	***D***

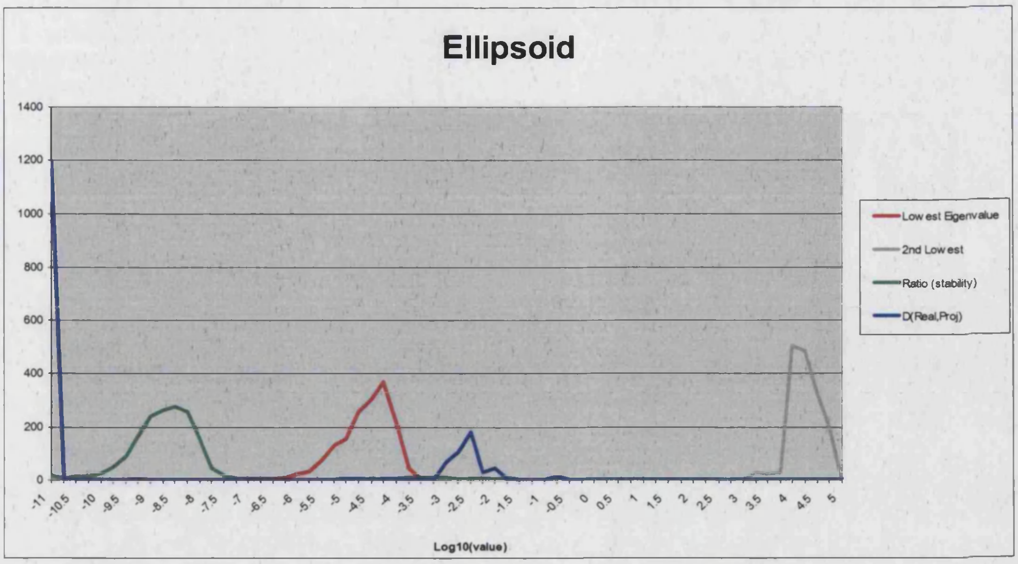
Table 2: Summary of statistics for the evaluation metrics of quadric patch fitting.

In the particular case of the tooth surface, the value of the threshold was set empirically (to a value of 1mm) by checking the histogram of values (more of which follows soon) and observing that the outliers in question were fairly well separated from the inlying *D* values. However, in general, a user of our implementation would have to adjust this threshold accordingly.

The histograms of the values summarised in Table 3 are given in Figure 24. It can be easily seen that in general values peak in the region of 10^{-1}, with the peaks for λ_2 at values consistently higher than those for λ_1 (as expected). Also, in all cases the histogram for *D* demonstrates a secondary peak at values that are in the band of 10^{-11} or lower (i.e. near zero), indicating a percentage of cases (typically around 5%) where the quality of fit around P is not merely satisfactory but indeed near-perfect.

Ellipsoid	**Lowest Eigenvalue**	**2nd Lowest**	**Ratio (stability)**	***D***
Average:	3.05742E-05	16110.92475	0.00021268	0.001332752
Standard Deviation:	2.83572E-05	10660.14037	0.006267672	0.007944858
Minimum:	4.49038E-10	9.34936E-05	2.74666E-13	0
Maximum:	0.000194906	50088	0.249425629	0.10004499
Tooth				
Average:	0.419441453	1528.872946	0.009252317	0.485321636
Standard Deviation:	1.103411526	2049.101961	0.039299468	0.822659366
Minimum:	1.84468E-07	0.00205797	6.20141E-11	0
Maximum:	8.67616	12588	0.400731758	7.6
Octahedron				
Average:	0.052893416	10270.04213	0.000151273	0.281431566
Standard Deviation:	0.075643784	6757.162969	0.003120395	0.258899973
Minimum:	1.34676E-08	3.27967E-05	3.38165E-09	0
Maximum:	0.484073	42628.6	0.15448414	1.41
Bumpy Sphere				
Average:	0.000259255	6834.897987	3.99731E-08	0.012930371
Standard Deviation:	0.000601341	1874.26086	1.03275E-07	0.017778042
Minimum:	0	0	5.16157E-13	0
Maximum:	0.032962	12151.5	5.21821E-06	0.106301458
OVERALL				
Average:	0.118156174	8686.184453	0.002404077	0.195254081
Standard Deviation:	0.294921252	5335.166539	0.012171909	0.27682056
Minimum:	0	0	2.74666E-13	0
Maximum:	8.67616	50088	0.400731758	7.6
	Lowest Eigenvalue	**2nd Lowest**	**Ratio (stability)**	***D***

Table 3: Summary of statistics for the evaluation metrics of quadric patch fitting following the a posteriori thresholding on D.



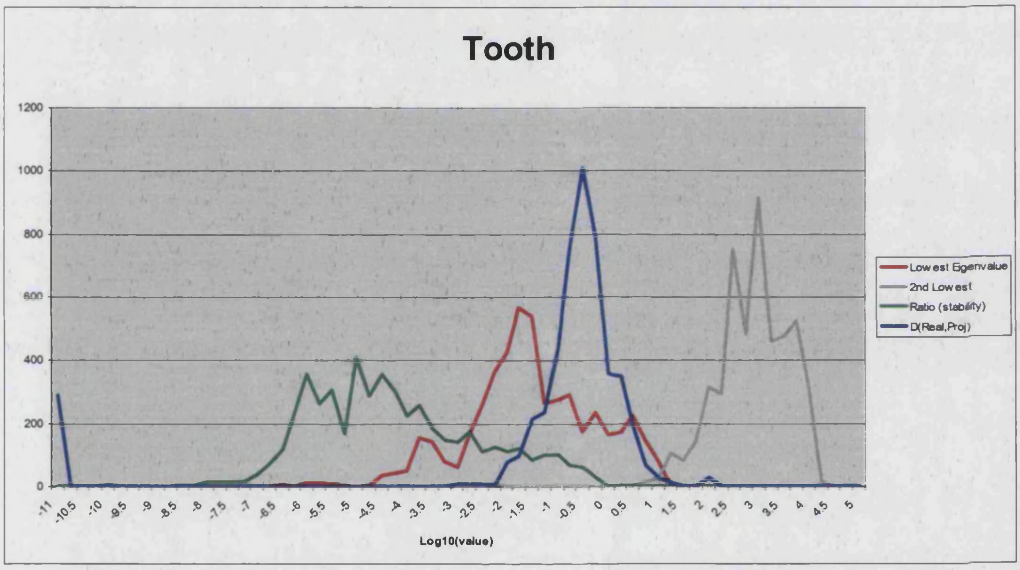
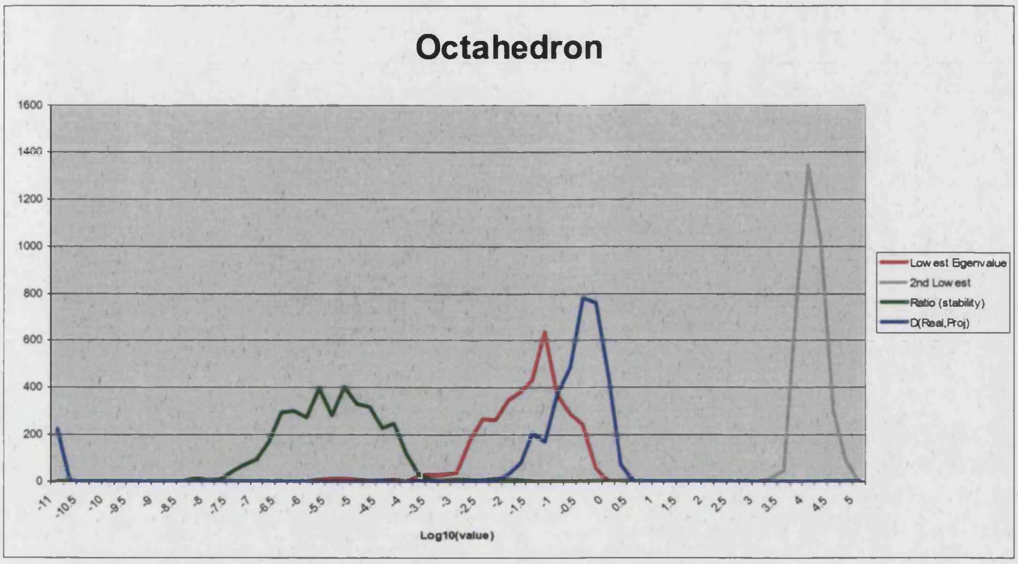


Figure 24 (part 1 of 2): Histograms for the values of the evaluation metrics of quadric patch fitting.



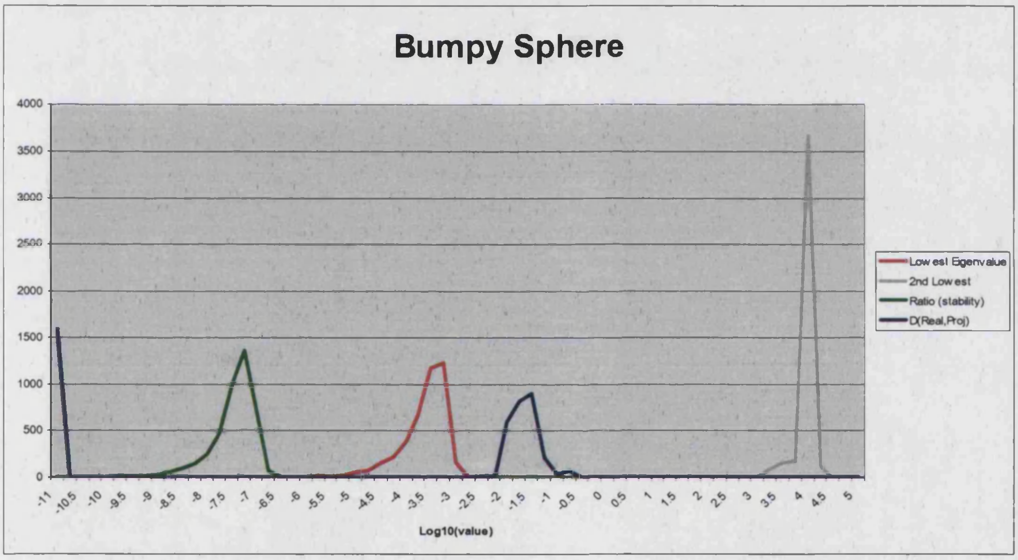


Figure 24 (part 2 of 2): Histograms for the values of the evaluation metrics of quadric patch fitting.

4.4.3 Agreement of Quadric Coefficients

An issue not covered in this evaluation is whether the quadric patches fitted to the data points actually correspond to the type of surface locally represented by the data (i.e. parabolic, elliptic or saddle). As noted in section 4.1, this is not an issue of importance here, though it has been in other application, e.g. segmentation of object parts from range data. Moreover, the answer, unfortunately, is not straightforward.

Let us first consider the case where the data points have explicitly and specifically been obtained by sampling a surface that is known to be quadric and therefore of known coefficients – such as for the trivial test patches used for the illustrations at the beginning of section 4.4. Let $a_0, b_0, c_0, e_0, f_0, g_0, l_0, m_0, n_0, h_0$ be the coefficients of the initial quadric surface, and $a, b, c, e, f, g, l, m, n, h$ the coefficients of the locally approximating fitted quadric at any point of the sampled test object surface. We could then check that the coefficients agree by evaluating a norm such as:

$$N_q = \left\| \begin{matrix} a-a_0 \\ b-b_0 \\ c-c_0 \\ e-e_0 \\ f-f_0 \\ g-g_0 \\ l-l_0 \\ m-m_0 \\ n-n_0 \\ h-h_0 \end{matrix} \right\|^2 . \qquad (4.16)$$

A small value of N_q (i.e. close to zero) would indicate that the coefficients agree in the sense that, provided that the original quadric is not almost degenerate, the quadric fitted is likely to be of the same type as the original quadric. However, a small value of N_q does not guarantee that the original and fitted quadrics are of the same type, nor does a large value of N_q necessarily mean that the fitting is poor in terms of algebraic error, nor that the fitted patch is unsuitable for reliable curvature calculation. In fact, it is claimed by Cazals et al. [CaPo03] that curvatures obtained by the fitted patch are reliable as long as the error of fit is small, *even if the coefficients do not agree.*

Secondly, let us consider the case where the original data point set does not correspond to a quadric surface. In this case, there is no way to establish whether the coefficients 'agree', chiefly because there are indeed no *a priori* values of $a_0, b_0, c_0, e_0, f_0, g_0, l_0, m_0, n_0, h_0$ to compare with the coefficients of the fitted quadric. More specifically, in our case where the surfaces are human bodies, and the main requirement is estimation of the surface curvature properties, any *a priori* classification of the parts of the surface as "elliptic", "parabolic", etc. is not possible except for illustrative purposes. For instance, we can only say that a part of the body *appears* to have an elliptic shape profile, and it is therefore reasonable to *expect* that the coefficients of a fitted quadric patch will correspond to an elliptic profile. However, we can only say whether that part of body surface *actually is* elliptic only *after* the fitting has been carried out and by using the result of the fitting in order to make the classification. It therefore becomes clear that if the fitted surface is, for example, a hyperboloid, this does not constitute a problem in the context of our research as long as it costitutes a local approximation that is good enough for calculation of principal curvatures. For the classification of parts of the surface, according to our hypothesis, into elliptic (synclastic), developable (parabolic) and anticlastic (saddle) as described in 4.1, the criterion will be the values of the curvatures and not the coefficients of the quadrics themselves.

4.4.4 Noise

In order to evaluate the behaviour of our proposed method with respect to noise, we have carried out an experiment consisting of the following steps:

- Random noise was added to the test ellipsoid. The noise was of a uniform distribution, initially with a width equal to 1/12th of the sampling resolution, and then at increments of the same amount up to 6/12ths (50%) of the resolution. This generated a sequence of 6 ellipsoids (7 including the initial one, with no noise added), each one with more noise than the previous one.
- For each ellipsoid in the sequence, the evaluation metrics described in section 4.1.1 were calculated per sample and then an average was taken per ellipsoid.

The results of these calculations are presented in Table 4, and the associated graphs are given in Figure 25.

Noise level (12^{ths} of sampling resolution)	Lowest Eigenvalue	2nd Lowest	Ratio (stability)	*D*
0	0.000030788	16313.997857433	0.000187656	0.001370643
1	0.115606094	16540.268453782	0.000014599	2.417420074
2	0.442034446	17355.423993077	0.000047318	4.902890881
3	0.941751815	18565.416452821	0.000086947	7.772850820
4	1.664044017	20755.129357372	0.000129269	11.849592409
5	2.112846546	22483.020165385	0.000140028	14.982373780
6	2.715546369	24899.830167949	0.000154595	18.927764027

Table 4: Mean values of the evaluation metrics for the test ellipsoid with increasing levels of noise.

It can be clearly seen from the graphs that as the amplitude of noise increases linearly, so do the mean values for λ_1 and D. This means that the behaviour of our proposed method is stable with respect to noise, even when the latter reaches levels that are comparable with the resolution of our data. Clearly, as noise increases the quality of fit diminishes (since lower values imply better fit for all the evaluation metrics concerned), however the drop in the quality of fit (represented by the values of λ_1 and D) is only proportional to the noise. On the other hand, the stability of the fit (represented by the value of λ_1/λ_2) does not seem to be dependent on the addition of noise and, indeed, seems on average to be better given noisy data than it is with ideal, perfect data.

In practice, when running the fitting algorithm on real human body scan data obtained via a scanner (and therefore the presence of some noise is inevitable), we prefer to apply a smoothing scheme to the data prior to fitting the quadric patches. In its simplest form, this is done by means of Gaussian smoothing. The location of each point P in the data is adjusted by taking a weighted average of the co-ordinates of the points in its neighbourhood, and the weight for each neighbour is taken as a Gaussian function (kernel) of its Euclidean distance from P. The weights are normalised so that their sum is equal to 1, and the width (σ) of the Gaussian is chosen according to: (i) the sampling resolution, and (ii) the amount of smoothing desired. For the data set used in this research (sampling resolution of the order of 1.5mm), we used the value of $\sigma = 0.5$mm, and if further smoothing is required we run multiple passes of the smoothing scheme using the same kernel.

To illustrate the effect of smoothing on the quality of the fitting results, we have run successive passes of our smoothing scheme on the last (noisiest) ellipsoid used for the noise evaluation experiment. After each pass we calculated, once again, the values of the fitting metrics. The results of this exercise are presented in Table 5, and the subsequent graphs are given in Figure 26.

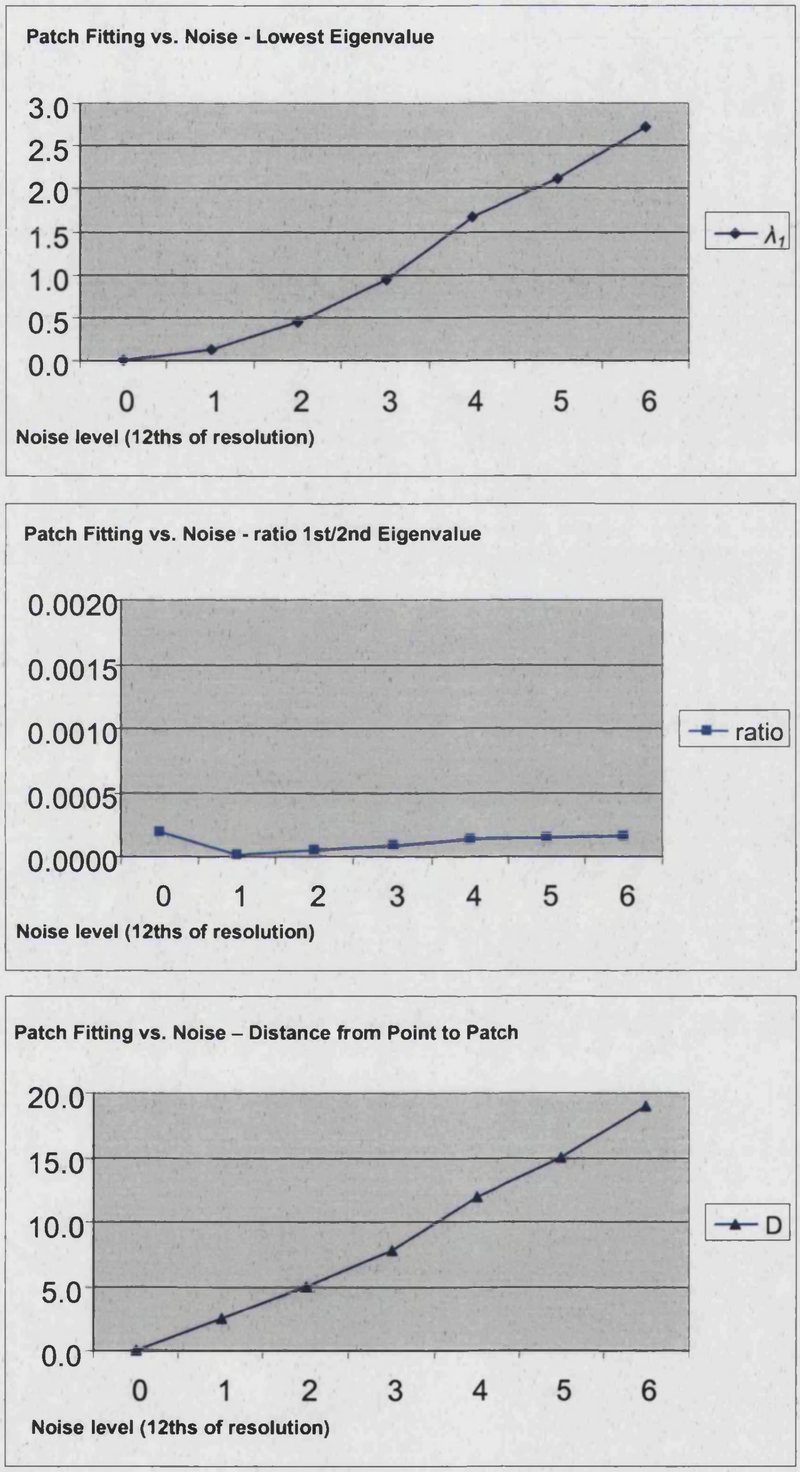


Figure 25: Graphs for the mean values of the evaluation metrics for the test ellipsoid with increasing levels of noise.

Passes	Lowest Eigenvalue	2nd Lowest	Ratio (stability)	*D*
0	2.715546369	24899.830167949	0.000154595	18.927764027
1	0.103470882	17142.645918816	0.001328509	1.626617143
2	0.028991362	17465.234830350	0.002481108	0.641414144
3	0.014740447	17862.529013055	0.001859027	0.402821751
4	0.011699518	18322.088658820	0.001663500	0.325577639

Table 5: Mean values of the evaluation metrics for a noisy ellipsoid with increasing levels of smoothing.

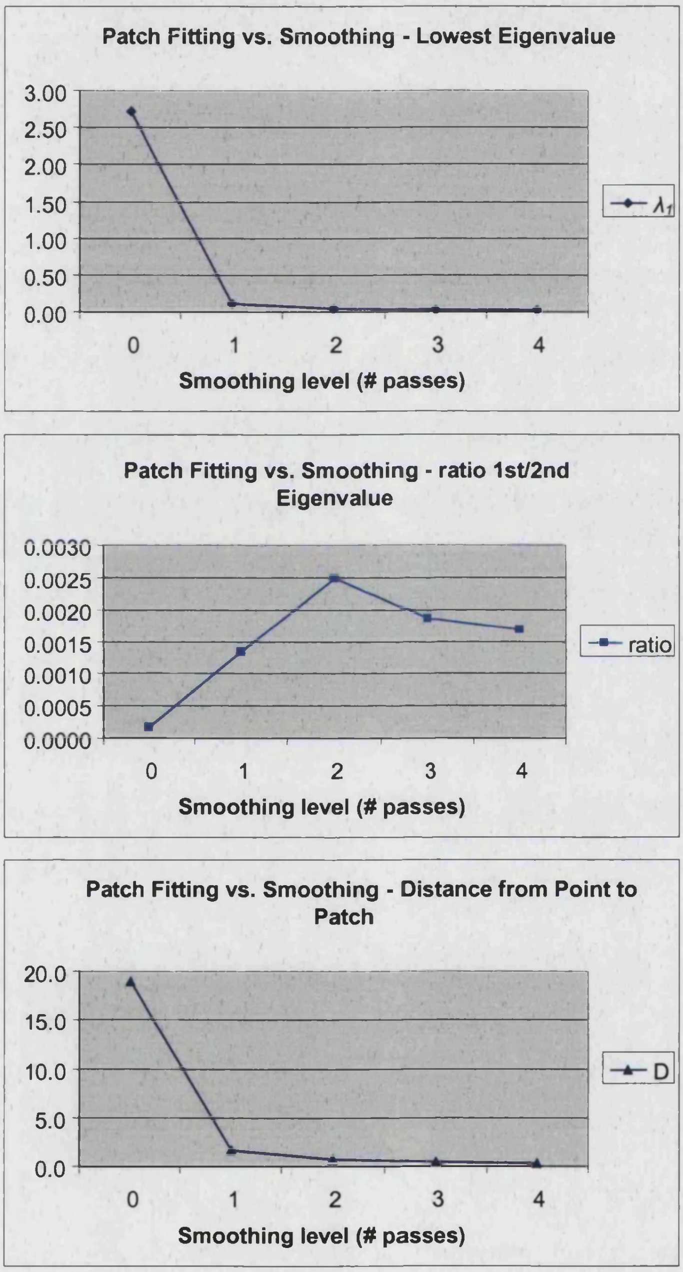


Figure 26: Graphs of the mean values of the evaluation metrics for a noisy ellipsoid with increasing levels of smoothing.

The graphs in Figure 26 indicate clearly that the quality of fit (represented by the values of λ_1 and D) indeed improves following smoothing. In fact, it appears that a small amount of smoothing (in our case, one pass) affects the quality of fit significantly, whereas subsequent passes result in a less significant level of further improvement. On the other hand, the stability of the fit (represented by the value of λ_1/λ_2) does not seem to be very dependent on the removal of noise by means of Gaussian smoothing. A little smoothing decreases the stability (possibly by introducing a degree of correllation between the data points), whilst further smoothing improves it a little.

4.5 Conclusions

The visual results from the methods presented in this chapter appear to tally with our expected perception of the shapes involved. Moreover, the numerical evaluation suggests that the local surface representation is true to the original surface, with sufficient resistance to noise. The outcome so far provides a solid infrastructure for representing local surface shape in a manner that:

- Overcomes the problem of absent/incomplete information on the structure of the point cloud, but also utilises whatever such information may be available.
- Enables further processing of the resulting surface representation, specifically for the purpose of calculating the surface curvature properties.

This latter calculation is the subject of the next chapter

5 Calculation of Curvature Characteristics

In this chapter we discuss the calculation of surface curvature characteristics from implicit representations generated using the local patch reconstruction presented in the previous chapter. More specifically, section 5.1 deals with the calculation of principal curvatures and the principal directions of curvature at a point on a surface represented in its implicit form. Section 5.2 shows the calculation of the Gaussian and mean curvatures and their use in classifying local surface shape. Section 5.3 introduces the alternative curvature measures of curvedness and shape index, and describes their advantages over Gaussian and mean curvature for the characterisation of surface shape. Section 5.4 discusses the problems arising from the nature of surface curvature properties as multicomponent ("vector") fields and the caveats introduced when attempting to separate focal surface sheets and track lines of principal curvature. The results from our implementation of the measures and methods discussed throughout this chapter are presented and evaluated in section 5.5.

5.1 Principal Curvatures and their Directions

It is important at this point to introduce some further definitions of basic concepts, the most important of which for this chapter are the *principal curvatures*, and their *principal directions*. Their importance lies in the fact that all the other differential geometry properties of interest in this research can subsequently be calculated from the principal curvatures and their directions. In most (but not all) cases these subsequent calculations are in fact fairly straightforward.

More specifically, the following concepts are of interest here [Port94]:

- *Tangent Line* of a planar curve C at a point P is a line with exactly one point of contact with C in the proximity of P (i.e. P itself). The *tangent line* has the same direction as the local direction of C at P.
- *Normal* of a planar curve C at a point P is a line that lies in the same plane as C, runs through P, and is perpendicular to the *tangent line* of C at P.
- *Circle of Curvature* of a planar curve C at a point P is a circle that is tangent to C at P (i.e. shares the same *tangent line* with C) and also approximates C locally at P.
- *Curvature* of a planar curve C at a point P is the rate of change of its direction

locally around *P*. Note here that:

  - The value κ of the *curvature* is expressed as the inverse of the radius *r* of the *circle of curvature* of *C* at *P*. Moreover, *r* is known as the *Radius of Curvature* of *C* at *P*.
  - The centre of the *circle of curvature* of *C* at *P* is also known as the *Centre of Curvature* of *C* at *P*, and lies on the *Normal* of *C* at *P*.
  - The locus of the *centres of curvature* of the curve is known as the *Focal Curve* or *Evolute* of *C*.
- *Tangent plane* *T* of a surface *S* in R^3 at a point *P* is a plane with exactly one point of contact with *S* in the proximity of *P* (i.e. *P* itself).
- *Surface Normal* of a surface *S* at a point *P* is a line that runs through *P* and is perpendicular to the *tangent plane T*.
- *Normal Planes* of a surface *S* at a point *P* are all the planes that run through the *surface normal*. Note that:
  - There is an infinite number of such planes in any direction $\underline{d}$ from *P* in the *tangent plane*.
  - These directions $\underline{d}$ are also known as the *Normal Directions* of *S* at *P*.
- *Normal Curve* to a surface *S* at a point *P* in direction $\underline{d}$ is the intersection between S and the *Normal Plane* of *S* at *P* in direction $\underline{d}$. Note that, by definition, a *normal curve* is always planar.
- *Normal Curvature* of a surface *S* at a point *P* in direction $\underline{d}$ is the *curvature* of the *normal curve* to the surface at *P* in direction $\underline{d}$.
- *Principal Curvatures* $\kappa(1)$ and $\kappa(2)$ of a surface *S* at a point *P* are the maximum and minimum *normal curvatures* of the surface at *P*. Note here that:
  - The *normal directions* of the surface at *P* where the *principal curvatures* occur are known as the *Principal Curvature Directions* of *S* at *P*.
  - The centres of the *circles of curvature* in the *normal planes* in which the *principal curvatures* occur are known as the *Centres of Curvature* of *S* at *P*.
  - The radii of these same circles are known as the *Radii of Curvature* of *S* at *P*.
  - The locus of the *centres of curvature* of *S* is known as the *Focal Surface* or *Evolute* of *S* (and is discussed in greater detail on section 6.5). Since there are, in general, two distinct curvatures for each point on S, the *focal surface* consists of two sheets.

Of all these concepts defined above, the most important ones for our research are the *principal curvatures* and the *principal curvature directions*. However, a difficulty has been encountered in calculating these two basic properties in the context of our research. If a surface is represented as a height function or as a parametric function, the calculation of its principal curvatures and directions has been thoroughly studied and well documented. Literature references are plentiful and easy to find (see, e.g. [Krey59][Port94][Hart95], and also section 5.1.1 below). This is not the case for the principal curvatures of implicit surfaces. The few examples available in the literature on this subject tend to center around the generation of a local height function (a Monge patch) that approximates the data around a point of interest, and then using this height function to calculate the prinicipal curvatures. However, such approaches result in losing the advantage of co-ordinate independence gained by using implicit surfaces in the first place. Besides, in our case the local implicit quadric patches used are approximations derived from our data points. A further approximation by converting our implicit patches to height functions carries no guarantee that the local surface shape will still be represented reliably. The best that can be done is to choose carefully the direction in which the height is measured – usually from a local estimate of the surface normal [CaPo03]. Even if this is done successfully, calculation of the principal directions of curvature from the height function subsequently requires transformation back to the global frame of reference of the object [CaPo03]. Therefore, it is desirable to seek a *direct* calculation of the principal curvatures from the implicit quadric patches.

5.1.1 Fundamental Forms

One of the most popular methods for calculating principal curvatures involves the use of *fundamental forms*. These are quantities that can be easily calculated for a parametric surface $R^2 \to R^3$ of the form:

$$f(u,v) = (x(u,v), y(u,v), z(u,v)) \quad , \qquad (5.1)$$

from the first and second partial derivatives, using the following equations [Krey59]:

- For the *first fundamental form*:

$$E = \left(\frac{\partial f}{\partial u}\right)^2 \qquad (5.2a)$$

$$F = \frac{\partial f}{\partial u} \cdot \frac{\partial f}{\partial v} \tag{5.2b}$$

$$G = \left(\frac{\partial f}{\partial v} \right)^2 \tag{5.2c}$$

- For the *second fundamental form*:

$$L = \frac{\partial^2 f}{\partial u^2} \cdot \bar{n} \tag{5.3a}$$

$$M = \frac{\partial^2 f}{\partial u \partial v} \cdot \bar{n} \tag{5.3b}$$

$$N = \frac{\partial^2 f}{\partial v^2} \cdot \bar{n} \tag{5.3c}$$

where:

$$\bar{n} = \frac{\frac{\partial f}{\partial u} \times \frac{\partial f}{\partial v}}{\left| \frac{\partial f}{\partial u} \times \frac{\partial f}{\partial v} \right|} \tag{5.4}$$

is the unit normal vector of *f* at *f(u,v)*.

The coefficients of the fundamental forms are usually presented in a matrix form, i.e:

$$A = \begin{bmatrix} L & M \\ M & N \end{bmatrix} \text{ and } B = \begin{bmatrix} E & F \\ F & G \end{bmatrix} \tag{5.5}$$

The principal curvatures of the surface can then be calculated by means of a 2x2 eigenproblem [Hart95]. The eigenvalues of $B^{-1}A$ (also known as the *Weingarten map* of the surface) are equal to the values of the principal curvatures of the surface, and the corresponding eigenvectors give the directions of the lines of curvature for these principal curvatures. In the cases where B cannot be inverted for particular points on the surface, then $B^{-1}A$ cannot be constructed and the curvatures cannot be calculated for that point. However, in that case we can at least obtain useful information regarding local degeneracies and singularities on the surface, since the invertibility of B depends on the values of the coefficients of the first fundamental form, which in turn are strongly related to the local surface shape. It is worth noting here that the first fundamental form (and, subsequently, B) measures the metric properties of the surface. Non-invertibility of B happens when $|B| = 0$, e.g. when we eventually have a curve instead of a surface [Krey59].

Therefore, fundamental forms provide an initial approach to solving the problem of calculating the principal curvatures for an implicit surface. If we could somehow calculate the coefficients of the fundamental forms directly from the implicit function $F(x,y,z) = 0$ and its derivatives, we could then formulate and solve the 2x2 eigenproblem described above.

Such a calculation of the fundamental forms of implicit surfaces does exist in the literature [Hart95], and is given in appendix B2. However, our implementation has highlighted two significant problems that make it rather inappropriate for our task.

The first problem is that the calculation method is not *really* (i.e. in practice) coordinate- and dimensionality-independent. One of the partial derivatives (F_z in the notation used for the formulation given in appendix B2) has to be singled out and feature in the denominator part of the equations used for calculation of the coefficients of the fundamental forms. Hartmann insists that this should make no difference to the final result. The choice of F_z for the denominator does not sacrifice generality, because it is possible to choose any partial derivative, and the definitions given by (B2.1) and (B2.4) can be appropriately re-written (i.e. by interchanging the subscripts of the partial derivatives) so that we still obtain the same values for the coefficients of the fundamental forms. This appears to be correct in theory, but does not seem to work very well in practice. In our prototype implementation, results tended to become unstable and unreliable for parts of the body surface locally parallel to the z-axis. In fact, this should be expected, since in that case the magnitude of F_z increases rapidly, thus causing numerical instability. In other forms of 3D data (such as geographic relief, faces and other deformed sheets where the surface is unlikely to run parallel to the direction of the observer – typically assigned a z-direction), such instabilities would be less likely to occur. However, a whole body is a deformed cylinder and its tangent planes may lie in any direction in 3D space. Therefore such an approach for calculating the fundamental forms is not really acceptable in our case. At best, we should choose between three sets of calculations, and every time select the one that is more suitable for the surface tangent plane at our point of interest.

However, calculating the fundamental forms in this manner presents a second problem, which is in fact even more serious. The approach does not seem to generate sufficient information for easy calculation of the principal curvature directions. If the surface were in Monge form (or any other height function), it would be sufficient to calculate the eigenvectors of $B^{-1}A$, and these would be the principal directions. They are 2D vectors, but this is sufficient in Monge form since there is a set orientation of the 3D surface around its normal, which can be used to relate the 2D vectors to the three dimensional space. In our case, however, it is inconvenient (albeit possible [CaPo03]) to have to establish a relationship between the 2D eigenvectors and our 3D space. There is no straightforward way of projecting these 2D vectors onto the tangent plane (i.e. where our expected 3D direction vectors should be lying), and there is no immediately obvious 'preferred' direction in the tangent plane to which the projected 2D vectors may be aligned in order to obtain the 3D vectors needed.

Therefore, we are left with the conclusion that the scope of this approach is limited. It is very easy to implement, and sufficient for calculating the *values* of the principal curvatures and the properties and features that can be derived therefrom. However, if we want to detect features that depend on the *directions* of the principal curvatures (such as ridges, discussed in the next chapter), then it is best to seek an alternative approach. One possibility is the one described by Eberly [Eber99], who shows that the principal curvatures and the principal curvature directions may be obtained from a 3x3 matrix computation. As it will be shortly shown, Eberly's approach is equivalent to the method given below, which is the one finally implemented.

5.1.2 Minimization Problem

A better approach in calculating the principal curvatures and their directions directly from the implicit quadric equation may be obtained by utilising the definition of principal curvatures as the minimum and maximum of normal curvatures. For a general implicit surface, the formulation by [Eber99] produces a 3x3 eigenproblem where two of the eigenvalues correspond to the principal curvatures, whilst their (3D) eigenvectors give the principal curvature directions in R^3. We follow a slightly different approach, in which it is convenient to take into account the fact that our implicit surfaces are quadric. This approach leads to a *non-linear* 2x2 eigenproblem with *exactly two* eigensolutions that correspond to the values and directions of the principal curvatures of the surface. The full details of our approach are given in appendix B3, but it can be outlined as follows:

If we represent the quadric in its matrix form, as in (4.6), we can then express it in partitioned form as:

$$F(\underline{x}) = \begin{bmatrix} \underline{x}^T & 1 \end{bmatrix} \begin{bmatrix} A & \underline{l} \\ \underline{l}^T & h \end{bmatrix} \begin{bmatrix} \underline{x} \\ 1 \end{bmatrix} = 0 \quad , \qquad (5.6)$$

where $\underline{l}$ is the vector $(l,m,n)^T$ and A the matrix of the coefficients a, b, c, e, f, g. It can then be shown from straightforward, elementary calculus, that the normal curvature at a point $\underline{x}_0$ on the surface, in any direction $\underline{u}$ on the tangent plane is:

$$\kappa(\underline{x}_0) = -\frac{\underline{u}^T A \underline{u}}{|\underline{p}_0|} \quad . \qquad (5.7)$$

We can therefore optimise (5.7) with respect to $\underline{u}$, to find the directions for which the normal curvature reaches its extreme values. By introducing appropriate Lagrange multipliers, the problem eventually reduces to a non-linear 4x4 eigenproblem of the form:

$$\begin{bmatrix} A-\lambda & \underline{p}_0 \\ \underline{p}_0^T & 0 \end{bmatrix} \begin{bmatrix} u \\ \mu \end{bmatrix} = 0 \quad . \qquad (5.8)$$

For a non-trivial solution we thus see that λ are the roots of

$$\begin{vmatrix} A-\lambda & \underline{p}_0 \\ \underline{p}_0^T & 0 \end{vmatrix} = 0 \quad . \qquad (5.9)$$

If we expand (5.9) we obtain a second order polynomial with respect to λ (the closed-form equations for the coefficients of this polynomial are given in appendix B4). Thus, there will be two solutions for λ and from those we may calculate two solutions for $\underline{u}$. In spite of the somewhat unconventional form of (5.9), it is easy to prove, by adapting slightly the methods used for the eigensolutions of a real symmetric matrix, that *λ(1)* and *λ(2)* are real, and that in general $\underline{u}(1)$ and $\underline{u}(2)$ are orthogonal. (See Appendix B of [Chal02] for details).

The principal curvatures *κ(1)* and *κ(2)* in principal directions $\underline{u}(1)$ and $\underline{u}(2)$ respectively are thus easily obtained from the eigensolutions of (5.8). In particular, we see from (B3.22) that the principal curvatures are given by:

$$\kappa(1) = \frac{-\lambda(1)}{|\underline{p}_0|} \text{ and } \kappa(2) = \frac{-\lambda(2)}{|\underline{p}_0|} \tag{5.10}$$

where $\underline{p}_0 = A\underline{x}_0 + \underline{l}$ (5.11)

is the normal to the quadric surface at point $\underline{x}_0$.

It must be noted that the signs of the principal curvatures are dependent on the orientation of the surface, i.e. on the sign of the surface normal $\underline{p}_0$ and ultimately on the choice of sign of the quadric function $F(\underline{x})$. Geometrical properties such as the side of the surface that the centre of each circle of curvature happens to lie are, of course, independent of such choices of sign. The implications regarding the various combinations of signs are discussed in the sections that follow.

5.2 Gaussian and Mean Curvature

There are two curvature properties of the surface that are of particular importance, namely the Gaussian and mean curvature. The Gaussian curvature (*K*) of a surface at any point is defined as the product of the principal curvatures at that point, i.e:

$$K = \kappa(1) \cdot \kappa(2) \quad , \tag{5.12}$$

whilst the mean curvature (*H*) is defined as the average of the principal curvatures:

$$H = \frac{\kappa(1) + \kappa(2)}{2} \quad . \tag{5.13}$$

These measures and their properties are well studied and documented in the literature (see, for example, [Krey59]). In the context of this research they are useful because their calculation is very easy to implement once the principal curvature has been obtained, and because they have well-understood properties that allow us easily to classify local surface shape. Moreover, K and H are invariant with respect to a number of co-ordinate transformations. It should also be noted that the Gaussian curvature K is invariant to the choice of sign for the principal curvatures but that the mean curvature H is not. For that reason, absolute values for the mean curvature $|H|$ are often given.

The mean curvature H is a measure of the local "sharpness" or "flatness" of the surface. More specifically, values of $|H|$ close to zero typically occur at flatter regions of the surface, whereas the value of $|H|$ increases as we approach folds and curvy regions and tends to infinity at cusps. As noted above, the sign of H depends on the orientation of the surface. If the orientation of the surface is not known, then the absolute value may still be of use, since higher values indicate feature-rich regions of the surface.

On the other hand, the sign of the K is independent of the surface orientation. This is because of the way K is defined: Principal curvatures of the same sign always produce a positive Gaussian curvature whether they are both positive or both negative. This makes K particulatly valuable for us when curvature is calculated for each data point from its corresponding locally fitted quadric patch. The way the patches are created (described in chapter 4) provides no guarantee that surface patches fitted to neighbouring points will have consistent orientations. Although it may be possible to make these neighbouring patches consistent by checking the patch normals and changing the signs so that they all point towards the same side of the surface, it is still likely that ambiguities may occur that affect the harmonisation of normals. However, even if the orientations of the fitted patches remain inconsistent, the value of K is not affected. Note that the same is not true for H, and this is why we often prefer to use $|H|$ rather than H in this research.

The sign of K at any point P on the surface is interpreted as follows [Krey59]:

- If $K > 0$, then the point is *elliptic*.
- If $K = 0$, then the point is *parabolic*.
- If $K < 0$, then the point is *hyperbolic*.

The classification of a point on a surface as *elliptic*, *hyperbolic* and *parabolic* refers to the local surface shape around the point, which can be respectively described as synclastic, anticlastic or developable (these shape profiles are discussed in section 4.1, and illustrated in Figure 14). Note that such a classification refers to the point *on the surface*, and does not necessarily correspond to a unique type of fitted quadric, as already discussed in section 4.4.3.

Moreover, on a smooth surface, the value of K varies smoothly as we traverse any smooth part on the surface. This means that there are regions of positive and negative Gaussian curvatures on a surface (as illustrated on the results on section 5.5), and parabolic points in general do not occur in isolation on the surface, but form lines on the boundaries between regions of positive and negative *K*. These lines are known as the *parabolic lines* of the surface, and constitute another type of surface feature that may be detectable on our data.

5.3 Curvedness and Shape Index

It is often desirable to study the scale-independent curvature characteristics of a surface, i.e. to identify two objects as having the same "shape", even if their "size" is different[11]. In that case, the Gaussian and mean curvatures are not very useful because they are not invariant to scaling transformations.

To illustrate this, consider the example of a sphere S_1 of radius r_1 (Figure 27) and its curvature at any point P_1. The principal curvatures at any point P_1 on sphere S_1 are always equal to:

$$\kappa_1(1) = \kappa_1(2) = \frac{1}{r_1} \tag{5.14}$$

and the Gaussian curvature K_1 throughout S_1 is therefore equal to:

$$K_1 = \frac{1}{r_1^2} \tag{5.15}$$

---

[11] As Koenderink notes [Koen90] on shape and size: "There doesn't seem to exist any general formalism that captures these notions and that people seem to agree on, although there are no essential difficulties involved".

Similarly, then, for another sphere S_2 of radius $r_2 > r_1$, the Gaussian curvature at any of its points will be equal to:

$$K_2 = \frac{1}{r_2^2} \quad . \tag{5.16}$$

But because $r_2 > r_1$ it follows from (5.15) and (5.16) that $K_2 < K_1$. This different value of a shape property for two surfaces that are commonly understood to be of the same "shape" contradicts the common sense dictum that "all spheres have the same shape" [Koen90].

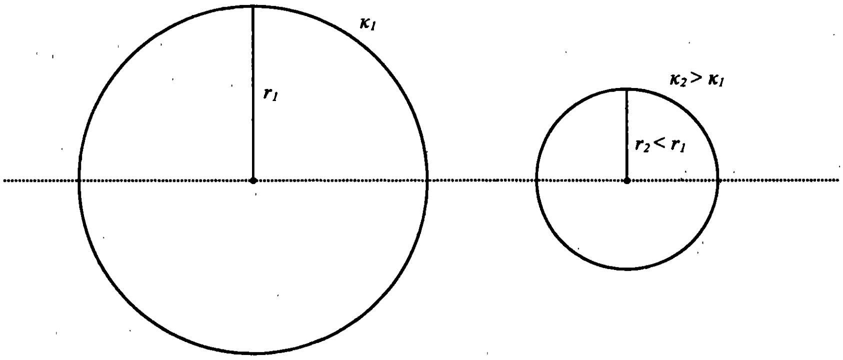


Figure 27: Lack of scale-invariance of curvature properties. Although the two sphere essentially represent the same "shape", the one on the right has a higher curvature.

For this reason, the concepts of *curvedness* (*C*) and *shape index* (*I*) have been introduced by Koenderink [Koen90] as alternatives to the Gaussian and mean curvatures. They are respectively defined as:

$$C = \frac{2}{\pi}\ln(\sqrt{\frac{\kappa_{\min}^2 + \kappa_{\max}^2}{2}}) \quad \text{, and} \tag{5.17}$$

$$I = -\frac{2}{\pi}\arctan\left(\frac{\kappa_{\max} + \kappa_{\min}}{\kappa_{\max} - \kappa_{\min}}\right) \quad , \tag{5.18}$$

where, in (5.18), the $\left(-\frac{\pi}{2}, \frac{\pi}{2}\right)$ branch of the arctangent function is used.

The curvedness is a convenient alternative to the absolute value of the mean curvature, and is independent of the orientation (sidedness) of the surface. The shape index is not similarly independent of the surface orientation, but it has a number of interesting properties that make it more than just a convenient alternative to Gaussian curvature. Firstly, it is defined everywhere except for planar points (where the principal curvatures both tend to infinity), and points where the two prinicipal curvatures are equal (these are known as *ordinary umbilics* and will be discussed in greater detail in section 6.1). Secondly, it has a bounded range of values [-1, 1], and *every* distinct shape (except for the planar points) corresponds to a *unique* value of the shape index. The correspondence between values of the shape index and types of local surface shape is shown in Figure 28.

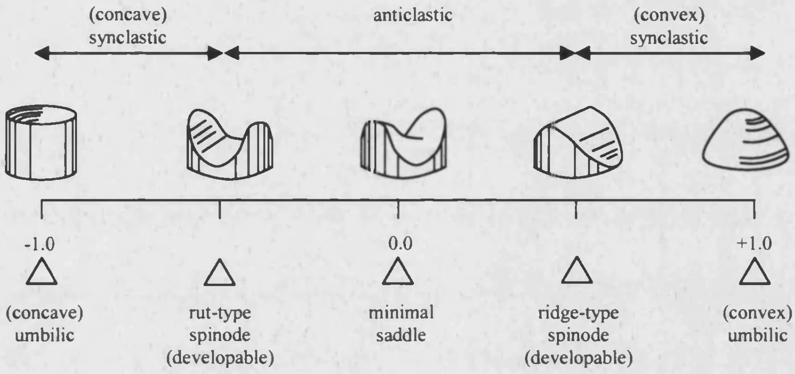


Figure 28: Interpretation of the values of the shape index (reproduced from Koen[90]).

Finally, and most importantly, the shape index is scale independent as is immediately obvious from the fact that any scaling of the surface by a factor of, say, f_s will result in a subsequent scaling of the principal curvatures by a factor of $1/f_s$.

5.4 Curvature Labelling and Tracking

For every point on a generic regular surface in R^3 there are two principal curvatures. These are usually referred to as the "red" and the "blue" curvature although [Port94] these terms are only used for convenience and it does not matter which is which. We can subsequently refer to the associated concepts using these labels. We therefore have the red and blue directions of principal curvature, and the red and blue centres of curvature which, as we move across the surface, form the red and blue sheets of the focal surface.

Moreover, the *lines of curvature* on a surface are defined as the curves on the surface that locally follow the directions of principal curvature. Therefore, these lines of curvature can also be labelled as red and blue, and at any point on the surface that is not *umbilic* (i.e. that has two distinct principal curvatures) a blue and a red line of curvature will intersect at right angles (since the directions of principal curvatures are then the orthogonal eigenvectors arising from the solution of the eigenproblem described in section 5.1.2).

Since the configuration of the lines of curvature tells us a lot about the shape of the surface, it would be desirable to detect these line of curvature on the surfaces that we are studying, which would mean to separate the field of $(\kappa(1), \kappa(2))$ of the principal curvatures we calculated. However, in practice this is not straightforward, and can in fact be quite difficult. For example, Koenderink notes ([Koen90], page 315) “Such fields cannot be ordered such that you have two-parameter line-like families. They are the worst nuisance in numerical work”.

As part of this research, the issue was investigated and an attempt was made to separate the red and blue curvatures for our surface data. Based on the assumption of continuity of the values of principal curvatures along the surface [Krey59][Koen90][Port94], and continuity of the lines of curvature, a region growing scheme was followed that is illustrated in Figure 29.

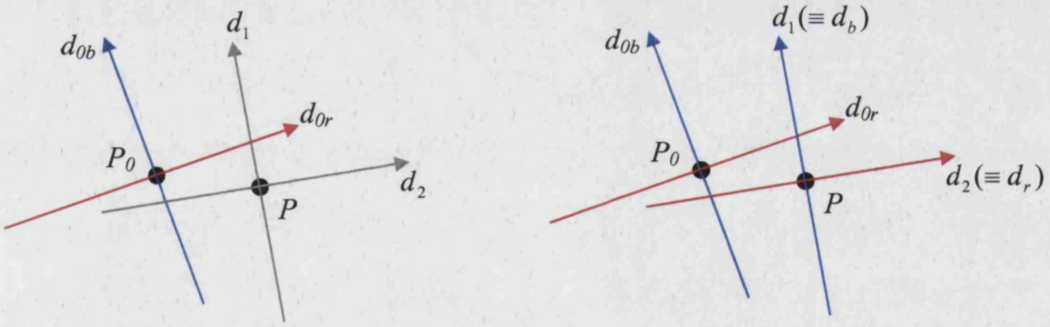


Figure 29: Propagation of red and blue curvature labels based on the directions of principal curvature. Left: Before. Right: After.

The labelling begins with a non-umbilic point P_0 on the data set that acts as a “seed”: its principal curvatures are arbitrarily labelled red and blue (κ_{0r} and κ_{0b}), and so are its corresponding principal curvature directions (d_{0r} and d_{0b}). Then, for every point P that is a neighbour of P_0 with principal curvatures $\kappa(1)$, $\kappa(2)$ and corresponding directions d_1, d_2 a comparison takes place between the directions of P and P_0. For this comparison (assuming that the direction vectors are normalised), the absolute values of the inner products of the direction vectors are used. These will have a value close to zero if the two vectors are almost normal, and close to 1 if they are almost parallel. The use of absolute value ensures that this comparison is invariant to changes in the sign of the principal curvature directions which, being eigenvectors, are of arbitrary sign. The principal curvature of *P* with direction most similar to the blue direction of P_0 is labelled the blue curvature of *P*, and vice-versa for the red curvatures. We have thus:

- If $d_{0b} \cdot d_1 > d_{0b} \cdot d_2$, then $d_1 \equiv d_b$ and $d_2 \equiv d_r$ (and $\kappa_1 \equiv \kappa_b$ and $\kappa_2 \equiv \kappa_r$),
- Else $d_1 \equiv d_r$ and $d_2 \equiv d_b$ (and $\kappa_1 \equiv \kappa_r$ and $\kappa_2 \equiv \kappa_b$).

Then, we propagate this labelling likewise for any neighbours of *P* that are not umbilic (i.e. in practice where $\kappa_1 \approx \kappa_2$ does not hold true) and are not already labelled.

The procedure terminates when there are no points left to label. If the point cloud is segmented in such a way that there are distinct regions such that for every point in one region there are no neighbours in the other, then a seed is placed in each such region and the entire process has to be carried out for each region.

However, the problems encountered during the execution seem to support Koenderink's argument. As will be illustrated in the next section, the propagation seems to produce reasonable results until it encounters a region in the proximity of an umbilic point or other irregularity on the surface (including regions that have been poorly sampled). In that case, propagation is not straightforward because the consistency of the results is not guaranteed once propagation extends beyond the first-order neighbours of the seed point. The behaviour of the blue and red curvature fields around such singularities is discussed in the next chapter.

5.5 Results

In order to present the results of the methods discussed in this chapter, we employ a visualisation by means of colour mapping. The figures in this section contain images of the test objects and body scans where the surface has been coloured according to the local value of a curvature property. The colours on the surface are interpreted according to the curvature property depicted in each figure. The correspondence between colours and curvature property values is illustrated in Figure 30. In all cases, a "rainbow-spectrum" colour map is used, where blue corresponds to low values, and red corresponds to high values. More specifically:

- For the visualisation of the *Gaussian curvature K*: colours range from blue for regions with low negative values (anticlastic, “saddle” regions), through to green for regions with values around zero (locally flat or parabolic) and eventually to red (synclastic). Note that the cut-off point for the extreme positive and negative values is very low, indicating the non-linear nature of this property: Most interesting features occur within a very short range of values.

- For the visualisation of the *absolute mean curvature* $|H|$: colours range from blue for regions with values close to zero (locally flat), through to green and eventually to red for regions with high positive values (sharply bent).
- For the visualisation of the *curvedness* C: colours range from blue for locally flat regions (i.e with extreme negative values, since C is defined as a logarithm) through to green and eventually to red for regions with extreme positive values (sharply bent).
- For the visualisation of the *shape index* I (please also refer to Figure 28): colours range from blue for regions with values from –1 (concave umbilics) upwards to cyan (“rut”-type spinodes), through to green for regions with values around zero (minimal saddles), to yellow (“ridge”-type spinodes) and eventually to red (convex umbilics).

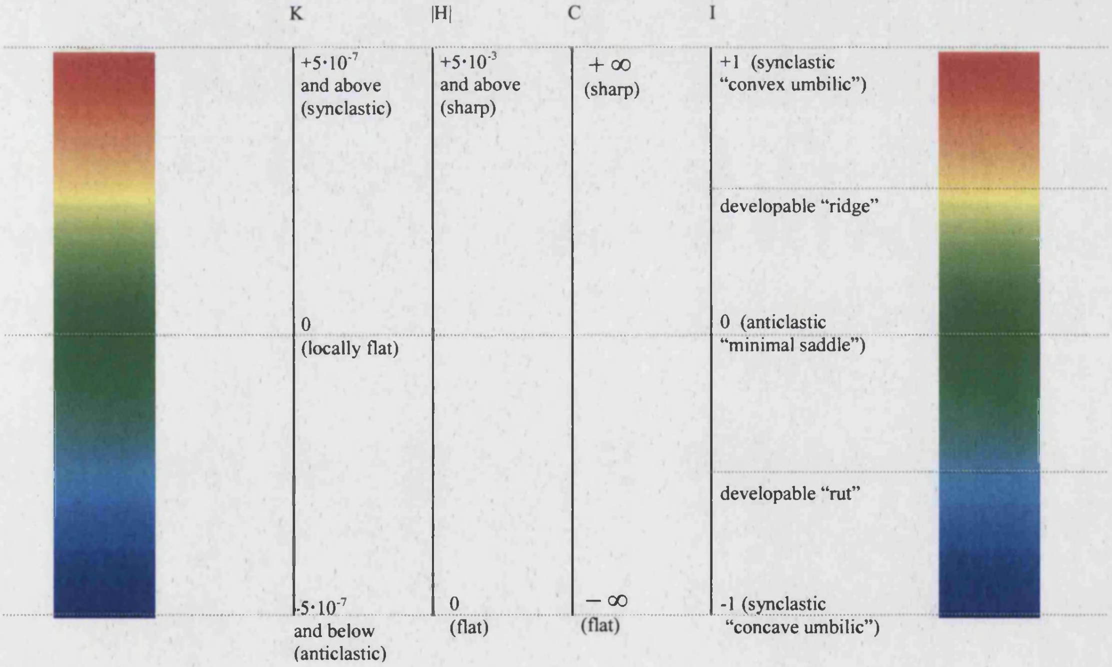


Figure 30: Interpretation of the colour mapping to the results.

The first example of such an illustration is given for the test ellipsoid (Figure 31). As expected, the colours of the Gaussian curvature map range from green to yellow to red, correctly corresponding to positive values throughout. The sharper regions of the surface are at the ends of the longer axis (poles), and this is where the red regions occur. This feature is also correctly represented on the mean curvature and curvedness maps, where higher values (shown in green) are shown at the poles.

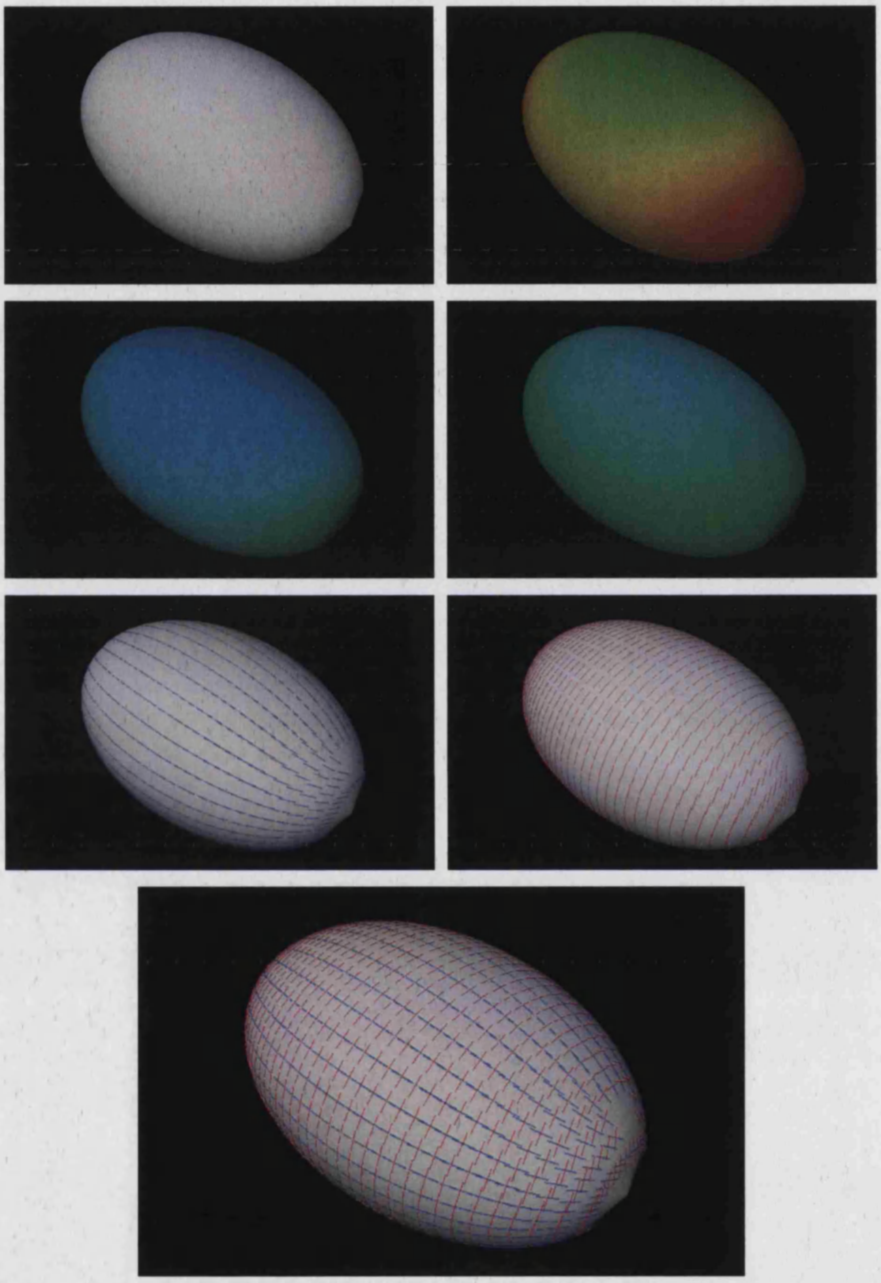


Figure 31: Curvature mapping results for the test ellipsoid. First row: the test ellipsoid, rendered with Phong shading (left), and its Gaussian curvature map(right). Second row: mean curvature and curvedness maps. Third row: the red and blue curvature directions, labelled using the scheme described in section 5.4. Bottom: the red and blue directions rendered together to give an impression of the lines of curvature on the surface.

The curvature directions are also shown in Figure 31. Note that in this case, the separation and tracking of the blue and red directions appears to have been consistent. Directions are not shown at the poles (they have been deliberately omitted as the cylindrical co-ordinate sampling used to generate this object, deliberately chosen to mimic scanner data, has there caused a singularity which, in turn, causes the fitting to fail), and also at the locations where the umbilic points of the ellipsoid are expected to be found [Port94]. Umbilics, however, will be discussed in greated detail in the next chapter.

The next example (tooth surface) is presented in Figure 32.

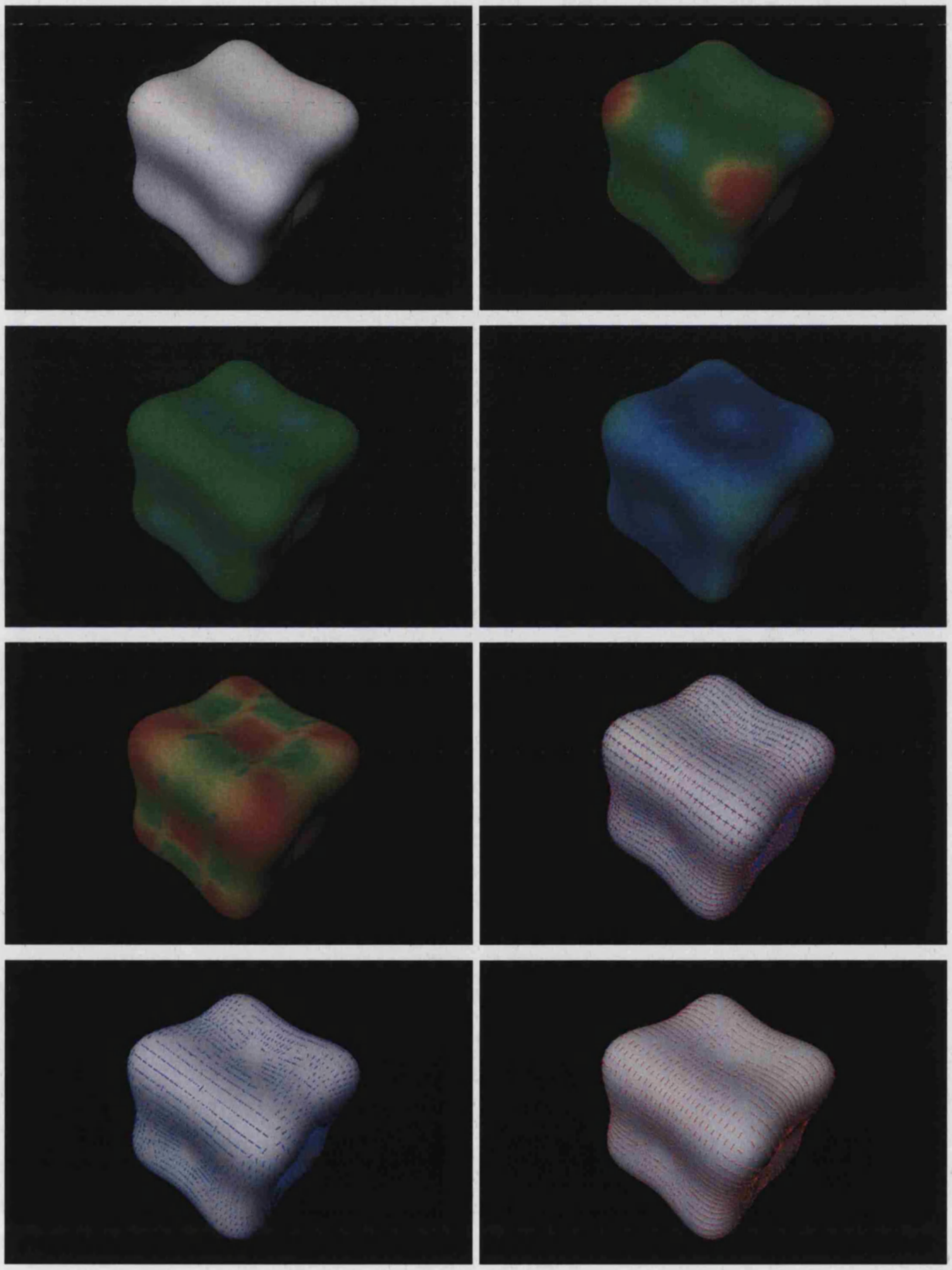


Figure 32: Curvature mapping results for the test tooth surface. Top row: the tooth surface rendered with Phong shading (left), and its Gaussian curvature map (right). Second row: mean curvature and curvedness and shape index maps. Third row: shape index map (left) and curvature directions. Bottom row: the red and blue directions shown separately.

For ease of interpretation, it is helpful to consider, the tooth surface to be a deformed cube. The Gaussian curvature map illustrates how the algorithm has correctly identified the higher curvature regions of the surface to be the eight "corners" of the tooth surface. It also illustrates the identification of the twelve saddle regions (shown in blue), one on each "edge". The curvedness map is also in agreement with these expectations (i.e. that the "edges" and "corners" are more curved than the "sides"). The mean curvature and shape index maps show the identification of the flat umbilic points, four on each "side" of the surface. Finally, the principal curvature direction images show how the tracking and separation of the red and blue curvature labels fails around these umbilic points, thus illustrating the difficulty of devising a consistent separation scheme.

The next example is the rounded octahedron, shown in Figure 33. The colours on the Gaussian curvature map are consistent with the absence of saddle regions on this surface, and the Gaussian cuvatures are higher at the six expected regions, i.e. the four "corners" on the equator and the two "poles". The shape index map for this object is also of interest, as it shows the "corners" and "poles" to be indeed synclastic, and the "edges" to be developable. Please refer to Figure 30 and Figure 28 which clarify the connection between these features, the values of the shape index and the colours used for their visualisation. It is interesting to note that, in the case of this particular test object, the surface normals are defined so that the "corners" and "poles" appear to be concave with respect to the shape index, as they would be seen by an observer standing *inside* the surface. This, however, does not affect the description of the actual surface shape because the surface normals are consistent. Because this is an implicit surface of the form $F(x,y,z) = 0$, the concepts of "inside" and "outside" are only relative to the sign of F, and therefore so are the concepts of "convex" and "concave". These can be swapped by simply changing the sign of F, however this does not change the shape of the surface described by F.

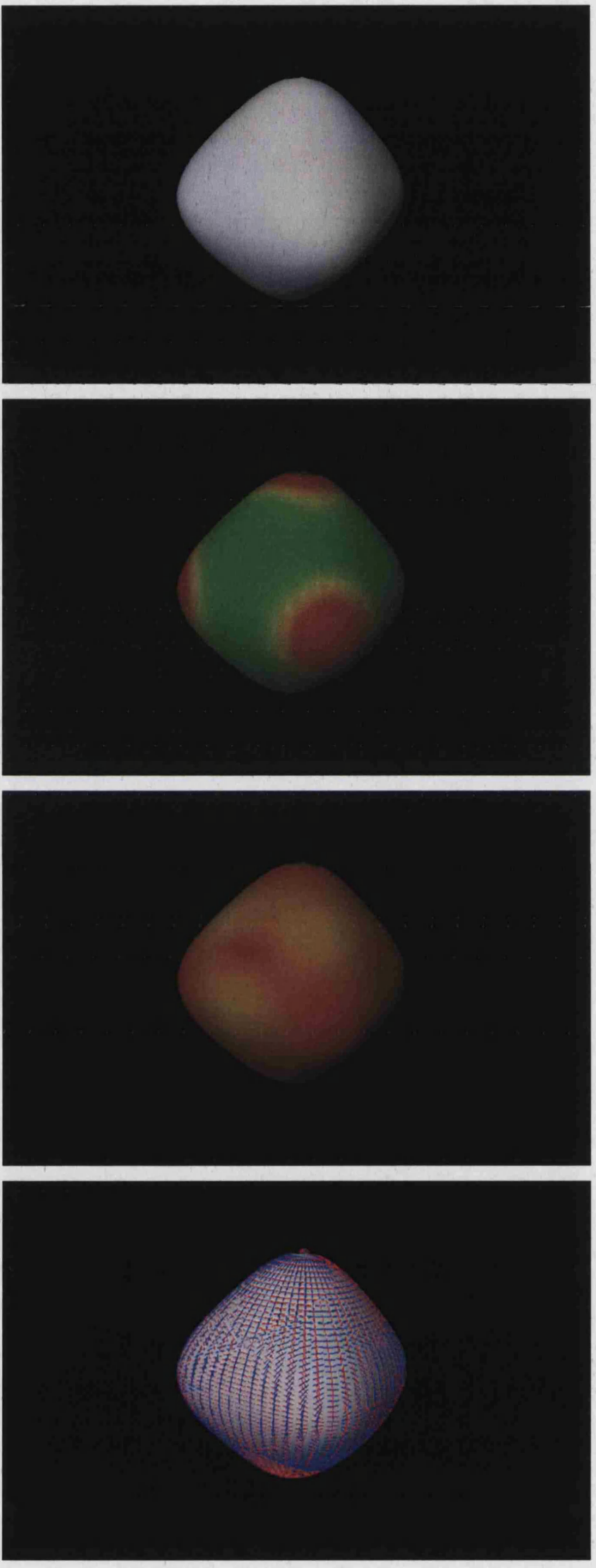


Figure 33: Curvature mapping results for the test rounded octahedron. From top to bottom: the surface rendered with Phong shading ;the Gaussian curvature map; the shape index map; and the curvature directions.

Last in the series of examples of test objects is the bumpy sphere (Figure 34), which illustrates the sensitivity of our algorithm to even small changes in shape. The most interesting result here is the curvedness map. On this test object which deviates only slightly from a perfect sphere (we used a bump factor of $\varepsilon = 0.05$, please refer to section 3.3.1 for details), the algorithm can still identify regions that are locally flatter (shown in blue) and locally sharper (shown in red) than the rest of the surface. Moreover, the configuration of such regions on the surface is consistent with its symmetry which is known (from [Port94]) to be the same as that of a regular tetrahedron.

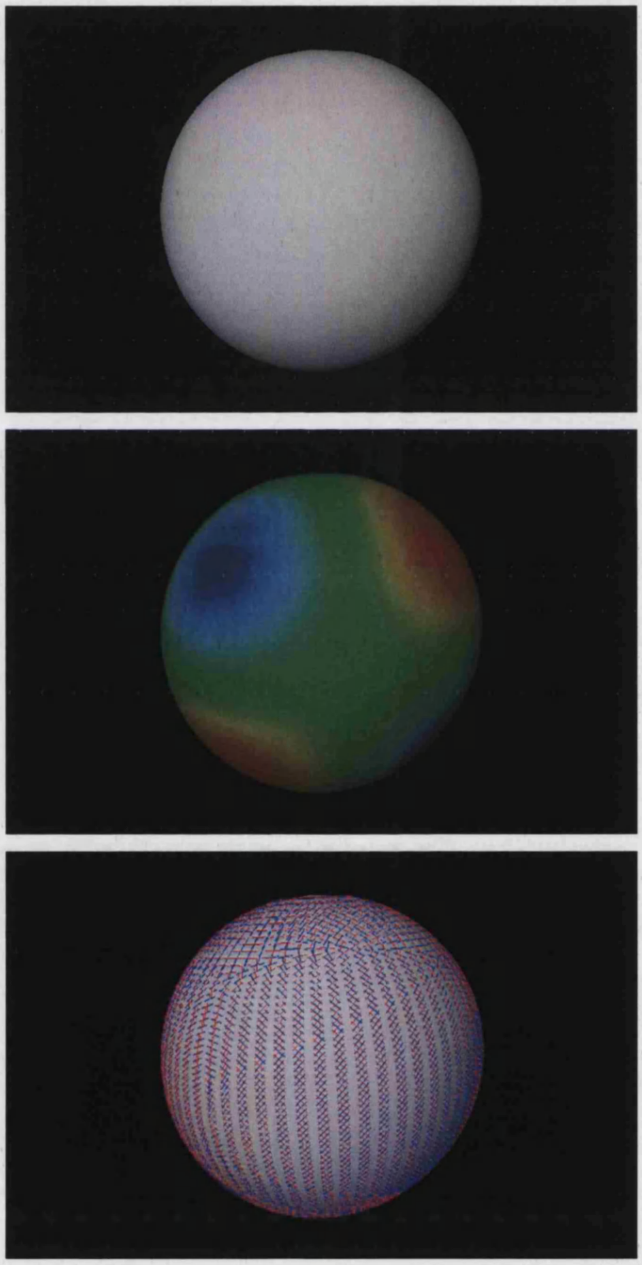


Figure 34: Curvature mapping results for the bumpy sphere. From top to bottom: the surface rendered with Phong shading ;the curvedness map; and the curvature directions.

The next series of images illustrate the results of our proposed algorithm on real data (human body scans). Figure 35, Figure 36 and Figure 37 show the curvature maps and principal directions for human torsos of medium resolution (2 cm). These torsos are canonical forms (quadmeshes) generated using the method described in [RBDT02]. The examples are picked so that genders and body types are roughly represented, with results shown for a thin and a fat male, and a thin and a fat female. In all cases, the curvature maps appear to be consistent with our common perception of shape for the particular body types. Note, for example, how the Gaussian curvature map highlights the presence of an overgrown abdomen for the more corpulent of the subjects, and the intense saddle regions on the waist of the slimmer "hourglass-figure" female.

It is also important to note that the curvature property maps appear to be distinctly different for scans of individuals of different body types. This suggests that it might indeed be possible to utilise curvature-related information for shape classification. In the next chapter, the implications of this observation are discussed in further detail, and an attempt at a numerical evaluation is included. Further experimentation involving a linear morphing produced from the principal mode of variation of an ASM (implemented by Avy Tahan [TBRB03]) between two different body types (results shown in Figure 38 and Figure 39) shows the births and deaths of synclastic, anticlastic and developable regions and the transitions between them as the surface deforms. It appears from the pictures that our detection is stable enough to illustrate these transitions smoothly, and also backs up the argument initially introduced by Porteous [priv. comm.] that there are indeed features on the surface that are de-formable rather than reformable. This also has implications for classification and tracking that are discussed in detail in the next chapter.

Finally, to illustrate the performance of the proposed method and its computational efficiency, we present in Figure 40 and Figure 41 the results returned for whole-body high-resolution (5mm) scans obtained using the Hamamatsu Body scanner. These scans typically consist of 50,000 to 70,000 points depending on the size of the subject. However, the curvature maps took less than 10 minutes to generate on a PC of the specification currently popular at the time of conducting the research (2.2Ghz Pentium IV with 512MB of RAM). These images have been subjected to heavy smoothing prior to processing (12 passes of a Gaussian kernel with $\sigma = 0.5$mm), yet they still seem to demonstrate a high level of feature details.

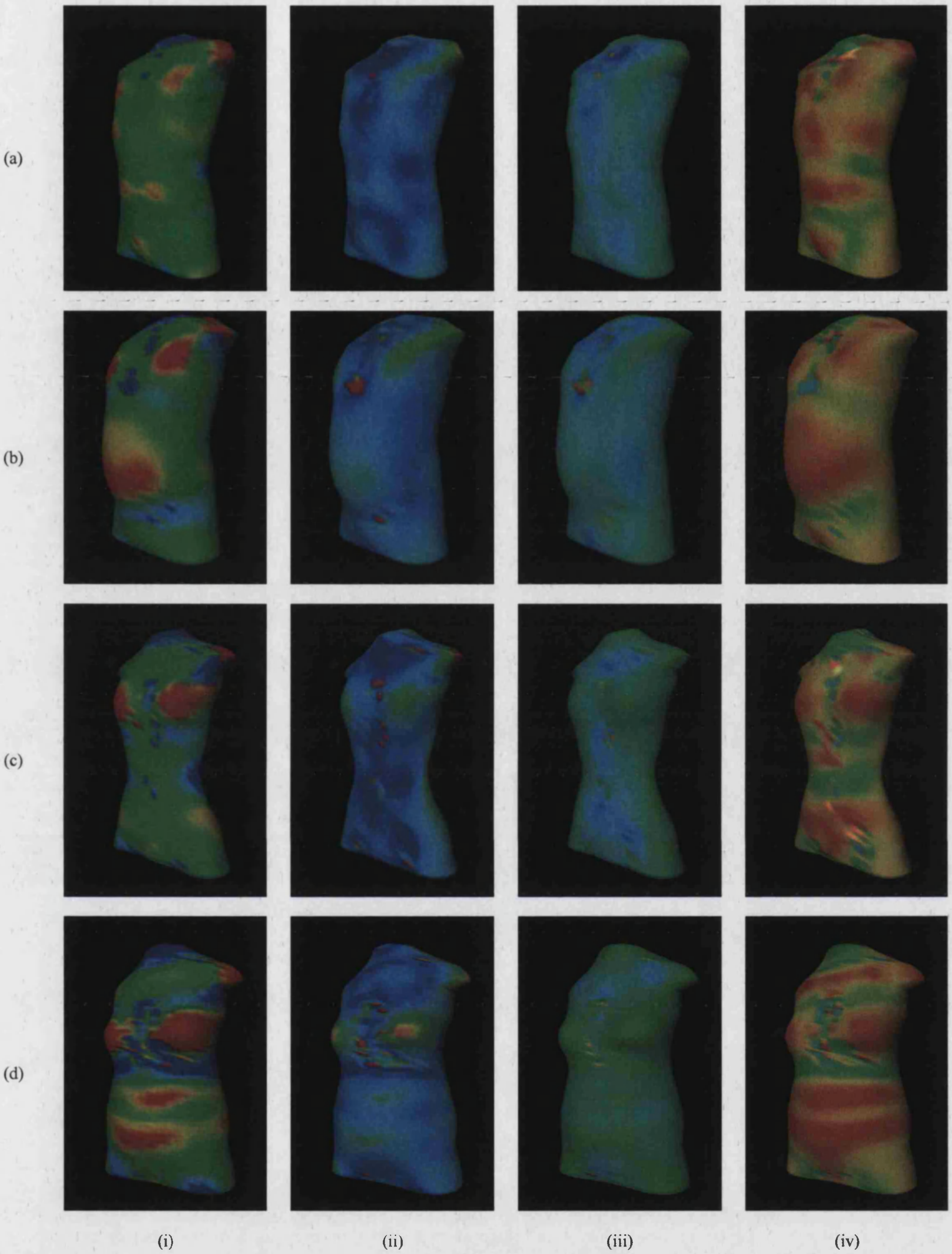


Figure 35: Results for scans of human torsos (front view). By row: Thin male, fat male, thin female, fat female. By column: Gaussian curvature, mean curvature, curvedness, shape index.

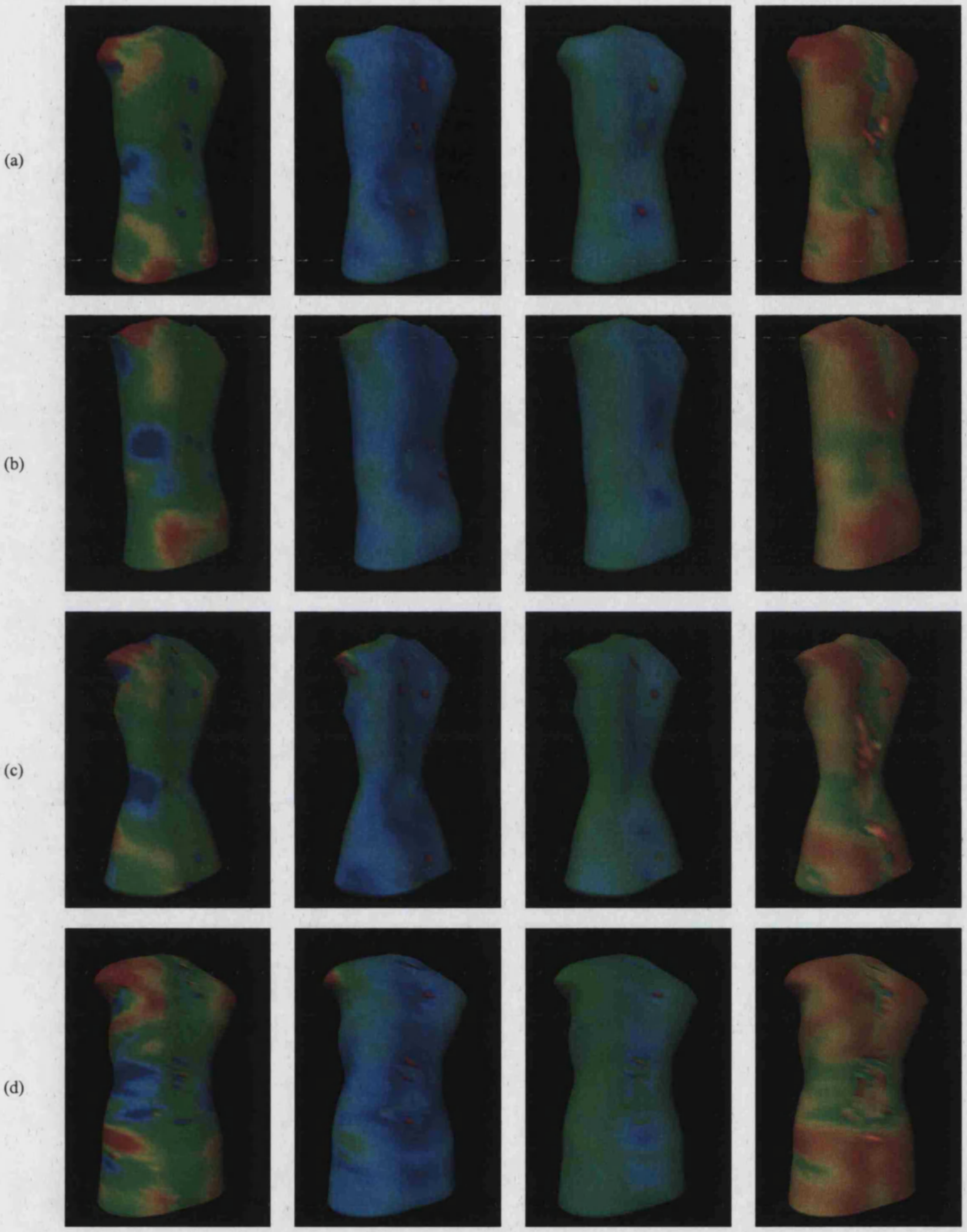


Figure 36: Results for scans of human torsos (rear view). By row: Thin male, fat male, thin female, fat female. By column: Gaussian curvature, mean curvature, curvedness, shape index.

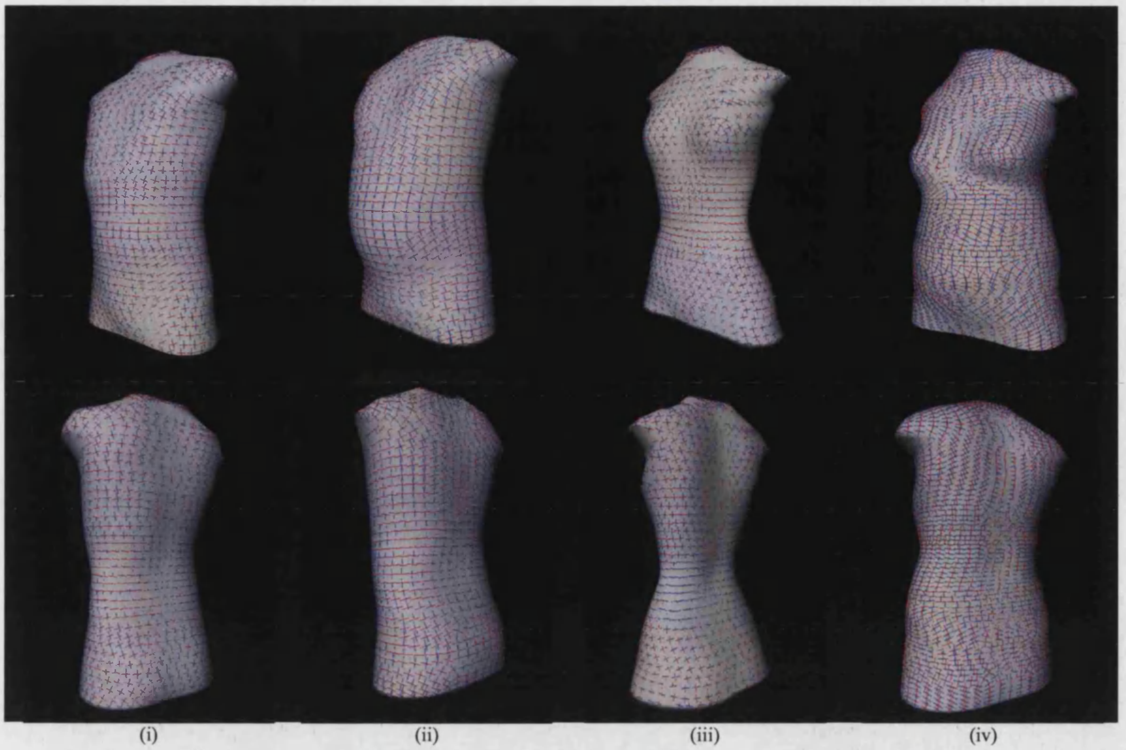


Figure 37: Curvature directions: Thin male, fat male, thin female, fat female

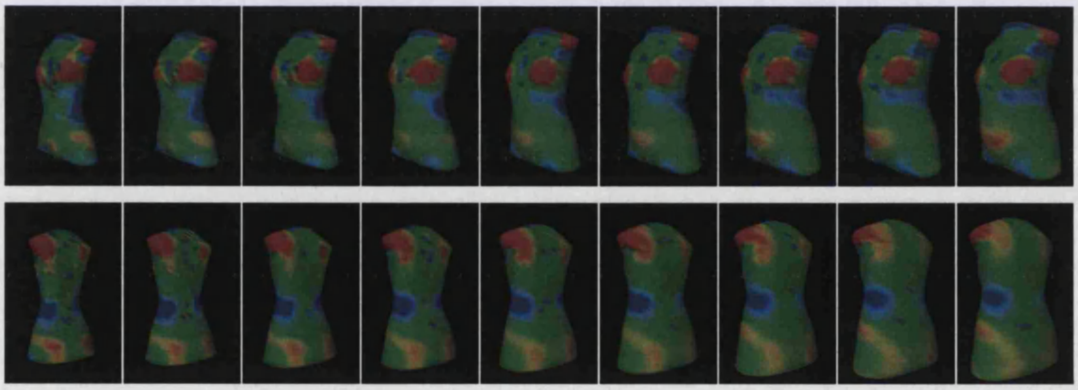


Figure 38: Changes to the Gaussian curvature map as a female torso deforms from a thin to a fat body type. Top row: front view. Bottom Row: Rear View.

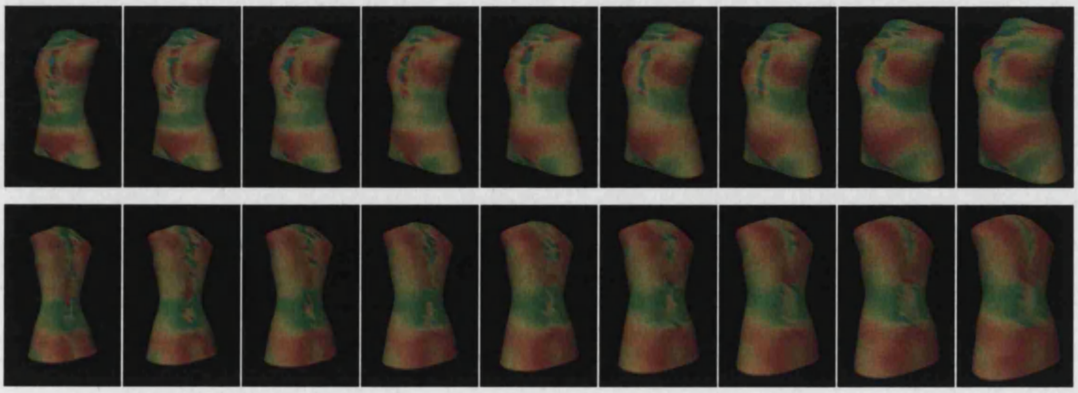


Figure 39: Changes to the shape index map as a female torso deforms from a thin to a fat body type. Top row: front view. Bottom Row: Rear View.

Figure 40: Results for a whole body Hamamatsu scan of a male subject. By row: (a) front view,(b) rear view. By column: Gaussian curvature, mean curvature, curvedness, shape index.

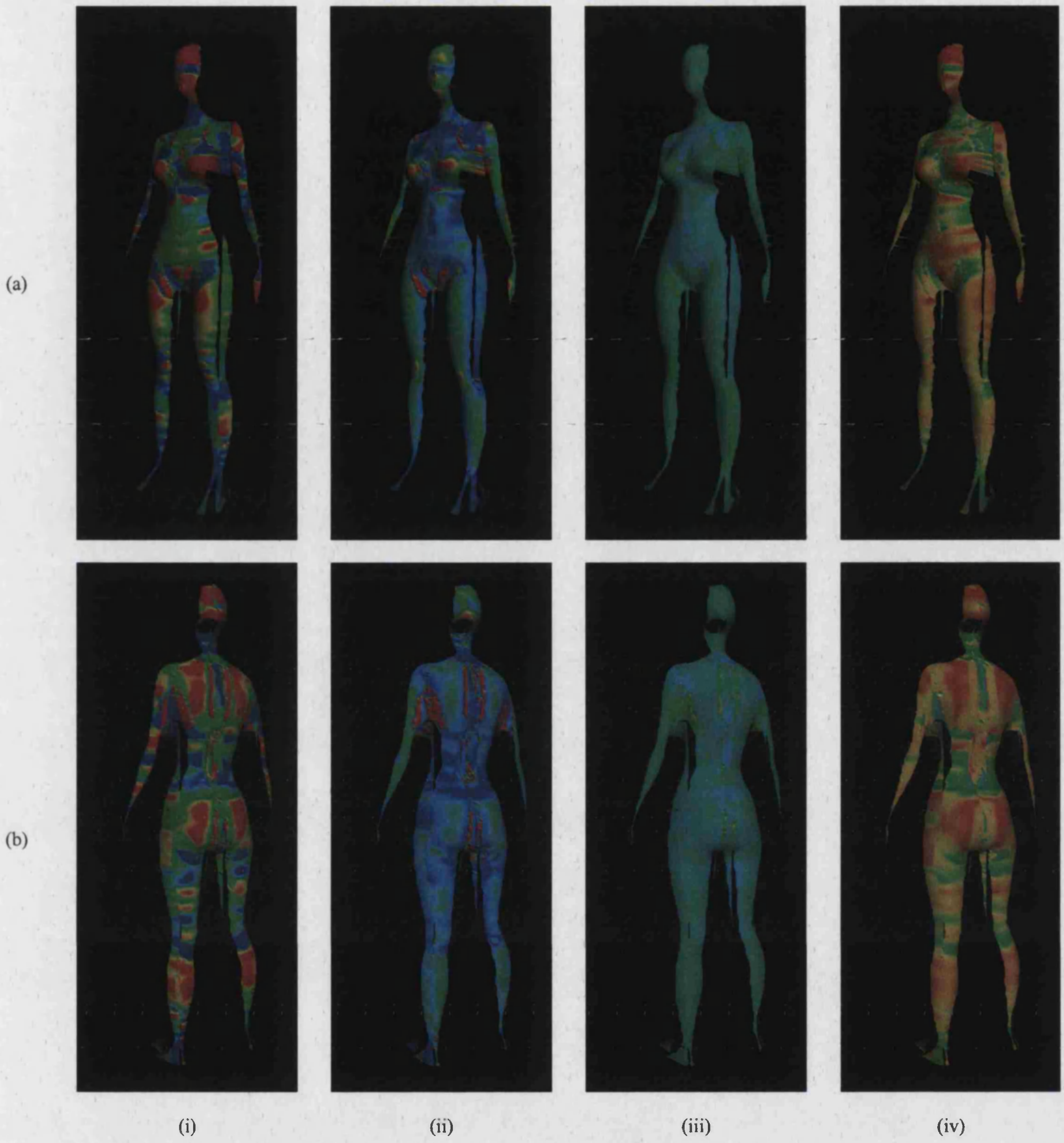


Figure 41: Results for a whole body Hamamatsu scan of a female subject. By row: (a) front view,(b) rear view. By column: Gaussian curvature, mean curvature, curvedness, shape index.

5.5.1 Evaluation Strategy

In order to evaluate the accuracy of the curvature calculation, we have again used the sampled test objects described in section 3.3.1. More specifically, for each test object the steps below were followed:

- For every point P in the point cloud describing the object:
  - A quadric patch was fitted using the minimization described in 4.2.
  - A 'closest' point to P (P_0) was found on the patch using the method described in 4.3.
  - The principal curvatures *of the fitted quadric patch* were calculated at P_0 by forming and solving the minimization problem described in 5.1.2. From them, the Gaussian (K_f) and absolute mean (H_f) curvatures as well as the curvedness (C_f) and the shape index (I_f) were calculated.
  - The principal curvatures *of the test object surface* were calculated at P_0 by using the object's appropriate analytic form as presented in section 3.3.1 and applying to it the generic method for calculating curvatures on implicit surfaces (appendix B4). From them, the Gaussian (K_a) and absolute mean (H_a) curvatures as well as the curvedness (C_a) and the shape index (I_a) were calculated.
  - The following differences:
    - $dK = |K_f - K_a|$
    - $dH = |H_f - H_a|$
    - $dC = |C_f - C_a|$
    - $dI = |I_f - I_a|$

    between the actual curvature properties' values and those obtained by quadric patch fitting were calculated.
- The mean, standard deviation, minimum and maximum of the differences were calculated, and a histogram of the values was built.

The purpose of this exercise is to assess to what extent we may expect our fitted patches on the point cloud to return curvature values that are true to those of the original surface. The next section illustrates the results of this procedure.

5.5.2 Analysis of the Results

The summary of statistics for the metrics defined above is given in Table 6, both for each individual test object and overall for the complete set of them. These values are all around zero (which is the desired target, since these are differences). It must be noted here that some of the sample points of each test object (a maximum of 0.24% in the case of the tooth) were excluded from the calculation of curvature, because the corresponding patches were found to be poorly fitted (i.e. the eigenvalues of the minimization problem were too high). With these points filtered out prior to the calculation of curvatures from the quadric patches, the quality of fit is more than satisfactory for the majority of cases. The only notable exception here is the case of the Tooth surface, where the average difference between the actual and calculated Gaussian curvatures is distinctly higher than in the other cases. A possible explanation is that this particular example is dominated by regions of high curvature, where even small disparities contribute significantly to the average of differences because the values of K do not increase linearly as the surface curvature increases. To check that, we calculated the average of normalised differences between actual and calculated Gaussian curvatures, defined as:

$$\frac{dK}{\sqrt{K_a^2 + \sigma_K^2}} , \tag{5.19}$$

where σ_K^2 is the variance of the (non-normalised) differences.

This normalised measure would indicate better whether the calculated Gaussian curvature K_f is reasonable in feature-rich regions of the surface where $|K_a|$ and $|K_f|$ are both expected to be very large. Indeed, the value of the average of normalised measures for the Tooth surface was found to be 0.00021294, significantly smaller than the value of 5.381708129 for the average of non-normalised differences. This indicates that the regions of high curvature on the particular example consitute a reasonable cause for the unusually high apparent disparity when normalisation is not taken into account.

The histograms of the values summarised in Table 6 are given in Figure 42. It can be easily that in all cases the peaks are at values very close to zero (and, in most cases, narrow), indicating a reasonably good percentage of cases where the curvature calculated using the local surface reconstruction primitives matches the actual curvature value on the object surface.

	dK	*DH*	*dC*	*DI*
Ellipsoid				
Average:	6.45082E-07	0.00118528	10.65046519	0.084997019
Standard Deviation:	8.32148E-06	0.001571175	0.454632609	0.17280295
Minimum:	4.66882E-10	1.17411E-05	9.02297115	6.30753E-05
Maximum:	0.000316576	0.04997927	13.76187916	1.812736031
Tooth				
Average:	5.381708129	0.745446679	4.713942255	0.264045818
Standard Deviation:	41.24083434	0.204273676	0.539743571	0.221535066
Minimum:	5.54457E-05	0.050877793	0.062957148	6.73997E-05
Maximum:	959.8700123	1.242109027	7.023377572	1.403811497
Octahedron				
Average:	0.002625546	0.21949875	3.651390176	0.228548741
Standard Deviation:	0.003211833	0.008729453	0.300122962	0.150952361
Minimum:	0.000495792	0.18262003	0.575675932	0.026264999
Maximum:	0.015915804	0.249987534	5.905569144	1.979440407
Bumpy Sphere				
Average:	0.046280696	0.447780885	4.133774722	0.137845109
Standard Deviation:	0.00377612	0.004821502	0.025691705	0.073781337
Minimum:	0.039076513	0.438137042	4.094856889	4.33343E-05
Maximum:	0.051834016	0.454654566	4.184927494	0.259871241
OVERALL				
Average:	1.357653754	0.353477898	5.787393087	0.178859172
Standard Deviation:	10.31195765	0.054848952	0.330047712	0.154767929
Minimum:	4.66882E-10	1.17411E-05	0.062957148	4.33343E-05
Maximum:	959.8700123	1.242109027	13.76187916	1.979440407
	dK	*dH*	*dC*	*DI*

Table 6: Summary of statistics for the evaluation of curvature calculation.

Finally, in order to enable the reader to assess the validity of the range of values given in Table 6, we present in Table 7 the ranges of actual (analytically calculated) values of the curvature properties for the test objects. These indicate that, overall, the disparities between the actual values of the curvature properties, and those derived from the fitted quadric patches, are indeed reasonably small.

	K	*H*	*C*	*I*
Ellipsoid				
Average:	1.168E-21	-6.572E-11	-15.405626	-0.859822
Standard Deviation:	1.018E-21	2.894E-11	0.2784008	0.0638692
Minimum:	2.5E-22	-1.522E-10	-15.81801	-0.9990025
Maximum:	5.127E-21	-3.2E-11	-14.798294	-0.7205643
Tooth				
Average:	-139870.1	-0.7461796	-0.2673201	-0.5468372
Standard Deviation:	9850498.8	0.2030413	0.5395197	0.2909095
Minimum:	-694303048	-1.2427536	-0.907234	-1
Maximum:	0.3861091	-0.4808104	6.4802861	-1.173E-05
Octahedron				
Average:	0.0026254	-0.2205587	-1.2193049	-0.5461634
Standard Deviation:	0.0032106	0.0092174	0.0206601	0.0794537
Minimum:	0.0004958	-0.25	-1.3238136	-1
Maximum:	0.015625	-0.2102738	-1.1989851	-0.506825
Bumpy Sphere				
Average:	0.0464017	-0.4486339	-0.9277678	-0.8407082
Standard Deviation:	0.0038069	0.0048702	0.0095376	0.0663762
Minimum:	0.0390766	-0.4553826	-0.9422421	-0.9992493
Maximum:	0.0518341	-0.4388056	-0.9107526	-0.7394013
OVERALL				
Average:	-34967.513	-0.3538431	-4.4550047	-0.6983827
Standard Deviation:	2462624.7	0.0542822	0.2120295	0.1251522
Minimum:	-694303048	-1.2427536	-15.81801	-1
Maximum:	0.3861091	-3.2E-11	6.4802861	-1.173E-05
	K	*H*	*C*	*I*

Table 7: Ranges of the actual (analytically calculated) values of the curvature properties for the test objects.

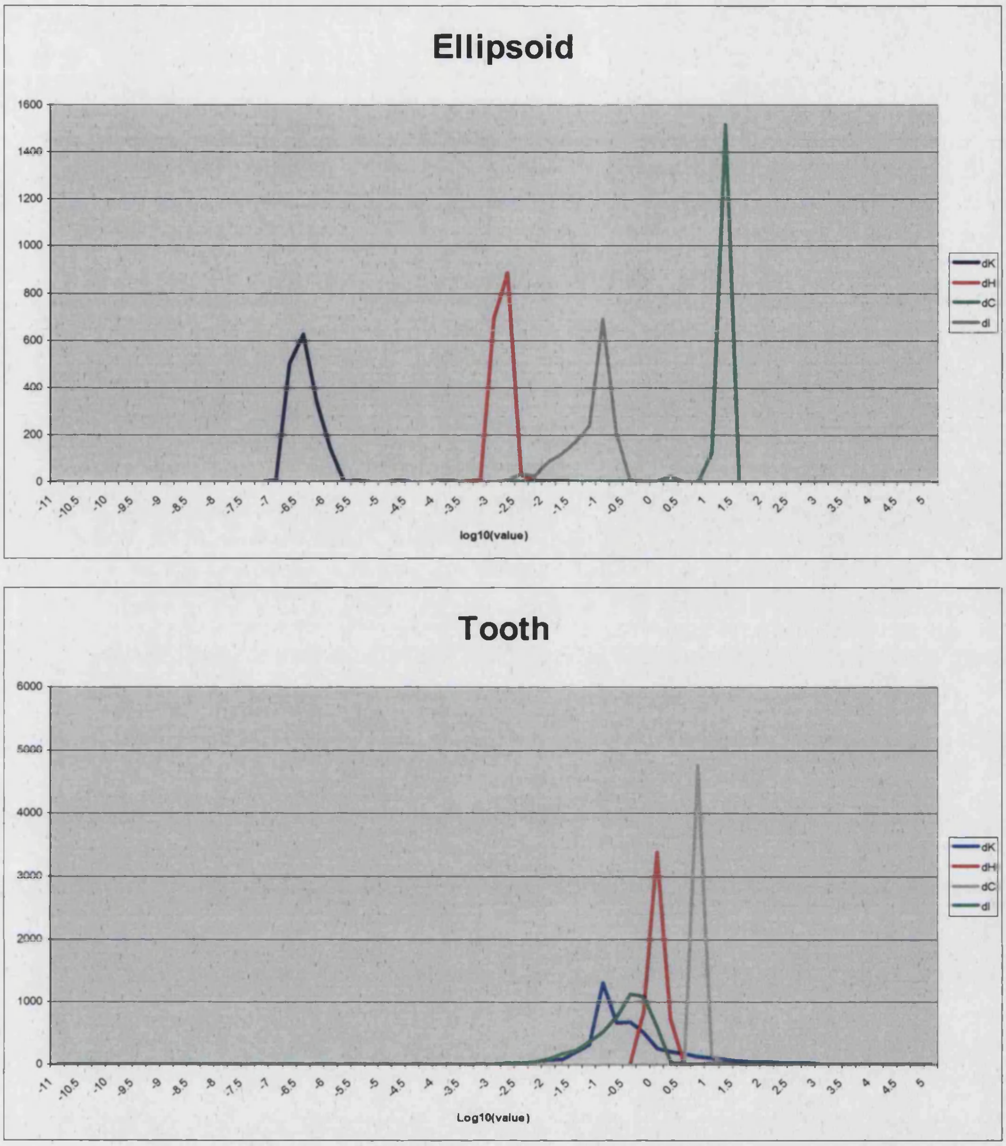


Figure 42 (part 1 of 2):Histograms of differences between the actual values of curvature properties on the test objects, and the values obtained via the fitted quadric patches.

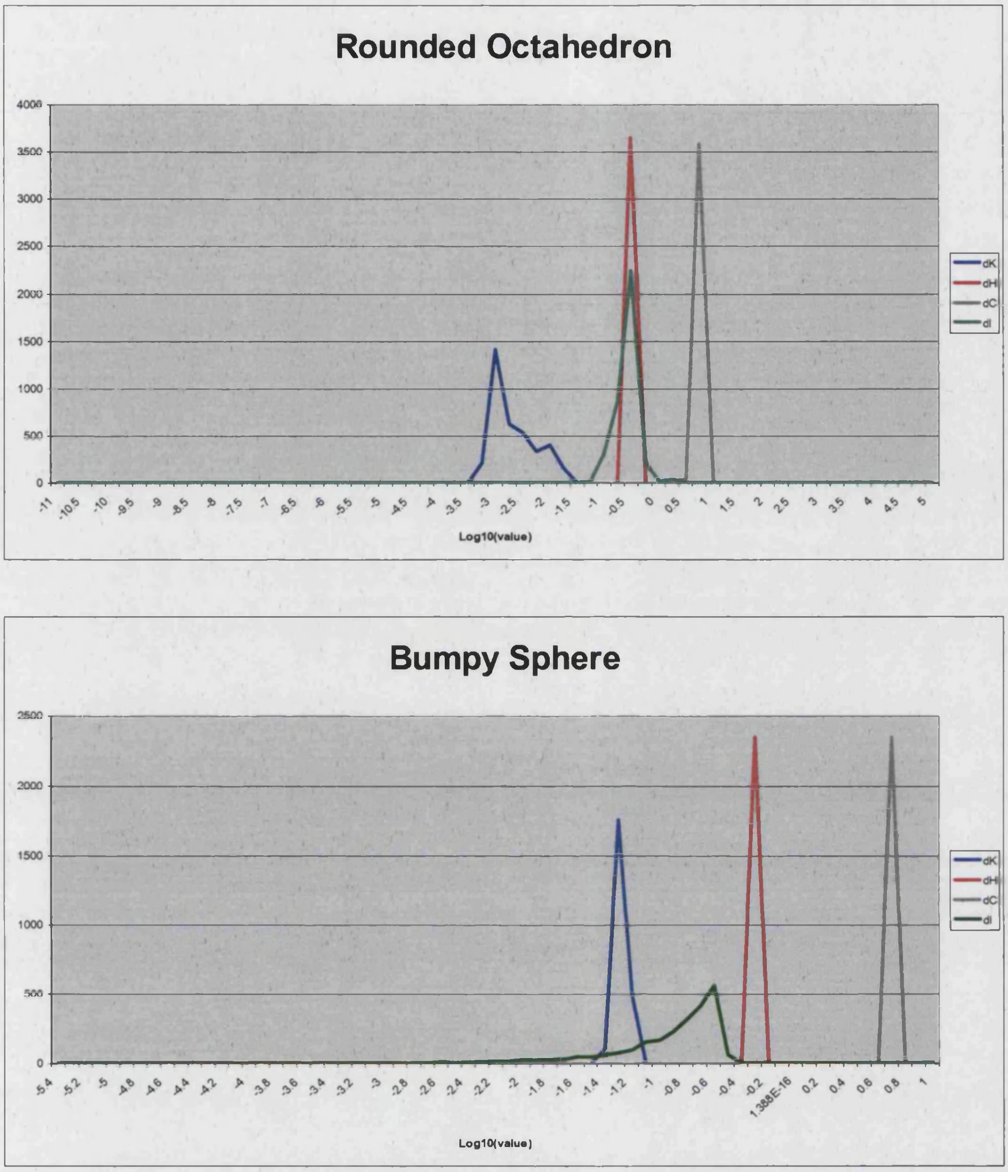


Figure 34 (part 2 of 2):Histograms of differences between the actual values of curvature properties on the test objects, and the values obtained via the fitted quadric patches.

5.5.3 Noise

In order to evaluate the behaviour of the proposed method with respect to noise, we have used it to generate the curvature property maps for the noisy ellipsoid used in the experiment presented in section 4.4.4. The Gaussian curvature maps of these noisy ellipsoids as generated by our implementation are shown in Figure 43.

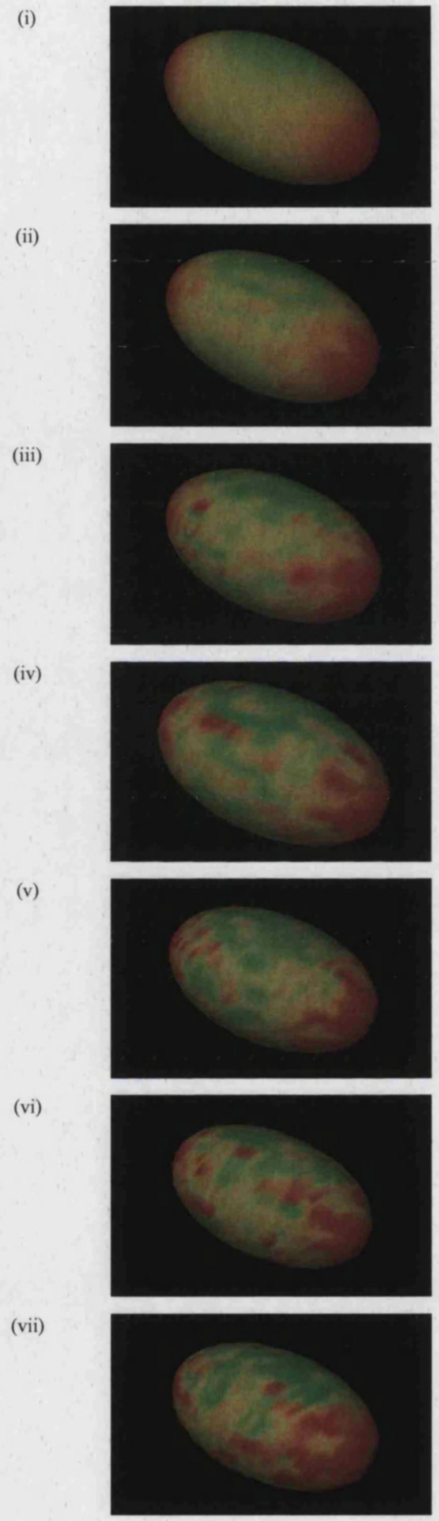


Figure 43: Behaviour of the curvature calculation algorithm on an ellipsoid with varying levels of uniform noise with amplitude ranging (top-to-bottom) from no noise at all to a level of 50% of the sampling resolution, at equal increments of 1/12th of the resolution.

It is worth noting on those images that although the noise causes the curvature values to fluctuate across the surface, the colour range is from green (zero values) to yellow and red (increasing positive). This means that despite the levels of noise, the quadric fitting process is still robust enough to fit patches that are locally synclastic, as the shape of the ellipsoid dictates.

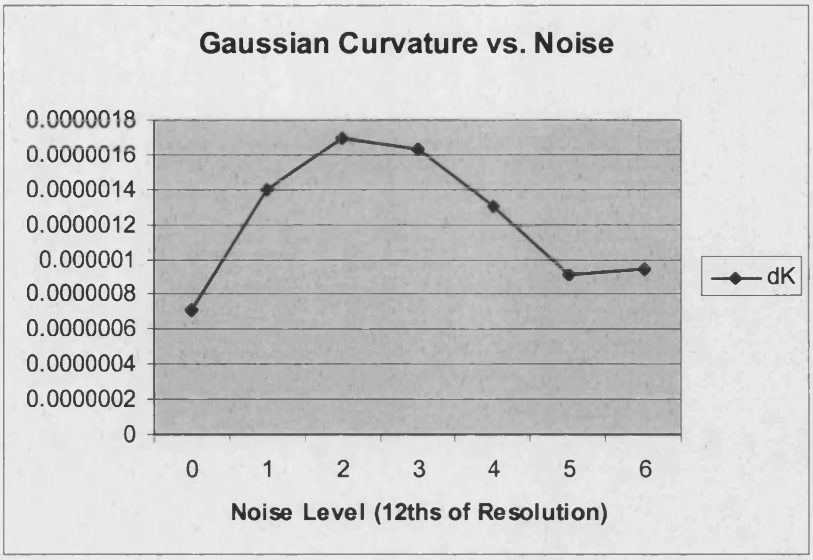
For a numerical evaluation of these results a scheme was followed that is similar to the one described in section 5.5.1. The values returned by the curvature calulation scheme from the fitted quadric patches were compared the against the actual values calculated analytically from the implicit form of the ellipsoid. The means of the absolute differences were calculated for each ellipsoid in the noise sequence, and the results are presented in Table 8, with the relevant graphs given in Figure 44 and Figure 45.

Noise level	dK	dH	dC	dI
0	0.000000706	0.001204181	10.652667827	0.084404744
1	0.000001397	0.001739198	10.692499923	0.088451751
2	0.000001697	0.001747102	10.680099264	0.085283996
3	0.000001632	0.001726713	10.671149602	0.084569160
4	0.000001302	0.001557857	10.659473242	0.080409717
5	0.000000905	0.001559081	10.674280838	0.075840766
6	0.000000942	0.001398561	10.656464429	0.079283045

Table 8: Mean values of the absolute differences between calculated and actual curvature property values for the test ellipsoid with increasing levels of noise.

These figures and graphs suggest that the initial introduction of noise does indeed (negatively) affect the quality of the results for the calculation of Gaussian and mean curvature, whereas the introduction of further noise causes the disparities to fluctuate around some plateau. This is most likely caused as a result of the indirect "smoothing" caused by the least-squares nature of the local patch fitting. On the other hand, the curvedness and shape index do not seem to be dependent on the noise level.

Finally, an experiment was carried out in order to assess the way in which curvature property calculations are affected by smoothing. The same sequence of shapes was used as for the corresponding experiment detailed in section 4.4.4 (a noisy ellipsoid subjected to increments of Gaussian smoothing).



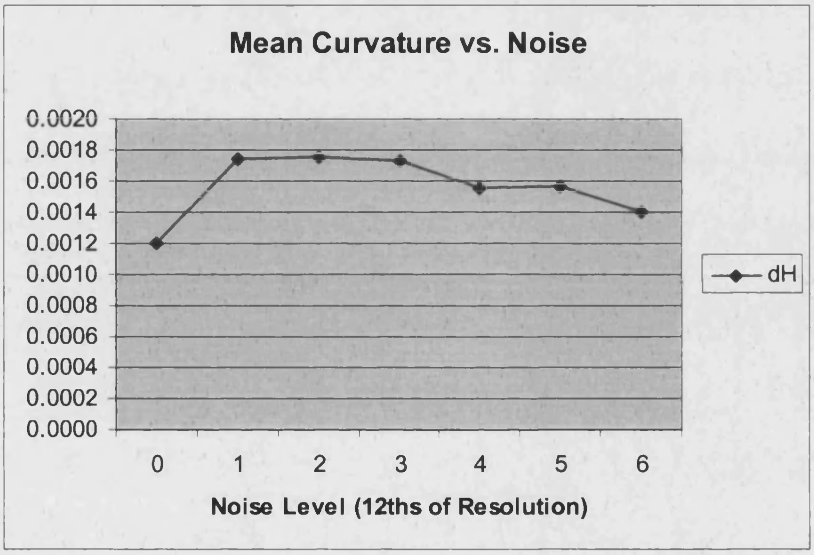
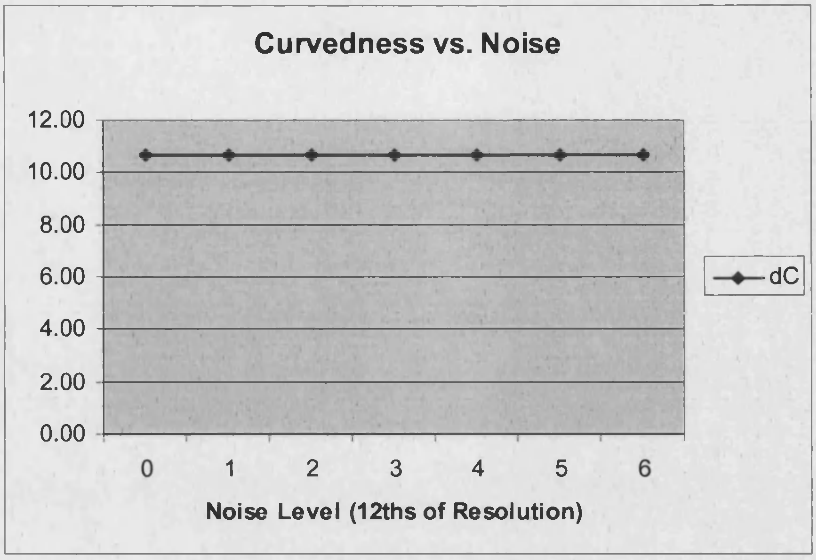


Figure 44: Graphs for the mean values of the absolute differences between calculated and actual values of the Gaussian and mean curvatures for the test ellipsoid with increasing levels of noise.



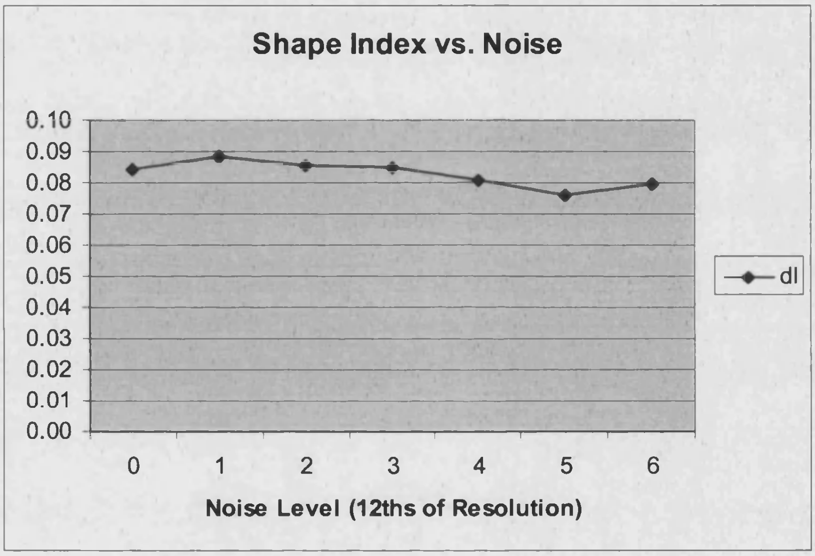


Figure 45: Graphs for the mean values of the absolute differences between calculated and actual values of the curvedness and shape index for the test ellipsoid with increasing levels of noise.

The findings of this latter experiment (disparities between actual and calculated curvature values in Table 9, and associated graphs in Figure 46 and Figure 47) are compatible with those of the smoothing experiment presented in section 4.4.4 and the noise experiment presented earlier in this section. Initial smoothing improves significantly the quality of the results for the Gaussian and mean curvature. Further smoothing provides a further, but less significant improvement. The results for the curvedness and shape index appear to be more stable and not very dependent on the level of smoothing.

Passes	dK	dH	dC	dI
0	0.000000706	0.001204181	10.652667827	0.084404744
1	0.000000250	0.001038403	10.566133241	0.065185142
2	0.000000245	0.001036965	10.567682303	0.065132070
3	0.000000239	0.001032799	10.567405812	0.069564818
4	0.000000233	0.001028776	10.566573787	0.074771824

Table 9: Mean values of the absolute differences between calculated and actual curvature property values for a noisy ellipsoid with increasing levels of smoothing.

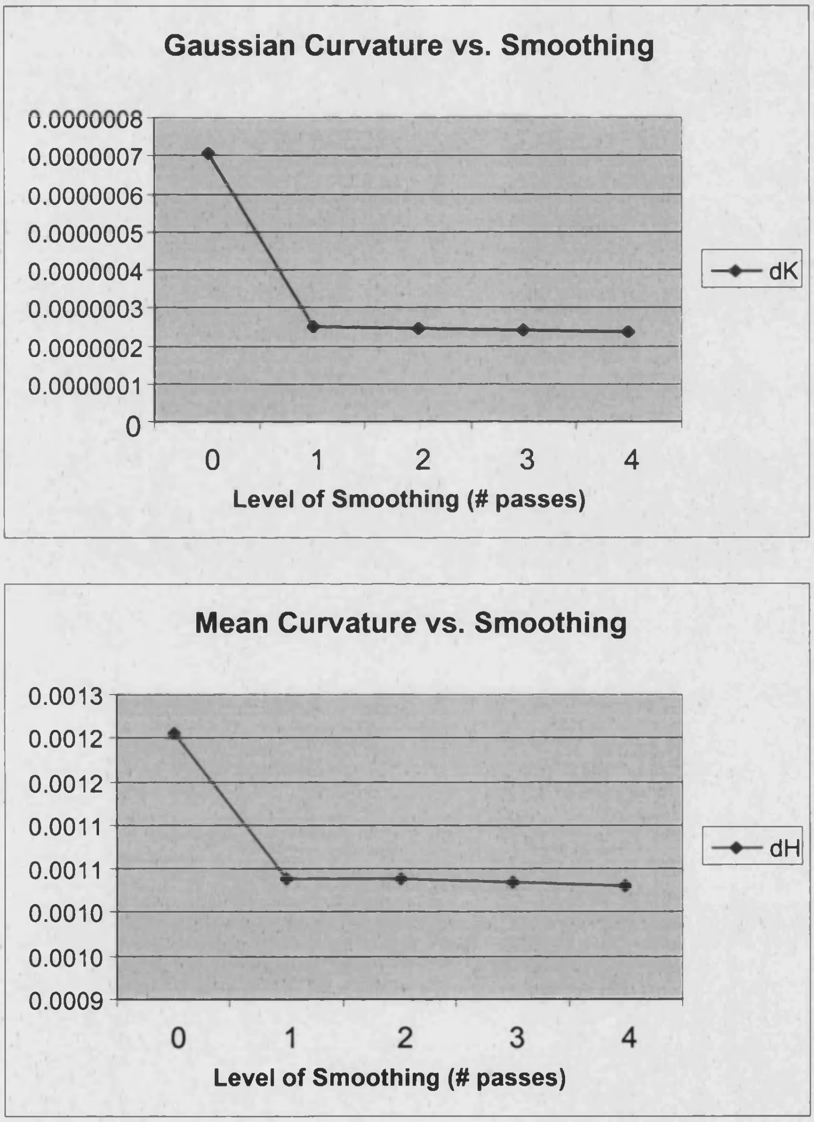
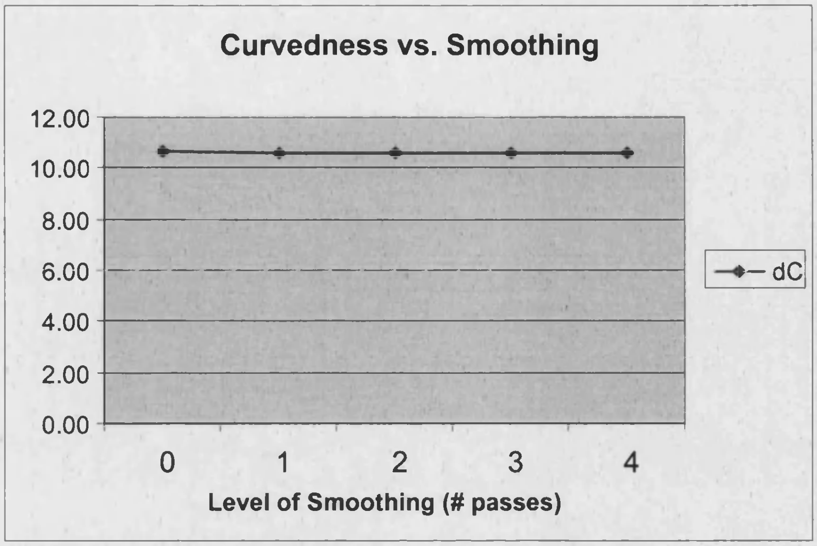


Figure 46: Graphs for the mean values of the absolute differences between calculated and actual values of the Gaussian and mean curvatures for a noisy ellipsoid with increasing levels of smoothing.



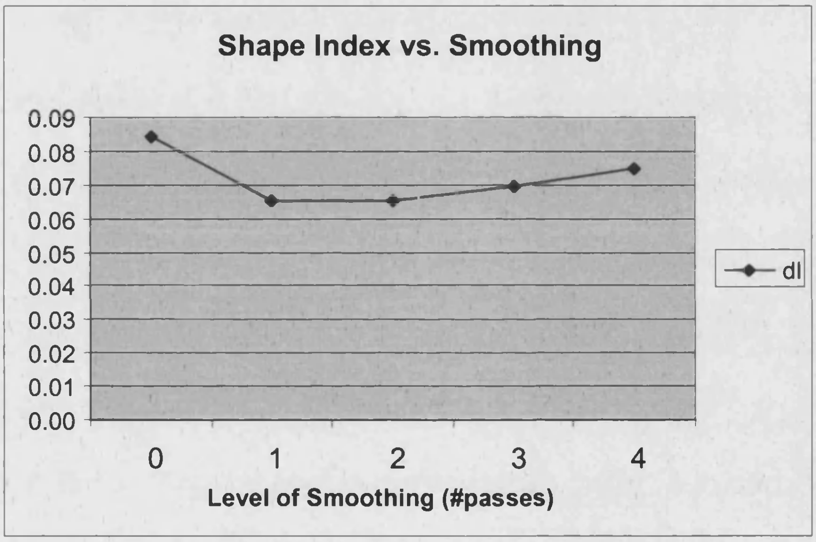


Figure 47: Graphs for the mean values of the absolute differences between calculated and actual values of the curvedness and shape index for a noisy ellipsoid with increasing levels of smoothing.

At this point we have shown that the proposed method for curvature calculation works and is accurate, robust and reliable. In the next chapter we can therefore proceed with confidence to consider the calculation (detection and location) of features dependent on the curvature properties.

5.6 Conclusions

The visual results from the methods presented in this chapter appear to tally with our expected perception of the shapes involved. Moreover, the numerical evaluation suggests that the surface curvature properties calculated are true to the original surfaces, deformable in accordance to surface deformations, and with sufficient resistance to noise. The outcome so far provides a set of information that describes the curvature, differential geometry and shape properties of the surface in a manner that:

- Enables us to classify each point in the original point cloud according to its local surface shape properties (e.g. as synclastic, anticlastic or developable)
- Enables us to produce rich visualisations of that classification (curvature maps).
- Enables further processing of the 3D data, specifically for the purpose of detecting curvature-related deformable surface features, such as ridges and flexcords.

This latter detection is the subject of the next chapter.

6 Feature Detection

This chapter is concerned with the detection of features on a surface using the differential geometry properties discussed in chapter 5. More specifically, it is focused on the issues arising from the representation of the surfaces used within the context of this research (i.e. because the surfaces available are represented as clouds of sample points on the original object's surface, rather than as continuous surfaces). More specifically, sections 6.1, 6.2, 6.3 and 6.4 discuss umbilic points, parabolic lines, ridges and flexcords, and present ways of detecting some of them using curvature properties. Section 6.5 discusses a potential alternative approach using focal surfaces, and the difficulties of implementing it in practice. In section 6.6 the results of the detection techniques discussed throughout this chapter are presented, along with an appropriate quantitative evaluation.

6.1 Umbilic and Minimal Points

On a surface in R^3, we define as *umbilic points* those points where the two principal curvatures coincide. This definition obviously includes the trivial case of planar points, where both curvatures are equal to zero. Non-planar umbilic points are known as *ordinary umbilics*, and it can be proven ([Port94], p184) that they are isolated on a surface.

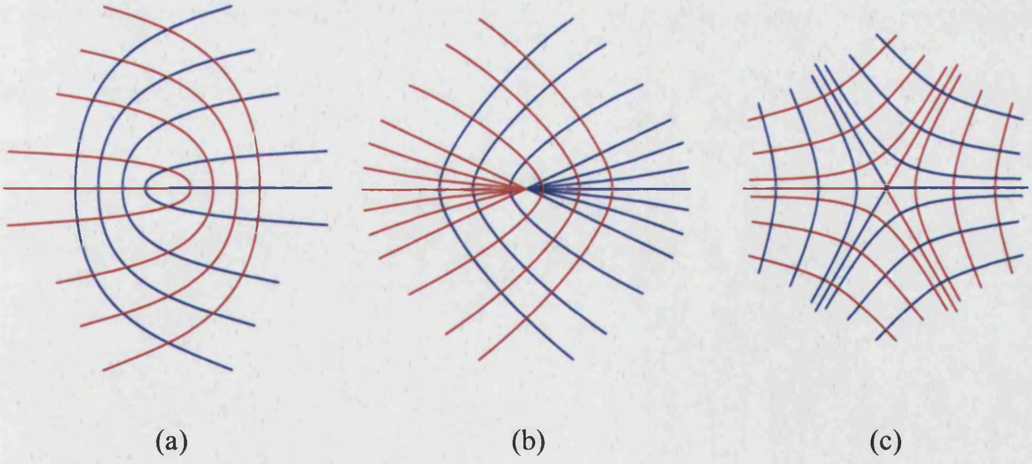

(a) (b) (c)

Figure 48: Types of ordinary umbilics, and the behaviour of lines of curvature around them (reproduced from [Port94]). (a) Lemon, (b) Monstar, (c) Star.

Ordinary umbilics are of particular interest because they constitute singularities that define the topology of a surface. The behaviour of lines of curvature (as described on section 5.4) on and around these points differs significantly from their normal behaviour away from umbilics. In fact, umbilic points are classified according to this behaviour (Figure 48). More importantly, there are strong links between the occurrence of umbilics on a surface and the configuration on it of other important differential geometry features such as ridges or flexcords (all these are covered in detail in [Port94], but see also [Koen90] and [BrGT96]).

The simple definition of umbilic points implies that, in theory, these points are very easy to detect once the principal curvatures have been obtained. However, in our case the detection is somehow more difficult. This is because we only have specific sample points of a surface, at which we calculate the principal curvatures. Unless such a datapoint happens to be an actual umbilic, it cannot be detected by using a naïve implementation that checks for the condition $\kappa(1) = \kappa(2)$.

What we have done in practice was to set a threshold T_u and then check that the difference between the values of the principal curvatures is lower than the threshold, i.e:

$$|\kappa(1) - \kappa(2)| < T_u \quad . \tag{6.1}$$

However, such an approach only partially solves the problem of detecting umbilic points on a sampled surface. It certainly allows us to characterise data points as being in the close proximity of a possible umbilic point (we will hereafter refer to these points as *near-umbilics*). The two problems remain of: (a) setting the value of the threshold such that false positives are minimized, and (b) pinpointing the exact (or, at least, more accurate) location of the umbilic point on the surface.

These two problems are strongly interlinked. We have experimented with various values of T_u and generally managed to find some that seem to generate sufficiently small clusters of usually 3-4 near-umbilics (results are illustrated in section 5.5). However, the second problem is more difficult as it is always highly likely that the actual umbilic point will be located *between* our data points (i.e. *outside* our data set). A possible solution may involve the use of the locally fitted quadric patches. These provide a continuous analytic representation of the surface around each data point (i.e. also between the data points), and may therefore contain detectable umbilics. This approach is not particularly straightforward (especially since it might involve interpolation between two or more quadric patches), however it appears to constitute a worthwhile direction for future work.

It is also worth noting the implications of detecting umbilics on the labelling of principal curvatures as "red" and "blue" and the separation of the corresponding the curvature vector ($[\kappa(1), \kappa(2)]$) fields. The method described in section 5.4 excludes umbilic points from the label propagation process, and it is in fact around umbilics that the label propagation tends to fail. It appears that a precise, localised detection of umbilic points could be achieved by using a detailed continuous representation of the surface around the umbilics. However, that representation would have to be suitable for enabling the local profile of the lines of curvature (as shown in Figure 48) to be analysed. Such a representation might also enable us to resolve the problem of separating the curvatures correctly across the entire surface (currently believed to be difficult or unsolvable [Koen90]), albeit probably not easily, because of the local and global consistency problem already illustrated in the previous chapter.

Finally, there is another type of feature that is defined similarly to umbilic points. These are the *minimal points*, i.e. points on the surface where the two principal curvatures have opposite sign, but same magnitude. The detection of such points from our data demonstrates the same set of problems as with umbilics. It is not sufficient to check the that the condition $\kappa(1) = -\kappa(2)$ is true, as this would only happen in the unlikely case that the surface is sampled exactly on the minimal point. Therefore, the best that can be done is the use of a small threshold on the value of $\kappa(1) + \kappa(2)$, which enables us to detect *near-minimal* points. Furher refinement of the results would only be possible by means of a local interpolation of quadric patches and an iterative search for the precise location of the minimal point.

6.2 Parabolic Lines

Parabolic lines have already been mentioned in section 5.2, in the context of the behaviour of the Gaussian curvatures. Their definition is also simple: they are the lines on the surface that are made up of points on which one of the two principal curvatures vanishes. Because the Gaussian curvature is the product of the two principal curvatures, it will also vanish at parabolic points. Therefore, we can detect parabolic points by searching across our surface for the zero-crossings of the Gaussian curvature (Figure 49).

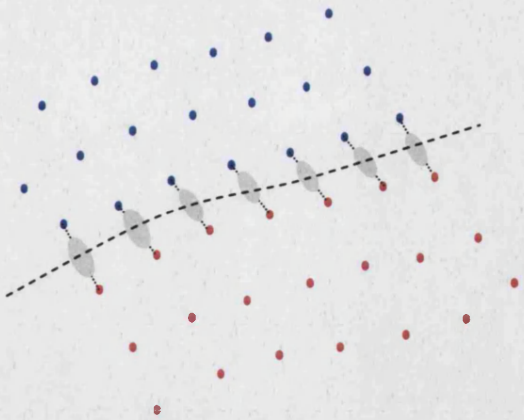


Figure 49: Detection of possible near-parabolic points by searching for zero-crossings of the Gaussian curvatures.Points shown in blue have negative Gaussian curvature, whereas points shown in red have positive Gaussian curvature. The grey regions indicate the potential location of parabolic points,and the dashed line illustrates the possible location of the parabolic line.

The locations of the possible parabolic points do not correspond to sample points of our data set. This means that what we actually detect are *near-parabolic* points, i.e. we can classify points in our data set as being in the proximity of a parabolic line. It is important to note that such a classification is independent of the type of quadric patch around each point.

As with the case of umbilics, because our surface is represented by sample points, locality of response is not guaranteed. Achieving accurate detection of the parabolic lines on a sampled surface such as a human body would involve some form of interpolation of the local quadric patches or some form of global surface reconstruction involving surface primitives of a higher order. This is important because it is impossible to detect isolated parabolic lines on a quadric surface, for the simple reason that there aren't any. It is shown in [Gass83] that a quadric surface is either parabolic throughout (i.e. when it is developable, with $K = 0$ everywhere), or has no parabolic points at all (i.e. when it is synclastic, with $K > 0$ everywhere, or when it is anticlastic, with $K < 0$ everywhere).

However, obtaining better representations of parabolic lines should be relatively easy. The detection of continuous runs of near-parabolic points and the use of an interpolation scheme for generating parametric curve representations such as a B-spline is straightforward and has already been used extensively in 2D imaging for the generation of parametric representations of edges from detected edgels [Guta99][DeMi02]. An implementation of such an approach has therefore not been carried out, as it is not within the scope of this work.

6.3 Ridges

Of all the surface features discussed in this chapter, ridges are deemed to be the most important – and they have certainly received most of the attention during this research. Their importance is threefold. Firstly, they are important for determining the topology of the surface. Secondly, they are known to be relatively invariant to deformations of the surface, and most specifically they are *de-formable.* As the surface deforms, they deform along with it and they only change their configuration if singularities occur [PoPu00]. They could therefore be used for classifying surface shape (a requirement of our hypothesis). Thirdly, their occurrence on a surface seems to affect the way in which we perceive surface shape [KoVD98]. This adds to their importance in the case of studying 3D representations of human bodies, because they have the potential of enabling classifications that are intuitive and compatible with our common experience in characterising body shape.

The commonly understood concept of a ridge (as it appears, for example, in geography) is immediately applicable for any height function. We normally understand ridges as the loci of changes in the sign of the gradient of the surface. As we traverse, e.g. a mountainous terrain, we cross a ridge at the point where we start moving downhill rather than uphill. In the geometrical sense, a 'rut' (crossed when we start moving uphill rather than downhill) is equivalent in that it still constitutes a change in the sign of gradient. If we change the sign of the height function (i.e. turn the landscape upside-down), ridges become ruts and vice-versa.

However, in three-dimensional differential geometry there is a different and more general definition of ridges: they are defined as extremities in the principal curvatures. This is necessary because in general there is no preferred direction in which to measure 'height' across the entire surface, and only the local surface normal can be used as a guide for defining *intrinsic* rather than extrinsic features.

More specifically, the definition of a ridge point is as follows (Figure 50): Let S be a surface in R^3 and P a point on S. Let $\kappa(1)$, $\kappa(2)$ be the principal curvatures of S at P; $d(1)$, $d(2)$ the corresponding principal curvature directions; and $l_1(u)$, $l_2(w)$ the corresponding lines of curvature (represented as parametric curves in R^3) on S that cross at P. Suppose further that:

$$l_1(u_0) = l_2(w_0) = P \quad , \tag{6.2}$$

for certain values u_0, w_0 of the parameters. The curvatures of l_1, l_2 can then also be represented as parametric functions $c_1(u)$, $c_2(w)$, and in that case:

$$c_1(u_0) = \kappa(1) \quad , \text{ and} \tag{6.3}$$
$$c_2(w_0) = \kappa(2) \quad . \tag{6.4}$$

We then call P:

(i) a *ridge point of the curvature of* $\kappa(1)$, if c_1 has a local extremum on u_0.
(ii) a *ridge point of the curvature of* $\kappa(2)$, if c_2 has a local extremum on w_0.

Simply phrased, this definition means that P is a point of a ridge of a principal curvature of *a particular* colour, if the curvature *that colour* is extremal in the direction of *that* curvature. More specifically, if the (magnitude of) the curvature is locally maximum at P, then we call it a *maximal ridge*, whereas if the (magnitude of) curvature is locally minimum at P, then we call it a *minimal ridge*. Using the typical red/blue terminology, we can say that P is a point of a *blue ridge*, if it is the *blue* curvature that is extremal at P. Of course, the definition goes vice-versa for *red ridges*.

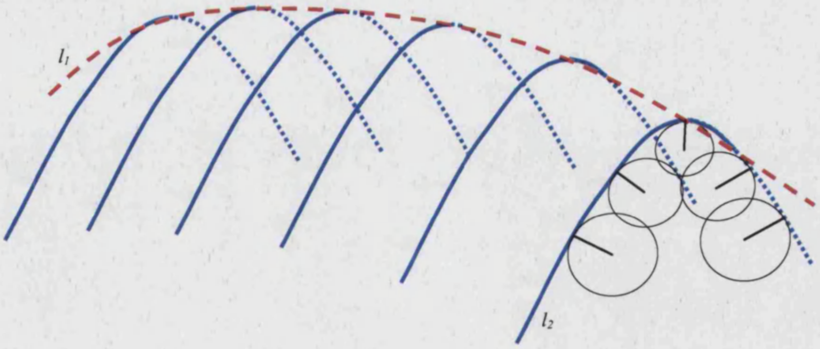


Figure 50: Illustration of the definition of ridges. The dashed red line (l_1) is a blue ridge - because the blue principal curvature is maximal along the direction of the blue principal curvature. The circle of curvature (shown here in black) along the blue line of curvature (l_2) reaches its minimum radius (and, subsequently, its maximum curvature) at the intersection with l_1, therefore l_1 is a maximal ridge.

With this definition, the equivalent concept for 2D curves is the *vertex* of a curve, i.e.

the point on a curve where the corresponding curvature is extremal. This is illustrated in Figure 51, and for clarity the circles of curvature along with their radii are also depicted.

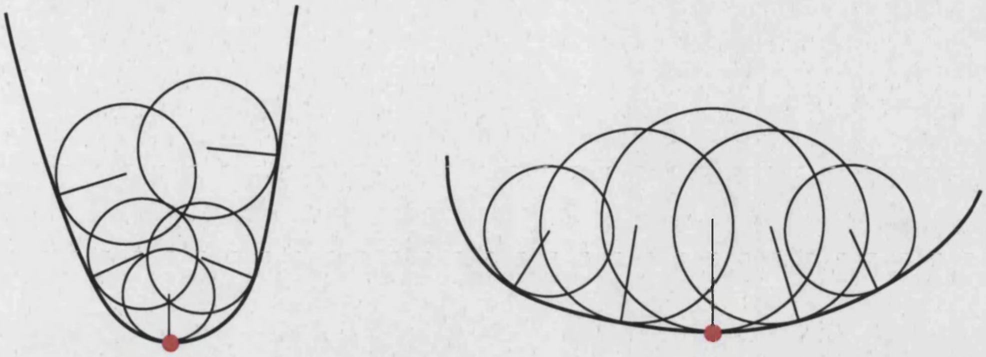


Figure 51: Vertices on a curve. Left: Maximal vertex (radius of circle of curvature becomes minimum, therefore curvature becomes maximum). Right: Minimal vertex (radius of circle of curvature becomes maximum, therefore curvature becomes minimum). The equivalent in 3D are the ridges as defined in this section.

6.3.1 Detection of Ridges

For our body data, we want to determine for each member of a point cloud whether it belongs to a ridge or not. By following the definition given above, formulating an algorithm is fairly straightforward. The main problems that need to be resolved for the algorithm to be applicable to our data are related, once again, to the fact that we only have samples of the surface, and not the entire surface. More specifically, to determine whether a (non-umbilic) point P in our data set can be classified as a ridge, we need to determine:

(a) which neighbouring points P_1 and P_2 to choose to compare their principal curvature magnitudes against those of P, and

(b) which of the curvatures of P_1 and P_2 are "red" and which are "blue", i.e. how they correspond to the lines of curvature that are determined by the principal directions of curvature of P.

Our approach is illustrated in Figure 52 and is not dissimilar to the one followed by Canny [Cann86] for edge detection in 2D images: it involves a non-extremal (in general, but in practice non-maximal) suppression step as well as a hysteresis thresholding step, as will shortly be described.

The first step of the detection is the selection of the appropriate neighbours. Let κ_r and

κ_b be the red and blue principal curvatures of P respectively, and v_r, v_b the corresponding eigenvectors.

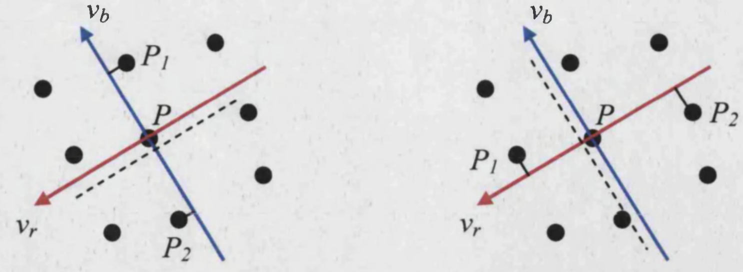


Figure 52: Ridge detection on the discretised data.Left: Blue ridge test (the values of the blue curvatures of P, P_1, P_2 are compared). Right: Red ridge test (the values of the red curvatures of P, P_1, P_2 are compared). In both cases, the dashed line illustrates the approximate direction of the potential ridge.

Suppose we want to look for blue ridge points. We choose P_1 and P_2 to be on *opposite* sides of v_r (to be precise, on opposite sides of the normal plane at P in the direction v_r since the neighbours are generally not co-planar). They must also be immediate neighbours of P. If there is connectivity information, this is straightforward. Otherwise, a proximity criterion must be set, following one of the approaches already discussed in section 4.1.1. Finally, from each side of v_r we narrow the search further, by choosing those points that are closest to the direction v_b (Figure 52, left). This should narrow the search down to no more than two points, which we label P_1 and P_2.

The second step is where we label the curvatures of P_1 and P_2 as "red" and "blue". To do that we employ the label propagation method described in section 5.4, only this time we do not need to propagate across the entire surface because we are only interested in the lines of curvature around P. Therefore, we set P to be the seed, and propagate only to P_1 and P_2. We can therefore now use the notation κ_{1r}, κ_{1b} for the red and blue curvatures of P_1, and κ_{2r}, κ_{2b} for the red and blue curvatures of P_2.

The third step is the non-extremal suppression, whereby P is classified as a ridge or non-ridge. To effect this classification, we decide as follows:

- If $|\kappa_b| > |\kappa_{1b}|$ and $|\kappa_b| > |\kappa_{2b}|$, then P is a *maximal blue ridge point.* (6.5a)
- If $|\kappa_b| < |\kappa_{1b}|$ and $|\kappa_b| < |\kappa_{2b}|$, then P is a *minimal blue ridge point.* (6.5b)

In order also to determine whether P is a red ridge point, we first need to repeat the first step for the red direction, i.e. choose neighbours P_1 and P_2 on opposite sides of v_b that are closest to v_r (Figure 52, right). Once chosen, we repeat the second step (label propagation), and then at the third step we compare red curvatures, i.e:

- If $|\kappa_r| > |\kappa_{1r}|$ and $|\kappa_r| > |\kappa_{2r}|$, then P is a *maximal red ridge point.* (6.6a)
- If $|\kappa_r| < |\kappa_{1r}|$ and $|\kappa_r| < |\kappa_{2r}|$, then P is a *minimal red ridge point.* (6.6b)

It must be noted here, that for every new P in the set that we investigate, we reset the initial (seed) curvature labels arbitrarily, since we are only interested in the local configuration of curvature lines around each point. As a result, after the process has been repeated for every P, there is no guaranteed consistency of the red and blue labels (such a consistency would be desirable but is so far unachievable as discussed in section 5.4). Therefore, at the end of the process we do not retain the colour label information associated with the ridges – this information is only generated to facilitate the detection. We do, however, retain the rest of the information, i.e. whether P is a minimal ridge point or a maximal ridge point. We also retain the value κ' of the principal curvature of P that was used to classify it as a ridge point, i.e. κ_b if criterion (6.5) was true, or κ_r if criterion (6.6) was true.

After all the points have been investigated and classified a fourth, optional step, may follow. This is a thresholding step with hysteresis, and is applied on the absolute value of principal curvature κ' (see above) that effected the classification of P as a ridge point. This is carried out as follows for a maximal ridge point (for details, see [Cann86]):

(a) Upper (T_u) and lower (T_l) thresholds are set.

(b) For every point P in the data set that has been classified as a *maximal* ridge point:

- (i) If $|\kappa'| > T_u$, then P is (permanently) classified as a maximal ridge point.
- (ii) If $|\kappa'| < T_l$, then P is declassified.
- (iii) If $T_l < |\kappa'| < T_u$, then P is (permanently) classified as a maximal ridge only if it is adjacent to a maximal ridge point that is also permanently classified.

In practice, both (b)(i) and (b)(ii) above are easy and straightforward to implement, and (b)(iii) involves an iterative propagation loop that terminates when there are no more points to investigate (adjacency is defined here as immediate neighbour if there is connectivity information in the data set, or otherwise by proximity). At the end of that propagation, any points left with $T_l < \kappa' < T_u$ that have not been permanently classified are declassified.

The purpose of this thresholding step is to keep only strong ridges. A similar thresholding could be carried out on the minimal ridge points, however these are of lesser interest as they constitute weaker features. In fact in most of the results that follow in section 6.6, only the maximal ridges are presented. Also, this thresholding is useful for the evaluation of the results. This is discussed in detail in section 6.6.3, along with the issue of setting the optimal values of the hysteresis thresholds.

6.4 Flexcords

In addition to ridges, *flexcords* are another type of feature that are important in codifying the topology of a surface ([Port94], also priv. comm.). The definition of a flexcord point (also known as a *sub-parabolic* point, for reasons that will become clear in the next section), is similar to that of a ridge. We have already defined a point on a surface as belonging to a *red* ridge if the red curvature on that point is an extremum in the *red* direction and vice-versa for a blue ridge. Similarly, we define a point on a surface as belonging to a *red* flexcord if the *red* curvature is an extremum in the *blue* direction and vice-versa for a blue flexcord.

It then follows that flexcords demonstrate attributes similar to those of ridges: they can be characterised as "blue" and "red" (depending on the label of the defining curvature in question), or as "minimal" and "maximal", depending on whether the (magnitude of) local curvature extremum is a maximum or a minimum. Moreover, they share the same set of properties that make ridges important, as discussed at the beginning of section 6.3: de-formability (rather than re-formability), conformance to topology feature configuration rules, and contribution to shape perception. Finally, because of the dual relationship of their definition with that of ridges, they are (at least theoretically) as easy to detect as ridges. A scheme such as the one employed for detecting ridges (section 6.3.1) can be implemented just as easily for flexcords. The only thing that changes is the selection, for each point P, of the appropriate neighbours P_1 and P_2 so that they correspond to the direction of (blue or red) curvature against which we want to classify the point (Figure 53). The rest of the scheme remains unchanged.

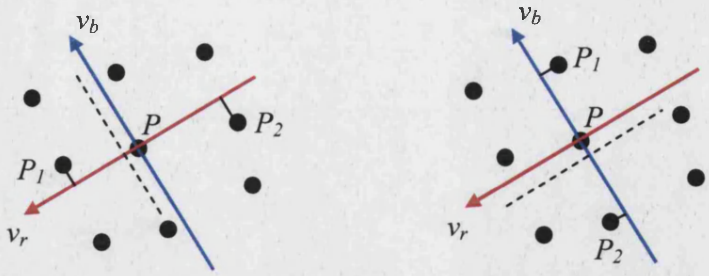


Figure 53: Flexcord detection on the discretised data.Left: Blue flexcord test (the values of the blue curvatures of P, P_1, P_2 are compared. Right: Red flexcord test (the values of the red curvatures of P, P_1, P_2 are compared. In both cases, the dashed line illustrates the approximate direction of the potential ridge. Note the similarity of these diagrams to those of Figure 52. The difference in each case is the label (colour) of the curvatures that participate in the comparison.

Such a scheme was indeed implemented in the course of this research. There is, however, a problem with flexcords that has prevented us from carrying out a rigorous (or, in fact, any other) evaluation of the results. This problem is twofold. Firstly, as opposed to ridges, flexcords are generally not easy to visualise on a surface, nor to predict their locations just by visually assessing an object. Secondly, they do not appear to have been studied as rigourously and thoroughly as ridges. Compared to the available literature about ridges, the amount of material on flexcords is disproportionally small. They are mentioned, for instance, by Porteous [Port94][PoPu00], Giblin et al. [BrGT96], Morris [Morr96], and Belyaev [BePK98], but the work of Morris [Morr96] is the only known attempt to explain the appearance properties of flexcords and to provide some examples on simple surfaces. There appears to be a lack of comprehensive examples of flexcords on simple surfaces. This, at the moment, is a factor that prevents both a thorough understanding of the concept of flexcords, and a rigorous evaluation of our implementation of the detection algorithms.

6.5 An alternative approach: using the Focal Surface.

In addition to the approaches already discussed for detecting umbilics, ridges and flexcords on a surface, it is also worth mentioning here a possible different approach that is based on an alternative definition of ridges and flexcords.

The study of the properties of these features on a surface has indicated that they are very closely linked to the underlying focal surface (see e.g. [Port94], chapter 10). More specifically, it has been shown that:

- Every ridge on the surface corresponds to a cusp (also known as a rib) on the focal surface, and vice-versa. Moreover, if the ridge is blue, then the rib occurs on the blue sheet of the focal surface (i.e. on the locus of the centres of blue curvatures), and a similar rule applies for red ridges.
- Every flexcord on the surface corresponds to a parabolic line on the focal surface, and vice-versa (this is why flexcords have also been known as sub-parabolic lines). Moreover, if the flexcord is blue, then the parabolic line occurs on the blue sheet of the focal surface (i.e. on the locus of the centres of blue curvatures), and a similar rule applies for red flexcords.
- For every ordinary umbilic point on the surface, the red and blue sheets of the focal surface come into contact at their points that correspond to that umbilic point.

The significance of these rules is that, if we had a sufficiently rich, detailed and reliable representation of the focal surface sheets, we could then implement a straightforward algorithm for detecting umbilics, ridges and flexcords on a surface. More specifically, we would only have to use the focal surface, and the work would consist of detecting on it features that are relatively easy to define and detect (i.e. cusps, parabolic lines, and contact). Then, for each such feature detected on the focal surface, we could assign the appropriate label to the corresponding point on the object's surface, thus constructing the differential geometry map on the object's surface.

This approach looks very elegant in theory, but unfortunately it is very difficult to implement for the discretised surface representations with which this research is concerned. An attempt has indeed been made by Chalatsopoulou [Chal02] to construct focal surfaces for such a sampled surface with no global analytic representation, and the quality of the results only highlights the practical problems of the process.

The most important problem is that the focal surface is not available in the first place, and it needs to be constructed. Given the currently available infrastructure, the way to build such a surface is to calculate the principal curvature for each point P in the point cloud S representing the surface, and then obtain from the centres of curvature a secondary point cloud S_f representing the focal surface. The following obstacles are then encountered which prevent us from using S_f for feature detection:

- <u>Stability of curvature calculation</u>: Unless the results of the curvature calculation are very accurate, the resulting centres of curvature will not form a correspondingly smooth and stable focal surface. An immediate consequence of this is that it is difficult to decide whether a point on the focal surface belongs to the blue sheet or to the red sheet, unless we refer back to the object's surface and make a classification there (which is, in itself, a problem, as discussed below).

- Suitability of surface representation: Even if the results of the curvature calculation are exceptionally accurate, and the main (object) surface is uniformly and densely sampled, the resulting centres of curvature are not guaranteed to form a uniformly and densely sampled representation of the focal surface sheets. In fact, even small changes in the local surface shape profile (on the main surface) may result in large perturbations of the local shape profile on the focal surface. Moreover, where there are regions on the main surface that are reasonably flat, the centres of curvatures move rapidly away from the main surface, the result being in effect a very sparse sampling of the sheets of the focal surface.
- Separation of curvature fields: This is essentially the same problem as the one discussed in section 5.4. Before any detection of features could be carried out on S_f, it would have to be separated into two subsets, each one corresponding to one of the (blue or red) sheets of the focal surface. Detection must then be carried out separately on each sheet. If there is a global analytic representation for the main surface, this problem is solvable to some extent (see e.g. the work by Morris [Morr96], complete with some free software to demonstrate the feasibility). However, for surfaces such as the ones studied in this research, where only local analytic representations of surface shape are available, this problem remains largely unsolved.

Because of these problems, no attempt has been made within this research to implement a technique that utilises the focal surface. However, this may constitute an interesting direction for future work, especially since it constitutes a significant motivation for resolving the problem of separating the blue and red curvature fields.

6.6 Results

In this section, we present the results of the ridge detection on the same test objects and body scans as those used in chapters 4 and 5. Throughout this section, points classified as ridges are marked on the surface by a white dot and points found to be near-umbilics are marked by a purple cross. The directions of the lines of a cross marking such a point correspond to the calculated directions of principal curvature – thus illustrating the potential instability of this particular calculation for umbilic points that are not isolated. Consider, for instance, the trivial example of a perfect sphere, as in Figure 54. The result is correct because all points are umbilic and no point belongs to a ridge. The curvature directions are clearly inconsistent between regions of the surface.

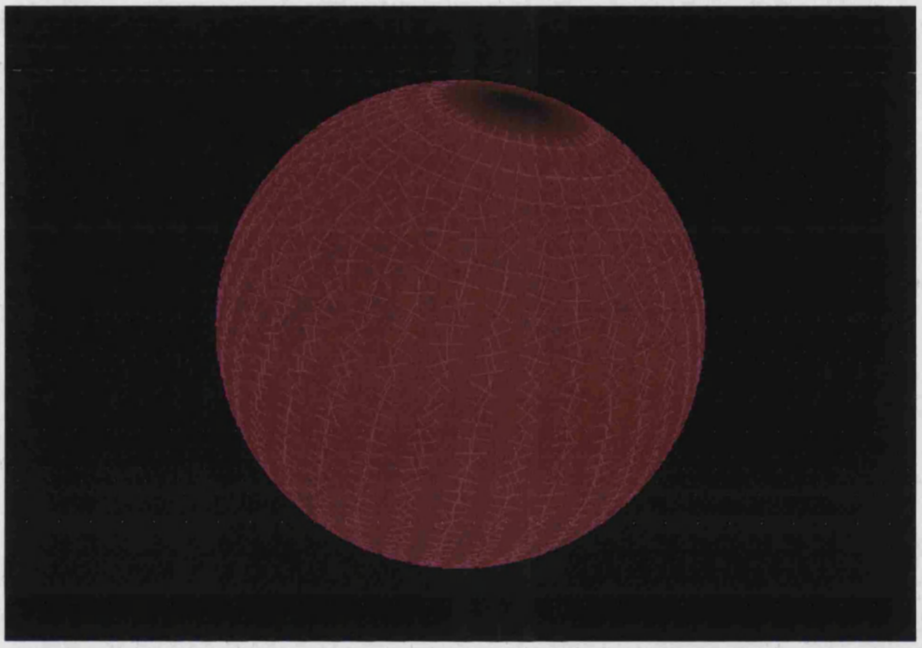


Figure 54: Results on the trivial example of a sphere (the results for the poles have been omitted because the singularities there cause patch fitting and subsequent calculations to fail).

6.6.1 Results of Ridge Detection on the Test Objects

A less trivial example is that of the ellipsoid, as shown in Figure 55. With the exeption of the poles (where no detection has been attempted because the local patch fitting fails), the results are near perfect. The ridges have been identified at the correct locations (i.e. at the intersections of the object with its planes of symmetry), and the locations of the umbilics have been approximated at the regions expected (as can be easily verified, e.g. by checking chapter 16 of [Port94]).

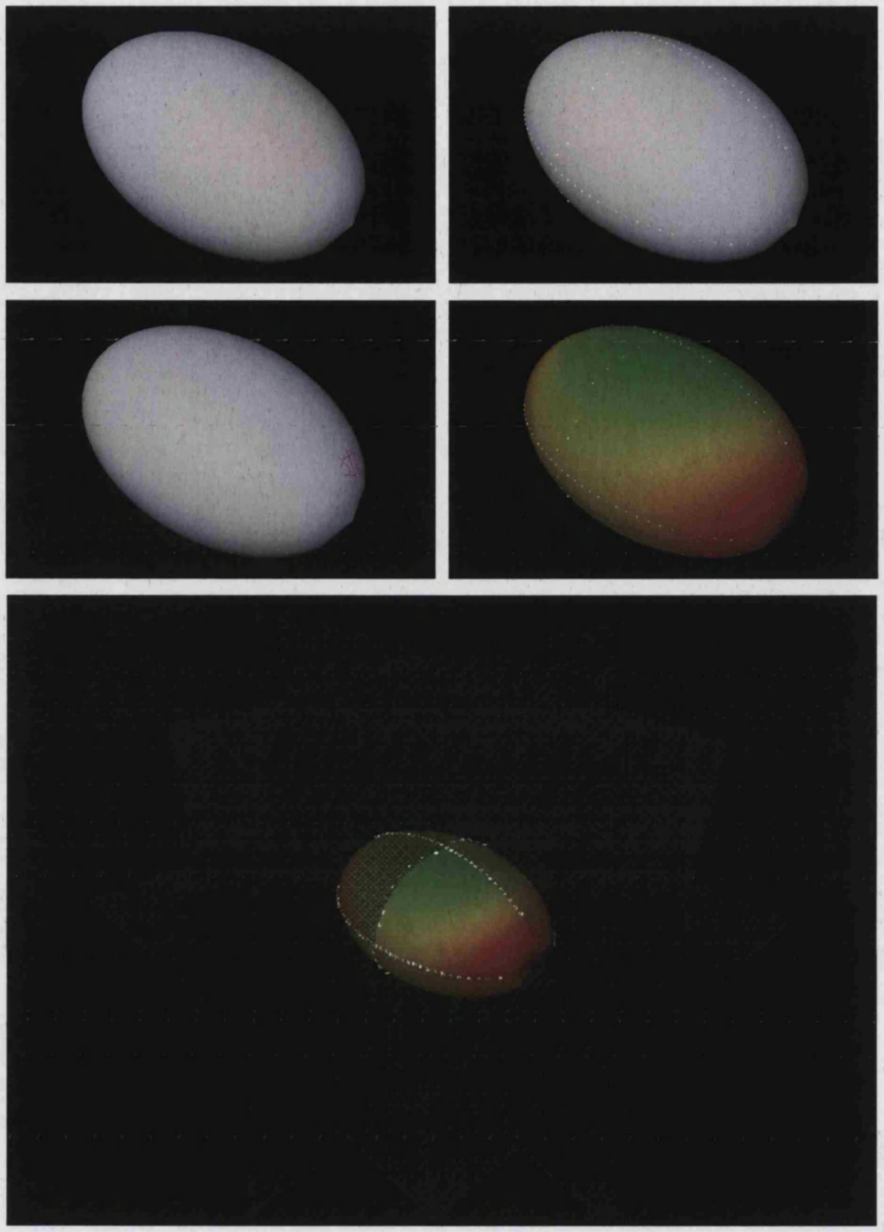


Figure 55: Ridges and umbilics on the test ellipsoid. Top row: The test object rendered with Phong Shading (left), and the detected ridge points shown (right). Second row: Detected near-umbilic points (left), and detected ridge points shown on the object's Gaussian curvature map (right). Bottom: ridge points and Gaussian curvature maps shown along with the planes of symmetry, to illustrate the accuracy of the ridge detection scheme.

Note that the ridge points following the "equator" of the ellipsoid are not exactly co-planar. This is because in the particular example the sample points were taken in such a manner that there are no points coinciding with the plane *Q* that defines that particular ridge. The points lie on planes parallel and close to *Q*, therefore these are successfully detected near-ridge points. The syntax used in the implementation for comparing the values of principal curvatures has enabled the algorithm to choose only one of two candidate points for ridge characterisation, thus allowing some non locality of response (within a certain distance band from the actual ridge locations). The significance of this will become clearer in subsequent examples. This observation also has an obvious implication: namely that the proposed algorithm clearly performs better as the resolution increases, as it is then more likely for sample points to occur close to actual ridges on the surface. Note that, for this ellipsoid, one of the three ridges (the "equator") is minimal, the other (the one not passing through the umbilic points) is maximal, and the third one (the one passing through the umbilic points) is mixed.

The next example is that of the tooth surface (Figure 56). In this example, only the maximal ridges are shown. This particular object has a very rich configuration of ridges, as indicated by its planes of symmetry. The illustration provided shows how the result is affected by changes in the thresholding. In this example, for simplicity the high and low hysteresis thresholds were set to the same value, and results for successive values equal to 0.0002, 0.0003 and 0.0005 are shown. Note that the number of ridge points detected (positives returned) is reduced as the threshold value increases, leaving only the ridge points correponding to stronger features of the surface such as the "edges" of the tooth. The issue of false positives (i.e. points that are detected as belonging to a ridge, but are far from any ridge line) is discussed later in this section. It must be noted that, at the time of conducting these experiments it is not been known whether there are any ridges on this surface other than those occuring at the intersections with the planes of symmetry – literature searches returned no relevant information. Therefore, any assessment of the false positives for this particular test object (such as, for example, the isolated points on the faces of the tooth surface as can be seen in Figure 56) is bound to introduce difficulties.

Finally, from the images in Figure 56 it is clear that not all sample points coincide exactly with planes of symmetry. Even so, the particular implementation seems to allow for some non-locality of response for near-ridge points.

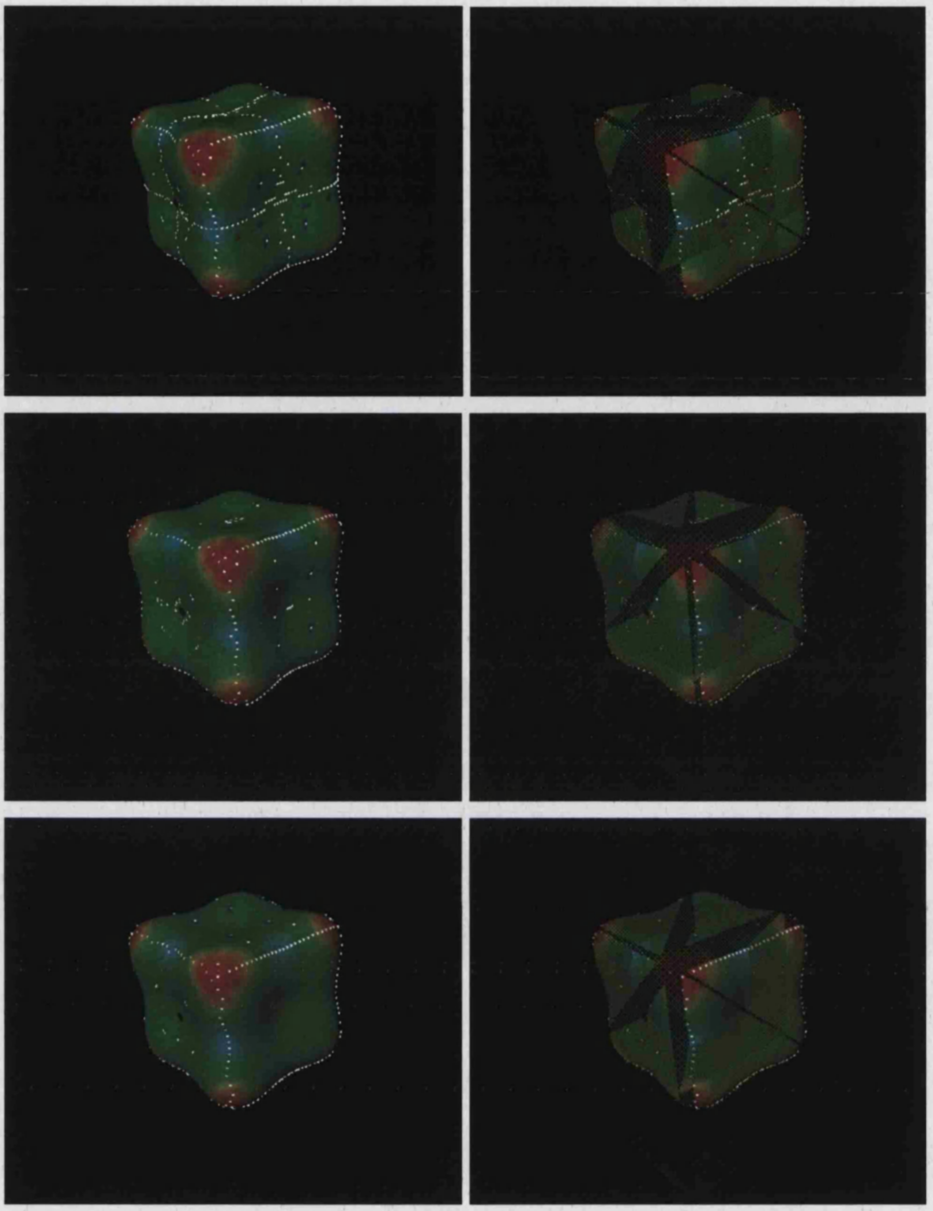


Figure 56: Ridge detection results on a tooth surface (shown here with its Gaussian curvature map) for various values of threshold as described in the text (from top to bottom: 0.0002, 0.0003, 0.0005). In the right hand column the planes of symmetry are also shown to illustrate the accuracy of the result.

The next example is the smoothed octahedron, on which the location of all ridges is known from [BePK98]. For this object, our proposed method has returned very satisfactory results (Figure 57). All the meaningful (maximal) ridges were detected, whereas the minimal ones demonstrated the expected instability and are not shown.

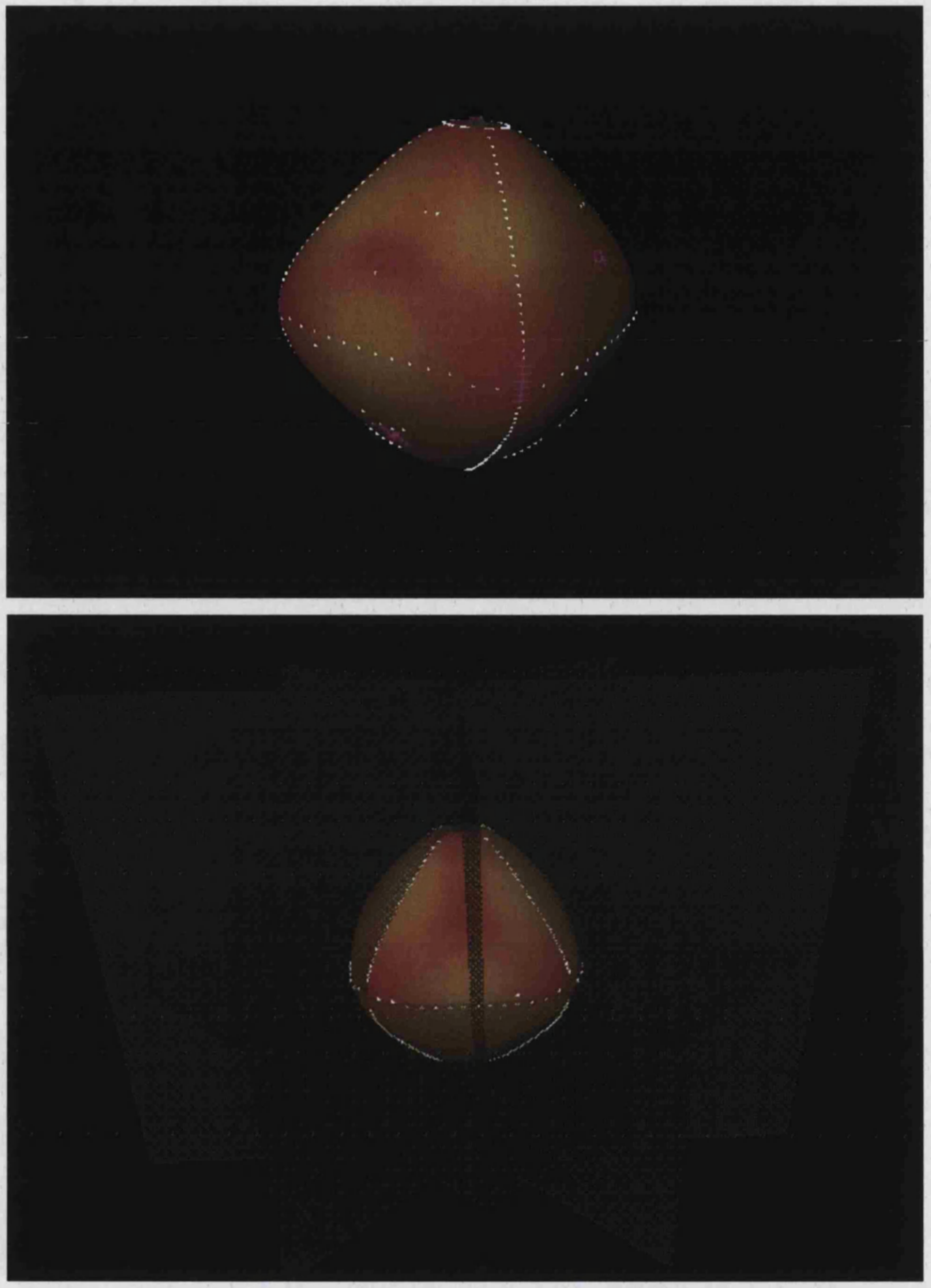


Figure 57: Ridges and umbilics on a rounded octahedron (shown here with its shape index map and, at the bottom, with its planes of symmetry). Note that only the maximal ridges are shown (locations of both maximal and minimal ridges for this object can be found in [BePK98]).

Last in the series of test objects is the bumpy sphere. The locations of the ridges for this object are known from [Port94] and are, in fact, illustrated in Figure 13. For comparison, the results returned by the implementation of our proposed method are shown in Figure 58. These results are remarkably close to the ground truth, despite a few ridge lines being truncated as a result of the hysteresis thresholding (a rigorous numerical evaluation follows later in this section). It is also evident that the locality of response has not been particularly good for this object. This is not only because of the sampling arrangement (most sample points in fact do not coincide exactly with planes of symmetry), but also because of the sensitivity of the entire local patch fitting, curvature calculation and ridge detection scheme to object shape. This particular shape is only slightly different from a perfect sphere. In fact, in terms of human visual perception it is practically indistinguishable from a perfect sphere, as Figure 59 clearly illustrates.

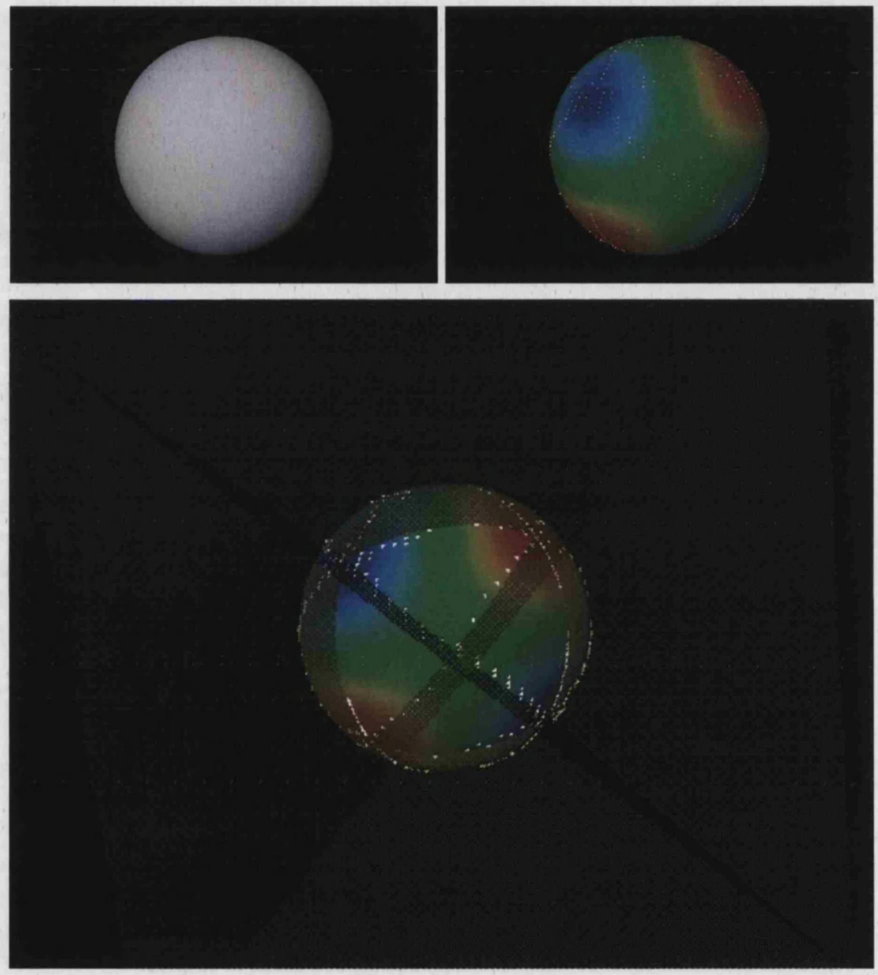


Figure 58: Ridges and umbilics on a bumpy sphere (shown here with its curvedness map and, at the bottom, with its planes of symmetry). Note that both minimal and the maximal ridges are shown.

In addition to the ridges occuring at the intersection of its surface with the planes of symmetry, this type of bumpy sphere has some more ridges, occuring as closed loops around each umbilic point (as shown in Figure 13). Unfortunately, these features are two small to be detectable by our method at a resolution comparable to that of the human body scans available. An attempt has been made to sample a bumpy sphere at a much higher resolution (0.5mm for a sphere of radius of 1100mm) and to try to detect these ridges on it. However, the sensitive nature of this object (as explained above) has caused the experiment to return too many false positives for these closed loops to be discernible after the detection. Figure 60 illustrates the result following repeated attempts made to optimise the outcome by adjusting the thresholding parameters (with the aid of the charts that will be described in section 6.6.4). Clearly, the optimisations have been limited as a result of the limited variation of the principal curvature values which, in return, made the implementation of the proposed detection scheme rather unstable (a further explanation for this instability is given in section 6.6.4).

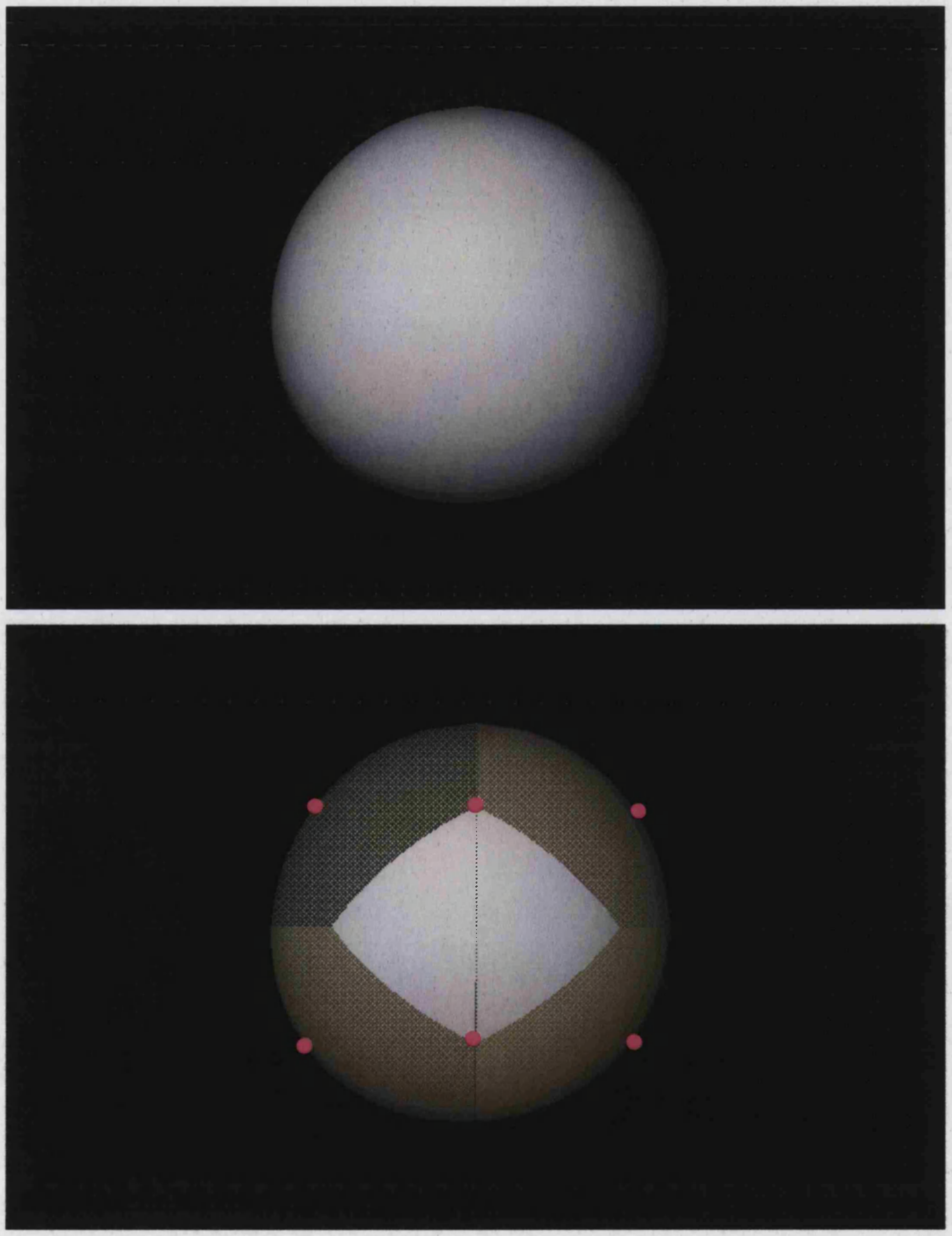


Figure 59: Sensitivity to shape. Cross section of the test bumpy sphere, purple dots show the umbilics, symmetry planes also shown.

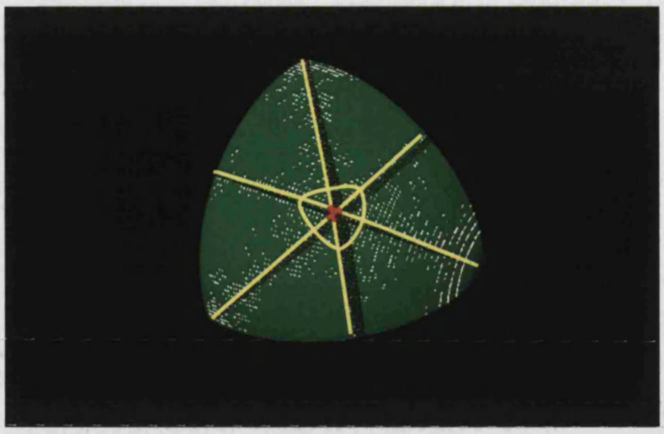


Figure 60:Detection of small ridge features (hoops around umbilics on a bumpy sphere). Yellow lines: Locations of ridges on surface, drawn by hand as shown in [Port94]. Purple dot similarly marks the umbilic. Planes of symmetry shown.

6.6.2 Results of Ridge Detection on Real Human Body (3D Scanner) Data

The examples that follow involve ridge detection on real data (human body scans). More specifically, Figure 61 and Figure 62 show the results for the medium resolution (2cm) torso models of a male and a female subject. The results are plotted against the Gaussian curvature maps and the shape index maps of the torso surfaces in order to provide a visual estimation of whether the output of the method is reasonable (since there is no known ground truth such as an analytic representation of these surfaces). The results appear to make sense and to be compatible with the visual perception of the shape of the particular subjects. The same could be said for the results shown on the high resolution (5mm) whole body Hamamatsu scans of Figure 63 and Figure 64. Section 6.6.5 presents an experiment carried out in order to assess the robustness of these results on deforming human body data and their potential suitability towards formulating a human body shape classification scheme. Section 7.4.2 includes a suggestion for a suitable scheme for rigorous numerical evaluation of ridge detection results on human body data.

In accordance with the findings presented in the previous chapter, it appears that higher data resolution produces better, more localised results with fewer artefacts, and returns a richer configuration of features even after heavy Gaussian smoothing of the body scan surface.

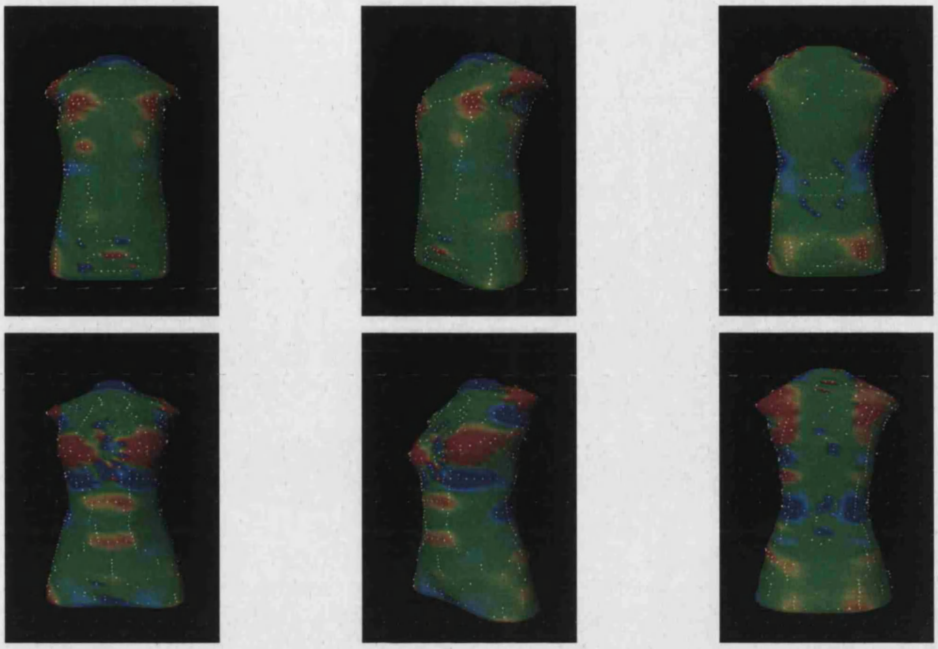


Figure 61: Results for low-resolution torsos (shown along with the Gaussian curvature map). Top row: male. Bottom row: female. Left hand column: front view. Middle column: side view. Right hand column: rear view.

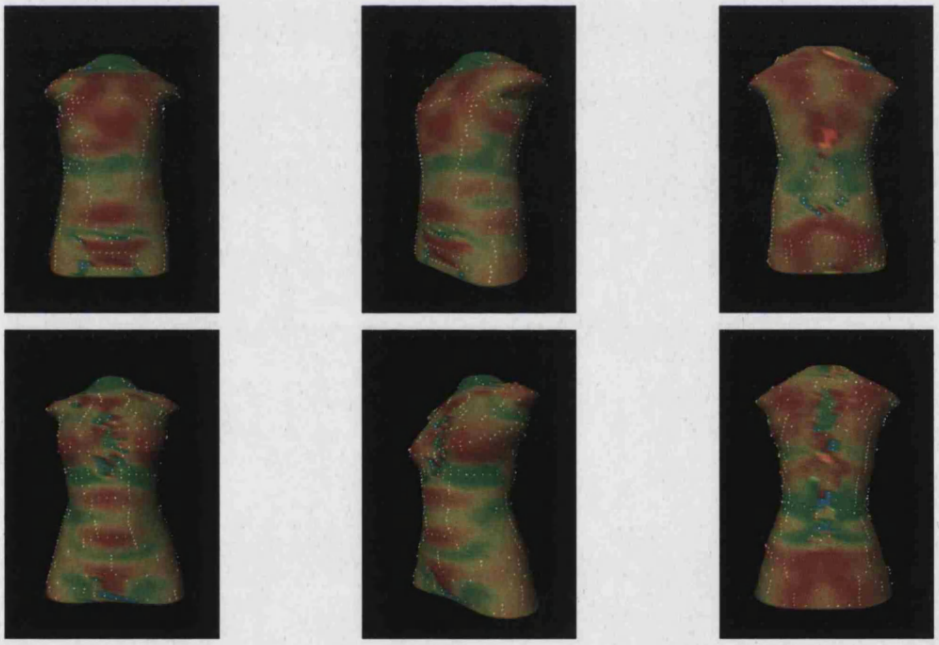


Figure 62: Results for low-resolution torsos (shown along with the shape index map). Top row: male. Bottom row: female. Left hand column: front view. Middle column: side view. Right hand column: rear view.

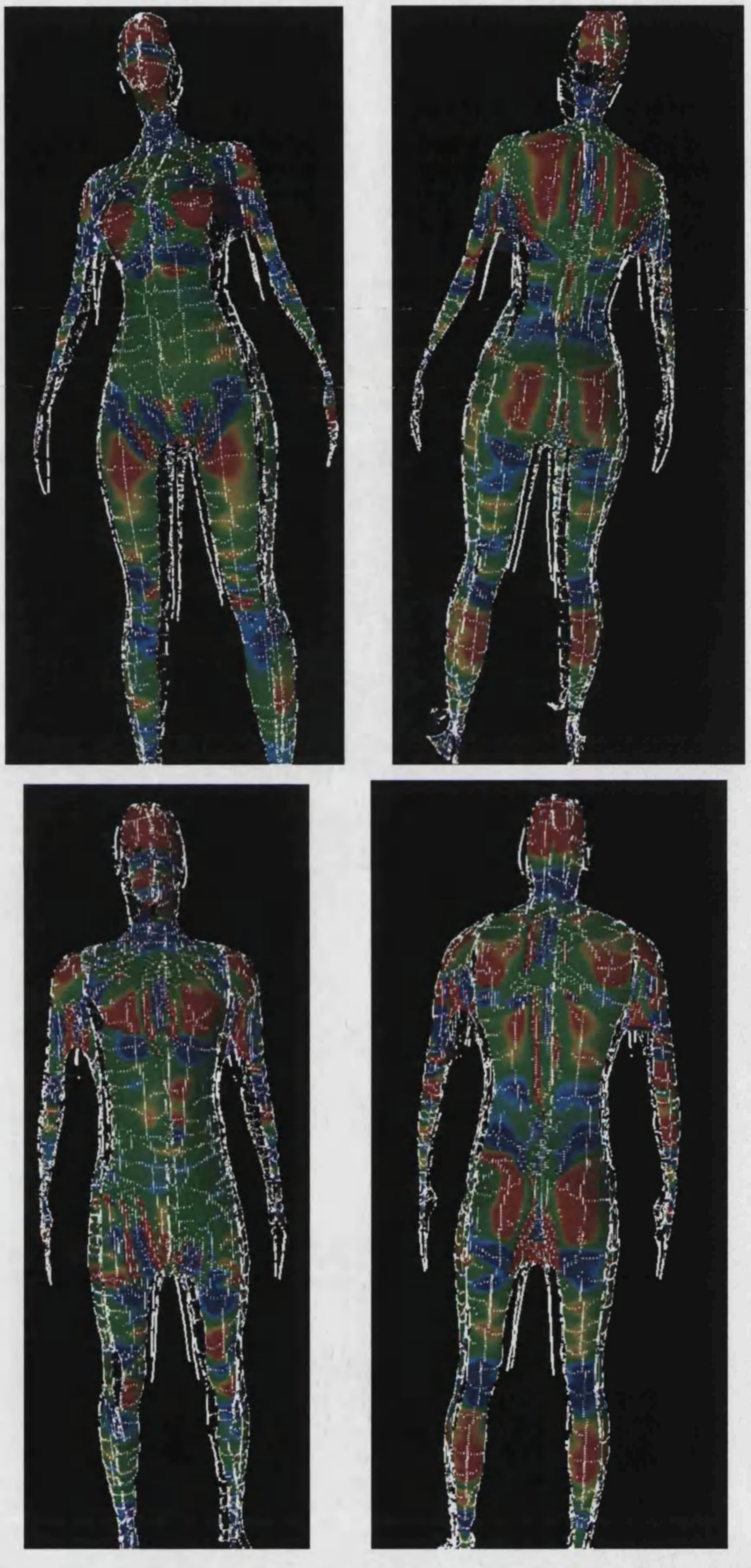


Figure 63: Maximal ridges for high-resolution Hamamatsu whole body scans, shown along with the Gaussian curvature map (no thresholding). Top row: Front and rear view of the scan of a female subject. Bottom row: Front and rear view of the scan of a male subject.

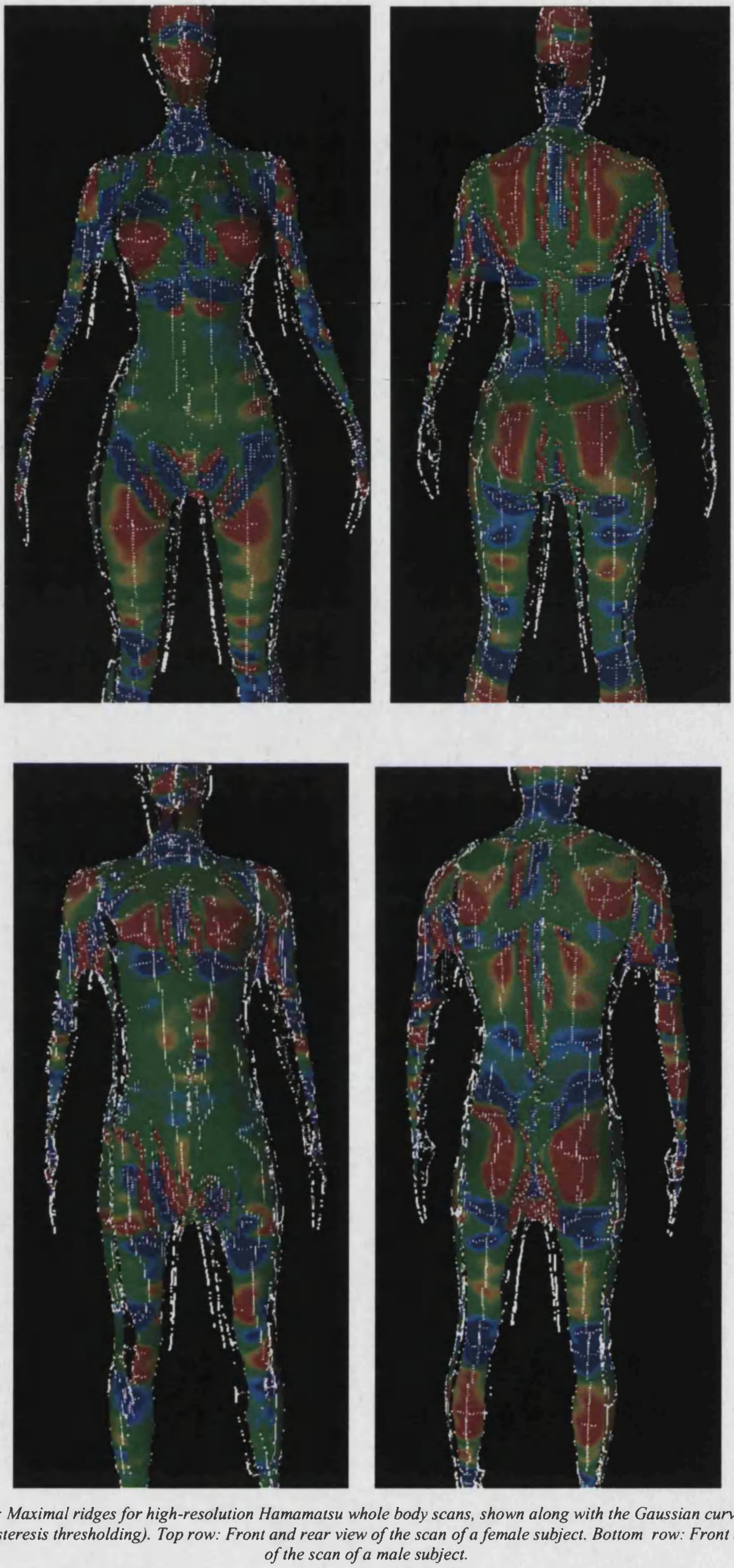


Figure 64: Maximal ridges for high-resolution Hamamatsu whole body scans, shown along with the Gaussian curvature map (including hysteresis thresholding). Top row: Front and rear view of the scan of a female subject. Bottom row: Front and rear view of the scan of a male subject.

6.6.3 Evaluation Strategy

Following a visual assessment of the results in the previous section, we need to establish whether they are acceptable in the numerical sense. To do so we have used a scheme for determining whether the points that have been classified as ridges in our test objects approximate known ridge points on the surfaces in question.

This exercise is essentially focused around the discrimination between true and false positives. It is therefore based on existing work by Foerstner [Foer96] on performance characterisation in computer vision, and by Scott et al [ScNP98] on realisable classifiers. The scheme has been designed as follows:

We have an algorithm (described in section 6.3.1) that receives as input a (partially structured) point set P of size N, and in the output we have some of the points of P flagged as "ridges".

Our objective can be phrased as follows: for each point that has been flagged as a "ridge" we want to establish whether or not it is close (i.e. within a distance threshold T_l) to a known ridge.

In order to establish the closeness of these "ridge" points to the actual ridges, we need to know where the actual ridges are in the first place. Therefore, we use test objects for which:

- Either: we know [Ian Porteous, priv.comm.] that all ridges are on (and only on) lines of symmetry, and we know where these lines lie (e.g. ellipsoid, tooth);
- Or: we know that all the ridges in which we are interested (i.e. strong maximal ridges) are on lines of symmetry, and we know where these lines lie (e.g. rounded octahedron).

This approach allows us to know the precise locations of the ridges of interest on the test objects. Since the analytic forms of the test objects are known, we can easily calculate their intersections with the planes of symmetry, thus obtaining the analytic forms of the ridges. Moreover, because all the ridges involved in this evaluation exercise are, in effect, planar curves, we can implement computationally simple criteria for the evaluation of ridge detection process that involve only the planes of symmetry, rather than the ridges themselves. All the criteria used are thus based around measuring the distance between a point classified as belonging to a ridge and a plane of symmetry. If that distance is within a threshold T_1, we then classify the point as a true positive, based on the fact that planes of symmetry correspond to ridges on the surface.

Our method can then be formulated as follows: for every test object surface represented as a point cloud P, let:

- S_j ($j = 1,\dots,M$) be a plane of symmetry on the surface.
- P_i ($i = 1,\dots,N$) be a flagged point (detected as a ridge point)
- d_{ij} be the distance between P_i and S_j.

All the possible planes of symmetry for the set of test objects involved in this exercise can be classified into the following types[12]:

- S_x: $\{x = 0\}$
- S_y: $\{y = 0\}$
- S_z: $\{z = 0\}$
- D_x: $\{y = z, y = -z\}$
- D_y: $\{x = z, x = -z\}$
- D_z: $\{x = y, x = -y\}$

Note that D_x, D_y and D_z correspond to *two* planes each. The sets of planes of symmetry for the test objects (depicted in Figure 65) are therefore as follows:

Ellipsoid:	$S_{ell} = \{S_x, S_y, S_z\}$
Tooth:	$S_{tth} = \{S_x, S_y, S_z, D_x, D_y, D_z\}$
Rounded Octahedron:	$S_{oct} = \{S_x, S_y, S_z, D_z\}$
Bumpy Sphere:	$S_{bsp} = \{D_x, D_y, D_z\}$

---

[12] Note here that all the test objects are defined in such a way that their barycentre coincides with the origin (refer back to section 3.3.1 for the analytic definitions).

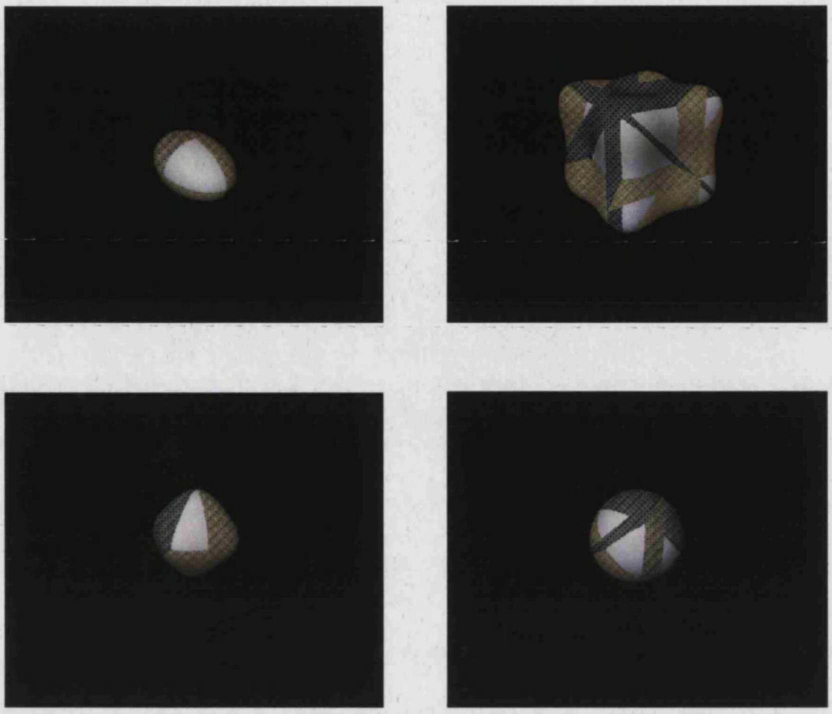


Figure 65: Planes of symmetry for the test objects.

To calculate the numbers of true positives and of false positives among the ridge points P_i detected, we then follow these steps:

```
Step 1: For (a value of $T_l$):
    Step 1.1: For (every $P_i$)
        Step 1.1.1: Calculate all the $d_{ij}$ (Distances from all planes of symmetry)
        Step 1.1.2: Let $d_{min}$ = the minimum of all the $d_{ij}$
        Step 1.1.3: If $d_{min} < T_l$
            Then $P_i$ is a true positive
            Else $P_i$ is a false positive
    End (of step 1.1)
End (of step 1)
```

Carrying out the above for various values of T_l allows us to construct a diagram of the number of true positives plotted against the number of false positives. This is not dissimilar to a ROC curve, although strictly speaking we cannot call it so. The reason is that we have no *a priori* knowledge of the actual number of positives and therefore, likewise, no knowledge of the actual number of negatives (points not on a ridge) in our set of sample points on the object. This is because it has not been possible to sample our test objects in a manner guaranteeing that sample points either coincide exactly with ridges, or are sufficiently far enough from them. In all cases there are sample points whose status as ridge points is ambiguous, and can only be *a priori* labelled as "near-a-ridge" at best. Such points lie mainly in the proximity of a ridge intersection, but also in cases where the ridge did not coincide with the sampling contours generated by the cylindrical or spherical sampling schemes employed. Whilst it may be possible to devise different sampling schemes that resolve this problem, they would necessarily be complicated and/or heuristic. Moreover, the resulting point clouds would no longer have a structure resembling that of our human body scans. In addition, such sampling would be likely to sacrifice the straightforward implicit connectivity information and thus make the implementation of the evaluation schemes unnecessarily complicated. We have therefore decided to settle for this ambiguity as to the number of positives (and negatives), since it does not significantly affect our ability to build graphs that enable easy and straightforward evaluation of the ridge detection results. The major consequence of this ambiguity is that the top-right end of the curve we calculate has no fixed location, i.e. we cannot calculate the true and false positive rates in terms of which the ROC curve lies inside a unit square and, ideally, has unit area under the curve.

It could also be argued that for the purposes of this evaluation we could count as a priori "positives" all the points within distance T_1 from a plane of symmetry, and thus count the number of positives necessary in order to calculate the true positive rate and subsequently use it in order to generate proper ROC graphs. However, there are two major problems with this approach. Firstly, for some of our test objects (especially the Tooth surface), we do not know the exact locations of all ridges. Apart from those ridges that occur at the intersections of the test objects with the planes of symmetry, we are unable to verify whether there are any other ridges, as there appears to be no such relevant information in the literature. This implies that we might be underestimating the number of positives and therefore miscalculate the true positive rate. Secondly, some of the tests (namely those involving the Tooth surface and the Rounded Octahedron) have been carried out by taking into account the detection of maximal ridge points only. In this case, the number of total positives would be the number of points within distance T_1 from a plane of symmetry, *minus* the number of those points that correspond to minimal ridges. However, this latter number is not known a priori, and is not straightforward to estimate (it would involve accurate location of the umbilic points on the two test objects since these are also the points where a ridge changes from minimal to maximal and vice versa [Port94]). This again implies that we might be over- or under-estimating the number of positives and therefore miscalculating the true positive rate.

Even without being able to calculate the true and false positive rates, we can treat the resulting chart of the number of true positives agains the number of false positives as a kind of ROC graph. If we were to use the true and false positive *rates* instead, all we would do is divide each coordinate of all the points comprising the graph respectively for the true positives by the number of positives, and for the false positives by the number of negatives. In that sense, the graphs presented in the following section can be claimed to be similar to the standard ROC graphs, lacking only a fixed location for their upper right corner (which means, inter alia, that it is not possible fully to determine the convex hull of the graph). Therefore the only thing we can say about where the chart lies relative to the main diagonal is whether or not there are more true positives than false positives. However, a chart that runs up the true positive axis (usually plotted as the abscissa) is ideal in the sense that it corresponds to a system that returns no false positives. Furthermore, the more true positives such a system can detect, the better it is doing since TP (the number of true positives) is bounded above by the number of positives P, even though in our case we do not always know what P is. The further such an ideal chart can go up the positive axis, the better.

In general, therefore, although we cannot qualitatively evaluate the performance from our charts as we can from an ROC curve, we can say qualitatively that, *for a given test object*, the closer the chart is to the true positive axis, and the higher it goes, the better.

Additionally, in the construction of our charts we need to take into account the hysteresis thresholds used during the ridge detection stage. This would give us an indication of how the detection algorithm performs for various values of the hysteresis thresholds, and also enable us to identify reasonable estimates for the optimal values of these thresholds. Hysteresis will hopefully increase the number of true positives and reduce the number of false positives. Extra care must be taken as the hysteresis step introduces two thresholds that may affect the result. We therefore have a total of three parameters (the distance threshold T_1 and the two hysteresis thresholds T_l and T_u) that we need to vary in order to produce performance charts.

6.6.4 Analysis of the results

The tables and graphs in this section contain the results of the evaluation method described in section 6.6.3 for each of the test objects used throughout the research. In the tables, each cell is associated with a pair of upper and lower threshold values and contains the corresponding number of true positives above the number of false positives. These are plotted on the graph following each table, using the number of false positives as the ordinate and the number of true positives as the abscissa. Points on the graphs that are near the abscissa indicate good performance, and their convex hull (which, for reasons explained in the previous section, can only be determined partially) provides an indication of the performance level.

The results of ridge detection evaluation for the test ellipsoid are given in Table 10. This is an ideal and simple shape, which explains why no false positives were found (in other words, the fitting, curvature estimation and ridge location work perfectly). The results for the tooth, rounded octahedron and bumpy sphere are presented respectively in Table 11, Table 12 and Table 13.

In all cases, as we move from left to right and from top to bottom along the table, the values of both the false positives and the true positives are monotonically non-increasing and eventually, for sufficiently high thresholds, they vanish[13]. This is expected because, as the thresholds increase, fewer points pass the hysteresis test that allows them to be classified as points belonging to a strong ridge feature line. Also, cells that are on the diagonal of the table correspond to equal values of the upper and lower hysteresis thresholds. This effectively means that no hysteresis was carried out, but simple thresholding was effected instead. Cells below the diagonal are crossed out, since the lower threshold cannot have a value higher than that of the upper threshold.

The case of the ellipsoid, despite being somewhat trivial, is worth a closer look because it is easy to estimate the number of true positives that we can expect. This test object is sampled as a 40x40 closed quadmesh. Because of the expected singularities near its "poles", we do not attempt a detection on the first two and the last two parallels, so the likely candidates for detection are arranged in a grid of $40 - 2 - 2 = 36$ rows (corresponding to the "parallels") and 40 columns (corresponding to the "meridians").

---

[13] It is worth noting that the sum of $TP+FP$ is also monotonically non-decreasing as a function of both hysteresis thresholds.

There are three ridges, as explained at the beginning of section 6.6.1. The first is the minimal ridge along the "equator", so we expect to detect up to 40 points relevant to that ridge (i.e. one for each column of points). The second ridge, maximal with no umbilics on it, is made up of two parts, each running along a meridian and therefore represented by up to 36 points (total 72 for that ridge). The same is the case for the third, mixed ridge, so a total of up 72 points should be detected "on" it (i.e. within distance T_1 from it). Finally, with the exception of the poles, these ridges intersect at 4 points along the equator, therefore this number should be subtracted from the sum in order to avoid counting twice any points that correspond to intersecion of two ridges. The total number of expected ridge points is therefore $40 + 72 + 72 - 4 = 180$, which is precisely the number of true positives detected, as Figure 66 illustrates for the lowest values of the hysteresis thresholds. It is also worth noting that, since we know this number, for this test case we have, in fact, enough information to construct a proper ROC curve. However, to do so would be otiose since it just lines up with the TP axis.

The chart in Figure 66 is ideal in the sense described above, whilst those in figures 67, 68 and 69 all show satisfactory performance, in that they are close to the true-positive axis and reach quite a high number of true positives.

Finally, it is worth noting that for the case of the bumpy sphere, the response demonstrates a peculiarity not observed for the other test objects. The numbers of true and false positives seem to be unaffected by the changes in the values of the hysteresis thresholds, until a point is reached where thresholding causes zero positives to be returned. In order to interpret this sensitivity, it is worth noting that this object is *almost* a sphere, i.e. its principal curvatures are *almost* constant, and the range of values of the principal curvatures is very narrow. Since these thresholds are imposed on the principal curvatures, it is therefore highly likely for the entire range of principal curvatures (in the particular case, between 0.000330 and 0.000339) to lie above or below the threshold values. For that reason, the graph in Figure 69 appears trivial. In order to make the analysis more interesting for the bumpy sphere, we construct a different table, where we vary instead the value of the distance threshold T_1 used for evaluation, and each cell contains the number of true positives (*TP*) written above the number of false positives (*FP*) for hysteresis threshold values below 0.001 (i.e. where *FP* and *TP* are independent of the values of the thresholds).

Upper->	*0.00001*	*0.0001*	*0.0002*	*0.0003*	*0.0004*	*0.0005*	*0.0006*	*0.001*	*0.01*
0.00001	180 0	180 0	180 0	180 0	180 0	180 0	180 0	180 0	0 0
0.0001	X	180 0	180 0	180 0	180 0	180 0	180 0	180 0	0 0
0.0002	X	X	180 0	180 0	180 0	180 0	180 0	180 0	0 0
0.0003	X	X	X	144 0	144 0	144 0	144 0	72 0	0 0
0.0004	X	X	X	X	144 0	144 0	144 0	72 0	0 0
0.0005	X	X	X	X	X	112 0	112 0	72 0	0 0
0.0006	X	X	X	X	X	X	84 0	72 0	0 0
0.001	X	X	X	X	X	X	X	12 0	0 0
0.01	X	X	X	X	X	X	X	X	0 0

Lower^

Table 10: True and false positives for the test ellipsoid (detection includes minimal ridges, and the distance threshold set was 10mm for a typical sampling resolution of 12mm). Note the number of true positives in the upper left corner, which co-incides with the number of actual positives (i.e. any further lowering of the hysteresis thresholds would did not affect the result).

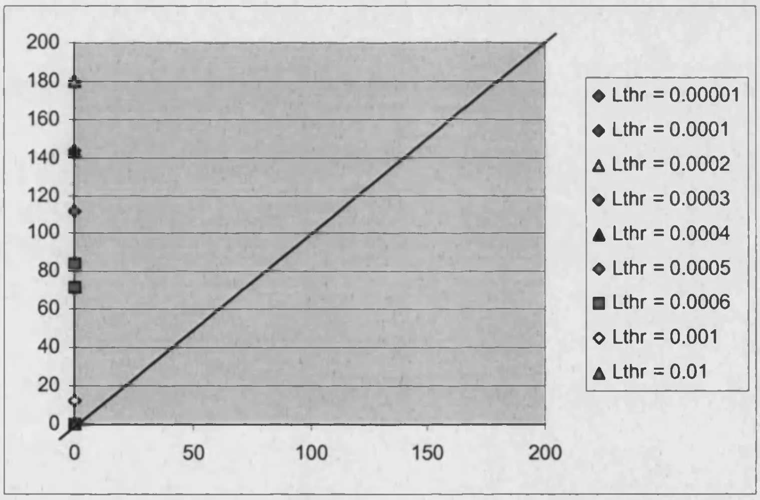


Figure 66: Ridge detection response diagram for the case of the test ellipsoid

Upper->	*0.00001*	*0.0001*	*0.0002*	*0.0003*	*0.0004*	*0.0005*	*0.0006*	*0.001*	*0.01*
0.00001	470 227	470 203	454 187	445 166	421 155	421 155	421 151	421 151	0 0
0.0001	X	454 201	438 187	429 166	405 155	405 155	405 151	405 151	0 0
0.0002	X	X	428 173	419 152	327 124	327 124	327 120	327 120	0 0
0.0003	X	X	X	275 65	232 52	228 52	232 48	232 48	0 0
0.0004	X	X	X	X	232 52	232 52	232 48	232 48	0 0
0.0005	X	X	X	X	X	232 50	232 48	232 48	0 0
0.0006	X	X	X	X	X	X	232 48	232 48	0 0
0.001	X	X	X	X	X	X	X	152 24	0 0
0.01	X	X	X	X	X	X	X	X	0 0

Lower^

Table 11: True and false positives for the test tooth surface (detection includes only maximal ridges)

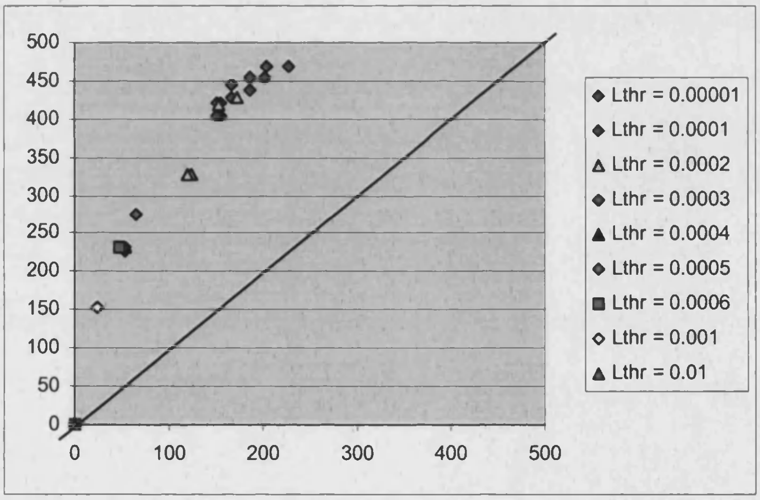


Figure 67: Ridge detection response diagram for the case of the test tooth surface.

Upper->	*0.00001*	*0.0001*	*0.0002*	*0.0003*	*0.0004*	*0.0005*	*0.0006*	*0.001*	*0.01*
0.00001	284 71	284 71	280 55	276 37	256 29	256 29	256 29	0 0	0 0
0.0001	X	284 71	280 55	276 37	256 29	256 29	256 29	0 0	0 0
0.0002	X	X	280 41	276 37	256 29	256 29	256 29	0 0	0 0
0.0003	X	X	X	260 37	256 29	256 29	256 29	0 0	0 0
0.0004	X	X	X	X	256 29	256 29	256 29	0 0	0 0
0.0005	X	X	X	X	X	256 29	256 29	0 0	0 0
0.0006	X	X	X	X	X	X	256 29	0 0	0 0
0.001	X	X	X	X	X	X	X	0 0	0 0
0.01	X	X	X	X	X	X	X	X	0 0

Lower^

Table 12: False and true positives for the test rounded octahedron (detection includes only maximal ridges)

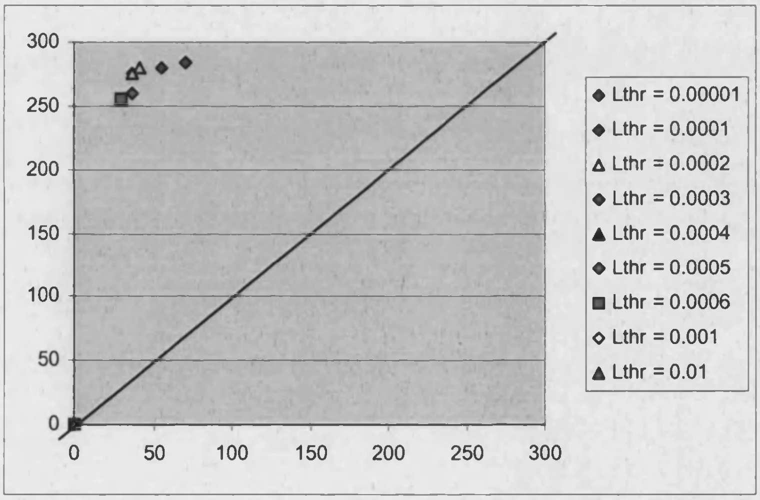


Figure 68: Ridge detection response diagram for the case of the test rounded octahedron

Upper->	*0.000001*	*0.00001*	*0.00002*	*0.00003*	*0.00004*	*0.00005*	*0.00006*	*0.0001*	*0.001*
0.000001	457 62	457 62	457 62	457 62	457 62	457 62	457 62	457 62	0 0
0.00001	X	457 62	457 62	457 62	457 62	457 62	457 62	457 62	0 0
0.00002	X	X	457 62	457 62	457 62	457 62	457 62	457 62	0 0
0.00003	X	X	X	457 62	457 62	457 62	457 62	457 62	0 0
0.00004	X	X	X	X	457 62	457 62	457 62	457 62	0 0
0.00005	X	X	X	X	X	457 62	457 62	457 62	0 0
0.00006	X	X	X	X	X	X	457 62	457 62	0 0
0.0001	X	X	X	X	X	X	X	457 62	0 0
0.001	X	X	X	X	X	X	X	X	0 0

Lower^

Table 13: False and true positives for the test bumpy sphere (detection includes minimal ridges, and the distance threshold set was 15mm for a typical sampling resolution of 12mm)

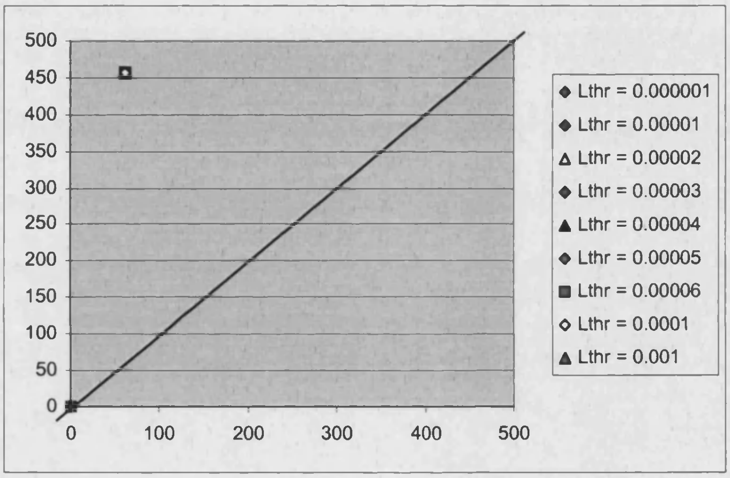


Figure 69: Ridge detection response diagram for the case of the test bumpy sphere

T1 ->	10	100	120	150	170	200	250	500	1000
TP:	26	401	429	457	474	488	500	518	519
FP:	493	118	90	62	45	31	19	1	0
FP/TP:	18.961538	0.294264	0.20979	0.135667	0.094937	0.063525	0.038	0.001931	0
sum:	519	519	519	519	519	519	519	519	519

Table 14: Summary of response results for various values of the distance threshold (in tenths of millimetres) in the case of a bumpy sphere

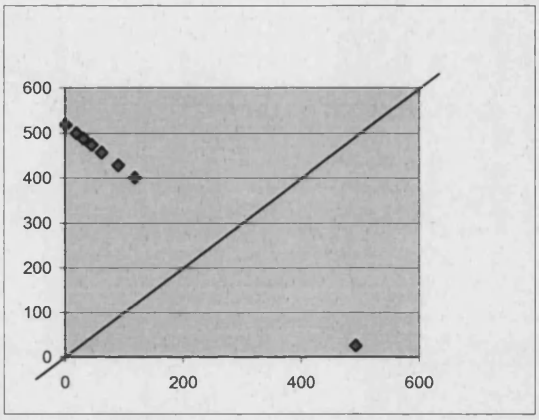


Figure 70: Summary of response results for various values of the distance threshold in the case of a bumpy sphere.

The results of this experiment are shown in Table 14 and the relevant graph is presented in Figure 70. It is clear from these findings that a very low value of T_1 (1mm, well below the sampling resolution) captures too few points within proximity of a symmetry plane. Therefore, most points classified as ridges are likely to appear too far away from a symmetry plane and therefore to be classified as false positives. As T_1 increases, more of the detected ridge points are close enough to a symmetry plane to be deemed to be true positives, thus increasing the TP/FP ratio. This particular experiment shows how the outcome of the evaluation of the detection algorithm depends not only on the adjustment of the parameters of the detection algorithm, but also on the adjustment of the parameters of the evaluation itself. This is inevitable because the discretised nature of our data only allows us to define our ground truth in approximate and relative terms in the non-trivial cases of interest.

6.6.5 Deformability

The final experiment in this chapter was carried out for the purpose of determining whether the ridge features detected by the algorithm proposed in section 6.3.1 are indeed de-formable rather than re-formable, i.e. whether small perturbations in the object's shape cause proportionally small perturbations in the configuration of the detected feature points. This test could be carried out using the bumpy sphere as a test object, but it is straightforward enough to allow us to proceed immediately to the human body models.

The first step was therefore to generate a sequence of shapes describing a deformation. This was accomplished by linear interpolation (weighted average) between two torso models. By incrementing the interpolation parameter by a constant amount, a series of deformation stages (frames) were thus obtained that each correspond to a small successive perturbation in the torso shape. Each of the resulting shapes was then processed using the methods described in this and previous chapters in order to calculate the curvature maps and detect its ridge points. The results are shown in Figure 71 below.

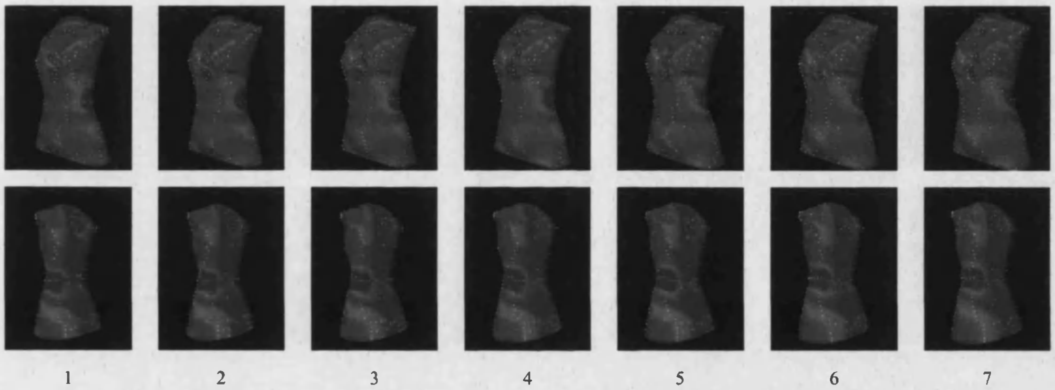


Figure 71: Ridge Points on a deforming torso, with the Gaussian curvature map rendered on the surface.

A visual inspection of the results suggests that the detected ridge points indeed constitute deformable features. However, a numerical evaluation is necessary in order to support this argument. Given that the proposed algorithm detects points, but these points in fact correspond to line features rather than point features, it has been essential to circumvent the (currently unsolved) problem of line feature tracking and correspondence in order to investigate this issue.

For this purpose, a simple distance metric was employed that does not require correspondence between the two sets (say A and B) of the detected ridge points between two shapes S_A and S_B. This distance d_r is defined as follows:

- For every point P_i in A we calculate the Euclidean distances between P_i and every P_j in B and take the minimum one for each P_i in A. We then sum these minimum distances.

- For every point P_j in B we calculate the Euclidean distances between P_j and every P_i in A and take the minimum one for each P_j in B. We then sum these minimum distances.
- We obtain d_r by adding these two sums.

Therefore, d_r may be defined by the equation:

$$d_r \equiv \sum_i \left(\min(P_i \in A, P_j \in B), \forall j\right) + \sum_j \left(\min(P_j \in B, P_i \in A), \forall i\right) \quad , \qquad (6.7)$$

and demonstrates two of the essential properties that are required of distance metrics. Firstly, it is independent of the order of A and B (i.e. its value does not change if we swap the labels of A and B between the set of ridge points for the two shapes). Secondly, if $A = B$, then $d_r = 0$.

We then calculate the following series of numbers:

- For each frame in the sequence (1 to 7), the distance between that frame and the first frame (1). This provides an estimate of the extent to which the configuration of ridge points changes as the shape increasingly deforms.
- For each frame in the sequence (2 to 7), the distance between that frame and its predecessor. This provides an estimate of the extent to which the configuration of ridge points changes between successive frames as the surface is deformed.

The results for the first calculation are presented in Table 15 and plotted in Figure 72. The graph indicates that the distance increases in an approximately linear manner (i.e. in proportion to the change in shape), and is therefore proportional to the perturbation in the object shape. This suggests that the detected ridge points indeed constitute deformable features.

frame	distance
1	0.0
2	31892.8
3	50934.4
4	68854.4
5	82635.6
6	93953.3
7	108634.0

Table 15: Distances d_r between each deformation frame and the initial shape.

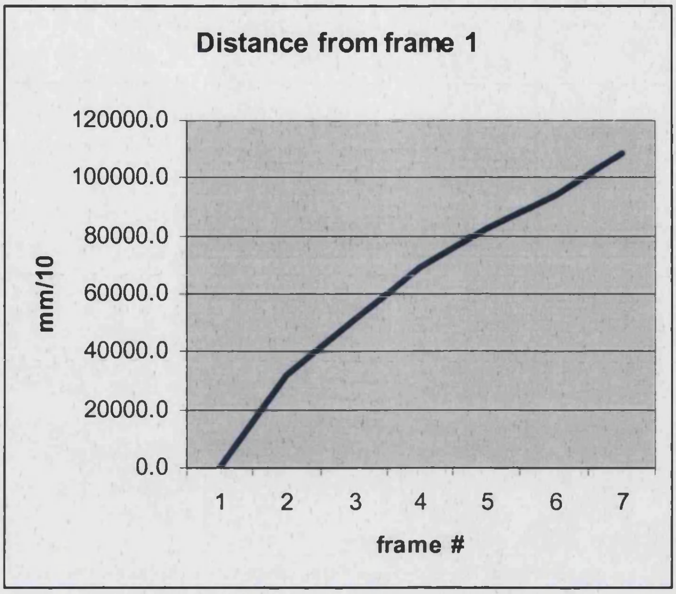


Figure 72: Distances d_r between each deformation frame and the initial shape.

	Distance	Cumulative
1 to 2	31892.8	31892.8
2 to 3	33452.7	65345.5
3 to 4	31853.9	97199.4
4 to 5	32701.6	129901.0
5 to 6	29086.2	158987.2
6 to 7	34713.4	193700.6

Table 16: Distances d_r between successive deformation frames.

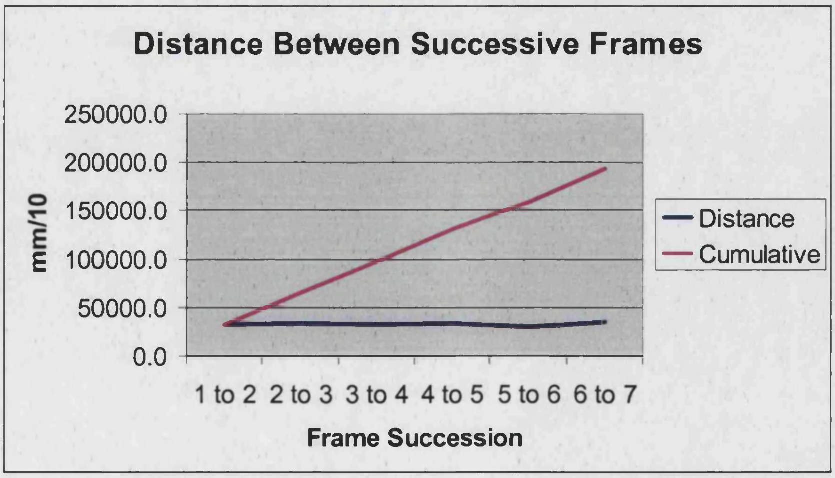


Figure 73: Distances d_r between successive deformation frames.

This argument is further supported by the results of the second calculation, which are shown in Table 16 and plotted in Figure 73. The table and both graphs indicate that the distance between successive frames is approximately constant. The cumulative distance therefore increases in an approximately linear manner (i.e. in proportion to the change in shape), and is therefore proportional to the perturbation in the object shape. This further suggests that the detected ridge points indeed constitute deformable features.

6.7 Conclusions

The visual results from the methods presented in this chapter appear to tally with our expected perception of the shapes involved. Moreover, the numerical evaluation suggests that the features detected are true to the original surfaces and deformable in accordance to surface deformations.

The outcome of this concluding step of the methodology proposed in this research far provides a set of deformable features that can be used to describe the topology of a surface. The implications of achieving the detection of these features in a fully automated manner and the arising opportunities, make more sense to discuss in the context of the proposed method in its entirety. This discussion is the subject of the next (and final) chapter.

7 Conclusions and Future Work

This final chapter begins by drawing some final conclusions from the work described in this thesis, in the broader context of 3D human body modelling. The chapter concludes by outlining some suggestions for future work.

7.1 Overview of the Results

The principal objective of this research was to develop, implement and evaluate an accurate, reliable, robust and fully automated methodology for calculating and visualising the geometry of a human body (with particular emphasis on the curvature shape characteristics), using point cloud data obtained from a three-dimensional surface scanner. In order to reach this objective, we developed a technique for local surface reconstruction from scanner data, and used local analytic surface representations in order to calculate curvature properties on the body surface. We also used the values of these curvature properties in order to estimate the location of deformable surface features (such as ridges and umbilic points) that are used to describe an object's shape and its surface topology. The techniques were evaluated by using test objects with known curvature properties and differential geometry feature maps that enabled quantitative comparisons for the purposes of assessing accuracy, reliability and robustness.

The results (both visual and numerical) presented in the concluding sections of chapters 4, 5 and 6 correspond to our expectations and common experience regarding the curvature and ridge properties of simple geometric shapes and human bodies. In addition, the quantitative tests carried out during the process of numerical evaluation confirm that the results are reasonable and correct within appropriate tolerance levels (as described in chapter 3), which is mainly related to the fact that the initial surface representations on which we work are sampled point clouds where: (a) the sampling is of varied quality and may introduce artefacts and errors; and (b) and there is no a priori analytic representation for the surface whose characteristics we are trying to calculate. The design of the methods developed throughout this research has been strongly focused on addressing these two obstacles, and our results indicate that this seems to have been successfully accomplished. Starting with sampled point clouds of our test objects and no other information, our proposed method returns curvatures and feature maps that are remarkably close to those expected. Moreover, when the input data consists of sampled surfaces of human bodies (i.e. 3D body scans), both the curvature and feature maps correspond to our intuitive perception of human body shape.

7.2 Conclusions

Although this research has not addressed the detection of a complete set of surface geometry features on a human body, its outcome does provide a starting point towards a fully automatic human body type classification and shape analysis from 3D scanner data. This we have achieved by overcoming the first major obstacle to providing a reliable analytic surface representation, namely the lack of reliable structure in the scan data and the existence of possible gaps and artefacts that are bound to contaminate the data as a result of the scanning techniques used. In that sense, this research constitutes a contribution towards the solution of the problem of converting *data* from 3D scanners to useful surface and shape *information* about the human body.

Moreover, this research contributes towards the development of hardware-independent shape analysis methods. This latter contribution is twofold:

(a) The proposed techniques are flexible with regard to the structure of the initial data. The techniques offer variants that allow them in practice to perform equally well on point clouds that are unorganized, semi-organized (i.e. accompanied by partial information about the connectivity of the points) or fully organized (i.e. triangulated).

(b) The proposed techniques are coordinate independent and therefore ideally suited to objects of arbitrary topology whose geometry generally *cannot* be represented by height functions. This includes free-form surfaces with branches, such as human bodies.

From a more mathematical point of view, there is a further level of significance related to the coordinate independence, since that is achieved by using implicit functions. This seems to deviate slightly from the standard practice of describing the differential geometry of objects by using height functions or parametric surface representations. Indeed, the calculation of the differential properties of implicit surfaces seems to be somewhat neglected in the sense that it seldom appears in the literature. However, the concept of implicit functions is by no means new, and their use offers a new potential in attempting to overcome the limitations and difficulties introduced by other kinds of surface representation when dealing with objects of arbitrary topology.

Finally, an important aspect of this research is that all proposed methods are fully automated. This should allow them to be incorporated in automated systems for shape analysis and classification (e.g. for applications related to medical research and the clothing industry). Such systems could therefore be operated by semi-expert or even non-expert users without the need for introducing human intervention steps that would inevitably affect the repeatability (and therefore the reliability and usefulness) of the results. Such full automation could therefore enable the development of novel software systems that utilise 3D human body shape information more fully than at present.

7.3 Investigation of the Hypothesis

As stated early in this thesis, our hypothesis was that:

- we can *reliably* construct *curvature maps* of human body surfaces from *3D scans*, such that:
- we can *reliably* extract *point features*,
- we can *reliably* extract *line features*,
- we can *classify* the surface.

With regard to the first point of the hypothesis, the results presented in section 5.5 clearly indicate that it is possible to implement an algorithm that takes a 3D scan as input, and produces a map of the curvature properties of the surface. The numerical evaluation that follows immediately after these results illustrates their reliablity, robustness and accuracy, provided that the aforementioned 3D scan meets certain quality criteria that form part of the set of assumptions stated in section 3.1.1. It can therefore be claimed that this point still holds true. We recognise, of course, that the scientific method is to attempt to falsify hypotheses. Strictly, therefore, we have failed to falsify this point of the hypothesis.

With regard to the second and third points of the hypothesis, the results presented in section 6.6 clearly indicate that the local surface representations (fitted quadric patches) and the curvature properties derived therefrom can be used to estimate, on the sampled point cloud that represents our initial surface, the location of point features such as ordinary umbilics and line features such as ridges, both of which are known to be important in describing an object's shape and topology. The numerical evaluation that follows immediately after these results illustrates their reliablity, again given that our initial set of assumptions holds true. Elsewhere in chapter 6 it is described how it would also be possible to use the curvature properties in order to estimate the locations of other point and line features such as minimal saddle points, parabolic lines, and sub-parabolic lines (flexcords). A few illustrative examples are given, but the results of such detections are not discussed or evaluated in the same detail as for ridges and umbilics. This is partly because some such features are not deformation-invariant and therefore tend to *re-form* as the surface deforms (i.e. they are of a lesser significance for the purposes of classifying and describing surface shape), as in the case of minimal points and lines of curvature. For others, there is an insufficient number of concrete examples of their locations and visual profiles, thus making it impossible to carry out a reliable quantitative evaluation of the results, as in the case of flexcords. However, the point is made that the detection of such features using the calculated surface curvature properties is not only possible but also relatively straightforward and could be incorporated in the development of a software system, if a particular application requires it. It is therefore reasonable to claim that these two points of the hypothesis have at least been partially met.

Finally, regarding the fourth point of the hypothesis, in principle objects could be classified according to their shape by investigating the configuration of features such as ridges and umbilics. These configurations follow strict rules, and may only change as the surface deforms from one shape to the other if a singularity (such as the birth or death of an umbilic) occurs. A relevant classification scheme has not been implemented as it is beyond the scope of this research. Indeed, as stated early in this thesis (section 1.4) such an implementation was neither expected not intended. However, the results of our curvature calculation (section 5.5) and ridge detection (section 6.6) implementations indicate that there are distinct configurations that are similar for individuals with similar body shape, and different for individuals of different body types. This constitutes a step in the right direction, as it indicates that it may indeed be possible to build a classification system that is based on the detection of features from the curvature properties, and that such classification might be fully automated with no human operator intervention once the input (a 3D scan) has been fed into the system. Although we have shown that the configuration of surface features is sensitive to shape, it is clear that two outstanding issues must be dealt with before such a scheme could be implemented:

(i) The feature detection and characterisation would itself have to be as consistent and complete as possible.

(ii) A notion of scale must be introduced, as small differences in the shape of the objects can always cause births, deaths, perestroikas and other topological changes.

The bumpy sphere used as one of our test objects in chapters 5 and 6 is an example of (ii). Any classification scheme must therefore incorporate the notions of scale and level of noise. In addition, if the ideal consistency and completeness of (i) is unattainable, the scheme must allow for these inevitable deficiencies to be taken into account. Clearly, further work is necessary for such processes to be fully and rigorously implemented, illustrated and evaluated.

Following this investigation, it can therefore be claimed that the majority of basic issues underlying curvature calculation, feature extraction and shape classification have been addressed. Although a number of additional issues have been generated in the course of this research, the quality of the results indicates both that this direction is worth pursuing further, and that it is feasible to attempt to address the remaining issues. There is thus ample scope for further work as described in the next section.

7.4 Further Work

In the course of the research, a number of issues arose that warrant further attention. As a result, the following directions have been identified for potential further work.

7.4.1 Patch Fitting with Orthogonal Distance Minimization

The choice of minimizing the algebraic distances in order to carry out the local surface patch reconstruction was a reasonable compromise on accuracy for the sake of algebraic and computational convenience. However, it would still be interesting and important to implement orthogonal distance regression and evaluate it against the current proposed method of minimization. If the results of such minimization lead to significantly more accurate estimation of curvature properties and surface features, then it is probably worth tolerating the computational overhead added as a result of the iterative solution of the equations for orthogonal distance minimization, especially since the speed and performance of current computer systems has improved dramatically since this research commenced. An important issue, therefore, is whether such iterative, non-linear fitting methods can be made sufficiently reliable, i.e. that we can be confident that, when they converge, they do so to the correct solution [ARCW02].

7.4.2 Evaluation

The range of evaluation tests carried out during this research has been focused on the objectives dictated by the hypothesis. More specifically, the tests have been designed to test whether sufficient ridge information is generated in order that we might describe and classify the surface. However, a number of other tests may be worth carrying out, which would help increase our understanding of surface features and the ways to measure and detect them.

Firstly, it would be interesting to carry out some tests on the results of calculating the principal curvature directions. Currently, the quality of the result of that particular calculation is assessed only indirectly: the directions are used for detecting ridges. If the ridge detection is found to be accurate enough, it can be assumed that the calculated principal curvature directions are good enough, at least for that purpose. However, it is essential also to carry out direct tests, using test objects such as those employed in this research, and compare them with the analytically calculated results obtained using the methods described in Appendix B4. This is not necessarily a straightforward task, as it introduces an issue of correspondence. There are two directions associated with each point, and a number of criteria ought to be devised in order to determine which pairs of vectors need to be compared. This, in turn, introduces the issue of separation of curvature fields (i.e. the ubiquitous separation of "red" and "blue" curvatures), and perhaps also the need for some more sophisticated tracking of features across the surface, one that should possibly involve the matching of templates and patterns.

Secondly, the evaluation of the results for local surface reconstruction requires a further investigation, especially for the cases where the minimization produces obviously poor results (i.e. when the lowest eigenvalue is not low enough) or even when it fails altogether (i.e. when the first eigenvalue is not significantly smaller than the second one). Such an investigation, starting perhaps with some form of visualisation of the error value maps on the surface, should aim to provide a better understanding of exactly how, when and why the fitting process fails. In particular, it is interesting to attempt to identify the local sample point configurations and local shape profiles that cause the local surface reconstruction algorithm to underperform. This may then enable significant improvements in the design and implementation of minimization techniques for local surface reconstruction.

Finally, there is one missing link in the overall evaluation strategy: the quantitative evaluation of results on the human body scans. The results on simple test objects look promising enough, and they have been possible to carry out because there have been available analytic representations for the test objects that enabled us to pre-calculate their curvature properties and compare them against the results of our proposed algorithms. Until recently, however, there has been no known alternative method for calculating the curvature properties of a free-form object such as a human body in a manner that would allow a credible comparison. This situation may soon change, as a result of recent advances on the modelling of surfaces by means of Radial Basis Functions (RBFs). These surface primitives are essentially implicit surfaces, and have been known for a few years as a method of surface reconstruction [TuOB99][BBCS99]. They have the advantage of providing local analytic descriptions of surface shape that are differentiable, thus enabling the calculation of curvature properties using the methods presented in this thesis. However, during most of the duration of this research, all implemetations of RBF fitting available in the literature have been known to be computationally demanding and therefore unsuitable for use on high-resolution human body scans such as those of interest for our research (i.e. clouds of a few tens of thousands of points). However, some more efficient implementations have been developed recently [CBCM01] that seem to overcome the computational problems, and since RBF fitting is compatible with the methods proposed here, a comparison test should now be feasible. Such testing should also provide a detailed list of advantages and disadvantages of choosing RBFs over simpler surface primitives such as our quadric surface patches.

7.4.3 Interpolation and Locality of Response

The detection of ridges and umbilic points in this research has been limited by the discretised nature of our data. The proposed algorithms really only detect near-umbilic and near-ridge points that tend to converge to the actual locations of these features as the sampling density increases. However, it would be desirable to devise some scheme for detecting more precise locations of these features from these near-ridges and near-umbilic points. This direction for further work is perhaps the most complicated of all those discussed in this chapter, as it most certainly involves some sort of interpolation. Such interpolation could be carried out between the values of the curvature properties, or even between the locally fitted quadric patches. The latter approach should lead to the generation of analytic local surface representations that cover larger areas of the body surface than those covered by the quadric patches. Moreover, these representations are likely to be of a higher order, which would in turn make them suitable for precise detection and tracking of the higher-order features of interest. As in the case of orthogonal distance mimimisation discussed earlier in this chapter, the tracking and pinpointing of features will involve work related to the tracking of curvature labels for the separation of curvature fields, and (perhaps) the use of pattern and template matching. The gain from such effort would be a further understanding of the possible configurations of features on a surface, a more complete, understandable and implementable mathematical framework for description of the features and topology of free-form surfaces, and maybe also the prospect of eventually managing to build automatically the complete maps of ridges, umbilics, and possibly even flexcords on the scan of any free-form 3D surface. This direction constitutes the direct and obvious sequel to the work presented in this thesis. In the context of modelling human bodies in particular, this would enable a more concise understanding of human body shape and the factors that affect it. Any such interpretative connections would have obvious advantages in the area of medical research.

7.4.4 Shape Classification

The ulimate test for both the accuracy and the usefuless of calculating surface shape characteristics of human bodies from 3D scanner data would be the design and implementation of a system capable of classifying human body shapes according to the configuration of surface shape features. There are a number of pre-requisites that need to be achieved before such a system can be designed. Firstly, a study needs to be carried out in order to *associate* configurations of ridges, flexcords and umbilic points with body shape types and at various scales. The rules that govern the deformations and singularities of such configurations would enable us to determine whether surface topologies are strictly different, and therefore corresponding to genuinely distinct body types. This would enable the definitions of surface shape *templates* for body shape types. To be meaningful, however, such a description would, as noted above, have to include a notion of scale[14]. Secondly, a set of criteria must be devised in order to enable *matching* between these templates and the features detected on the 3D scan of each human body that we wish to classify. The transformations involved in such a matching algorithm would most certainly allow some degree of deformation, and are also likely to be more efficient than current registration algorithms such as ICP, which are mainly suited to rigid objects and may converge to the wrong result even if properly initialised unless the two surfaces to be aligned have a precise partial overlap [BiFi96]. It *may* be possible to build such a matching system without first resolving the issues of tracking, interpolation and accurate location discussed above, and the existence of such a system would help solve some of the tracking problems discussed above (an attempt at such a curvature-based classification method is presented in [AbMK99]). However, it is clear that the matching and classification would improve dramatically if based on better and more complete feature detection results.

7.5 The Broader Context

Most of the work in this research has been focused on the calculation of curvature properties on surfaces represented as scan data. In the process, a great deal of empirical understanding was gained as to the meaning of curvature and its association with the underlying surface shape. It has been progressively possible to formulate this understanding in succinct mathematical terms. More importantly, this research has offered an opportunity to investigate diverse sources of literature for methods of calculating surface curvature and using it to codify surface shape. Some of the concepts discussed in these sources have emerged relatively recently and are only a few years old. Some others, such as the surface evolutes, date as far back as the 17th century, as Porteous accounts in the opening chapter of his book [Port94]. In retrospect, this work has been an attempt to utilise diverse pieces of knowledge into a compact framework that is both easy to understand (in terms of abstract concepts) and straightforward to implement computationally. It also became clear, from the very early stages of the work, that even an entity such as an unorganized point cloud may contain rich, higher-order information that is extractable. Some of this information is actually perceivable, as a result of both the function of our visual system, and of our (often unconscious) knowledge of human body shape and its properties. Therefore, the successful extraction of shape features from a point cloud or similar 3D scan must somehow incorporate that

---

[14] This is important because, as we saw in chapters 4, 5, 6, the success of the whole method in terms of quality of results and noise resistance relies heavily on the initial smoothing steps. These steps inevitably result in some body surace features to be eliminated. The choice of which such body features are safe to eliminate depends on the specific application.

knowledge. A compact (probably mathematical) framework that describes it and a concomitant computational system that utilises it are therefore highly desirable.

APPENDIX A : Bibliography – References

[AbMK99] Abbasi, S., Mokhtarian, F., Kittler, J., "Curvature Scale Space Image in Shape Similarity Retrieval", Multimedia Systems, Vol 7, pp467-476, Springer Verlag 1999.

[Akim78] Akima, H., "A Method of Bivariate Interpolation and Smooth Surface Fitting for Irregularly Distributed Data Points", ACM Transactions on Mathematical Software, Vol 4, No 2, pp148-159, June 1978.

[AmBK98] Amenta, N., Bern, M., Kamvysselis, M., "A New Voronoi-Based Surface Reconstruction Algorithm", Computer Graphics Proceedings, Annual Conference Series (SIGGRAPH'98), Vol. 32, pp415-421, ACM Press, ISBN 0-89791-999-8, Orlando, Florida, July 1998.

[APFG99] D'Apuzzo, N., Plaenkers, R., Fua, P., Gruen, A., Thalmann, D., "Modeling Human Bodies from Video Sequences", Electronic Imaging, Procs. of SPIE Photonics West Symposium, Vol. 3461 (Videometrics VI), pp. 36-47, El-Hakim, S,F., Gruen, A. (Eds.), San Jose, CA, January 1999.

[ARCW02] Ahn, S.J., Rauh, W., Cho, H.S., Warnecke H.J., "Orthogonal Surface Fitting of Implicit Curves and Surfaces", IEEE Trans PAMI, Vol 24, No 5, May 2002.

[BaCA96] Bardinet, E., Cohen, L., Ayache, N., "Tracking Medical 3D Data with a Deformable Parametric Model", Lecture Notes in Computer Science, Vol I (1064), Procs. of 4th European Conf. on Computer Vision (ECCV 1996), pp317-328, Buxton, B.F. and Cipolla, R. (Eds.), Springer Verlag, ISBN 3-540-61122-3 Cambridge, UK, April 1996.

[BaHo95] Baumberg, A., Hogg, D., "An Adaptive Eigenshape Model", Procs. of the 6th British Machine Vision Conference (BMVC95), Vol. 1, pp87-96, Pycock D., Ed., Birmingham, BMVA Press, 1995.

[BaIW93] Bajaj, C., Ihm, I., Warren, J., "Higher-Order Interpolation and Least-Squares Approximation Using Implicit Algebraic Surfaces", ACM Trans. on Graphics, Vol. 12, No. 4, pp327-347, October 1993.

[BaSh97] Barequet, G., Sharir, M., "Partial Surface and Volume Matching in Three Dimensions", IEEE Trans PAMI, Vol. 19, No.9, pp929-948, September 1997.

[BaTh99] Babski, C., Thalmann, D., "A Seamless Shape for HANIM Compliant

Bodies", Procs. of the 4th Symposium on Virtual Reality Modelling (VRML99), pp21-28, ACM Press, Paderborn, Germany, 1999.

[BBCS99] Bernardini, F., Bajaj, C.L., Chen, J., Schikore, D.R., "Automatic Reconstruction of 3D CAD Models from Digital Scans", Intl. Journal on Computational Geometry and its Applications, 9(4-5):327, 1999.

[BDDV00] Buxton B., Dekker L., Douros I., Vassilev T., "Reconstruction and Interpretation of 3D Whole Body Surface Images", Scanning 2000 Proceedings, Paris, May 2000.

[BeGo88] Berger, M., Gostiaux, B., "Graduate Texts in Mathematics - Differential Geometry: Manifolds, Curves and Surfaces", Springer Verlag 1988, ISBN 0-387-96626-9.

[BePK98] Belyaev, A., Pasko, A.A., Kunii, T.L., "Ridges and Ravines on Implicit Surfaces", Proc. of Computer Graphics International '98, pp530-535, IEEE Computer Society, ISBN 0-8186-8445-3, Hannover, June 1998.

[Besl92] Besl, P.J., "A method for registration of 3D shapes", IEEE Trans. PAMI, Vol.14, No.2, February 1992, pp239-256.

[BFHS98] Boulic, R., Fua, P., Herda, L., Silaghi, M., Monzani, J.S., Nedel, L., Thalmann, D., "An Anatomic Human Body for Motion Capture", EMMSEC'98, Bordeaux, September 1998.

[BiFi96] Bispo, E.M., Fisher, R.B., "Free-form Surface Matching for Surface Inspection", Procs of 6th IMA Conf. on Mathematics of Surfaces, 1996.

[BLCC00] Blane, M.M., Lei, Z., Civi, H., Cooper, D.B., "The 3L Algorithm for Fitting Implicit Polynomial Curves and Surfaces to Data", IEEE Trans. PAMI, Vol.22, No.3, pp298-313, March 2000.

[BlVe99] Blanz, V., Vetter, T., "A Morphable Model For The Synthesis Of 3D Faces", Computer Graphics Proceedings, Annual Conference Series, ACM SIGGRAPH, pp187-194, 1999.

[Boas83] Boas, M., "Mathematical Methods in the Physical Sciences", John Wiley & Sons 1983, ISBN 0-471-04409-1.

[BoBS85] Boggs, P.T., Byrd, R.H., Schnabel, R.B., "A Stable and Efficient Algorithm for NonLinear Orthogonal Distance Regression", Department of Computer Science, University of Colorado at Boulder, CU-CS-317-85, December 1985.

[Book79] Bookstein, F.L., "Fitting Conic Sections to Scattered Data", Computer Graphics and Image Processing, Part 9, pp56-71, 1979.

[Book89] Bookstein, F.L., "Principal Warps: Thin-Plate Splines and the Decompositions of Deformations", IEEE Trans. PAMI, Vol.11, No.6, pp567-585, June 1989.

[Book91] Bookstein, F.L., "Morphometric Tools for Landmark Data - Geometry and Biology", Cambridge University Press, 1991.

[BrGK95] Brechbuehler, C., Gerig, G., Kuebler, O., "Parametrisation of Closed Surfaces for 3-D Shape Description", CVIU, Vol.61, No.2, March 1995, pp154-170.

[BrGT96] Bruce, J.W., Giblin, P.J., Tari, F., "Ridges, Crests and Sub-Parabolic Lines of Evolving Surfaces", IJCV 18(3), pp195-210, 1995-1996.

[BrHT97] Brett, A.D., Hill, A., Taylor, C.J., "A Method of 3D Surface Correspondence for Automated Landmark Generation", Procs. of British Machine Vision Conference (BMVC97), pp709-718, Clark A.F. (Ed), ISBN 0-952-18987-9, Essex 1997.

[BrSt96] Brunnstroem, K., Stoddart, A.J., "Genetic Algorithms for Free-Form Surface Matching", procs. of 13th Int. Conference on Pattern Recognition, Vol IV. pp673-689, Vienna, Austria, 1996.

[BrTa98] Brett, A.D., Taylor, C.J., "A Method of Automated Landmark Generation for Automated 3D PDM Construction", Procs. of British Machine Vision Conference (BMVC98), pp914-923, Carter, J.N., and Nixon, M.S. (Eds.), ISBN 1-901725-04-9, Southampton 1998.

[CaCl78] Catmull, E., Clark, J., "Recursively generated B-spline surfaces on arbitrary topological meshes", Computer-Aided Design, 10:350-355, September 1978.

[Cann86] Canny, J., "A Computational Approach to Edge Detection", IEEE Trans. PAMI, Vol. 8, No. 6, November 1986.

[CaPo03] Cazals, F., Pouget, M., "Estimating Differential Quantities Using Polynomial Fitting of Osculating Jets", Procs. of Eurographics Symposium on Geometry Processing, pp177-275, Kobbelt, L., Schroeder, P., Hoppe, H. (Eds), The Eurographics Association 2003.

[CBCM01] Carr, J.C., Beatson, R.K., Cherrie, J.B., Mitchell, T.J., Fright, W.R., McCallum, B.C., Evans, T.R., "Reconstruction and Representation of 3D Objects with Radial Basis Functions", Computer Graphics Proceedings, Annual Conference Series (ACM SIGGRAPH 2001), pp67-76, ISBN 1-58113-374-X, Los Angeles, California, August 2001.

[CDTL95] Cootes, T.F., DiMauro, E.C., Taylor, C.J., Lanitis, A., "Flexible 3D Models from Uncalibrated Cameras", Proceedings of the 6th British Machine Vision Conference (BMVC95), pp147-156, Pycock D., Ed., Birmingham, BMVA Press, 1995.

[Chal02] Chalatsopoulou, D., "Obtaining a Mathematically Based Description of the Surface of Human Body Scans", MRes Dissertation, Department of Computer Science, UCL, 2002.

[ClDR00] Clarenz, U., Diewald, U., Rumpf, M., "Anisotropic Diffusion in Surface Processing", Proceedings of Visualization 2000, T. Ertl, B. Hamann and A. Varshney (Eds.), 2000, pp. 397-405.

[ClDR03] Clarenz, U., Diewald, U., Rumpf, M., "Processing Textured Surfaces via Anisotropic Geometric Diffusion", IEEE Transactions on Image Processing, 2003.

[CoCo96] Cohen, I., Cohen, L., "A Hybrid Hyperquadric Model for 2-D and 3-D Data Fitting", Computer Vision and Image Understanding, Vol 63, No 3, pp527-541, May 1996.

[CoET99] Cootes, T.F., Edwards. G., Taylor, C.J., "Comparing Active Shape Models with Active Appearance models", Procs. of 10th British Machine Vision Conference (BMVC99), Vol.1, pp173-182, Pridmore, T.P. and Elliman, D. (Eds.), ISBN 1-901725-09-X, Nottingham, September 1999.

[CoTa98] Cootes, T.F., Taylor, C.J., "Active Shape Models", Work Report, Department of Medical Biophysics, University of Manchester (available from: http://www.isbe.man.ac.uk/~bim/Models/asms.html), August 1998.

[CRAA04] Costa, L.d.F., Reis, S.F.d., Arantes, R.A.T., Alves, A.C.R., Mutinari, GC., "Biological Shape Analysis by Digital Curvature", Pattern Recognition 37, pp515-524, 2004.

[CTCG95] Cootes, T.F., Taylor, C.J., Cooper, D.H., Graham, J., "Active Shape Models - Their Training and Application", Computer Vision and Image Understanding, Vol. 61, No. 1, pp38-59, January 1995.

[CuSt98] Cunnington, S.J., Stoddart, A.J., "Self-calibrating surface reconstruction for the ModelMaker", Procs. of British Machine Vision Conference (BMVC98), Vol. 2, pp790-799, Carter, J.N., and Nixon, M.S. (Eds.), ISBN 1-901725-04-9, Southampton 1998.

[CzWa00] Csakany, P., Wallace, A.M., "Computation of Local Differential Parameters on Irregular Meshes", The Mathematics of Surfaces IX, Institute of Mathematics and its Applications, Cambridge 2000, pp19-33.

[DaJW98] Daanen, H.A.M., Jeroen van de Water, G., "Whole Body Scanners", Elsevier Displays 19 (1998), pp111-120.

[DDBT99] Dekker L., Douros I., Buxton B., Treleaven P., "Building Symbolic Information for 3D Human Body Modeling from Range Data", Proceedings of 2nd International Conference on 3D Digital Imaging and Modeling, (3DIM'99), pp388-397, IEEE Computer Society, Ottawa, Canada, ISBN 0-7695-0062-5, October 1999.

[DeHW97] Demers, M.H., Hurley, J.D., Wulpern, R.C., "Three-Dimensional Capture for Body Measurement using Projected Sinusoidal Patterns", Procs. of Electronic Imaging, SPIE Proceedings Vol. 3023, pp13-25, Three-Dimensional Image Capture, Richard N. Ellson; Joseph H. Nurre; Eds., 1997.

[Dekk00] Dekker, L., "3D Human Body Modelling from Range Data", PhD Thesis, University College London, 2000.

[DeMi02] Destrempes, F., Mignotte,M., "Unsupervised Statistical Method for Edgel Clustering with Application to Shape Localization". Procs. of 3rd Indian Conference on Computer Vision, Graphics and Image Processing, ICVGIP'02, Space Application Centre (ISRO), pp411-416, Ahmedabad, India, December 2002.

[Diet96] Dietz, U., "B-Spline Approximation with Energy Constraints", Reverse Engineering, B.G. Teubner Stuttgart, 1996 (Darmstadt University - http://www.mathematik.tu-darmstadt.de/ags/ag3/Projekte/BMBF/Lambrecht/proc_0.html).

[DMSB99] Desbrun, M., Meyer, M., Schroeder., P., Barr, A.H., "Implicit Fairing of Irregular Meshes using Diffusion and Curvature Flow", Computer Graphics Proceedings, Annual Conference Series, ACM SIGGRAPH, pp317-324, 1999.

[DoBu02] Douros, I., Buxton, B.F., "Three-Dimensional Surface Curvature Estimation using Quadric Surface Patches", Scanning 2002 Proceedings, Paris, May 2002.

[DoDB99a] Douros, I., Dekker, L., Buxton, B., "Reconstruction of the surface of the human body from 3D scanner data using B-splines", SPIE proceedings vol. 3640 (Three-Dimensional Image Capture and Applications II), pp234-245, Nurre, J.H. and Corner, B.R. Eds., San Jose, California, January 1999.

[DoDB99b] Douros, I., Dekker, L., Buxton, B., "An Improved Algorithm for Reconstruction of the Surface of the Human Body from 3D Scanner Data Using Local B-spline Patches", mPeople Workshop Proceedings, IEEE-ICCV99, pp29-36, Hilton, A., and Fua, P. Eds., Corfu, Greece, September 1999.

[Doo78] Doo, D., "A Subdivision Algorithm for Smoothing Down Irregularly Shaped Polyhedrons". Procs. of the International Conference on Interactive Techniques in Computer Aided Design, pp157-165. Bologna, Italy, IEEE Computer Society, 1978.

[DoSa78] Doo, D., Sabin, M., "Behaviour of Recursive Division Surfaces Near Extraordinary Points", Computer Aided Design, 10(6), pp356-360, September 1978.

[Dour99] Douros, I., "Human Body Modelling Standards and Activities", Interim Project Report, Centre for 3D Electronic Commerce, Department of Computer Science, University College London, July 1999.

[Dour04] Douros, I., "3D Scanning State-of-the-Art: Technologies, Trends and Applications Within and Beyond the Clothing Industry", Proceedings of the Workshops on 3D Scanning and Virtual Try-On Systems, Donghua Fashion Forum of Shanghai International Fashion Festival, China, March 2004.

[Eber99] Eberly, D., "Principal Curvatures of Surfaces", Magic Software, Inc., http://www.magic-software.com.

[FeAy94] Feldmar, J., Ayache, N., "Rigid and Affine Registration of Smooth Surfaces Using Differential Properties", Lecture Notes in Computer Science, Vol 801, ECCV 1994, pp397-406.

[FiFE97] Fisher, R.B., Fitzgibbon, A.W., Eggert, D.W., "Extracting Surface Patches from Complete Range Descriptions", IEEE Vol 1, pp148-155, 1997.

[FiFi95] Fitzgibbon, A.W., Fisher, R.B., "A Buyer's Guide to Conic Fitting", Proceedings of the 6th British Machine Vision Conference (BMVC95), Vol. 2, pp513-522, Pycock D., Ed., Birmingham, BMVA Press, 1995.

[FiPF99] Fitzgibbon, A., Pilu, M., Fisher, R.B., "Direct Least Square Fitting of Ellipses", IEEE Trans. PAMI, Vol.21, No.5, pp476-480, May 1999.

[Fitz01] Fitzgibbon, A W., "Robust Registration of 2D and 3D Point Sets", Proceedings of the British Machine Vision Conference (BMVC01), pp411-420, Cootes, T.F. and Taylor C. (Eds.), ISBN 1-901725-16-2, Manchester, BMVA Press, 2001.

[FlBH96] Fletcher, S., Bullpit, A., Hogg, D., "Global Alignment of MR Images Using a Scale Based Hierarchical Model", Lecture Notes in Computer Science, Volume 1 (1064), Proceedings of ECCV 1996, pp283-292, Buxton, B. and Cipolla, R, Eds, Cambridge, UK, Springer Verlag, April 1996.

[Floa00] Floater, M.S., "Meshless Parameterisation and B-Spline Surface approximation", Procs. Of The Mathematics of Surfaces IX, Institute of Mathematics and its Applications, pp1-18, Cipolla, R.and Martin, R. Eds, Springer, Cambridge, UK 2000.

[Foer96] Foerstner, W., "10 Pros and Cons Against Performance Characterisation Algorithms", Workshop on Performance Characteristics of Vision Algorithms, Cambridge, 1996

[FoPo01] Forsyth, D., Ponce, J., "Computer Vision: A Modern Approach", Prentice Hall 2001, ISBN: 0-13-085198-1.

[FrRS98] Frantz, S., Rohr, K., Stiehl, H.S., "Multi-Step Procedures for the Localisation of 2D and 3D Point Landmarks and Autmatic ROI Size Selection", Lecture Notes in Computer Science - Proceedings of the 5th European Conference on Computer Vision (ECCV98), Vol. 1 (1406), pp687-703, Burkhardt, H. and Neumann, B. (Eds.), ISBN 3-540-64569-1, Freiburg, Germany, June1998.

[FuBr96] Fua,P., Brechbuehler,C., "Imposing Hard Constraints on Soft Snakes", Lecture Notes in Computer Science, Volume 1 (1064), Proceedings of ECCV 1996, pp495-506, Buxton, B. and Cipolla, R, Eds, Cambridge, UK, Springer Verlag, April 1996.

[FuPT98] Fua, P., Plaenkers, R., Thalmann, D., "Realistic Human Body Modelling", Fifth International Symposium on the 3-D Analysis of Human Movement, Chattanooga, TN, July 1998.

[GaGS94] Gander, W., Golub, G.H., Strebel, R., "Least Squares Fitting of Circles and Ellipses", BIT, Vol 34, No 4, pp558-578,1994.

[Gass83] Gasson, P., "Geometry of Spatial Forms", Ellis Horwood Limited, ISBN 0-85312-620-8, Chichester 1983.

[Glas84] Glassner, A.S., "Space Subdivision for Fast Ray Tracing", IEEE

Computer Graphics and Applications, Vol. 4, No. 10, pp15-22, October 1984.

[GoAy93] Gourdon, N., Ayache, N., "Registration of a Curve on a Surface Using Differential Properties", ISSN 0249-6399, INRIA 1993.

[Good91] Goodall, C., "Procrustes Methods in the Statistical Analysis of Shape", Journal of the Royal Statistical Society B, Vol. 53, No. 2, pp285-239, 1991.

[Gord91] Gordon, A. P., "Face Recognition Based on Depth Maps and Surface Curvature", Procs of SPIE Conference on Geometric Methods in Computer Vision, Vol. 1570, pp234-247, San Diego CA, 1991.

[GoRe70] Golub, C.H., Reinsch, C., "Handbook series linear algebra. Singular value decomposition and least squares solutions", Numerical Mathematics, Vol.14, 1970, pp403-420.

[Gour97] Goursat, E. "Étude des Surfaces Qui Admettent tous les Plans de Symmétrie d'un Polyèdre Régulier." Annals Sci. École Normal Superieure, Vol 4, 159-2000, 1897.

[GoVL96] Golub, G., Van Loan, C.F., "Matrix Computations" - Third Edition, The Johns Hopkins University Press, Baltimore 1996, ISBN 0-8018-5414-8.

[Gowe71] Gower, J.C., "Statistical methods of comparing different multivariate analyses of the same data", in Mathematics in the Archaeological and Historical Sciences, Hodson, F.R., Kendall, D.G., Tautu, P., Edinburgh University Press, 1971, pp138-149.

[Gowe75] Gower, J.C., "Generalized Procrustes Analysis", Psychometrika, Vol. 40, No. 1, pp 33-51, March 1975.

[GuBM00] Guest, E., Berry, E., Morris, D., "Using the CSM Correspondence Calculation Algorithm to Quantify Differences Between Surfaces", BMVC 2000, Vol II, pp456-465.

[GuTa99] Guichard, F., Tarel, J.P., "Curve Finder Combining Perceptual Grouping and a Kalman Like Fitting", Procs. of 7th International Conference on Computer Vision (ICCV'99), pp1003-1008, Corfu, Greece, September 1999.

[HaFl96] Halir, R., Flusser, J., "Numerically Stable Direct Least Squares Fitting of Ellipses", Proceedings of International Conference of Pattern Recognition, August 1996.

[HaFr99] Hancock, P.J.B., Frowd, C.D., "Evolutionary Generation of Faces", Procs. Of AISB99, pp93-99, 1999.

[HaMM00] Hall, P., Marshall, D., Martin, R., "Merging and Splitting Eigenspace Models", IEEE Trans PAMI Vol 22, No 9, September 2000.

[HareXX] Har'el, S., "Curvatures of curves and surfaces - a parabolic approach", Department of Mathematics, Technion - Israel Institute of Technology.

[Hart95] Hartmann, E., "Blending an Implicit with a Parametric Surface", Computer Aided Geometric Design 12 (1995), 825-835, esp. appendix on implicit surfaces (p.834).

[HaSt88] Harris, C., Stephens, M., "A combined Corner and Edge Detector", Procs of the 4th ALVEY Vision Conference, pp147-151, The Plessey Company plc., 1988.

[HaZi00] Hartley, R., Zisserman, A., "Multiple View Geometry", Cambridge University Press, ISBN: 0521623049, March 2000.

[HBGS99] Hilton, A., Beresford, D., Gentils, T., Smith, R., Sun, W., "Virtual People: Capturing Human Models to Populate Virtual Worlds", Procs of IEEE Intl. Conf. On Computer Animation, pp174-185, 1999.

[HDDM93] Hoppe, H., DeRose, T., Duchamp, T., McDonald, J., Stuetzle, W., "Mesh Optimisation", Computer Graphics Proceedings, Annual Conference Series (SIGGRAPH'93), pp19-26, ACM Press, ISBN 0-89791-601-8, Anaheim, California, August 1993.

[HeFG99] Hermosillo, G., Faugeras, O., Gomes, J., "Cortex Unfolding Using Level Set Methods", INRIA Technical Report, Theme 3, No. 3663, April 1999.

[HeGa94] Heckbert, P.S., Garland, M., "Multiresolution Modeling for Fast Rendering", Proc. Graphics Interface 1994, Canadian Inf. Proc. Soc., pp43-50.

[HeGa99] Heckbert, P.S., Garland M., "Optimal Triangulation and Quadric-based Simplification", Computational Geometry Theory and Applications, 1999, pp49-65.

[HeHo97] Heap, T., Hogg, D., "Improving Specificity in PDMs using a Hierarchical Approach", Procs. of British Machine Vision Conference (BMVC97), pp80-89, Clark A.F. (Ed), ISBN 0-952-18987-9, Essex 1997.

[HHNH01] Hammond, P., Hutton,T.J., Nelson-Moon,Z.L., Hunt,N.P., Madgwick, A.J.A., "Classifying Vertical Facial Deformity using Supervised and Unsupervised Learning", Methods of Information in Medicine 40, pp365-372, 2001.

[HiBT97] Hill, A., Brett, A.D., Taylor, C.J., "Automatic Landmark Identification using a New Method of Non-rigid Correspondence", Proceedings of IPMI '97 Conference, vol. 1230, pp483-488, Springer Verlag 1997.

[HiGe98] Hilton, A., Gentils, T., "Popup people: Capturing human models to populate virtual worlds", Submitted to SIGGRAPH'98, January 1998.

[HiIW95] Hilton, A., Illingworth, J., Windeatt, T., "Statistics of Surface Curvature Estimates", Pattern Recognition, Vol 28, No 8, pp1201-1221, 1995.

[Hilt99] Hilton, A., "Towards Model-Based Capture of a Person's Shape, Appearance and Motion", mPeople Workshop Proceedings, IEEE-ICCV99, pp37-44, Hilton, A. and Fua, P. Eds., Corfu, Greece, September 1999.

[Hopp94] Hoppe, H., "Surface reconstruction from unorganized points", PhD Thesis, Department of Computer Science and Engineering, University of Washington, June 1994.

[Hori95] Horiguchi, C., "Sensors That Detect Shapes", Development Report, Hamamatsu Photonics, Journal of Advanced Automation Technology Vol.7, No.3, pp210-216, 1995.

[Hori98] Horiguchi, C., "BL (Body Line) Scanner - The development of a new 3D measurement and reconstruction system", Hamamatsu Photonics, International Archives of Photogrammetry and Remote Sensing, Col. XXXII, Part 5, Hakodate, pp421-429, 1998.

[Horn87] Horn, B.K.P., "Closed-form Solution of Absolute Orientation Using Unit Quaternions", Journal of the Optical Society of America, Vol.4, No.4, April 1987, pp629-642.

[HoTT88] Hong, T.M., Thalmann, N.M., Thalmann, D., "A General Algorithm for 3D Shape Interpolation in a Facet-Based Representation", Graphics Interface, pp229-235, 1988.

[HSIW96] Hilton, A., Stoddart, A., Illingworth, J., Windeatt, T., "Reliable Surface Reconstruction from Multiple Range Images", Lecture Notes in Computer Science, Volume 1 (1064), Proceedings of ECCV 1996, pp117-126, Buxton, B. and Cipolla, R, Eds, Cambridge, UK, Springer Verlag, April 1996.

[HuBH01] Hutton, T., Buxton, B.F., Hammond, P., "Dense Surface Point Distribution Models of the Human Face", Procs. Of IEEE Workshop on

Mathematical Methods in Biomedical Image Analysis, pp153-160, Hawaii, 2001.

[HuCH00] Hutton, T.J., Cunningham, S., Hammond, P., "An Evaluation ofActive Shape Models for the Automatic Identification of Cephalometric Landmarks", European Journal of Orthodontics 22(5), 2000, pp499-508.

[HuHD99] Hutton, T.J., Hammond, P., Davenport, J.C., "Active Shape Models for Customised Prosthesis Design", Proceedings of the Joint European Conference on Artificial Intelligence in Medicine and Medical Decision Making, Lecture Notes in Artificial Intelligence 1620, pp448-452, Horn et al. Eds., Springer Verlag 1999.

[JaTa80] Jackins, C.L., Tanimoto, S., "Oct-trees And Their Use in Representing 3D Objects", Computer Graphics and Image Processing, Vol. 14, No. 3, pp249-270, November 1980.

[JoBW95] Jones, P.R.M., Brooke-Wavell, K., West, G., "Format for Human Body Modelling from 3D Body Scanning", HUMAG Research Group, Loughborough University, International Journal of Science and technology, Vol.7, No.1, 1995 (pp. 7-16).

[JoHe99] Johnson, A., Hebert, M., "Using Spin Images for Efficient object recognition in cluttered 3D scenes", IEEE Trans PAMI, 21 (1999), pp433-449.

[Joll02] Jolliffe, I.T., "Principal Component Analysis", 2nd Edition, Springer Verlag, ISBN 0387954422, October 2002.

[KaDT03] Kartsounis, G., Douros I., Treleaven, P.C., "BodyXML - An XML-based High Level Representation Framework of Human Bodies", 3D Modelling 2003 Proceedings, Paris, April 2003.

[KaSK00] Kanai, T., Suzuki, H., Kimura, F., "Metamorphosis of Arbitrary Triangular Meshes" , IEEE Computer Graphics and Applications, March/April 2000, pp62-75.

[KHGB95] Kumar, S., Han, S., Goldgof, D., Bowyer, K., "On Recovering Hyperquadrics from Range Data", IEEE Trans on Pattern Analysis and Machine Intelligence, Vol 17, No 11, pp1079-1083, 1995.

[KLML96] Kent, J.T., Lee, D., Mardia, K.V., Linney, A.D., "Using Curvature Information in Shape Analysis", Proceedings in Image Fusion and Shape Variability Techniques (Eds. KV Mardia, CA Gill and IL Dryden). Leeds University Press. 1996, pp.88-99. (ISBN 0 85316 1739)

[KMMA98] Kalra, P., Magnenat-Thalmann, N., Moccozet, L., Aubel, A., Thalmann,D., "Real-Time Animation of Realistic Virtual Humans", IEEE Computer Graphics and Applications, pp42-56, September/October 1998.

[Koen90] Koenderink, J.J., "Solid Shape", MIT Press, 1990.

[KoTa98] Kotcheff, A.C.W., Taylor, C.J., "Automatic Construction of Eigenshape Models by Direct Optimisation", Medical Image Analysis, vol. 2, pp. 303-314, 1998.

[KoVD94] Koenderink, J.J., van Doorn, A.J., "Two-plus-one-dimensional differential geometry", Pattern Recognition Letters, v.15(5), pp 439-443, March 1994.

[KoVD98] Koenderink, J.J., van Doorn, A.J., "The Structure of Relief", Advances in Imaging & Electron Physics, vol.103, pp.65-150, 1998.

[KoVD98b] Koenderink, J.J., van Doorn, A.J., "Shape from Chebyshev Nets", Lecture Notes in Computer Science, Proceedings of the 5th European Conference on Computer Vision (ECCV98), Vol. II (1407), pp215-225, Burkhardt, H. and Neumann, B. (Eds.), ISBN 3-540-64569-1, Freiburg, Germany, June1998.

[Krey59] Kreyszig, E., "Differential Geometry", Mathematical Expositions No. 11, Oxford University Press 1959.

[LaTC99] Lanitis, A., Taylor, C.J., Cootes, T.F., "Modeling the Process of Ageing in Face Images", IEEE Proceedings of the 7th International Conference on Computer Vision, Vol. 1, pp131-142, Corfu, Greece, September 1999.

[LCRH02] Lee, J.H., Chai, J.X., Reitsma, P.S.A., Hodgins, J.K., Pollard, N., "Interactive Control of Avatars Animated with Human Motion Data", Computer Graphics Proceedings, Annual Conference Series (SIGGRAPH'02), pp491-500, ACM Press, ISBN 1-58113-521-1, San Antonio, Texas, July 2002.

[LeKi95] Lee, S., Kim, Y.J., "Direct Extraction of Topographic Features for Gray Scale Character recognition", IEEE Trans. PAMI, Vol.17, No.2, pp724-729, July 1995.

[LiJo94] Li, P., Jones, P.R.M., "Anthropometry-Based Surface Modeling of the Human Torso", HUMAG Research Group, Loughborough University, Proceedings of the 1994 ASME International Conference and Exhibition, pp. 469-474, Minneapolis 1994.

[LoCl87] Lorensen, W.E., Cline, H.E., "Marching Cubes: a High-Resolution 3D Surface Construction Algorithm", Computer Graphics, Vol.21, No.4, (SIGGRAPH '87 Conference Proceedings), pp163-169, July 1987.

[LoEF95] Lorusso, A., Eggert, D.W., Fisher R.B, "A Comparison of Four Algorithms for Estimating 3-D Rigid Transformations", Proceedings of the 6th British Machine Vision Conference (BMVC95), pp237-246, Pycock D., Ed., Birmingham, BMVA Press, 1995.

[LSSC98] Lee, A.W.F., Sweldens, W., Schroeder, P., Cowsar, L., Dobkin, D., "MAPS: Multiresolution Adaptive Parameterisation of Surfaces", Computer Graphics Proceedings, Annual Conference Series (SIGGRAPH'98), pp95-104, ACM Press, ISBN 0-89791-999-8, Orlando, Florida, July 1998.

[LuHa99] Luo, B., Hancock, E.R., "Matching Point-sets using Procrustes Alignment and the EM Algorithm", Proceedings of 10th British Machine Vision Conference,Vol.1, pp43-52, Pridmore, T.P. and Elliman, D. Eds., BMVA Press, 1999.

[LuMM98] Lukacs, G., Martin, R., Marshall, D., "Faithful Least-Squares Fitting of Spheres, Cylinders, Cones and Tori for Reliable Segmentation", Lecture Notes in Computer Science, Vol I, ECCV 1998, pp671-686.

[MITe95a] McInerney, T., Terzopoulos, D., "A Dynamic Finite Element Surface Mode for Segmentation and Tracking in Multidimensional Medical Images with Application to Cardiac 4D Image Analysis", Journal of Computerized Medical Imaging and Graphics (Special Issue on Cardiopulmonary Imaging.), Vol. 19, Part 1, pp69-83, January 1995.

[MITe95b] McInerney, T., Terzopoulos, D., "Topologically Adaptable Snakes", Procs. of 5th International Conference on Computer Vision, pp 840-845, Cambridge, MA, USA, June 1995.

[MITe97] McInerney, T., Terzopoulos, D., "Medical Image Segmentation Using Topologically Adaptable Surfaces", Proc. First Joint Conference of Computer Vision, Virtual Reality, and Robotics in Medicine and Medical Robotics and Computer-Assisted Surgery (CVRMed-MRCAS'97, Grenoble, France), Lecture Notes in Computer Science, Vol. 1205, pp23-32, Springer-Verlag, Berlin, 1997.

[MoKY00] Mokhtarian, F., Khalili, N., Yuen, P., "Free-Form 3-D Object Recognition at Multiple Scales", BMVC 2000, Vol II, pp446-455.

[MoKY01] Mokhtarian, F., Khalili, N., Yuen. P., "Multi-Scale Free-Form 3-D Object recognition using 3D models", Image and Vision Computing 19, 2001, pp271-281.

[MoKY98] Mokhtarian, F., Khalili, N., Yuen. P., "Multi-Scale 3-D Free-Form Surface Smoothing", Procs. of British Machine Vision Conference (BMVC98), pp730-739, Carter, J.N., and Nixon, M.S. (Eds.), ISBN 1-901725-04-9, Southampton 1998.

[Morr95] Morris, R., "The Use of Computer Graphics for Solving Problems in Singularity Theory", Visualization and Mathematics: Experiments, Simulations and Environments, Hege, H.C. and Polthier, K. (Eds), Springer Verlag, ISBN 3-540-61269-6, Heidelberg 1995.

[Morr96] Morris, R., "The Sub-Parabolic Lines of a Surface", Mathematics of Surfaces VI, IMA new series 58, pp79-102, Mullineux, G. (Ed.), Clarendon Press, Oxford, 1996.

[NeTh98] Nedel, L.P., Thalmann, D., "Modelling and Deformation of the Human Body Using an Anatomically-Based Approach", Proceedings of Computer Animation'98, pp34-40, Philadelphia, Pennsylvania, USA, June 1998.

[NFSK95] Neuenschwander, W., Fua, P., Szekely, G., Kuebler, O., "Deformable Velcro Surfaces", IEEE Proc. of the Fifth International Conference on Computer Vision (ICCV '95), pp. 828-833, Cambridge, MA, June 1995.

[NgNS99] Nguyen,V., Nzomigni,V., Stewart,C.V., "Fast and Robust Registration of 3D Surfaces Using Low Curvature Patches", Proceedings of 2nd International Conference on 3D Digital Imaging and Modeling, (3DIM'99), pp201-208, IEEE Computer Society, Ottawa, Canada, ISBN 0-7695-0062-5, October 1999.

[Nobl88] Noble, J.A., "Finding Corners", Image and Vision Computing, Vol.6, No.2, pp121-128, May 1988.

[Nurr97] Nurre, J., "Locating Landmarks on Human Body Scan Data", IEEE International Conference on Recent Advances in 3-D Digital Imaging and Modeling. pp289-295, Ottawa, Canada, 1997,.

[OBAT03] Ohtake, Y., Belyaev, A., Alexa, M., Turk, G., Seidel, H.-P., "Multi-Level Partition of Unity Implicits", Computer Graphics Proceedings, Annual Conference Series (SIGGRAPH'03), pp463-470, ACM Press, ISBN 1-58113-710-9, San Diego, California, July 2003.

[OhBe01] Ohtake, Y., Belyaev, A., "Mesh Optimization for Polygonized Isosurfaces", Eurographics 2001, Computer Graphics Forum, Vol 20, No. 3, pp368-376, 2001.

[OhBS02] Ohtake, Y., Belyaev, A., Seidel, H-P., "Mesh Smoothing by Adaptive and Anisotropic Gaussian Filter Applied to Mesh Normals", Procs. of Vision Modelling and Visualisation, pp203-210, Erlangen, Germany, 2002.

[OhBS03] Ohtake, Y., Belyaev, A., Seidel, H-P., "A Multi-Scale Approach to 3D Scattered Data Interpolation with Compactly Supported Basis Functions", Shape Modelling International '03, Seoul, Korea, May 2003.

[OlBu99] Oliveira, J., Buxton, B., "Creating Light-Weight Virtual Humans for Virtual Environments", Proceedings of EUROGRAPHICS'99, pp15-18, ISSN 1017-4656.

[OnGo99] Ong, E., Gong, S., "A Dynamic Human Model using Hybrid 2D-3D Representations in Hierarchical PCA Space", Proceedings of 10th British Machine Vision Conference, Vol.1, pp33-42, Pridmore, T.P. and Elliman, D. Eds., 1999.

[PaPo01] Pae, S.I., Ponce, J., "On Computing Structural Changes in Evolving Surfaces and their Appearance", International Journal of Computer Vision, Volume 43, Issue 2, July 2001, pp113-131.

[PaVG95] Pajdla, T., Van Gool, L., "Matching of 3-D Curves Using Semi-Differential Invariants", IEEE 1995.

[Pear01] Pearson, K., "On Lines and Planes of Closest Fit to Systems of Points in the Space", Philosophical Magazine 2, pp559-572, 1901.

[PFTV92] Press, W.H., Flannery, B.P., Teukolsky, S.A., Vetterling, W.T., "Numerical Recipes in C: The Art of Scientific Computing", 2nd Edition, ISBN: 0521431085, Cambridge University Press, October 1992.

[PiTi99] Piegl, L.A., Tiller, W., "Computing Offsets of NURBS Curves and Surfaces", Computer Aided Design 31 (1999), pp147-156.

[PlFA99] Plaenkers, R., Fua, P., D'Apuzzo, N., "Automated Body Modeling from Video Sequences", mPeople Workshop Proceedings, IEEE-ICCV99, pp45-52, Hilton, A. and Fua, P., Eds., Corfu, Greece, September 1999.

[PoPu00] Porteous, I.R., Puddephat, M.J., "Landmarks of a Surface", The Mathematics of Surfaces IX, Institute of Mathematics and its Applications, Cambridge 2000, pp114-125.

[Port88] Porteous, I., "The Circles of a Surface", IMA Conference on the Mathematics of Surfaces 1988, pp135-143.

[Port94] Porteous, I., "Geometric Differentiation", (In particular chapters 1,6,10-16), Cambridge University Press 1994.

[QiTe97] Qin, H., Terzopoulos, D., "Triangular NURBS and their Dynamic Generalizations", Computer Aided Geometric Design, vol. 14, 1997, pp325-347.

[RBDT02] Ruiz ,M., Buxton, B.F., Douros, I., Treleaven, P.C., "Web-based Software Tools for 3D Body Database Access and Shape Analysis", Scanning 2002 Proceedings, Harbour, Paris, May 2002.

[Rosi93] Rosin, P.L, "A Note on the Least Squares Fitting of Ellipses", Pattern Recognition Letters, No 14, pp799-808, 1993.

[RuBu01] Ruiz, M.C., Buxton, B., "A Model-Based Procedure for Fitting a Surface to Scans of Clothed People", Scanning 2001 Proceedings, Harbour, Paris, May 2001.

[SaDM02] Sapidis, N., Djebali, M., Melkemi, M., "A Fit-and-Merge Algorithm for Range-Image Segmentation and Model Reconstruction", Journal of Computing and Information Science in Engineering, ASME, Vol. 2, pp285-293, December 2002.

[Sam90a] Samet, H., "Applications of Spatial Data Structures", Addison-Wesley, Reading MA, 1990.

[Sam90b] Samet, H., "The Design and Analysis of Spatial Data Structures", Addison-Wesley, Reading MA, 1990.

[Samp82] Sampson, P.D., "Fitting Conic Sections to Very Scattered Data: An Iterative Refinement of the Bookstein Algorithm", Computer Graphics and Image Processing, No 18, pp97-18, 1982.

[SaZu92] Sander, P.T., Zucker, S.W., "Singularities of principal direction fields from 3d images", IEEE Trans PAMI Vol 14, No 3, March 1992.

[ScJe00] Schaufler,G., Jensen,H.W., "Ray Tracing Point Sampled Geometry", Procs. of 11th Eurographics Workshop on Rendering, pp319-328, Brno, Czech Republic, 2000.

[ScML98] Schroeder, W., Martin, K., Lorensen, B., "The Visualisation Toolkit: An Object-Oriented Approach to 3D Graphics", 2nd Edition, Prentice Hall 1998.

[ScNP98] Scott, M.J.J., Niranjan, M., Prager, R.W., "Realisable Classifiers:

Improving Operating Performance on Variable Cost Problems", BMVC Procs 1998, pp306-315.

[SCSA02] Starck, J., Collins, G., Smith, R., Hilton, A., Illingworth, J., "Animated Statues" Journal of Machine Vision Applications, 2002.

[ScZL92] Schroeder, W.J., Zarge, J.A., Lorensen, W.E., "Decimation of Triangle Meshes", Computer Graphics Proceedings, Annual Conference Series (SIGGRAPH'92) , Vol.26, No.2, pp65-70, ACM Press, ISBN 1-58113-521-1, San Antonio, Texas, July 1992.

[SeSS98] Sederberg, T.W., Sewell, D., Sabin, M., "Non-Uniform Recursive Subdivision Surfaces", Computer Graphics Proceedings, Annual Conference Series (SIGGRAPH'98), pp387-394, ACM Press, ISBN 0-89791-999-8, Orlando, Florida, July 1998.

[Shel54] Sheldon, W.H., "Atlas of Men: A Guide for Somatyping the Adult Male at All Ages". New Yoric: Harper & Row, 1954.

[Smit99] Smith, A.D.C., "The Folding of the Human Brain: From Shape to Function", PhD Thesis, University of London, September 1999.

[SmTA96] Smyth, P.P., Taylor, C.J., Adams, "Automatic Measurement of Vertebral Shape Using Active Shape Models", Procs of BMVC96, pp705-714, BMVA Press, 1996.

[SPTH99] Saachi, R., Poliakoff, J.F., Thomas, P.D., Haefele, K.H., "Curvature Estimation for Segmentation of Triangulated Surfaces", Proceedings of 3DIM'99, pp536-543, IEEE Computer Society, ISBN 0-7695-0062-5, Ottawa, Canada, 1999.

[Stam98] Stam, J., "Exact Evaluation of Catmull-Clark Subdivison Surfaces at Arbitrary Parameter Values", Computer Graphics Proceedings, Annual Conference Series (SIGGRAPH'98), pp395-404, ACM Press, ISBN 0-89791-999-8, Orlando, Florida, July 1998.

[StBa98a] Stoddart, A.J., Baker, M.S., "Progressive Splines", 10th IMDSP Workshop, Alpbach, Austria, July 1998.

[StBa98b] Stoddart, A.J., Baker, M.S., "Reconstructions of smooth surfaces with arbitrary topology adaptive splines", Proceedings of the 5th European Conference on Computer Vision (ECCV98), Burkhardt, H. and Neumann, B. (Eds.), ISBN 3-540-64569-1, Freiburg, Germany, Springer Verlag, June 1998.

[StBa98c] Stoddart, A.J., Baker, M.S., "Surface reconstruction and compression using multiresolution arbitrary topology G1 continuous splines", procs. of 14th Int. Conf. on Pattern Recognition, pp788-791, 1998.

[Step65] Stephenson, G., "An Introduction to Matrices, Sets and Groups for Science Students", Longmans, 1965.

[SuSP94] Sullivan, S., Sandford, L., Ponce, J., "Using Geometric Distance Fits for 3D Object Modelling and Recognition", IEEE Trans PAMI, Vol 16, No 12, December 1994.

[TaMe99] Tang, C.K., Medioni, G., "Robust Estimation of Curvature Information from Noisy 3D Data for Shape Description", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 24, Issue 6, pp858-864, June 2002.

[Taub94] Taubin, G., "Algebraic Curves and Surfaces in 3D Computer Vision", Procs. Of The Mathematics of Surfaces V, Institute of Mathematics and its Applications, Fisher, R. Ed, Springer, Oxford University Press, 1994.

[TBRB03] Tahan, A., Buxton, B., Ruiz, M., Bougourd, J., "Point Distribution Models of Human Body Shape from a Canonical Representation of 3D Scan Data", in: Procs. 3D Modelling'03, Harbour, Paris 2003.

[Thal90] Thalmann, D., "Human Modelling and Animation", Computer Animation, pp129-149. Springer-Verlag, 1990.

[ThOl03] Thodberg, H.H., Olafsdottir, H., "Adding Curvature to Minimum Description Length Shape Models", Procs. of 14th British Machine Vision Conference (BMVC03), Norwich 2003 (further details yet TBA)

[TiLe00] Timsari, B., Leahy, R., "An Optimisation Method for Creating Semi-Isometric Flat Maps of the Cerebral Cortex", SPIE Proceedings, Medical Imaging: Image Processing, Vol.3979, Hanson, K.M., Ed. pp698-710, 2000.

[TrFi95] Trucco, E., Fisher, R.B., "Experiments in Curvature-Based Segmentation of Range Data", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol 17, Part 2, 1995, pp171-182.

[TrSH00] Tricoche, X., Schauermann, G., Hagen, H., "Higher order singularities in piecewise linear vector fields", The Mathematics of Surfaces IX, Institute of Mathematics and its Applications, Cambridge 2000, pp99-113.

[Tsch78] Tschebyscheff, P.L., "Sur la Coupe des Vetements", Association francaise pour l'avancement des sciences, Congres de Paris, p154, 1878.

[TuLe94] Turk, G., Levoy, M., "Zippered Polygon Meshes from Range Images", Computer Graphics Proceedings, Annual Conference Series (SIGGRAPH'94), pp311-318, ACM Press, ISBN 0-89791-667-0, Orlando, Florida, 1994.

[TuOB99] Turk, G., O'Brien, J, "Shape Transformation Using Variational Implicit Functions", Computer Graphics Proceedings, Annual Conference Series (ACM SIGGRAPH 99), pp335-342, August 1999.

[UeRo99] Ueshiba, T., Roth, G., "Generating Smooth Surfaces with Bicubic Splines over Triangular Meshes: Toward Automatic Model Building from Unorganized 3D Points", Proceedings of 3DIM'99, pp302-311, IEEE Computer Society, ISBN 0-7695-0062-5, Ottawa, Canada, 1999.

[VWVG99] Vanden-Wyngaerd, J., Van Gool, L., Koch, R., Proesmans, M., "Invariant-based Registration of Surface Patches", IEEE Proceedings of the 7th International Conference on Computer Vision, Vol. 1, pp131-142, Corfu, Greece, September 1999.

[WaLo03] Walder, C.J., Lovell, B.C., "Kernel Based Algebraic Curve Fitting", Procs of 5th Intl. Conf. on Advances in Pattern Recognition (ICAPR03), pp387-390, Mukherjee, D.P. and Pal, S. (Eds.), ISBN 81-7764-532-3, Allied Publishers PVT Ltd., Kolkata, India, December 2003.

[WaPS00] Wang. Y., Peterson, B., Staib, L., "Shape-base 3D Surface Correspondence Using Geodesics and Local Geometry", Computer Vision and Pattern Recognition, Vol. 2 (2000), pp644-651.

[Wats00] Watson, G.A., "Least Squares Fitting of Parametric Surfaces to Measured Data", ANZIAM J., 42(E) ppC68-C95, 2000.

[WDFD00] Wells, J.C.K., Douros, I., Fuller, N.J., Elia, M., Dekker, L., "Assessment of Body Volume Using 3-dimensional Photonic Scanning", Ann. N.Y. Acad. Sci. 2000, v904, pp247-54.

[WeFR98] Werghi, N., Fisher, R.B., Robertson, C., "Modelling Objects Having Quadric Surfaces Incorporating Geometric Constraints", Proceedings of the 5th European Conference on Computer Vision (ECCV98), Vol. 2 (1407), pp185-201, Burkhardt, H. and Neumann, B. (Eds.), ISBN 3-540-64569-1, Freiburg, Germany, June1998.

[WFAR99] Werghi, N., Fisher, R.B., Ashbrook, A., Robertson, C., "Faithful Recovering of Quadric Surfaces From 3D Range Data", Proceedings of 3DIM'99, pp280-289, IEEE Computer Society, ISBN 0-7695-0062-5, Ottawa, Canada, 1999.

[YaFa98] Yamany, S., Farag, A., "Free-form Surface Registration Using Surface Signatures", Procs. of ICCV 1998, Vol 2., pp1098-1104.

[YuKM99] Yuen,P., Khalili, N., Mokhtarian, F., "Curvature Estimation on Smoothed 3D Meshes", Procs. of 10th British Machine Vision Conference (BMVC99), Vol.1, pp133-142, Pridmore, T.P. and Elliman, D. (Eds.), ISBN 1-901725-09-X, Nottingham, September 1999.

[Zhan95] Zhang, Z., "Parameter Estimation Techniques: A Tutorial Application to Conic Fitting", INRIA Research Report No 2676, October 1995.

APPENDIX B : Mathematical Background

B1: Local Fitting of Quadric Surface Patches

(An early version of the methodology presented in this Appendix was published in 2002 [DoBu02].)

Given a continuous surface B_s represented as a (sample) point cloud P_s, then for every point P in the point cloud we need to find the values of parameters a, b, c, e, f, g, l, m, n and h of a quadric surface function F that best describes the shape of B_s within a small neighbourhood $N(P)$ of P.

Our quadric surface function $F(x,y,z)$ is defined as:

$$F = ax^2 + by^2 + cz^2 + 2exy + 2fyz + 2gzx + 2lx + 2my + 2nz + h = 0 \qquad \text{(B1.1)}$$

which can conveniently be represented in matrix form as:

$$F(\underline{\alpha}, \underline{x}) = \underline{\alpha} \cdot \underline{x} = 0 \qquad , \qquad \text{(B1.2)}$$

where: $\underline{\alpha} = [a \quad b \quad c \quad 2e \quad 2f \quad 2g \quad 2l \quad 2m \quad 2n \quad h]^{\mathrm{T}}$ (B1.3a)

and: $\underline{x} = [x^2 \quad y^2 \quad z^2 \quad xy \quad yz \quad zx \quad x \quad y \quad z \quad 1]$ (B1.3b)

Without sacrificing generality, it is mathematically more stable if we choose the centroid of the points in $N(P)$ to be the origin of our local coordinates system (see [HaZi00]).

If there are enough points in the neighbourhood $N(P)$, then each datum $(x(i), y(i), z(i))$ will not lie perfectly on the surface φ, i.e. we shall have:

$$ax^2 + by^2 + cz^2 + 2exy + 2fyz + 2gzx + 2lx + 2my + 2nz + h = \varepsilon \qquad \text{(B1.4)}$$

ε is the algebraic error that we need to minimize with respect to $\underline{\alpha}$, i.e:

$$\min\left\{\sum_{i=1,\ldots,n}\varepsilon^2\right\} \tag{B1.5}$$

$$\Rightarrow \qquad \min\left\{\sum_{i=1,\ldots,n}F^2(\underline{\alpha},\underline{x}_i)\right\} \tag{B1.6}$$

Equation (B1.2) has an obvious degenerate solution with *all* elements of $\underline{\alpha}$ being simultaneously zero. We therefore need to introduce a constraint that prevents this, at the same time allowing any one or proper subset of them to have a zero value. It is then obvious (and also convenient, as will be shown below) that we can set the constraint so that the sum of their squares is positive and, say, equal to 1:

$$a^2 + b^2 + c^2 + h^2 + (2e)^2 + (2f)^2 + (2g)^2 + (2l)^2 + (2m)^2 + (2n)^2 = 1 \tag{B1.7}$$

or, in matrix notation: $\underline{\alpha}^T\underline{\alpha} = 1$. (B1.8)

We can then introduce a Lagrange multiplier, and (B1.6) becomes:

$$\min\left\{\sum_{i=1,\ldots,n}F^2(\underline{\alpha},\underline{x}_i) + N\lambda(1-\underline{\alpha}^T\underline{\alpha})\right\} \tag{B1.9}$$

$$\Rightarrow \qquad \min\left\{\sum_{i=1,\ldots,n}(\underline{\alpha}\cdot\underline{x}_i)^2 + N\lambda(1-\underline{\alpha}^T\underline{\alpha})\right\} \tag{B1.10}$$

We can proceed by differentiating (B1.19) with respect to $\underline{\alpha}$ and setting the derivatives to zero, thus obtaining:

$$2\left(\sum_{i=1,\ldots,n}(\underline{x}_i^T\underline{x}_i)\right)\underline{\alpha} - 2N\lambda\underline{\alpha} = 0 \tag{B1.11}$$

$$\Rightarrow \qquad \frac{1}{\mathrm{N}}\left(\sum_{i=1,\ldots,n}(\underline{x}_i^T\underline{x}_i)\right)\underline{\alpha} - \lambda\underline{\alpha} = 0 \tag{B1.12}$$

If we define a *design matrix D* of dimensions nx10, whose rows are the $\underline{x}_i$, $i = 1,\dots,n$, we then have:

$$\sum_{i=1,\dots,n}(\underline{x}_i^T \underline{x}_i)$$

$$\frac{1}{\mathrm{N}}\left(\sum_{i=1,\dots,n}(\underline{x}_i^T \underline{x}_i)\right) = D^T D \equiv A \qquad \text{(B1.13)}$$

and (B1.12) can be expressed as an eigenproblem:

$$A\underline{a} = \lambda\underline{a} \qquad \text{(B1.14)}$$

Moreover, because of the way it is defined in B1.13, A is a real and symmetric, and because B1.5 and B1.6 are non-negative, it has 10 positive eigenvalues. The eigenvectors represent all possible solutions of (B1.5), i.e. they are sets of coefficients for the quadric that we are trying to fit. The corresponding eigenvalues are a measure of the error of fit, therefore the optimal solution is the eigenvector that corresponds to the smallest eigenvalue.

B2: Curvature of Implicit Surfaces using the Fundamental Forms

(Extracted from [Hart95])

Let: $F(x,y,z) = 0$ be an implicit quadric surface in R^3, and F_x, F_y, F_z its first order partial derivatives. Then the coefficients E, F, G of the *First Fundamental Form* are:

$$E = 1 + \frac{F_x^2}{F_z^2}, \quad F = \frac{F_x F_y}{F_z^2}, \quad \text{and } G = 1 + \frac{F_y^2}{F_z^2} \qquad \text{(B2.1)}$$

$$\text{hence: } EG - F^2 = \frac{F_x^2 + F_y^2 + F_z^2}{F_z^2} \qquad \text{(B2.2)}$$

And the magnitude of the gradient is defined as:

$$|\nabla F| = \sqrt{F_x^2 + F_y^2 + F_z^2} \qquad \text{(B2.3)}$$

Furthermore, let F_{xx}, F_{yy}, F_{zz}, $F_{xy} = F_{yx}$, $F_{yz} = F_{zy}$, $F_{xz} = F_{zx}$ be the second order partial derivatives. Then, the coefficients *L, M, N* of the *Second Fundamental Form* are:

$$L = \frac{1}{F_z^2 |\nabla F|} \begin{vmatrix} F_{xx} & F_{xz} & F_x \\ F_{zx} & F_{zz} & F_z \\ F_x & F_z & 0 \end{vmatrix} \quad , \qquad \text{(B2.4a)}$$

$$M = \frac{1}{F_z^2 |\nabla F|} \begin{vmatrix} F_{xy} & F_{yz} & F_y \\ F_{zx} & F_{zz} & F_z \\ F_x & F_z & 0 \end{vmatrix} \quad , \text{ and} \qquad \text{(B2.4b)}$$

$$N = \frac{1}{F_z^2 |\nabla F|} \begin{vmatrix} F_{yy} & F_{yz} & F_y \\ F_{zy} & F_{zz} & F_z \\ F_y & F_z & 0 \end{vmatrix} \quad . \qquad \text{(B2.4c)}$$

According to [Hart95], the choice of F_z for the denominator does not sacrifice generality: it is possible to choose any partial derivative, and the definitions given by (B2.1) and (B2.4) can be appropriately re-written so that we still obtain the same values for the coefficients of the fundamental forms. This is correct in theory – the problems arising when trying to apply these equations in practice are discussed in section 5.1.1.

B3: Curvature of Implicit Surfaces using Minimization

We are using the general implicit second order (quadric) form of a surface, as given in equation (B1.1). It may also be conveniently written in matrix form as:

$$\begin{bmatrix} x & y & z & 1 \end{bmatrix} \begin{bmatrix} a & e & g & l \\ e & b & f & m \\ g & f & c & n \\ l & m & n & h \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = 0 \quad . \qquad \text{(B3.1)}$$

In partitioned form this becomes:

$$F(\underline{x}) = \begin{bmatrix} \underline{x}^T & 1 \end{bmatrix} \begin{bmatrix} A & \underline{l} \\ \underline{l}^T & h \end{bmatrix} \begin{bmatrix} \underline{x} \\ 1 \end{bmatrix} = 0 \quad , \qquad \text{(B3.2)}$$

where $\underline{l}$ is the vector $(l,m,n)^T$ and A the matrix of the coefficients a, b, c, e, f, g. This form represents any such surface in the space. By taking the partial derivatives and forming the gradient we can obtain the normal vector to the surface:

$$\nabla F = 2\begin{bmatrix} ax+ey+gz+l \\ ex+by+fz+m \\ gx+fy+cz+n \end{bmatrix} \equiv 2\begin{bmatrix} p \\ q \\ r \end{bmatrix} \equiv 2\underline{p} \quad , \qquad \text{(B3.3)}$$

where

$$\underline{p} = \underline{p}(\underline{x}) = \begin{bmatrix} A & \underline{l} \end{bmatrix}\begin{bmatrix} \underline{x} \\ 1 \end{bmatrix} \quad , \qquad \text{(B3.4)}$$

again in the notation used previously.

To obtain formulae for the principal curvatures, and incidentally also for the directions of principal curvatures, we consider a line in direction $\underline{u}$ tangential to the surface F at point $\underline{x}_0$. Then, by assumption we have the equations

$$F(\underline{x}_0) = 0 \quad , \qquad \text{(B3.5)}$$

and $$p(\underline{x})^T\underline{u} = 0 \quad . \qquad \text{(B3.6)}$$

Now any point $\underline{x}$ on the line $\underline{u}$ may be represented as:

$$\underline{x} = \underline{x}_0 + \xi\underline{u} \qquad \text{(B3.7)}$$

where if $$\underline{u}^2 = 1 \quad , \qquad \text{(B3.8)}$$

ξ is the distance along the line from $\underline{x}_0$.

In the plane that is formed by $\underline{u}$ and $\underline{p}_0$, we define the unit normal

$$\underline{n}_0 \equiv \frac{\underline{p}_0}{|\underline{p}_0|} \quad . \qquad \text{(B3.9)}$$

Note in passing that $\underline{n} = \dfrac{\underline{p}}{|\underline{p}|}$, and let:

$$\underline{x} = \underline{x}_0 + \xi\underline{u} + \eta\underline{n}_0 \quad . \qquad \text{(B3.10)}$$

Thus, on the quadric;

$$F(\underline{x}) = 0 \Rightarrow F(\underline{x}(\xi,\eta)) = 0 \quad . \qquad \text{(B3.11)}$$

This defines a planar curve, whose curvature may be computed in the usual way with (ξ,η) treated as (x,y) co-ordinates, i.e.

$$\kappa = \frac{\dfrac{d^2\eta}{d^2\xi}}{\left[1+\left(\dfrac{d\eta}{d\xi}\right)^2\right]^{\frac{3}{2}}} \quad . \qquad \text{(B3.12)}$$

To evaluate the derivatives of η with respect to ξ, we note then that, on the surface (B3.11):

$$dF = 0 = \nabla F \frac{\partial \underline{x}}{\partial \xi} d\xi + \nabla \frac{\partial \underline{x}}{\partial \eta} d\eta \qquad \text{(B3.13)}$$

$$\Rightarrow \quad dF = 0 = 2\left[\underline{u}^T \quad \underline{p}\right] d\xi + 2\left[\underline{n}_0^T \quad \underline{p}\right] d\eta \qquad \text{(B3.14)}$$

But, from (B3.3), $\nabla F = 2\underline{p}$. (B3.15)

We thus see that:

$$\frac{d\eta}{d\xi} = -\frac{\underline{u}^T \underline{p}}{\underline{n}_o^T \underline{p}} \quad . \qquad \text{(B3.16)}$$

It then follows that, since only $\underline{p}$ depends on $\underline{x}$, and hence on ξ,η that:

$$\frac{d^2\eta}{d\xi^2} = -\left[\frac{\left(\underline{n}_0^T \underline{p}\right)\left(\underline{u}^T \dfrac{\partial \underline{p}}{\partial \xi}\right) - \left(\underline{u}^T \underline{p}\right)\left(\underline{n}_0^T \dfrac{\partial \underline{p}}{\partial \xi}\right)}{\left(\underline{n}_0^T \underline{p}\right)^2}\right] \quad . \qquad \text{(B3.17)}$$

From (B3.4), according to which $\underline{p} = A\underline{x} + \underline{l} = \frac{1}{2}\nabla F$, we obtain

$$\frac{\partial \underline{p}}{\partial \xi} = A\frac{\partial \underline{x}}{\partial \xi} = A\underline{u} \quad . \qquad \text{(B3.18)}$$

We therefore see that:

$$\frac{d^2\eta}{d\xi^2} = -\left[\frac{(\underline{n}_0^T \underline{p})(\underline{u}^T A\underline{u}) - (\underline{u}^T \underline{p})(\underline{n}_0^T A\underline{u})}{(\underline{n}_0^T \underline{p})^2}\right] \qquad \text{(B3.19)}$$

and, that at the point $\underline{x}_0$ when $\xi = \eta = 0$,

$$\left.\frac{d^2\eta}{d\xi^2}\right|_{\underline{x}_0} = -\frac{(\underline{u}^T A\underline{u})}{\underline{n}_0^T \underline{p}_0} = -\frac{\underline{u}^T A\underline{u}}{|\underline{p}_0|} \qquad \text{(B3.20)}$$

since $\underline{u}^T\underline{p}_0 = \underline{u}^T\underline{n}_0 = 0$. The curvature is thus given by:

$$\left.\frac{d^2\eta}{d\xi^2}\right|_{\underline{x}_0} = -\frac{\underline{u}^T A\underline{u}}{|\underline{p}_0|} \quad , \qquad \text{(B3.21)}$$

$$\Rightarrow \quad \kappa(\underline{x}_0) = -\frac{\underline{u}^T A\underline{u}}{|\underline{p}_0|} \quad . \qquad \text{(B3.22)}$$

This is the curvature $\kappa(\underline{x}_0)$ in any direction $\underline{u}$.

Since, $F(\underline{x}(\xi,\eta))$ is a planar curve on the surface $F(\underline{x}) = 0$ then $\kappa(\underline{x}_0)$ is a surface curvature at point $\underline{x}_0$, measured in direction $\underline{u}$.

The principal curvatures are defined as the maximum and minimum values of the surface curvatures $\kappa(\underline{x}_0)$ at each point $\underline{x}_0$ over the directions $\underline{u}$ in which the curvature is measured. Thus, we need to maximize and minimize $\kappa(\underline{x}_0)$ by varying the direction $\underline{u}$ of the line $\underline{x} = \underline{x}_0 + \xi\underline{u}$.

Since $|\underline{p}_0|$ is a constant, we thus optimise:

$$\underset{\underline{u}}{ext}\{\underline{u}^T A\underline{u}\} \qquad \text{(B3.23)}$$

subject to the constraints

$$\underline{p}_0^T\underline{u} = \underline{p}^T(\underline{x}_0)\underline{u} = 0 \qquad \text{(B3.24)}$$

and $\underline{u}^2 = 1$ (B3.25)

by the method of Lagrange multipliers. Thus, we finally need to solve:

$$\underset{\underline{u}}{ext}\left\{\underline{u}^T A\underline{u} + 2\mu\underline{p}_0^T\underline{u} + \lambda(1-\underline{u}^2)\right\} \quad , \qquad \text{(B3.26)}$$

where $\underline{u}$ is a three vector $[u\ v\ w]^T$. Now, if we let

$$U = \underline{u}^T A\underline{u} + 2\mu\underline{p}_0^T\underline{u} + \lambda(1-\underline{u}^2) \qquad \text{(B3.27)}$$

then, since

$$\frac{\partial U}{\partial u} = 2A\underline{u} + 2\mu\underline{p}_0 - 2\lambda\underline{u} = 0 \quad , \qquad \text{(B3.28)}$$

we need to solve:

$$(A-\lambda)\underline{u} + \mu\underline{p}_0 = 0 \quad , \qquad \text{(B3.29)}$$

with $\underline{p}_0^T\underline{u} = 0$, (B3.30)

and $\underline{u}^2 = 1$. (B3.31)

Finally, we note that (B3.29) and (B3.30) may be written in a matrix form:

$$\begin{bmatrix} A-\lambda & \underline{p}_0 \\ \underline{p}_0^T & 0 \end{bmatrix}\begin{bmatrix} u \\ \mu \end{bmatrix} = 0 \qquad \text{(B3.32)}$$

and for a non-trivial solution we thus see that, λ are the roots of

$$\begin{vmatrix} A-\lambda & \underline{p}_0 \\ \underline{p}_0^T & 0 \end{vmatrix} = 0 \quad . \qquad \text{(B3.33)}$$

If we expand (B3.33) we obtain a second order polynomial with respect to λ. Thus, there will be two solutions for λ and from those we may calculate two solutions for $\underline{u}$. In spite of the somewhat unconventional form of (B3.33), it is easy to prove, by adapting slightly the methods used for the eigensolutions of a real symmetric matrix, that $\lambda(1)$ and $\lambda(2)$ are real, and that in general $\underline{u}(1)$ and $\underline{u}(2)$ are orthogonal. (See Appendix B of [Chal02] for details).

The principal curvatures $\kappa(1)$ and $\kappa(2)$ in principal directions $\underline{u}(1)$ and $\underline{u}(2)$ respectively are thus easily obtained from the eigensolutions of (B3.32). In particular, we see from

(B3.22) that the principal curvatures are given by:

$$\kappa(1) = \frac{-\lambda(1)}{|\underline{p}_0|} \text{ and } \kappa(2) = \frac{-\lambda(2)}{|\underline{p}_0|} \qquad \text{(B3.34)}$$

where $\underline{p}_0 = A\underline{x}_0 + \underline{l}$. (B3.35)

B4: Solution of the Non-linear Eigenproblem for Prinicipal Curvatures

We know from section 5.1.2 that the directions of principal curvature can be found by minimizing the normal curvature. This reduces to the eigenproblem $D = 0$ whereby:

$$D = \begin{vmatrix} F'' - \lambda I & F' \\ {}^{t}F' & 0 \end{vmatrix} = 0 \qquad \text{with respect to } \lambda. \qquad \text{(B4.1a)}$$

where:

$$F' = \begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} \qquad \text{and} \qquad F'' = \begin{bmatrix} F_{xx} & F_{xy} & F_{xz} \\ F_{yx} & F_{yx} & F_{yz} \\ F_{zx} & F_{zx} & F_{zz} \end{bmatrix} \qquad , \qquad \text{(B4.1b)}$$

therefore (B4.1) becomes:

$$D = \begin{vmatrix} F_{xx} - \lambda & F_{xy} & F_{xz} & F_x \\ F_{xy} & F_{yy} - \lambda & F_{yz} & F_y \\ F_{xz} & F_{yz} & F_{zz} - \lambda & F_z \\ F_x & F_y & F_z & 0 \end{vmatrix} = 0 \qquad \text{(B4.2)}$$

Note here that:

1. $F_{xy} = F_{yx}$, $F_{xz} = F_{zx}$, $F_{yz} = F_{zy}$, and

2. (B4.1) is equivalent to (B3.33). In fact, (B3.33) is the special case of (B4.1) if F is a quadric. However, (B4.1) can be used for any implicit surface, as is also

indicated in [BeGo88].

Expanding (B4.2) results in a quadratic equation of the form:

$$A\lambda^2 + B\lambda + C = 0, \qquad \text{where:} \tag{B4.3}$$

$$A = -\left(F_x^2 + F_y^2 + F_z^2\right) \tag{B4.3a}$$

$$B = F_x^2\left(F_{yy} + F_{zz}\right) + F_y^2\left(F_{xx} + F_{zz}\right) + F_z^2\left(F_{xx} + F_{yy}\right) + 2\left(K_x + K_y + K_z\right) \tag{B4.3b}$$

$$C = -\left(F_x^2 F_{yy} F_{zz} + F_y^2 F_{xx} F_{zz} + F_z^2 F_{xx} F_{yy}\right) + 2\left(K_x F_{xx} + K_y F_{yy} + K_z F_{zz}\right) + N_{xyz} \tag{B4.3c}$$

with the shorthands defined as follows:

$$\begin{aligned} K_x &= F_y F_z F_{yz} & L_x &= F_y F_z F_{xy} F_{xz} & M_x &= F_x^2 F_{yz}^2 \\ K_y &= F_x F_z F_{xz} & L_y &= F_x F_z F_{xy} F_{yz} & M_y &= F_y^2 F_{xz}^2 \\ K_z &= F_x F_y F_{xy} & L_z &= F_x F_y F_{xz} F_{yz} & M_z &= F_z^2 F_{xy}^2 \end{aligned}$$

and $N_{xyz} = -2\left(L_x + L_y + L_z\right) + \left(M_x + M_y + M_z\right)$

The second degree polynomial equation (B4.3) can be solved using standard methods (i.e. via the discriminant), and returns a maximum of two real[15] solutions λ_i, $i = 1,2$. The *values* of the principal curvatures can therefore be obtained by:

$$k_i = \frac{\lambda_i}{|F'|}, i = 1,2. \tag{B4.4}$$

whereas the corresponding *directions* of the principal curvatures are the eigenvectors of (1), and can be obtained by backsubstituting and solving for v_i the (linear) system:

$$\left(F'' - \lambda_i I\right) \cdot v_i = F' \qquad \text{for } i = 1,2. \tag{B4.5}$$

Special case: if the implicit surface is expressed by an equation that contains only single-dimension terms (e.g. ellipsoid or tooth, rather than octahedron or bumpy

---

[15] It is not obvious from (B4.3) that the solutions are in principle always real. Reality is proven from the matrix eigenproblem (B3.32).

sphere), then all the second order derivatives F_{ij} reduce to zero unless $i = j$. In that case, the coefficients of (B4.3) become:

$$A = -\left(F_x^2 + F_y^2 + F_z^2\right) \tag{B4.6a}$$

$$B = F_x^2\left(F_{yy} + F_{zz}\right) + F_y^2\left(F_{xx} + F_{zz}\right) + F_z^2\left(F_{xx} + F_{yy}\right) \tag{B4.6b}$$

$$C = -\left(F_x^2 F_{yy} F_{zz} + F_y^2 F_{xx} F_{zz} + F_z^2 F_{xx} F_{yy}\right) \tag{B4.6c}$$

Moreover, the linear system (B4.5) now becomes tridiagonal and its solution is very easy and straightforward. The components of the two eigenvectors are given by:

$$v_{i_j} = \frac{F_j}{F_{jj} - \lambda_i}, \qquad \text{where } i = 1,2 \text{ and } j = \{x,y,z\} \tag{B4.7}$$

APPENDIX C : Implementation and Platform Issues

The implementations of the algorithms and methodologies discussed throughout this thesis have been implemented on a PC platform running Microsoft Windows NT 4.0.

In order to achieve consistency and coherence, as well as to adhere to good programming practices, the implementations were encapsulated in a single application under the working name of BodyAnalyser.

BodyAnalyser was developed exclusively in C++, and the development tool used was Microsoft Visual Studio, version 4.2. The library used for handling the 3D graphics and implementing a rapid prototype of a 3D viewer with the appropriate interface facilities was OpenInventor, version 2.2.1. Throughout the development, the maximum possible effort was made to keep the core processing modules separated from the user interface code – theoretically they could be reused on any platform, although this has not been tested at the time of writing since it has been well beyond the scope of the research.

Implementations of a number of functions handling standard matrix operations (such as inversion, multiplication and Gaussian eliminations) were kindly contributed by Mr. Timothy Hutton and were only subject to minor alterations in order to achieve compatibility with proprietary code.

Perhaps the greatest programming difficulty encountered during the research was the implementation of a fast and reliable (i.e. computationally stable) eigensolution module. The one originally employed was taken from the NAG (Numerical Algorithms Group) library, which is known to be one of the best verified implementations around. However, the version of NAG available at the time was developed on an out-of-date platform (using Fortran code compiled into exported DLLs), and its incorporation into our C++ code soon proved to make it impractical because we had to resort to unorthodox programming tricks in order to access the binary code without causing clashes at the linking stage. The side-effect of those tricks was an unresolvable memory leak that could only be circumvented by implementing a lot of disk I/O operations, which in turn reduced the performance of the implementation to the point were it became impractical to use it on serious data (i.e. high resolution whole body scans –

which could take up to 40 mins each to process). As the computation became progressively more complex, these performance levels were prohibiting repetitive experimentation and debugging of the code, so unfortunately the use of the NAG modules had to be discontinued. However, since all the (linear) eigenproblems involved in this research involved real symmetric matrices, another reputable, albeit specialised implementation was found in [PFTV92] ("Numerical Recipes in C"). This was eventually used and proved to produce effectively the same numerical results. Because it was incorporated into our implementation at source code level, its seamless integration with the rest of the code has been possible without the need to employ any unorthodox methods.

In addition to the above, Matlab R12 was occassionaly used for the rapid prototyping and testing of some mathematical methods, prior to their implementation in C++.

BodyAnalyser was programmed to export intermediate numerical results into plain text form. Wherever these results were tabulated (i.e. in most cases), the text files were organized using the comma-separated variable (CSV) convention and given the *.csv extension. They could then be opened and processed further using Microsoft Excel, which has been the tool of choice for most of the evaluation work appearing throughout this thesis, as well as for the generation of the relevant tables, statistics, diagrams and charts.