3D OBJECT RECOGNITION USING SCALE SPACE OF CURVATURES

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

ERDEM AKAGÜNDÜZ

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
DOCTOR OF PHILOSOPHY
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

JANUARY 2011

Approval of thesis:

3D OBJECT RECOGNITION USING SCALE SPACE OF CURVATURES

submitted by **ERDEM AKAGÜNDÜZ** in partial fulfillment of the requirements for the degree of **Doctor of Philosophy in Electrical and Electronics Engineering Department, Middle East Technical University** by,

Prof. Dr. Canan ÖZGEN ___________
Dean, Graduate School of **Natural and Applied Sciences**

Prof. Dr. İsmet ERKMEN ___________
Head of Department, **Electrical and Electronics Engineering**

Asst. Prof. Dr. İlkay ULUSOY
Supervisor, **Electrical and Electronics Engineering Dept., METU** ___________

Examining Committee Members:

Prof. Dr. Volkan ATALAY
Computer Engineering Dept., METU ____________________

Asst. Prof. Dr. İlkay ULUSOY
Electrical and Electronics Engineering Dept., METU ____________________

Prof. Dr. Uğur HALICI
Electrical and Electronics Engineering Dept., METU ____________________

Prof. Dr. A. Aydın ALATAN
Electrical and Electronics Engineering Dept., METU ____________________

Asst. Prof. Dr Pınar DUYGULU ŞAHİN
Computer Engineering Dept., Bilkent Uni. ____________________

Date: 06.January.2011

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname : Erdem AKAGÜNDÜZ

Signature :

ABSTRACT

3D OBJECT RECOGNITION USING SCALE SPACE OF CURVATURES

Erdem AKAGÜNDÜZ

Ph.D., Department of Electrical and Electronics Engineering

Supervisor: Asst. Dr. İlkay ULUSOY

January 2011, 136 pages

In this thesis, a generic, scale and resolution invariant method to extract 3D features from 3D surfaces, is proposed. Features are extracted with their scale (metric size and resolution) from range images using scale-space of 3D surface curvatures. Different from previous scale-space approaches; connected components within the classified curvature scale-space are extracted as features. Furthermore, scales of features are extracted invariant of the metric size or the sampling of the range images. Geometric hashing is used for object recognition where scaled, occluded and both scaled and occluded versions of range images from a 3D object database are tested. The experimental results under varying scale and occlusion are compared with SIFT in terms of recognition capabilities. In addition, to emphasize the importance of using scale space of curvatures, the comparative recognition results obtained with single scale features are also presented.

Keywords: 3D Object Recognition, 3D Feature Extraction, 3D Pattern Recognition

ÖZ

EĞRİLİK ÖLÇEK UZAYI KULLANARAK 3B NESNE TANIMA

Erdem AKAGÜNDÜZ
Doktora, Elektrik Elektronik Mühendisliği Bölümü
Tez Yöneticisi: Yrd. Dr. İlkay ULUSOY

Ocak 2011, 136 sayfa

Bu tezde, 3B yüzeylerden ölçek ve çözünürlükten bağımsız bir 3B öznitelik çıkarım yöntemi önerilmiştir. Öznitelikler, 3B yüzey eğrilikleri kullanılarak ölçek bilgileri (metrik boy ve çözünürlük) kaybedilmeden çıkarılmaktadır. Önceki ölçek uzayı yaklaşımlarından farklı olarak, sınıflandırılmış ölçek eğrilik uzayındaki bileşik elemanlar öznitelik olarak kullanılmıştır. Bu öznitelikler metrik uzunluk ve örneklemeden bağımsız çıkarılmaktadırlar. Geometrik kıyım yöntemi, nesne tanıma amacıyla kullanılmış ve farklı ölçeklerde, engelli nesneler içeren 3B yüzey veri tabanlarıyla test edilmiştir. Deney sonuçları nesne tanıma kabiliyetleri bakımından SIFT ile karşılaştırılmıştır. Ölçek uzayı kullanmanın önemini göstermek için yöntem hem ölçek uzayı kullanılarak hem de kullanılmadan çalıştırılıp, sonuçlar karşılaştırılmıştır.

Anahtar Kelimeler: 3B Nesne Tanıma, 3B Öznitelik Çıkarımı, 3B Örüntü Tanıma

ACKNOWLEDGEMENTS

The author wishes to express his gratitude to his supervisor Asst. Dr. İlkay Ulusoy for her guidance, advice, criticism, encouragements and insight throughout the research.

The author would also like to thank his parents, Nurhayat and Feyzi Akagündüz and his brother Haldun Akagündüz for their endless support.

And finally for Sevil, “Like everything in my life, this thesis would never be complete without you.”

TABLE OF CONTENTS

ABSTRACT iv
ÖZ v
ACKNOWLEDGEMENTS vi
TABLE OF CONTENTS vii

CHAPTER

1. INTRODUCTION 1
    1.2 Problem Definition 2
    1.2 Motivation 3
    1.3 Contribution 4
    1.4 Outline of the thesis 4
2. RELATED WORK 7
    2.1 3D Feature Extraction 7
        2.1.1 3D Surface Curvatures 8
        2.1.2 3D Scale Invariant Features 8
        2.1.3 3D Descriptors 9
    2.2 3D Object Representation 10
        2.2.1 3D Object Recognition 11
    2.2 3D Facial Detection 11
    2.4 3D Object Registration 12
3. PROCESSING 3D INFORMATION 13
    3.1 3D Information Formats 13
        3.1.1 Range Images 13
        3.1.2 Polygonal Data 14
        3.1.3 Parametric Representations 15
        3.1.4 Digital Elevation Models 16
    3.2 Common Defects of 3D Scanner Outputs 17

3.2.1 False Holes (False Invalid Points) 17
3.2.2 Exploded/Imploded Regions 18
3.2.3 Noise 19
3.3 Processing 3D range scans 20
3.3.1 Processing False Holes 20
3.3.2 Processing Exploded/Imploded Regions 22
3.3.3 Processing Noise 22
4. SURFACE CURVATURES 24
4.1 Reliable Curvature Estimation 24
4.2 Principal Curvatures 26
4.3 Mean (H) and Gaussian (K) Curvatures 27
4.4 Shape Index (S) and Curvedness (C) 30
4.5 Transform Invariance of Curvatures 32
4.5.1 Effect of Scale and Sampling on Curvature – Scale/Resolution Ratio . 33
4.5.2 Rule for Thresholding 36
4.6 Comparison of HK and SC Curvature Spaces 36
4.6.1 Mathematical Comparison of HK and SC Curvature Spaces 37
4.6.2 HK&SC Coupled Curvature Space 39
5. SCALE-SPACE OF CURVATURES 41
5.1 Scale-Space Theory in Computer Vision 41
5.2 Gaussian Pyramiding 42
5.2.1 Gaussian Pyramiding over 3D Surface Scans 43
5.2.2 Invalid Points in 3D Surface Pyramids 44
5.3 Scale Space of Curvatures 45
5.3.1 Curvatures in Scale-Space 46
6. FEATURE EXTRACTION AND OBJECT REPRESENTATION 50
6.1 Construction of the UVS Volume 50
6.2 Extraction of Features 53
6.3 Scale-Space Localization of the Features 55
6.4 Transform Invariant Topology Construction 58
6.5 Importance of Scale Space Search 63
6.6 Feature Extraction Robustness under Noise 66

6.7 Local Region Descriptions 68
  6.7.1 Depth Histograms 68
  6.7.2 Normal Histograms 70
  6.7.3 Curvature Histograms 72
  6.7.4 Spin Images 74
7. EXPERIMENTAL WORK 76
7.1 Face Detection and Pose Estimation 76
  7.1.1 Transform Invariant Four-Node Face Topology 77
  7.1.2 Face Detection on Bosporus Database 78
  7.1.3 Experimental Work on 3D Face Detection 82
  7.1.4 3D Facial Pose Estimation 92
  7.1.5 Conclusion on 3D Facial Detection and Pose Estimation 94
7.2 3D Object Recognition 96
  7.2.1 Transform Invariant n-Node Topology 96
  7.2.2 Learning the n-node Topology – Geometric Hashing 98
  7.2.3 HK-SC Curvature Space Classification Comparison and Threshold Fine Tuning 100
  7.2.4 3D Object Recognition Tests 105
  7.2.5 3D Object Registration Experiments 109
  7.2.6 Conclusions on 3D Object Recognition and Registration 114
7.3 Delineation of Slope Units Based On Scale And Resolution Invariant 3D Curvature Extraction 115
  7.3.1 Slope Unit Generation by Conventional Method 116
  7.3.2 The Proposed Method 117
  7.3.3 Results and Comparison 120
8. CONCLUSIONS AND FUTURE DIRECTIONS 122
8.1 Conclusions 122
8.2 Future Directions 125
REFERENCES 126
CURRICULUM VITAE 132

LIST OF TABLES

TABLES

Table 1. Shape Classification in HK curvature space. Corresponding examples of surface regions are seen in Figure 13 28

Table 2. Shape Index Classification. Colours are consistent with the previous figures.. 30

Table 3. Shape Index and Curvedness Classification. Colours are consistent with the previous figures. 31

Table 4. The definition of the regions seen in Figure 19 38

Table 5. HK&SC coupled space Classification. Colours are consistent with the previous figures 40

Table 6. Extracted features from the surfaces given in Figure 38 with their locations in UVS space 58

Table 7. The radius values (in mm) of the extracted features in Figure 41 65

Table 8. Mean of the differences between the marked and calculated features for the nose tips and inner eye pits in the training set 80

Table 9. Tested poses and number of facial points included for each pose 82

Table 10. Detection percentage of the nose and the inner eye pits for various poses are listed for all methods. The first column from left (ours) in each main column is the proposed method. Then M1, M2, M3 and M4 are given in the columns from left to right respectively. Best results for each pose are underlined and bolded. 89

Table 11. Detection errors are compared for all methods. The mean and the standard deviation of the error (the absolute distance between the marked points and the calculated points) are given. The values are in millimetres. The mean is written over the standard deviation. 90

Table 12. C(U,N) number of features obtained from k largest features, using N-node topology. For the sake of simplicity the last node is chosen as the base node. 98
Table 13. Average Recognition Rates for All Methods in All Experiments 108
Table 14. Similar and different slope unit areas. 121

LIST OF FIGURES

FIGURES

Figure 1. Sample range image: “bunny” from [Stuttgart Database] 14
Figure 2. The polygonal form of the object in Figure 1 is obtained by combining the valid vertices obtained from different range scans of the object. 15
Figure 3. a) The DEM image is depicted. b) The 3D render of the DEM image is shown. 16
Figure 4. A facial scan taken from FRGC v1.a 3D database is shown. Invalid regions around eyebrows are marked as invalid in the depth image. 18
Figure 5. A spike on the iris of a facial scan from FRGC 3D facial database. 18
Figure 6. Nose is imploded in the 3D range scan (left) because of extreme illumination on the nose tip (FRGC v1.a). 19
Figure 7. a) 3D artificial surface without noise b) Normal directions over the noise-free surface. c) 3D artificial surface with noise. d) Normal directions over the noisy surface [Bozkurt et. al. 2009]. 19
Figure 8. The “hub” object from the Stuttgart 3D range image database is shown. There is no defect on the range scan; however the object possesses a number of actual holes. 20
Figure 9. a) Texture for the 3D Scan. b) Black areas denote the background. Red areas denote the invalid points selected to be extrapolated. Blue areas denote the neighbouring valid points used for extrapolation. White areas denote valid points. c) Processed point cloud. 21
Figure 10. Surfaces de-noised with Gaussian (left) and Bilateral (right) filters [Bozkurt 2008]. 23
Figure 11. The result of the bilateral filtering on a FRGC sample. 23
Figure 12. a) The normal plane with the maximum curvature is seen. b) The normal plane with the minimum curvature is seen c) $\kappa_1 = 1.56$ and $\kappa_2 = -2.37$. The surface is a patch from a monkey saddle: $z(x,y) = x^3 - 3 \cdot x \cdot y^2$.... 26

Figure 13. Shape types in correspondence with Table 1. Shapes at rightmost column are hyperbolic and from top to bottom they have the properties: $\kappa_1 + \kappa_2 < 0$, $\kappa_1 + \kappa_2 = 0$ and $\kappa_1 + \kappa_2 > 0$. The centre column show parabolic or flat-umbilic (planar) shapes where one of the principal curvatures is equal to zero ($\kappa_1 = 0 \cup \kappa_2 = 0$). The leftmost column show the elliptic shapes where both principal curvatures have the same sign. The upper is convex elliptic (a peak: $\kappa_1 < 0 \cap \kappa_2 < 0$) and the lower one is concave elliptic (a pit: $\kappa_1 > 0 \cap \kappa_2 > 0$). 28

Figure 14. The HK classification on the (κ_1, κ_2) plane. Region colours (and the numbers) correspond to Table 1. The separating lines are the zero-thresholds for H values and their equations are: $(\kappa_1 + \kappa_2) / 2 = +H_{zero}$ and $(\kappa_1 + \kappa_2) / 2 = -H_{zero}$. The separating curves are the zero-threshold for K values with the equation: $(\kappa_1 \cdot \kappa_2) > +K_{zero}$ and $(\kappa_1 \cdot \kappa_2) > -K_{zero}$. 29

Figure 15. The SC classification on the (κ_1, κ_2) plane. Region colours correspond to Table 3. The separating lines are the constant S values: S = ±5/8, ±3/8 and ±3/16. The circle around the origin is the zero-threshold for C values and its equation is: $\sqrt{\left(\kappa_1^2 + \kappa_2^2\right)/2} = C_{zero}$. 31

Figure 16. a) Two scaled version of the same shape. Bigger one (1) is 10 times larger than the other. (Smaller one (2) is barely seen on the lower–left of the bigger one). b) Two S maps are identical (in both S maps a grey scale level is mapped to a distinct real value. So having same intensities on each SI map proves that they have the same real S values) c) C (curvedness) map intensities (i.e. the C values) are different. For the small shape, the curvedness is barely detected. For H and K maps, the effect would be similar to C. 33

Figure 17. a) 25x25 sampled surface. b) 50x50 sampled surface. c) On the left S map for the 25x25 sampled surface is seen. In the middle S map for 50x50 sampled surface is seen. On the right we have the S map for the original 100x100 sampled surface. As the surface resolution gets lower and lower, the S map resolution also changes but the S values (the grey level intensities) do not get affected. 34

Figure 18. a) The original surface (1) with 100x100 resolution; Surface 2 with 50x50 resolution scaled by ½; Surface 3 with 25x25 resolution scaled by ¼; b) H maps for each surface is depicted. Intensities (e.g. real H values) do not change for different scaled H maps, since scale/resolution ratio is constant. 35

Figure 19. The difference in region description of HK and SC spaces over the (κ_1, κ_2) plane is depicted. Each number designates a region which is described in Table 4. The shape index thresholds are ±5/8, ±3/8, ±3/16, as it is for Figure 4. The other thresholds are selected such that $K_{zero} = H^2_{zero}$ and $C_{zero} = H_{zero}$. 37

Figure 20. The HK&SC coupled space classification in (κ_1, κ_2) plane. The colours and numbers are correspondent with Table 5. 40

Figure 21. a) Gaussian pyramiding over a 1D signal, (higher scale levels towards the bottom). b) Gaussian Pyramiding on a 2D image. In each step, the image is both down-sampled and smoothed (higher scale levels towards the right) [Burt and Adelson 1983]. 42

Figure 22. Gaussian Pyramiding on 3D scans is seen. In each scale level, the sampling rate is halved, but the metric distances are preserved. 43

Figure 23. The scale levels of a depth image constructed by Gaussian pyramiding are seen. The black regions are the invalid points. Around the valid point boundaries the smoothing filter avoids blending with invalid points so that the sharp boundaries are preserved. 44

Figure 24. The smaller elements vanish as we go up the scale axis. 45

Figure 25. The scale-space of the surface normals. Red values correspond to x components, green values correspond to y components and blue components correspond to z components of the surface normals. 46

Figure 26. The scale-space of mean curvature values. The concave regions which have positive mean curvature values, are painted in red, where convex regions with negative mean curvature values are painted in blue. 47

Figure 27. The scale-space of Gaussian curvature values. The parabolic regions with positive Gaussian curvature values are painted in red colour, where the hyperbolic regions with negative Gaussian curvature values are painted in blue colour. 47

Figure 28. The scale-space of shape index values. The shape index values and colour is mapped according to Table 2 in Chapter 3. 48

Figure 29. The scale-space of curvedness values. The gray level intensities designate the curvedness values, where zero curvedness value corresponds to black. 49

Figure 30. Three levels from the HK pyramid of a face model surface. 51

Figure 31. The HK UVS volume after morphological operations printed above the surface patch. The features are indicated with their surface shapes as colours (blue for peaks, cyan for pits, red for saddle ridges, etc.). 51

Figure 32. UVS Volume Construction. 52

Figure 33. The weights are demonstrated in the HK UVS volume. The bigger the weight is the lighter the colour becomes. 53

Figure 34. Labelled layers of UVS space constructed by HK values where scale level is increasing from left to right. The original surface level is indicated by “S=0”. Labels are given by colours. (peak: blue, saddle ridge: red, convex cylinder: purple, pit: cyan, saddle valley: yellow, concave cylinder: green, hyperbolic: orange, plane: gray.) 54

Figure 35. Ten largest extracted features are shown as squares where the feature center (x_i) is given by the square center, the feature size (radius) (r_i) is given by the square size and the feature type is given by its colour. 55

Figure 36. Three unit volume Gaussian surfaces at different scales ($\sigma_A= \sigma_K$, $\sigma_B= 2*\sigma_K$, $\sigma_C= 4*\sigma_K$) and their respective UVS volumes. 56

Figure 37. Localization of the peak feature (a) using both methods (top: weighted average, bottom: single maximum). b) For the ideal surface both methods localize the peak feature correctly. c) For the noisy surface, the feature is still correctly localized using weighted averages (top) but the localization fails under noise when the single maximum method is used (bottom). 57

Figure 38. UVS volumes for a. rotated towards left, b. frontal and c. expressional (disgusted) faces. 58

Figure 39. The feature extraction algorithm is scale invariant since the extracted features numerically depend on the scaling ratio of a surface. The numbers are the radii of the extracted feature elements. 62

Figure 40. Ten largest features extracted using a) original image without scale-space search, b) scaled image (by 0.8) without scale-space search, c) original image with scale-space search, d) scaled image (by 0.8) with scale-space search. 64

Figure 41. Ten largest extracted features from four different object surfaces. For each object the original (middle), 1.2 times scaled (left) and 0.8 times scaled (right) versions are given. The feature types and sizes are given by colour and square size respectively (peak: blue, saddle ridge: red, convex cylinder: purple, pit: cyan, saddle valley: yellow, concave cylinder: green, hyperbolic: orange, plane: gray). 66

Figure 42. Ten largest extracted features from the noisy versions of the objects: screwdriver (top), duck (middle) and pig (bottom). a) %15 noise b) %30 noise c) %45 noise d) %60 noise. 67

Figure 43. a) Interest point center localization error in mm for different noise levels. b) Radius calculation error in mm for different noise levels. 67

Figure 44. a) The original range image taken from [Stuttgart Database]. b) %25 occluded version of the range image in (a). c) The depth histogram of the original range image (blue) and the %25 occluded image (red). 69

Figure 45. a) Nose is a feature element of peak type. Its radius is the radius of the blue circle around the feature. The local depth histogram of the region within the radius of the feature centre is depicted. 70

Figure 46. a) The original range image taken from [Stuttgart Database]. b) The 2D normal histogram of the original range image. c) %25 occluded version of the range image in (a). d) The 2D normal histogram of the %25 occluded image. 71

Figure 47. a) The nose is a feature element of peak type. Its radius is the radius of the blue circle around the feature. The local normal histogram of the region within the radius of the feature centre is depicted. 71

Figure 48. a) The original range image taken from [Stuttgart Database]. b) %25 occluded version of the range image in (a). c) The shape index histogram of the original range image (blue) and the shape index histogram of the %25 occluded image (red). 73

Figure 49. a) The nose is a feature element of peak type. Its radius is the radius of the blue circle around the feature. The local shape index histogram of the region within the radius of the feature centre is depicted. 74

Figure 50. The spin images of point p from different views are shown. The spin images are similar [Johnson and Hebert 1999]. 75

Figure 51. The training phase. 79

Figure 52. a) The marked nose tip. b) The calculated nose peak centre. c) The marked inner right eye pit. d) The calculated inner right eye pit centre. e) The marked inner left eye pit. f) The calculated inner left eye pit centre.. 80

Figure 53. The testing phase. 81

Figure 54. a) Texture for the 3D Scan. b) Black areas denote the background. Red areas denote the invalid points selected to be extrapolated. Blue areas denote the neighbouring valid points used for extrapolation. White areas denote valid points. c) Cleaned cropped and re-sampled point cloud. 84

Figure 55. A successful and a problematic case for the poses a) with glasses. b) Mouth open c) Hand on one eye d) Look right 75º e) Look right and down 45º. 92

Figure 56. Artificially created facial scans. Leftmost scan is the original scan from FRGC database. Other scans are created by applying the following transformations respectively from left to right: rotation around Y-axis by +10º,+20º,-10º,-20º, rotation around X-axis by +15º,-15º and rotation around Z-axis by +45º. 93

Figure 57. Virtually created facial scans and their estimated pose angles. 94

Figure 58. Demonstration of geometric hashing: All the topologies constructed from a range image are compared to the topology vectors in the database. The database model which receives the greatest number of votes is taken as the match of the test object. 99

Figure 59. Angle threshold tests for both methods. 103

Figure 60. Length threshold tests for both methods. 103
Figure 61. Type tests for both methods. 103
Figure 62. Feature number tests for both methods. 104
Figure 63. Database size tests for both methods. 104
Figure 64. Confusion matrices of the methods: SSFE, SSFE+spin, MSFE, MSFE+spin, SIFT, from left to right respectively. Original testing images are used. 107
Figure 65. Confusion matrices of the methods: SSFE, SSFE+spin, MSFE, MSFE+spin, SIFT, from left to right respectively. Scaled versions of the testing images are used. 107
Figure 66. Confusion matrices of the methods: SSFE, SSFE+spin, MSFE, MSFE+spin, SIFT, from left to right respectively. Occluded versions of the testing images are used. 108
Figure 67. Confusion matrices of the methods: SSFE, SSFE+spin, MSFE, MSFE+spin, SIFT, from left to right respectively. Scaled and occluded versions of the testing images are used. 108
Figure 68. Two artificial surfaces with a Gaussian peak and a Gaussian pit, which are 135º in-plane rotated (rotation around z-axis) of each other. 112
Figure 69. Matched features from the registered artificial range images. 112
Figure 70. Two range images of the screwdriver from Stuttgart database. There is rotation around y-axis. 113
Figure 71. Matched features from the registered range images from Stuttgart Database. 113
Figure 72. Matched features from the registered range images from Stuttgart Database. The surfaces are scaled versions of the same range image. 114
Figure 73. a) Watershed boundaries determined using the DEM. b) Drainage line overlaid with watershed of the region c) Watershed boundaries determined using the reverse DEM (shown with red lines) overlaid with watershed boundaries which are present on both the left and right side of a sub-basin. d) Slope unit of region obtained in 3D. 117
Figure 74. a) Original Surface Patch. b) 4th Scale Level of the surface patch c) Candidate ridge and drainage lines are shown in black. d) Ridge and

drainage lines after thinning. e) Final ridge and drainage lines after the elimination of outliers. f) Detected line tips and corners are shown in pink and an example interest region for a tip is shown by a triangle. g) Line tips connected to available corners and tips are shown in green. h) Segmented landslide regions are shown in different colours. 119

Figure 75. The overlay of conventional and proposed method outputs. Legend: black line: conventional slope unit, pink line: proposed method slope unit, yellow colour: similar regions, blue colour: generalized units of conventional method or detailed units of proposed method, green colour: detailed units of conventional method or generalized units of proposed method. 120

Figure 76. 3D view of slope units from proposed and conventional methods Legend: black line: conventional slope unit, pink line: proposed method slope unit. 121

CHAPTER 1

INTRODUCTION

Mankind’s curiosity on the nature and science has led him from the invention of the wheel to this very age of unmanned machines. Each and every day, numerous new clever systems are being introduced to our lives. They are able to see, comprehend and act according to our needs. I personally choose to call this an invasion. I strongly believe that this “intelligent” and technological invasion will linger until mankind is able to create his alike. In other words, until he is satisfied with playing God, if he will ever be…

On our part of playing God, we are interested in visual understanding, the most intimate ability, the tremendous divine gift we are given. Computer vision, the science of mimicking this ability, will be our main concern in this dissertation. Moreover, we will be specifically examining 3D object representation, the key human ability that enables him to understand and differentiate real world objects.

Although quite young for a scientific field, computer vision has surprisingly had fruitful fifty years of past. With the increasing availability of various digital imaging systems, first image processing then computer vision have become one of the most promising scientific fields in signal processing. Moreover, the field has advanced so fast that even the academicians sometimes find it difficult to catch up with the new advances. For instance face detection, once a very hot topic of the field, is now a simple application in a low-cost cellular phone.

Since it is highly a practical field, the achievements in imaging technology profoundly affect the direction of research. For instance, some problems like segmentation of human body movements from short distances, which was hot and difficult problem for 2D imagery, is now a simple and solved problem using time-of-flight cameras, and has several usages in game industry. This is a single example among several, which shows the increasing usage of 3D information in computer vision.

Processing 3D information, namely point 3D clouds of 3D surfaces, is not always simpler than the conventional 2D image processing, as in the human body segmentation example. 3D image processing and 3D object representation are relatively new problems and have various open points. However since they promise acquisition of the real world with depth information, (not fully practically) independent of illumination and with metric size information; they may offer easier solutions compared to their corresponding solutions in 2D.

3D information can be acquired using different devices such as, 3D range scanners, stereo systems, lidar scanners, time-of-flight cameras, satellite imagery, etc. The acquired signal is usually a grid or cloud of 3D points. The signal carries vast amount of information which is difficult to process in real-time. Thus, methods to transform this data into a sparse representation are usually needed and are common problems of the field. Furthermore some repeatability properties of signal processing are inherited for their 3D versions as well, such as transform invariance, scale invariance, and robustness to noise because finding repeatable salient points or regions over 3D surfaces (point clouds) is a basic prerequisite for 3D object detection, recognition and even matching similar objects, namely registration.

In this thesis, we aim at constructing a representation method for 3D object surfaces. We propose a feature extraction method which is scale, sampling and orientation invariant. In this chapter, we commence by the problem definition, our motivation and contribution.

1.2 Problem Definition

Finding repeatable, robust and invariant salient points and/or regions, is one of the very important problems of visual pattern recognition. These salient points and/or regions, usually called features, can be used to represent an object or match similar objects. Depending on the signal type, the feature extraction method may change. In order to evaluate a feature extraction method, some basic properties may be defined. [Tuytelaars and Mikolajczyk 2007] propose a number of basic properties feature detectors. These are repeatability (invariance and robustness), distinctiveness, locality, quantity, accuracy and efficiency. Most of these are competing properties, such that it is impossible to satisfy all of them simultaneously. Usually, a method is designed or selected according to the application needs and signal type.

Scale and sampling invariance of a feature detector implies that the detector is capable of providing the same salient point or region for the scaled and re-sampled versions of a signal. Extracting feature sizes (radii or effective regions) covariant to scaling is another ability strongly related to scale invariance, which implies that the extracted size of the feature is covariant with the size or sampling of the signal. In other words, for a rescaled version of a signal, the same features are detected and for each detected feature, their sizes are scaled covariant to the scaling ratio.

Representing the meaningful portion of information within a signal is crucial. For this reason, representing an object within a 3D point cloud is an important and difficult problem. The basic needs of a representation are being sparse, repeatability and distinctiveness.

In this thesis, we deal with the above problems of 3D point cloud processing. In particular, we focus on obtaining higher levels of scale and sampling invariance in feature extraction. Furthermore we struggle to construct a sparse yet repeatable 3D object representation. We particularly chose these problems since they are still open problems of 3D computer vision.

1.2 Motivation

In the previous decade, some important approaches on feature extraction from 3D point clouds and 3D object representation can be found in the literature. The related studies are given in the next chapter. Two important open points in the literature can be summarized as feature extraction with full scale invariance and a generalized object representation scheme that can be used by different applications.

Little work on the limits of scale invariance on 3D feature extraction is reported. More importantly, for the reported scale invariant 3D feature extractors, none to our knowledge deal with the problem of covariant feature size extraction on scale varying databases. Therefore we analyze the existing scale space approaches and seek for a way to achieve higher scale invariance, by manipulating the construction of the signal scale-space. By this way, we also look for a method to achieve signals with their effective region size.

Second, a generic transform invariant topological 3D surface representation standard is missing. For these reasons we are motivated on finding a generic 3D representation for 3D objects with better scale invariance properties compared to methods from the literature.

1.3 Contribution

Although the problems referred in the problem definition section are comprehensively examined, and still being examined in the literature, there are still open points. In this thesis we propose solutions for some of these problems such as the ones indicated in the previous section.

The proposed feature extraction method is novel in the sense that it constructs the signal scale-spaces with a different strategy than the previous approaches, as it will be explained later in detail later. Via controlled experiments on artificial surfaces, we show that our approach might be beneficial in terms of accuracy and robustness to noise.

Secondly, as it will be explained later in the succeeding chapters, surfaces can be classified into certain types using different curvature spaces. For instance using Gaussian (K) and Mean (H) curvatures or using Shape Index (S) and Curvedness (C), surfaces are classified into the same eight types (pit, peak, etc.). Since shape index is scale invariant, previous comparisons show that using SC classification gives better results. However, using a scale-space approach, we show that HK classification gives better results in object recognition, which is a valuable clue for surface curvatures in scale space.

We propose a transform invariant topological surface representation which can be used to recognize, register or detect objects. The representation is defined within a feature vector, which is experimented for object recognition and detection.

Last, but not the least, we show that using surface curvatures in Digital Elevation Models (DEMs) provide fast and efficient methods for landslide region detection, compared to slow conventional methods.

1.4 Outline of the thesis

We commence this thesis by examining the related work in chapter 2. Previous studies on 3D feature extraction, 3D face detection and 3D object recognition using range images are given in this chapter.

The essential definitions of 3D signal processing are presented in chapter 3, "Processing 3D Information". Basic 3D surface data formats are introduced and basic methods on pre-processing 3D data for the purpose of cleansing common defects such as noise are discussed.

The next chapter, "Surface Curvatures" gives fundamental 3D curvatures definitions, such as: principal curvatures (κ_1, κ_2), mean curvature (H), Gaussian curvature (K), shape index (S), curvedness (C). In addition, the classification capabilities of these curvature definitions are described and discussed.

Chapter 5, "Scale Space of 3D Surfaces and 3D Curvature", dives into to the principal discussion in the thesis. Methods to obtain scale-spaces of 3D surfaces and curvature values from 3D surfaces, is analyzed. Crucial clues for scale and resolution invariant object recognition are obtained from the behaviour of 3D curvature values in scale-space.

The key technique proposed in this thesis is described in Chapter 6, "Feature Extraction and Object Representation". As its title implies, the method to extract transform invariant features from 3D surfaces is given in detail. Moreover the construction of transform invariant topologies using the extracted features is given in detail. Finally, a number of descriptors to define local regions around the extracted features are introduced.

The succeeding chapter, "Experimental Work" provides all implementations we have done using the proposed feature extraction and object representation techniques. The first subsection "3D Face Detection" gives the implementation of the proposed 3D feature extraction technique on 3D face detection and 3D facial pose estimation. Detection of facial surfaces is tested on Bosporus Database [Bosporus Database 2007] and compared to four main methods from the literature [Lu and Jain 2005], [Lu and Jain 2006], [Chang et. al. 2006], and [Colombo et. al. 2005]. Then the implementation of the main subject of this thesis is given in "3D Object Recognition". Using more than ten thousand range images from Stuttgart range image database [Stuttgart Database], object category recognition tests are carried out. Furthermore, comparisons between, multi-scale vs. single Scale features and recognition capabilities of HK and SC curvature definitions are given. Most importantly our proposed method is compared to 2D SIFT. In addition, the qualities of the extracted scale invariant features are evaluated for the purpose of 3D registration. Finally in "Landslide Region Detection Using Curvature values obtained from DEMs", using scale space of curvatures, regions with possible landslide activity are extracted from digital elevation models (DEMs). The proposed method is compared to a conventional, high computation cost technique [Multi-Watershed].

Finally, Chapter 8 concludes the dissertation. Final discussions are given and possible future directions are evaluated.

CHAPTER 2

RELATED WORK

As introduced in the previous chapter, in this dissertation, we discuss major issues of feature extraction and object representation. We also intend to propose novel techniques for them. Furthermore, the proposed techniques are evaluated on various applications such as, 3D facial recognition, 3D object recognition and 3D registration. Considerable numbers of studies and contributions, which are related to the theory and the experiments provided in this thesis, have been reported. For the sake of clarity, literature on each subject is examined separately in this chapter.

2.1 3D Feature Extraction

Feature extraction is one of the basic topics in image processing. Feature is a salient region on a signal, which locally defines or represents the signal behaviour. For example, corner is a good feature for 2D images since experiments show that removing the corners from images intensely impedes human recognition [Biederman 1987].

The main problem in this thesis, extracting features from 3D surfaces, is strongly correlated to the 2D feature extraction literature. For a detailed analysis on 2D invariant feature extraction, reader may refer to [Tuytelaars and Mikolajczyk, 2007]. For 3D surface change is related to surface curvatures. We commence by 3D surface curvatures, and then we continue with scale invariance concept in 3D pattern recognition. Finally we visit the literature on describing 3D surface behaviour locally.

2.1.1 3D Surface Curvatures

Using surface curvatures on range images for the purpose of image recognition and image segmentation [Fan et. al. 1986, Fan et. al. 1989, Besl and Jain 1986, Ittner and Jain 1985] dates back to 1980s, when range image acquisition started to become available. The practical side of the problem, estimating true curvature values from acquired range image is also studied in this era [Flynn and Jain 1989]. In the following years, mean and Gaussian curvatures are preferred methods in 3D surface classification literature. However, this representation lacks the notion of scale invariance, and the shape index (S) and curvedness (C) [Koenderink and Doorn 1992] are proposed, which are also based on principle curvature values.

Using H and K values or S and C values, HK and SC curvature spaces are constructed in order to classify surface patches into types such as: pits, peaks, saddles, parabolas, hyperbolas or planes. Since both HK and SC spaces classify surface patches in to similar types, their classification capabilities are comparable. For this reason, there is an ongoing debate on the advantages and disadvantages of using mean & Gaussian (HK) or shape index & curvedness (SC) curvature spaces for object recognition applications. In [Cantzler and Fisher 2001], HK and SC curvature descriptions are compared in terms of classification, impact of thresholds and impact of noise levels and it is concluded that SC approach has some advantages at low thresholds, in complex scenes and in dealing with noise. However in that study the curvatures are calculated only at the lowest scale, i.e. the given resolution. Scale-spaces of the surfaces or the curvatures are not defined. Another comparative study has been carried out in [Li and Hancock 2004] where curvature values obtained from the shading in 2D images are used and HK and SC histograms are created. The comparison results show that SC histograms are slightly more successful in terms of classification. Yet again, the tested resolution is the pixel resolution of the 2D image and the effect of sampling is ignored.

2.1.2 3D Scale Invariant Features

In the previous decade, scale concept has become common in computer vision. [Lindeberg 1998]'s work on automatic scale selection in feature point detection and same author's renowned book on scale space [Lindeberg 1994] attracted so much attention that, the very famous scale invariant feature transform [Lowe 2004] was discovered in the following years. However, for that decade, nearly all of the studies on scale concept were on 2D image processing. Although shape index [Koenderink and Doorn 1992] was introduced for 3D surfaces, scale invariant feature extraction from surfaces or range images did not attract much attention until recent years.

Recently [Li and Guskov 2007] extracts multi-scale salient features using only two scale levels of the surface normals and analyzes its performance on object recognition for the Stuttgart range image database. [Pauly et. al. 2003] extracts multi-scale features which are classified based on surface variation estimation using covariance analysis of local neighbourhoods, in order to construct line features. [Lo and Sieberta 2009] define the 2.5D SIFT, the direct implementation of SIFT [Lowe 2004] framework on range images, however they present their comparison with simple match matrices and avoid giving a comparison of recognition capabilities. All of these methods construct a scale space of the surface using difference of Gaussians (DoG) and seek for the maxima within this scale space. In addition, although these studies all promise scale invariance, none of them, or any other study to our knowledge, test the recognition ability of their proposed scale invariant 3D representation technique using a scale varying database of 3D models.

The other crucial requirement for 3D recognition, namely robustness under occlusion, has been issued in various ways previously [Johnson and Hebert 1999], [Merchán et. al. 2008]. Yet again to our knowledge, no particular study has been reported on testing the recognition capabilities with respect to both scale and occlusion in range image databases.

2.1.3 3D Descriptors

Range images, which have been usually processed as 2D images, carry both the 3D metric and geometric information of objects in the scene. Several local or global 3D point features and 3D descriptors are derived from these sampled surfaces and used for 3D object recognition, 3D object category recognition, 3D surface matching and 3D registration. Some of these features are SIFT [Lowe 2004] (as directly applied to 2D rendered range images), 2.5D SIFT [Lo and Sieberta 2009], multi-scale features [Li and Guskov 2007], [Pauly et. al. 2003], spin images [Johnson and Hebert 1999], 3D point signatures [Chua and Jarvis 1997], 3D shape context [Frome et. al. 2004], surface depth, normal and curvature histograms [Hetzel et. al. 2001], 3D point fingerprints [Sun et. al. 2001], extended Gaussian images [Horn 1984] or density-based 3D shape descriptors [Akgül et. al. 2007].

3D descriptors or histograms generally define the whole or a part of an object, using different properties of the surfaces such as curvatures, normal directions, distances to a base point etc. They are very powerful in representing a surface patch for recognition purposes. However when they are globally defined, they are brittle against occlusion. On the other hand, local descriptors are defined around feature points. However, detecting feature points and estimating the effect region of the local descriptor around a feature point are serious problems. Using fixed sized local descriptors obtained from random points on the surface [Johnson and Hebert 1999] is one of the earlier approaches.

2.2 3D Object Representation

The concept of representing a designated element on a signal, for example an object in an image, is application-based and application-dependent. Thus any proposed method related to the application field actually defines its own representation in some certain manner. For this reason, it is difficult to mention about a homogenous object representation literature. However there is still an observable evolution of 3D object representation methods in the literature.

This pioneering studies on 3D object recognition focus on model-based approached since first applications are on CAD-based vision systems. [Arman and Aggarwal 1993] present a dense review on these type approaches. Since these approaches generally define an object using basic shapes, like cubes, cylinders, etc.; the applications were limited to industrial objects where object is clearly acquired, and there is no need for segmentation. However as the need for detecting objects in a cluttered 3D scene rises up, more curvature based methods are required. As explained in the previous sections, curvature based approaches locally define shapes using surface behaviour. Approaches to represent objects using curvature values can be found in the literature [Flynn and Jain 1988]. These approaches are followed by feature based studies, which use curvature information over the surface to detect repeatable and invariant salient points and regions.

Advanced representations of 3D surfaces can be divided into two major branches. The first one is using different surface descriptors as defined in the previous sub-section. Other approach is using topologies where the spatial information of salient regions is taken into consideration. This topological representation may be considered both local and global. Furthermore various proposed methods provide robustness under occlusion for topological (or graph-based) 3D object representations.

2.2.1 3D Object Recognition

Similar to face detection, the literature on 2D object recognition is highly populated. However it is difficult to say that it is saturated, because there are still many open problems for both 2D and 3D object recognition. Reader may refer to [Edelman 1997, Roth and Winter 2008] for a detailed survey on 2D object recognition.

3D object recognition literature is mainly the combination of the studies on 3D surface curvatures, 3D feature extractors, 3D descriptors and 3D object representation. For this reason the major approaches on the subject is already given within the previous subsections. There are a couple of good survey papers on the subject [Osman et. al. 2004, Campbell and Flynn 2001]. It is quite expected that updated versions of these reports will soon be published since new developments quickly affect the field, such as probabilistic models' increasing usage in object recognition [Simon and Seitz 2007, Hu and Zhu 2010].

2.2 3D Facial Detection

2D face detection and recognition have attracted so much attention for the last forty years that the research in these fields is fairly saturated. For a review of 2D face detection and recognition algorithms, reader may refer to [Yang et. al. 2002] and [Zhao et. al. 2000] respectively. Although it is recently reported that 2D and 3D face recognition algorithms are comparable [Chang et. al. 2005], studies on 3D face detection and recognition methods have shown a remarkable progress in the last few years as 3D scanners become more and more available. For detailed surveys of 3D face recognition, reader may refer to [Kittler et. al. 2005] and [Bowyer et. al. 2004].

Most of the 3D face detection and recognition approaches are based on local facial features such as eyes, mouth, nose, profile, silhouette and face boundary. Success of these approaches depends highly on the success of the feature detection algorithms. Since, in this study, we also present a method for local feature extraction and we apply our method for face detection, we would rather summarize literature for 3D facial feature extraction and face detection.

Previous studies show that, nose tip and inner eye pits have usually been selected as anchor points on faces. [Colbry et. al. 2005] use "shape index" [Koenderink and Doorn 1999] for determining the anchor points. Chang and Bowyer [Chang et. al. 2006] also use shape index for the detection of inner eye pits and the nose tip is detected by using the eye pit locations. [Colombo et. al. 2005] find anchor point candidates by using shape index values and PCA is used in order to classify the anchor points. In addition shape index has also been used to recognize objects other than faces [Worthington and Hancock 2001]. [Nagamine et.al. 1992] finds five feature points by using a similar method of [Colombo et. al. 2005] and use those feature points to standardize face pose. [Lu and Jain 2005] first extract nose tip as the closest point to the camera. Secondly, possible locations of the two inner eye pit points are selected by using shape index and cornerness measure extracted from the face surface. Anchor point detection is performed in a similar way for rotated 3D data in [Lu and Jain 2006].

2.4 3D Object Registration

Surface registration is an intermediate but crucial step within the computer vision systems workflow. The goal of registration is to find the Euclidian motion between a set of range images of a given object taken from different positions in order to represent them all with respect to a reference frame. Registration in general can be divided into two: coarse registration and fine registration [Salvi et. al. 2006]. In coarse registration, the main goal is to compute an initial estimation of the grid motion between two clouds of 3D points using correspondences between both surfaces. In fine registration, the goal is to obtain the most accurate solution as possible. Needless to say that the latter method usually uses the output of the former one as an initial estimate so as to represent all range image points with respect to a common reference system. Then it refines the transformation matrix by minimizing the distance between the temporal correspondences, known as closest points.

In coarse registration, which is the main type of registration we intend to accomplish in this thesis, uses point-to-point correspondences, such as the [Chua and Jarvis 1997, Johnson and Hebert 1999]. An important aspect of coarse registration is the way of computing the motion when correspondences are found. Robustness in the presence of noise is another important property, because there are usually no corresponding regions between views. Studies on using a (partial) graph-to-graph approach on range images, instead of point-to-point correspondences, is not reported to our knowledge.

For a wide literature survey on registration of range images reader may refer to [Salvi et. al. 2006].

CHAPTER 3

PROCESSING 3D INFORMATION

In this chapter, different formats of 3D information are investigated. Since raw output of 3D scanners is used as the main source of 3D information, methods to cleanse certain defects from these types of data are also examined.

3.1 3D Information Formats

The 3D information can be found in various forms. Basically it is a point cloud of vertices in a three-dimensional coordinate system. However depending on the type of the signal, 3D information take different names.

3.1.1 Range Images

Range image (scan) is a 2.5D image showing the distance to points in a scene from a specific point, normally associated with some type of sensor device. The resulting image has pixel values which correspond to the distance. If the scanner which is used to produce the range image is properly calibrated, the pixel values can be given exactly in physical units such as meters.

The sensors used to obtain range images are usually called 3D scanners. The raw output of these 3D scanners is a regular grid of 3D points lying on a u-by-v mesh, namely a depth image (map) of the scene. This representation is known as the Graph Representation [Flynn and Jain 1989], where a single 3D point on the surface is given by:

$$\mathbf{X}(u,v) = \left[x(u,v) \quad y(u,v) \quad z(u,v) \right] \tag{1}$$

In Figure 1, a 2D demonstration of a range image is seen. Each pixel value indicates the distance of that pixel from the scanner. However a range image encapsulates more information than a 2D image since for each pixel coordinate *(u, v)*, there are three metric values: *x(u, v)*, *y(u, v)* and *z(u, v)*. For this reason, they are also referred as 2.5D data. The regions with zero pixel value are invalid regions. These regions have no valid depth values thus no valid 3D coordinates.

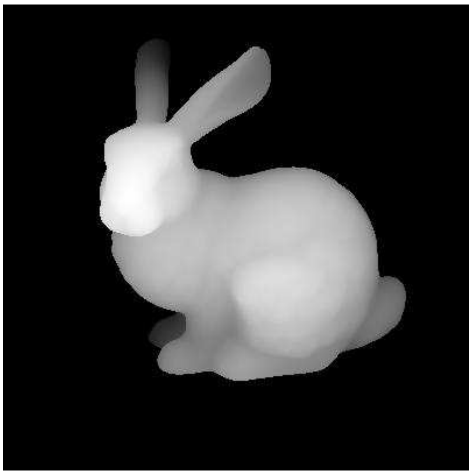


Figure 1. Sample range image: “bunny” from [Stuttgart Database]

There are several types of 3D scanners such as laser, ultrasound and optical. Each technology comes with its own limitations, advantages and costs. For pattern recognition and robot vision applications using real world objects, 3D optical scanners are usually preferred.

The range image is the most basic form of 3D data. This regular data may be manipulated in different ways for different applications. The next subsection briefly goes over some other frequently used 3D data formats.

3.1.2 Polygonal Data

For visualization purposes, computer graphics based applications require a set of polygonal surfaces. For instance, if the valid points are extracted from a range image and vertex groups that form a plane are found, the 3D information within the range image is converted to 3D polygonal data. Polygonal data is crucial since most commercial 3D graphical interfaces used in today’s computers are designed to process polygonal meshes.

Unlike range images, polygonal data may include a complete model of the object. When a range image of an object is taken, only the 3D points across the viewing angle are captured. However combining various range images of an object taken from different viewing angles, a polygonal mesh of the complete model of the object can be constructed (Figure 2).

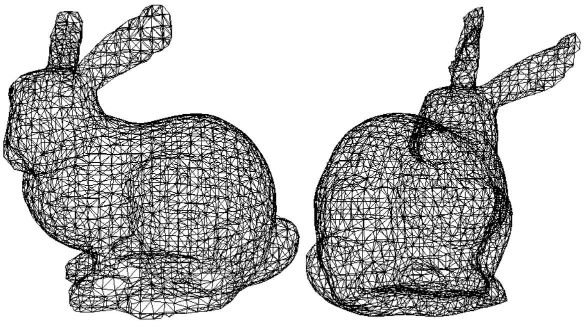


Figure 2. The polygonal form of the object in Figure 1 is obtained by combining the valid vertices obtained from different range scans of the object.

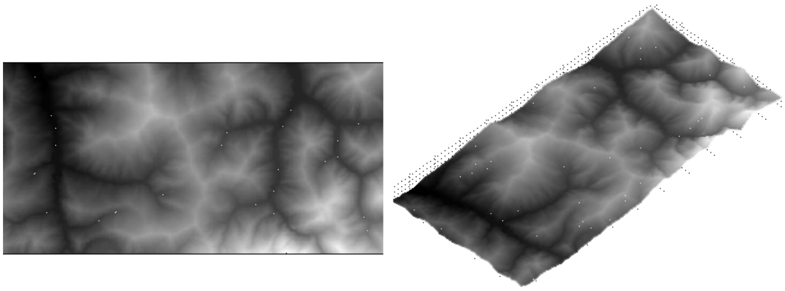
The resolution of polygonal models may be altered without losing 3D shape information. For example, an area with low curvatures (e.g. a planar area) may be represented using few vertices. However depending on the data acquisition technique, this area might be full of 3D points, which are mostly redundant. In this case, a process called decimation is applied on polygonal meshes. Regions with low curvature value are found and the sampling around these regions is reduced using several techniques.

3.1.3 Parametric Representations

Some computer vision based techniques and graphics applications require a continuous representation of 3D data in order to apply analytic methods instead of numeric ones. For this reason methods like spline fitting may be applied to 3D scanner outputs in accordance with the required application. In these methods the main motivation is to transform the digitized data into continuous and parametric form by fitting a certain type of function to the valid vertices of the object surface. Having continuous surface functions and derivatives are the benefits of this technique. Regression and spline fitting are some the examples of parametric representations.

3.1.4 Digital Elevation Models

A digital elevation model (DEM) is a digital representation of ground surface topography or terrain (Figure 3). It is also widely known as a digital terrain model (DTM). A DEM can be represented as a raster (a grid of squares) or as a triangular irregular network. DEMs are commonly built using remote sensing techniques; however, they may also be built from land surveying. DEMs are used often in geographic information systems, and are the most common basis for digitally-produced relief maps.



a) b)

Figure 3. a) The DEM image is depicted. b) The 3D render of the DEM image is shown.

DEMs are similar to 2D images where the pixel intensity designates the elevation in metric coordinates. Usually each pixel is captured with a fixed metric sampling rate (such as using 20m intervals), thus the horizontal coordinates also represent metric coordinates and the information is actually 2.5D.

The quality of a DEM is a measure of how accurate elevation is at each pixel and how accurately is the morphology presented. Several factors play an important role for quality of DEM-derived products, such as: terrain roughness, sampling density, grid resolution or pixel size, interpolation algorithm, vertical resolution and terrain analysis algorithm.

DEMs are very similar to range scans in the sense that the depth and the pixel size are in metric coordinates. Thus 3D curvatures techniques can be applied to them as well. In chapter 8, a method to detect landslide regions using curvature information of DEMs is proposed.

There are a number of other 3D data formats, such as volumetric representations. We chose not to mention them here, since the four formats mentioned above, namely range scans, polygonal data, parametric representations and DEMs are the basic formats that will be used in this thesis. Range scans will be our basic source signal on which we will process our feature extraction techniques. We will sometimes convert them to polygonal data for visualization purposes since most 3D visualization software and hardware use 3D polygonal formats. Parametric representations will be needed for finding surface gradients on local surface patches. Splines will be fit on small local regions and surface gradients will be extracted from their explicit analytical equations. Finally surface curvatures will be calculated on scale-spaces of DEMs in order to automatically detect landslide regions.

3.2 Common Defects of 3D Scanner Outputs

3D scanners can send trillions of light photons toward an object and receive only a small percentage of those photons back via the optics that they use. For widely used optical scanners, a lighter surface will reflect lots of light where a darker surface will reflect only a small amount. For this reason more specular and darker regions, such as dark hair, might not be detected by these types of scanners and might be marked as invalid on the scan. Moreover, physical limitations of the sensor may lead to noise on the acquired data set. Sample points can be corrupted by quantization noise or motion artefacts. Furthermore multiple reflections and heavy noise can produce off surface points (outliers). Besides, holes on the model surface occur due to occlusion, critical reflectance properties, constraints in the scanning path or limited sensor resolution. Many scanners tend to create ghost geometry when the scanned object is textured. In this section some common defects of 3D scanner are investigated.

3.2.1 False Holes (False Invalid Points)

Due to various reasons, 3D scanners may mark regions as invalid. 3D scanners usually work on optical principles to calculate 3D coordinates of points on their image planes. However for example, dark hair may reflect light photons like a mirror and the scanning system may be unable calculate valid coordinates for that region. Then, the scanner marks points in this particular region as invalid. It is clearly seen in 3D facial databases [Bosporus Database 2007] that the, hairy regions such as: dark hair, eyebrows, eyelashes, darker facial hair are usually marked as invalid by the optical scanner used (Figure 4).

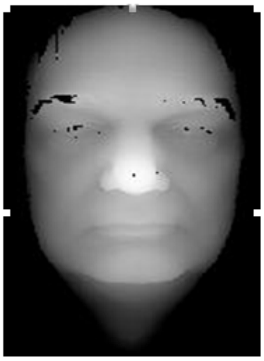


Figure 4. A facial scan taken from FRGC v1.a 3D database is shown. Invalid regions around eyebrows are marked as invalid in the depth image.

3.2.2 Exploded/Imploded Regions

Optical scanners tend to create spikes mostly because of similar reasons why they create holes. For regions with critical reflectance properties (such as specular regions like dark hair as mention in the previous subsection) the reflected light on image planes might be localized incorrectly. Thus instead of having invalid coordinates, these points might have valid flag, but they carry incorrect depth values.

In Figure 5, a spike on the iris of a facial scan from FRGC v1.a 3D facial database is seen. Since iris is a highly specular region, one particular pixel depth is incorrectly calculated and a spike is formed.

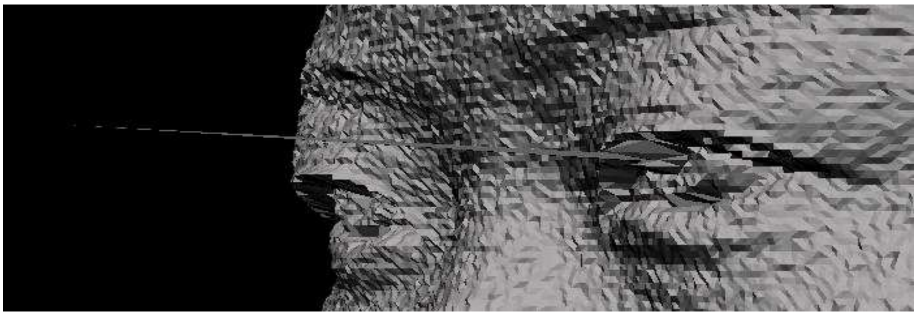


Figure 5. A spike on the iris of a facial scan from FRGC 3D facial database.

These incorrectly calculated points are not necessarily outwards the direction of the camera. The might be imploded inwards the 3D surface. In addition these regions are not necessarily single points; they might be a group of points if the reflectance is critical for that whole region (Figure 6).

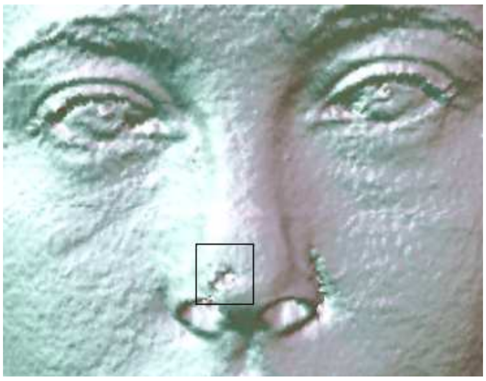
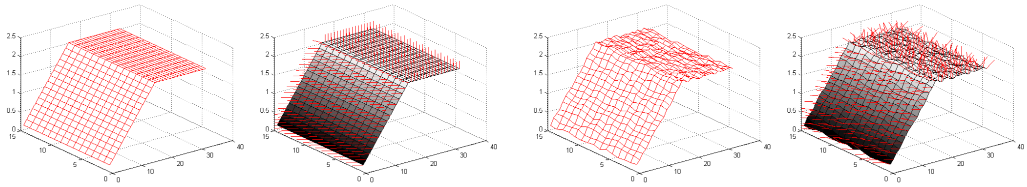


Figure 6. Nose is imploded in the 3D range scan (left) because of extreme illumination on the nose tip (FRGC v1.a).

3.2.3 Noise

For any remote sensing device, from a radio receiver to 3D scanner, some kind of noise is observed on the output signal. Transmitting media or the components of the device itself may cause the noise. Despite any effort to neutralize this noise, it is impossible to completely cleanse it from any electronic remote sensing device.



a) b) c) d)

Figure 7. a) 3D artificial surface without noise b) Normal directions over the noise-free surface. c) 3D artificial surface with noise. d) Normal directions over the noisy surface [Bozkurt et. al. 2009].

Since most 3D scanners use optics to acquire the signal, the type of noise is similar to 2D noise. 3D scanner outputs experience amplifier noise, which is highly Gaussian. There is also powerful quantization noise if the resolution of the scanner is relatively low [Bozkurt

2008]. 2D noise is convolved within the intensity of the pixels. 3D noise disturbs the 3D coordinates of the points in the point cloud. A demonstration of noise over an artificial surface is depicted in Figure 7.

3.3 Processing 3D range scans

In order to neutralize the defects of 3D range scans, a processing on the 3D point cloud may be needed. Unsurprisingly, depending on the type of the defect, the processing method varies. Only after these defects are neutralized, the range scan is properly prepared for the required application. In this section, methods to process these defects are inspected.

3.3.1 Processing False Holes

Theoretically holes are invalid marked points, which actually belong to a group of points with valid 3D coordinates. In Figure 4, a good example is given from FRGC database, where some point on the eyebrows of a 3D facial scan are marked as invalid. However these so called holes may belong to the image background, or an actual hole which sees through the image background (Figure 8), which should have invalid coordinates.

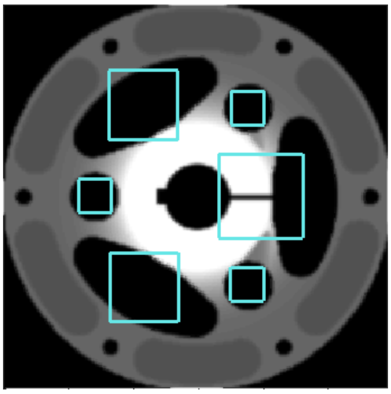


Figure 8. The “hub” object from the Stuttgart 3D range image database is shown. There is no defect on the range scan; however the object possesses a number of actual holes.

The range image in Figure 8, has no defects, however the object possesses a variety of actual holes. Thus, before processing the image for false invalid points, one should decide if the invalid marked points is a defect or not. In order to solve this problem manual and automatic methods are used. The manual methods are not feasible when thousands of range image are concerned, where automatic methods have no guarantee of truly discriminating false holes among invalid points. Most 3D scanners come up with a software which can be used to manually fill these holes. These interfaces help the user to mark the region encapsulating the false hole, and interpolate it using the neighbouring valid points. The interpolation could be linear or quadratic. On the other hand a fully automatic method would detect the false holes by means of some intelligent criterion. This criterion could be as simple as the size of the hole, or as complex as a model-based approach guessing where the object might have an actual hole.

In this thesis, we have completed our experiments on FRGC v.1a 3D facial database [FRGC1a], Bosporus 3D facial database [Bosporus Database 2007] and Stuttgart 3D range image database [Stuttgart Database]. These databases include near twenty thousand range images in total. FRGC database includes the raw scanner outputs which are noisy and full of false holes. We have applied automatic processing to these holes, where holes having a certain size smaller than a threshold were considered false, and interpolated linearly using the neighbouring valid points (Figure 9).

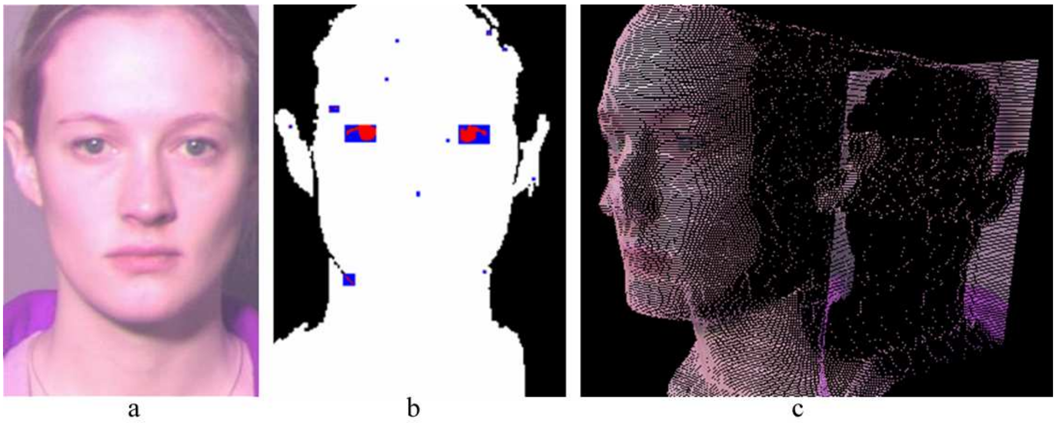


Figure 9. a) Texture for the 3D Scan. b) Black areas denote the background. Red areas denote the invalid points selected to be extrapolated. Blue areas denote the neighbouring valid points used for extrapolation. White areas denote valid points. c) Processed point cloud.

The same process was also applied to Bosporus database although a manual processing was applied to this database during the acquisition. Stuttgart database is an artificial database, which is constructed artificially by obtaining range images from 3D objects by virtually scanning them. For this reason this database is free of defects and does not need any processing.

3.3.2 Processing Exploded/Imploded Regions

It is nearly impossible to construct a decision system which will detect any incorrect valid point, because the structure of the scanned surface is usually unknown. For example a spike on the surface may very well be a valid surface structure if the scanned surface includes a needle. For this reason widely used techniques to process these regions are manual. Similar to manual hole filling software, there are several programs which provide semiautomatic tools to cleanse these defects from 3D range images.

On the other hand, if there is preliminary structural knowledge about the scanned surface, then certain model-based automatic methods may be applied. For example, Nesli Bozkurt, a former graduate student from our computer vision laboratory in METU, constructed a facial defect cleansing system which uses the symmetry information of the face. So for facial scans, a lateral symmetry criterion is sought over the 3D facial surface and any element (a hole or a spike) violating this criterion is considered as a defect and handled properly [Bozkurt 2008].

3.3.3 Processing Noise

3D scanner outputs experience amplifier noise, which is highly Gaussian. There is also powerful quantization noise if the resolution of the scanner is relatively low [Bozkurt 2008]. This noise can be partially removed by using a Gaussian smoothing filter. However Gaussian filters experience problems with preserving the edge details. Among numerous methods, Bilateral Filtering is proven to be a powerful yet simple, non-iterative scheme for edge-preserving smoothing in 2D [Bozkurt 2008]. This image processing technique has been successfully extended to 3D by making small modifications in the application.

Bilateral filter is distinguished from Gaussian blur by its additional “range” term. In this case, not only the distances between positions matter, but also the variation of the intensities are also taken into account in order to keep the features from derogation. Consequently, the edges where high intensity differences occur are successfully preserved. In 2D, the intensity values are a function of position values, whereas in 3D the position in fact, is the signal itself. Hence, the modification of bilateral filter to be applied to 3D data is not very straightforward. Like most image processing algorithms that are extended to surfaces, normal information at each point of the surface can be used to form an intensity space like in images.

The effect of noise cleaning using 3D bilateral filtering is depicted in Figure 10, which is taken from [Bozkurt 2008]. As seen from the figure, the edges are properly preserved in 3D bilateral filtering (Figure 10).

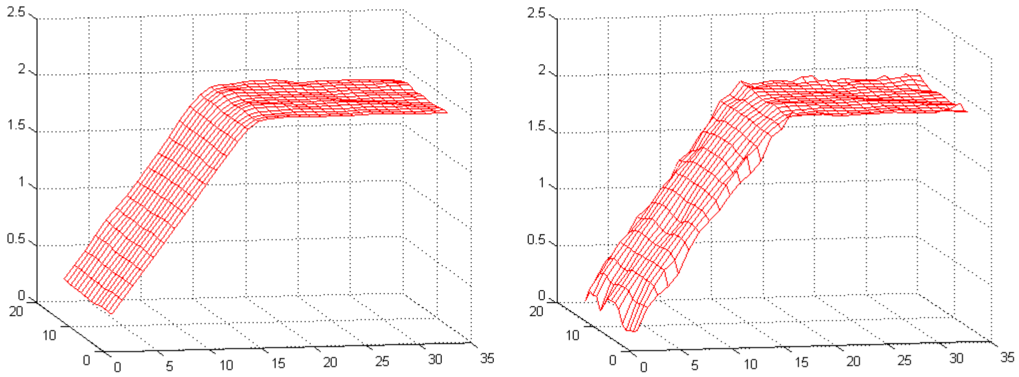


Figure 10. Surfaces de-noised with Gaussian (left) and Bilateral (right) filters [Bozkurt 2008].

When the bilateral filter is applied to original scans from the FRGC v1.a database, the results show that the edge details around the eye is well preserved whereas the facial surface is cleared off the noise (Figure 11).

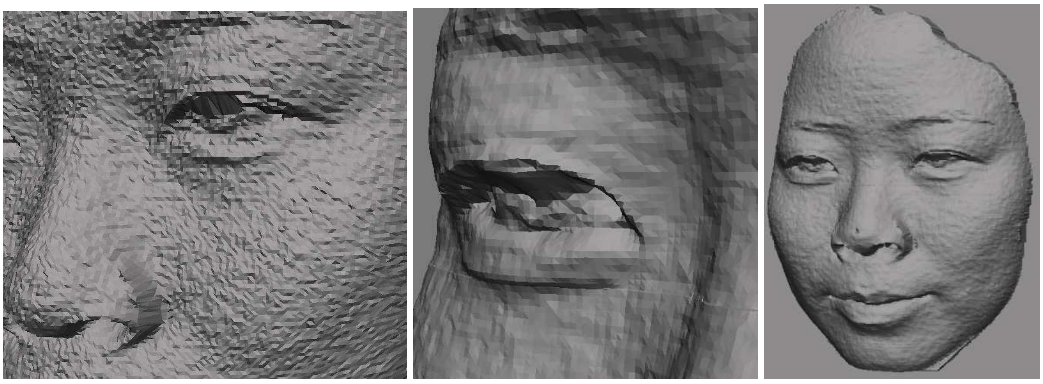


Figure 11. The result of the bilateral filtering on a FRGC sample. .

CHAPTER 4

SURFACE CURVATURES

In order to construct intelligent systems capable of classifying different shapes, one should find a way to understand or model the behaviour of a solid shape. When it comes to 3D shapes and surfaces, the surface curvatures are the key concept defining this behaviour. For a general review of surface curvatures and surface behaviour, reader may refer to Koenderink's renowned book on solid shapes [Koenderink 1990].

In this chapter the general concepts of surface curvatures, such as: surface gradients, principal curvatures (κ_1, κ_2), Mean curvatures (H) and Gaussian curvatures (K), shape index (S) and curvedness (C) are reminded. In addition, the transform invariance and the effects of scaling and sampling on these curvatures are discussed. By the end of the chapter, a theoretical comparison of HK and SC fundamental shape definitions is presented. We commence by discussing the most reliable way to obtain a curvature of a digitized surface.

4.1 Reliable Curvature Estimation

The behaviour of the surface may be calculated by the first partial derivative of the surface function. In the previous chapter, the general digital surface data format is analyzed specified as 2.5D data or the Monge Patch, which is defined as:

$$\mathbf{X}(u,v) = \begin{bmatrix} u & v & f(u,v) \end{bmatrix}; \{u,v\} \varepsilon\, \mathbf{R} \qquad (2)$$

As it is previously stated, u and v parameters are matched to x and y coordinates and this surface can be specified as the height $f(u, v)$ above the so-called support plane defined by the two coordinates (u, v). At any point p on this continuous surface, we can choose two orthogonal vectors in the tangent plane and examine the surface's behaviour in those directions. The vectors in the direction of the orthogonal parameters (u, v) are called the surface gradients and are equal to:

$$\partial \mathbf{X}(u, v)/\partial u = [\,1 \quad 0 \quad f_u\,]\,,\ \partial \mathbf{X}(u, v)/\partial v = [\,0 \quad 1 \quad f_v\,] \tag{3}$$

f_u and f_v are the partial derivatives of the height function $f(u, v)$ with respect to u and v. Accordingly the surface normal n, which is orthogonal to surface gradients at point p, is defined as:

$$\mathbf{n} = \frac{\partial \mathbf{X}}{\partial u} \times \frac{\partial \mathbf{X}}{\partial v} = \begin{bmatrix} -f_u & -f_v & 1 \end{bmatrix} \tag{4}$$

Theoretical computation of the surface gradients is simple when an explicit definition of the continuous surface function is known. However this is not the case when you are dealing with digitized scans in real world problems. For this reason a digital version of all functions or kernels must be used. There are different methods of obtaining partial derivates from a digital signal. [Flynn and Jain 1989] focuses on reliable digital curvature estimation using different types of these methods and strives to find a conclusion on the best method in order to estimate surface gradients of discrete surfaces. They implement analytical methods such as: fitting orthogonal polynomials and splines and linear regression around the point at which the gradient estimates are done. In addition they implement numerical estimates where finite difference is used.

This pioneering study emphasises on the importance of Gaussian smoothing before any digital curvature estimation. They check the robustness of all methods against noise and indicate that pre-smoothing enhances the estimation success profoundly. For this reason in this study all of the surfaces are pre-smoothed in order to lower the effect of any type of noise.

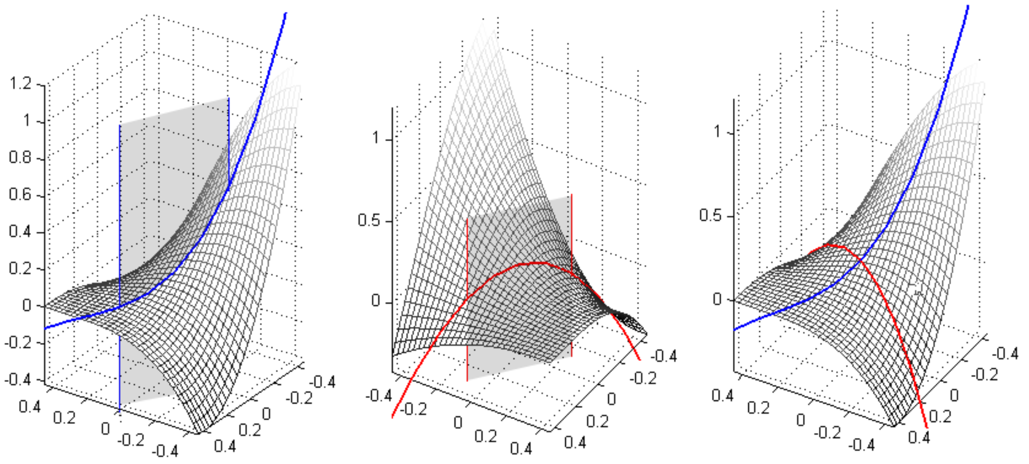
In their study, Flynn and Jain use 7x7 or 5x5 patches around the interest point, according to the method implemented. Needless to say that when estimating digital curvature this neighbouring patch size is of extreme importance since the discrete derivatives highly depend on the sampling rate. For this reason, even though they make very important indications in their study; it is difficult to rely on some of their observations since the data they use is very much lower in resolution compared to today’s digitized scans.

One of the main motivations of this thesis is to eliminate the effect of sampling or the neighbouring patch selection when estimating the digital curvatures for object recognition. In this study, 3x3 patches are used and quadratic functions are fit to estimate curvatures.

However these curvatures are investigated in a scale space of the surface. The details of these procedures are given in the succeeding sections.

4.2 Principal Curvatures

In differential geometry, the two principal curvatures at a given point of a surface measure how the surface bends by different amounts in different directions at that point. At each point *p* of a differentiable surface in 3D Euclidean space one may choose a unique unit normal vector. A normal plane at *p* is one that contains the normal, and will therefore also contain a unique direction tangent to the surface and cut the surface in a plane curve. This curve will in general have different curvatures for different normal planes at *p*. The principal curvatures at *p*, denoted κ_1 and κ_2, are the maximum and minimum values of this curvature. The figure below depicts these curvatures and their normal planes.



a) b) c)

Figure 12. a) The normal plane with the maximum curvature is seen. b) The normal plane with the minimum curvature is seen c) $\kappa_1 = 1.56$ and $\kappa_2 = -2.37$. The surface is a patch from a monkey saddle: $z(x,y) = x^3 - 3 \bullet x \bullet y^2$.

The principal curvature values and the principle directions of the curvatures are calculated by taking the eigenvalue decomposition of the Hessian Matrix which is defined as:

$$\mathbf{H} = \begin{bmatrix} \dfrac{\partial^2 \mathbf{X}}{\partial u^2} & \dfrac{\partial^2 \mathbf{X}}{\partial uv} \\ \\ \dfrac{\partial^2 \mathbf{X}}{\partial uv} & \dfrac{\partial^2 \mathbf{X}}{\partial v^2} \end{bmatrix} \tag{5}$$

The eigenvalues of this symmetric matrix give the principal curvatures κ_1 and κ_2, where the eigenvectors give the principle curvature directions. Surface points can be classified according to their principal curvature values at that point. A point on a surface is called:

Elliptic: ($\kappa_1 \cdot \kappa_2 > 0$) if both principal curvatures have the same sign. The surface is locally convex or concave.

Umbilic: ($\kappa_1 = \kappa_2$) if both principal curvatures are equal and every tangent vector can be considered a principal direction (and *Flat-Umbilic* if $\kappa_1 = \kappa_2 = 0$).

Hyperbolic: ($\kappa_1 \cdot \kappa_2 < 0$) if the principal curvatures have opposite signs. The surface will be locally saddle shaped.

Parabolic: ($\kappa_1 = 0$, $\kappa_2 \neq 0$) if one of the principal curvatures is zero. Parabolic points generally lie in a curve separating elliptical and hyperbolic regions.

This is the basic classification for surfaces according to their principal curvatures. Examples of these types of shapes can be seen in Figure 13.

Mean (H) and Gaussian (K) curvatures, shape index (S) and curvedness (C) are also calculated using the principal curvatures and more essential classification are made according to their values.

4.3 Mean (H) and Gaussian (K) Curvatures

Using principal curvatures, Mean (H) and Gaussian (K) Curvatures are calculated as:

$$H = \frac{\kappa_1 + \kappa_2}{2}, \; K = \kappa_1 \cdot \kappa_2 \tag{6}$$

H is the average of the maximum and the minimum curvature at a point, thus it gives a general idea on how much the point is bent. K is the multiplication of the principal curvatures and its sign indicates whether the surface is locally elliptic or hyperbolic. [Besl 1986] was the first to use HK values for the purpose of surface segmentation. Using HK values, the regions are defined as in

Table 1. Shape Classification in HK curvature space. Corresponding examples of surface regions are seen in Figure 13.

	K>0	K=0	K<0
H<0	Convex (1) (*Elliptic or Umbilic*)	Ridge (2) (Convex *Parabolic*)	Saddle Ridge (3) (*Hyperbolic*)
H=0	(*Not possible*)(4)	Planar (5)	Minimal (6)
H>0	Concave (7) (*Elliptic or Umbilic*)	Valley (8) (Concave *Parabolic*)	Saddle Valley (9) (*Hyperbolic*)

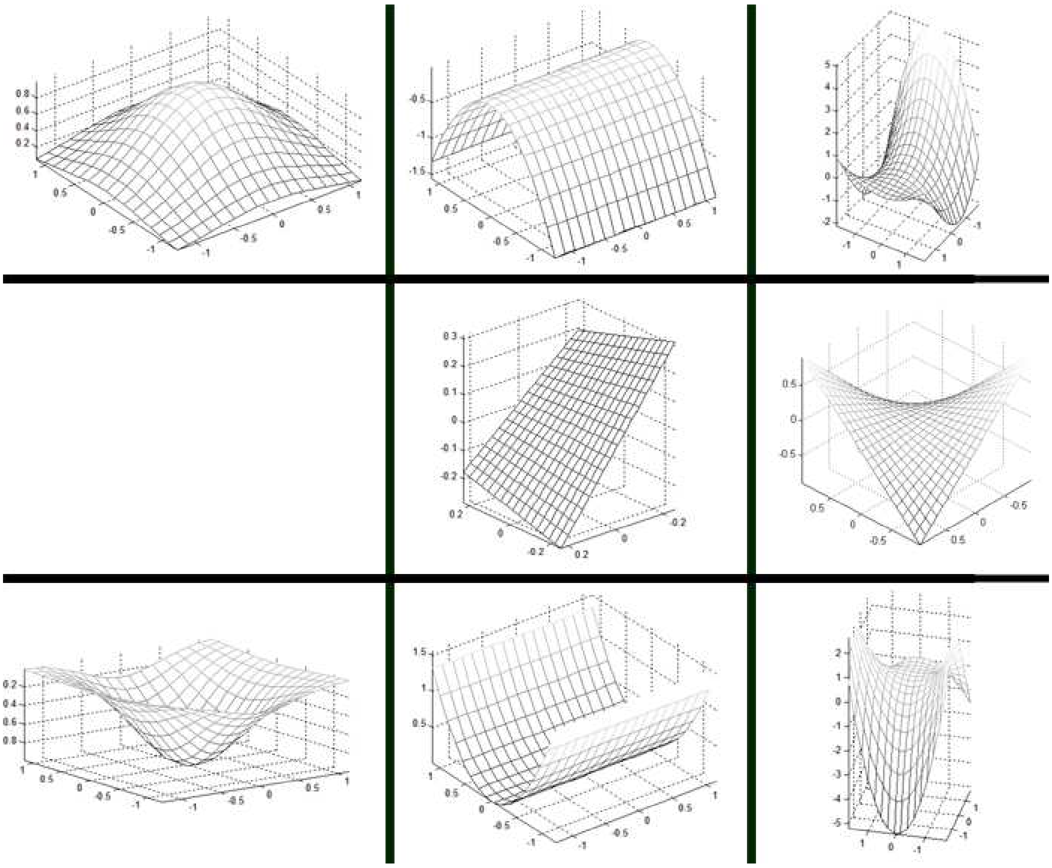


Figure 13. Shape types in correspondence with Table 1 Shapes at rightmost column are hyperbolic and from top to bottom they have the properties: $\kappa_1 + \kappa_2 < 0$, $\kappa_1 + \kappa_2 = 0$ and $\kappa_1 + \kappa_2 > 0$. The centre column show parabolic or flat-umbilic (planar) shapes where one of the principal curvatures is equal to zero ($\kappa_1 = 0 \cup \kappa_2 = 0$). The leftmost column show the elliptic shapes where both principal curvatures have the same sign. The upper is convex elliptic (a peak: $\kappa_1 < 0 \cap \kappa_2 < 0$) and the lower one is concave elliptic (pit: $\kappa_1 > 0 \cap \kappa_2 > 0$).

Because of noise and other reasons it is impossible to have an exact zero value for HK values. Thus zero-thresholds are used to decide if a value is zero or not. Everything below this threshold is considered as zero.

$$H = \begin{Bmatrix} (\kappa_1 + \kappa_2)/2 & if \left|(\kappa_1 + \kappa_2)/2\right| > H_{zero} \\ o & otherwise \end{Bmatrix}, \; K = \begin{Bmatrix} \kappa_1 \cdot \kappa_2 & if \left|\kappa_1 \cdot \kappa_2\right| > K_{zero} \\ o & otherwise \end{Bmatrix} \quad (7)$$

In order to better comprehend the relation between the HK values and the principal curvatures κ_1 and κ_2, the regions in Table 1 are drawn on the (κ_1, κ_2) plane in the figure above:

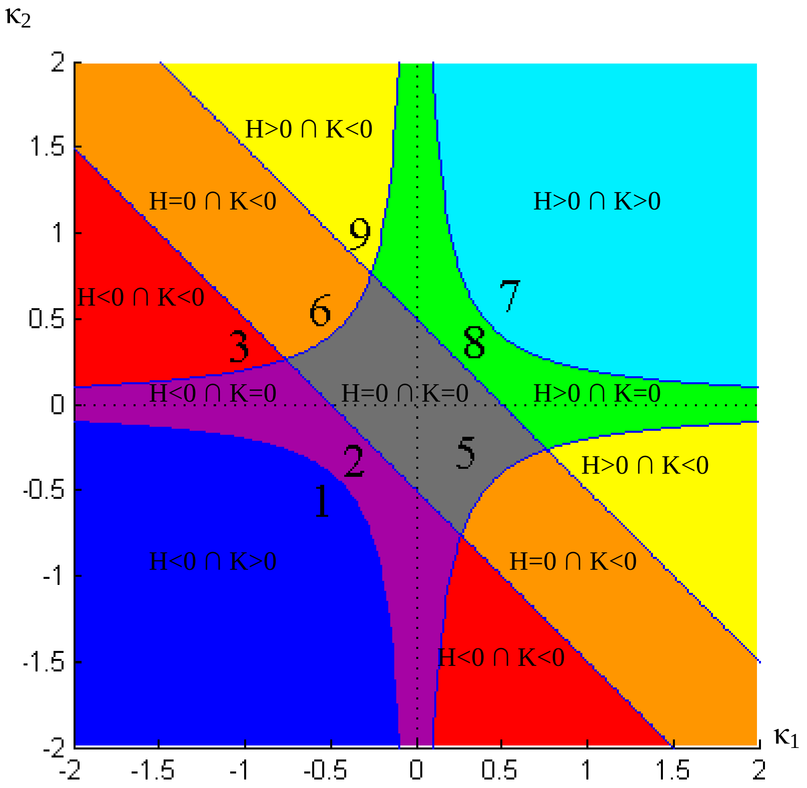


Figure 14. The HK classification on the (κ_1, κ_2) plane. Region colours (and the numbers) correspond to Table 1. The separating lines are the zero-thresholds for H values and their equations are: $(\kappa_1 + \kappa_2) / 2 = +H_{zero}$ and $(\kappa_1 + \kappa_2) / 2 = -H_{zero}$. The separating curves are the zero-threshold for K values with the equation: $(\kappa_1 \bullet \kappa_2) > +K_{zero}$ and $(\kappa_1 \bullet \kappa_2) > -K_{zero}$.

As discussed in [Cantzler and Fisher 2001], there are two main ambiguities in HK representation. The planar region is not symmetric and the parabolic region gets narrower for higher curvatures.

Most importantly H and K values are very much dependent on thresholds. Shifting the thresholds directly changes the regions on the (κ_1, κ_2) plane. Since for a scaled or re-sampled version of a surface patch, κ_1 and κ_2 values will differ. Thus HK values are not scale or resolution invariant. However since they depend on κ1 and κ2 values (which are orientation invariant) they are also orientation invariant.

4.4 Shape Index (S) and Curvedness (C)

[Koenderink and Doorn 1992] defines an alternative curvature representation using the principal curvatures. This approach defines two measures: the shape index (S) and the curvedness (C). Shape index (S) defines the shape type and curvedness (C) decides if the shape is locally planar or not.

$$S = \frac{2}{\pi}\cdot \arctan\left(\frac{\kappa_1 + \kappa_2}{\kappa_1 - \kappa_2}\right) \quad (\kappa_1 > \kappa_2), \text{ and } \quad C = \sqrt{\frac{\kappa_1^2 + \kappa_2^2}{2}} \qquad (8)$$

The shape index value of a point is independent of the scaling of that shape and it changes between [-1,+1]. However C is not scale or resolution invariant. Both S and C are orientation invariant. A discussion on transform, scale and resolution independency of all curvature types are given in the next subsection.

[Koenderink and Doorn 1992] uses S value in order to classify a point. S values changes between [-1,+1] where -1 defines cup shapes (convex elliptical) and + 1 defines cap shapes (concave elliptical). All other types of local shape correspond to real number between [-1,+1] (Table 2).

Table 2. Shape Index Classification. Colours are consistent with the previous figures

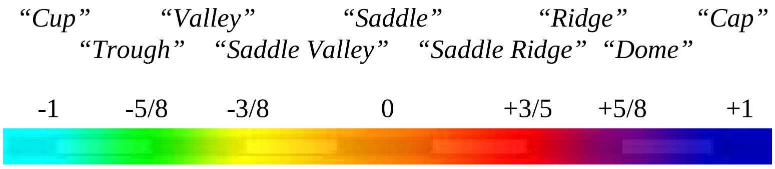


Table 3. Shape Index and Curvedness Classification. Colours are consistent with the previous figures.

Convex (Elliptic) (1)	$S \in [+5/8,1] \cap C>C_{zero}$
Convex (Parabolic) (2)	$S \in [+3/8,+5/8] \cap C>C_{zero}$
Saddle Ridge (3)	$S \in [+3/16,+3/8] \cap C>C_{zero}$
Planar (5)	$C<C_{zero}$
Hyperbola (6)	$S \in [-3/16,+3/16] \cap C>C_{zero}$
Concave (Elliptic) (7)	$S \in [-1,-5/8] \cap C>C_{zero}$
Concave (Parabolic) (8)	$S \in [-5/8,-3/8] \cap C>C_{zero}$
Saddle Valley (6)	$S \in [-3/16,+3/16] \cap C>C_{zero}$

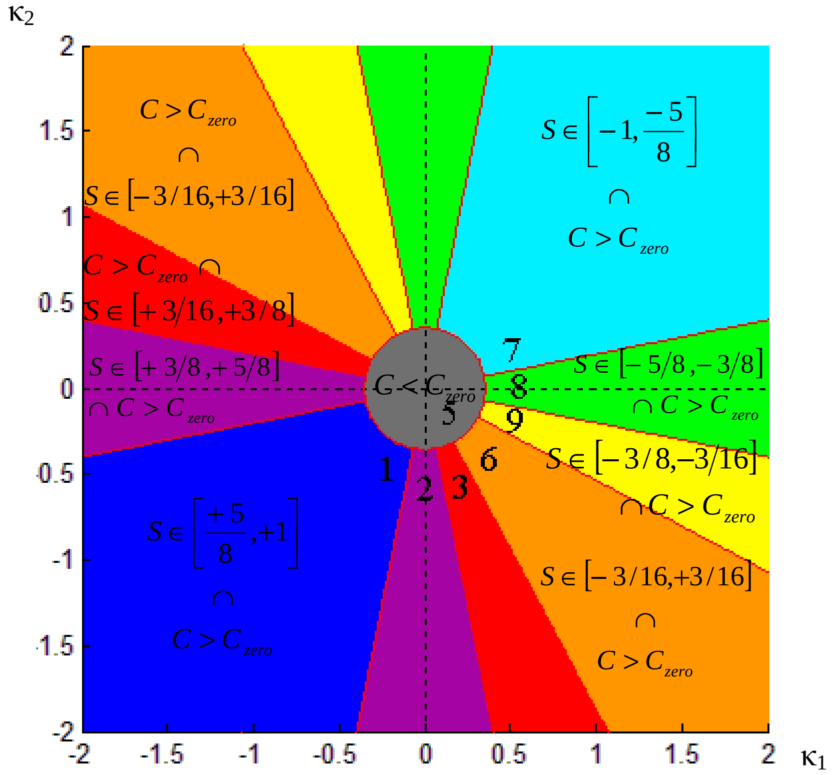


Figure 15. The SC classification on the (κ_1, κ_2) plane. Region colours correspond to Table 3. The separating lines are the constant S values: S = ±5/8, ±3/8 and ±3/16. The circle around the origin is the zero-threshold for C values and its equation is: $\sqrt{\left(\kappa_1^2+\kappa_2^2\right)/2} = C_{zero}$.

[Koenderink and Doorn 1992] defines constant shape index values in order to define shape types. These values are given in Table 3above. However their original classification does not differentiate hyperbolic regions into three different types (yellow-orange-red region, i.e. saddle valley, hyperbola and saddle ridge). For this reason we define another constant shape index value (3/16) for this purpose in the table above. The Curvedness (C) values are used to understand if the region is planar or not. For planar regions C value is very close to zero (i.e. below the zero threshold C_{zero}).

In order to better comprehend the classification of S and C values, it is a good idea to observe them on the (κ_1, κ_2) plane. The regions are coloured (and numbered) correspondent to Table 3 (and the previous figures).

4.5 Transform Invariance of Curvatures

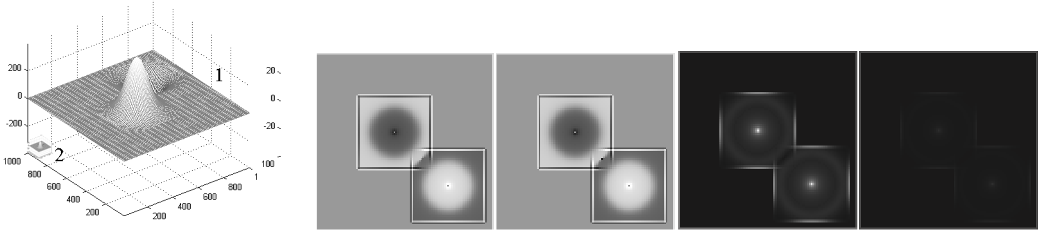
The curvature values obtained from a surface is independent of any 3D rotation. In other words for the same surface, when rotated in 3D space the principal curvature values stay the same. This is a very important property since when recognizing objects, local surfaces on that object may be captured in an arbitrary orientation and provided that the principal curvature values stay the same; the features obtained from principal curvature are said to be orientation invariant. Thus the value H, K, S and C are independent of translation and rotation in 3D space.

Unfortunately when it comes to scaling, things get complicated. Since scaling changes the finite difference between each digital sample, the discrete partial derivatives on the surface also changes for a scaled version of the same surface. Given that the principal curvatures are calculated from the Hessian, i.e. the second order partial derivatives of a surface; κ_1 and κ_2 are directly dependent to scaling and vary proportionally when the surface is scaled in 3D space. In addition since the 3D scans are discrete surfaces, the sampling rate also directly affects the principal curvature values. Obviously while principal curvature values are affected, any other type curvature will also be affected. For this reason H, K and C values are not scale or resolution invariant. However S values, due to their mathematical design, are independent of scale and resolution. The effect of scaling and sampling (resolution) is profoundly analysed in the next subsection.

4.5.1 Effect of Scale and Sampling on Curvature – Scale/Resolution Ratio

In this subsection the scale and resolution invariance of the four curvature types (H, K, S, C) are practically tested. Although their scale/resolution independency or dependency is theoretically or mathematically proven, the empirical tests are observed for a better understanding of the concept of *scale/resolution ratio*, which will be introduced within this subsection.

Shape Index is theoretically (mathematically) proven and designed to be scale invariant. In order to test the scale invariance of S practically, the simplest test would be having different scaled versions of a digitized surface (a Monge Patch) and comparing their S maps. Imagine we have two versions of the surface, one of which is 10 times bigger than the other in x, y and z coordinates (10 times scaled version). However their resolutions are exactly the same: 100x100 (Figure 16).



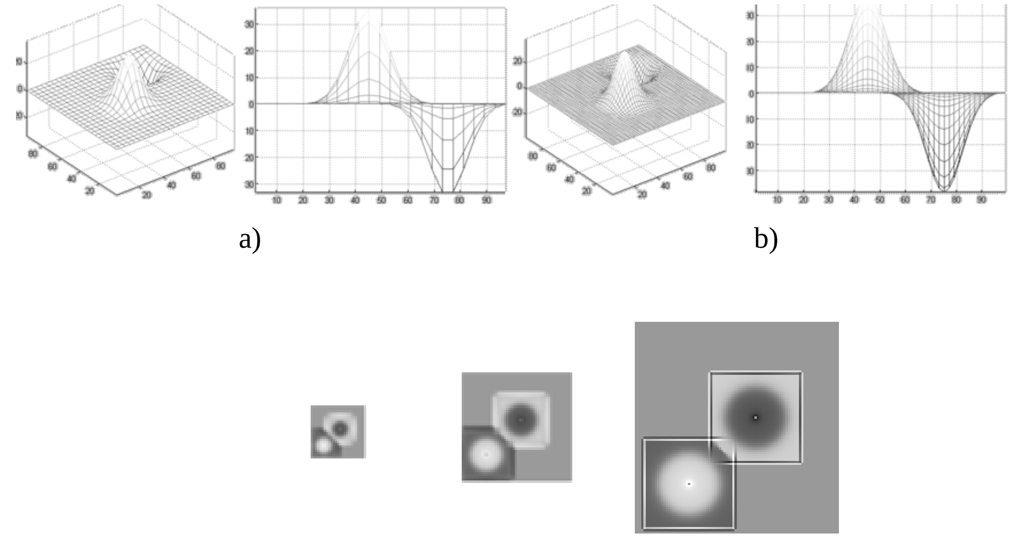
a) b) c)

Figure 16. a) Two scaled version of the same shape. Bigger one (1) is 10 times larger than the other. (Smaller one (2) is barely seen on the lower–left of the bigger one). b) Two S maps are identical (in both S maps a grey scale level is mapped to a distinct real value. So having same intensities on each SI map proves that they have the same real S values) c) C (curvedness) map intensities (i.e. the C values) are different. For the small shape, the curvedness is barely detected. For H and K maps, the effect would be similar to C.

It is seen that both shapes have identical S maps but different C maps. All grey level intensities are calibrated such that, a grey level indicates a distinct real curvature value. Thus, if two maps are of the same grey values, then the real values corresponding to those maps are exactly the same just like in Figure 16.b.

In order to see the resolution (sampling) invariance of the shape index S, we make another test using three different sampled versions of the same shape from the previous figure: one digitized with 25x25 resolution, other with 50x50 and the original surface with 100x100.

Note that the shapes have identical size, which means they occupy the same size in 3D space.



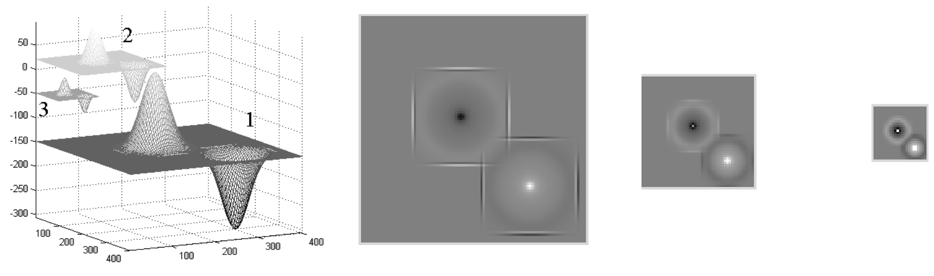
a) b)

c)

Figure 17. a) 25x25 sampled surface. b) 50x50 sampled surface. c) On the left S map for the 25x25 sampled surface is seen. In the middle S map for 50x50 sampled surface is seen. On the right we have the S map for the original 100x100 sampled surface. As the surface resolution gets lower and lower, the S map resolution also changes but the S values (the grey level intensities) do not get affected.

In Figure 17.c we have the S map for the three surfaces. As the surface resolution gets lower and lower, the S map resolution also changes but the S values (the grey level intensities) do not get affected, which means shape index is independent of resolution even though the principal curvatures values are not.

Needless to say that if the same test was carried out for H, K and C maps, the values would have been affected and thus H, K and C are not resolution invariant. However there’s a way to get the same principal curvature values, thus the same H, K and C values for different sampled or scaled versions of the same shape. It can be mathematically proven that if the scale/resolution ratio is held constant for different scaled and sampled versions of the same shape, the principal curvature values also stay the same. Thus, H, K and C may also become scale and resolution invariant. In order to prove this idea practically, we prepare our final test on three different versions of the same shape. In this test the scale and the resolution is modified correspondingly. When the scale is halved, then the resolution is also exactly halved. So the scale/resolution ratio stays constant for each surface.



a) b)

Figure 18. a) The original surface (1) with 100x100 resolution; Surface 2 with 50x50 resolution scaled by ½; Surface 3 with 25x25 resolution scaled by ¼; b) H maps for each surface is depicted. Intensities (e.g. real H values) do not change for different scaled H maps, since scale/resolution ratio is constant.

This test shows that if we keep the scale/resolution ratio constant for a surface patch, the principal curvature values becomes independent of scale and resolution and so the H, K and C values. Thus thresholds we use for H, K and C in any recognition system also become universal if we keep the scale/resolution ratio constant. However this ratio is affected by many things such as: the settings and type of the digitizer (scanner) and even the distance of the object from the digitizer. This ratio must be set to a constant value before any threshold is used. For this purpose in this thesis any 3D scan is re-sampled into a constant scale/sampling ratio of 0.5mm/sample in each *u* and *v* directions.

Furthermore by settings the scale/resolution values constant and by making H, K, and C values independent of scale and resolution, the advantage of S as a natural scale invariant values vanishes. Consequently a proper comparison of HK and SC spaces can be made since both spaces become independent of scale and resolution. The next section makes an introductory comparison of the two curvature spaces and introduces a coupled version of the two.

4.5.2 Rule for Thresholding

As proven in the previous subsection, a constant scale/resolution ratio will make H, K and C thresholds universal. However these universal threshold values depend on the fixed scale/resolution ratio. In our studies the fixed scale/resolution ratio is 0.5mm/sample. The threshold values for H, K and C are 0.03, 0.0009 and 0.03 respectively. If a different scale/resolution ratio will be used; in order to get the same classification results, the threshold should be decided consistent with the formulas below:

Let new scale/sampling ratio be ratio_{ss};

$$\begin{aligned} H_{zero} &= C_{zero} = 0.03 \times \left(0.5 / ratio_{ss}\right) \\ K_{zero} &= \left(H_{zero}\right)^2 \end{aligned} \tag{9}$$

4.6 Comparison of HK and SC Curvature Spaces

There has been an ongoing debate on the advantages and disadvantages of using HK (Mean – Gaussian) or SC (shape index – curvedness) curvatures for object recognition applications. [Cantzler and Fisher 2001] make a comparison of HK and SC curvature descriptions in terms of classification, impact of thresholds and impact of noise levels. They conclude that SC approach has some advantages at low thresholds, complex scenes and dealing with noise. However they calculate the curvatures at the lowest scale, e.g. the given resolution. Scale-spaces of the surfaces or the curvatures are not defined. Another comparative study have been carried out in [Li and Hancock 2004] where using curvature values obtained from the shading in 2D images, HK and SC histograms are created. The comparison results show that SC histograms are slightly more successful in terms of classification. Yet again the tested resolution is the pixel resolution of the 2D image, and the effect of sampling is ignored.

When calculating the H, K and C values, the scale/resolution ratio is highly effective. However due to its scale invariant nature shape index (S) values are independent of the resolution or the scale. Thus it is no wonder that SC methods give better results against HK when the comparison is carried out at an uncontrolled scale/resolution level. As explained in the previous subsection, in order to make H, K and C values scale invariant the scale/resolution ratio must be set to a constant value for the whole database. In addition a scale space of the surface should be constructed so that all features obtained by using H, K, S and C values will also carry the scale level information. Obtaining the absolute scales of these features is a crucial ability for 3D object representation and the only way of achieving such information is constructing the scale-spaces of the surfaces and the scale-space of the curvature values. There have been different attempts at constructing a scale-space of the surface and defining scale-invariant features however to our knowledge, there have been no study on comparison of HK and SC classification capabilities using scale-spaces of the surface in the database.

In this thesis, our main motivation and contribution is to make both theoretical (mathematical) and empirical comparisons of HK (Mean/Gaussian) and SC (Shape Index/Curvedness) curvature descriptions in a scale-space of the targeted surface. By doing this we also aim at comparing the scale parameter obtained by using HK and SC, on a local surface element. For this purpose we calculate the scale spaces of the given surfaces and the curvatures. The details of this part are given in the next chapter. Then we make empirical comparison on the classification results of scale invariant HK and SC for the Stuttgart database. This comparison is given in the results chapter. Before concluding this chapter, a mathematical comparison of HK and SC spaces is given below.

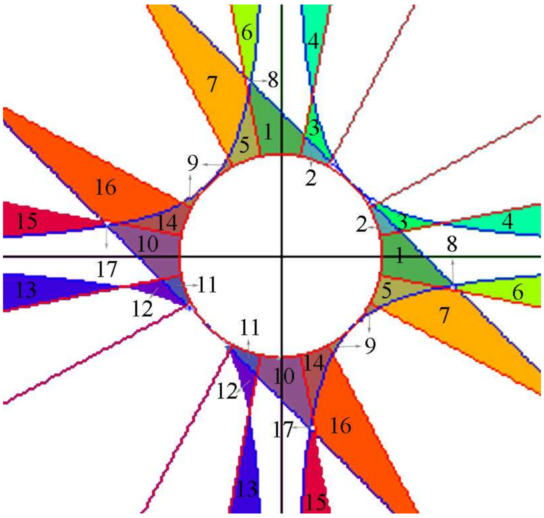


Figure 19. The difference in region description of HK and SC spaces over the (κ_1, κ_2) plane is depicted. Each number designates a region which is described in Table 4. The shape index thresholds are ±5/8, ±3/8, ±3/16, as it is for Figure 4. The other thresholds are selected such that $K_{zero} = H^2_{zero}$ and $C_{zero} = H_{zero}$.

4.6.1 Mathematical Comparison of HK and SC Curvature Spaces

In this chapter mathematical analysis of HK and SC curvature spaces are given over the (κ_1, κ_2) plane. Eight fundamental types of regions are indicated for both of the methods. Since both HK and SC values are obtained from (κ_1, κ_2) values by using different mathematical operators, the eight fundamental regions have slight differences for the two curvatures spaces. Actually these slight differences play the decisive role on their success for object classification. In Figure 19, the differences in region definition over (κ_1, κ_2) planes are seen. Each region is indicated by a number, which is listed in Table 4.

Table 4. The definition of the regions seen in Figure 19

Reg. No.	HK Definition	SC Definition	Mathematical Definition
1	Plane	Convex Parabolic	$H<\|H_{zero}\| \cap K<\|K_{zero}\| \cap S\in [-5/8,-3/8] \cap C>C_{zero}$
2	Plane	Convex Elliptic	$H<\|H_{zero}\| \cap K<\|K_{zero}\| \cap S\in [-1,-5/8] \cap C>C_{zero}$
3	Convex Parabolic	Convex Elliptic	$H>+\|H_{zero}\| \cap K<\|K_{zero}\| \cap S\in [-1,-5/8] \cap C>C_{zero}$
4	Convex Elliptic	Convex Parabolic	$H>+\|H_{zero}\| \cap K>+\|K_{zero}\| \cap S\in [-5/8,-3/8] \cap C>C_{zero}$
5	Plane	Saddle Valley	$H<\|H_{zero}\| \cap K<\|K_{zero}\| \cap S\in [-3/8,-3/16] \cap C>C_{zero}$
6	Saddle Valley	Convex Parabolic	$H>+\|H_{zero}\| \cap K<-\|K_{zero}\| \cap S\in [-5/8,-3/8] \cap C>C_{zero}$
7	Hyperbolic	Saddle Valley	$H<\|H_{zero}\| \cap K<-\|K_{zero}\| \cap S\in [-3/8,-3/16] \cap C>C_{zero}$
8	Hyperbolic	Convex Parabolic	$H<\|H_{zero}\| \cap K<-\|K_{zero}\| \cap S\in [-5/8,-3/8] \cap C>C_{zero}$
9	Plane	Hyperbolic	$H<\|H_{zero}\| \cap K<\|K_{zero}\| \cap S\in [-3/8,+3/8] \cap C>C_{zero}$
10	Plane	Concave Parabolic	$H<\|H_{zero}\| \cap K<\|K_{zero}\| \cap S\in [+3/8,+5/8] \cap C>C_{zero}$
11	Plane	Concave Elliptic	$H<\|H_{zero}\| \cap K<\|K_{zero}\| \cap S\in [+5/8,+1] \cap C>C_{zero}$
12	Concave Parabolic	Concave Elliptic	$H<-\|H_{zero}\| \cap K<\|K_{zero}\| \cap S\in [+5/8,+1] \cap C>C_{zero}$
13	Concave Elliptic	Concave Parabolic	$H<-\|H_{zero}\| \cap K>+\|K_{zero}\| \cap S\in [+3/8,+5/8] \cap C>C_{zero}$
14	Plane	Saddle Ridge	$H<\|H_{zero}\| \cap K<\|K_{zero}\| \cap S\in [+3/16,+3/8] \cap C>C_{zero}$
15	Saddle Ridge	Concave Parabolic	$H<-\|H_{zero}\| \cap K<-\|K_{zero}\| \cap S\in [+3/8,+5/8] \cap C>C_{zero}$
16	Hyperbolic	Saddle Ridge	$H<\|H_{zero}\| \cap K<-\|K_{zero}\| \cap S\in [+3/16,+3/8] \cap C>C_{zero}$
17	Hyperbolic	Concave Parabolic	$H<\|H_{zero}\| \cap K<-\|K_{zero}\| \cap S\in [+3/8,+5/8] \cap C>C_{zero}$

In order to make the two spaces similar to each other, the zero-thresholds are arranged such that of the threshold lines and curves of HK and SC spaces are tangent to each other, when it is possible. Similar to [Cantzler and Fisher 2001]'s approach, H_{zero} and C_{zero} are taken equal, so that the line $H=H_{zero}$ is tangent to the circle $C=C_{zero}$. In addition $K_{zero} = H^2_{zero}$, so that the hyperbola $K=K_{zero}$ is tangent to the circle $C=C_{zero}$. Although, the regions defined by the curvature spaces become very similar after these arrangements, there are still fundamental differences in region definitions.

The planar region definition for SC space is a circle centred on the (κ_1, κ_2) plane origin and thus it is symmetric. However the planar region definition for HK space is different and not symmetric according to κ_1= κ_2 line, although it is symmetric according to κ_1=- κ_2 line. For this reason for the planar region defined by HK space, is defined much differently by the SC space (regions 1, 2, 5, 9, 10, 11, 14).

Different definitions of elliptic and parabolic regions for the two curvature spaces also get diverse when the principal curvature values get larger. There are regions defined as parabolic for one curvature space and elliptic for the other curvature space (regions 4, 6, 13, 15). The parabolic region for HK curvature space get narrower for larger principal curvature values, while it gets wider for the SC curvature space. There are also other regions (regions 3, 7, 8, 12, 16, 17) where the definitions are ambiguous.

4.6.2 HK&SC Coupled Curvature Space

On the other hand there are regions where the region definitions are the same for the both curvature spaces. For this reason in Figure 19, the regions for which both HK and SC curvature spaces definitions are the same are seen. We name this combined space of HK and SC curvatures, the HK&SC coupled curvature space. The mathematical definitions are given in Table 5.

This new curvature space do not cover all the regions over the (κ_1, κ_2) plane. The regions which are ambiguous, in other words which are classified different in HK and SC spaces, are excluded from the classification. So if a local surface patch is classified as one of the regions in Table 5, then they are classified same in both HK and SC curvature spaces.

It is very difficult to decide which classification is best by just making a theoretical comparison among the curvature spaces. It is difficult to conclude on the best method without experimentation. One region may be classified as planar by HK curvature space, convex parabolic as SC curvature space, and may not be classified by the HK&SC coupled curvature space (region 1 in Figure 20). However the effect on classification success can only be understood by experimental study. In this thesis we compare the first two curvature spaces for object recognition purposes. In addition we do this using the scale-space of the surface and curvatures. The next chapter gives the details on constructing these scale-spaces.

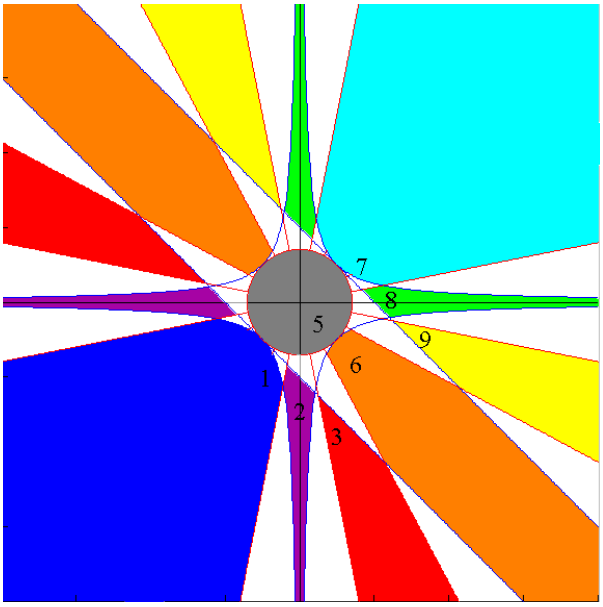


Figure 20. The HK&SC coupled space classification in (κ1, κ2) plane. The colours and numbers are correspondent with Table 5.

Table 5. HK&SC coupled space Classification. Colours are consistent with the previous figures

Convex (Elliptic) (1)	$H<-\lvert H_{zero}\rvert \cap K>+\lvert K_{zero}\rvert \cap S\in[+5/8,1] \cap C>C_{zero}$
Convex (Parabolic) (2)	$H<-\lvert H_{zero}\rvert \cap K<\lvert K_{zero}\rvert \cap S\in[+3/8,+5/8] \cap C>C_{zero}$
Saddle Ridge (3)	$H<-\lvert H_{zero}\rvert \cap K<-\lvert K_{zero}\rvert \cap S\in[+3/16,+3/8] \cap C>C_{zero}$
Planar (5)	$H<\lvert H_{zero}\rvert \cap K<\lvert K_{zero}\rvert \cap C<C_{zero}$
Hyperbola (6)	$H<\lvert H_{zero}\rvert \cap K<-\lvert K_{zero}\rvert \cap S\in[-3/16,+3/16] \cap C>C_{zero}$
Concave (Elliptic) (7)	$H>+\lvert H_{zero}\rvert \cap K>+\lvert K_{zero}\rvert \cap S\in[-1,-5/8] \cap C>C_{zero}$
Concave (Parabolic) (8)	$H>+\lvert H_{zero}\rvert \cap K<\lvert K_{zero}\rvert \cap S\in[-5/8,-3/8] \cap C>C_{zero}$
Saddle Valley (6)	$H>+\lvert H_{zero}\rvert \cap K<-\lvert K_{zero}\rvert \cap S\in[-3/16,+3/16] \cap C>C_{zero}$

CHAPTER 5

SCALE-SPACE OF CURVATURES

The scale-space representation is the formal theory of handling an *n*-dimensional signal at different scales, such that a one-parameter family of the smoothed versions of that signal is constructed. In the next subsection, the use of scale space theory in the field of computer vision is briefly summarized.

5.1 Scale-Space Theory in Computer Vision

Typically, an object contains structures at many different scales. Thus a true representation may only be constructed using the information obtained from different scales of the object. In this section the formal definition of scale in computer vision is given. For further reading, reader should refer to Tony Lindeberg's renowned book on scale-space theory in computer vision [Lindeberg 1994].

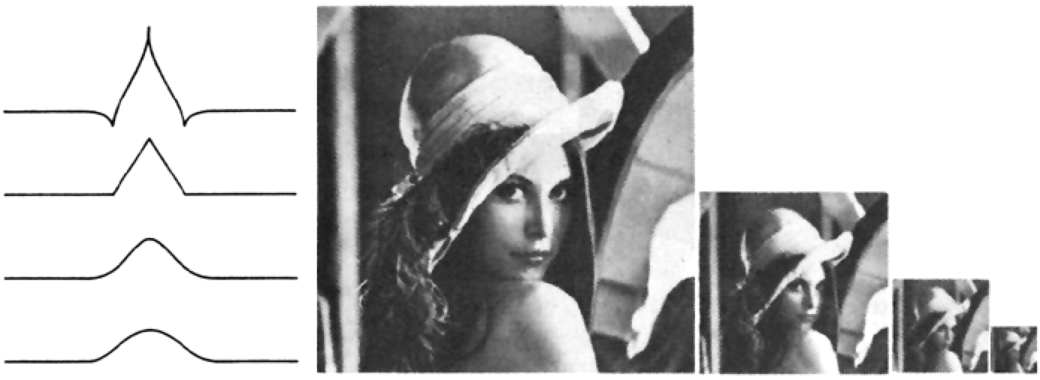
Computer vision deals with deriving meaningful or useful information from 2D, 2.5D or 3D images. The digital equipments used to obtain such images provide this information as discrete signals, which have certain sampling rates (resolutions). However the useful information within the image resides independent of this sampling rate. Scale-space theory provides some directions to overcome this dependency by constructing different scales levels of the image. In this manner, scale levels are constructed as different stages of sampling of the original image; so that an operator will deal with different useful features residing in different scale levels.

The most common method to obtain higher scales for an image is pyramiding. A pyramid representation of a signal is a set of successively smoothed and sub-sampled representations of the original signal organized in such a way that the number of pixels decreases with a constant factor from one layer to another [Lindeberg 1994]. The next section summarizes the most common pyramiding method: Gaussian pyramiding.

5.2 Gaussian Pyramiding

Similar to other low-pass pyramiding approaches, Gaussian pyramiding is a type of multi-scale signal representation, in which a signal or an image is subject to repeated Gaussian smoothing; so that a linear scale space of that signal is constructed.

In [Burt and Adelson 1983] Gaussian pyramids of 2D images are constructed by reducing an image into its half resolution (Figure 21). For this purpose "*Reduce*" and "*Expand*" operations are defined such that at each "*Reduce*" operation the data is smoothed and down sampled into half, and similarly at each "*Expand*" operation the data is up-sampled into double resolution by Gaussian smoothing [Burt and Adelson 1983].



a) b)

Figure 21. a) Gaussian pyramiding over a 1D signal, (higher scale levels towards the bottom). b) Gaussian Pyramiding on a 2D image. In each step, the image is both down-sampled and smoothed (higher scale levels towards the right) [Burt and Adelson 1983].

The main motivation behind reducing a signal is to obtain down-sampled version of that signal, so that the size of an applied operator becomes independent of the resolution of that signal. For example, if a 3x3 kernel is convolved over the image, only features fitting inside 3x3 region could be detected. But if the kernel is applied to all reduced levels, larger features can also detected using the same operator.

5.2.1 Gaussian Pyramiding over 3D Surface Scans

In this thesis, Gaussian pyramiding is applied to 3D surface scans and 3D curvatures in order to obtain scale-space representations of 3D objects. However it is not a straightforward procedure to apply these operations to 3D, since there is an ambiguity in the definitions of scale and resolution for 3D scans. Before understanding the “scale” and “resolution” (or the sampling rate) of a 3D scan, it is beneficial to examine their meanings in 2D.

For 2D images, the notion of scale is the same as resolution. In other words, the size of the signal is calculated by its sampling rate because a metric size within an image is not defined. For this reason, when a scale-space of a 2D signal is constructed, at each scale level the image is solely sub-sampled.

On other hand, the sampled points carry metric information for 3D scans. In other words independent of the sampling rate of the 3D scan, the metric distances between points on the signal are absolute. For this reason, two scans of the same object taken from different distances, will sample the object in different resolutions; however the object surface within the 3D scan will carry the same absolute metric size. This little detail may seem insignificant; however it plays a profound role in construction of scale-space for 3D scans.

Gaussian kernel and its derivatives are singled out as the only possible smoothing kernels [Lindeberg 1994]. Pyramid approach for multi-resolution is usually chosen because in Gaussian pyramiding, image size decreases exponentially with the scale level and hence also the amount of computation required to process the data. But it also has some drawbacks such as a coarse quantization along the scale direction which makes it algorithmically hard to relate image structures across scales [Lindeberg 1994].

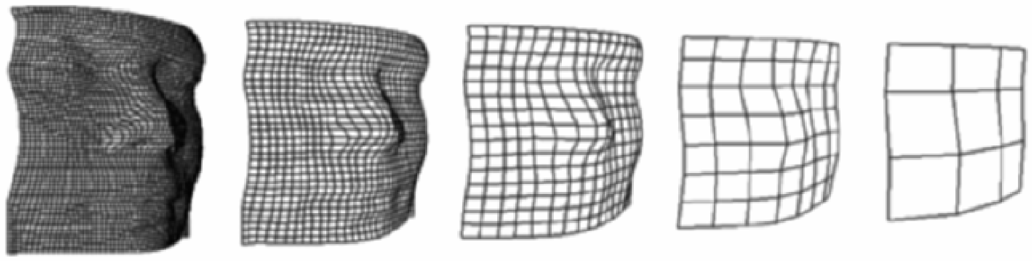


Figure 22. Gaussian Pyramiding on 3D scans is seen. In each scale level, the sampling rate is halved, but the metric distances are preserved.

As seen Figure 22, a Gaussian Pyramid of a 3D scan is depicted. At each scale level the sampling rate is halved as expected. But the metric distances within the image stay unaffected.

5.2.2 Invalid Points in 3D Surface Pyramids

As it was discussed thoroughly in Chapter 2, there are invalid points on 3D scans, which are either background or simply invalid since they could not been acquired properly by the scanner. For most 3D data processing methods, these invalid points are simply ignored and calculations are carried out using only the valid points. However if a pyramid of a 3D range image, which contains a group of invalid points, is to be constructed; these invalid points should be handled properly.

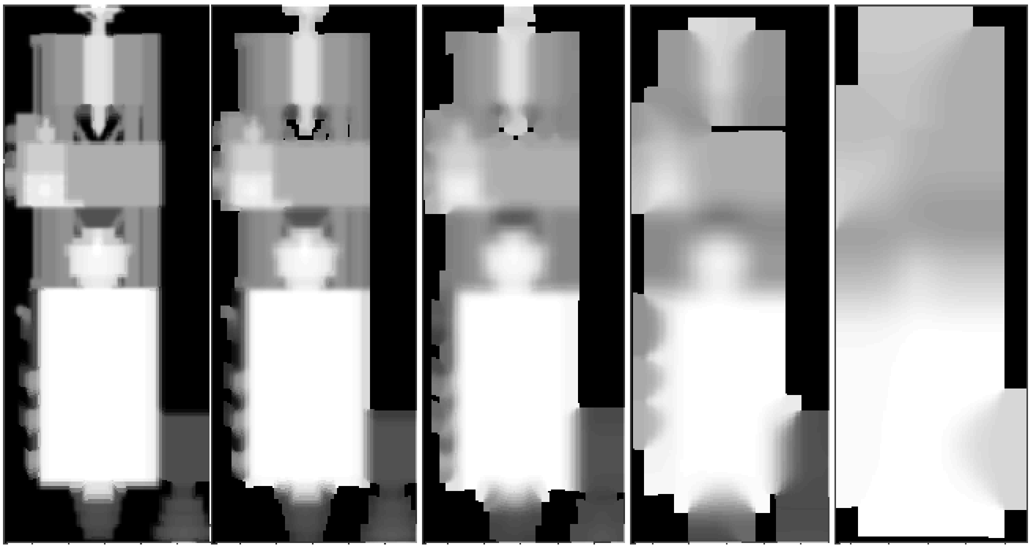


Figure 23. The scale levels of a depth image constructed by Gaussian pyramiding are seen. The black regions are the invalid points. Around the valid point boundaries the smoothing filter avoids blending with invalid points so that the sharp boundaries are preserved.

When constructing a pyramid for a 2D intensity image as in Figure 21.b, the Gaussian operator is convolved throughout the image with no exceptions. However in order to convolve a Gaussian filter over a depth image, including invalid points; it is possible that one might experience difficulties in boundary regions where valid and invalid points are next to each other. In these occasions, the segments of the Gaussian filter, which corresponds to invalid points, should be omitted. This way the true shape of the object may be preserved in higher scales. In Figure 23, the scale space of a surface is constructed using Gaussian pyramiding. Around the valid point boundaries the smoothing filter avoids blending with invalid points. This way, sharp boundaries of an object can be preserved.

A similar approach was carried out by [Lo and Sieberta 2009] in order to avoid false key points which would be resulted as a consequence of applying a standard Gaussian mask over the sharp boundaries on the range image. Instead they applied a Gaussian-tapered segmentation mask in order to isolate the area of interest while avoiding sharp boundaries.

5.3 Scale Space of Curvatures

In this section, the scale levels of a depth signal are analyzed. In the previous figure, scale levels of a range image were given. In Figure 24, the same range image is seen from a different angle. It is clearly seen that, as we go up the scale axis, the smaller elements vanish and only the larger elements reside. This is the basic motivation behind constructing a scale space.

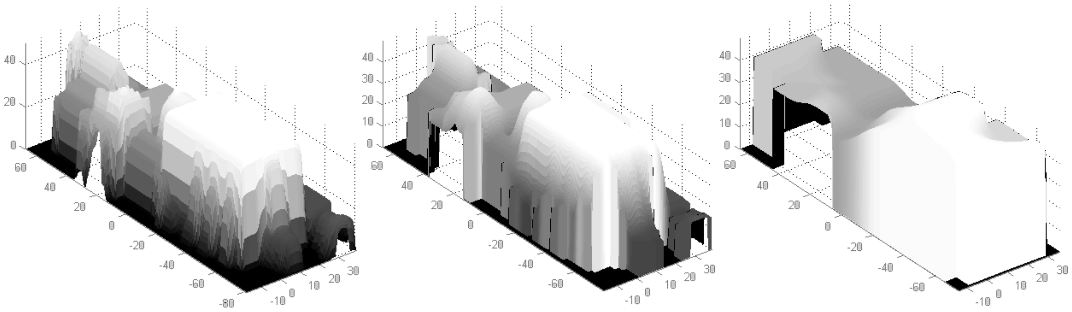


Figure 24. The smaller elements vanish as we go up the scale axis.

The normal directions of each scale level are shown in Figure 25. In this figure, red values correspond to x components, green values correspond to y components and blue components correspond to z components of the surface normals. It is also seen from the normal scale-space that smaller elements vanish in higher scales. The thin edge which is denoted by 1 in Figure 25, is visible in lower scales but vanishes in higher scales, since this element is relative small.

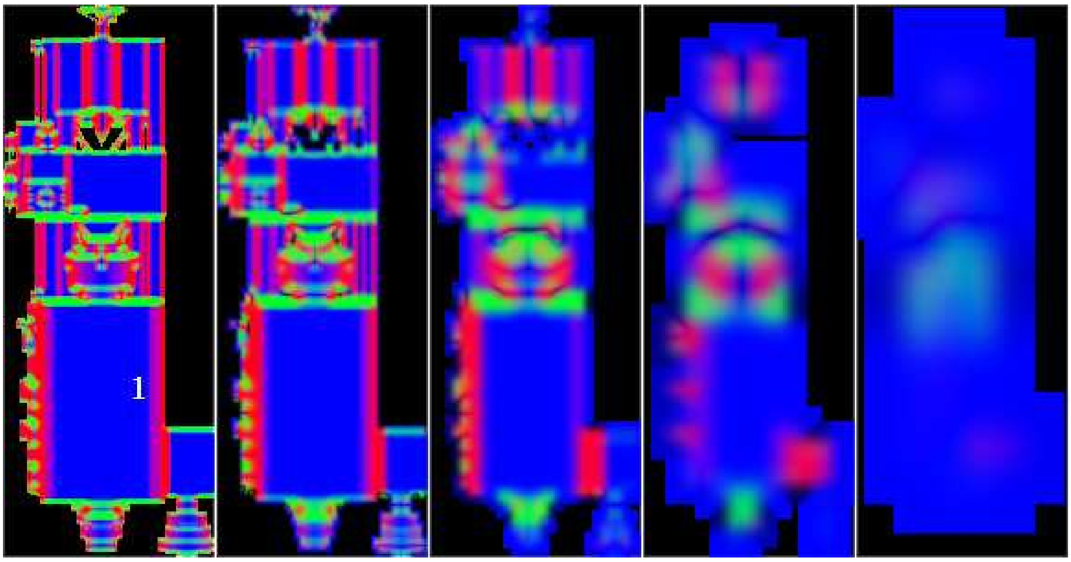


Figure 25. The scale-space of the surface normals. Red values correspond to x components, green values correspond to y components and blue components correspond to z components of the surface normals.

The benefit of constructing a scale space of a range image is more clearly seen, if the curvatures obtained from different scale levels are analyzed. The next subsections emphasize on constructing different curvature scale-spaces.

5.3.1 Curvatures in Scale-Space

As explained in Chapter 3, the curvatures are calculated using surface gradients. Thus they give basic information on surface behaviour. In order to calculate surface gradients analytically, explicit surface functions may be used. However in real world applications, the 3D surfaces are digitized into sampled points and there’s no global explicit function of the surface. Hence, the surface gradients are calculated within a neighbourhood of sampled points. For this reason the calculated curvatures are local approximations, which are valid at a certain scale. Therefore the curvatures are calculated for each scale level, and the curvature scale-spaces are constructed.

In this section different curvature values obtained from different scale levels are depicted, so that the general idea of curvature scale-space can be properly comprehended. First, in Figure 26, the scale space of mean curvature values are seen. The concave regions, which have positive mean curvature values, are painted in red, where convex regions with negative mean curvature values are painted in blue. For both regions the magnitude of the curvature is demonstrated by colour intensity.

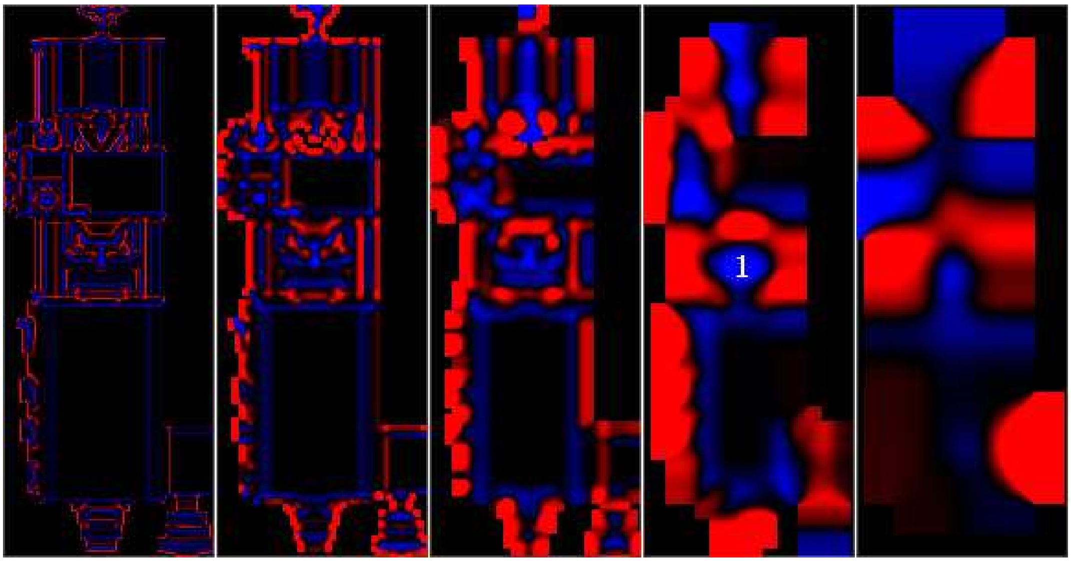


Figure 26. The scale-space of mean curvature values. The concave regions which have positive mean curvature values, are painted in red, where convex regions with negative mean curvature values are painted in blue.

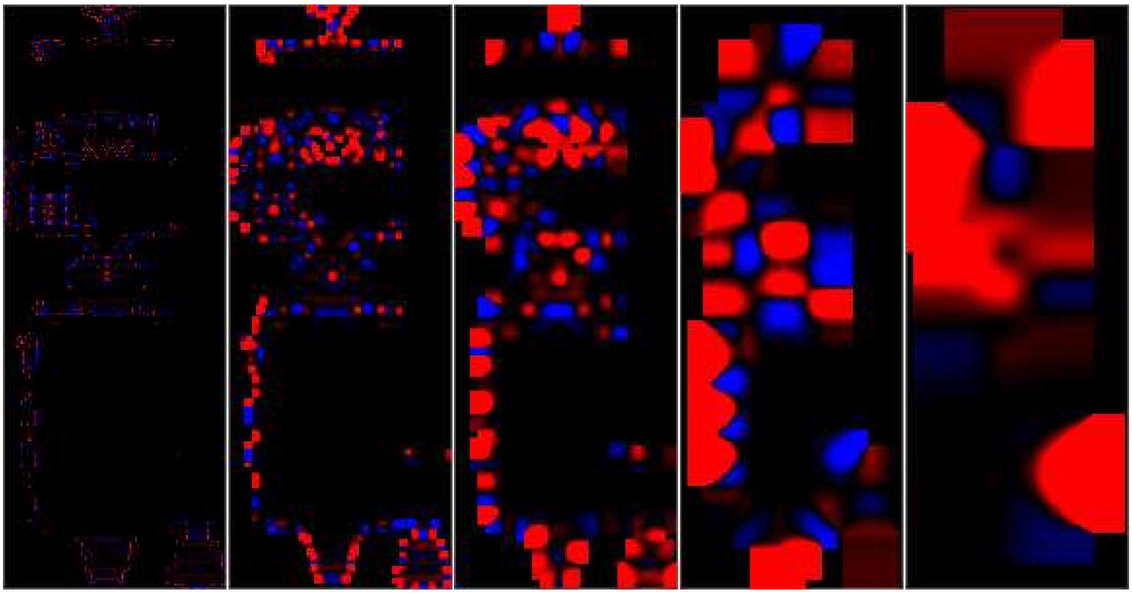


Figure 27. The scale-space of Gaussian curvature values. The parabolic regions with positive Gaussian curvature values are painted in red colour, where the hyperbolic regions with negative Gaussian curvature values are painted in blue colour.

As seen from the figure, the convex region denoted by number 1 is designated as a peak in higher scales since this element is relative large for the given resolution of the 3D scan.

Similarly if we examine the Gaussian curvature values, again the larger features are observed in higher scales. In Figure 27, the Gaussian curvature values obtained from different scales of the surface are depicted. The parabolic regions with positive Gaussian curvature values are painted in red, where the hyperbolic regions with negative Gaussian curvature values are painted in blue. For both regions the magnitude of the curvature is demonstrated by colour intensity.

Moreover, when the shape index values obtained from different scale levels of the surface are examined, the affect of scale-space is clearly seen. Since shape index value is capable of classifying the surface into the fundamental surface types (except planes), the scale-space of the shape index values clearly demonstrate the scale coordinates of the surface features. . In Figure 28, the shape index values obtained from different scales of the surface are depicted. The shape index values and colour is mapped according to Table 2 in Chapter 3.

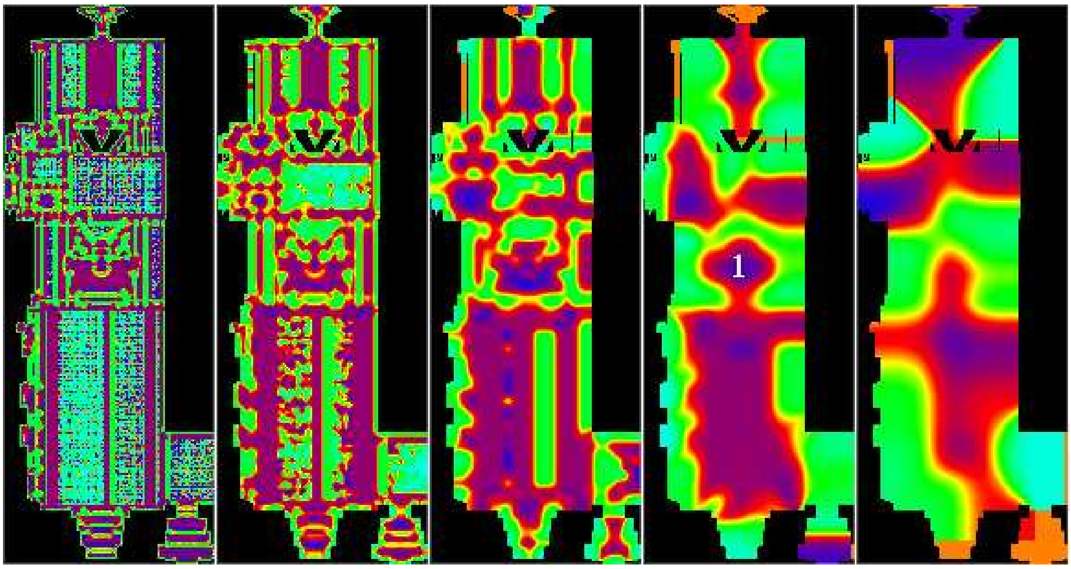


Figure 28. The scale-space of shape index values. The shape index values and colour is mapped according to Table 2 in Chapter 3.

As seen from the figure, the convex region denoted by number 1 is designated as a peak in higher scales and is not existent in lower scales.

Finally if we examine the curvedness values obtained from different scale levels of the surface, we clearly extract the planar regions in the scale-space. Curvedness values can be used to detect planar regions, since regions having sufficiently small curvedness values are defined as planes. In Figure 29, the gray level intensities designates the curvedness values, where zero curvedness value corresponds to black.

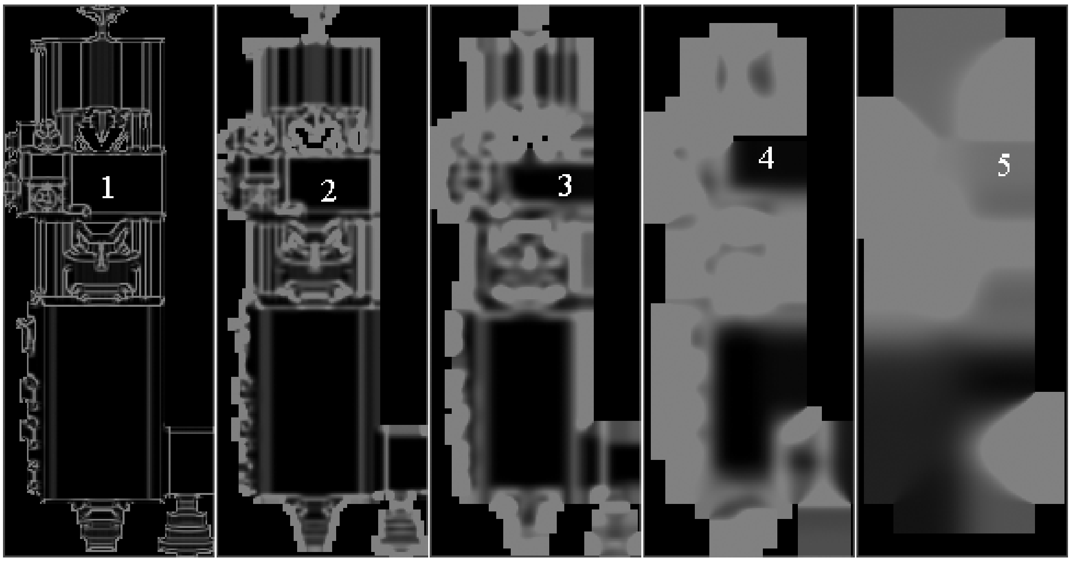


Figure 29. The scale-space of curvedness values. The gray level intensities designate the curvedness values, where zero curvedness value corresponds to black.

The region denoted by numbers 1, 2, 3 and 4 in different scale levels corresponds to a planar region on the surface and it is designated as plane in the first four scale levels in the curvedness scale –space. However in the fifth curvedness scale-space level is it not designated as plane, since in this scale, the region is designated as pit.

In this chapter the concept scale-space is defined for 3D surfaces. Moreover, it is clearly shown that larger features can only be extracted using the scale-space of 3D surfaces and curvatures. In the next chapter, the details of feature extraction from the scale-space of 3D surfaces and curvatures are presented.

CHAPTER 6

FEATURE EXTRACTION AND OBJECT REPRESENTATION

In this chapter, the proposed method to extract scale and orientation invariant features using scale-spaces of 3D surfaces and curvatures is explained. There are previous methods which focus on extracting features from scale-space of different types of signals [Lowe 2004]. For this reason, certain aspects of the proposed method, such as scale-space localization are given in comparison with the previous scale-space techniques.

6.1 Construction of the UVS Volume

As it was stated in the previous chapter, estimating surface feature labels at the given resolution of the original surface restricts us to find the features only at the lowest scale (the given resolution). In order to detect other curvatures which have higher scales, a scale space representation must be introduced. For this purpose, the Gaussian pyramid of the input surface may be generated. It should be remembered that Gaussian pyramiding does not change the absolute size of the model surface, but the resolution of the surface halves as we go up the scale space. In other words, the absolute metric information is kept.

Subsequently the curvature values for each scale are calculated and H, K, S and C maps for each pyramid level are obtained. Towards the higher levels of a scale-space representation, the smaller surface elements vanish and bigger elements reside. For example in Figure 30, eye pits and chin peak vanish at the third pyramid level but nose peak and nose saddle still reside.

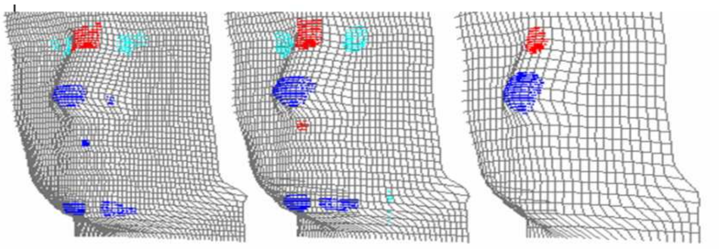


Figure 30. Three levels from the HK pyramid of a face model surface.

Subsequently, the higher levels of each curvature pyramid is up-sampled to the original size by using “Expand” (the inverse of “Reduce”) as explained in Chapter 4. After this expansion, any label on the sth level of the pyramid will widen 2^s times in resolution. Thus all levels of the curvature pyramid will have the same resolution. An example of such an equalized curvature scale-space was introduced in the previous chapter. Putting each level of the expanded pyramid on top of each other, we obtain a 3D volume which we call as UVS space where u and v are the surface dimensions and s is the scale dimension. Afterwards suing different curvature UVS volumes (all of which have the same resolution), the voxels are classified as explained in Chapter 3. For instance, using H and K values, HK classification is carried out and for each voxel, a feature label (or type) is obtained (Figure 31).

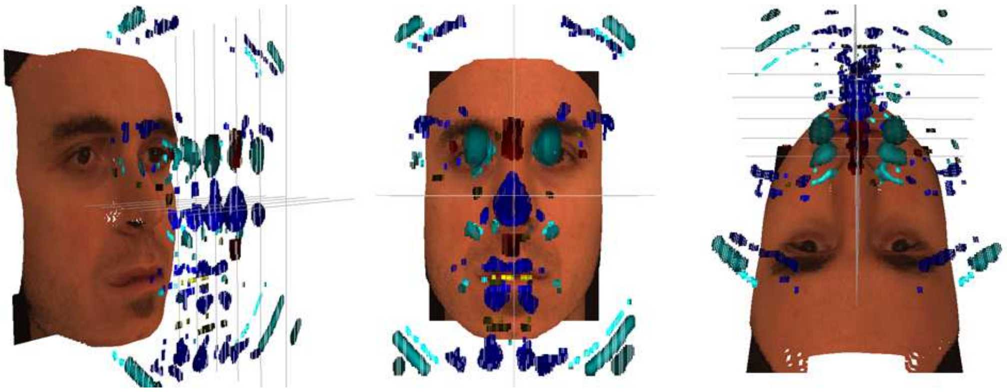


Figure 31. The HK UVS volume after morphological operations printed above the surface patch. The features are indicated with their surface shapes as colours (blue for peaks, cyan for pits, red for saddle ridges, etc.).

For each classification type (HK or SC) a different classified UVS volume is obtained. We refer to them as the HK UVS volume or the SC UVS volume. The process of UVS volume construction is depicted in Figure 32.

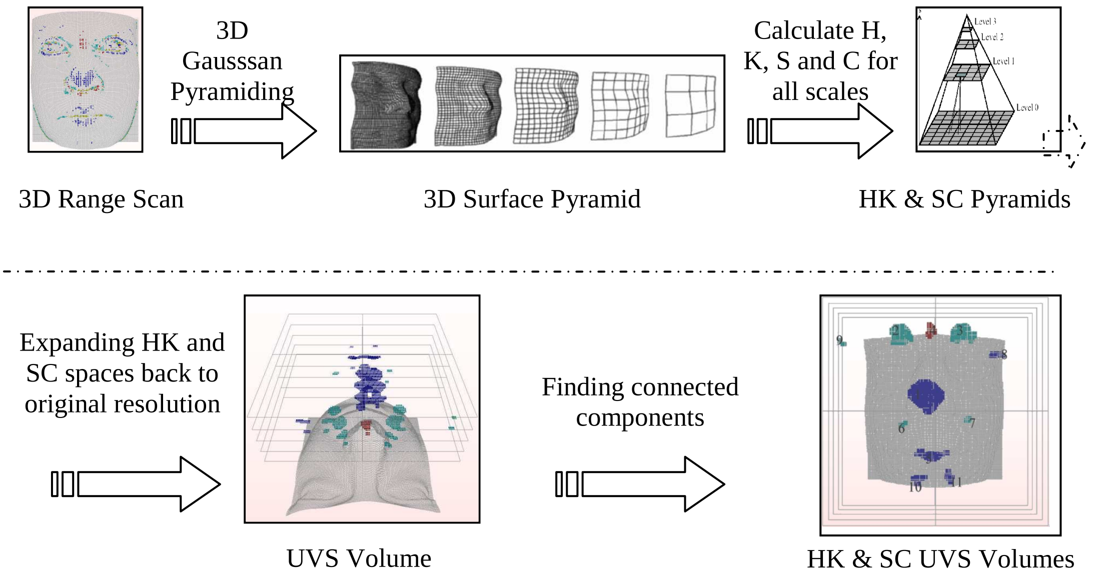


Figure 32. UVS Volume Construction.

Within the UVS Volume, each voxel is classified according to its curvature value. As it was stated in Chapter 3, the voxels are classified by thresholding. Unsurprisingly some the curvature values are much higher so that their value above the threshold is higher than the others. For this reason, a weight value is assigned to each voxel using the amount of curvature values that exceed the thresholds. The location and scale of a feature in the UVS volume are estimated using these weights. For the HK UVS volume, the weights are taken as the 2nd norm of the absolute differences of the curvature values from the applied threshold values:

$$w_{i,j} = \left((\mathbf{H_{i,j}} - \in_{\mathbf{H}})^2 + (\mathbf{K_{i,j}} - \in_{\mathbf{K}})^2 \right)^{\frac{1}{2}} \qquad (10)$$

In Figure 33, the weights are demonstrated within the HK UVS volume. The bigger the weight is the lighter the colour becomes. For example, the weights of the eye pit feature can be observed from its colour which changes from light cyan at the smaller scales to dark cyan at the larger scales.

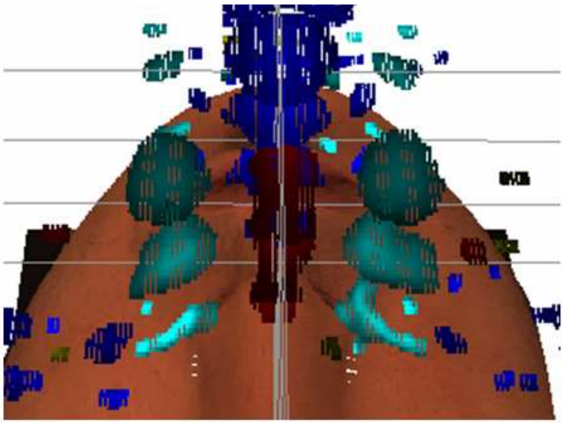


Figure 33. The weights are demonstrated in the HK UVS volume. The bigger the weight is the lighter the colour becomes

For the SC UVS volume, the weights are simply taken as the curvedness (C) values.

These weights can very-well be calculated in different manners. In the next subsections, we thoroughly discuss the reasons why we have selected equation, since it profoundly affects scale-space localization.

6.2 Extraction of Features

In order to extract scale invariant features (features with their scale information) from a UVS Volume, several steps are followed. First, each voxel of a UVS volume is checked for its similarity within its 10 neighbours (8 at the same scale level, 2 up and down through scale). If all labels of the neighbours have the same surface shape label (pit, peak etc.), the centre voxel continues to carry its label. Otherwise it becomes a blank voxel. After voxel relabeling, a single opening (erosion and dilation respectively) operation is applied where 3×3×3 sized kernels are used. Then we extract connected labels in the UVS space. Finally, each connected component represents a feature on the surface which will later become a node of a topology graph. In Figure 34, it is clearly seen that a feature element, for example the handle of the screwdriver, has components at a number of successive scale layers (the orange hyperbolic surface patch).

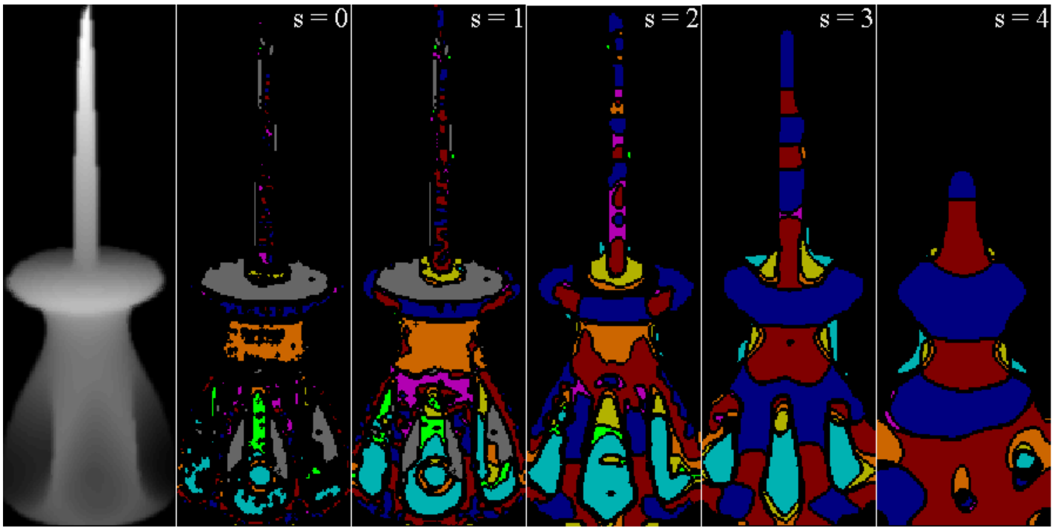


Figure 34. Labelled layers of UVS space constructed by HK values where scale level is increasing from left to right. The original surface level is indicated by “S=0”. Labels are given by colours. (peak: blue, saddle ridge: red, convex cylinder: purple, pit: cyan, saddle valley: yellow, concave cylinder: green, hyperbolic: orange, plane: gray.)

Inside the classified scale-space, each connected component consists of the same type of voxels and is considered as a feature element on the surface. This particular classification type such as pit, peak, and plane; is considered as the type of that feature element (t_i). The total number of voxels inside the connected component represents the feature’s volume (v_i).

When the curvature scale spaces are calculated, the corresponding scale-space of surface coordinates and scale-space of surface normals are also constructed. Thus, for each connected component, using the corresponding voxels within the scale space of surface coordinates; the centre of mass of the connected component is also calculated and called as the positional centre of the feature (x_i). Remember that this centre of mass is calculated by averaging the voxels using the corresponding weights calculated in (9).

Similarly using the scale-space of the surface normals, a weighted average of the normal direction for each connected component can be found ($\mathbf{n}_i$). This vector designates the average orientation of the feature in 3D world.

In addition, since a connected component may have different number of elements in different scales, a weighted average of the scale value is calculated for each connected component as the actual scale of that feature (s_i).

Although a connected component may have various numbers of elements in different scale levels, it has the biggest number of elements on the scale which is closest to its actual scale. The area (A_i) of the connected component at this layer is used to calculate the radius (r_i) (10) which also defines the size of that fundamental element.

$$r = \sqrt{A / 4 \cdot \pi} \tag{10}$$

Finally, for each feature element extracted from the surface, the following attributes are obtained: the type (t_i), the volume (v_i), the positional centre of mass ($\mathbf{x}_i$), the orientation vector ($\mathbf{n}_i$), the scale (s_i) and the size (r_i) (Figure 35).

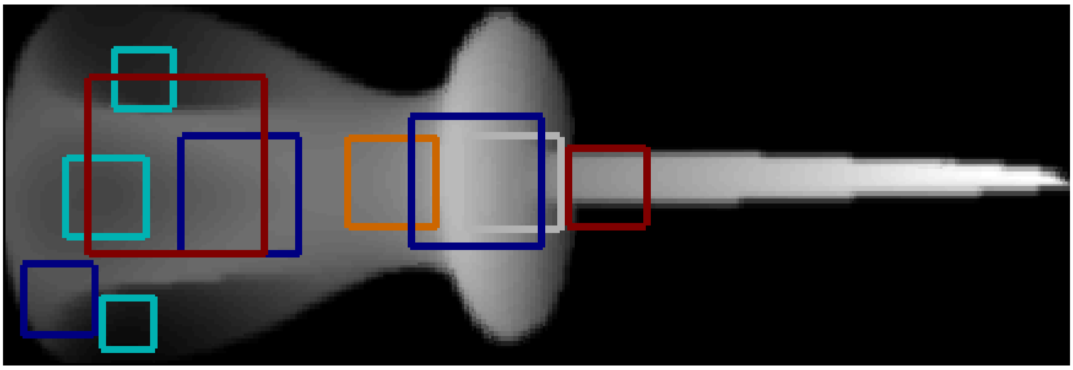


Figure 35. Ten largest extracted features are shown as squares where the feature center (xi) is given by the square center, the feature size (radius) (ri) is given by the square size and the feature type is given by its colour.

6.3 Scale-Space Localization of the Features

Since we use Gaussian smoothing to construct the scale-space volume, the relation between the smoothing kernel size (σ) and the scale dimension (s) between two scale layers (A and B) of the UVS volume is as follows:

$$\frac{\sigma_B}{\sigma_A} = 2^{(s_A - s_B)} \tag{11}$$

This relation is verified on synthetic Gaussian surfaces having different scales as follows. Figure 36 shows UVS spaces for three synthetically generated unit volume Gaussian surface models. Each Gaussian surface has the half standard deviation of the one on its right, i.e. $\sigma_A = \sigma_K$, $\sigma_B = 2*\sigma_K$, $\sigma_C = 4*\sigma_K$.

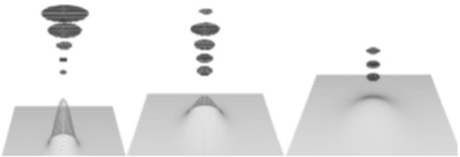


Figure 36. Three unit volume Gaussian surfaces at different scales (σA= σK, σB= 2*σK, σC= 4*σK) and their respective UVS volumes.

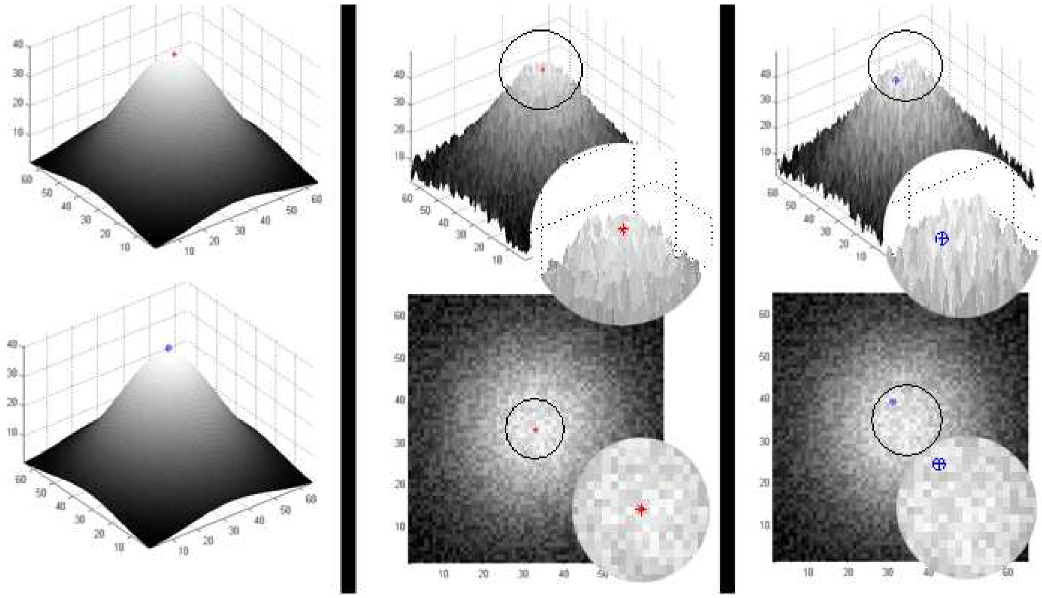
Then according to (3) the scale values for these Gaussian surfaces must have the following relation:

$$\sigma_B / \sigma_A = 2 = 2^{(S_A - S_B)} \rightarrow S_A - S_B = 1 \tag{12}$$

After computing the weighted averages we observe that the estimated locations are the centers of the synthetic Gaussians and scales satisfy the relation given in (4) where SA = 3.86, SB = 2.80 and SC = 1.92. These values are consistent with Equation (4).

As stated by [Lindeberg 1994], the scale, at which a scale space blob assumes its maximum normalized grey-level blob volume over scales, is likely to be a relevant scale for representing that blob. In order to do so, the scale should be normalized. However, in 3D data where we actually know the metric measurements, all data is normalized if sampling is the same for all and in that case normally the scale layer for which the blob extends the most can be taken as the scale of that blob. Based on this idea, we did not take the maximum extending layer but a weighted average over the layers so that we could interpolate between the layers which helps us in reaching a better resolution in scale dimension.

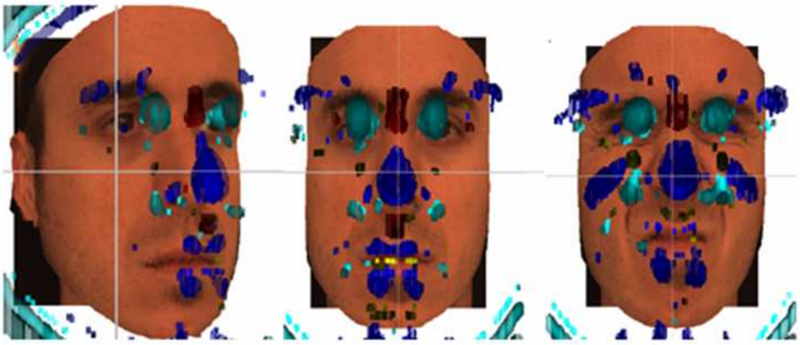
The localization of the features in the UVS volume and on the surface is crucial. This localization should be robust to noise and any type of transformations. For example in [Lowe 2004], the SIFT descriptor is sought in a scale space of different octaves, where all local maxima (or minima) is selected as a feature. Instead of using the weighted averages, a similar approach to [Lowe 2004] may be applied to our method where a single maximum for each connected component in the UVS space is found. However, as it is seen in Figure 37, this approach will fail under noise or in complex scenes. Imagine we have a simple surface with a single peak and its noisy version. When the centre of the peaky feature is sought over the surface with noise, it is seen that a local maximum value inside a connected component may divert the centre from its original position although the surface is smoothed in higher scales (Figure 37.c). Our method localizes the features correctly (Figure 37.b) and is robust to noise.



a) b) c)

Figure 37. Localization of the peak feature (a) using both methods (top: weighted average, bottom: single maximum). b) For the ideal surface both methods localize the peak feature correctly. c) For the noisy surface, the feature is still correctly localized using weighted averages (top) but the localization fails under noise when the single maximum method is used (bottom).

Our approach is tested on the three face scans shown in Figure 38 with their HK UVS volumes. Table 6 lists the estimated UVS coordinates of the features for all three faces. As can be seen in the table, scales of the features are nearly the same for all faces. Locations of the features for frontal face and for expressional face are very close to each other as can be expected. Absolute coordinates (*x*,*y*,*z*) are computed from the UV coordinates and they coincide with the correct locations of the features on the facial surface.



a. b. c.

Figure 38. UVS volumes for a. rotated towards left, b. frontal and c. expressional (disgusted) faces.

Table 6. Extracted features from the surfaces given in Figure 38 with their locations in UVS space

	U	V	S	U	V	S	U	V	S
L.Eye	75	122	2.5	123	121	2.5	118	124	2.5
R.Eye	35	123	2.5	71	123	2.5	66	124	2.4
Nose	43	92	2.9	97	91	2.9	90	95	3
Saddle	52	123	2.7	97	122	2.7	91	124	2.5
	a. rotated face			b. frontal face			c. expressional face		

6.4 Transform Invariant Topology Construction

In this section, using the feature elements extracted from the UVS space, a global 3D object representation is constructed. A scale and orientation invariant representation is proposed, where the spatial topology of the object is given as a graph structure which carries the relative information among the features over the 3D surface. The relativity is not only in terms of spatial information but in terms of orientation and scaling as well.

As explained in the previous subsection, for each feature element the following attributes are obtained: the type (t_i), the volume (v_i), the positional centre of mass ($\mathbf{x}_i$), the orientation vector ($\mathbf{n}_i$), the scale (s_i) and the size (r_i). If each feature is referred as a node in a topology graph where the nodes carry the feature element's attributes and the links between the nodes carry some relative information; a topological representation may be obtained.

In order to make this representation orientation and scale invariant, the links between the nodes must carry "relative" or in other words "normalized" information. An example of this type of relation could be the length between two nodes normalized using a scale invariant measure specific for that topology. The relative 3D direction between two nodes might also be used. Furthermore scale difference between the nodes would carry scale invariant information. These relative link attributes can be listed such as:

- Normalized distance from Node A to B: $\left|\vec{\mathbf{x}}_B - \vec{\mathbf{x}}_A\right| / r_A \; or \; \left|\vec{\mathbf{x}}_B - \vec{\mathbf{x}}_A\right| / 2^{s_A}$. The distance between two nodes can be normalized using the scale or the size of a base node (A) in the topology. Thus this relation stays invariant under scaling and orientation of the source signal.

- Link Vectors or Link Angles: $(\vec{\mathbf{x}}_B - \vec{\mathbf{x}}_A) / \left|\vec{\mathbf{x}}_B - \vec{\mathbf{x}}_A\right|$: The unit vector from a node (B) to a specific base node (A) in the topology will also remain invariant under scale. However this link vector will be variant under orientation. In order to make this information both scale and orientation invariant, the angles between these unit vectors might be used. For an *n*-node topology there would be *n-1* unit vectors. For any two these unit vectors, an angle can be calculated. This angle will be invariant of both scale and orientation. For *n-1* number of unit vectors, we would obtain C(*n-1,2*) number of angles, which is also equal to *(n-1)·(n-2)/2*. The angle can be calculated as:
$$\alpha_{BAC} = \alpha_{CAB} = \cos^{-1}\left(\left[(\vec{\mathbf{x}}_B - \vec{\mathbf{x}}_A) / \left|\vec{\mathbf{x}}_B - \vec{\mathbf{x}}_A\right|\right]^{\mathrm{T}} \cdot \left[(\vec{\mathbf{x}}_C - \vec{\mathbf{x}}_A) / \left|\vec{\mathbf{x}}_C - \vec{\mathbf{x}}_A\right|\right]\right)$$

- Normal Vector Difference: $\vec{\mathbf{n}}_B - \vec{\mathbf{n}}_A$: The difference vector between the unit normal vector of node B ($\mathbf{n}_B$) and unit normal vector of the specific base node A ($\mathbf{n}_A$) will stay invariant of orientation and scale.

- Feature Scale Difference and Size Ratio: $s_B - s_A$ & r_B / r_A: As explained in the previous chapter, the scale difference between two nodes is invariant to scaling. The ratio of the size of a node (B) (which is strongly related to scale of that feature) to the size a specific base node (A) will stay invariant of scaling as well.

Imagine we have a four-node topology with nodes A, B, C and D. Assume that node A is defined as the base node of the topology. Then the following vector will be scale and orientation invariant:

$$\begin{aligned} \lambda_i = [\quad & t_A, t_B, t_C, t_D, \ldots \\ & \ldots \frac{|\vec{\mathbf{x}}_B - \vec{\mathbf{x}}_A|}{r_{A(or\,2^{s_A})}}, \frac{|\vec{\mathbf{x}}_C - \vec{\mathbf{x}}_A|}{r_A}, \frac{|\vec{\mathbf{x}}_D - \vec{\mathbf{x}}_A|}{r_A}, \ldots \\ & \ldots \alpha_{BAC}, \alpha_{BAD}, \alpha_{DAC} \ldots \\ & \ldots \vec{\mathbf{n}}_B - \vec{\mathbf{n}}_A, \vec{\mathbf{n}}_C - \vec{\mathbf{n}}_A, \vec{\mathbf{n}}_D - \vec{\mathbf{n}}_A, \ldots \\ & \ldots r_B / r_A\, (or\, s_B - s_A), r_C / r_A, r_D / r_A \quad] \end{aligned} \tag{13}$$

This feature vector λ_i has 16 elements (as scalars or vectors). For an *n*-node topology, the number of elements in this vector will be *n+3·(n-1)+(n-1)·(n-2)/2*. This four-node relation may also be shown on a topological chart as shown in Figure 38. Node A is called the base node because the link relations are calculated relative to this node.

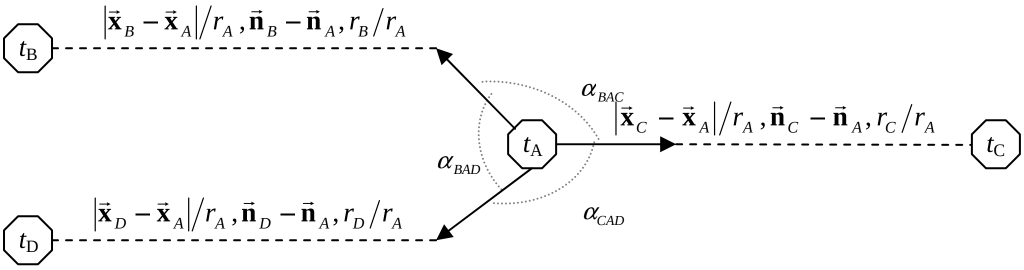

$|\vec{\mathbf{x}}_B - \vec{\mathbf{x}}_A| / r_A, \vec{\mathbf{n}}_B - \vec{\mathbf{n}}_A, r_B / r_A$
t_B
α_{BAC}
$|\vec{\mathbf{x}}_C - \vec{\mathbf{x}}_A| / r_A, \vec{\mathbf{n}}_C - \vec{\mathbf{n}}_A, r_C / r_A$
t_A
t_C
α_{BAD}
$|\vec{\mathbf{x}}_D - \vec{\mathbf{x}}_A| / r_A, \vec{\mathbf{n}}_D - \vec{\mathbf{n}}_A, r_D / r_A$
α_{CAD}
t_D

Figure 38. Four-node, scale and orientation invariant feature vector is shown in a topological chart.

This vector is orientation and scale invariant since all relations are defined relative to node A. However for some applications, orientation and/or scale invariance may not be desired. For example, if the metric size of the object to be recognized is known, then scale invariance is unnecessary. Similarly if the orientation of the object relative to the sensor device is fixed, then orientation invariance capability of a recognition system will be redundant. For this reason orientation and/or scale dependent versions of this vector may be defined.

$$\begin{aligned}\lambda_i = [\quad & t_A,\, t_B, t_C, t_D, \ldots \\ & \ldots\left|\vec{\mathbf{x}}_B - \vec{\mathbf{x}}_A\right|, \left|\vec{\mathbf{x}}_C - \vec{\mathbf{x}}_A\right|, \left|\vec{\mathbf{x}}_D - \vec{\mathbf{x}}_A\right|, \ldots \\ & \ldots \alpha_{BAC}, \alpha_{BAD}, \alpha_{DAC} \ldots \\ & \ldots \vec{\mathbf{n}}_B - \vec{\mathbf{n}}_A, \vec{\mathbf{n}}_C - \vec{\mathbf{n}}_A, \vec{\mathbf{n}}_D - \vec{\mathbf{n}}_A, \ldots \\ & \ldots r_A, r_B, r_C, r_D \quad]\end{aligned} \tag{14}$$

The Equation 14 is an example of an orientation invariant but not scale invariant feature vector representing a four-node topology, since the link lengths and feature sizes are not normalized according to the base node A. On the other hand the feature vector in Equation 15 is scale invariant but not orientation invariant because the feature normal vectors and link vectors are not normalized according to node A.

$$\begin{aligned}\lambda_i = [\quad & t_A,\, t_B, t_C, t_D, \ldots \\ & \ldots \frac{\left|\vec{\mathbf{x}}_B - \vec{\mathbf{x}}_A\right|}{r_A}, \frac{\left|\vec{\mathbf{x}}_C - \vec{\mathbf{x}}_A\right|}{r_A}, \frac{\left|\vec{\mathbf{x}}_D - \vec{\mathbf{x}}_A\right|}{r_A}, \ldots \\ & \ldots \frac{\left(\vec{\mathbf{x}}_B - \vec{\mathbf{x}}_A\right)}{\left|\vec{\mathbf{x}}_B - \vec{\mathbf{x}}_A\right|}, \frac{\left(\vec{\mathbf{x}}_C - \vec{\mathbf{x}}_A\right)}{\left|\vec{\mathbf{x}}_C - \vec{\mathbf{x}}_A\right|}, \frac{\left(\vec{\mathbf{x}}_D - \vec{\mathbf{x}}_A\right)}{\left|\vec{\mathbf{x}}_D - \vec{\mathbf{x}}_A\right|}, \ldots \\ & \ldots \vec{\mathbf{n}}_A, \vec{\mathbf{n}}_B, \vec{\mathbf{n}}_C, \vec{\mathbf{n}}_D, \ldots \\ & \ldots r_B/r_A, r_C/r_A, r_D/r_A \quad]\end{aligned} \tag{15}$$

In Figure 39, a simple demonstration of scale invariance of the proposed method is demonstrated. The smaller surface has half the scale of the bigger one. The radii of the peaks and pits extracted from both of the surfaces are given in the figure. It is clearly seen that the scale ratio is kept constant. In addition the radii are halved for the smaller surface, which is exactly half of the bigger surface in all axes.

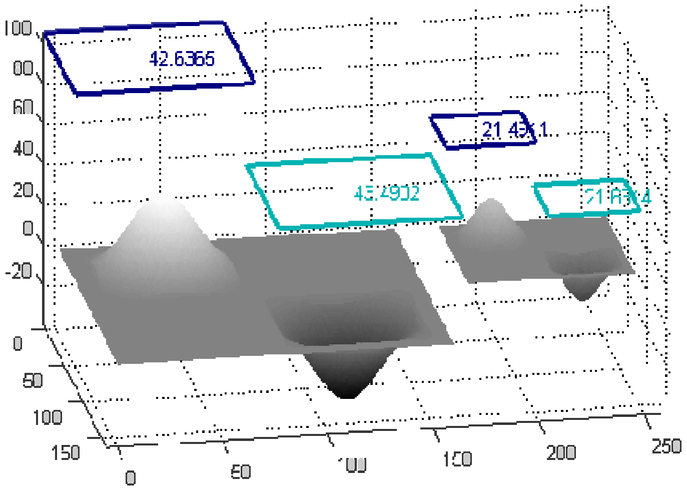


Figure 39. The feature extraction algorithm is scale invariant since the extracted features numerically depend on the scaling ratio of a surface. The numbers are the radii of the extracted feature elements.

Using the equations (13), (14) or (15) a feature vector from n-nodes may be obtained. However excessive number of nodes might be (actually are usually) extracted from a range image. Therefore, many *n*-node combinations may be obtained from a range image and these different *n*-node feature vectors represents different parts of the 3D surface. The algorithm to obtain a complete set of feature vectors which represents the whole 3D surface is given below:

1. Extract all feature elements from the surface:

   $Node^1$: $(t_1), (v_1), (\mathbf{x}_1), (\mathbf{n}_1), (s_1), (r_1)$

   $Node^2$: $(t_2), (v_2), (\mathbf{x}_2), (\mathbf{n}_2), (s_2), (r_2)$

   $Node^3$: $(t_3), (v_3), (\mathbf{x}_3), (\mathbf{n}_3), (s_3), (r_3)$

   $Node^4$: …

2. Select *M* nodes with *M* largest *radii* (r_i) OR *volume* (v_i).
3. Order these selected *M* largest nodes according to the criteria below:
    a. Order them according to their type.
    b. If there is more than one occurrence of a type, order them according to their *radius* (r_i) OR *volume* (v_i).
4. For the ordered *M* nodes do the following:

    a. Select node group size *n*: ($2<n<M$)
    b. Obtain all possible (*k number of*) combinations of node groups of *n* among the *M* ordered nodes: $k=C(M,n)$
    c. Among each *n*-node combination, keep the order of the nodes according to 3.

5. For all *k* combinations,
    a. Select the first feature as the base node.
    b. Extract the feature vectors λ_i using (13), (14) or (15)
    c. Then stack these row vectors in a feature matrix $\Lambda = [\lambda^T \lambda_2^T \lambda_3^T \ldots \lambda_k^T]^T$.

This feature matrix Λ will be a scale and/or orientation representation depending on the equation used to calculate the feature vectors (13), (14) or (15)

6.5 Importance of Scale Space Search

In order to show the significance of scale space search, the feature extraction results with and without scale-space search are depicted in this subsection.

When the scale-space search is omitted, the curvatures are found only at the given resolution. Even though scale/sampling ratio is controlled (usually it is not controlled since this ratio changes even when the distance between the 3D scanner and the object changes), the types and the sizes of the features are detected wrong when only the given resolution is considered. In Figure 40, the extracted features of the original screwdriver object and its scaled version (by 0.8), both with and without scale-space search, are depicted. The features extracted from the original and the scaled versions of the screwdriver object using only the given resolution are usually mislabelled (Figure 40.a,b). For example, only some planar features are located on the handle of the object and the actual shape of the handle could not be extracted. Thus, most surface structures are generally labelled as planes when only the given resolution is considered. Since the original resolutions of 3D images are very high, even inside a peaky region, a point may be considered as a plane because the neighbouring points are very close to each other. As a result, many small planar regions are detected in large concave areas. Thus, only when a scale space search is performed, the real types and sizes of the surface features could be extracted (Figure 40.c,d).

Second of all, the feature sizes can only be correctly extracted if scale-space search is used. Since Figure 40.d is 0.8 times resized version of Figure 40.c, the radius of a feature found in Figure 40.d is also 0.8 times smaller than the radius of the corresponding feature found in Figure 40.c (the numerical values are given in Table 7). However, this property of robustness under scaling cannot be observed when only the given resolution is used for feature detection. Although most features are correspondent in Figure 40.a and Figure 40.b, their radii are faulty. The shape in Figure 40.b is smaller than the one in Figure 40.a by a ratio of 0.8, however the radii of the features numbered as 1 and 2 in Figure 40.b are larger than the radii of the corresponding features in Figure 40.a (Table 7). Thus it is clearly seen that in order to extract features with their size properties, scale-space search is a must.

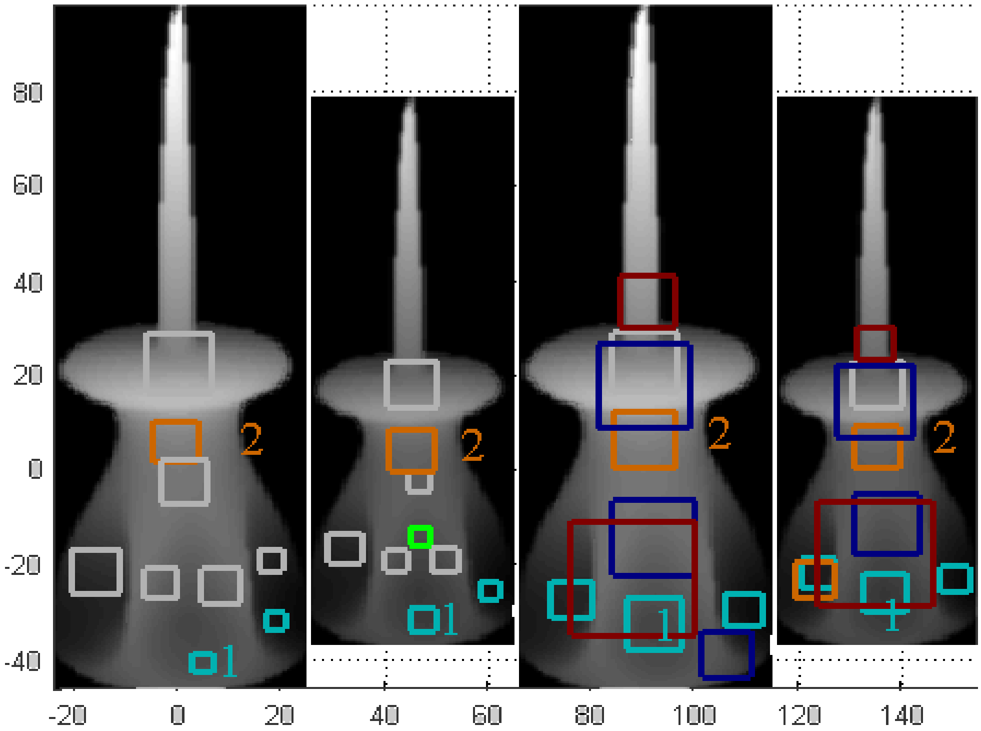


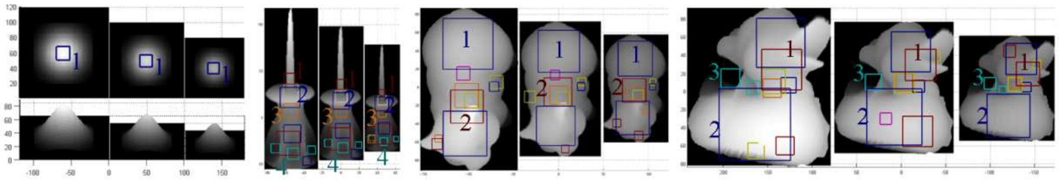
Figure 40. Ten largest features extracted using a) original image without scale-space search, b) scaled image (by 0.8) without scale-space search, c) original image with scale-space search, d) scaled image (by 0.8) with scale-space search.

As seen in Figure 41, the scaled versions are created for various objects and the features are extracted from each one of them. For each object, the original object is in the middle, while 1.2 times and 0.8 times scaled versions lie on the left and the right side of the original respectively. For each object, ten largest extracted features are depicted as squares on the objects. Some features are numbered to help for better visualization and the radii of these numbered features are given in Table 7.

For all object triples, the majority of the extracted features can be detected in all versions. Also, the radius of each feature changes according to the size of the feature, i.e. the object (Table 7). Of course, some other features appear or disappear among different sizes as expected. An artificially created bumpy surface and its scaled version are given in Figure 41.a and only a peak in the center of the shape surface is extracted as a feature for each version. The radii of these extracted features are given in the first row of Table 7. The radius values are consistent with the sizes of the features and the objects. Similarly, some features are also labelled on the original and the scaled versions of screwdriver (Figure 41.b), pig (Figure 41.c) and duck (Figure 41.d) and corresponding radii values are listed in Table 7 also. All feature sizes are detected in consistency with the amount of scaling.

Table 7. The radius values (in mm) of the extracted features in Figure 41

Object / Feature	x 1.2	Original	x 0.8
Figure 5.a / 1	28.7	24.5	21.1
Figure 5.b / 1	17,15	14,16	10.64
Figure 5.b / 2	32,71	26,66	22.71
Figure 5.b / 3	23,20	18,08	13,36
Figure 5.b / 4	18,17	15,83	12,41
Figure 5.c / 1	83,91	69,35	57,70
Figure 5.c / 2	54,37	46,33	38,87
Figure 5.d / 1	67,19	51,96	39,89
Figure 5.d / 2	113,86	93,36	77,47
Figure 5.d / 3	31,68	26,44	21,40



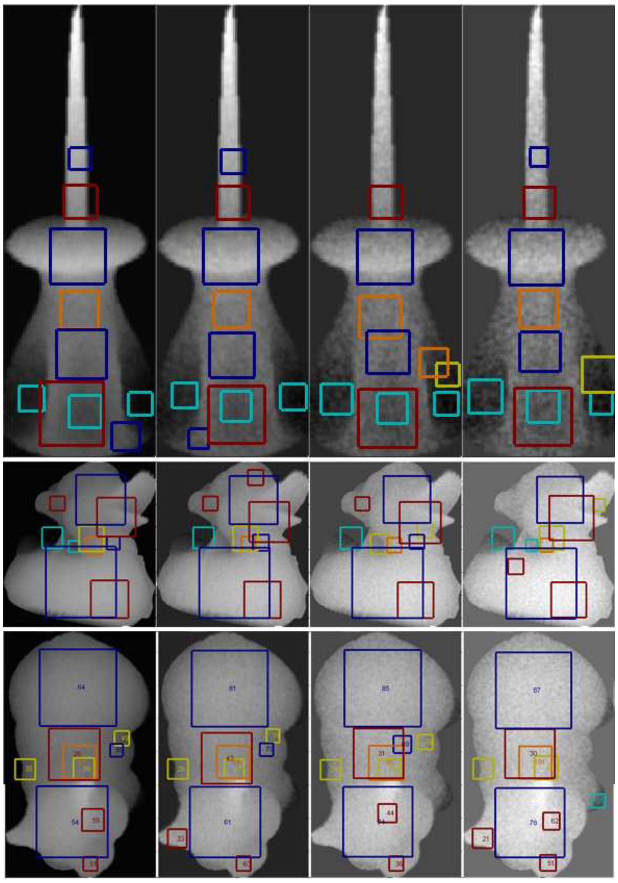
a) b) c) d)

Figure 41. Ten largest extracted features from four different object surfaces. For each object the original (middle), 1.2 times scaled (left) and 0.8 times scaled (right) versions are given. The feature types and sizes are given by colour and square size respectively (peak: blue, saddle ridge: red, convex cylinder: purple, pit: cyan, saddle valley: yellow, concave cylinder: green, hyperbolic: orange, plane: gray).

6.6 Feature Extraction Robustness under Noise

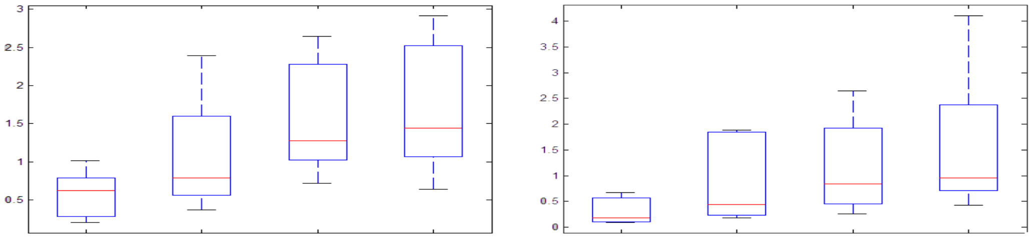
In order to see the affect of noise, the noisy versions of the three objects with different levels of noise are created (Figure 42). Subsequently, features are obtained from the noisy versions. It is easily seen that similar features could generally be extracted from all versions and thus noise does not affect the feature extraction procedure drastically. Even when the there’s %60 noise over the image, features which are found in the higher scales of the UVS space are extracted clearly. Since Gaussian filters are used for each scale-space level construction, noise vanish in higher scales and thus features large enough to be found on higher scale levels are extracted without error.

In Figure 43, the average of the interest point center localization error is given for different noise levels. If a feature could be extracted from both the noisy version and the original (more than %90 of features were correspondent in the original and noisy versions) the average of error between the original feature centers and the centers obtained from the noisy version are calculated and this average error for each noise level is depicted (Figure 43.a). In addition, the error of feature size (radius) calculation for each noisy version is depicted separately in Figure 43.b. As seen from the figures, very high noise levels do not affect interest point localization and radius calculation noticeably.



a) b) c) d)

Figure 42. Ten largest extracted features from the noisy versions of the objects: screwdriver (top), duck (middle) and pig (bottom). a) %15 noise b) %30 noise c) %45 noise d) %60 noise.



a) b)

Figure 43. a) Interest point center localization error in mm for different noise levels. b) Radius calculation error in mm for different noise levels.

6.7 Local Region Descriptions

As stated in the previous subsections, the 3D surface may be globally or locally represented using scale and orientation feature points. This is a quite robust and efficient way to recognize categories of objects. For example; two different facial surfaces will both have two eye pits, one nose peak and a nose saddle. Thus, in order to detect 3D facial surfaces, a simple four-node topology including these facial features could be trained. However, so as to recognize a certain 3D facial surface among a database of facial surfaces; a global topology might not provide sufficient representation. For this reason many 3D recognition approaches attempt to make global or regional definitions of 3D surfaces.

Histogram representation is a compact and efficient method to globally or partially define a 3D surface. A part or whole of a 3D surface or a 3D scene may be represented with different histogram methods such as, depth histograms, normal histograms, curvature histograms or spin images.

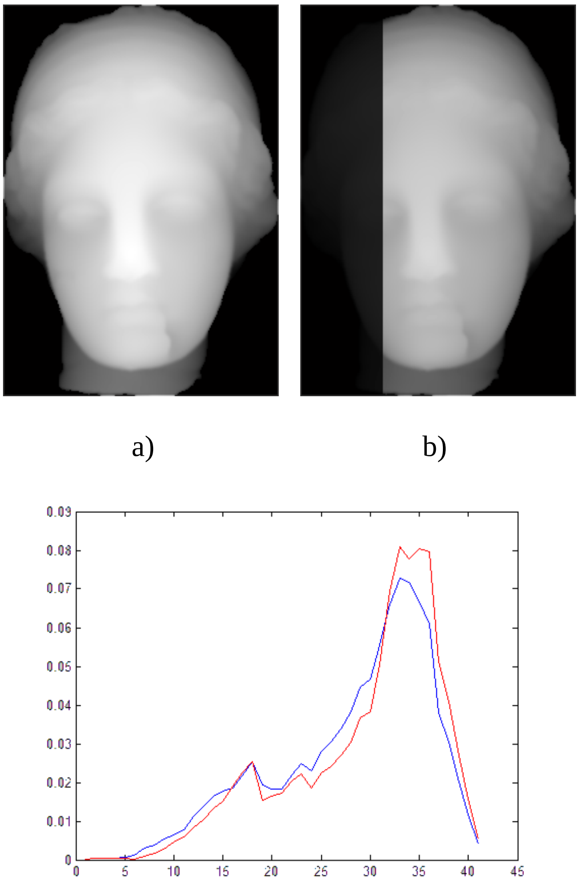
In this chapter, four of the well-known histogram representations for 3D surfaces is reviewed. The representations will later be used to define local regions around the extracted feature points. These representations are depth histograms, normal histograms, shape index histograms and spin images. There several other local descriptors in the literature, namely splash [Stein and Medioni 1992], cyclic images of radial contours (CIRCON) [Torre-Ferrero et.al 2009], point signatures [Chua and Jarvis 1997], intrinsic point signatures [Zhong 2009], 3D point fingerprints [Sun et.al. 2001], 3D shape context (3DSC) [Frome et.al. 2004], spherical spin images [Ruiz-Correa 2001] etc. It is beyond the scope of this thesis to decide which one is the best. Thus the selected four, which are implemented and tested for object recognition, are briefly explained in the succeeding subsections.

6.7.1 Depth Histograms

This is simplest histogram that can be obtained from a range image. Since pixel intensities directly correspond to depth values (distance from the sensor device), this is simply a grey level histogram of the range image. This intensity distribution provides valuable cues about the shape of the 3D surface.

If normalized, these histograms stay invariant under scale changes. They are also invariant under translation and image plane rotation. However they can be very sensitive to the perceived depth range. If there are large and abrupt changes in the depth range, e.g. due to occlusion effects, the whole histogram will be shifted and recognition might no longer be guarantee. For this reason, intensity histograms can only be relied on for the recognition of surfaces with sufficient depth range [Hetzel et. al. 2001].

[Hetzel et. al. 2001] uses depth histograms of the whole range scans for object recognition. Although this is a scale and image plane rotation invariant manner, it is very sensitive to occlusions. In Figure 44, a range image and its %25 occluded version are depicted. The depth histograms obtained from each image are shown as well. The effect of occlusion on the depth histograms is clearly seen.



a) b)

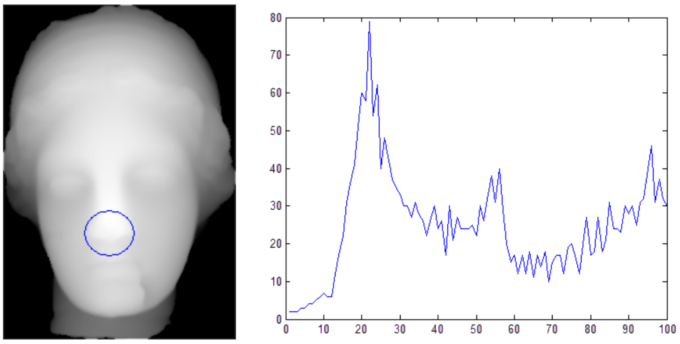
c)

Figure 44. a) The original range image taken from [Stuttgart Database]. b) %25 occluded version of the range image in (a). c) The depth histogram of the original range image (blue) and the %25 occluded image (red).

In order to make depth histograms more robust to occlusion, local depth histograms can be calculated. For this purpose, a feature centre point and feature size should be provided for each local histogram obtained from the range image. Using the scale and orientation invariant feature points as histogram centres and their radii as the regions of interest; a local histogram representation which is robust to occlusions may be proposed. In this representation, for each node extracted from the UVS volume, a depth histogram of the region within the radius of the feature centre is introduced as a new feature element:

$$\mathit{Node}^{i}: \qquad (t_i), (v_i), (\mathbf{x}_i), (\mathbf{n}_i), (s_i), (r_i), [\mathrm{Hist}_{\mathrm{Depth},i}] \tag{16}$$

In Figure 45.b, a local depth histogram obtained from the feature in Figure 44.a is depicted.



a) b)

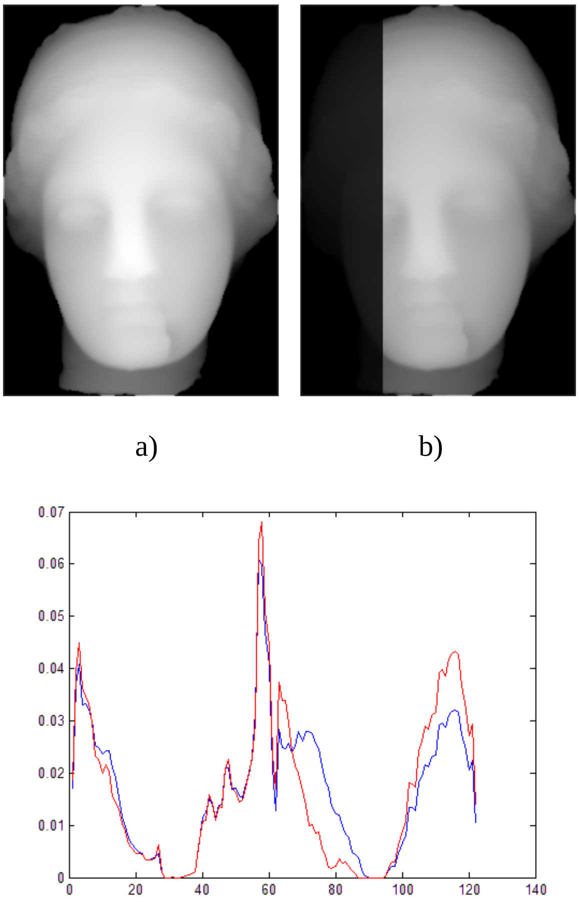
Figure 45. a) Nose is a feature element of peak type. Its radius is the radius of the blue circle around the feature. The local depth histogram of the region within the radius of the feature centre is depicted.

6.7.2 Normal Histograms

As it was explained in chapter 2, surface normals are calculated from the first order derivatives of the surface. A unit normal vector carries two independent variables. There are two main representations commonly used for this purpose. The first just discards the z-component of the normal vector and represents it as a pair (x, y). This corresponds to a projection of the orientation hemisphere on the unit circle. The second possibility is a representation as a pair of angles (θ, φ) in sphere coordinates. The angles can be calculated as follows:

$$\phi = \arctan\left(\frac{n_z}{n_y}\right), \theta = \arctan\frac{\sqrt{\left(n_y^2 + n_z^2\right)}}{n_x} \tag{17}$$

[Hetzel et. al. 2001] uses normal histograms of the whole range scans for object recognition. They use a 2D histogram of (θ, φ) values. Although this is a scale and image plane rotation invariant manner, it is very sensitive to occlusions. In Figure 46, a range image and its %25 occluded version are depicted. The normal histograms obtained from each image are shown as well. The effect of occlusion on the normal histograms is clearly seen.


a)

b)

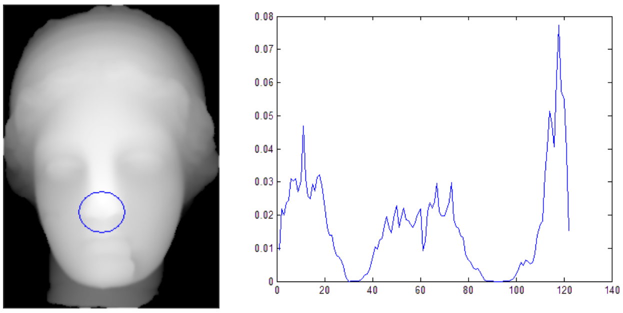
c)

Figure 46. a) The original range image taken from [Stuttgart Database]. b) The 2D normal histogram of the original range image. c) %25 occluded version of the range image in (a). d) The 2D normal histogram of the %25 occluded image.

In order to make normal histograms more robust to occlusion, local normal histograms can be calculated. For this purpose, similar to local depth histograms, a feature centre point and feature size should be provided for each local histogram obtained from the range image. Using the scale and orientation invariant feature points as histogram centres and their radii as the regions of interest; a local histogram representation which is robust to occlusions may be proposed. In this representation, for each node extracted from the UVS volume, a normal histogram of the region within the radius of the feature centre is introduced as a new feature element:

$$Node^i: \quad (t_i), (v_i), (\mathbf{x}_i), (\mathbf{n}_i), (s_i), (r_i), [\mathrm{Hist}_{\mathrm{Normal},i}] \tag{18}$$

In Figure 47.b, a local normal histogram obtained from the feature in Figure 46.a is depicted.



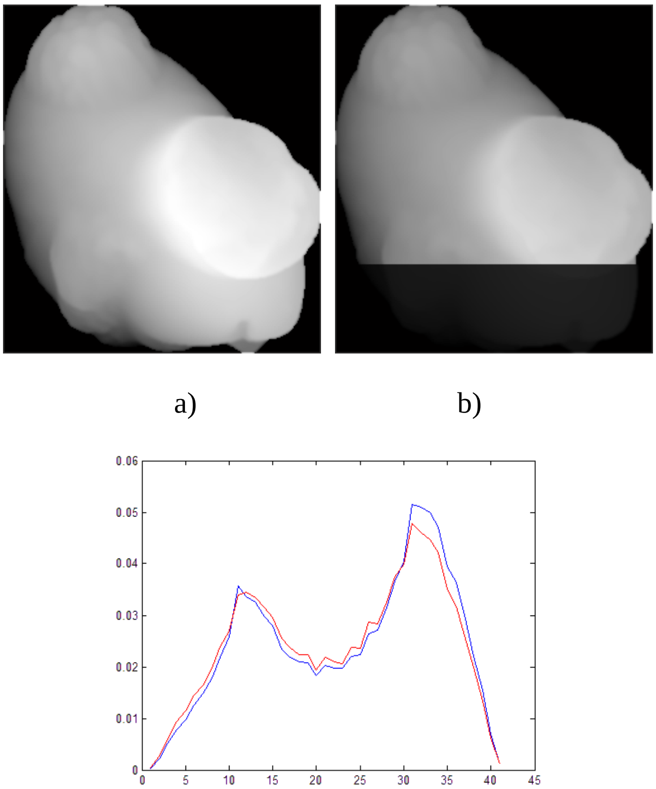
a) b)

Figure 47. a) The nose is a feature element of peak type. Its radius is the radius of the blue circle around the feature. The local normal histogram of the region within the radius of the feature centre is depicted.

6.7.3 Curvature Histograms

Curvature histograms simply carry the curvature distribution information of the input 3D surface. This curvature value might be κ_1, κ_2, H, K, C, S or another value derived from the first or second derivatives of the surface. [Hetzel et. al. 2001] uses shape index (S) histograms of whole range scans for object recognition purposes. They experimentally prove that shape index histograms are better in representing range images compared to depth or normal histograms [Hetzel et. al. 2001]. Furthermore they are advantageous compared to other curvature histograms since a single shape index values can classify a local region (or pixel) into all fundamental types except planes. No other curvature definition is that much discriminative in terms of classification.

Shape index histograms are scale and orientation variant by nature. This representation is even robust to some degree of off-plane rotation. However if the histogram belongs to a whole range scan of the object, sensitiveness to occlusion is inherited. In Figure 48.a and c, a range image and its %25 occluded version are depicted. Shape index histograms obtained from each image are shown as well. The dramatic effect of occlusion on shape index histograms is clearly seen.



a) b)

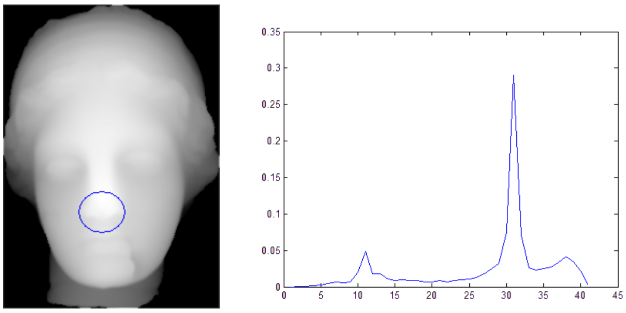
c)

Figure 48. a) The original range image taken from [Stuttgart Database]. b) %25 occluded version of the range image in (a). c) The shape index histogram of the original range image (blue) and the shape index histogram of the %25 occluded image (red).

In a similar manner to the previous approach on depth and normal histograms, shape index histograms may be altered to become more robust to occlusion by calculating local histograms. For this purpose, similar to local depth and normal histograms, a feature centre point and feature size should be provided for each local histogram obtained from the range image. Using the scale and orientation invariant feature points as histogram centres and their radii as the regions of interest; a local histogram representation which is robust to occlusions may be proposed. In this representation, for each node extracted from the UVS volume, a shape index histogram of the region within the radius of the feature centre is introduced as a new feature element:

$$\mathit{Node}^{i}: \qquad (t_i), (v_i), (\mathbf{x}_i), (\mathbf{n}_i), (s_i), (r_i), [\mathrm{Hist}_{\mathrm{ShapeIndex},i}] \tag{19}$$

In Figure 49.b, a local shape index histogram obtained from the feature in Figure 48.a is depicted.



a) b)

Figure 49. a) The nose is a feature element of peak type. Its radius is the radius of the blue circle around the feature. The local shape index histogram of the region within the radius of the feature centre is depicted.

6.7.4 Spin Images

Spin Image is another mapping of nearby points on a surface from 3D to 2D [Johnson and Hebert 1999]. The spin-image for point **p** is found by recording the distance of all nearby points **x** from the surface normal **n** (α - *alpha*) and the distance from **x** to **p** along n (β - *beta*). Corresponding points from different views have similar spin-images, thus spin image is an orientation invariant representation (Figure 50).

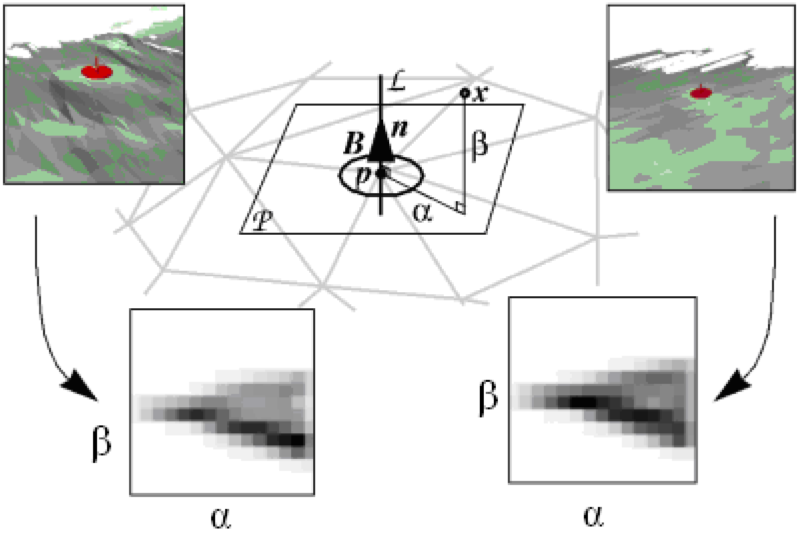


Figure 50. The spin images of point p from different views are shown. The spin images are similar [Johnson and Hebert 1999].

Since spin images are found by using the distances from the neighbouring points, there is no notion of scale invariance. On the other hand there is the notion of resolution invariance which if affected by quantization noise if resolution difference is extreme. However with minor modifications, it is possible to construct resolution and scale invariant spin images. If a patch of a surface is normalized to unity area beforehand, the calculated spin image will stay independent of scaling. More importantly when the patch area is normalized, the distances α and β will be fixed to certain limits, which is crucial in defining α_{max} and β_{max}. The 2D resolution of the spin image is defined according to α_{max} and β_{max}. Very similarly, if the normalized surface patch is re-sampled to fixed resolution, the spin image becomes utterly invariant under resolution changes. In addition the scale/sampling ratio is controlled by this way, which is extremely crucial for our feature point extraction algorithm.

Up to now, the theory behind scale and orientation invariant feature extraction using scale-space of curvatures and defining local regions using different histogram types are introduced. Beginning with the next chapter, the experiments and comparisons carried out to evaluate the proposed methods are given.

CHAPTER 7

EXPERIMENTAL WORK

In this chapter, all the experimental work performed using the proposed feature extraction method and object representation method is presented. The experiments are formed of mainly three parts. The first part focuses on the 3D face detection problem. The second part, which is considerably richer than the others, concentrates on object representation and recognition. This part includes curvature classification comparisons, object recognition on various databases and object registration. Finally the third part presents an application on a different discipline, where surface curvatures of DEMs are used to detect landslide regions. In all experiments, results are compared with methods from the literature.

7.1 Face Detection and Pose Estimation

In our first experiment, we make our first attempt to test the feature extraction algorithm proposed in this thesis. The features are extracted from facial scans in order to detect the location and the orientation (pose) of the face using the HK UVS classification. The feature elements used to categorize a face are two inner eye pits, a nose peak and a nose saddle (nose bridge). These points are chosen because they are enough to estimate the pose of the face so that the orientation of the face can be detected. Also, these points are easily detected by our algorithm and are the ones which are not severely affected by facial gestures. These four elements are later used to construct graph representation using the equation (15) from the previous chapter.

In order to test our representation for face localization, Bosporus 3D face database [Bosporus Database 2007] is used. Bosporus 3D face database includes more than 20 different poses and facial expressions of 78 subjects and that is why it is very suitable for testing the proposed method against various transformations. The neutral and frontal poses of the subjects are used for training. 17 different poses and expressions are used for testing. In order to test our method on facial pose estimation, 35 artificially rotated versions of five different facial scans from the FRGC v1.a database [FRGCv1a] are created.

The proposed method is compared with four different techniques. Only the feature and face detection parts of the algorithms are considered for the compared four algorithms. The compared algorithms are Colombo, Cusano and Schettini's "3D Face Detection Using Curvature Analysis" [Colombo et. al. 2006], Lu and Jain's "Automatic Feature Extraction for Multiview 3D Face Recognition" [Lu and Jain 2006], Lu and Jain's "Multimodal Facial Feature Extraction for Automatic 3D Face Recognition" [Lu and Jain 2005] and Chang et al.'s "Multiple Nose Region Matching for 3D Face Recognition under Varying Facial Expression" [Chang et al 2006].

In the next section, we commence by defining the feature vector used to represent the facial topology.

7.1.1 Transform Invariant Four-Node Face Topology

In order to define a representation for a facial surface, salient and meaningful features should be designated. The significant features on human face are the nose, eyes, chin etc. Thus, any feature set globally defining a face must encompass some of them. For this purpose, we have chosen a four node graph for our facial topology representation, where the nose peak (N), the left eye pit (E_L), the right eye pit (E_R) and the nose saddle (S) form the nodes. These features are chosen because they are the most stable features against facial expressions. We indicate the nose peak (N) as the base node (since it is the single peak inside the topology). Accordingly, using equation (20) from the previous chapter, a scale and orientation invariant feature vector is defined as:

$$\begin{aligned} \lambda_{FACE} = [\quad & t_N, t_{E_L}, t_{E_R}, t_S, \ldots \\ & \ldots \frac{|\vec{\mathbf{x}}_{E_L} - \vec{\mathbf{x}}_N|}{r_{N\,(or\,2^{s_N})}}, \frac{|\vec{\mathbf{x}}_{E_R} - \vec{\mathbf{x}}_N|}{r_N}, \frac{|\vec{\mathbf{x}}_S - \vec{\mathbf{x}}_N|}{r_N}, \ldots \\ & \ldots \alpha_{E_L N E_R}, \alpha_{E_L NS}, \alpha_{E_R NS} \ \ldots \\ & \ldots \vec{\mathbf{n}}_{E_L} - \vec{\mathbf{n}}_N, \vec{\mathbf{n}}_{E_R} - \vec{\mathbf{n}}_N, \vec{\mathbf{n}}_S - \vec{\mathbf{n}}_N, \ldots \\ & \ldots r_{E_L} / r_N \ (or\, s_{E_L} - s_N), r_{E_R} / r_N, r_S / r_N \quad] \end{aligned} \tag{20}$$

For the any facial feature set $\lambda_{FACE,}$ the types (t_N,t_{EL}, t_{ER}, t_S) are predetermined as peak, pit, pit and saddle ridge. Thus they are redundant in the feature vector. In addition for a human face, the normal orientation of most of the features are frontal. For this reason, not much information will be lost, if the normal differences are excluded from the feature vector. Consequently the feature vector can be simplified into:

$$\begin{aligned} \lambda_{FACE} = [\quad & \ldots \frac{|\vec{\mathbf{x}}_{E_L} - \vec{\mathbf{x}}_N|}{r_N}, \frac{|\vec{\mathbf{x}}_{E_R} - \vec{\mathbf{x}}_N|}{r_N}, \frac{|\vec{\mathbf{x}}_S - \vec{\mathbf{x}}_N|}{r_N}, \ldots \\ & \ldots \alpha_{E_L \cdot N \cdot E_R}, \alpha_{E_L \cdot N \cdot S}, \alpha_{E_R \cdot N \cdot S} \ \ldots \\ & \ldots r_{E_L} / r_N, r_{E_R} / r_N, r_S / r_N \quad] \end{aligned} \tag{21}$$

7.1.2 Face Detection on Bosporus Database

[Bosporus Database 2007] is 3D facial database including 21 different poses of 78 different subjects. In this section, the 3D facial detection experiments on this database are presented.

7.1.2.1 Learning the Four-Node Topology

In this study, the four-node face topology given in (21), is modelled by a single Gaussian with a diagonal covariance matrix. In the training phase, feature vectors extracted from the frontal and neutral pose of the Bosporus 3D face database are used. Other poses and expressions of the database are used in testing. The complete workflow of the training phase is depicted in Figure 51.

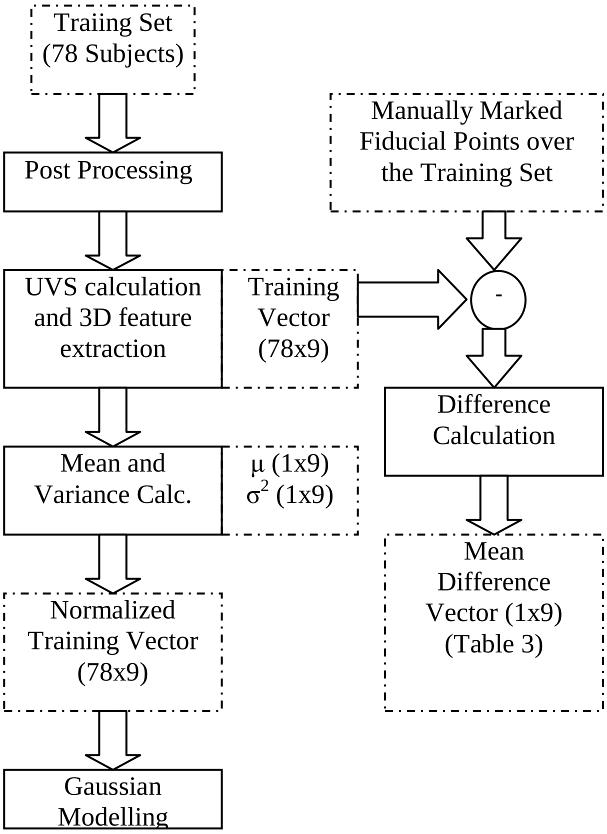

Traiing Set
(78 Subjects)
Post Processing
Manually Marked Fiducial Points over the Training Set
UVS calculation and 3D feature extraction
Training Vector (78x9)
-
Difference Calculation
Mean and Variance Calc.
μ (1x9)
σ2 (1x9)
Mean Difference Vector (1x9) (Table 3)
Normalized Training Vector (78x9)
Gaussian Modelling

Figure 51. The training phase.

In training face, first of all, data is processed in order to fill the gaps and remove the noise. Then, the features from the 78 neutral and frontal faces are obtained. Among these fundamental elements, one of the one-peak, one-saddle, two-pit combination corresponds to the true combination of the nose, nose saddle and the two eyes. Using the true combinations of the neutral facial scans, we extract 78 scale and orientation invariant feature vectors using (21). These 78 feature vectors are used to train a Gaussian with diagonal covariance matrix where the data is first normalized to zero mean and unit variance. In the database, the facial components such as nose pit, inner eye corners were marked manually also. These manual markings are compared with detected points during testing in order to qualitatively check the performance of the algorithms. (Since nose saddle was not marked in the database, we don’t check detection performance for nose saddle.) The facial components were marked on 2D texture images of 3D scans because localizing them on 2D images are easy for humans. However, they may not be the actual centres of the pits and peaks. For example, the nose tip was marked as the nose in the database. However the centre of the nose peak does not occur at the nose tip but at somewhere else on the nose surface. Our system locates the actual centres of the surface peaks and pits. Since the difference between the marked point and the calculated centre has a characteristic, for example the nose peak centre location computed by our method and the marked nose tip differ with a specific vector distance (Figure 52), we simply calculated the mean of these difference vectors from all training facial scans (Table 8). Then, this mean is used as an additive constant to the results obtained from our method during testing.

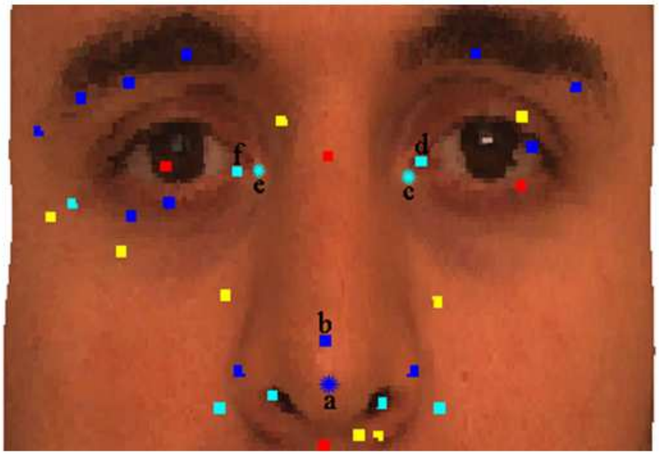


Figure 52. a) The marked nose tip. b) The calculated nose peak centre. c) The marked inner right eye pit. d) The calculated inner right eye pit centre. e) The marked inner left eye pit. f) The calculated inner left eye pit centre.

Table 8. Mean of the differences between the marked and calculated features for the nose tips and inner eye pits in the training set

μ (mm)	X-axis	Y-axis	Z-axis
Nose Tip	0.6705	0.4239	3.0322
Left inner Pit	1.3376	-1.1215	1.6762
Right inner Pit	0.2556	-0.1047	2.0832

7.1.2.2 Testing the Four-Node Topology

The complete workflow for the testing phase is given in Figure 53. First post processing is applied. The noised is cleared and the false holes are filled. The scans are re-sampled and the scale/sampling ratio is fixed to 0.5mm/sample. Then HK UVS space is constructed and features over the facial topology are located for each scan. Among these features, for all combinations of one peak, two pits and one saddle ridge, four-node topology vectors are obtained as candidates or face. These vectors are also normalized by mean and variance values obtained during training. Next, the vector with the highest probability is selected as the best candidate. Finally, the training mean and variance are used to un-normalize the best candidate vector in order to achieve the actual location. Also, the difference vector (Table Table 8), which is calculated in the training phase, is added to the computed locations for nose peaks and eye pits in order to be able to make comparisons with the manual markings. The resulting x-y-z coordinates are taken as the coordinates of the nose tip, inner eye pits and the nose saddle but we evaluate the results only for the nose tip and the inner eye pits.

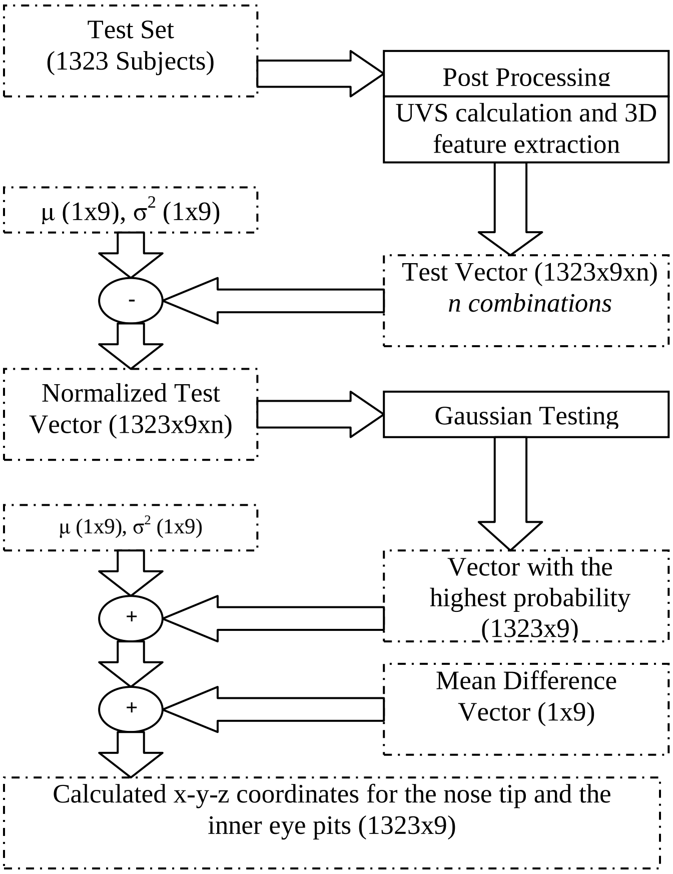

Test Set
(1323 Subjects)
Post Processing
UVS calculation and 3D feature extraction
μ (1x9), σ² (1x9)
Test Vector (1323x9xn)
n combinations
-
Normalized Test Vector (1323x9xn)
Gaussian Testing
μ (1x9), σ² (1x9)
Vector with the highest probability (1323x9)
+
Mean Difference Vector (1x9)
+
Calculated x-y-z coordinates for the nose tip and the inner eye pits (1323x9)

Figure 53. The testing phase.

7.1.3 Experimental Work on 3D Face Detection

In this section the experimental results on Bosporus database are demonstrated. These experiments consist of testing the model on 1323 scans which include poses different from the train poses. Within the database, the anchor points like nose tip, eye pits etc. were marked for test scans as well. Using this as the ground truth we have evaluated our algorithm for detection accuracy in terms of percentage and anchor point localization error in millimetres. In addition to these, we have implemented four other 3D face detection and facial feature point localization methods from the literature. We have tested these methods with the same database and compared our success rates with theirs. In this section we explain the test set. Then other implemented methods are summarized and comparative results are tabulated and visualized.

Table 9. Tested poses and number of facial points included for each pose.

Type of the Pose	Nose Tip	Inner Left Eye Pit	Inner Right Eye Pit
Smile	69	70	71
Disgust	68	68	68
Mouth Open	66	67	67
Eyebrows Up	69	70	70
With Glasses	68	67	67
Hand on one Eye	70	70	28
Hand on mouth	68	68	68
Look Right 30º	70	70	70
Look Right 45º	70	70	67
Look Right 60º	68	69	43
Look Right 75º	58	69	5
Look Up 15º	70	70	70
Look Up 30º	69	70	70
Look Down 15º	70	70	70
Look Down 30º	67	68	68
Look Right & Up 45º	69	70	25
Look Right & Down 45º	61	70	30

7.1.3.1 Test Set

Among many different poses in the Bosporus database, some poses such as smile, disgust, mouth opened, eyebrows up, eye glasses worn, hand on one eye, and hand on mouth are selected to be tested. In addition to these, rotated scans such as look 30º right, 45º right, 60º right, 75º right, 15º up, 30º up, 15º down, 30º down, 45º right and up and 45º right and down are also included. These various poses include orientation changes (e.g. look 45º right), shape deformations (e.g. disgust face) and occlusions (e.g. hand on one eye). Thus, the model is tested for all these conditions and variations.

The tested poses are listed in Table 9. As seen from the table, for some poses (e.g. look right 75 º) very few number of test data (e.g. the inner right eye pit) exist. The reason is that when the face is rotated right for more than 45º, the inner eye pit may disappear and become impossible to be marked.

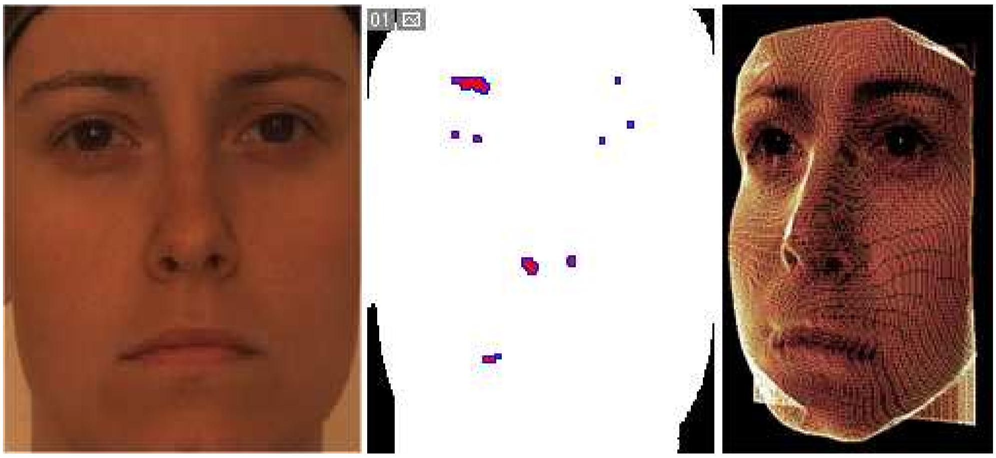
7.1.3.2 Post-Processing on 3D Facial Scans

Bosporus 3D face database [Bosporus Database 2007] is acquired by Inspeck Mega Capturor II [Inspeck-web]. Fortunately, during the construction of the data-base, almost all existing problems such as holes and spikes were cleaned manually from the models using scanner’s built in 3D scan cleaning software. However, 3D models in the database still have considerable amount of holes around dark hair, eye-brows, and eyelids. By holes we mean 3D points labelled as invalid. In addition, some spiky outliers are also resident near some extremely specular regions like eye pupils. In all models, background is also labelled as invalid by default.

To begin with, the connected components of points which were labelled invalid are detected. Among these connected ‘holes’, we select the ones having areas smaller than a threshold (100 in this case) assuming that the background will be the biggest connected component. Using the valid coordinates of the points neighbouring these selected holes we extrapolate the values of these invalid points and thus fill the holes (Figure 54).

As seen in Figure 54.b, this regular grid also includes some invalid points. If the scanner output is to be used to construct a polygonal mesh, the points with invalid labels can be automatically discarded by methods like Delaunay triangulation which is used to construct polygonal data. However in this study, it is crucial for us that the regular grid form of the 3D point cloud is preserved due to “Gaussian Pyramiding Reduce and/or Expand” operations. For this reason, we have chosen not to discard these invalid points but to fill them by assigning the minimal valid z coordinate value to the invalid z coordinate. Then again for invalid x and y coordinates, we linearly extrapolate their values using the valid x and y coordinates that reside on the same row and column respectively. As a result, we obtain a regular grid of valid 3D points as seen in Figure 12.c.

After filling holes, the 3D surface is needed to be smoothed using a Gaussian-like operator because reliable curvature estimation on 3D data is very sensitive to quantization and sensor noise [Flynn and Jain 1989]. Therefore we have applied a 5-by-5 Gaussian operator on the grid for smoothing noise and spiky regions.



a) b) c)

Figure 54. a) Texture for the 3D Scan. b) Black areas denote the background. Red areas denote the invalid points selected to be extrapolated. Blue areas denote the neighbouring valid points used for extrapolation. White areas denote valid points. c) Cleaned cropped and re-sampled point cloud.

Finally we resample the smoothed 3D point cloud into an M-by-N regular grid of 3D points, such that the average scale/sampling ratio will be 0.5 mm per point. Since each face has different metric size, each image is re-sampled into a different resolution, which will satisfy average 0.5 mm scale/sampling ratio. It should be remembered that re-sampling the data does not change the metric distances. Since the 3D scanner output has absolute 3D coordinates, the relative distances maintain their original values even after re-sampling.

7.1.3.3 Compared Methods

We evaluate our results in comparison with four other methods we have implemented. These four methods are: M1) Xiaoguang and Jain's "Multimodal Facial Feature Extraction for Automatic 3D Face Recognition" [Lu and Jain 2005], M2) Xiaoguang and Jain's "Automatic Feature Extraction for Multiview 3D Face Recognition" [Lu and Jain 2006], M3) Chang et al.'s "Multiple Nose Region Matching for 3D Face Recognition under Varying Facial Expression" [Chang et. al. 2006], and M4) Colombo, Cusano and Schettini's "3D Face Detection Using Curvature Analysis" [Colombo et.al. 2005].

M1 assumes that the scan is frontal and each row of the depth image is searched for the maximum z value. For each column, the number of maximum z value positions is counted and the column with the maximum number is selected as the midline. Then, nose tip candidates are assumed to be located on the midline and a vertical Z profile analysis is done on this line. Since nose bridge presents a contiguous increase in z values, nose tip is defined as the last point of one of the three non-decreasing continuous series over the midline. After detecting three-candidate nose tips, a horizontal z profile analysis is done for these three points. Sum of the differences between candidate points and their neighbours are calculated and the point with maximum difference is selected as the nose tip. After detecting the nose tip, possible locations of the two inner eye pit points are calculated statistically. In search region, shape index and corner analysis is done. Needless to say, the method is not orientation invariant. If the subject looks upwards during scanning, for example, the chin may be detected as the nose. Also, the method depends on lots of heuristics which may not be satisfied all the time.

M2 attempts to find the nose tip as the first step also. In order to find the nose tip, 3D data is rotated around Y- axis (axis perpendicular to the horizontal plane) from –90º to 90º with 2º intervals. For each rotated version, the point with maximum z value is selected as the nose tip candidate. The best three candidates are selected to be analyzed by PCA. However, since we obtain high success in locating the correct nose position, PCA is not applied but best candidate is selected by sum of differences between candidate points and their neighbours as described in M1. After finding the nose tip, possible locations of the two inner eye pit points are calculated statistically similarly as explained for M1. However, different from M1, search regions for pit regions vary among different poses. The method is partially orientation invariant however it has the potential tendency to detect spiky regions as the nose and the method is still fragile under rotation of face around X and Z axes.

M3 attempts to find the inner eye pits first. They use H and K values. The nose tip is expected to be a peak and the inner eye points are expected to be pit-like regions. First, small pit regions are removed. Then, pairs of pit regions with small differences in both Y and Z directions are selected as candidate eye pit regions. The nose tip is found next. Over the line drawn perpendicular to the midpoint of the line formed by the two pit regions, the point with maximum z value is selected as the nose tip. Unsurprisingly this method is not scale and resolution invariant and very brittle under spiky noise. Also, the method is not robust to in plane rotations.

In all these methods, first of all, the location for a type of anchor point is estimated independently and then other anchor points are located based on the first detected one. Thus, the spatial structure of anchor points is not considered efficiently. In contrast to this, M4 attempts to find possible anchor point triangles (i.e. nose tip, inner left eye, inner right eye). The method uses H and K values to find the anchor point candidates. A nose tip is expected to be a peak and the inner eye points are expected to be pit-like regions. After finding all candidate points, impossible triangles are removed in order to decrease computational complexity. In doing this, distances between anchor points are thresholded by some values defined based on the training data. 50 face triangles are manually labelled for 50 training faces and the area inside the triangles are analyzed by PCA in order to learn a face model. For test images, the triangular area is extracted for each candidate anchor point triangle and analyzed by the same principle components. This method is similar to our method in the sense that the feature combinations are searched; however it is variant under scale and resolution. Also, our method estimates the face location based on only the features extracted from a scale space and their spatial layout but M4 applies PCA analysis to the part of the face defined by the candidate anchor points. Since this is done for each candidate triangle it is very time consuming.

7.1.3.4 Comparative Results

For evaluation purposes, we have prepared two tables including the results from our method and the four other methods from the literature. The first table depicts the percentage accuracies of the methods in locating the anchor points. If the detected point is in the 2 cm radius of its manually marked position, then the detection is considered to be correct (Table 10). 2 cm is very high but we intended to accept as many detection results as possible in the first step, because the localization accuracy is investigated later among these detections. 17 poses and 3 landmark points for each pose (3x17=51 results in total) are tabulated for each method in the table. The best detection rates are underlined and bolded. We see that our method has the best rating for 35 out of all 51 results. M4 has the best rate for 15 out of 51 results, which are mostly for the detection of the nose tip. M1 and M2 each have the best results in 2 cases. M3 has not given any best result among all poses and landmarks. M4 and our method are highly robust against facial expressions (smile, disgust, mouth open, eyebrows up) and present over %97 success in anchor point detection. For nose detection M4 is slightly better, 1.5 % more than ours. But for eye detection, our method is much better (by more than 30%) than M4 for various facial expressions. Methods M1, M2 and M3 are not robust against facial expressions and show less performance than others.

Again M4 and our method achieve well for in plane rotations up to 600. All of the methods fail for rotations greater than 600. For rotations around Y-axis, M3 has nearly no correct detection, whereas M1 and M2 have severely decreasing success as the amount of rotation increases. For rotations around X axis, M1 is better than M2 and M3. For complex rotations around X and Y axis, our method performs the best where others show no performance at all. For all kinds and amounts of rotations, our method performs much better for eye detection than all of the other methods.

For occlusion scenarios, our method and M4 perform the best. We can also say that M4 is better for nose detection but our algorithm is much better for eye detection.

In the second table (Table 11), the detection accuracies in millimetres are examined. For all five methods, only the successfully detected scans are considered and the error is computed as the absolute distance between the marked points and the calculated points. The means and the standard deviations of the error are compared in Table 6 and the lowest ones are bolded and underlined. Since only the successfully detected landmarks are considered even though the detection rates were not very successful for a method, the localization results may seem successful. Thus, to evaluate a method’s success better we should consider the methods which have both the lowest error mean in Table 6 and the best detection success in Table 5. These methods are also indicated in Table 6 with white font colour over dark grey background. Our method has both the lowest error mean and the best detection rate in 31 out of 51 cases where method M4 has such a success only for 5 of the cases. The other methods could not achieve this success for any pose.

When we compare precision of the algorithms, i.e. the exact locations of the anchor points, although M4 performs slightly (1 mm on the average) better than our algorithm for nose localization, our algorithm performs much better (4 mm on the average) for eye localization. The reason for this is that M4 aims to find the nose tip and the manual markings were done for the nose tip also. However, our method aims to find the center of the nose peak which is different from the nose tip. Although we tried to compensate the difference between the nose peak center and the nose tip by adding a mean difference vector obtained during training (Table 8) to our results, due to different nose shapes this usually causes a little correction. For eye pits, the problem is not as severe as the nose case because our algorithm aims pit centers and manual markings were done for pit centers also. Thus, our algorithm's success becomes more obvious for eye pits.

Now, we will compare the algorithms pose by pose. When we check the algorithms for neutral and frontal poses, M1 and M2 have acceptable rates since they have some assumptions which are valid only for frontal neutral poses. M3 has the worst performance since an error in detecting the pit region misleads the detection of nose peak. M4 has the best rate for nose detection but there is a noticeable decrease in the eye pit detection. This is caused because of the rotation applied to each candidate triangle since pits on both sides of the mouth may also be selected as eye pits. Our method performs the best for both nose and eye detection for frontal poses.

For mouth open pose, success rates of M1 and M3 increase and success rate of M2 decreases when compared to frontal and neutral pose. M4 performs the best for nose but the worst for the eye pits and the reason is the same as explained above for the neutral pose. Our method performs the best for both nose and eye detection. For disgust pose, M3 and M4 are severely affected especially in eye detection since eye pits are severely deformed in this pose. Our algorithm performs the best among all for especially eye detection. For eyebrows up pose, our method performs the best and M2 is the least affected one among the others. For smile pose, success rates of all methods decrease and M4 is the most affected one especially for the eye pits. Again, our method performs the best among all in the average. For yaw rotations only our method presents a performance for rotations more than 450 and all others fail. For various yaw rotations, M3 is the most severely affected one and our method performs the best especially for eyes. For look upwards and downwards poses, all the methods except ours are severely affected. For look bottom right and upper right poses, although our method performs the best, all of the methods have decreased success rates. For hand on one eye pose, all of the methods are affected. But M4 performs the best for nose detection and our algorithm performs the best for eye detection.

Table 10. Detection percentage of the nose and the inner eye pits for various poses are listed for all methods. The first column from left (ours) in each main column is the proposed method. Then M1, M2, M3 and M4 are given in the columns from left to right respectively. Best results for each pose are underlined and bolded.

%	Nose					Left Eye					Right Eye				
	Ours	M1	M2	M3	M4	Ours	M1	M2	M3	M4	Ours	M1	M2	M3	M4
Mouth Open	99,8	97	75,8	80,3	**100**	**100**	97	77,3	87,9	75,8	**100**	97	74,2	84,8	75,8
Disgust	99,3	89,7	92,6	58,8	**100**	**98,7**	91,2	92,6	69,1	63,2	**98,7**	91,2	92,6	77,9	64,7
Eyebrows Up	**100**	79,7	92,8	68,1	**100**	**100,0**	79,7	92,8	63,8	72,5	**100,0**	81,2	92,8	72,5	71
Smile	97,1	82,6	78,3	65,2	**98,6**	**99,3**	82,6	78,3	72,5	62,3	**99,3**	85,5	79,7	78,3	68,1
Look Right 30º	99,3	70	81,4	4,3	**100**	**99,3**	81,4	81,4	54,3	64,3	**99,3**	75,7	80	8,6	68,6
Look Right 45º	94,3	48,6	72,9	7,1	**100**	**94,3**	65,7	72,9	41,4	72,9	**94,3**	55,2	67,2	7,5	71,6
Look Right 60º	**82,9**	7,4	54,4	0	60,9	**86,5**	10,1	55,1	24,6	63,8	**83,5**	9,3	39,5	2,3	44,2
Look Right 75º	28,3	0	**32,2**	0	2,9	34,5	0	**42,6**	14,7	39,7	**60,0**	0	0	0	0
Look Up 30º	96,8	78,3	43,5	39,1	**97,1**	**95,9**	76,8	44,9	71	68,1	**95,9**	76,8	44,9	75,4	76,8
Look Up 15º	99,7	95,7	82,9	61,4	**100**	**99,6**	88,6	82,9	71,4	68,6	**99,6**	90	82,9	70	74,3
Look Down 15º	98,6	81,4	52,9	75,7	**100**	**100**	87,1	52,9	88,6	74,3	**100**	87,1	51,4	81,4	80
Look Down 30º	92,2	59,7	17,9	40,3	**98,5**	**98,1**	77,6	14,9	92,5	76,1	**98,1**	76,1	16,4	88,1	80,6
Look Right & Down 45º	**49,7**	14,3	22,2	4,7	35,7	**44,9**	24,3	28,6	11,4	41,4	**64,7**	20,1	26,7	10,3	53,3
Look Right & Up 45º	**74,6**	16,2	51,5	0	22,9	**77,6**	20	58,6	32,9	42,9	**77,6**	20,1	44,1	4,4	32,1
Hand on one Eye	72,0	53,6	23,2	57,9	**95,7**	**71,3**	59,4	26,1	59,4	65,2	**76,4**	60,7	25	64,3	75
Hand on mouth	96,6	59,7	14,9	16,4	**100**	**96,8**	62,7	14,9	74,6	82,1	**96,6**	62,7	14,9	59,7	79,1
With Glasses	88,8	89,7	30,9	75	**100**	86,3	**92,5**	31,3	67,2	83,6	88,5	**91**	29,9	59,7	79,1

Table 11. Detection errors are compared for all methods. The mean and the standard deviation of the error (the absolute distance between the marked points and the calculated points) are given. The values are in millimetres. The mean is written over the standard deviation.

μ (mm) σ (mm)	Nose					Left Eye					Right Eye				
	Ours’	M1	M2	M3	M4	Ours	M1	M2	M3	M4	Ours’	M1	M2	M3	M4
Mouth Open	**3,01** 2,57	3,96 2,54	5,35 2,40	4,27 2,94	3,05 1,71	**3,07** 1,72	6,08 2,95	6,91 3,99	8,51 4,09	7,57 4,86	**2,61** 1,42	6,13 3,23	5,55 3,17	11,30 3,84	5,98 4,55
Disgust	4,50 3,20	6,09 3,51	6,20 3,01	7,38 4,88	**4,34** 2,67	**3,16** 1,76	5,94 4,49	5,69 4,08	9,41 5,00	7,60 5,15	3,11 ,154	5,56 3,72	5,12 2,97	11,16 4,29	6,71 4,24
Eyebrows Up	**3,05** **1,94**	3,80 2,80	5,29 2,92	4,51 2,87	3,21 2,16	**2,79** 1,50	6,36 3,10	6,38 3,17	9,35 4,54	7,01 4,42	**2,72** 1,72	6,17 3,02	5,91 2,96	11,36 4,07	7,40 5,16
Smile	4,02 2,60	5,65 3,85	5,09 2,47	6,58 4,25	**3,90** 2,12	**3,12** 1,78	6,21 3,92	6,37 3,86	8,74 4,41	6,60 3,99	**3,19** 1,68	5,84 3,00	5,52 2,98	10,36 3,93	6,88 4,84
Look Right 30º	4,93 2,77	7,39 3,97	**4,61** 2,99	9,58 1,05	4,65 2,88	**2,77** 1,65	7,51 3,95	7,66 3,89	14,51 3,48	7,91 4,22	**3,65** 2,53	5,84 3,94	5,25 2,93	13,58 4,70	7,69 5,52
Look Right 45º	7,75 2,49	10,93 4,87	**5,66** 3,71	15,10 3,85	7,78 3,86	**3,40** 2,17	7,85 4,04	6,86 3,27	12,61 4,38	8,69 4,62	**5,54** 4,18	7,68 4,13	7,91 4,92	13,34 4,22	7,64 5,34
Look Right 60º	9,90 2,97	10,44 3,81	**8,45** 3,85	-	9,81 3,98	**3,94** 1,72	6,94 5,15	6,71 3,29	12,93 5,84	9,85 5,95	7,47 5,21	11,19 5,07	7,78 3,62	19,20 -	**7,04** 4,95
Look Right 75º	**12,48** 3,14	-	13,73 4,46	-	13,76 7,69	**4,21** 2,23	-	8,23 5,19	11,42 6,63	9,28 5,31	**2,89** 0,75	-	-	-	-
Look Up 30º	4,27 2,20	3,48 2,71	6,06 2,93	4,53 2,77	**3,44** 2,41	**3,65** 1,67	8,75 4,97	5,57 2,72	9,29 3,99	6,55 4,07	**3,82** 1,94	8,49 4,14	5,14 3,27	11,24 4,16	7,12 4,69
Look Up 15º	3,32 1,80	3,48 3,00	4,74 2,24	4,84 3,21	**3,07** 2,12	**2,99** 1,67	6,30 2,92	5,28 2,87	9,33 4,18	7,80 4,60	**3,01** 1,71	6,27 3,52	5,35 3,22	11,21 3,43	6,71 4,44
Look Down 15º	4,13 2,42	6,17 3,90	5,72 3,00	5,46 3,24	**4,10** 2,42	**2,98** 1,52	6,19 3,80	7,03 4,17	7,29 4,02	6,29 4,42	**2,97** 1,43	6,03 4,10	5,95 3,33	9,20 3,96	5,34 3,84
Look Down 30º	**4,58** 3,10	7,46 5,32	5,83 2,59	6,04 3,75	4,61 2,48	**3,42** 1,52	4,67 3,71	4,64 3,92	6,87 3,40	4,74 3,24	**3,32** 1,31	5,42 3,47	5,47 3,70	7,51 3,50	4,47 3,23
Look Right & Down 45º	**8,63** 3,08	11,46 4,77	8,87 4,00	14,14 4,80	8,78 4,11	**4,64** 1,71	8,24 4,93	7,24 4,27	10,10 4,84	6,62 4,36	**5,94** 3,61	10,09 4,22	7,38 4,05	7,35 4,51	7,51 5,15
Look Right & Up 45º	11,59 3,61	13,53 4,23	**9,22** 4,41	-	11,35 4,28	**4,13** 2,27	9,99 4,36	7,37 4,44	12,47 4,79	9,32 5,26	7,58 4,84	11,7 6,57	8,84 6,69	11,37 -	**5,47** 3,18
Hand on one Eye	**2,89** 1,85	5,29 3,85	6,01 2,20	5,40 3,75	3,82 2,22	**2,65** 1,50	7,39 4,02	7,42 5,35	9,90 4,96	6,62 4,33	5,83 3,78	5,98 3,42	**4,48** 2,02	10,06 5,11	8,06 4,72
Hand on mouth	**3,72** 2,58	4,76 3,35	5,52 2,68	7,69 5,59	3,98 2,36	**2,93** 1,82	6,05 3,46	7,70 3,73	10,48 4,84	6,68 4,05	**2,82** 1,37	5,24 3,30	5,76 4,26	12,65 4,50	6,17 4,11
With Glasses	**2,90** 1,96	4,33 2,75	5,14 2,58	4,49 2,66	3,03 1,70	**3,47** 2,49	7,16 4,01	6,41 3,04	10,63 4,82	7,49 3,90	**3,92** 2,79	7,38 4,31	7,82 3,67	11,94 4,38	7,61 4,13

7.1.3.5 Comparison in terms of Complexity

When an algorithm is implemented, computation load is the other important criteria which should always be considered besides its success. Computational load can be measured in terms of complexity. We investigate complexities of the five algorithms in terms of Big-O notation. Our method, M1, M2 and M3 each has the same complexity of O(MN) where M and N are the sizes of the 3D data grid. M4's complexity is O(MNmn2) where m is the number of possible nose peak candidates and n is the number of possible eye pit candidates. As can easily concluded from the values, M4 requires much more computation than other algorithms and our algorithm requires the same amount of computation as M1, M2 and M3 although its success is much better than those.

7.1.3.6 Problematic Cases

Even though the results of our method are apparently better than other methods, we still can point out some problematic cases for our method. Since the method proposed is scale and transform invariant, for frontal poses and poses rotated less than 45º, success rates higher than 98% were achieved. 100% were achieved for a majority. However when the face is rotated over 45º, nose tip and eye pits may become invisible. When an element is occluded because of rotation and other reasons (like hand over one eye), the method may fail to find the correct one-peak, two-eyes and one-saddle combination. For these over rotated faces, the nose usually loses its fundamental property of becoming a peak and is not selected among the best combination. Similarly when one eye is occluded with a hand or glasses in front of it, the eye pit vanishes and correct combination is again lost. For these reasons the success rates for these types of poses are relatively lower than other poses, but they are still higher than the other compared methods. Such cases are shown in Figure 55. In Figure 55.e. for faces rotated 45º rights and down, two results are depicted. For the upper figure the face is successfully registered, however for the lower figure the method fails since no right eye pit could be detected because of occlusion. Similarly in Figure 55.d. for faces rotated 75º rights, two results are depicted. Again for the upper figure, the eye pits, the nose peak and the nose saddle are detected and the method is successful. However for the lower figure inner right eye pit is invisible and is not detected, thus causing the GMM to select a bad combination.

However for frontal poses and poses rotated less than 45º, where success rates over 98% were achieved, there are very rare occasions where false combinations are selected. Some examples of these rare occasions are given in Figure 55.a. for the pose with glasses. The optical scanners are very sensitive to specular surfaces and lenses. The eye glasses may cause false spikes or holes over the surface, which cannot be easily overcome by post processing. For the lower figure the glasses cause a false peak over the right eye and a false combination is selected. In Figure 55.b. where the mouth open faces are scanned, a false combination is selected for the lower figure. The eye pits are found correctly but, somehow the upper lip is selected instead of the nose tip. Actually this is the single false detection case for this pose. Yet again for Figure 55.c. the right eye is occluded with the right hand. For the upper figure the inner right eye pit is still visible and the method succeeds, however for the lower figure the pit is completely occluded, thus causing the method to fail.

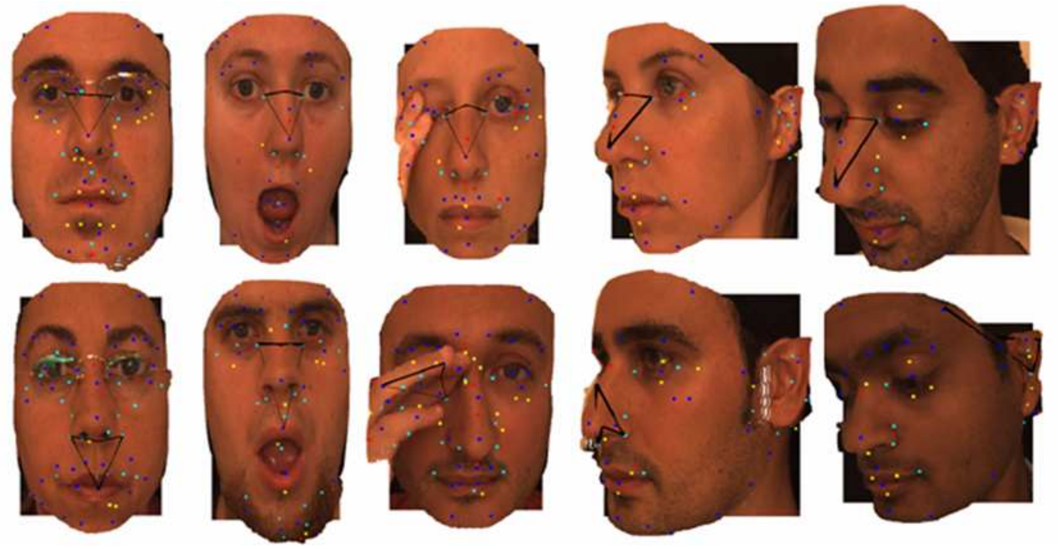


Figure 55. A successful and a problematic case for the poses a) with glasses. b) Mouth open c) Hand on one eye d) Look right 75º e) Look right and down 45º

7.1.4 3D Facial Pose Estimation

As another attempt to evaluate the quality and capability of the extracted features, a pose estimation algorithm is proposed in this section. Using the 3D features extracted from the HK UVS volume; poses of rotated faces are estimated. So as to make the experiments numerically comparable, virtually rotated versions of 3D frontal facial models form the FRGC ver1.a are used.

7.1.4.1 Construction of Artificially Rotated 3D Facial models

In order to make controlled tests on pose estimation virtually rotated versions of FRGC v1.a frontal facial scans are created using a software constructed in our laboratory, METU-CVIS. The software is a 3D virtual scanner program where, different scenes using different object formats such as range images or 3D object models, can be created and exported in range scan format. The software is created by Nesli Bozkurt, as a part of her M.Sc. thesis [Bozkurt 2008].

For this purpose, 35 surface models are artificially created by applying 7 different space transformations to 5 original (frontal) surfaces from the FRGC v1.a database. These transformations are rotation around Y-axis by +10º, +20º,-10º,-20º, rotation around X-axis by +15º,-15º and rotation around Z-axis by +45º. A set of artificially created range scans are given in Figure 56.

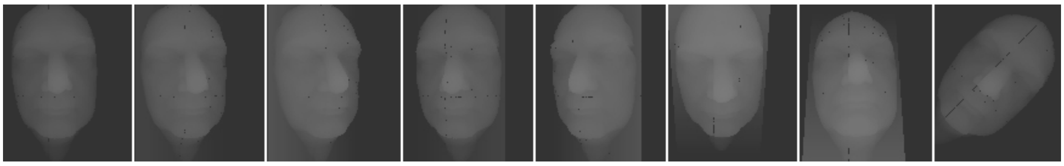


Figure 56. Artificially created facial scans. Leftmost scan is the original scan from FRGC database. Other scans are created by applying the following transformations respectively from left to right: rotation around Y-axis by +10º,+20º,-10º,-20º, rotation around X-axis by +15º,-15º and rotation around Z-axis by +45º.

The four-node facial topology given in the previous subsection is sought in the original and artificially created models by using the feature vector λ_{FACE} given in (21). Again, all four-node, "peak-pit-pit-saddle" combinations are extracted and the best candidate (the one with the highest probability distribution within the GMM) is selected as the actual facial four-node topology.

When the facial nodes (nose peak, eye pits and nose bridge) are detected on the original and the rotated scans, the 3D locations of these nodes are known. Using this data, the amount of rotation of a rotated 3D facial scan might be calculated with respect to the original frontal 3D facial scan. So as to do this, transformation matrix from extracted nodes of the rotated scan to the extracted nodes of the original frontal scan must be calculated. In other words, this transformation matrix gives the relations between 3D reference frames of the rotated and original 3D scans. If we define the rotated reference frame C_R and original reference frame C_O, the transformation matrix can be calculated as:

$$\mathbf{C_R} = [\mathbf{T}_{4x4}]\cdot\mathbf{C_O} \Rightarrow [\mathbf{T}_{4x4}] = \mathbf{C_R}\cdot\mathbf{C_O}^{-1} \tag{22}$$

Figure 57 shows the estimation results. Range images, HK UVS volume depictions and the estimated angles for the rotated scans are given. It is seen from the figure that for rotation around the Y-axis, the estimates are bad since localization of the eye pits become weak as they get occluded behind the nose saddle. However for rotation around the X-axis better estimates are obtained since the eye pits and the nose peak do not get occluded. Similarly the estimates for rotation around the Z-axis are good as no occlusion occurs.

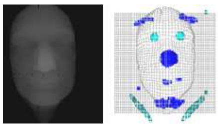
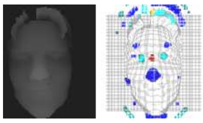
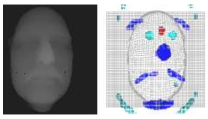
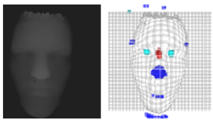
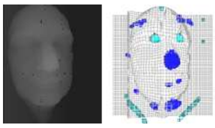
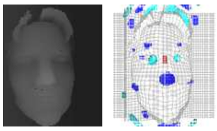
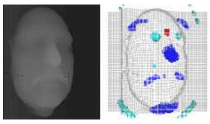
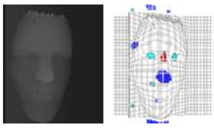
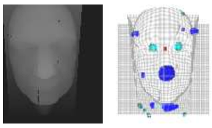
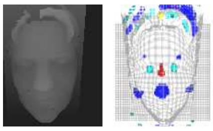
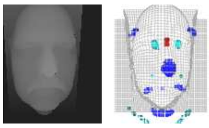
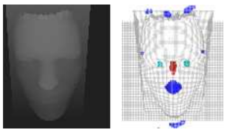
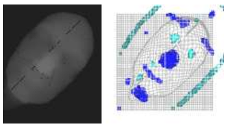
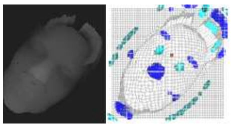
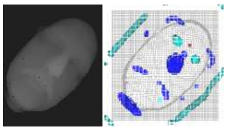
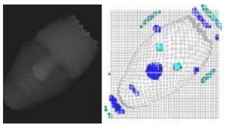
Original (Frontal) Scans			
			
Rotated +10° by Y			
 Estimated +8.84° by Y	 Estimated +8.43° by Y	 Estimated +11.37° by Y	 Estimated +8.62° by Y
Rotated +15° by X			
 Estimated +14.84° by X	 Estimated +13.79 by X	 Estimated +16.21° by X	 Estimated +14.55° by X
Rotated +45° by Z			
 Estimated+46.14° by Z	 Estimated +48.17 by Z	 Estimated +47.01° by Z	 Estimated +43.26° by Z

Figure 57. Virtually created facial scans and their estimated pose angles

7.1.5 Conclusion on 3D Facial Detection and Pose Estimation

In this chapter, the scale invariant features are extracted from facial range images for the purpose of face detection and pose estimation. Four-node facial graph topology is constructed using a peak, two pits and a saddle ridge feature element. Using this 3D topological element the locations of facial features are detected and the facial pose is found.

The detection method is compared with four representative methods in the literature. These methods are selected because some of them are only applicable to faces whereas some of them are for general object detection methods applied to faces.

The methods are trained using 78 neutral and frontal scans and tested using 1323 scans of 17 different poses of various rotation, expression and occlusion. The results show that the proposed method is more successful in terms of detection percentage, localization accuracy and computational efficiency compared to the other methods.

The main reasons that the proposed method is more powerful in detection and localization are the scale and transform invariance in detecting the anchor points and the same invariance in constructing the spatial graph structure. The method easily overcomes false positive results caused by spiky noises, rotation and scale. Since the proposed method requires no prior compulsory poses (like frontal pose or a side view), the detection rates are high for any kind of pose. As long as the nose-eyes-nose bridge quadruple is detected, the face is detected for many poses, with near 100% success.

For frontal poses with different expressions such as, mouth opened, smile, disgust, eyebrows up and hand on mouth near 100% success is achieved. For poses with rotation less than 45°, the percentage success rates are still over 97%. For occluded poses such as with glasses and hand on one eye, the results are acceptable and better than the compared methods.

For our proposed method, the detection accuracy in millimetres is also better than the existing methods. A few millimetres of error mean and variance are achieved for rotations less than 45°. Considering that the database was manually marked, a few millimetres of mean error can be regarded as absolute success because human marking error is also not negligible.

The facial detection is brittle for cases where one of the anchor points is not visible because of rotation or occlusion. If the eye pit is occluded with a hand or because of high degrees of rotation, its detection fails and the correct quadruple is not located.

The pose estimation is carried out on artificially rotated versions of five FRGC1.a facial scans. The rotations in x, y and z axes are estimated with around ±2° error x, y and ±4° error in z axes. The proposed method is extremely powerful for pose estimation since it is invariant in any type of transformation including rotation, scaling, re-sampling. The method is also robust to noise since the features are found within a scale-space where extreme amounts of smoothing are performed.

7.2 3D Object Recognition

In this chapter, we finally arrive at the core subject of this dissertation, 3D object recognition. Using the proposed feature extraction method and 3D topology model, objects are classified and/or categorized. We commence by defining the n-node topology derived from the definitions in Chapter 5. Then the method used in this study to classify objects, namely geometric hashing, will be given in detail. Tests carried out in Stuttgart Database [Stuttgart Database], will be covered and discussions and performance comparison to other methods are also given in this chapter.

7.2.1 Transform Invariant n-Node Topology

The method to extract transform invariant features and construct a transform invariant topology model is thoroughly covered in Chapter 5. In this subsection we simply give the details of the topological model derived from the proposed model. The general model for an n-node topology can be formulized as:

$$
\begin{aligned}
\lambda_i = [\quad & t_A,\, t_B, t_C, t_D, \ldots, t_N \\
& \ldots \left|\vec{\mathbf{x}}_A - \vec{\mathbf{x}}_N\right|, \left|\vec{\mathbf{x}}_B - \vec{\mathbf{x}}_N\right|, \ldots, \left|\vec{\mathbf{x}}_M - \vec{\mathbf{x}}_N\right|, \\
& \ldots \alpha_{ANB}, \alpha_{ANC}, \alpha_{AND}, \ldots, \alpha_{LNM}, \\
& \ldots \vec{\mathbf{n}}_A - \vec{\mathbf{n}}_N, \vec{\mathbf{n}}_B - \vec{\mathbf{n}}_N, \ldots, \vec{\mathbf{n}}_M - \vec{\mathbf{n}}_N, \\
& \ldots r_A, r_B, r_C, \ldots, r_N \qquad]
\end{aligned}
\tag{23}
$$

In this vector there are n number of type values (first row), n-1 number of link lengths, C(n,2) number of angles, n-1 number of normal difference vectors and n number of radii, totally (n^2+8n-5)/2 number of values.

On the other hand in its scale invariant version, the link lengths and radii are normalized according to the radius of the base vector. The base vector is selected as the node having a certain type (e.g. peak) with the largest radius. This vector is defined as:

$$
\begin{aligned}
\lambda_i = [\quad & t_A,\, t_B, t_C, t_D, \ldots, t_N \\
& \ldots \frac{|\vec{\mathbf{x}}_A - \vec{\mathbf{x}}_N|}{r_N}, \frac{|\vec{\mathbf{x}}_B - \vec{\mathbf{x}}_N|}{r_N}, \ldots, \frac{|\vec{\mathbf{x}}_M - \vec{\mathbf{x}}_N|}{r_N}, \\
& \ldots \alpha_{ANB}, \alpha_{ANC}, \alpha_{AND}, \ldots, \alpha_{LNM}, \\
& \ldots \vec{\mathbf{n}}_A - \vec{\mathbf{n}}_N, \vec{\mathbf{n}}_B - \vec{\mathbf{n}}_N, \ldots, \vec{\mathbf{n}}_M - \vec{\mathbf{n}}_N, \\
& \ldots \frac{r_A}{r_N}, \frac{r_B}{r_N}, \frac{r_C}{r_N}, \ldots, \frac{r_M}{r_N} \qquad]
\end{aligned}
\tag{24}
$$

This vector is used to represent a range image, and consequently to classify the object within a database. It is impossible to know which element of this vector carries better classification quality, unless proper experiments are completed. In addition the nodes, or in other word, the extracted features could be obtained using the HK or SC curvature classification types. Thus, choosing the appropriate feature classification method is another important issue for object recognition. In the next subsections, all these issues will be handled with certain test procedures. However, before that, the next subsection gives the details of the topology classification method, which basically decides which topology belongs to which object or which category of objects.

Similar feature vectors to (23) and (24) can be constructed by using SIFT if all range images are rendered to gray scale images, where the depth value designates the gray level intensity. By using the SIFT feature attributes, namely the pixel location (x_i :{u_i ,v_i}), the scale (σ_i), the orientation (θ_i) and the descriptor (d_i), a feature vector similar to (24) can be created:

In Equation (25), $|x_i$-$x_j|/\sigma_j$ designates the normalized distance, $\alpha_{i\text{-}j\text{-}k}$ designates the link angles, θ_i-θ_j designates the orientation difference and σ_i/σ_j designates the size ratio and di designates the SIFT descriptor.

$$
\begin{aligned}
\boldsymbol{\lambda}_i = [\quad & \frac{|\vec{\mathbf{x}}_B - \vec{\mathbf{x}}_A|}{\sigma_A}, \frac{|\vec{\mathbf{x}}_C - \vec{\mathbf{x}}_A|}{\sigma_A}, \ldots, \frac{|\vec{\mathbf{x}}_N - \vec{\mathbf{x}}_A|}{\sigma_A}; \\
& \alpha_{BAC}, \alpha_{BAD}, \ldots, \alpha_{MAN}; \\
& \theta_B - \theta_A, \theta_C - \theta_A, \ldots, \theta_N - \theta; \\
& \sigma_B/\sigma_A, \sigma_C/\sigma_A, \ldots, \sigma_N/\sigma_A; \\
& \mathbf{d}_A, \mathbf{d}_B, \ldots, \mathbf{d}_N \qquad]
\end{aligned}
\tag{25}
$$

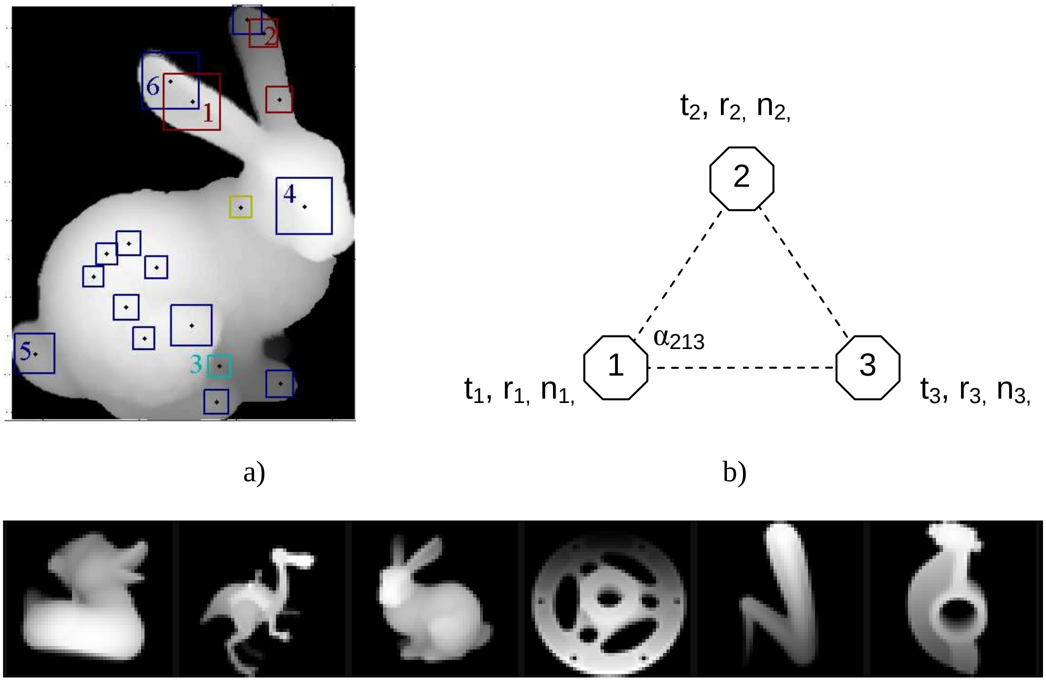
7.2.2 Learning the n-node Topology – Geometric Hashing

The feature vector used in this study can be considered as a partial topology of a general topology, where each part contains N elements out of total W (W>N) elements on a range image. This feature vector resembles to a partial graph belonging to a global graph. For each range image, U largest features (W>U>N) in terms of radius are selected and all N-tuple combinations of these k features (totally C(U,N) feature vectors) are used to construct the feature vectors defining that range image (Table 12).

Table 12. C(U,N) number of features obtained from k largest features, using N-node topology. For the sake of simplicity the last node is chosen as the base node.

Feature 1	$t_A, t_B, t_C, t_D, ..., t_N, \lvert\vec{\mathbf{x}}_A - \vec{\mathbf{x}}_N\rvert, \lvert\vec{\mathbf{x}}_B - \vec{\mathbf{x}}_N\rvert, ..., \lvert\vec{\mathbf{x}}_M - \vec{\mathbf{x}}_N\rvert,$ $\alpha_{ANB}, \alpha_{ANC}, \alpha_{AND}, ..., \alpha_{LNM}, \vec{\mathbf{n}}_A - \vec{\mathbf{n}}_N, \vec{\mathbf{n}}_B - \vec{\mathbf{n}}_N, ..., \vec{\mathbf{n}}_M - \vec{\mathbf{n}}_N, r_A, r_B, r_C, ..., r_N$
Feature 2	$t_A, t_C, t_D, t_E, ..., t_O, \lvert\vec{\mathbf{x}}_B - \vec{\mathbf{x}}_O\rvert, \lvert\vec{\mathbf{x}}_C - \vec{\mathbf{x}}_O\rvert, ..., \lvert\vec{\mathbf{x}}_N - \vec{\mathbf{x}}_O\rvert,$ $\alpha_{BOC}, \alpha_{COD}, \alpha_{DOE}, ..., \alpha_{MON}, \vec{\mathbf{n}}_B - \vec{\mathbf{n}}_O, \vec{\mathbf{n}}_C - \vec{\mathbf{n}}_O, ..., \vec{\mathbf{n}}_N - \vec{\mathbf{n}}_O, r_B, r_C, r_D ..., r_O$
…	…
Feature C(U,N)	$..., t_S, t_T, , t_U, ... \lvert\vec{\mathbf{x}}_S - \vec{\mathbf{x}}_U\rvert, \lvert\vec{\mathbf{x}}_T - \vec{\mathbf{x}}_U\rvert,$ $...\alpha_{PUS}, \alpha_{SUT}, ...\vec{\mathbf{n}}_S - \vec{\mathbf{n}}_U, \vec{\mathbf{n}}_T - \vec{\mathbf{n}}_U, ... r_S, r_T, r_U$

In this study, these feature vectors are used in a geometric hashing method for object recognition purpose. The main reason behind using geometric hashing is that it allows partial matching of small topologies in a general topology so that the recognition process becomes robust to noise, rotation and even occlusion. In our hashing method, indexing is done by the types of the n-tuples and each entry includes the feature vector of the triplet besides the code of the pose and the object. In other words, for a range image the C(U,N) long feature vector is constructed. At the pre-processing stage of hashing, for each range image in the training set of category of object (for example range images of bunny taken from various angles), the feature vectors are calculated. Thus we obtain Num_Tr$_i$ number of feature vectors for the object i with Num_Tr$_i$ number of training range image in the database.   This database construction stage can be computed offline. For various range image training sets belonging to different objects, this operation is completed. Consequently at the recognition stage, features of the test model (i.e. the model to be recognized) are extracted and then related hash table indexes are obtained. By comparing the hash table entries using a similarity measure between the feature vectors, matching features are found. Corresponding to the indexes in the training sets, matched model's vote is incremented by one for a particular training set. Finally, the database model which receives the greatest number of votes is taken as the match of the test object.



a)

b)

c)

Figure 58. Demonstration of geometric hashing: All the topologies constructed from a range image are compared to the topology vectors in the database. The database model which receives the greatest number of votes is taken as the match of the test object.

Figure 58 demonstrates the geometric hashing procedure. For a range image (bunny, Figure 58.a) the feature are extracted. Largest U (6 in this case) features are selected. Then combination of all N-tuples (3) are found and the feature vector is constructed (totally C(6,3) = 20 feature vectors). Then for each range image in all of the training sets in the database, this feature vector is matched. For each correct match the vote of that training set is incremented and the object is recognized by final voting.

7.2.2.1 Similarity Measure

The similarity measure defined distance between two feature vectors. For two features to be compared, they must have the same types feature points. It is not logical to compare two pits and a saddle with two peaks and a saddle. For any groups of feature points with the same type combinations, the definition of a similarity measure is proper.

For comparable feature vectors, if the distance between any corresponding feature element, namely the difference between the normal directions or the difference between the angles of each triple or the difference between two link length ratios etc., is below a given threshold then the two vectors are referred as similar.

The threshold values that indicate the similarity between two feature vectors are found by making many experiments on the data. The fine-tuned threshold values are given in the next subsection.

7.2.3 HK-SC Curvature Space Classification Comparison and Threshold Fine Tuning

Since both HK and SC spaces classify surface patches in to similar types, their classification capabilities are comparable. For this reason, there is an ongoing debate on the advantages and disadvantages of using mean & Gaussian (HK) or shape index & curvedness (SC) curvature spaces for object recognition applications. In [Cantzler and Fisher 2001], HK and SC curvature descriptions are compared in terms of classification, impact of thresholds and impact of noise levels and it is concluded that SC approach has some advantages at low thresholds, in complex scenes and in dealing with noise. However in that study the curvatures are calculated only at the lowest scale, i.e. the given resolution. Scale-spaces of the surfaces or the curvatures are not defined. Another comparative study has been carried out in [Li and Hancock 2004] where curvature values obtained from the shading in 2D images are used and HK and SC histograms are created. The comparison results show that SC histograms are slightly more successful in terms of classification. Yet again, the tested resolution is the pixel resolution of the 2D image and the effect of sampling is ignored.

When calculating H, K, C and S values, the scale/resolution ratio is highly effective. However due to its scale invariant nature, shape index (S) values are independent of the resolution or the scale. Thus it is no wonder that SC methods give better results than HK methods when the comparison is carried out at an uncontrolled scale/resolution level. In order to make H, K and C values also scale invariant, the scale/resolution ratio of the scan must be set to a constant value for the whole database. In addition to this, a scale space of the surface should be constructed so that the features which also carry the scale level information can be obtained by using H, K, S and C values. Obtaining the absolute scales of these features is crucial for 3D object representation and the only way of achieving such information is constructing the scale-spaces of the surfaces in terms of curvature values.

There have been different attempts at constructing a scale-space of a surface and defining scale-invariant features [Akagündüz and Ulusoy 2007]. However, there has been no study which uses a scale-space approach for the comparison of HK and SC for their classification capabilities. For this experiment, our main motivation and contribution is to make both mathematical and experimental comparisons of HK and SC curvature descriptions in their scale-spaces. For this purpose we calculate the scale spaces of the curvatures of the given surfaces for each method separately. Then we extract scale and orientation invariant features from each space. By this we mean that features are extracted with their scale information both in SC scale space and HK scale space. Finally, we use these features in an object recognition task and compare the performances of the methods so as to decide which curvature space is better in terms of feature quality and object recognition. In addition, in order to optimize the parameters and examine the effect of parameters on each method, several tests are run using different parameters values.

The mathematical comparison of HK and SC spaces are given in Chapter 3. Furthermore the proposed feature extraction and topology construction method is given in Chapter 4 and Chapter 5. The recognition method to test the feature vectors are generalized in the previous subsections. In this subsection we first give the details of the recognition method and then proceed with the experiments.

To begin with, For each range image, ten largest features in terms of volume are selected and all triple combinations of these 10 features (totally C(10,3)=120 feature vectors) are used to construct the feature vectors defining that range image. As explained previously, at the pre-processing stage of hashing, invariant features are extracted and saved to the hash table for each pose and each training model range image. Consequently at the recognition stage, features of the test model (i.e. the model to be recognized) are extracted and then related hash table indexes are obtained. By comparing the hash table entries using the similarity measure given before, corresponding to these indexes, matched model's vote is incremented by one. Finally, the database model which receives the greatest number of votes is taken as the match of the test object. In this study, Stuttgart database [Stuttgart Database] is used where there are 256 scans for 42 objects. As defined in the web page of the database, 66 poses of each object is used for training and the rest of the poses are used for testing.

In this study, two groups of experiments are performed. First group of experiments is carried out in order to decide on the best threshold values and features. When comparing two feature vectors, we first check if the types of the three features are identical. If so, we check if the absolute difference between the angles ($\alpha_{i,j,k}$) is at most thr_ang degrees. Then similarly, we check if the absolute difference between the norms of distances (($d^2_{i,j}$,+ $d^2_{j,k}$ + $d^2_{i,j})^{1/2}$) of the compared feature vectors is at most thr_dist mm. A series of experiments are held both on HK and SC methods in order to decide on the values of these parameters for each method. The second group of tests is carried out in order to compare HK and SC in object recognition against the number of objects in the database. For the sake of simplicity and speed, the first group of tests is carried out using only 5 objects but the object recognition tests are run for various number of objects.

7.2.3.1 Experiments for Thresholds

It is an important and difficult task to decide on similarity thresholds. There are two similarity measures defined for our feature vector: the angle similarity and the distance similarity. The angle similarity is the absolute difference between the angle attributes of the two feature vectors. The threshold value for absolute angle difference is set to 3˚, 5˚, 10˚, 20˚ and the recognition performances are observed separately for HK and SC methods (Figure 59), while the distance threshold is fixed to 20 mm. The recognition performance is defined as the percentage of correctly recognized poses among all test poses. The HK method is robust for the angle thresholds. Both methods perform the best when the angle threshold is 10˚. Similarly, the distance threshold is set to 5, 10, 20, 30 mm, while the angle threshold is fixed to 10˚ and the performances of both methods are given in Figure 60. The HK method is robust for different distance thresholds. But both methods perform the best when the distance threshold is 20 mm.

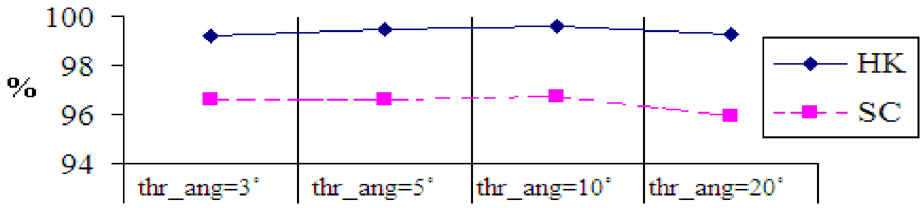


Figure 59. Angle threshold tests for both methods.

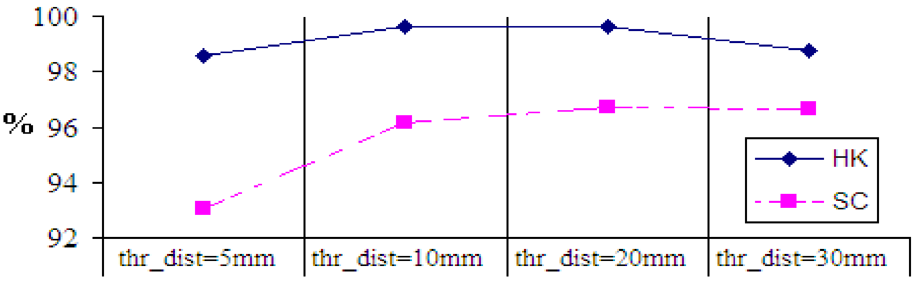


Figure 60. Length threshold tests for both methods.

7.2.3.2 Experiments for Types

Both methods classify regions into eight fundamental types, namely peak (1), convex cylinder (2), saddle ridge (3), plane (5), hyperbola (6), pit (7), concave cylinder (8) and saddle valley (9). Some of these types are more useful where some types are even unreliable. Different groups of types are included in the feature vector and tested for each method (Figure 61). The HK method is robust against different type combinations but the best results for both are obtained when only the types 1, 3, 5, 7 and 9 are included in the feature vector.

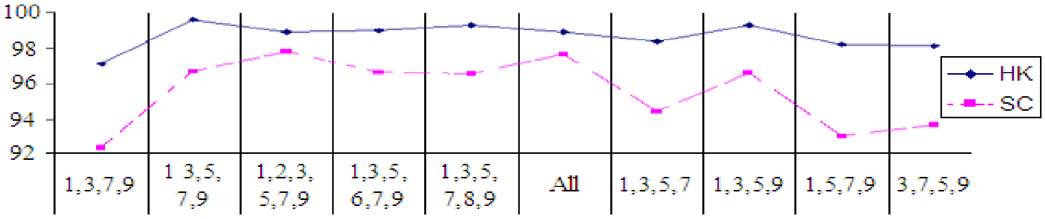


Figure 61. Type tests for both methods.

7.2.3.3 Experiments for Feature Numbers

In our method, a number of features with the largest volumes are selected for each range image and triple combinations among these features are found in order to construct the feature vectors. Taking fewer numbers of features would speed up the process. However, good quality features may be omitted. On the other hand, taking more features will slow down the process and some bad quality features may be included. Thus we search for the optimum number of features by testing different numbers such as 6, 10, 14, 16 and 18 (Figure 62). The HK method is stable against different number of features but the SC method is severely affected by the feature number. When ten largest features are selected from each range image, both methods perform the best.

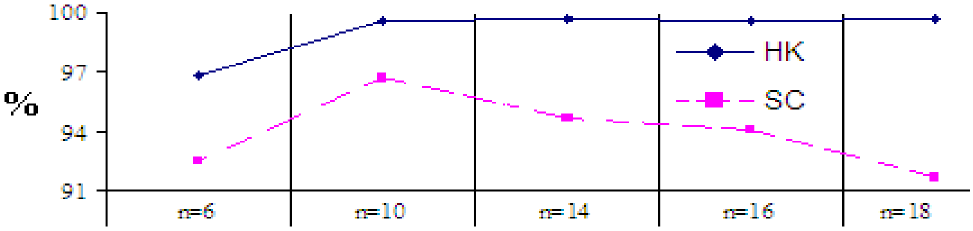


Figure 62. Feature number tests for both methods.

7.2.3.4 Experiments for Database Size

After deciding on the threshold values and the feature vector content, three experiments are run where database size is increased from 5 objects to 42 (all) objects. By this way the robustness of each method is tested under varying number of objects in the database (Figure 63). The performance of the SC method decreases linearly with the size of the database where the performance of the HK method decreases logarithmically.

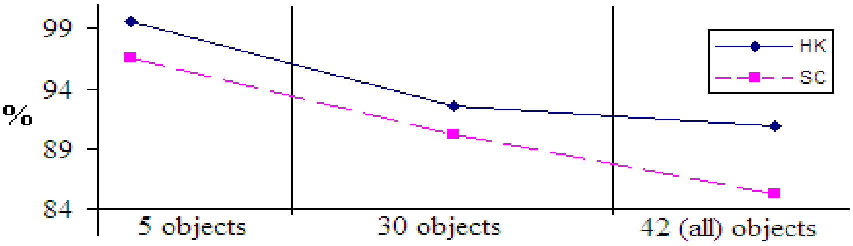


Figure 63. Database size tests for both methods.

When compared with a scale-space approach, the HK method outperforms the SC method in all tests. We think that the main reason behind this is the definition of planar regions in the HK method. This definition embraces many ambiguous regions more consistent than SC’s. In addition, the SC method is less robust to feature type changes and database size. As the number of selected features for a range image increases, the quality of features reduces faster for the SC method. All results confirm that the HK method is more successful.

7.2.4 3D Object Recognition Tests

In this subsection the object recognition capability of the proposed 3D feature are tested. Es mentioned before, multi-scale features are extracted from sampled surfaces and then used to construct a scale and orientation invariant topological representation of object categories. Features are located over the surface with their metric size irrespective of the surface resolution. Stuttgart Range Image database [Stuttgart Database] is used for testing the proposed features in object category recognition. Originally this database contains 42 objects with 258 different virtual scans of each (totally 10836 scans). However the database does not include scale varying or occluded scans. In this study, for the sake of testing the scale invariance and occlusion robustness limits of the proposed method, ¼ scaled, %25 occluded and both ¼ scaled and %25 occluded versions of these scans are virtually created and tested. In order to classify the object categories, geometric hashing [Lamdan and Wolfson 1988] is used. Besides, the results are compared to SIFT [Lowe 2004], for which each original, scaled or occluded range image is rendered to a gray scale image.

For the sake of observing the benefits of using multi-scale features, the 3D features are extracted with and without scale-space search. First, the features are extracted using the conventional method, in which the surface is classified using the H and K curvatures obtained from the given resolution. This method will be referred to as single scale feature extraction method (SSFE). Then, the features are extracted using the proposed multi-scale feature extraction method where the scale-space of curvatures is used. This method will be referred to as multi-scale feature extraction method (MSFE). The feature vectors are obtained as in (24) and hashing is applied as explained above for both methods (SSFE and MSFE) separately.

In order to see the benefits of using feature point descriptors, spin images around the feature centers are also extracted and appended to the feature vector as in (25) for both

SSFE and MSFE separately. We will refer to these methods as ‘SSFE+spin’ and ‘MSFE+spin’. As explained in the previous section, the effect regions of the spin images are designated by the radii of the features.

Finally the SIFT features are obtained from the gray scale images which are obtained by rendering the range image surfaces. Then (23) is used to construct the feature vectors by the SIFT features. This method will be referred to as SIFT.

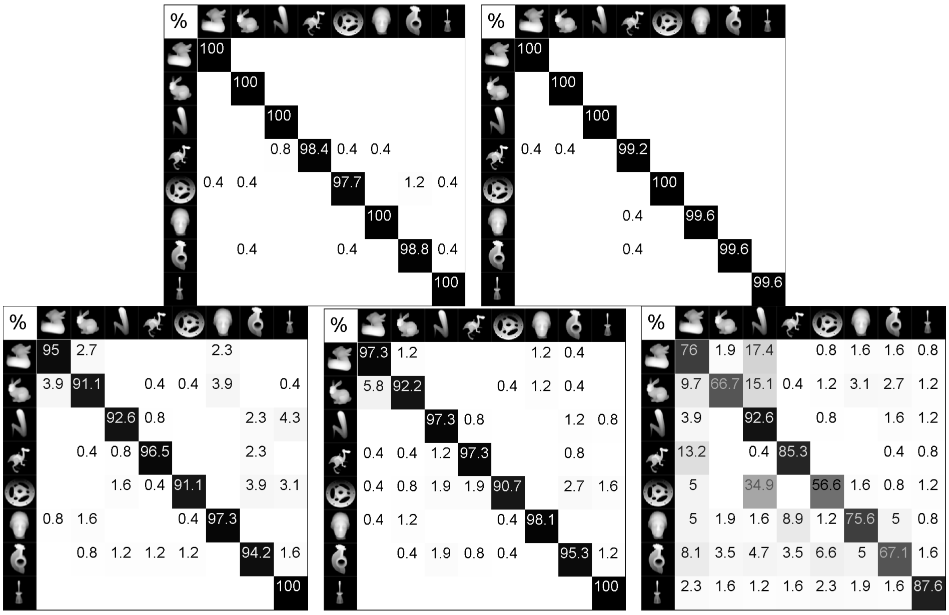
All methods (i.e. SSFE, SSFE+spin, MSFE, MSFE+spin, SIFT) are tested for object recognition. Four groups of experiments are carried out to test for orientation invariance, scale invariance and robustness to occlusions. 8 objects from the Stuttgart database (auto, bunny, chicken, ente, hub, eager, rocker, screwdriver) are used. For each model, 66 original range scans are used for training. 192 original, 192 scaled, 192 occluded and 192 both scaled and occluded range scans are used for testing. For eight objects, a total of 6672 range scans are used in these experiments.

7.2. 4.1 Recognition under Rotation

The database originally includes 192 testing images scanned from angles different than the 66 training images. Thus experimenting on the original training and testing models provides results on recognition performance of rotated models. Previously, [Li and Guskov 2007] and [Hetzel et.al. 2001] carried out these experiments on Stuttgart database and achieved recognition rates over %98 and %93 respectively. As seen from Table 13 and Figure 64, SSFE and SSFE+spin achieve similar performance. MSFE and MSFE+spin are very close, however the performance of SIFT is dramatically behind the other methods.

7.2.4.2 Recognition under Scaling

In the second part of the experiments, scaled versions of range images are used for testing. Each test image is scale to its ¼ area, by scaling each axis by ½. As seen from Table 13 and Figure 65, MSFE and MSFE+spin perform much better than SSFE and SSFE+spin since they use scale space of curvatures, i.e., they are scale invariant. Although SIFT is scale invariant as well, its performance is much behind, proving that SIFT is not descriptive for gray scale rendered range images.



%								
	100							
		100						
			100					
			0.8	98.4	0.4	0.4		
	0.4	0.4			97.7		1.2	0.4
						100		
		0.4			0.4		98.8	0.4
								100

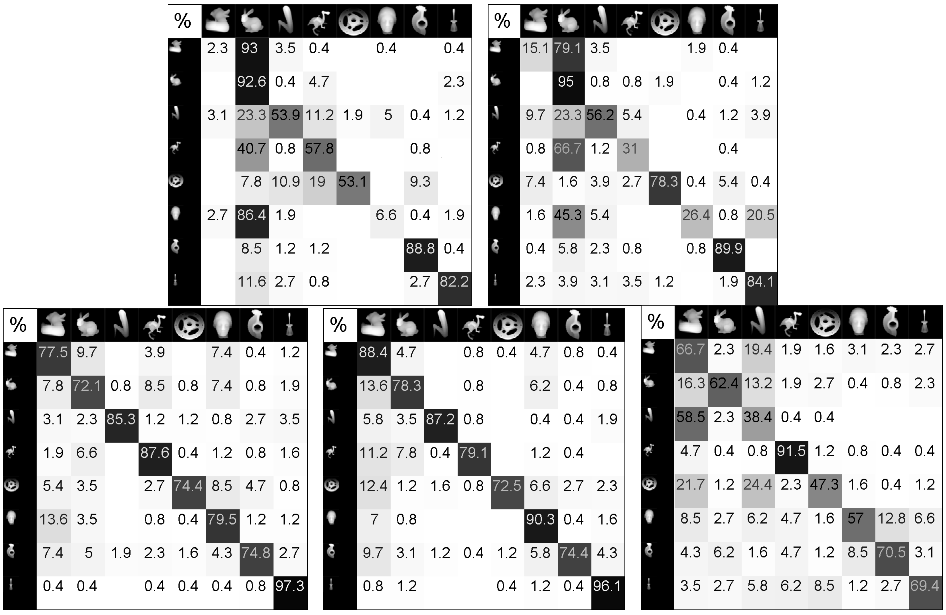
%								
	100							
		100						
			100					
	0.4	0.4		99.2				
					100			
				0.4		99.6		
				0.4			99.6	
								99.6

%								
	95	2.7				2.3		
	3.9	91.1		0.4	0.4	3.9		0.4
			92.6	0.8			2.3	4.3
		0.4	0.8	96.5			2.3	
			1.6	0.4	91.1		3.9	3.1
	0.8	1.6			0.4	97.3		
		0.8	1.2	1.2	1.2		94.2	1.6
								100

%								
	97.3	1.2				1.2	0.4	
	5.8	92.2			0.4	1.2	0.4	
			97.3	0.8			1.2	0.8
	0.4	0.4	1.2	97.3			0.8	
	0.4	0.8	1.9	1.9	90.7		2.7	1.6
	0.4	1.2			0.4	98.1		
		0.4	1.9	0.8	0.4		95.3	1.2
								100

%								
	76	1.9	17.4		0.8	1.6	1.6	0.8
	9.7	66.7	15.1	0.4	1.2	3.1	2.7	1.2
	3.9		92.6		0.8		1.6	1.2
	13.2		0.4	85.3			0.4	0.8
	5		34.9		56.6	1.6	0.8	1.2
	5	1.9	1.6	8.9	1.2	75.6	5	0.8
	8.1	3.5	4.7	3.5	6.6	5	67.1	1.6
	2.3	1.6	1.2	1.6	2.3	1.9	1.6	87.6

Figure 64. Confusion matrices of the methods: SSFE, SSFE+spin, MSFE, MSFE+spin, SIFT, from left to right respectively. Original testing images are used.



%								
	2.3	93	3.5	0.4		0.4		0.4
		92.6	0.4	4.7				2.3
	3.1	23.3	53.9	11.2	1.9	5	0.4	1.2
		40.7	0.8	57.8			0.8	
		7.8	10.9	19	53.1		9.3	
	2.7	86.4	1.9			6.6	0.4	1.9
		8.5	1.2	1.2			88.8	0.4
		11.6	2.7	0.8			2.7	82.2

%								
	15.1	79.1	3.5			1.9	0.4	
		95	0.8	0.8	1.9		0.4	1.2
	9.7	23.3	56.2	5.4		0.4	1.2	3.9
	0.8	66.7	1.2	31			0.4	
	7.4	1.6	3.9	2.7	78.3	0.4	5.4	0.4
	1.6	45.3	5.4			26.4	0.8	20.5
	0.4	5.8	2.3	0.8		0.8	89.9	
	2.3	3.9	3.1	3.5	1.2		1.9	84.1

%								
	77.5	9.7		3.9		7.4	0.4	1.2
	7.8	72.1	0.8	8.5	0.8	7.4	0.8	1.9
	3.1	2.3	85.3	1.2	1.2	0.8	2.7	3.5
	1.9	6.6		87.6	0.4	1.2	0.8	1.6
	5.4	3.5		2.7	74.4	8.5	4.7	0.8
	13.6	3.5		0.8	0.4	79.5	1.2	1.2
	7.4	5	1.9	2.3	1.6	4.3	74.8	2.7
	0.4	0.4		0.4	0.4	0.4	0.8	97.3

%								
	88.4	4.7		0.8	0.4	4.7	0.8	0.4
	13.6	78.3		0.8		6.2	0.4	0.8
	5.8	3.5	87.2	0.8		0.4	0.4	1.9
	11.2	7.8	0.4	79.1		1.2	0.4	
	12.4	1.2	1.6	0.8	72.5	6.6	2.7	2.3
	7	0.8				90.3	0.4	1.6
	9.7	3.1	1.2	0.4	1.2	5.8	74.4	4.3
	0.8	1.2			0.4	1.2	0.4	96.1

%								
	66.7	2.3	19.4	1.9	1.6	3.1	2.3	2.7
	16.3	62.4	13.2	1.9	2.7	0.4	0.8	2.3
	58.5	2.3	38.4	0.4	0.4			
	4.7	0.4	0.8	91.5	1.2	0.8	0.4	0.4
	21.7	1.2	24.4	2.3	47.3	1.6	0.4	1.2
	8.5	2.7	6.2	4.7	1.6	57	12.8	6.6
	4.3	6.2	1.6	4.7	1.2	8.5	70.5	3.1
	3.5	2.7	5.8	6.2	8.5	1.2	2.7	69.4

Figure 65. Confusion matrices of the methods: SSFE, SSFE+spin, MSFE, MSFE+spin, SIFT, from left to right respectively. Scaled versions of the testing images are used.

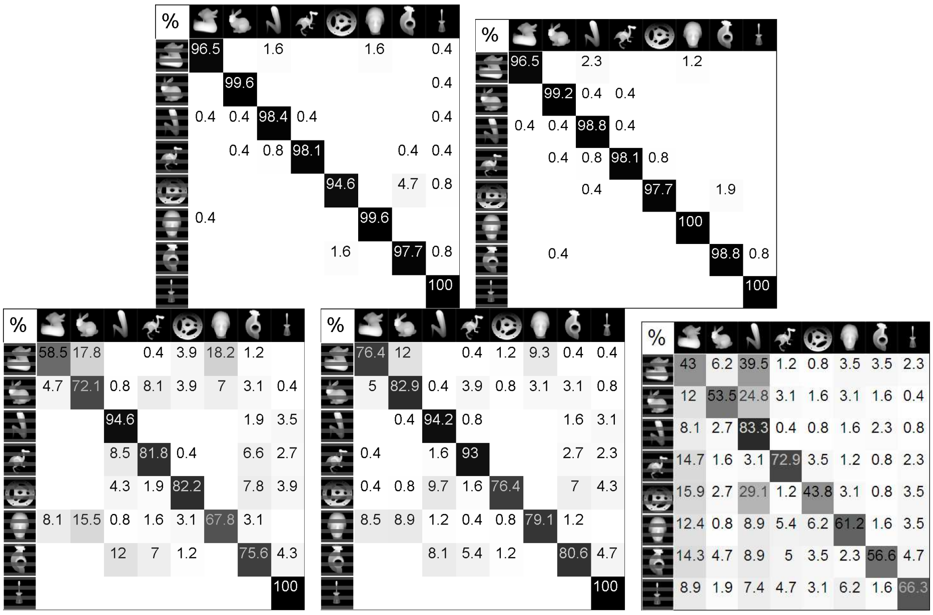


Figure 66. Confusion matrices of the methods: SSFE, SSFE+spin, MSFE, MSFE+spin, SIFT, from left to right respectively. Occluded versions of the testing images are used.

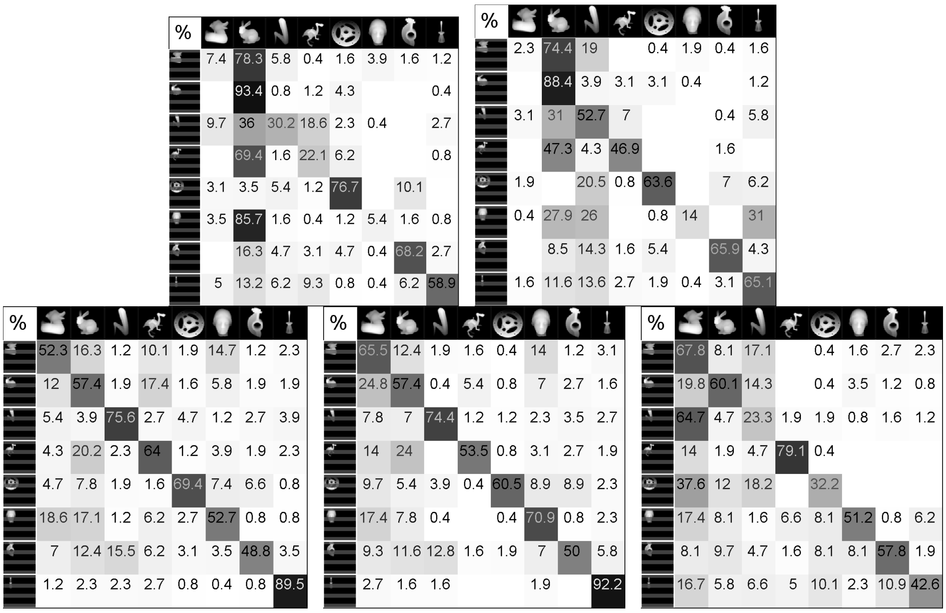


Figure 67. Confusion matrices of the methods: SSFE, SSFE+spin, MSFE, MSFE+spin, SIFT, from left to right respectively. Scaled and occluded versions of the testing images are used.

Table 13. Average Recognition Rates for All Methods in All Experiments

	SSFE	SSFE+spin	MSFE	MSFE+spin	SIFT
under rotation	99,37%	99,76%	94,71%	96,02%	75,91%
under scaling	54,55%	59,50%	83,58%	84,39%	62,79%
under occlusion	98,06%	98,64%	79,07%	85,32%	62,31%
under scaling and occlusion	45,30%	49,85%	63,71%	65,55%	51,70%

7.2.4.3 Recognition under Occlusion

The third part of the experiments analyzes the methods' robustness to occlusion, where %25 occluded version of each testing image is used. For each testing range image, randomly selected ¼th area of the range scan is occluded. Results (Table 13 and Figure 66) show that, SSFE and SSFE+spin performs tremendously good, since the test set is not scale varying and feature based methods are robust to occlusions. The performance of MSFE+spin shows that for occluded images, adding spin image descriptor to the feature vector affects the recognition performance considerably. The performance of SIFT on the other hand, is still behind other methods.

7.2.4.4 Recognition under both Scaling and Occlusion

The final stage of the experiments investigates the recognition performance of the methods under both scaling and occlusion. Each test image is first scaled by ¼th of its surface area and then occluded by ¼ of its area. Since testing under both scaling and occlusion is quite challenging, all methods are affected significantly. Still, having the advantage of being both scale invariant and local feature based (thus robust to occlusions) MSFE and MSFE+spin performs better than the other methods. SSFE's and SSFE+spin's performances are below %50 where SIFT's performance is close to theirs (Table 13 and Figure 67).

7.2.5 3D Object Registration Experiments

Surface registration is an intermediate but crucial step within the computer vision systems workflow. The goal of registration is to find the Euclidian motion between a set of range images of a given object taken from different positions in order to represent them all with respect to a reference frame. Registration in general can be divided into two: coarse registration and fine registration [Salvi et. al. 2006]. In coarse registration, the main goal is to compute an initial estimation of the grid motion between two clouds of 3D points using correspondences between both surfaces. In fine registration, the goal is to obtain the most accurate solution as possible. Needless to say that the latter method usually uses the output of the former one as an initial estimate so as to represent all range image points with respect to a common reference system. Then it refines the transformation matrix by minimizing the distance between the temporal correspondences, known as closest points.

For a wide literature survey on registration of range images reader may refer to [Salvi et. al. 2006].

In this thesis we perform coarse registration using the proposed scale invariant features. The homogenous transformation, which includes 3D rotation, translation and scaling between two range images, is estimated. However different from previous approaches not single features are matched as correspondences, instead the triples that were used to recognize object categories are used.

7.2.5.1 Triplet Correspondences

In the previous sections, n-node topologies of scale invariant features were constructed. Using these n-node topologies, transform invariant object recognition was performed and analyzed. The topology set included n=3 number of nodes (features), since experiments have proved that it is empirically the optimum. Up to now, only the best match, in other words the quantity of the all matches, between a set of training range images and a testing image was taken into consideration, since the problem was to recognize object categories. However for registration the quality of the matched features are also important. Only a general consensus between all matched triplets would give the true transformation and prove that the matched triplets are actually true features that represent the objects.

For this purpose using the extracted triplets in the previous subsection, a coarse registration is carried out using “random sample consensus” (RANSAC) method.

7.2.5.2 Random Sample Consensus (RANSAC) on Triplets

RANSAC is an abbreviation for "RANdom SAmple Consensus". It is an iterative method to estimate parameters of a mathematical model from a set of observed data which contains outliers. It is a non-deterministic algorithm in the sense that it produces a reasonable result only with a certain probability, with this probability increasing as more iterations are allowed. The algorithm was first published by [Fischler and Bolles 1981].

A basic assumption is that the data consists of "inliers", i.e., data whose distribution can be explained by some set of model parameters, and "outliers" which are data that do not fit the model. In addition to this, the data can be subject to noise. The outliers can come, e.g., from extreme values of the noise or from erroneous measurements or incorrect hypotheses about the interpretation of data. RANSAC also assumes that, given a (usually small) set of inliers, there exists a procedure which can estimate the parameters of a model that optimally explains or fits this data.

The RANSAC algorithm is often used in computer vision, e.g., to simultaneously solve the correspondence problem and estimate the fundamental matrix related to a pair of stereo cameras.

In this thesis, very similar to solving the correspondence problem in stereo images, the method is used to estimate the homogeneous transformation between two range images. Instead of using candidate point matches, candidate triplet matches are used. As explained in the previous section, the triplets are matched by some well-defined similarity measures. In order to eliminate the false matches, and obtained the transformation only between the true triplet matches, RANSAC is run.

RANSAC also requires a similarity measure to test for the homogenous transformation between two triplet matches. However for RANSAC, different from the similarity measures used the find candidate matches, only the spatial information is used. In other words, at any iteration of RANSAC, if there is candidate transformation, the absolute Euclidian difference between the first triplet and the transformed second triplet is used. The output of the RANSAC is a homogenous transformation vector, which defines the transformation between any two corresponding points on the registered range images, such that:

$$\overline{\mathbf{P}} = \mathbf{A}\cdot\mathbf{P} = \begin{bmatrix}\overline{P}_x \\ \overline{P}_x \\ \overline{P}_x \\ \overline{s}_p\end{bmatrix} = \left[\begin{array}{c|c} \mathbf{R}_{3x3} & \mathbf{T}_{3x1} \\ \hline \mathbf{0}_{1x3} & S \end{array}\right] \cdot \begin{bmatrix}P_x \\ P_y \\ P_z \\ s_p\end{bmatrix} = \begin{bmatrix}\mathbf{R}\cdot\mathbf{P} + s\mathbf{T} \\ S\cdot s_p\end{bmatrix} \tag{26}$$

7.2.5.3 3D Object Registration Results

In this subsection some experiments on registration using the proposed feature are presented. The first experiments is the simplest case, where there is only in-plane rotation between two artificial surfaces. There’s 135º in-plane rotation between the surfaces (Figure 68).

The result of RANSAC is given below together with the ideal transformation matrix. The results prove a quite successful coarse registration. In Figure 69, the matched features can also be seen. Colours, as usual, designate different feature types.

RANSAC result:

$$\left[\begin{array}{ccc|c} -0.6627 & 0.6479 & -0.1149 & 43.1966 \\ -0.6896 & -0.6807 & 0.0394 & 156.433 \\ 0.0342 & 0.1096 & 0.9344 & -5.0095 \\ \hline 0 & 0 & 0 & 1.0772 \end{array}\right],$$

and the ideal result:

$$\left[\begin{array}{ccc|c} -0.7071 & 0.7071 & 0 & 45 \\ -0.7071 & -0.7071 & 0 & 155 \\ 0 & 0 & 1 & -5 \\ \hline 0 & 0 & 0 & 1 \end{array}\right]$$

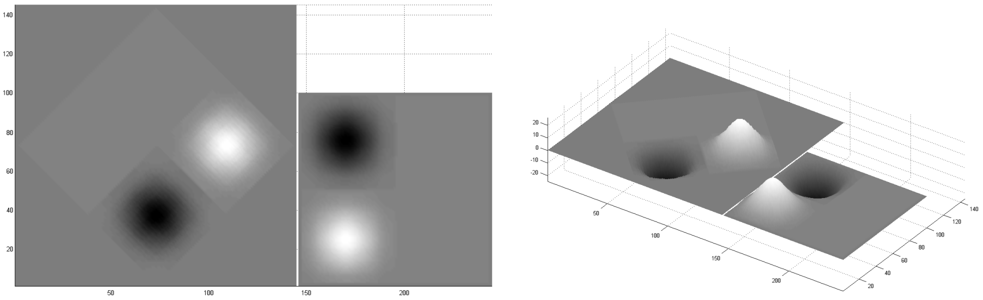


Figure 68. Two artificial surfaces with a Gaussian peak and a Gaussian pit, which are 135º in-plane rotated (rotation around z-axis) of each other.

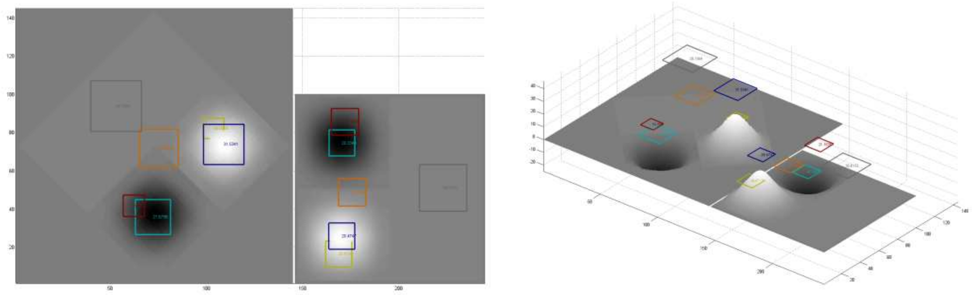


Figure 69. Matched features from the registered artificial range images.

The same experiment was also performed on range images from the Stuttgart database. Since these images are not captured with controlled rotation, we have chosen two images with rotation on a single axis. However the exact amount of rotation is unknown (Figure 70).

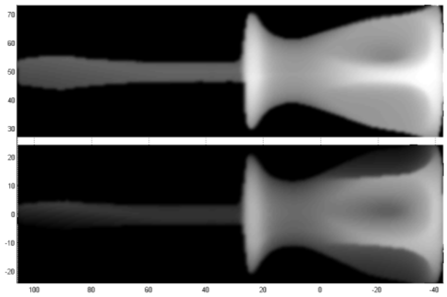


Figure 70. Two range images of the screwdriver from Stuttgart database. There is rotation around y-axis.

The result of RANSAC is given below together with the ideal transformation matrix. The results on Stuttgart database also prove a quite successful coarse registration. In Figure 71, the matched features can also be seen. Colours, as usual, designate different feature types.

RANSAC result:

$$\left[\begin{array}{ccc|c} 0.8616 & -0.0265 & 0.2065 & 0.06336 \\ -0.0032 & 0.9691 & -0.0105 & -0.4065 \\ -0.1914 & 0.0068 & 0.8159 & 8.8006 \\ \hline 0 & 0 & 0 & 1.0042 \end{array}\right],$$

and the ideal result (α° of rotation assumed):

$$\left[\begin{array}{ccc|c} \cos(\sim\alpha) & 0 & \sin(\sim\alpha) & 0 \\ 0 & 1 & 0 & 0 \\ \sin(\sim\alpha) & 0 & \cos(\sim\alpha) & T_z \\ \hline 0 & 0 & 0 & 1 \end{array}\right]$$

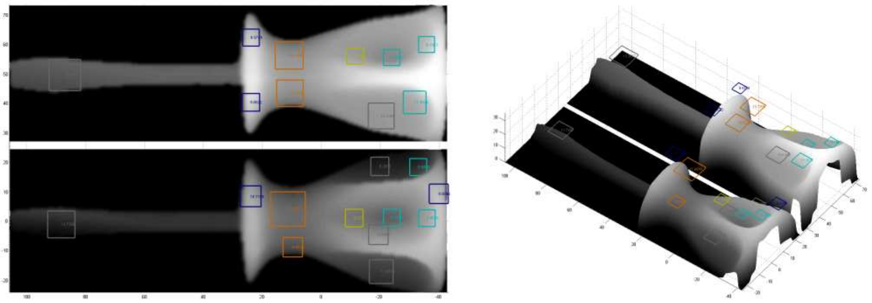


Figure 71. Matched features from the registered range images from Stuttgart Database.

Since the extracted features are scale invariant (and so as the triplets), it is possible to register scaled versions of range images and calculated the scaling ratio between to objects. For this purpose using the %25 scaled versions of the Stuttgart database objects (vs. the originals), registration is performed. The result of RANSAC is given below together with the ideal transformation matrix. The results on scaled range images demonstrate successful scale invariant coarse registration. In Figure 72, the matched features can also be seen. Colours, as usual, designate different feature types.

RANSAC result:
$$\left[\begin{array}{ccc|c} 0.9442 & -0.0143 & -0.0623 & 0.6810 \\ 0.0349 & 0.9899 & -0.0204 & 0.4130 \\ 0.0519 & -0.0067 & 0.9449 & 0.6304 \\ \hline 0 & 0 & 0 & 0.4955 \end{array}\right],$$

and the ideal result:
$$\left[\begin{array}{ccc|c} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 0 & \mathbf{0.5} \end{array}\right]$$

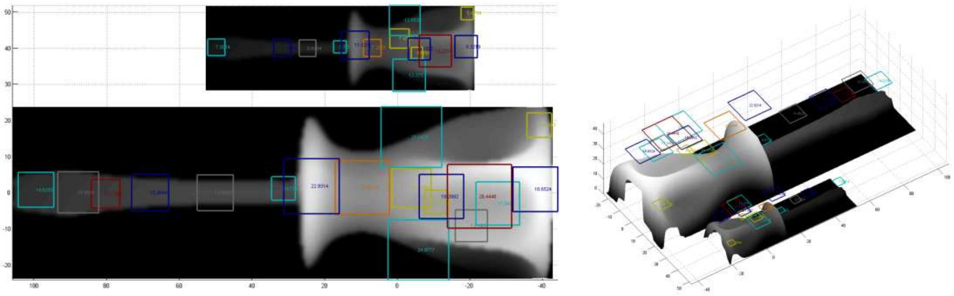


Figure 72. Matched features from the registered range images from Stuttgart Database. The surfaces are scaled versions of the same range image.

7.2.6 Conclusions on 3D Object Recognition and Registration

In order to test the scale invariance and occlusion robustness limits of the proposed method, ¼ scaled, %25 occluded and both ¼ scaled and %25 occluded versions of these scans are virtually created and tested. In order to classify the object categories, geometric hashing is used. So as to show the importance of scale space search of surface curvatures, in addition to multi-scale features, single scale features are also extracted and tested using the same classification method. Moreover, the results are also compared with SIFT, for which each original, scaled or occluded range image is rendered to a gray scale image.

If there is no scaling, single scale feature based methods (SSFE or SSFE+spin) are quite powerful for recognition on range images. Results prove that for scale varying databases, multi-scale features perform much better. The proposed method (MSFE+spin) performs best for experiments 2 and 4, where scaled range images are tested. Under occlusion, feature based recognition is hardly affected. On the other hand, SIFT shows poor performance. Although known to be quite potent for textured images, SIFT proves the opposite for gray scale rendered range images.

Results also show that using surface descriptors, such as spin images, may positively affect the performance of the extracted features. In experiment three, where occluded range images are tested, the performance of MSFE+spin is considerably higher than MSFE. When the range image is occluded, there are not enough quality features to represent the image. However, describing local surfaces with spin images when accurate effective regions are defined, the performance upgrades. This is not the case for SSFE and SSFE+spin, because the effective size extracted with single-scale features is not accurate. Thus, the surface description with spin images is faulty for the single-scale case.

7.3 Delineation of Slope Units Based On Scale And Resolution Invariant 3D Curvature Extraction

Landslides, which are one of the major geo-hazards threatening the settlements, occur on the slopes of the terrain. In order to predict landslide occurrences, previously occurring landslides are related with several landslide susceptibility parameters and landslide susceptibility maps are obtained. Preparation of a landslide susceptibility map requires selection of appropriate slope mapping unit, which can be regular grids (pixels), slope units, unique condition areas or some morphological features. Among these mapping units, use of slope units has several advantages over other mapping units. As the landslides occur on the slopes grid (pixel)-based approaches let the analyzer to consider the areas which are not prone to landslides and hence decrease the performance of susceptibility mapping algorithms. Using slope units as landslide susceptibility mapping unit overcomes this problem.

Slope unit is a method used to subdivide the space into regions based on certain hydrological criteria. Physically the slope unit can be considered as the left or right side of a sub-basin of a watershed. Therefore, slope unit can be identified by the intersection of a ridge line and a valley line. Usually partitioning of a region into sub-basins or slope-units is virtually impossible and such a partitioning requires high computational effort. Hence in this study, a scale and resolution invariant, fully automatic method using scale space of 3D curvature values is proposed for creating slope units to be used in landslide susceptibility mapping. All process as explained in the subsections is automatic. The performance of the proposed method is tested by comparing it with a conventional method of obtaining slope units.

At this point we would like to thank Dr. Arzu Erener and Dr. H. Şebnem B. Düzgün from Geodetic and Geographic Information Technologies Department at Middle East Technical University for their cooperation in this study. They helped us build the conventional method which is compared to our proposed landslide detection technique.

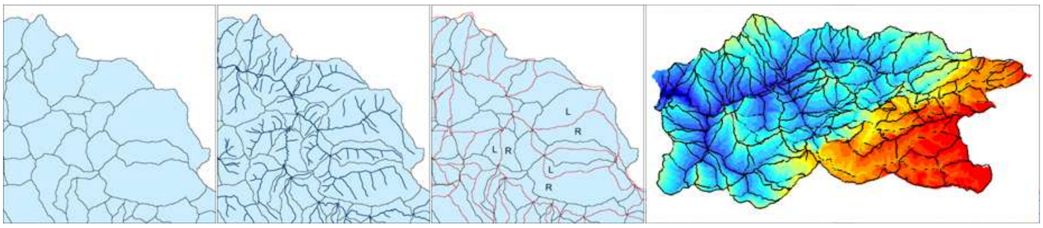
7.3.1 Slope Unit Generation by Conventional Method

A GIS-based hydrologic analysis and modelling tool, Arc Hydro [Maidment 2002], is employed to draw the dividing lines for identifying slope units. In this approach, the outline of the watershed polygon is obtained from digital elevation model (DEM) by using the hydraulic model tool, where the watershed boundary is the watershed divides or ridge line. The low elevation areas in the DEM which are surrounded by higher terrain and disrupting the flow of water in the path are filled in order to form watershed boundaries. The flow direction is calculated by examining the eight neighbours of a cell according to the eight direction pour point method [Multi-Watershed].

Then the associated flow accumulation grid is computed, which contains the accumulated number of cells upstream, for each cell in the input grid. In addition, a grid representing a stream network is created by querying the flow-accumulation for cell values above a certain threshold. This threshold is defined either as a number of cells or as a drainage area in square kilometers. In general, the recommended size for stream threshold definition is 1% of the maximum flow accumulation. Small threshold values result in denser stream network with large number of delineated catchments. Then the DEM and the flow accumulation are used to determine backward flow direction to identify cells that drain through a given outlet. In this backward tracing, the cells with homogeneous flow accumulation values are classified as watershed boundaries, which are later converted to polygons (Figure 73.a). After obtaining watershed boundaries, the next step is overlaying the watershed boundary by the drainage lines (Figure 73.b).

There is a need to divide the watershed polygon by the drainage line to obtain two slope units. Hence the drainage line is obtained by using reverse of the DEM, which is obtained by turning the high DEM values into low values, or vice versa. This process makes the original drainage line to be transformed into a watershed divide [Xie et. Al 2004]. The combination of the watershed deduced from DEM and watershed deduced from reverse

DEM gives the slope units (Figure 73.d). As can be seen from the procedure, the conventional method for obtaining slope units requires usage of GIS tools and operations with several stages.



a) b) c) d)

Figure 73. a) Watershed boundaries determined using the DEM. b) Drainage line overlaid with watershed of the region c) Watershed boundaries determined using the reverse DEM (shown with red lines) overlaid with watershed boundaries which are present on both the left and right side of a sub-basin. d) Slope unit of region obtained in 3D.

7.3.2 The Proposed Method

In this study, a scale and resolution invariant fully automatic method which is based on 3D curvature values is proposed for slope unit generation. The method considers the digital elevation model (DEM) of the terrain as a regular grid of 3D points lying over a u-by-v mesh (u being the latitude and v, the longitude), namely a depth map of the surface. On this 3D regular grid surface, first drainage lines and ridge lines are segmented according to their mean (H) curvature values and normal directions (n) in the scale space of the surface grid. Then these lines are connected in order to define the polygons of slope units. All the steps within the method are automatic and sequential. The method is detailed below step by step.

7.3.2.1 Re-sampling

Since the 3D coordinates are absolute in DEMs, the size information is always preserved on the surface. However, 3D curvature values computed over the surface are highly effected by the sampling (e.g. resolution) of the surface. Thus, a scale and resolution invariant method is required in order to compute curvature values for the points over the surface. For this reason, first the grid surface is re-sampled according to its scale (e.g. m meters / n points). In this study, the sample data has both latitude and longitude scale-resolution ratio of 20m/sample points. Thus if the proposed method is applied to a DEM surface with different sampling rate (resolution), then its scale-resolution ratio should be converted to 20m/points.

7.3.2.2 Scale Space of Curvatures

Surface curvature is simply the change in surface gradients. Both Mount Everest and a small hill may have the same curvature values at different scales but their curvature values at the same scale level are different. In order to examine a given surface at different scale levels, a scale pyramid is constructed. In this study Burt and Adelson’s Gaussian Pyramid method is applied [Burt and Adelson 1983] in order to extract a scale space of the surface. In this scale space, different sized elements become visible in different scales. For example, Mount Everest will be visible in a much higher scale than the small hill which will be visible in a lower one.

The drainage lines and the ridge lines over DEMs are similar sized structures and they reside at a certain layer of the scale space. Smaller or bigger structures having similar shapes (that is shapes having similar curvature values but different sizes) vanish in the scale level where they exist, and vice versa. For this reason, it was searched through the scale space and found out that the fourth level of Gaussian Pyramid (which was obtained by applying four successive Reduce [Burt and Adelson 1983] and four successive Expand [Burt and Adelson 1983] operations to the surface) includes the drainage and ridge lines. The original surface patch and its 4th level scale surface are shown in Figure 74.a and Figure 74.b.

7.3.2.3 Ridge and Drainage Line Extraction

The proposed method uses value H and vector n to determine if an area is a ridge line or a drainage line or neither of them. H and n at the fourth scale level for all points on the reduced surface are calculated. Then candidate ridge and drainage lines are obtained by thresholding as follows:

$$\begin{array}{ccc} \textit{drain.age line} & \textit{if} & (H > +0.0002)\,\&\,(\cos(\mathbf{n}_z) > 0.7) \\ \textit{ridge line} & \textit{if} & (H < -0.0002)\,\&\,(\cos(\mathbf{n}_z) > 0.7) \\ \textit{none} & \textit{if} & \textit{otherwise} \end{array} \tag{27}$$

Here n_z is the z component of the surface normal assuming that the z direction is always altitude (elevation) in the DEM. The selected regions are depicted in Figure 74.c. In order to obtain the actual ridge and drainage lines, thinning is applied over the candidate regions (Figure 74.c) and actual drainage and ridge lines are obtained (Figure 74.d). However some outlier lines are also resulted from small regions. In order to eliminate them, length of each connected line is calculated and small ones (smaller than 50 points for this case) are deleted (Figure 74.e).

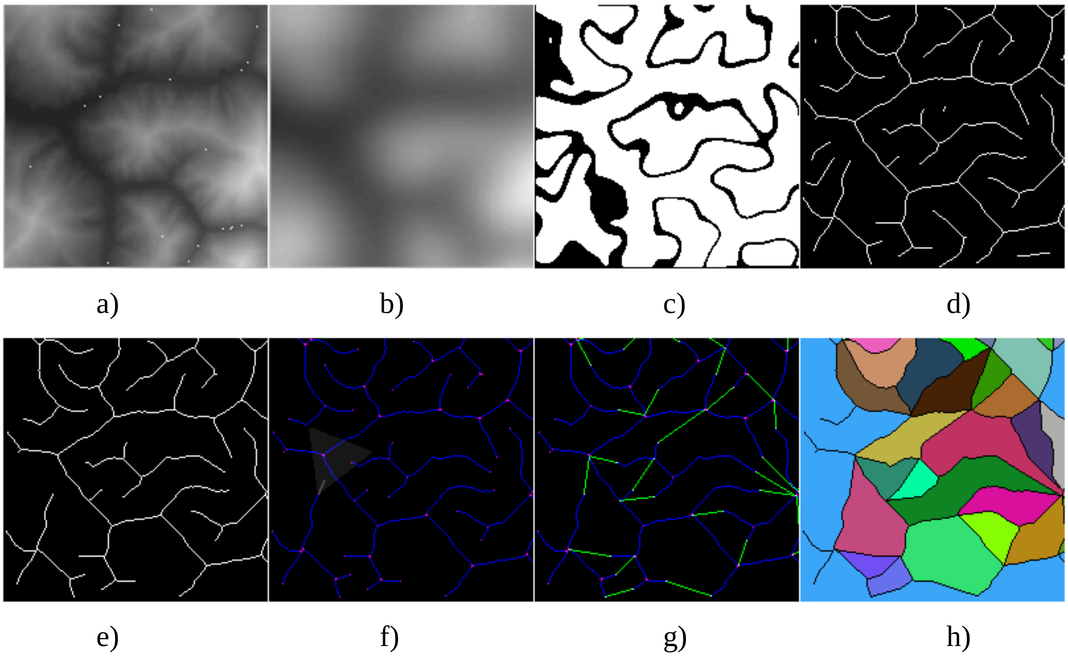


Figure 74. a) Original Surface Patch. b) 4th Scale Level of the surface patch c) Candidate ridge and drainage lines are shown in black. d) Ridge and drainage lines after thinning. e) Final ridge and drainage lines after the elimination of outliers. f) Detected line tips and corners are shown in pink and an example interest region for a tip is shown by a triangle. g) Line tips connected to available corners and tips are shown in green. h) Segmented landslide regions are shown in different colours.

7.3.2.4 Obtaining Polygons of Slope Units

In order to find slope unit candidates, the closed regions are extracted from these lines by using a very simple approach. First the corners and tips on each line are found by applying kernels over the binary image. Then for each tip, a tip direction is determined. Finally, a tip is connected to the closest corner which resides inside a region around the tip direction vector (i.e. ±30° degrees around the direction vector). If a corner does not exist in this region, then the tip is connected to the closest tip existing in this region. If a tip does not exist in the region, then no connection is made (Figure 74.f and Figure 74.g).

After line tips are connected to line corners and line tips, the slope unit regions surrounded by drainage and ridge lines are determined. Finally by finding the connected components and their contours, candidate regions are segmented and labelled as shown in Figure 74.h.

7.3.3 Results and Comparison

The conventional and proposed methods are overlaid at a small test area which is approximately 25 km^2 to represent the different and matching regions of generated slope units. Figure 3 presents the matching and different regions. The matching regions occupy 1.2 km^2 (Table 14). The proposed method occupies larger area of detailed slope units as compared to the conventional method (Figure 75). The 3D visualization of slope units from both methods is illustrated in Figure 76.

The proposed method is tested by using MATLAB® with an Intel Centrino® Duo Computer in windows XP®. It takes less than a minute to make all the calculations including re-sampling, obtaining the 4t^h scale, curvature calculations, thresholding, morphological operations and region labeling for a surface patch including 512x256 points of nearly 50km^2 area. On the other hand in the conventional method the process of DEM filling, flow direction, flow accumulation, stream network determination and watershed boundary determination requires use of different GIS tools which may take longer time depending on the resolution of the DEM and area of the study region. Considering that further performance optimization on codes is plausible and the complexity of the algorithm is O(n^2) in Big-O notation, clearly the method is far faster than the conventional methods.

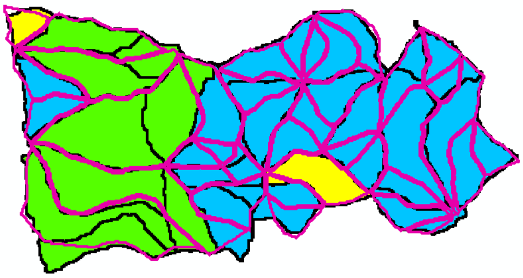


Figure 75. The overlay of conventional and proposed method outputs. Legend: black line: conventional slope unit, pink line: proposed method slope unit, yellow colour: similar regions, blue colour: generalized units of conventional method or detailed units of

proposed method, green colour: detailed units of conventional method or generalized units of proposed method.

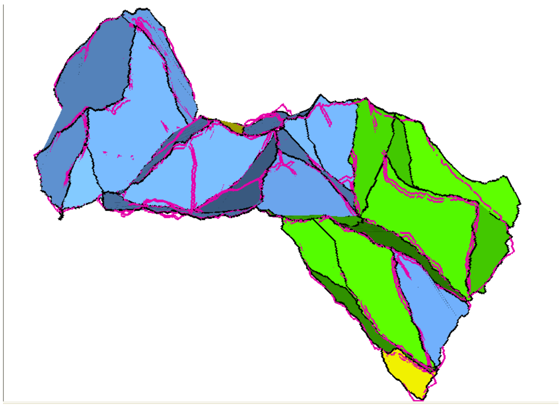


Figure 76. 3D view of slope units from proposed and conventional methods Legend: black line: conventional slope unit, pink line: proposed method slope unit.

Table 14. Similar and different slope unit areas.

Slope Unit	Area (km^2)
Matching	1.2
Detailed in Conventional Method Detail but generalized in Proposed Method	9.55
Generalized in Conventional Method general but detailed in Proposed method	14.62
Sum	25.37

CHAPTER 8

CONCLUSIONS AND FUTURE DIRECTIONS

8.1 Conclusions

In this thesis, a generic, scale and resolution invariant method to extract 3D features from 3D surfaces, is proposed. The method extracts features from 3D surfaces using scale spaces of surface curvatures. The basic idea behind the technique is to extract features independent of their scale and/or resolution, without losing the metric size information. The extracted features are tested for their registration, pose estimation and various classification capabilities.

Studies, which use scale-space of 3D surfaces or 3D curvatures, tried to extract features independent of scale and/or sampling. Many of these studies followed the path of SIFT, thus applied the “difference of Gaussian” (DoG) approach when creating the scale-space. In this thesis, we approach the problem from another perspective, where connected components of voxels within the curvature scale-space are used. We discuss the advantages of using this method, and conclude that it may have advantages in terms of better localization and robustness to noise, according to the controlled experiments performed on artificial scans.

Furthermore, the metric information, such as the metric size of a bump on a 3D surface in millimetres, is necessarily required for certain applications like robotics, 3D-slam or object recognition. Since range images provide the metric size, a feature obtained from these signals can have metric information. The proposed feature extraction method also provides this metric size information, independent of the resolution of the original image or the type of the scanner. Therefore, the extracted features are not just salient points on the 3D surface, but meaningful structures with their type, size, and orientation information.

Using the proposed scale invariant feature detector, a transform invariant surface topology representation was proposed. The topological representation consisted the relative properties of the nodes extracted from the object surface, thus it was covariant under certain transformations like rotation and scaling. Using this representation, we experimented on a number of implementations.

Our first experiments were on 3D face detection. We have used the Bosporus database which includes more than 20 different poses and facial expressions of 78 subjects, which was very suitable for testing the proposed method against various transformations. The method was compared with four representative methods in the literature. The results show that the proposed method is more successful in terms of detection percentage, localization accuracy and computational efficiency compared to the other methods.

Using another well-known 3D facial database, the FRCG v1.a, we have tested the pose estimation capabilities of our feature extraction method. In order to make controlled tests on pose estimation, virtually rotated versions of FRGC v1.a, frontal facial scans were created using a software constructed in our laboratory, METU-CVIS. The software was a 3D virtual scanner program, where range scanner output can be created and exported. The rotations were estimated with around ±2º error in x and y axes and ±4º error in z axes. Although the estimation capabilities were not fine, but coarse; the results showed that the representation allowed us to make a quick and efficient pose estimation of 3D faces.

In order to test the recognition capabilities of the proposed method, a transform invariant n-node topology definition was also proposed. Using n largest nodes extracted from the 3D surface, a transform invariant topology was defined. Each node represented an extracted feature, thus encapsulated type, size and orientation information. Before using this transform invariant n-node topology for recognition tests, we sought for the correct way to construct our curvature scale space. Since surface type definitions (such as pit, peak) may be defined using HK or SC curvatures, we did an initial study to compare the recognition capabilities of the two curvature scale spaces. Although shape index (S) was scale invariant in nature and mostly preferred in the literature, if curvature scale space were used to extract surface features, we have observed that HK curvature space gave better results. As a consequence of these results we have used HK curvature scale space in extracting our 3D surface curvatures in our succeeding experiments.

In order to test the scale invariance and occlusion robustness limits of the proposed method, ¼ scaled, %25 occluded and both ¼ scaled and %25 occluded versions of these scans are virtually created and tested. In order to classify the object categories, geometric hashing is used. So as to show the importance of scale space search of surface curvatures, in addition to multi-scale features, single scale features are also extracted and tested using the same classification method. Moreover, the results are also compared with 2D SIFT, for which each original, scaled or occluded range image is rendered to a gray scale image.

For the sake of observing the benefits of using multi-scale features, the 3D features are extracted with and without scales-space search. First, the features are extracted using the conventional method, in which the surface is classified using the H and K curvatures obtained from the given resolution (SSFE method). Then, the features are extracted using the proposed multi-scale feature extraction method (MSFE method). In order to see the benefits of using feature point descriptors, spin images around the feature centres are also extracted and appended to the feature vector for both SSFE and MSFE separately. (‘SSFE+spin’ and ‘MSFE+spin’). The effect regions of the spin images are designated by the radii of the features.

Results prove that for scale varying databases, multi-scale features perform much better. The proposed method (MSFE) performs best for experiments, in which scaled range images are tested. If there is no scaling, single scale feature based methods (SSFE or SSFE+spin) are quite powerful for recognition on range images. On the other hand, 2D SIFT shows poor performance. Although known to be quite potent for textured images, SIFT proves the opposite for gray scale rendered range images.

Results also show that using surface descriptors, such as spin images, may positively affect the performance of the extracted features. In experiment three, where occluded range images are tested, the performance of MSFE+spin is considerably higher than MSFE. When the range image is occluded, there are not enough quality features to represent the image. However, describing local surfaces with spin images when accurate effective regions are defined, the performance upgrades. This is not the case for SSFE and SSFE+spin, because the effective size extracted with single-scale features is not accurate and so as the surface description with spin images.

We have also run a couple of experiments in order to evaluate the proposed representation for 3D object registration. For this purpose, a modified version of RANSAC algorithm was proposed, which used triplets of nodes. Similar to facial pose estimation, the results were no fine, but showed that the technique is suitable for coarse object registration.

Finally we have used scale space of curvatures in order to detect landslide regions from digital elevation models. This method was compared to a conventional method (watershed) which was computationally very expensive. Result showed that the curvature scale-spaces provide valuable enough information for successful landslide detection.

In our final conclusions in this thesis, it is important to note that scale-space approach is crucial for any pattern recognition application. Recent studies show that the use of multi-scale features is increasing. Not only the academics society but also industry is showing more interest in the concept. In addition, the depth information is becoming more important and many new hardware are being used by important industrial application, such as time-of-flight (ToF) cameras. Consequently, we strongly believe that research on this field will continue to have importance for a while. In the next section, possible future directions are discussed.

8.2 Future Directions

The fields of pattern recognition and computer vision have considerable number of open problems. In this thesis, we seek answers to some particular problems such as improved feature scale invariance and more robust object representations. We apply the proposed solutions to certain applications. Firstly, it would be a good idea to test the proposed methods for robotics applications where succeeding scenes are to be registered to each other or objects in a real-world scene are to be recognized. ToF cameras can be used for such experiments and capabilities of the proposed method can be evaluated.

The experiments in thesis compares the results to 2D SIFT, using gray scaled rendered range images. Another way to benchmark the proposed methods would be to make comparisons to recently proposed methods. Although none of the recently proposed methods have yet become a standard as the original SIFT, such a comparison might give better clues on performance.

Another important contribution to the literature would be the construction of a scaled database, where real-world objects in cluttered environments are scanned. There are such samples of range images, however evaluating methods using a few images is generally not the sufficient way to understand their capabilities. Thus massive range images or ToF sequence databases of scaled objects within cluttered scenes will enable the researcher to build proper methods that can be used in real world robotics applications.

REFERENCES

[Akagündüz and Ulusoy 2007] E. Akagündüz and İ. Ulusoy, " 3D Object Representation Using Transform and Scale Invariant 3D Features," Proc. IEEE ICCV, Workshop on 3dRR, pp. 1-8, Oct 2007,

[Akgül et.al. 2007] C.B. Akgül, B. Sankur, F. Schmitt, Y. Yemez, “Density-Based 3D Shape Descriptors”, Eurasip Journal on Applied Signal Processing, Vol. 2007, Article ID 32503

[Arman and Aggarwal 1993] F. Arman, J. K. Aggarwal, “Model-based object recognition in dense range images – a review” J. of ACM Computing Surveys, (CSUR) vol 25, no 1, March 1993.

[Besl and Jain 1988] P. J. Besl and R. C. Jain (1988). Segmentation Through Variable-Order Surface Fitting, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 10, no. 2, pp. 167-192.

[Besl and Jain 1986] P. J . Besl and R. C. Jain. “Invariant Surface Characteristics for 3D Object Recognition in Range Images” Computer Vision, Graphics Image Processing, 33:33-80, 1986.

[Biederman 1987] I. Biederman, “Recognition-by-components: A Theory of Human image Understanding” Psychological Review, vol. 2, no 94.,pp 115-147, 1987.

[Bosporus Database 2007] Savran A., O. Çeliktutan, A. Akyol, J. Trojanova, H. Dibeklioğlu, S. Esenlik, N. Bozkurt, C. Demirkır, E. Akagündüz, K. Çalışkan, N. Alyüz, B. Sankur, İ. Ulusoy, L. Akarun, T. M. Sezgin, “3D face recognition performance under adversorial conditions,” in Proc. eNTERFACE’07 Workshop on Multimodal Interfaces, Istanbul, Turkey, 2007.

[Bowyer et.al. 2004] K.W. Bowyer, K.I. Chang, and P. J. Flynn: "A Survey Of Approaches To Three-Dimensional Face Recognition," Proc. 17th. IEEE Int. Conf. on Pattern Recognition, pp. 358-361, 2004, doi:10.1109/ICPR.2004.1334126.

[Bozkurt 2008] N. Bozkurt “2d/3d Imaging Simulator” Master Thesis submitted to the Graduate School of Natural And Applied Sciences of Middle East Technical University, 2008.

[Bozkurt et. al. 2009] N. Bozkurt, U. Halıcı, İ. Ulusoy, E. Akagündüz: “3D Data Processing for Enhancement of Face Scanner Data” in Proc. IEEE Signal Processing and Application Conference SIU 2009.

[Burt and Adelson 1983] P. J. Burt and E. Adelson (1983). The Laplacian Pyramid as a Compact Image Code" IEEE Trans. Communications, vol. 31, no. 4, pp. 532-540.

[Campbell and Flynn 2001] R. J. Campbell and P. J. Flynn, “A Survey of Free-form Object Representation and Recognition Techniques”, J. Computer Vision and Image Understanding, vol 81, no 2, February 2001.

[Cantzler and Fisher 2001] H. Cantzler, R. Fisher, "Comparison of HK and SC Curvature Description Methods," 3dim,pp.285, Third International Conference on 3-D Digital Imaging and Modeling (3DIM '01), 2001

[Chang et. al. 2005] K.I. Chang, K.W. Bowyer, and P.J. Flynn, "An Evaluation of Multi-modal 2D+3D Face Biometrics," IEEE Trans. Pattern Analysis and Ma-chine Intelligence, vol. 27, no. 4, pp. 619-624, April 2005, doi:10.1109/TPAMI.2005.70.

[Chang et. al. 2006] K.I. Chang, K.W. Bowyer, and P.J. Flynn, " Multiple Nose Region Matching for 3D Face Recognition under Varying Facial Expression," IEEE Trans. Pattern Analysis Machine Intelligence, vol. 28, no. 10, pp. 1695-1700, Oct 2006, doi:10.1109/TPAMI.2006.210.

[Chua and Jarvis 1997] C. Chua, R. Jarvis "Point Signatures: a new representation for 3D object recognition." Int. J. of Computer Vision, vol 25 , Issue 1 October 1997, pages: 63 – 85

[Colbry et. al. 2005] Colbry, G. Stockman, and A. Jain, "Detection of Anchor Points for 3D Face Verification," Proc. IEEE Conf. Computer Vision and Pattern Recognition, vol. 3, pp. 118-118, June 2005, doi:10.1109/CVPR.2005.441.

[Colombo et.al. 2005] Colombo, C. Cusano, R. Schettini, "3D Face Detection Using Curvature Analysis," J. Pattern Recognition, vol. 39, no. 3, pp. 444-455, Mar 2006, doi:10.1016/j.patcog.2005.09.009.

[Edelman 1997] S. Edelman “Computational Theories in Object Recognition”, Trends in Cognitive Science, vol 1, issue 8, November 1997.

[Fan et. al. 1986] T. J. Fan, G. Medioni, R. Nevatia, “Description of Surfaces from Range Data Using Curvature Properties”, Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp, 86-91, 1986.

[Fan et. al. 1986] T. Fan, G. Medioni, and R. Nevatia. Surface Segmentation and Description from Curvature Features. In *Proc. 1987 DARPA Image Understanding Workshop*, pages 351-359, 1987.

[Fan et. al. 1989] T. J. Fan, G. Medioni, R. Nevatia, “Recognizing 3-D Objects Using Surface Descriptors”, IEEE Trans. Pattern Recognition and Machine Intelligence, vol 11, no 11, November 1989.

[Fischler and Bolles 1981] M. A. Fischler and R. C. Bolles "Random Sample Concensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography", Comm. Of the ACM. Vol. 24 pp.381-395, 1981

[Flynn and Jain 1988] P. J. Flynn and A. K. Jain. Surface Classification: Hypothesis Testing and Parameter Estimation. In Proc. IEEE Conf. Computer Vision and Pattern Recognition, pages 261-267, June 1988.

[Flynn and Jain 1989] P.J. Flynn and A.K. Jain, “On Reliable Curvature Estimation,” Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 110-116, June 1989, doi:10.1109/CVPR.1989.37837.

[FRGC1a] Facial Recognition Grand Challenge version 1.a, 2D and 3D facial Database.

[Frome et.al. 2004] A. Frome, D. F. Huber, R. Kolluri, T. Bülow, J. Malik, "Recognizing objects in range data using regional point descriptors." European Conf. on Computer Vision ECCV 2004, vol 3, pp. 224-237

[Hetzel et.al. 2001] G. Hetzel, B. Leibe, P. Levi, B. Schiele, “3D Object Recognition from Range Images using Local Feature Histograms”, in Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR'01), 2001.

[Horn 1984] B.K.P. Horn, "Extended Gaussian Images", Proceedings of the IEEE, pp. 1671-1686, December, 1984.

[Hu and Zhu 2010] W. Hu and S.C. Zhu “Learning a Probabilistic Model Mixing 3D and 2D Primitives for View Invariant Object Category Recognition”, Proc. IEEE Conf. Computer Vision and Pattern Recognition, 2010.

[Inspeck-web] Inspeck Mega Capturar II http://en.souvr.com/product/200712/324.html (last visited December 2007)

[Ittner and Jain 1985] D. J . Ittner and A. K. Jain. “3-D Surface Discrimination from Local Curvature Measures” In *Proc. IEEE 1985 Conf. Computer Vision and Pattern Recognition*, pages 119-123, May 1985

[Johnson and Hebert 1999] A. E. Johnson and M. Hebert, “Using Spin Images For Efficient Object Recognition In Cluttered 3D Scenes," IEEE Trans. Pattern Analysis Machine Intelligence, vol. 21, no. 5, pp. 433-449, May 1999, doi:10.1109/34.765655.

[Kittler et.al. 2005] J. Kittler, A. Hilton, M. Hamouz, and J. Illingworth, "3D Assisted Face Recognition: A Survey of 3D Imaging, Modelling and Recognition Approaches," Proc. IEEE Conf. Computer Vision and Pattern Recognition, vol. 3, pp. 114- 114, June 2005, doi:10.1109/CVPR.2005.377.

[Koenderink 1990] J.J. Koenderink . “Solid Shape” MIT Press, Cambridge, Massachusetts, USA, 1990

[Koenderink and Doorn 1992] J.J. Koenderink and A. J. Doorn. “Surface shape and curvature scale”, Image Vis. Computer., vol. 10, no. 8, pp. 557–565. 1992

[Lamdan and Wolfson 1988] Y. Lamdan and H. J. Wolfson, “Geometric Hashing: A General and Efficient Model-Based Recognition Scheme”, Proc. Second Int. Conf. Computer Vision (Tampa, FL), pp. 238-249, Dec 1988

[Li and Guskov 2007] X. J. Li, I. Guskov, “3D Object recognition from range images using pyramid matching”, Proc. ICCV’07 Workshop on 3D Representation for Recognition (3dRR-07), 2007, pp. 1-6

[Li and Hancock 2004] Y. Li, E. R. Hancock: Face Recognition using Shading-Based Curvature Attributes. ICPR (3) 2004: 538-541.

[Lindeberg 1994] T. Lindeberg “Scale-Space Theory in Computer Vision”, Kluwer Academic Publishers/Springer, Dordrecht, Netherlands, 1994.

[Lindeberg 1998] T. Lindeberg “Feature Detection with Automatic Scale Selection”, Int. J. of Computer Vision, vol 30, no 2, pp. 76-116, 1998

[Lo and Siebert 2009] T.W.R. Lo and J.P.Sieberta, "Local feature extraction and matching on range images 2.5D SIFT ", J. Computer Vision and Image Understanding, Volume 113, Issue 12, December 2009, pages 1235-1250

[Lowe 2004] D. G. Lowe, "Distinctive Image Features from Scale-invariant Key points," Int. J. Computer Vision, vol. 60, no. 2, pp. 91-110, Nov 2004.

[Lu and Jain 2005] X. Lu and A.K. Jain, "Multimodal Facial Feature Extraction for Automatic 3D Face Recognition," Technical Report MSU-CSE-05-22, Comp. Sci. and Eng., Michigan State Univ., August 2005.

[Lu and Jain 2006] X. Lu and A.K. Jain, "Automatic Feature Extraction for Multiview 3D Face Recognition," Proc. 7th IEEE Int. Conf. on Automatic Face and Gesture Recognition, pp. 585-590, Apr. 2006, doi:10.1109/FGR.2006.23.

[Maidment 2002] Maidment R.D "Arc Hydro GIS for Water resources." ESRI Press, 380 New York Street, Redlands, California. 2002 ISBN-13:978-1-58948-034-6

[Merchán et. al. 2008] P. Merchán, A. Adán, S. Salamanca, “Identification and Pose under Severe Occlusion in Range Images” Int. Conf. On Pattern Recognition ICPR 2008: pp. 1-4.

[Multi-Watershed] The multi watershed deliniation method : GIS software in support of regional watershed analysis. (http://hydrology.neng.usu.edu/mwdtool/MWDManual.pdf)

[Nagamine et.al. 1992] T. Nagamine, T. Uemura, and I. Masuda. "3D Facial Image Analysis for Human Identification," Proc. 11th ICPR Int. Conf. Pattern Recognition, Vol.I. Conference A: Computer Vision and Applications, ICPR 1992.

[Osman et. al. 2004] M. K. Osman, M. Y. Mashor, M. R. Arshad, “A Review on 3D Object Representation and Recognition”, 2004

[Pauly et.al. 2003] M. Pauly, R. Keiser, M. Gross “Multi-scale Feature Extraction on Point-Sampled Surfaces”, Proc. EUROGRAPHICS’03, vol. 22, no. 3, 2003

[Roth and Winter 2008] P. M. Roth and M. Winter, “Survey of Appearance-based Methods for Object Recognition”, Technical Report ICG-TR-01/08, Inst. for Computer Graphics and Vision, Graz Univ. of Tech, Austria, January, 2008.

[Ruiz-Correa 2001] S.Ruiz-Correa, L.Shapiro,M.Meila, "A new signature-based method for efficient 3D object recognition"

[Salvi et. al. 2006] J. Salvi, C. Mataboch, D. Fofi, J. Forest, "A Review of Recent Range Image Registration Methods with Accuracy Evaluation", J. Of Image and Vision Computing, Vol.25, No.15, June 2006

[Simon and Seitz 2007] I. Simon and S. M. Seitz, “A Probabilistic Model for Object Recognition, Segmentation and Non-rigid Correspondence” Proc. IEEE Conf. Of Computer Vision and Pattern Recognition, pp 1-7, 2007

[Stein and Medioni 1992] F. Stein, G. Medioni ,Structural Indexing, Efficient 3-D Object Recognition, IEEE Transactions on Pattern Analysis And Machine Intelligence, Vol. 14, No. 2, February 1992

[Stuttgart Database] IPVR Department Image Understanding, Stuttgart Range Image Database, http://range.informatik.uni-stuttgart.de/

[Sun et.al. 2001] Y. Sun, M. A. Abidi, "Surface Matching by 3D Point’s Fingerprint," ICCV, vol. 2, pp.263, Eighth International Conference on Computer Vision (ICCV'01) - Volume 2, 2001

[Torre-Ferrero et.al 2009] C. Torre-Ferrero et.al. "A similarity measure for 3D rigid registration of point clouds using image-based descriptors with low overlap" ICCV 2009

[Tuytelaars and Mikolajczyk 2007] T. Tuytelaars and K Mikolajczyk “Local Invariant Feature Detectors: A Survey”, Foundations and Trends in Computer Vision, vol. 3, no. 3, 2008,

[Worthington and Hancock 2001] P. L. Worthington and E. R. Hancock, “Object Recognition Using Shape-from-shading,” IEEE Trans. Patterns Analysis Machine Intelligence, vol. 23, no. 5, pp.535-542. 2001

[Xie et. al. 2004] Xie M., Esaki T. and Zhou G., 2004. GIS-Based Probabilistic Mapping of Landslide Hazard Using a Three-Dimensional Deterministic Model, Natural Hazards 33: 265–282.

[Yang et.al. 2002] M.H. Yang, D.J. Kriegman, and N. Ahuja, "Detecting Faces In Images: A Survey," IEEE Trans. Pattern Analysis Machine Intelligence, vol. 24, no. 1,pp. 34-58, Jan 2002, doi:10.1109/34.982883.

[Zhao et.al. 2000] W. Zhao, R. Chellappa, A. Rosenfeld, and P. Phillips, "Face Recognition: A Literature Survey," Technical Report CAR-TR-948, CfAR, Mary-land Univ., 2000.

[Zhong 1997] Y. Zhong "Intrinsic Shape Signatures: A shape descriptor for 3D object recognition." ICCV 2009

CURRICULUM VITAE

PERSONAL INFORMATION

Surname, Name: AKAGÜNDÜZ, Erdem

Nationality: Turkish (TC)

Date and Place of Birth: 2 September 1979 , İskenderun

Marital Status: Married

email: akagunduz@ieee.org

EDUCATION

Degree	Institution	Year of Graduation
MS	METU Elec. & Electronics Engineering D.	2004
BS Minor	METU Department of Sociology	2001
BS	METU Elec. & Electronics Engineering D.	2001
High School	Bornova Anadolu High School, İzmir	1997

WORK EXPERIENCE

Year	Place	Enrolment
2009- …	ASELSAN-MGEO-Image Proc. Dep.	Senior Engineer
2008-2009	Univ. of York, UK	Research Fellow
2001-2008	METU Elec. & Electronics Engineering D.	Research Assistant
2001-2002	ETA Technologies A.Ş, METU Technopolis	Project Engineer

FOREIGN LANGUAGES

Advanced English

Beginners German

PUBLICATIONS

Book Chapters

- Erdem Akagündüz and İlkay Ulusoy, "*3D Object Registration and Recognition using Range Images*" from the book "Object Recognition", InTech, February 2011

International Journals

- İlkay Ulusoy, Erdem Akagündüz, Murat Yirci, "*Anatomical and Dynamic Volume Spline Model Applied to Facial Soft Tissue*", J. Computer Animation and Virtual Worlds, 2011 (*under publication during writing of this thesis*)
- İlkay Ulusoy, Erdem Akagündüz, Fidan Sabuncuoğlu, Serkan Görgülü, Özlem Üçok, "*Use of Dynamic Volume Spline Method to Predict Facial Soft Tissue Changes Associated with Orthognathic Surgery*", J. Oral Surgery, Oral Medicine, Oral Pathology, Oral Radiology and Endodontology (OOOOE), Vol 110, Issue 5, 2010.
- Erdem Akagündüz, İlkay Ulusoy, "*3D Object Recognition from Range Images Using Transform Invariant Object Representation*" J. IET Electronic Letters, Vol. 46, Issue 22, 2010.
- Erdem Akagündüz, İlkay Ulusoy, "*3D Face Detection Using Transform Invariant Features*" J. IET Electronic Letters, Vol. 46, Issue 13, 2010.
- İlkay Ulusoy, Erdem Akagündüz, Gerhard-Wilhelm Weber "*Inverse Solution for Parameter Estimation of a Dynamic Volume Spline Based Forehead Skin Model*" J. Inverse Problems In Science And Engineering, Vol 18, Issue.5, 2010.
- İlkay Ulusoy, Erdem Akagündüz, Gerhard-Wilhelm Weber "*Estimation of Parameters for Dynamic Volume Spline Models*" J. Inverse Problems In Science And Engineering, Taylor & Francis, (Special Issue Int. Conf. Engineering Optimization EngOpt2008) Vol. 18, Issue 4, 2010.

International Conferences, Oral Presentations

- Erdem Akagündüz, İlkay Ulusoy, "*Scale and Orientation Invariant 3D Interest Point Extraction Using HK Curvatures*" Workshop on 3D Representation for Recognition (3dRR-09) in conjunction with ICCV 2009.

- Erdem Akagündüz, Arzu Erener, İlkay Ulusoy, H. Şebnem B. Düzgün, "*Deliniation Of Slope Units Based On Scale And Resolution Invariant 3D Curvature Extraction*", IEEE International Geoscience & Remote Sensing Symposium, July 6-11, IGARSS 2008.
- Erdem Akagündüz, İlkay Ulusoy, "*3D Object Representation Using Transform and Scale Invariant 3D Features*" Workshop on 3D Representation for Recognition (3dRR-07) in conjunction with ICCV 2007.
- Erdem Akagündüz, İlkay Ulusoy, "*Extraction of 3D Transform and Scale Invariant Patches from Range Scans*" 2nd Beyond Patches Workshop "Patches Everywhere" Workshop in conjunction with CVPR 2007.
- Erdem Akagündüz, İlkay Ulusoy, Nesli Bozkurt, Uğur Halıcı, "*A Physically-based Facial Skin Model to Simulate Facial Expressions on Digitally Scanned 3D Models*" 22nd International Symposium on Computer and Information Sciences, 2007. ISCIS 2007.
- Erdem Akagündüz, Uğur Halıcı, "*Simulation and Synchronization of Human Facial Expressions and Lip Motion for Turkish Syllables*", International 12. Turkish Symposium on Artificial Intelligence and Neural Networks, TAINN 2003.

International Conferences, Poster Presentations

- Erdem Akagündüz, Ömer Eskizara, İlkay Ulusoy, "*Scale-Space Approach For The Comparison Of HK And SC Curvature Descriptions as Applied to Object Recognition*" IEEE International Conference on Image Processing, Nov 7-9, ICIP 2009
- Özlem Üçok, İlkay Ulusoy, Erdem Akagündüz, Erol Akın, Fatih Örs "*Semi Automatic Tool for 3D CT/MR Segmentation of Facial Tissue Layers*" 14th Congress of BaSS, 9th Scientific Congress of BgDA, Varna, Bulgaria, 6-9, May 2009.
- Erdem Akagündüz, Uğur Halıcı "*Animation Of Lip Motions And Facial Expressions Using 3d Homeomorphic Models*" 4th Indian Conference on Computer Vision, Graphics and Image Processing, ICVGIP 2004.

National Conferences, Oral Presentations

- Erdem Akagündüz, İlkay Ulusoy “*HK ve SC Eğrilik Tanımlarının 3B Nesne Tanıma Amacıyla Ölçek Uzaylarında Karşılaştırılması*” IEEE 17. Sinyal İşleme ve Uygulamaları Konferansı, SIU 2009.
- Ömer Eskizara, Erdem Akagündüz, İlkay Ulusoy “Geometrik Kıyım Yöntemi ile 3B Nesne Tanıma” IEEE 17. Sinyal İşleme ve Uygulamaları Konferansı, SIU 2009.
- Nesli Bozkurt, Uğur Halıcı, İlkay Ulusoy, Erdem Akagündüz, “*Yüz Tarayıcısı Verilerinin İyileştirilmesi için 3B Veri İşleme*” IEEE 17. Sinyal İşleme ve Uygulamaları Konferansı, SIU 2009.
- Erdem Akagündüz, Ömer Eskizara, İlkay Ulusoy “*Yüz Dokusunun Biyomekanik Özelliklerinin Çıkarılması İçin Çoklu Kamera Bazlı Bir Deney*” 13. Biyomedikal Mühendisliği Ulusal Toplantısı, BIYOMUT 2008.
- Erdem Akagündüz, İlkay Ulusoy “*3B Yüzlerin Ölçek ve Dönüşümden Bağımsız Öznitelikler ile Gösterilmesi*” IEEE 16. Sinyal İşleme ve Uygulamaları Konferansı, SIU 2008.
- Ahmet Oğuz Öztürk, Uğur Halıcı, İlkay Ulusoy, Erdem Akagündüz, "*Stereo Görüntü ve Yapısal Işık Kaynağı Kullanarak 3B Yüz Geriçatımı*” IEEE 16. Sinyal İşleme ve Uygulamaları Konferansı, SIU 2008.
- Erdem Akagündüz, İlkay Ulusoy, Nesli Bozkurt, Uğur Halıcı, “*2.5B Verinin Ölçekten Bağımsız Gösterilmesi*” IEEE 15. Sinyal İşleme ve Uygulamaları Konferansı, SIU 2007.
- Soner Büyükatalay, Uğur Halıcı, Erdem Akagündüz, İlkay Ulusoy, Kemal Leblebicioğlu."*Çoklu Görüntülerden 3B Yüz Modeli Oluşturma*" IEEE 14. Sinyal İşleme ve Uygulamaları Konferansı, SIU 2006.
- Soner Büyükatalay, Erdem Akagündüz, Uğur Halıcı, İlkay Ulusoy, Kemal Leblebicioğlu, “*3B Yüz Modeli Oluşturma ve Yüz İfadesi İçeren Konuşma Canlandırması*”, Polis Bilişim Konferansı, Ankara, PB 2005.
- Erdem Akagündüz, Uğur Halıcı, İlkay Ulusoy “*Türkçe Dudak Hareketlerinin Ve Yüz İfadelerinin 3 Boyutlu Ortamda Benzetimi Ve Bir Türkçe Ses Makinesiyle Eş Zamanlı Hale Getirilmesi*” IEEE 12. Sinyal İşleme ve Uygulamaları Konferansı, SIU 2004.

International Workshop Technical Reports

- Savran A., O. Çeliktutan, A. Akyol, J. Trojanova, H. Dibeklioğlu, S. Esenlik, N. Bozkurt, C. Demirkır, E. Akagündüz, K. Çalışkan, N. Alyüz, B. Sankur, İ. Ulusoy, L. Akarun, T. M. Sezgin, "*3D face recognition performance under adversorial conditions*," in Proc. eNTERFACE'07 Workshop on Multimodal Interfaces, Istanbul, Turkey, 2007. ENTERFACE 2007.