3D OBJECT RECOGNITION BY GEOMETRIC HASHING FOR ROBOTICS APPLICATIONS

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

BY

AYKUT HOZATLI

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

FEBRUARY 2009

Approval of the thesis:

3D OBJECT RECOGNITION BY GEOMETRIC HASHING FOR ROBOTICS APPLICATIONS

submitted by **AYKUT HOZATLI** in partial fulfillment of the requirements for the degree of Master of Science **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**

Assist. Prof. Dr. İlkay Ulusoy ____________________
Supervisor, **Electrical and Electronics Engineering**

Examining Committee Members:

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

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

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

Asst. Prof. Dr. AFŞAR SARANLI ____________________
Electrical and Electronics Engineering Dept., METU

SEFA AKYÜREK, MSc. in EEE ____________________
MGEO, ASELSAN Inc.

Date: 12.02.2009

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, Last name : Aykut HOZATLI

Signature :

ABSTRACT

3D OBJECT RECOGNITION BY GEOMETRIC HASHING FOR ROBOTICS APPLICATIONS

Hozatlı,Aykut

M.Sc., Department of Electrical and Electronics Engineering

Supervisor : Assist. Prof. Dr. İlkay Ulusoy

February 2009, 84 pages

The main aim of 3D Object recognition is to recognize objects under translation and rotation. Geometric Hashing is one of the methods which represents a rotation and translation invariant approach and provides indexing of structural features of the objects in an efficient way. In this thesis, Geometric Hashing is used to store the geometric relationship between discriminative surface properties which are based on surface curvature. In this thesis surface is represented by shape index and splash where shape index defines particular shaped surfaces and splash introduces topological information. The method is tested on 3D object databases and compared with other methods in the literature.

Keywords: 3D object recognition, Geometric hashing, Splash, Curvature based recognition

ÖZ

ROBOTİK UYGULAMALARINA YÖNELİK GEOMETRİK İNDEKSLEME YÖNTEMİYLE 3 BOYUTLU NESNE TANIMA

Hozatlı,Aykut

Yüksek Lisans, Elektrik ve Elektronik Mühendisliği Bölümü

Tez Yöneticisi : Yrd. Doç. Dr. İlkay Ulusoy

Şubat 2009, 84 sayfa

3 boyutlu nesne tanıma kavramının asıl amacı döndürme veya kaydırma işlemine maruz kalmış nesnelerin tanınabilmesidir. Geometrik İndeksleme, nesnelerin döndürme ve kaydırma durumlarından bağımsız, nesnenin yapısındaki özelliklerin etkili bir şekilde saklanmasını sağlayan yöntemlerden biridir. Bu tez çalışmasında Geometrik İndeksleme belirgin yüzey eğimlerinden elde edilen yüzey özelliklerinin geometrik ilişkilerinin saklanmasında kullanılmıştır. Bu çalışmada yüzey bilgisi belirli tip yüzeylerin tanınmasını sağlayan yüzey indeksi ve yüzeyin biçimsel bilgisini sağlayan sıçrama tekniğiyle ifade edilmiştir. Metod 3 boyutlu veritabanları üzerinde test edilmiş ve diğer metodlarla karşılaştırılmıştır.

Anahtar Kelimeler: 3 boyutlu nesne tanıma, Geometrik indeksleme, Eğimlere dayalı tanıma

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my supervisor Assist. Prof. Dr. İlkay ULUSOY for her guidance, valuable advices and support throughout the research.

With his advices and reviews, he helped me to improve my works, I would like to thank to Erdem Akagündüz.

I would like to thank to my lovely family, my father Fevzi, mother Özden and brother Aytac. They are always with me with their encouragement, support and patience.

I would also like to thank to my colleagues at Aselsan Inc. for their support, guidance and advices throughout this thesis work and to Aselsan Inc. for the facilities provided for the completion of this thesis.

.

TABLE OF CONTENTS

TABLE OF FIGURES

FIGURES

CHAPTER 1

INTRODUCTION

1.1. MOTIVATION

Object recognition tries to identify and locate a specified object in the given image or image sequences. Recognition concept has been essential matter for robotic applications. One fundamental point in vision based robotics application is to determine the presence of objects in the robot's point of view. If an object existed in its range, next step is to be able recognize it properly. Identification of materials or classification of bodies is the important rings of robot's manipulation chain. Considering recognition purpose of vision based systems and robotics we have constructed a hybrid method by using different approaches.

In many cases there is much more information in the scene than we can use. Therefore only informative and distinctive part of the object information is utilized for the aim. We have tried to use as much as few and rich information for description of object. Stable and robust approaches are used at every step of the recognition process. We have used shape index parameter for feature extraction which is dimensionless and scale invariant parameter used to define all surface shapes. It is based on first and second order derivatives of the surface.

For description of the point sets, splash approach is used. Method is based on obtaining surface normal relation of the points and their surrounding regions. This method is also customized to have less computation cost.

All of the points are stored to a database in the form of hash table. Geometric hashing allowed us to construct view point independent and transformation invariant way for recognition. Hashing approach is used to speed up the matching process by splitting the process to two separate part, preprocessing and recognition.

The complete algorithm implemented in this thesis has reached to % 91.03 recognition rates which is promising result as compared to results obtained with same database.

1.2. PREVIOUS STUDIES

Object recognition has been a challenging problem since over two decades. It is one of the most fundamental goals of all vision researches. During this time period several methods have been developed like model based approaches which relied on priori exact knowledge of models [1][6][7][12], apperance based approaches which needs to have model scenes from several viewpoints [27], index based approaches in which salient features are stored to a database with specific indexing methods [1][18][20][27]. The main problems of the all recognition concept were common. Being able to recognize or identify objects when they have undergone transformations or have partial occlusions. These problems have been handled at some of the researches. However several of them have concluded with successfull results, they had an expensive computation to meet all requirements of transform and occlusion independent recognition method [24][25]

Recognition process has some key steps for all methods. Description of objects as much as few but rich information is the core of the process to obtain stable, robust and low cost recognition method. Considering this aim most of the researches concentrated on extraction of meaningful and distinctive points from scene or object. Extraction process of interest points or called features is still the one of the most challenging topic on recognition.

Using local features, in which obtained from small patches of the object are recent way of representations of the objects. Partial ambiguous informations from several patches are fused to obtain disctinctive representation of the objects. In few applications, reliable curvature estimation of object has provided viewpoint independent cue for 3D object classification. Representation of features based on curvature estimation is proposed by using mean and Gaussian curvatures in [11], shape index parameter in [3], spin image in [24][25].

Gaussian, mean curvatures and shape index have been proven that they are good local shape descriptors for 3D objects even object has partial occlusion or under rigid transformation. But only extraction of these surface points is not sufficient for representation of the object. Further processing of these points is necessary for stable representation. In between mentioned curvature estimators, shape index method had precedence to the mean (H) and Gaussian (K) curvatures. Besides mean and Gaussian curvatures are so correlated [3] and their classification method based on comparing HK values respect to zero, shape index provided broad and specific ranges for each of the local shapes except planar surfaces.

Spin image method has been used for defining 2D array for each oriented points on the object. This approach have computed distance relation between selected point and all other vertices to construct a two dimensional array. Computation cost of the spin image is high and it is closely dependent on resolution of the scene.

Splash approach is introduced and used in [1]. Method is based on finding surface normals around a given point and relating them with the center point's surface normal. Firstly a circular region is defined around given point and sample points are determined on this circle. Each point's surface normal is computed and relation with the center point's surface normal is obtained. Method provided the direction of surface and gave information about structure of the local patch.

The point signature is also similar to the splash method. A spherical region is defined around a given point. The surface patch that is intersecting with spherical region determined the working area. Two planes, the tangent plane to the given point and the plane covering the working area are defined. The signed distance between these planes give the point signature of the given point.

Indexing based methods became popular as a solution to the computation complexity of the matching algorithms. One of the most commonly used method is geometric hashing and was proposed in [18][20]. The problem was obvious; complexity is raised exponentially while using brute force in matching and hashing approach reduces the matching cost significantly by accessing only to relevant information.

Splitting the recognition algorithm in to two parts as preprocessing and recognition provides faster responses. A database is constructed during preprocessing stage and searched during recognition stage.

The algorithms that have significant properties are explained in following and in the respective chapters in detail.

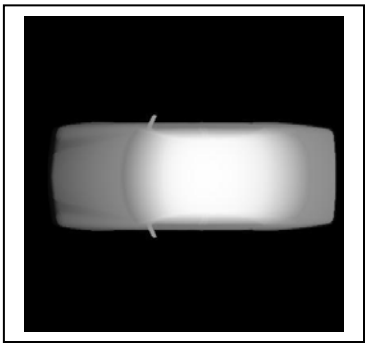
1.3. SCOPE OF THE THESIS

The main purpose of the thesis to construct a robust recognition system by using different approaches together. Methods which are invariant to geometric transformations and robust to occlusion are used.

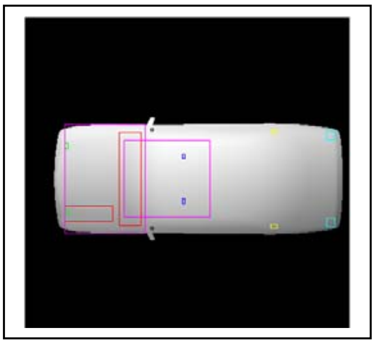
The other important purpose is to develope a computationally efficient method.All these are satisfied by the 3D geometric hashing method where orientation invariant 3D surface features are used. The subsections of the recognition method are shown in Figure 1.

Preprocessing Stage

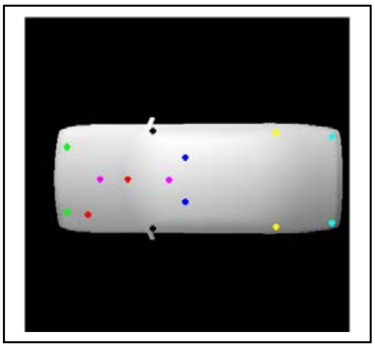
1 - Get an object scene and read the 3D data of object from the range images



2- Extract local features from the scene which represent the object



3- Define the extracted regions as point sets



4- Define object centered coordinate frames and establish geometric relationships between points

5- Store geometric information to a database in the form of a hash table

Recognition Stage

5- Apply steps 1 to 4 of preprocessing stage to target (test) scene

6- Retrieve information from the hash table by using indexes obtained from step 5.

7- Check the retrieved information for verification purpose

8- Vote an object for each matching feature

9- Find the model which receives the highest vote and determine it as the matching model

Figure 1. Scope of Thesis

1.4. ORGANIZATION OF THESIS

In Chapter 2 theoretical backround of recognition methods is summarized and compared to eachother. This chapter gives information about previously used algorithms and techniques.

In Chapter 3, 3D geometric hashing method is discussed. The preprocessing and the recognition stages are explained.

In Chapter 4 proposed method for 3D object recognition is explained. Feature extraction, Splash and 3D geometric hashing method and how they are used is discussed.

Chapter 5 is devoted to experiments and results part of the thesis. Works experienced during implementation of the thesis are explained and results are given in this part.

Chapter 6 provides a conclusion of the thesis.

CHAPTER 2

3D FEATURE DETECTION AND DESCRIPTION

2.1. FEATURE DETECTION AND EXTRACTION

Object recognition is used to classify the objects and distinguish one from another. The basic steps of object recognition are feature extraction, feature description and matching.

Feature extraction usually maps a larger variable space to a smaller feature space. It is used for the reduction of data to increase the algorithm speed or to cope with limited storage size. It is also used for performance improvement in accuracy. The other advantage of feature extraction is to improve of data understanding by extracting informative knowledge of data or by visualizing the data. These primitive features especially depend on the property of the input space. In various applications mathematical morphology, edge and corner detection, curvatures are selected to represent the input data. In our implementation, we have used local surface properties like surface curvatures. Surface curvatures are robust against variation of viewpoint and gives distinctive information about the object. Most of the curvature parameters are obtained using principal curvature parameter of the surface. Gaussian, mean curvatures and shape index are the most common examples of curvature estimator obtained using principal curvatures. These parameters will be explained in the following sections in detail.

2.1.1. TANGENT PLANE

Tangent plane of a surface is briefly a plane that contains all tangent vectors at a given point on the surface. In broader sense, a surface can be defined in parameterized form as z=f(x, y) or F(x, y, z) = c. If we assume P = (x_0, y_0, z_0) is a point on this surface and if f and F have continuous partial derivatives at the point (x_0, y_0) then we can say that f(x, y) has a tangent plane at point (x_0, y_0, z_0). The equation of tangent plane is defined as follows

$$\frac{df}{dx}(x_0, y_0)(x - x_0) + \frac{df}{dy}(x_0, y_0)(y - y_0) - (z - z_0) = 0 \tag{2.1}$$

The normal vector to the surface is an orthogonal vector to the tangent plane at point P on the surface.

$$k(x - x_0) + l(y - y_0) + m(z - z_0) = 0 \tag{2.2}$$

The normal vector is defined as **n** = (k, l, m).

By considering equation (2.1) and (2.2) the tangent plane is normal to the vector

$$n = (\frac{df}{dx}(x_0, y_0), \frac{df}{dy}(x_0, y_0), -1) \tag{2.3}$$

2.1.2. NORMAL AND PRINCIPLE CURVATURES

Normal curvature is the curvature that is projected to the tangent plane T and contained the surface normal vector ***n*** at the given point P. Normal curvature is a real number. The sign of the normal curvature value depends on the choice of the direction of the normal vector. If the curvature has a constant sign in either side of the normal vector, then it is said that surface lies through one side of the tangent plane. But if the curvature's sign changes by changing the normal vector direction then the surface is said to lie on both sides of the tangent plane [8]. This means that the surface has curves on both sides of the tangent plane. In case of changing normal curvature sign, we can have an opinion about the surface shape in a small neighbourhood of the given point P.

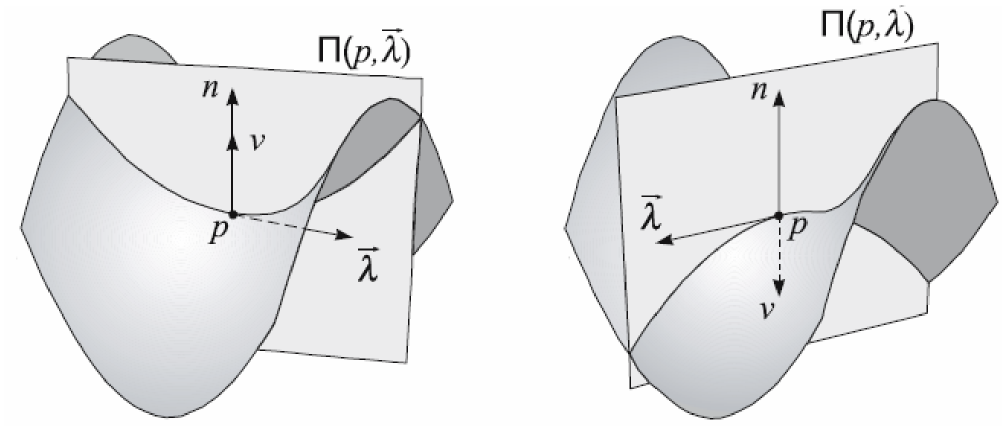


Figure 2. Curvatures with changing normal vectors. [8].

Figure 2 displays the parameters,where ***n*** is the normal vector, ***P*** is the given point for curvature computation, λ is the direction of tangent vector, ***v*** is the direction of curvature, Π is the plane passing through point ***P*** and tangent vector in the direction of λ.

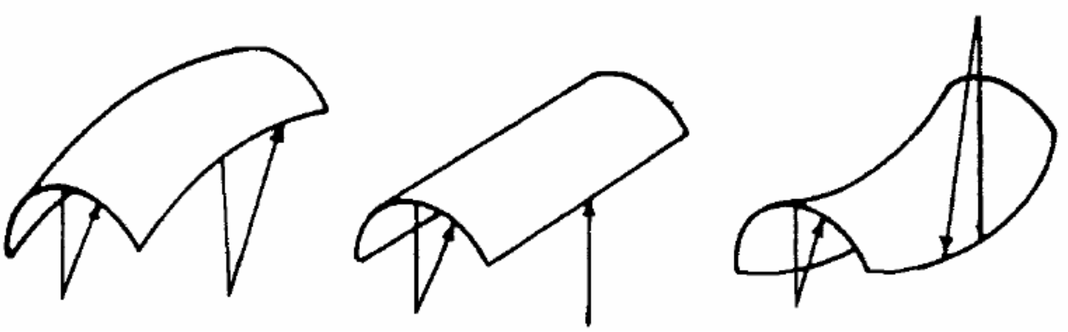
For determination of principal curvatures, first of all, tangent plane is obtained on the surface for the given point ***P**.* Since the tangent plane includes tangent vectors in all directions, we can intersect the surface with normal planes at each direction of the tangent vectors. By computing curvatures of possible intersections, we can obtain one maximum and one minimum curvature value. These maximum and minimum values of normal curvature are called principal curvatures, and named as κ_1 and κ_2. The sign of the principal curvature is positive if it has the same direction as the normal vector and negative for the other direction. And also vectors always lie perpendicular to eachother. Principal curvatures of a given point P on a surface give information about the shape of the local surface around point ***P***. From this information, we can describe the surface patch. By some well known descriptions such as Gaussian Curvature, Mean Curvature, Shape Index and Curvedness, principal curvatures are used to define the curvature estimators.

In the following sections these curvature estimators or classifiers are examined.

2.1.2.1. GAUSSIAN CURVATURE

Gaussian curvature ***K*** at a point on a surface is the product of extreme curvatures of the given point cut out by normal planes. The positive sign Gaussian curvature means that there is a peak or valley at the local surface of the given point. The negative values show that there is a saddle point. The zero curvatures define the surfaces where at least one of the principal curvatures corresponds to a flat region.[9].In Figure 3 some surface examples are given.

$$K = \kappa_1 . \kappa_2 \tag{2.4}$$



Positive Gaussian Zero Gaussian Negative Gaussian

Figure 3. Surface types for Gaussian Curvatures [10]

2.1.2.2. MEAN CURVATURE

The average of the principal curvatures defines the Mean Curvature, denoted by ***H.***

$$H = \frac{1}{2}(\kappa_1 + \kappa_2) \tag{2.5}$$

Also conversely if Gaussian and Mean Curvatures are given principal curvatures are obtained by

$$\kappa_1 = H + \sqrt{(H^2 - K)} \tag{2.6}$$

$$\kappa_2 = H - \sqrt{(H^2 - K)} \tag{2.7}$$

Gaussian and Mean Curvatures satisfy the following relation

$$K \le H^2 \tag{2.8}$$

Local shapes can be classified according to their Mean and Gaussian Curvatures.In their original work, Besl and Jain [11] calculated mean (**H**) and Gaussian (**K**) curvatures and classified each point on surface according to their Gaussian and mean curvature values as given in Table 1

Table 1 Surface classifications according to Gaussian and Mean Curvature values

	K>0	K=0	K<0
H<0	Peak	Ridge	Saddle Ridge
H=0	-	Flat	Minimal
H>0	Pit	Valley	Saddle Valley

2.1.2.3. SHAPE INDEX

One of the most commonly used techniques besides Gaussian and Mean Curvatures is the extraction of local surface shapes using shape index. It was introduced by Koenderink and Doorn in [2]. Koenderink and Doorn explained that Gaussian and Mean curvature extractions are not very informative for local surface shapes. Then they defined a new scale invariant parameter called shape index to represent local shapes. Shape index is a dimensionless numerical value that is defined in the range of [-1, +1]. It is expressed in terms of principal curvatures as follows.

$$S = \frac{1}{\pi} \arctan \frac{(\kappa 2 + \kappa 1)}{(\kappa 2 - \kappa 1)} \qquad \kappa 2 \leq \kappa 1 \qquad \textbf{(2.9)}$$

In the recent researches it has been shown that shape index gives quite well results on local surface shapes. This index defines all types of surface shapes except planar shapes where both principal curvatures have zero values [13]. The later applications [12] have also shown that extraction of local shapes by using shape index gives better results. Since shape index is invariant to rotation and translation it gives quite stable results even transformation is applied to the surface [5]. By shape index classification can be made better even when noise exists in the scenes. This makes the shape index classifications more distinctive over the Gaussian and Mean classification. Instead of having only zero thresholds like in Gaussian and Mean curvature classifications, shape index gives specific threshold ranges for each seperate surface shapes.

In the Figure 4 model scene and its corresponding shape index map are displayed. In the range image, dark pixels represent points away from the camera and light pixels represent points closer to the camera. Also larger shape index values define convex surfaces and smaller values define concave surfaces. In Figure 4 brighter points correspond to larger shape index values such as dome and ridge while darker pixels correspond to small shape index valued shapes such as rut and cup.

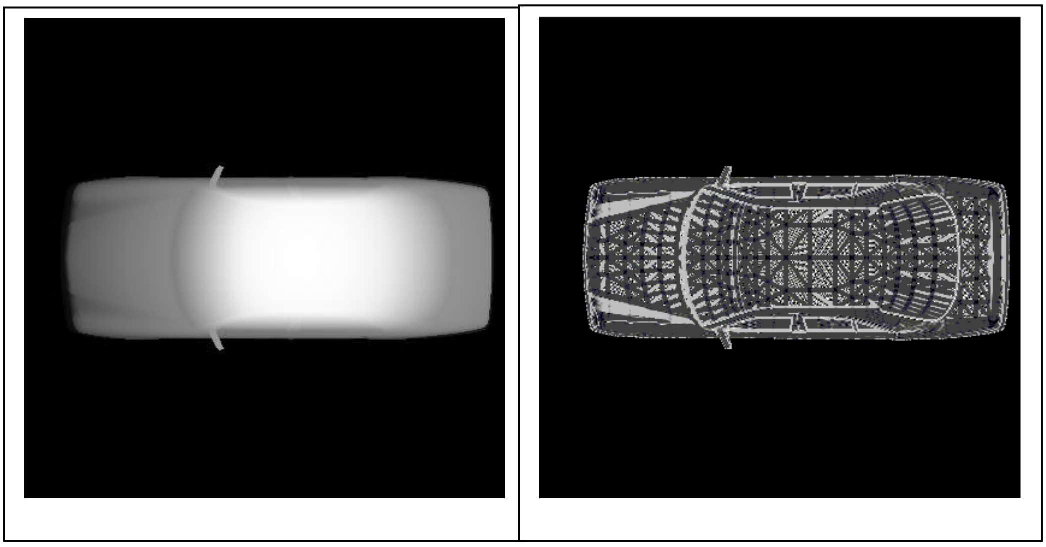


Figure 4. Range image of auto model, and auto model with its shape index values as grey level.

The shape index ranges and the corresponding topographic shapes are listed in the Table 2 .

In this table different surface shapes are shown with their corresponding shape index values. As seen from the table shape index is undefined at the planar surfaces because of principal curvatures at planar surfaces have zero values and makes the shape index value undefined.

Table 2 Surface classification according to shape index values.[6]

Class	Region-type	Shape-index
Dome	Elliptic	[5/8, 1)
Ridge	Parabolic	[3/8, 5/8)
Saddle ridge	Hyperbolic	[1/8, 3/8)
Plane	Hyperbolic	Undefined
Saddle-point	Hyperbolic	[−1/8, 1/8)
Saddle-rut	Hyperbolic	[−3/8, −1/8)
Rut	Parabolic	[−5/8, −3/8)
Cup	Elliptic	[−5/8, −1)

2.2. FEATURE DESCRIPTION

The next step of the recognition process is making computations for obtaining relationships between the extracted feature points. Feature description provides meaningful representation of the datasets obtained from the extraction stage. Each point is related to each other and at the end, a global identifier is obtained. Some methods are represented in the following sections. These methods rely on description of local surface patches to obtain global representation of the object. Each one of the approaches uses similar but exceptional attributes of the surfaces. Surface normals, signed distance measures and parametric specifications are some attributes used at selected methods.

2.2.1. SPIN IMAGE

Spin image is a 3D surface representation and local shape description. It is basically comprised 3D oriented points which contain 3D coordinates and surface normal information. Oriented points are used to construct object centered local frames. Frames are defined by using darboux frame approach with surface normals and principal curvatures of points [23]. Against geometric transformation, object centered frame approach provides a robust description of object points.

Algorithm is implemented assuming that the object is given in a polygonal surface mesh description. It is based on the fact that any point on the object can be represented with respect to a frame by using only two parameters Figure 5. Two parameter description of 3D points reduces the three dimensional space to two dimensional space.

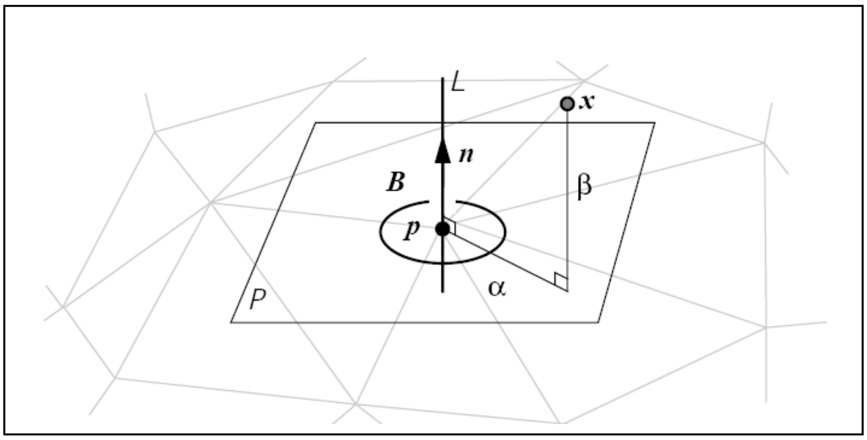


Figure 5. Representation of points with respect to a local base [23]

Description parameters are obtained by considering the scheme given in Figure 5. ***P*** tangent plane of the given oriented point *p*, ***L*** is the surface normal axis of the point *p*, ***n*** is the surface normal and ***x*** is the point on the surface outside of the oriented point *p*.

The spin image uses two parameters for each vertice on the object which are α and β. First parameter α is determined by computing the perpendicular distance of point ***x*** to the surface normal axis ***L***, which is also called as radial distance. Second parameter β is defined as perpendicular distance of point ***x*** with respect to the tangent plane ***P*** and is also called as axial distance. [23][24]

Since 3D information of the points are available as priori, the computation of α and β is as follows [24].

$$\alpha = \sqrt{\| x - p \|^2 - (n.(x-p))^2} \qquad \textbf{(2.10)}$$

$$\beta = n.(x-p) \qquad \textbf{(2.11)}$$

Computation of α and β with respect to the selected local frame provides the representation of all surface mesh vertices in 2D instead of 3D. Spin image of an object is projection of 3D points to 2D space. This interpretation is robust against transformations. A dense data is formed for each of the oriented point and is stored into 2D array. Each bin of the array corresponds to the spin image parameters (α and β) of the oriented point.

Two conditions need to be satisfied to store points into the 2D array. The first one is that, if a vertex has a distance below a predetermined value, the second one is that if the angle difference between oriented point's surface normal and the vertex point surface normal is below a threshold. If both conditions are satisfied then the point is added to the 2D array. Array bin address is found by using α and β parameters. If a new point is added to the array bin the previous value of that bin is incremented by using bilinear interpolation. This means that instead of incrementing the actual bin, surrounding bin values are incremented.

2.2.2. POINT SIGNATURES

Point signature is another approach for describing object surfaces in 3D scene. It basically relies on finding the signed distance values of a region around a given point. In this method, the sign concept makes the distance unit more distinctive than the absolute distance measurement.

For a given point *p* on the surface, a sphere is defined which is centered at *p* with radius *r*. The object surface intersecting with this sphere forms a 3D contour *C* on the surface. To describe the curved surface, a plane is fitted to the contour *C*. The plane constructed so that the distance of the points on the contour to the plane will be minimum. Plane normal vector is called n_1 and when *r* approaches to the zero, n_1 represents the unit normal vector of point p. Another plane is formed centered at point p is called P^l. This plane is formed by translating the contour plane until it will cover point *p*. Again when the radius approaches to zero, plane P^l will become tangent plane of point *p*. The distances of points on contour *C* to plane P^l determines the point signature. The distance is a signed value and its sign is determined such that if the normal vector and the point projection are at the same side the sign will be negative and if they are at either sides sign will be positive.

Instead of projection of all points on contour *C*, points on *C* are sampled with a specific step size defined in degrees so that sample point signatures are used as salient features. The recommended step size is 15° [26]. Starting point of sampling is determined by defining a vector from point *p* to the point which has the biggest positive distance. By starting from this point, point signatures are obtained for the samples and represented as discrete values.

Point signature is also presented as a transformation invariant solution for 3D surfaces. Significant number of interest points and their point signature profiles are needed to describe the whole scene. Object recognition is achieved by indexing the point signature values and comparing the model and the target scene features.

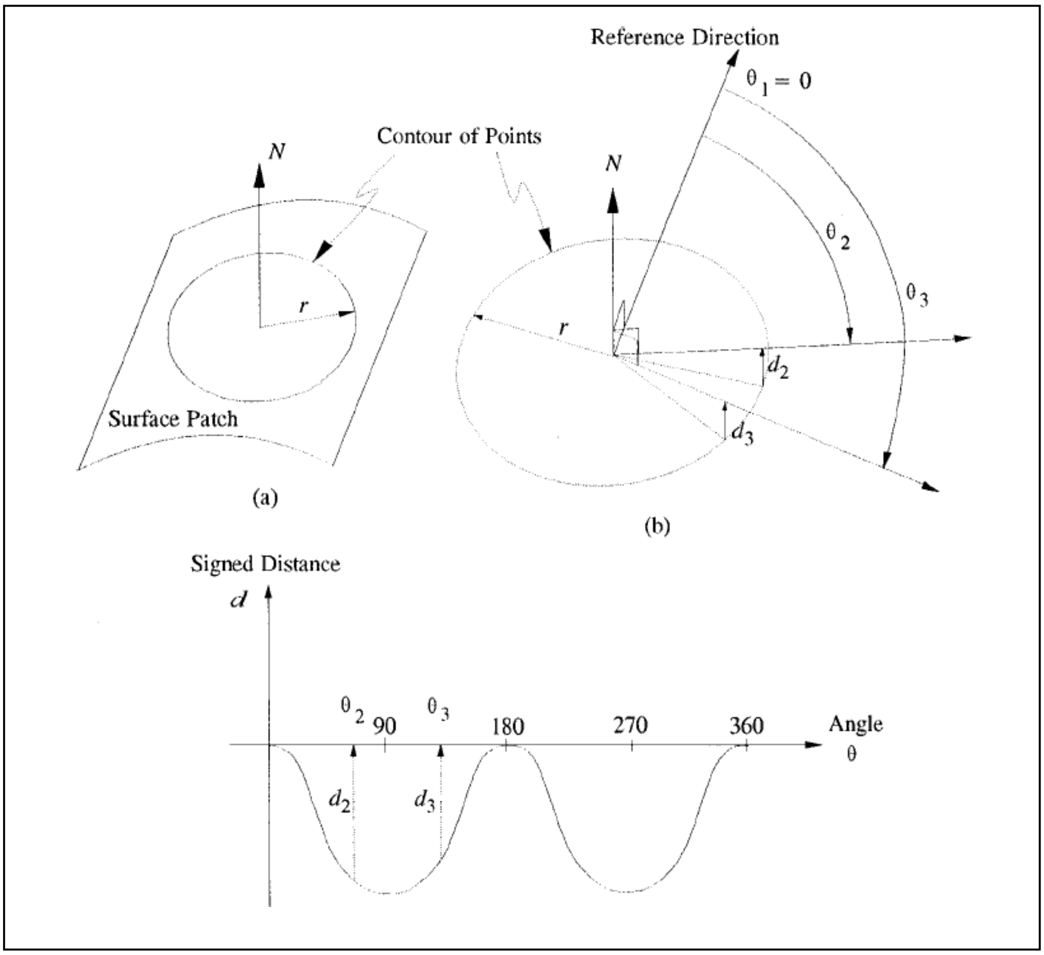


Figure 6. Representation of Point Signature Method [25]

2.2.3. SPLASH APPROACH

Surface information is commonly used because of its capability to include necessary information for robust distinction. Curvature based data extraction like in [3][4][11] satisfies stable, rich information for recognition process. By Stein and Medioni in [1] another approach, splash, is introduced. Considering surface orientation changes in local patches, they have used surface normal relations of points. Approach has also considered to meet the requirements of rotation, translation invariance and noise robustness in a realiable way. Idea was originated from picture of Harold Edgerton, 1936 Milk Splash (Figure 7).

In the splash idea, at a given point *p* on the surface, surface normal is computed and this normal is called as the reference normal of the splash (*n*). By considering point *p* as a center, a circular slice radius ρ (typically $10 < \rho < 25$) away from the point *p* is computed. By starting from an arbitrary point on the circle, sample points are determined with steps of Θ angles (typically $1° < \Theta < 15°$) [1]. At each sample point on the circle, a new surface normal is computed and each is called as n_θ. From the start point to the end, samples of surface normals are collected.

At the next step, each obtained surface normal is related to the reference normal. The angular difference between reference normal *n* and the splash normal n_θ is computed. This relation approximates the structure of the surface patch around point p.

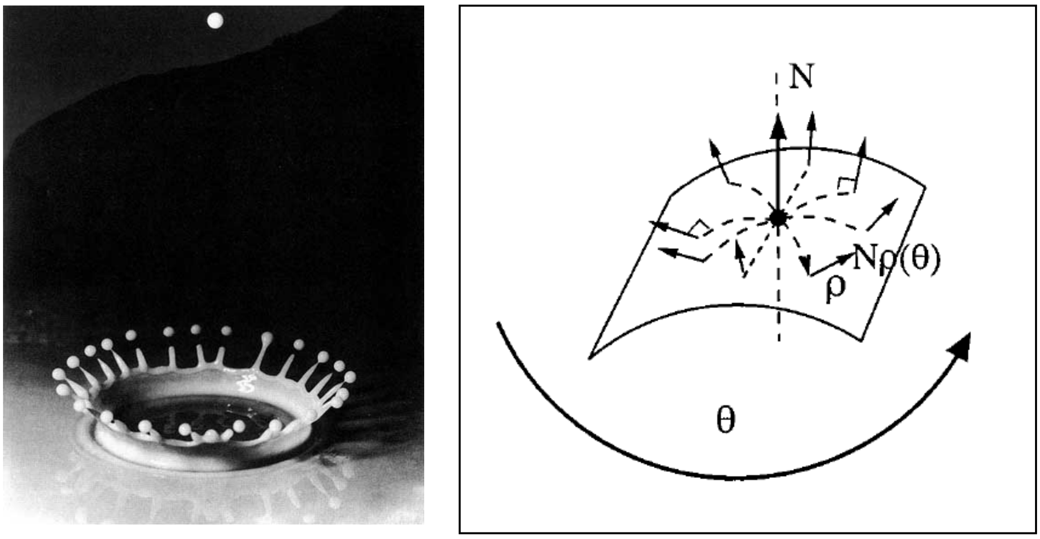


Figure 7. Milk Splash [2] and Splash Description [27]

Figure 7 shows the splash concept. By using normal vectors around a given point, surface topology is obtained.

Super splash is extended case of splash approach which is composed of splashes with different radii. With more than one splash profile, rich topology information can be obtained around a given point.

In the encoding part of Stein and Medioni approach, obtained splash values are used to define 3D curves. They have applied line fitting method to each curve and obtained number of lines with intersecting each other. Then they have used super segment approach in which the angle relation between lines is used as descriptors of the local patch. For the recognition step they have used these angle relations for index table storage.

CHAPTER 3

3D GEOMETRIC HASHING

In object recognition, matching stage can be extremelly slow when there is huge number of features in the given scene. Because of the exponential growth of the computation during search process, whole recognition performance is affected. Brute force methods check also redundant information during matching process. Indexing based recognition methods alleviate this shortcoming of brute force approaches. In the indexing based systems, distinctive properties of objects are stored to a database and then searched during matching phase. But instead of using brute force approach only relevant information is retrieved. Relevance is obtained by using target scene information. Geometric hashing is such a method that is based on indexing. Method splits the whole recognition process into two stages: preprocessing and recognition. This chapter is devoted to explain the geometric hashing algorithm. In the following sections it will be explained in detail.

The ultimate goal of object recognition is to be able to recognize objects even if they have undergone geometric transformation or had partial occlusion. Geometric hashing is one of the methods which represents a rotation and translation invariant approach and provides indexing of structural features of the objects in an efficient way. Structural or geometric features of an object could be points, lines, curvatures or their geometric relations.

The main advantage of geometric hashing is that, it introduces a recognition approach which considerably speeds up the process by splitting the recognition process into two stages. One of the stages is preprocessing and the other one is recognition.

The model information is obtained at the preprocessing step and then stored to a hash table. The content of the hash table is independent of the model scenes. Therefore model data encoding can be done seperately or offline. By doing extraction and hash table construction process offline, preprocessing stage does not effect the recognition time. For each iteration, which hash table bin will be used is determined according to the information obtained from model. At the preprocessing stage, all of the available models are analyzed and interest points are used to construct a hash table.

At the recognition stage, given test model is analyzed and information of the model is obtained again as in preprocessing step. Then the previously prepared hash table is searched to determine the matching models. Extracted information provides the access to only the relevant bin of the hash table without searching the whole table. Accessing only to the matching bins speeds up the search, because irrelevant or redundant information are discarded at the beginning.

Geometric hashing deals only with transformation invariant sets of model points. Feature extraction is not the scope of the algorithm. Only available point sets are used in the approach. The main idea behind the geometric hashing is that the extracted points have specific geometric relationships between each other. The distance between points are fixed even they have undergone translation or rotation. Also if we draw lines between points these lines are connected to eachother with fixed angles. Rotation and translation do not change these properties therefore the geometric features of the points remain constant.

The ultimate aim of this approach is representing the point sets by making them invariant to geometric transformation. This can be done by defining an orthogonal coordinate frame from triple point sets and representing all other points with respect to this defined frame.

Two stages of the geometric hashing algorithm are explained in the following sections in detail.

3.1. PREPROCESSING

Preprocessing stage can also be called as off-line stage. Because computations in this stage do not effect the recognition time. Feature extraction and description of training model image are performed to construct database or hash table before the recognition of a scene is applied. Since geometric hashing does not deal with feature extraction side, the input to the system should be transformation invariant point sets. Point sets can be obtained from the geometric characteristics or the physical quality of object model. For instance model scene can contain corners, edges, curvatures or regions that have some distinct characteristics. For geometric hashing input, these feature sets need to be represented as points. A point can define different features in applications as previously mentioned like intersections of edges, corners or center points if the feaure is a scattered region.

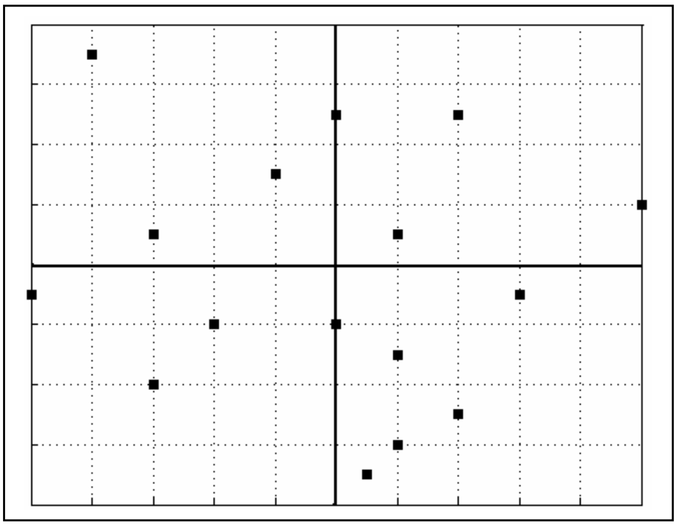


Figure 8. Point set in universal coordinate frame.

To represent points as meaningful, coordinate frames are constructed. In geometric hashing object centered coordinate frames are constructed. Object centered coordinate frames are used to represent the object with respect to its intrinsic axis. The main advantage of the object centered frame over viewer centered frame is its view point independency. Although it is difficult to construct, it serves object based solution and satisfies conditions for transformation invariance.

Frames are used as basis and can be defined from three non collinear points. If we have N points in our dataset, 3 non collinear of them used for frame definition and then the remaining (N-3) points are described respect to this frame. Coordinate frame establishment is repeated for each non-collinear triple combination of the point set.

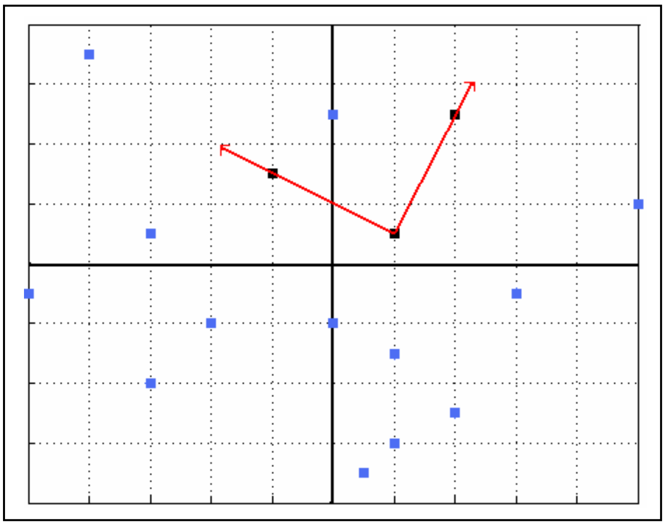


Figure 9. New coordinate frame is defined with three non-collinear points. Remaining points (blue dots) will be represented with respect to the new frame (frame in red).

The representation of points is stored to a hash table. In most simple way the coordinates are used for key parameters to select hash table bins to store. If position of a point in a base is given than it is stored into the hash table so that the 3D position of the point gives the exact address or indice of the hash table bin. When this bin is accessed, the model number and the label of the basis, is recorded.

Assume that point X has coordinates (X_N, Y_N, Z_N) according to the new reference frame and basis is defined by using points numbered (1,4,7) for model A then the resulting hash table entry will be as seen below.

Hash Table	
X_N, Y_N, Z_N	Basis_1_4_7, Model A
...	...
...	...
...	...
...	...
...	...

Figure 10. Hash Table for Initial Geometric Hashing Method

3.2. RECOGNITION

Recognition stage is the second phase of the geometric hashing algorithms. In this stage previously prepaired hash table is searched for finding correspondent of the target model. This stage consists of similar steps as in the preprocessing stage.

In this stage candidate scene features are searched through the hash table. Search for all scene features are required but irrelevant model and feature comparisons are obviated by hashing algorithm.

As in preprocessing stage salient features are extracted by using feature extraction methods. A basis frame is defined by using non-collinear points from feature dataset. When a basis is defined the ramaining points are represented with respect to the new defined reference frame. The new positions of the feature points are used to access hash table bins. The position of the point is used to access hash table bin and then the model and the basis' vote are incremented. For the rest of the points and the basis anternatives this procedure is repeated.

The models received high votes are used for verification process. In the verification process, transformation matrix between the candidate models and the target model is found and then the target points are tried to align with the candidate models. If the model points coincide with target model then the modelis selected as the recognized model.

.

CHAPTER 4

PROPOSED METHOD

4.1. FEATURE EXTRACTION

Feature extraction process is the beginning and the core step of the recognition system. We have used local surface properties for object representation. In CHAPTER 2, we have showed different curvature estimators. Among these estimators we have chosen to use shape index parameter. Shape index parameter was proven that it provides better curvature estimation performance against Gaussian and mean curvature estimators. Since it gives specific threshold values for each surface shape, stable shape assignment can be obtained using shape index.

For description of local surfaces of the object, we have calculated shape index parameter for each point on the scene. Shape index values are then used to classify points according to Table 2 . Shape index parameter provides us to define 7 different local shapes except planar surfaces in which principal curvature values are both zero. Points that belongs to the same class are groupped for further processing. Each point group is represented in a compact and informative form. This kind of representation is provided by using connected component labelling concept. This concept is used for representing the adjacent points having same shape index values as one component. For instance points in which shape index values are falling into dome shape class aregroupped and altogether labelled as a component.

Several components are obtained during implementation process but most informative ones are selected for the representation of each object. In the following section connected component labelling is explained.

4.2. CONNECTED COMPONENT LABELLING

Connected component labelling is usually used to group connected pixels in an image. This task also refers to the connectivity of the pixels. Connectivity is explained simply as the relation between two or more pixels. Two pixels are called connected if locations are adjacent and their pixel values are the same. Most commonly used connectivity measures are 4 and 8 connected components. For the 4-connectedness, pixel is checked for its 4 neighbours (up,down, left and right neighbours). For the 8-connectedness, pixel is checked for 4-connectedness plus four diagonals.

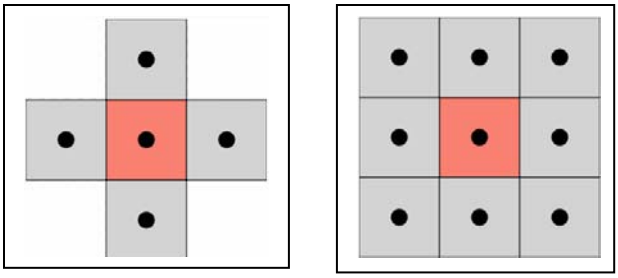


Figure 11. 4-connectedness and 8-connectedness for a given element

Group of pixels which are connected to each other is called connected component. Connected components available in an image is marked with different numbers, this marking is called connected component labelling.

Connected component labelling is done by scanning the given image pixel by pixel. Two pixels p_1 and p_2 are connected if there is a path between them. During scanning, pixel p_1 and its four neighbours are checked. If any of the adjacent pixels has a label from previous labelling steps, p_1 is labelled as its neighbour. But if there is not any pixel labelled yet p_1 is marked as a new label. Through the whole image this process is repeated. At the end of scanning, seperate connected components are labelled [14].

At component labelling 4-connectedness behaves stricter than 8-connectedness. Because 8-connectedness covers 4-connectedness but vice versa is not valid. During feature extraction process 8-connectedness and 4-connectedness of the pixels are checked on images and compared.

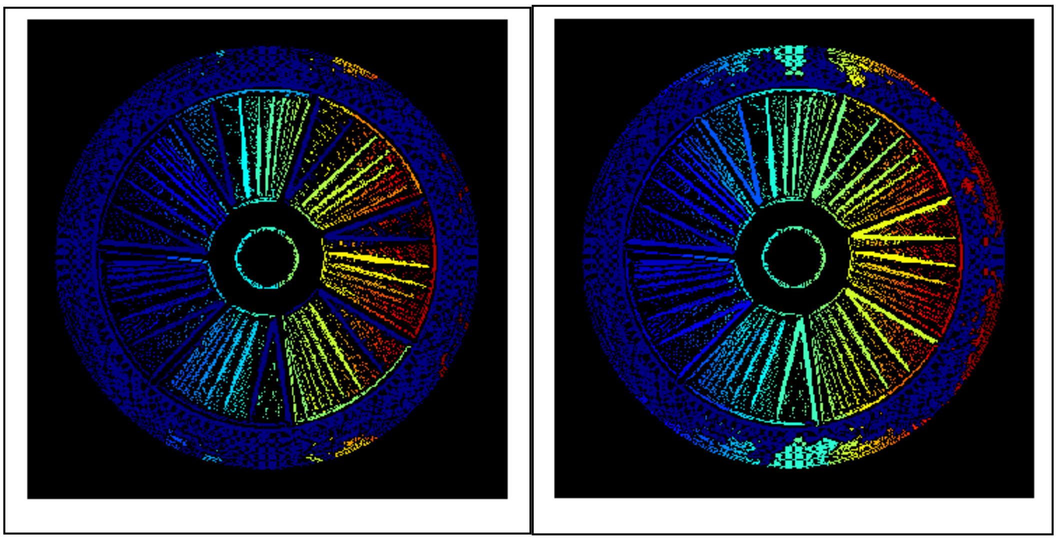


Figure 12. Component labelling with 8-connectedness at left and component labelling with 4-connectedness at right.

Results for 4-connectedness and 8-connectedness component labelling is shown in Figure 12. Components consist of 8-connected pixels have wide areas against small but greater number of components in 4-connected pixels. Components formed by 4-connected neighbour pixels are preferred rather than 8-connected. Because each component area and its center point is meaningful at the rest of the implementation. As the area of component is becoming scatterred the representation of the component with a single center point becomes challenging. Center point can represent the group better if the component pixels are located together. By using 4-connectedness, we have increased the number of groups and decreased the number of pixels located in each group and obtained a better composition.

In Figure 13 shown below for auto model shape index values are computed for each pixel and then pixels falling in to the each surface shape is displayed seperately. In Figure 13 below each color shows different labelled components.

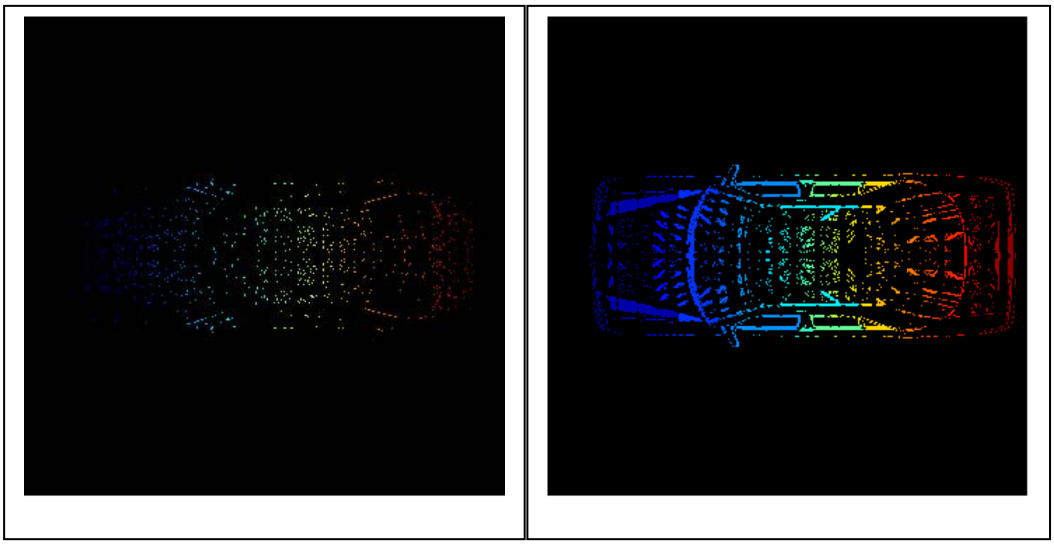


Figure 13. Dome components at left and Ridge components at right for auto model.

In the representation above each dome and ridge component on the image is determined according to its 4-connected pixel neighbours. Each color specifies different component label.(Red group 1, Blue group 2 ,Yellow group 3 and etc.)

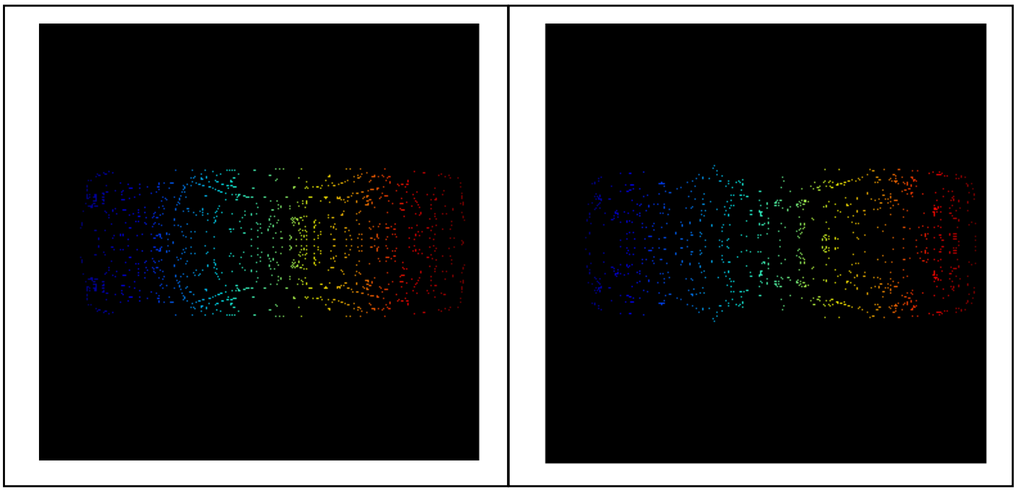


Figure 14. The left figure shows Saddle Ridge points and right one displays Saddle Points.

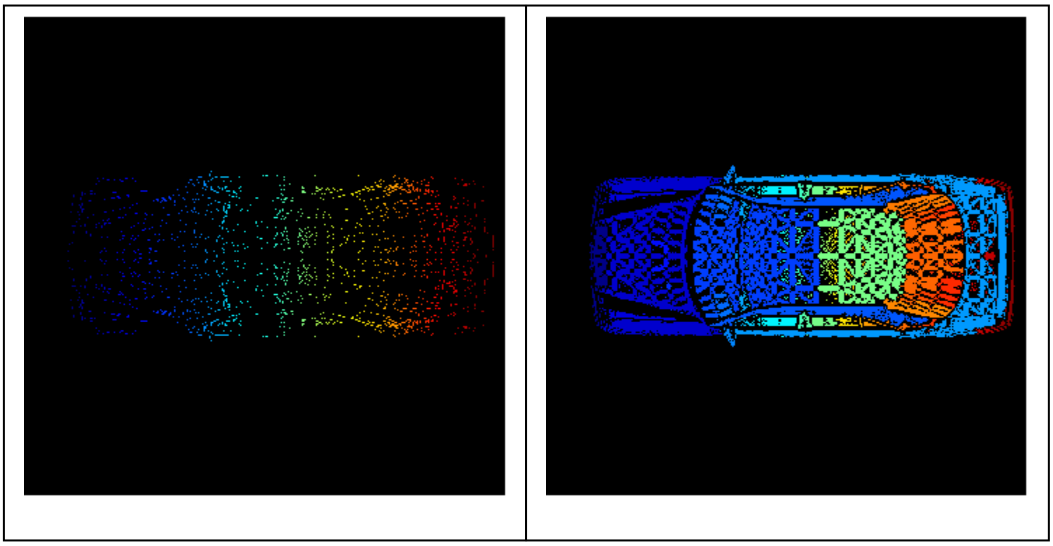


Figure 15. The left figure shows Saddle Rut points and right one displays Rut points on the model

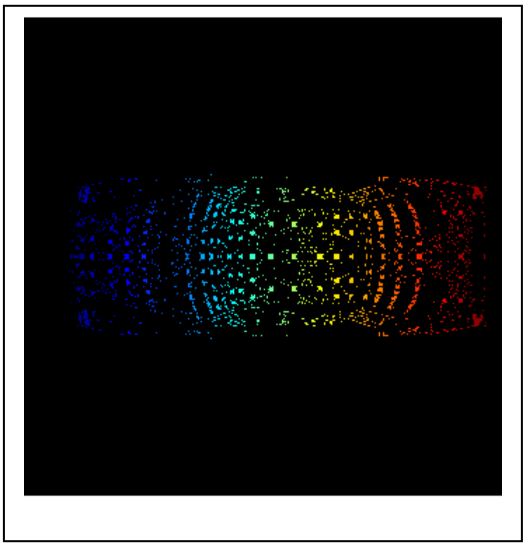


Figure 16. The figure shows Cup points on the model.

4.3. DECOMPOSITION OF FEATURE POINTS

The most primitive way to analyze a scene is processing each point on that scene. But it is neither time nor memory efficient. Therefore there is a necessity to process as much as few number of points to identify a given object. By computing shape index values for all pixels we have assigned points into seperate surface shapes. This assignment of points are done by looking connectivity of pixels, so points which locates together as one piece has become a single component. Each component is represented with its center point and its area. Area is obtained as actual number of pixels in the component region. These two parameters are used to sort and find most informative components in the image. In the analysis it was seen that components with having wider areas can survive even if the object is rotated in the scene. That is why we have used components with wider areas in the implementation process.

To be able to classify objects first of all seven different surface shapes (Table 2 are determined by computing shape index values. For each one of the shapes, components are obtained and they are sorted with respect to their areas. Two components with the widest areas from each shape type are selected to represent that surface in the image. Since there are 7 surface shapes, 14 components are collected to identify the object. Each surface type is numbered seperately. Beginning from dome to cup each surface shape is numbered from 1 to 7 respectively. This numbering is used in geometric hashing algorithm as the key parameter to access the hash table entries.

In Figure 17 for auto model each surface shape (dome, ridge, saddle ridge, saddle rut, rut, saddle point, cup) is represented with rectangles in a different color. Dome in Black, Ridge in Red, Saddle Ridge in Green, Saddle Point in Blue, Saddle Rut in Yellow , Rut in Magenta , Cup in Cyan .

Size of the rectangle is related with the size of the connected component. Each component is represented with smallest bounding rectangle that covers all of its points. So the bigger the rectangle is the greater number of points in that component.

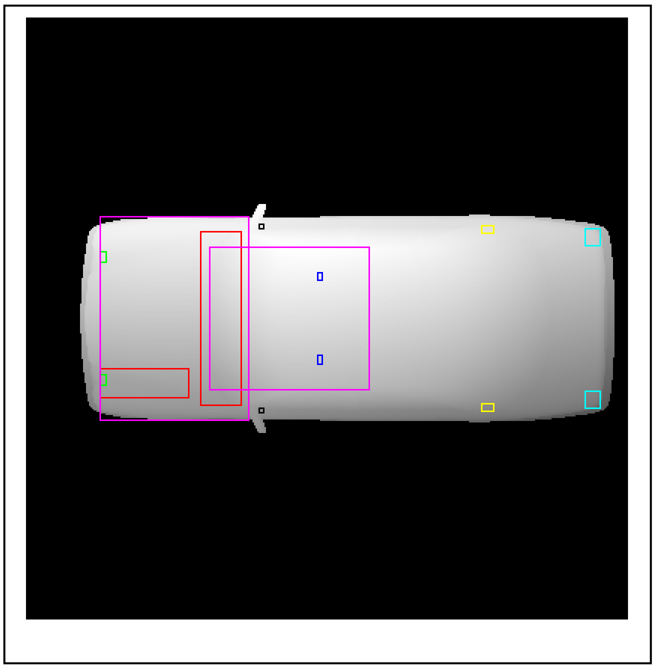


Figure 17. Extracted Salient Regions

The rectangles show the surface shapes but we need to express each of them with a single point. For this purpose we express each rectangle with its center point. So the rectangle region becomes a single point. In Figure 18 center points of rectangles are shown.

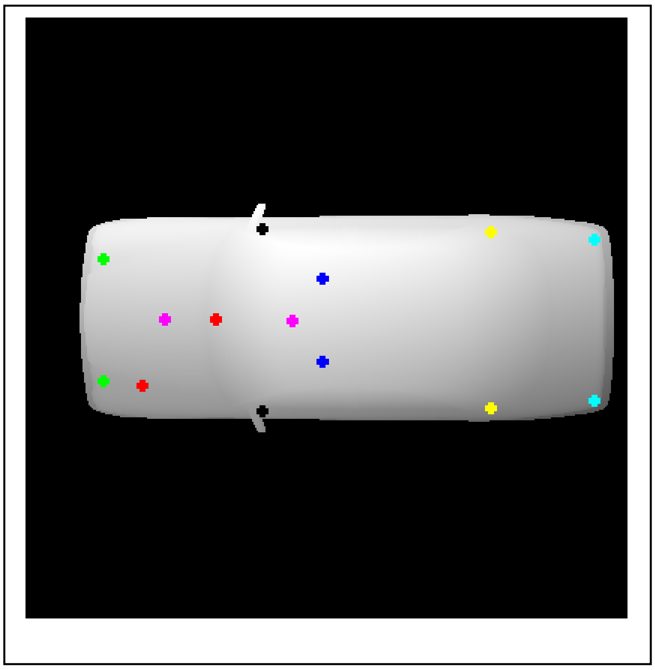


Figure 18. Point Set Representation of Extracted Regions

PSEUDO CODE FOR FEATURE EXTRACTION METHOD

START

Read X, Y, Z coordinates of all pixels from range image files

Calculate principal curvature values for all pixels

Obtain shape index parameter using principal curvature values.

Classify shape index values according to the corresponding shapes.(Dome, Ridge, Rut, Saddle Point, Cup, Saddle Rut, Saddle Ridge) (Table 2)

Loop for each class of shapes

Compute 4-connectedness of points
Apply connected component labelling

Loop end

Determine area of each labelled component

Determine center point of each component

Give numbers to each class according to their shape (1 to 7) (Table 2)

Sort areas of each component group

Take two components having biggest area from each class

Retrieve 3D coordinates of each center point from range image file

Record center points of selected components and their shape numbers (1 to 7)

Send recorded data to geometric hashing algorithm as point set

FINISH

4.3.1. SPLASH APPROACH

Feature primitives are served as point set at the end of the feature extraction stage. For description of these points additional methods are used. As explained in CHAPTER 2, different approaches are available in the literature. Depending on the local information of the objects, a global representation is obtained. In one of the approach, spin image, 3D information is converted to 2D space and stored to an array. In this approach a dense data is obtained and computational cost is considerably high. In the splash approach, surface normal relationship between a reference point and its neighbour points is found and then compared. To obtain rich information, this idea is extended to super splash model. In point signature, by using a method similar to the splash, signed distances are obtained and compared.

In the description part of the thesis we have used surface normal informations and splash approach. We have also worked on super splash approach to obtain more information about local patches.

In the splash idea, on surface at a given point p, surface normal is computed and this normal is called as reference normal of the splash (***n***). By considering point p as a center, a circular slice of radius ρ (typically $10 < \rho < 25$) is computed. By starting from an arbitrary point on the circle with steps of Θ angles (typically $1° < \Theta < 15°$, we have used 9 degree step sizes in the implementation), sample points are determined [1]. At each sample point on the circle, new surface normal is computed and each is called n_θ. From the start point to the end, samples of surface normals are collected.

At the next step each obtained surface normal is related to the reference normal. The angular difference between reference normal ***n*** and the splash normal n_θ is computed. This relation has provided how the structure of the surface patch changes around point p.

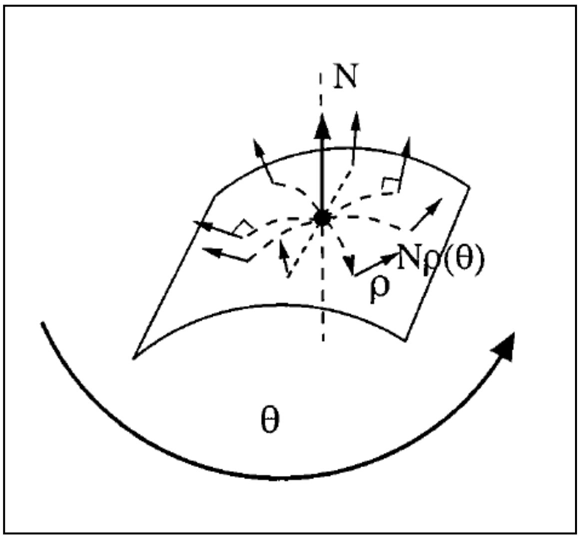


Figure 19. Splash Description [27]

Super splash is an extended form of splash concept and is composed of splashes with different radii. This concept provides further information about the local region around a given point.

In the encoding part of Stein and Medioni, splash values are represented with 3D curves. They have applied line fitting method to the curve and obtained number of lines which are intersecting with each other. The angle relation between these lines is then used for indexing. They have already used this kind of approach in recognition with super segments. Line segments and the relation of them are used as index information for a database.

In our application we have customized this idea and defined a computational efficent version of it. We have obtained surface normal of reference point and the surface normals of samples on the splash region as well. Then angular relation is also computed. At the end of this computation for the splash region we have used angular relation of 40 samples (360° / 9° step size) to convert them to a 1D parameter. For 1D parameter we have calculated the average of angular difference values of surface normals. For each splash profile we have obtained only one value. This value represents the average orientation change of surface patch around a given point. When super splash is used, more than one splash region is applied to the point then each splash profile serves its own average value and the total average serves as the super splash average of the given point. This kind of approach has lowered the dimension of the splash concept of Stein and Medioni [1].

4.3.2. SELECTION OF SPLASH POINTS

It is important to choose specific points for splash utilization. Simply it can be applied to each point on the image but informative and disctintive results could not be obtained in that case. This kind of application also provides disadvantage from the view of computation time. Therefore, areas that can provide rich results should be selected. These areas usually correspond to regions where the curvatures exist. Splashes in flat surfaces give less distinctive information compared to highly curved areas [1]. Therefore we have chosen regions which are extracted by using shape index computation as target locations to apply the splash. As seen in Table 2 seven different types of curvatures are defined. Since each shape region is represented with its center point, these center points are also selected as splash profile center points. Because these points have sufficiently robust and distintive information for a splash.

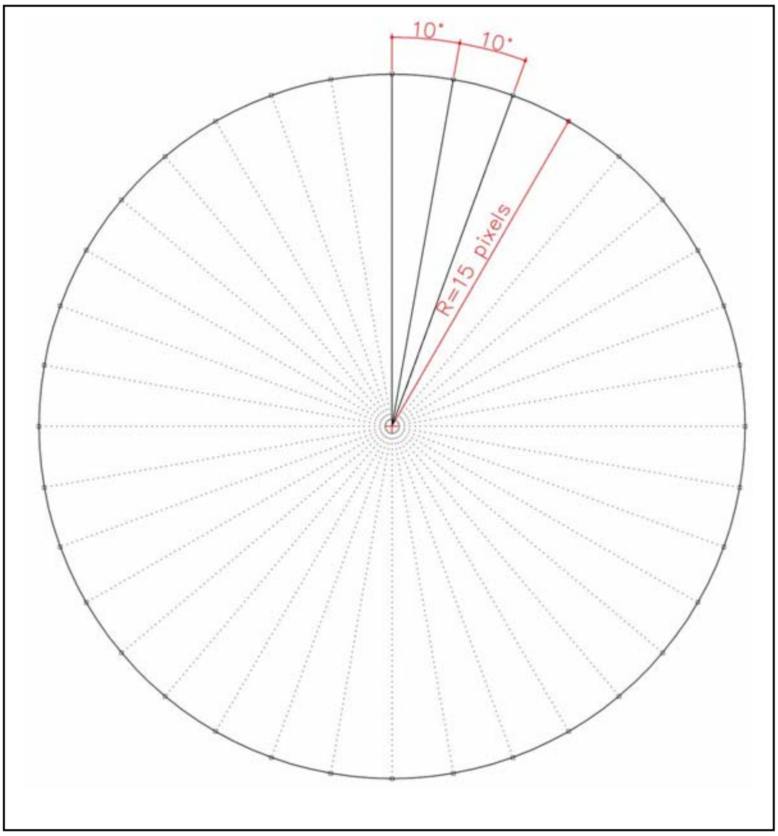


Figure 20. Splash Profile

Figure 20 describes the top view of the splash profile. 15 pixels away from the given feature point, a circular region is established. On this circular region sample points are selected with 10˚ step sizes.

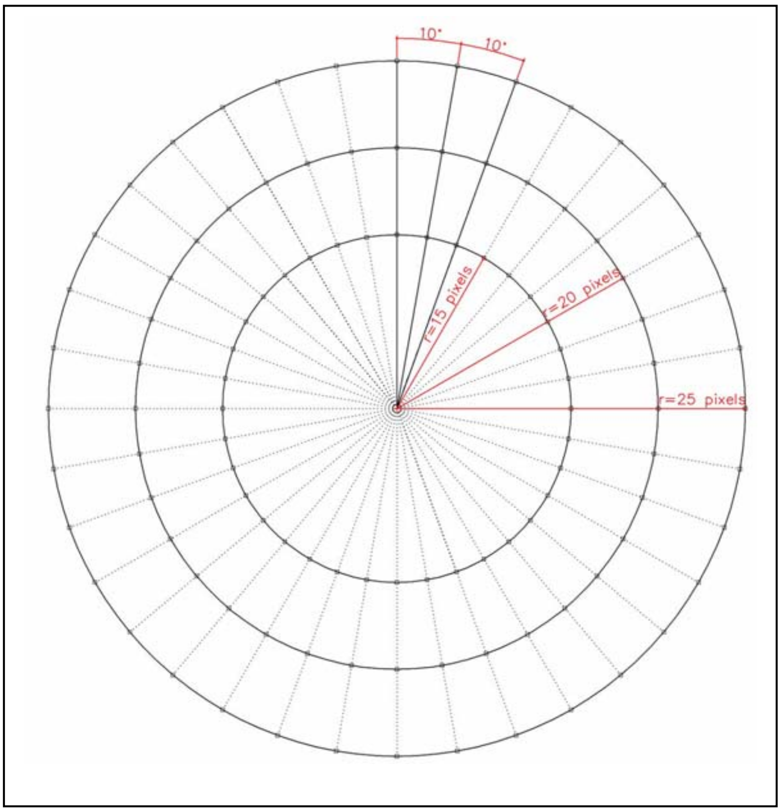


Figure 21.Super Splash Profile

In Figure 21, top view of the super splash profile is displayed. In this figure super splash is composed of three splash regions with different radii. Splash regions with 15, 20 and 25 pixels radii are used to construct super splash. In the same fashion, on each splash region sample points are determined by 10˚ step sizes. Three splash data are then averaged to obtain a single value for the given point.

4.4. GEOMETRIC HASHING

The geometric hashing algorithm is used to obtain a transformation invariant recognition system. By defining object centered reference frames, viewpoint independent object representation is obtained. This ability is supported by low computation cost of the indexing approach of the hashing method. In this section two phases of the geometric hashing algorithm are explained. In the subsection of preprocessing phase, representation of the feature set according to the object centered coordinate frames is explained. Hash table entry of model information is discussed in the following sections also. Recognition stage of the method and the matching approach is given in detail as well.

4.4.1. PREPROCESSING

Preprocessing stage can also be called as the off-line stage. Because computations in this stage do not effect the recognition time. Feature extraction and description of the training model images are performed to construct a database or a hash table before the recognition of a test scene is applied. Since geometric hashing does not deal with feature extraction, the input to the system should be transformation invariant point sets. Point sets can be obtained from geometric characteristics or physical quality of the object model. For instance model scene can contain corners, edges, curvatures or regions that have some distinct characteristics. For geometric hashing input, these feature sets need to be represented as points. A point can define different features in applications as previously mentioned like intersections of edges, corners or center points if the feature is scattered region.

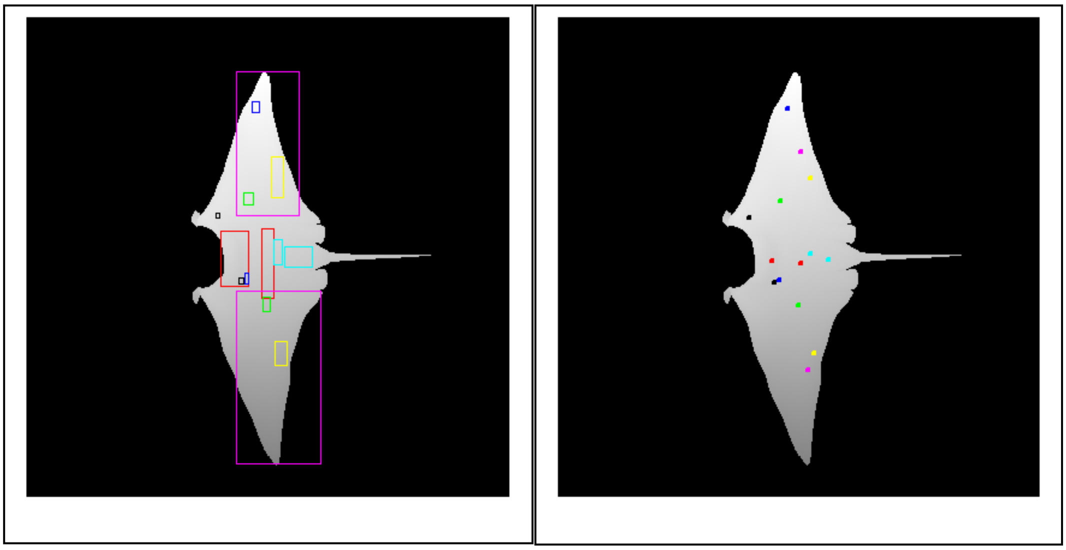


Figure 22. Manta figure and extracted salient regions and bounding rectangles of each component are displayed at left and pointset correpondence is displayed at right. Each rectangle is represented with its center point in the right figure. Each color corresponds to a different surface structure.

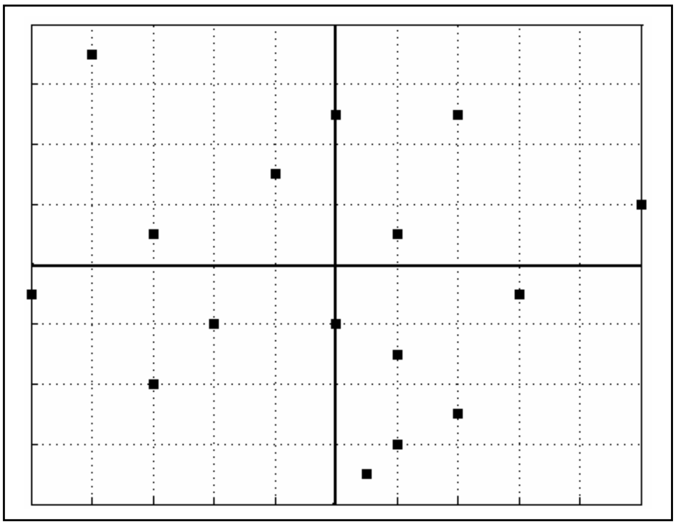


Figure 23. Point set in universal coordinate frame.

To represent points as meaningful, coordinate frames are needed. Frames are used as basis and defined from three non collinear points. If we have N points in our dataset 3 of them used for frame and then the remaining (N-3) points are described on this frame. Coordinate frame establishment is repeated for each non-collinear triple combination of the point set.

In the following sections the construction of coordinate frames and relationships of the points with respect to new reference frames are explained.

4.4.1.1. COORDINATE FRAMES

Coordinate frames are reference systems for locating points in terms of numerical quantities in space. Coordinate system is a mathematical language of defining position and orientation of geometrical objects. A few numerical calculation is sufficient to find relationship of two points if their set of points of coordinates are known.

Cartesian coordinate system is one of the simplest and useful coordinate system. It is simply constructed by using an origin point *O* and three perpendicular vectors, intersecting in the origin which they form *Ox, Oy* and *Oz* axis. In which a point can be described by using three parameters *(x, y, z).* Each parameter is defined in its own surfaces which are one of the *X; Y* and *Z* surfaces.

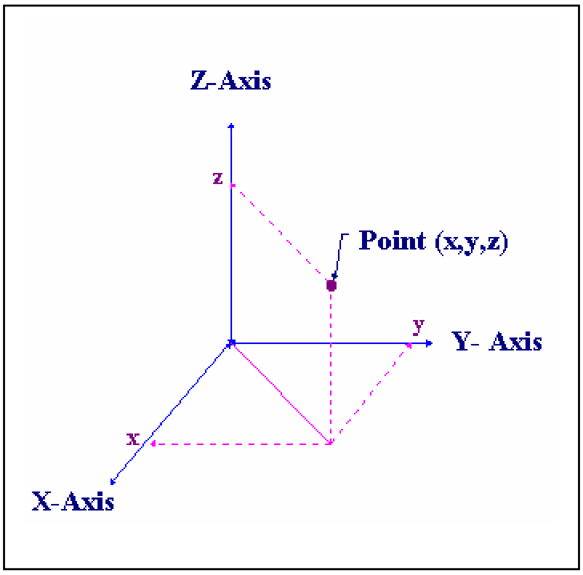


Figure 24. Orthogonal Coordinate System

4.4.1.2. COLLINEARITY

Collinearity is the concept to determine if points lie on the same line. Since two points in the space can form a line and they are always collinear, collinearity is defined for three or more points. Three or more points are assumed to be collinear when they lie on the same straight line. Collinearity is simply checked by computing the determinant of the point positions. If three point are defined as,

Point 1= (x_1, y_1, z_1)

Point 2= (x_2, y_2, z_2)

Point 3= (x_3, y_3, z_3) then

$$\det\left(\begin{pmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{pmatrix}\right) = 0 \qquad \textbf{(3.1)}$$

Equation (3.1) proves that points are collinear.

If three points satisfy the equation then this means that the determinant of the points is zero and the points are collinear [16]. Collinearity of points is used at the construction of the coordinate frames. A coordinate frame can be constructed if at least three non collinear points are available in the space. By using these points a new coordinate frame can be obtained. Then this obtained reference frame can be related to the universal coordinate frame.

4.4.1.3. FRAME RELATIONS AND TRANSFORMATION MATRIX

Coordinate system construction lets points to be defined in space with their 3D positions. Besides universal coordinate systems, positions can be defined at different references also. The reference frame in which a point is defined can be understood from the notation. Assume that point P is defined at frame A, this can be interpreted as ^{A}P . This interpretation shows that *(x, y, z)* values of point P is determined according to frame A.

Most of the time, there is a necessity to define new frames to express points with respect to. In this kind of applications relative positions and orientations of points are determined considering two frames. If new frame B has a different orientation according to the initial frame A, then the relationship between them can be determined by the rotation matrix. In notation, the matrix ${}^{A}_{B}R$ describes the rotation of frame B relative to frame A. This 3x3 rotation matrix defines the relative orientation between frame A and frame B. Rotation matrix is a square matrix whose transpose is equal to its inverse.

$$ {}^{A}_{B}R^{T} = {}^{A}_{B}R^{-1} \qquad (3.2) $$

Also rotation matrix of frame B respect to frame A is equal to the inverse of the rotation matrix of frame A respect to frame B.

$$ {}^{A}_{B}R = {}^{B}_{A}R^{-1} \qquad (3.3) $$

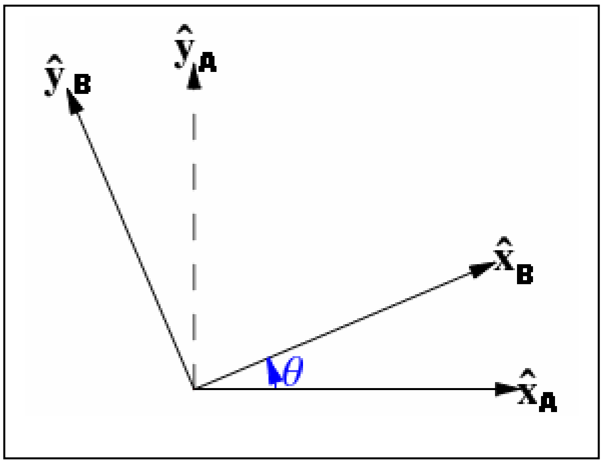


Figure 25.Frame Rotation in Z Axis [17]

In Figure 25 if we assume that Z axis points towards us (out of the page) we can say that frame B is rotated about Z axis by θ degrees in counterclockwise relative to frame A.

In 3D space when a frame is rotated with respect to the principal *X, Y, Z* axis rotation matrices are defined as follows.

$$R_x(\alpha) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & \sin\alpha \\ 0 & -\sin\alpha & \cos\alpha \end{pmatrix} \tag{3.4}$$

$$R_y(\beta) = \begin{pmatrix} \cos\beta & 0 & -\sin\beta \\ 0 & 1 & 0 \\ \sin\beta & 0 & \cos\beta \end{pmatrix} \tag{3.5}$$

$$R_z(\gamma) = \begin{pmatrix} \cos\gamma & \sin\gamma & 0 \\ -\sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{3.6}$$

Here the coordinate frame is rotated with respect to x axis about α degrees, with respect to y axis about β degrees and with respect to z axis about γ degrees in counterclockwise direction.

Rotation matrices are sufficient to express the relation between two frames when the origins of the frames are overlapped and just the orientation is changed. If the origins of the two frames do not coincide, this means that there is a translation between the frames. Then this position difference is described by a translation term.

Assuming that a point is defined at frame B and another frame A exists at the same space which has the same orientation as B, but differs from B by only a displacement. Then the point is described with respect to A as an addition of vectors [15] (Figure 26). In this expression the point position is not changed, only its description is updated according to the new frame's notation. The new description is obtained by using the relation between two frames.

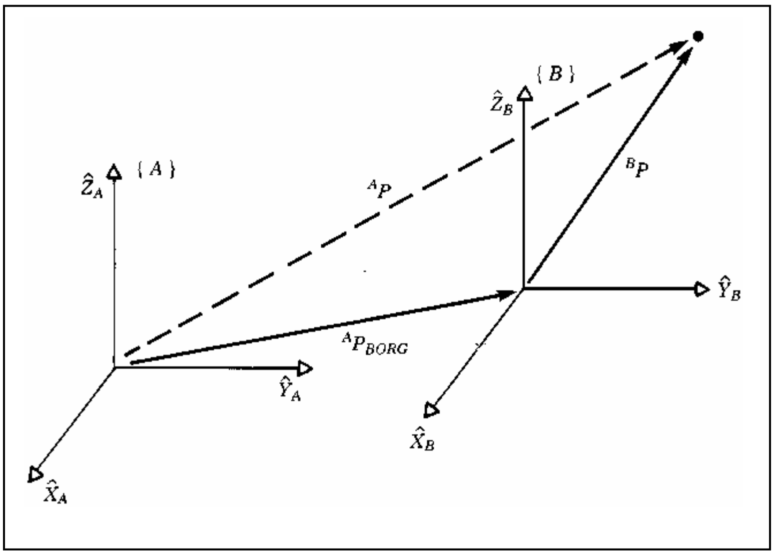


Figure 26. Translation of frames A and B [15]

$$^{A}P = {}^{B}P + T \qquad (3.7)$$

Where $\boldsymbol{T}$ is the amount of displacement between A and B and is also called as the translation term.

Point representations in a known frame are usually changed according to new coordinate frames. When origins of the frames coincide, rotation matrix can be sufficient for determination of the orientation difference between frames. If only origins translated without any orientation difference, translation vector gives how much displacement is applied to the frame. When both of the situations exist, i.e. if the frame origins do not coincide and the orientation of the frames are not same, then we need to apply translation and rotation consecutively.

First of all, it is assumed that only orientation is changed. To compansate this difference, rotation matrix between two fames is premultiplied with the point *P*.

After rotation is compansated between two frames, then translation parameter is added to the result to obtain general transformation between two frames for point *P*.

$$ {}^{A}P = {}^{A}_{B}R \; {}^{B}P + T \tag{3. 8} $$

It is shown in the conceptual form

$$ {}^{A}P = {}^{A}_{B}T \; {}^{B}P \tag{3. 9} $$

So the rotation and translation between two frames can be expressed with one matrix which is called transformation matrix. Transformation matrix is a 4x4 matrix which consists of 3x3 rotation and 3x1 translation components.

$$ \begin{pmatrix} {}^{A}P \\ \cdots \\ 1 \end{pmatrix} = \begin{pmatrix} & {}^{A}_{B}R & & \vdots & T \\ \cdots & \cdots & \cdots & \vdots & \cdots \\ 0 & 0 & 0 & \vdots & 1 \end{pmatrix} \begin{pmatrix} {}^{B}P \\ \cdots \\ 1 \end{pmatrix} \tag{3. 10} $$

The matrix is called homogeneous transformation matrix. Homogeneous coordinates represent the points in the projective plane. If there is a point *(X, Y, Z)* in the Euclidean plane, the homogeneous correpondence of this point is *(X, Y, Z, 1)*. The number 1 as the fourth parameter is used for scaling. Therefore for general description, a point in homogeneous coordinates is defined as $(X\alpha, Y\alpha, Z\alpha, \alpha)$ for any non zero α.

$$(X, Y, Z) \rightarrow (X\alpha, Y\alpha, Z\alpha, \alpha)$$

$$(X, Y, Z, 1) \equiv (X\alpha, Y\alpha, Z\alpha, \alpha)$$

Homogeneous representation of the given points is described above.

4.4.1.4. REPRESENTATION OF FEATURE POINTS

Each member of feature dataset is represented in new object centered reference farmes. To be able to describe them as much as distinctive we have extracted attributes of each reference frame and each feature point. Splash approach, relationships of basis points and the properties of the remaining feature points, are used together for robust description.

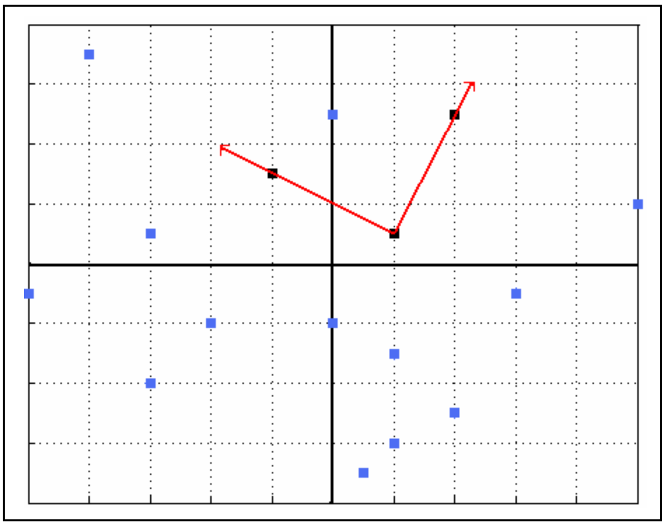


Figure 27. New coordinate frame is defined with three non-collinear points. Remaining points (blue dots) will be represented with respect to the new frame (frame in red).

After the definition of the new coordinate frame from non-collinear point set, there is a need to represent the remaining feature points in the new frame. To do that, we have processed feature informations step by step. Firstly points' pixel values are used to obtain *(X, Y, Z)* coordinates from the range image files. In range images, *(X, Y, Z)* values are represented with respect to the universal coordinate frame. After selecting one of the triple non-collinear set, there is a need to find relationship between universal coordinate frame and the new frame. Considering position of the frame points, rotation matrix is obtained. Rotation matrix relates two frames according to their orientations. Also translation term is taken into account to find the displacement between the origins of the frames. From both of these terms 4x4 transformation matrix is obtained.

$$\text{Transformation Matrix} = \begin{pmatrix} {}^{A}_{B}R & \vdots & T \\ \dots\dots\dots & \vdots & \dots \\ 0 \;\; 0 \;\; 0 & \vdots & 1 \end{pmatrix}$$

Transformation matrix of the given basis is used to find the coordinates of the remaining points of that frame. The coordinates of the points with respect to the new frame is meaningful for entries of the hash table.

If we assume that point A is defined at the universal frame as (X_a, Y_a, Z_a) and the same point is defined in the new frame as (X_a, Y_a, Z_a) the relation of the same point with respect to different frames is obtained by applying transformation to the point.

$$ {}^{A}P = {}^{A}_{B}T \;\; {}^{B}P \qquad \textbf{(3. 11)}$$

By applying the transformation matrix to all points for the given basis, the new representations of the points are obtained.

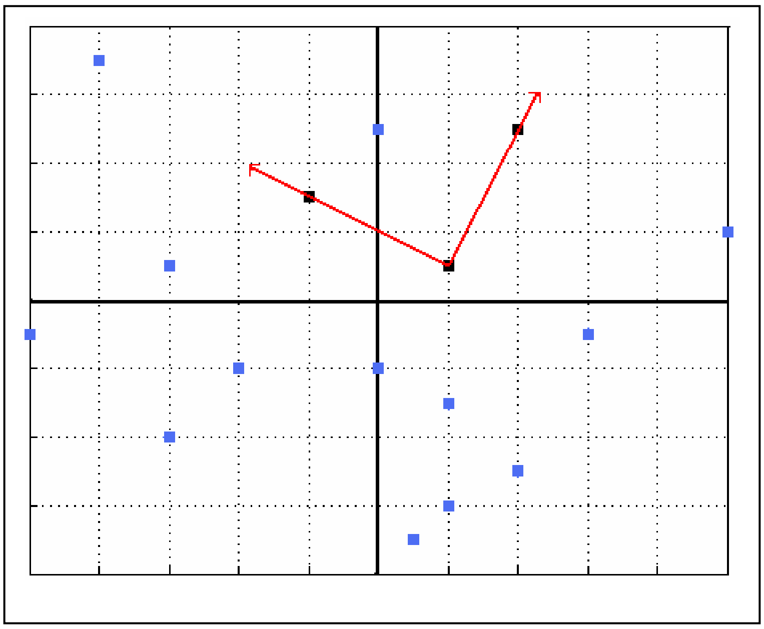


Figure 28. Representation of coordinates respect to new coordinate frame

New coordinate representations are used to find Euclidean distance between origin (X_0, Y_0, Z_0) of the new frame and the feature point in new frame (X_N, Y_N, Z_N) The Euclidean distance is calculated as in equation below.

$$\text{Euclidean Distance} = \sqrt{(X_N - X_0)^2 + (Y_N - Y_0)^2 + (Z_N - Z_0)^2} \qquad \textbf{(3. 12)}$$

Besides Euclidean distance computation, we have used other attributes of reference frame points. For discrimination of reference basis and feature points we have computed splash profiles for each of the frame points. Also angular relationships between frame points are used to represent a stable solution.

4.4.1.5. SPLASH INFORMATION FOR BASIS

In the following sections we will introduce few specific methods to define each basis. This will let us to find right basis and points easily during recognition stage. Euclidean distance, computed previously is a stable parameter to represent a point in the frame. Each point is represented with its distance parameter in its own frame. During preprocessing stage, several models are analyzed and it's likely to find same distance parameters at different models. Therefore decription of points with only distance parameter is not a good approach for robust algorithm. We need to find more discriminative features to describe points and basis. To be able to find matching feature points in the model, first step is finding correspondent reference frames. Therefore we applied splash approach to each point of the reference frames. Splash retrieves surface topology information around the given frame point. This surface information is invariant to geometric trasformation and gives robust results.

In the application, we have determined frame points as splash centers. We have applied the following steps for each frame point seperately.

We have formed three different circular regions 10, 15 and 20 pixels away from the basis point. On each circular slice we have determined arbitrarily 40 points with 9 degrees step sizes. Then surface normal vectors are computed for these selected sample points. Surface normals give us the surface topology at the close neighbourhood of the selected points.

Surface normal of center point is also computed and selected as reference surface normal. Angle difference between surface normals obtained from the samples and the reference normal is computed. For each splash region we have obtained 40 angle difference values. The average of angle diffrences for each splash region determines the approximate surface direction difference with respect to the center point. At the end, we obtain 3 different average splash angles from three splash regions with different radii. The average of these three splash angles determines the super splash angle for the given basis point. This information provides us to recognize the basis points with respect to their surface topologies.

For each point of the reference frame, the procedure explained above is repeated. Consequently splash attribute is attached to each member of the coordinate frame.

In Figure 29, super splash regions attached to the basis points are showed. The splash information obtained from the basis points are used at model library construction and during recognition process of target objects.

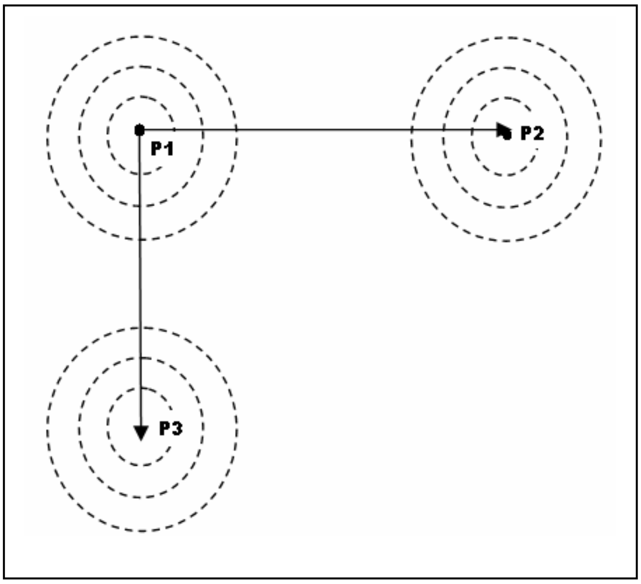


Figure 29. Splash regions around points of coordinate frames.

4.4.1.6. ANGULAR INFORMATION FOR BASIS

As in the splash approach, another parameter invariant to geometric transformation is angular relation of the basis points. If we define a triangle by using basis points we can use its internal angles as salient parameters. The angle relation is not effected even if the triangle is rotated or translated. This parameter is also used for storing the data points into the hash table.

Until now, euclidean distances of points to the frame origin, splash angles of frame points and internal angles of triangle defined by frame points are obtained. Each of these parameters is informative and is used for hash table entries.

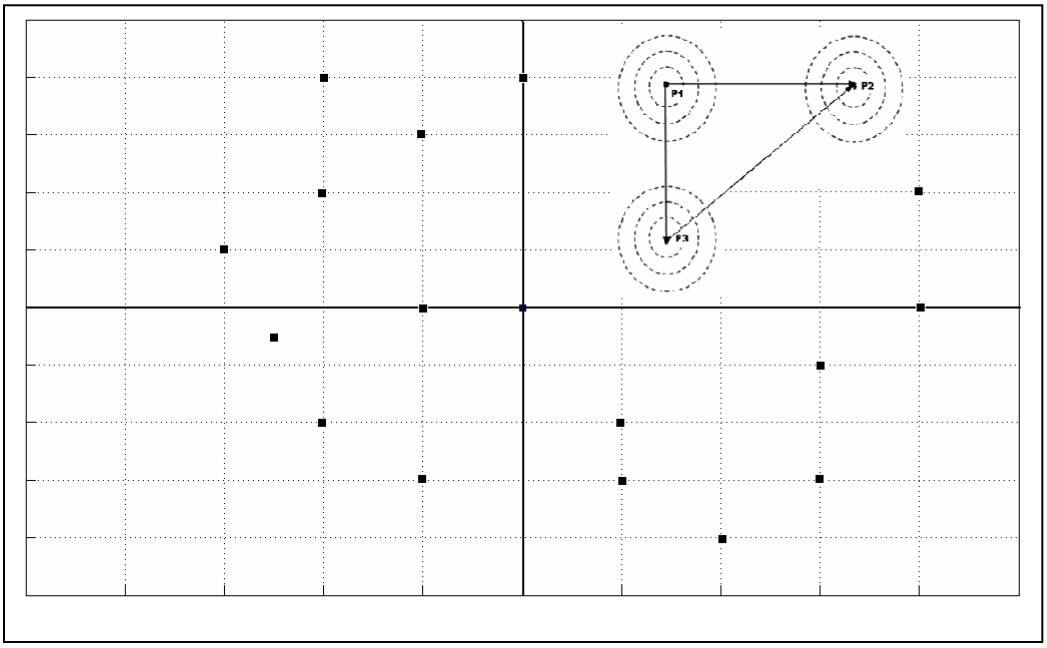


Figure 30. One possible coordinate frame with splash regions on the universal coordinate system.

4.4.1.7. HASH TABLE ENTRY

Inputs to the hash table are determined by concatenating different parameters. Distance, splash angles and internal angles are obtained to be able to discriminate one model from another. For hash table entries we need two items, an address for table bin and data to store in. The first item is very important and needs to be very accurate for the recognition stage. Because geometric hashing algorithm provides to reach only directed addresses of the table. It does not search all table for the recognition. Therefore we determine an address pointer or a key which is invariant to geometric transformations and very discriminative. We have used feature extraction information for determination of the key.

In the feature extraction process, we compute shape index values of all pixels then we group them according to their surface shapes (dome, ridge, rut etc.). During the selection of point sets we have used components with bigger areas. From each surface type, two components are selected to represent each shape class which are numbered from 1 to 7 where 1 corresponds to dome and 7 corresponds to cup. The remaining shapes are numbered according to the order of the shape classification table (Table 2). These numbers are also used for geometric hashing process. During coordinate frame construction, each point's shape classification is also known from its number. For example, for the reference frame which comprises points from dome, rut and saddle points, (1,6,4) is the code or the description of the basis.

Until here, we have obtained two types of data: first type represents the basis and the second type represents remaining points in that basis. First type includes shape classification number of points in the basis, splash angles of points and internal angle of basis triangle. Second type includes Euclidean distance of points in their own basis. Each hash table entriy will cover all of these information in its bin.

In our application, hash table is formed as three dimensional (7x7x7, since type numbers are at most 7) link list and key parameter of table is selected as points' shape types. The remaining parameters (splash angle, internal angle, distance and model) are located in the bins.

HASH TABLE	
Address	**Data**
Type 1, Type 2, Type 3	Euclidean Distances of points in Basis, Internal Angle 1, Internal Angle 2, Internal Angle 3, Splash Angle 1, Splash Angle 2, Splash Angle 3, Model
...	...
...	...
...	...
...	...
...	...
...	...
...	...
...	...
...	...
...	...
...	...
...	...
...	...

Figure 31. Hash Table Structure and Data Distribution

At the initial form of the algorithm, several trials are experienced to find a robust key parameter for hash table access. At one of the trials, we have used internal angles as key parameters to access hash table. Three angle values in an order is used for accessing. But when the object has been transformed in 3D space, feature point locations, their angle information and geometric relation between points has been changed. Even we have given broad threshold values in recognition stage, the recognition process computation time drastically insreased and recognition rate was low. It is crucial to select an access parameter, because if wrong bin is reached in hash table, all the information necessary for comparison is changed and the result is directly affected. Therefore the most important parameter of the process is assumed to be the address or the key parameter. Especially to be able obtain relevant information from the hash table and make recognition process robust, this parameter is determined carefully.

This kind of results led us to change the key parameter to a more stable and invariant variable against rigid transformations. Selection of type of points as the key parameter solves the problem of geometric transformation. Since its value does not change depending on the replacement of point locations or rotation of the basis points, vulnerable part of the recognition process is stabilized as much as possible.

4.4.2. RECOGNITION

Recognition stage of the geometric hashing algorithm is the final stage to match a given scene with a model in the database. This part can be also called as the on-line phase. The aim in this stage is finding correspondent of the target image in the model library. For matching of target scene, same process is followed as is explained in the preprocessing part. Firstly features of the scene are extracted. The extracted point set is supplied to the geometric hashing part of the algorithm. This point set contains shape types and positions of points with respect to the universal frame. Types are numbers between 1-7 and correspond to different surface shapes as in Table 2 . This type number of each point is meaningful if the basis is defined. Since type values of each base point is used to access to the hash table in the recognition phase. In other words, they are keys to access the hash table.

Recognition process is similar to the preprocessing stage. Point set extracted from scene is used to define the coordinate frames or the basis. All triple combinations are checked for collinearity condition. When triple points are non-collinear then the triple is determined as a coordinate base. Remaining points in the point set other than the basis points are represented with respect to the new formed base. The representation contains Euclidean distance of each point according to the base origin.

As in preprocessing, for robust recognition and specific representation, we need to dicriminate one basis from another. In the same fashion, basis points are connected to each other with lines to define a triangle. Internal angles of this triangle are the other parameters for the description of the target scene. Until here, we have obtained three parameters which are types of basis points, Euclidean distances of points in the new basis and internal angles of basis triangle. The only remaining parameter is the splash angle of the each basis point. To obtain this information, splash regions are defined around the basis points. Radius ρ (10, 15 and 20 pixels) away from the basis point a splash region is defined and the average surface normal difference is recorded for that splash region. Repeating this for two other splash regions we have 3 splash values for each basis point.

The above process is repeated for each basis combination of the given point set. At the end of the process we have sufficient information for the representation of the target model. Target scene information is held in the memory in the same fashion as in hash table until a match is found (Figure 31). After the determination of the information that is needed to find the match of the target model, it is used in the previously prepaired hash table. As mentioned above the key parameter consists of the type numbers of basis points. This number such as (1, 6, 2) identifies the basis and can be thought as a label of the basis. With this label, the bin of hash table is reached with which contains all of the information extracted from this basis (splash angles, internal angles and distances).

When the hash table is accessed the information at the accessed bin is retrieved. For proper voting each parameter group is controlled with each other. Internal angle, splash and distance information of the bin is extracted seperately. Since each parameter is recorded to the hash table in an order, extraction of the information from the bin is made in the same order.

For verification, following steps are applied. One of the basis is selected from the target scene arbitrarily. The whole basis information is obtained from the memory as shape types of basis points, splash angles of basis, internal angles of basis and the Euclidean distances of remaining points with respect to this basis. The shape type of the basis is used to access the hash table. If basis shape types are (1, 4, 3) then Hash table bin correponding to (1, 4, 3) is retrieved. The search location for the target basis is determined according to type of the basis points.

At the second step, the internal angle of the target basis is compared with the internal angle part of the hash table bin. Sum of absolute difference of internal angle parameters are computed and if the result is less than a predetermined threshold value then the splash angle parameters are compared. Again sum of absolute difference of splash angle values are calculated. If the result is less than the splash threshold value, the target basis is assumed to be found in the hash table bin.

The later step is comparing the distance values of the other points. For each matching distance value vote for the hash table model is incremented. This process is repeated for each point in this basis.

The overall procedure is repeated for all basis alternatives of the target scene. At the end of voting operation the model with the highest vote is accepted as the recognized model.

PSEUDO CODE FOR GEOMETRIC HASHING

Preprocessing Stage

Extract fatures of the model scene

START

Pick three non-collinear points from the extracted point set to define a base.

Express euclidean distances of the remaining points with respect to the new frame.

Parameter 1 = Euclidean Distances of points

Define super splash regions for each basis point.

Compute average splash angles from the splash region values.

Parameter 2 = Average Splash angles of base points

Define a triangle using the base points.

Obtain internal angles for the base triangle.

Parameter 3 = Internal Angles of base triangle

Parameter 4 = Use type of base points obtained from extraction process.

Access hash table bin by using parameter 4, store Splash angles, Internal Angles, Euclidean Distances of points and Model of analyzed scene as

“Distance, Internal Angle1, Internal Angle2, Internal Angle3, Splash Angle1, Splash Angle2, Splash Angle3, Model”

Repeat the process until all bases are computed.

END

Recognition Stage

Extract fatures of the target scene

START

Pick three non-collinear points from the extracted point set to define a base.

Express euclidean distances of the remaining points with respect to the new frame.

Parameter 1 = Euclidean Distances of points

Define super splash regions for each basis point.

Compute average splash angles from the splash region values.

Parameter 2 = Average Splash angles of base points

Define a triangle using the base points.

Obtain internal angles for the base triangle.

Parameter 3 = Internal Angles of base triangle

Parameter 4 = Use type of base points obtained from extraction process.

Access hash table bin by using parameter 4, extract Splash angles, Internal Angles, Euclidean Distances of points and Model from the hash table bin.

Compute differences of all possible splash values in the hash table bin with Splash Values of target scene using Sum of Absolute Differences (SAD).

Take SAD result and compare it with the splash threshold value.

Compute differences of all possible internal angle values in the hash table bin with internal angle values of the target scene using Sum of Absolute Differences.

Take SAD result and compare with the internal angle threshold value.

Compute diffrences of all possible Euclidean distance values of points in hash table bin with Euclidean distance values of the target scene using Sum of Absolute Differences.

Take SAD result and compare it with the distance threshold value.

Vote for points which satisfy splash, internal angle thresholds and point distance threshold.

Repeat the process until all bases are checked.

Find the model which has the highest number of votes and recognize the target scene as the model that receives highest vote.

END

CHAPTER 5

EXPERIMENTS AND RESULTS

The main corcern of this research is able to recognize 3D objects. Considering geometric transformation of objects we have aimed to obtain a robust and transformation invariant algorithm. As is explained in the previous chapters, surface curvatures are used to extract features, splash approach is used to describe feautures and geometric hashing is used to obtain an efficient and fast recognition system based on indexing. These algorithms are effectively used and we have obtained recognition rates for each model for the given dataset.

This chapter is devoted to the experiments and the works on recognition of datasets against geometric transformations. The test database files are range image files which can provide 3D information about the scene. 42 different free form object models (Figure 43) are used to test the algorithm robustness on different types of shapes. The dataset models are explained in the following sections. The effect of various threshold values in the recognition stage is also tested.

In the experiments Matlab is utilized as the software development environment. All previously mentioned algorithms are implemented using version R2007a.

5.1. EXPERIMENT DATASETS

In object recognition experiments, we have used range model images obtained from several different view points of the objects [22]. Each model has different type of structure and surface. The model images are seperated two groups which are training set that is used for hash table construction and test set to check the recognition process. The training set contains 2772 images, 66 images from 42 models. There is 23°- 26° viewpoint difference between each image. The test set database with 10836 images, 258 from each model is used to test the system. 258 test images also contain 66 images from the training set. The remaining 192 images have viewpoint change with 11.5° - 13° from each other [12]. Model scenes can be seen in Figure 43 .

5.2. RANGE IMAGES

Intensity or colored images are insufficient for producing three dimensional information. Instead range image format is used to find 3D information of points in a scene. In each pixel of the range image, distance between a predetermined reference frame and the given image point on scene is represented. By using this data 3D structure of the scene is extracted. Range images can also be called as depth images or depth maps. Range images are represented in two forms. In the first case 3D points are given in its own coordinate frame and the points are not ordered. In the second one, each point is produced in a matrix form. In image *(x,y)* pixel pairs corresponds to the depth value for that pixel [21].

```
rif-Header
This Header includes comments and describes the format of this
file, the Header ends with special char "|\n"(The binary
version of this format: xyz digitnumbers as double(network
order (use ntohl())
width, height and maskarray as int (0 or 1 for maskbit)
 All as you would expect them)
1: precision
2: format
3: width
4: height
5: x-value array [width*height]
6: y-value array [width*height]
7: z-value array [width*height]
8: mask array [width*height]
|
float
ascii
400
400
```

Figure 32. Range image file format

In our application the files including range data is in .rif file format. This file includes all necessary information of the 3D coordinates. The file has a header and explains all of the information the file includes. The beginning part of the file is shown in Figure 32.

In the file, precision of the points and structural properties of the file are given. Just after these lines 3D information of the scene is given consecutively. First 400x400 data is for X coordinates, following for Y and Z coordinates.

5.3. MATCHING PROCESS FOR TEST MODELS

Recognition stage of the geometric hashing algorithm is explained in the previous chapters. Voting process has included checking parameters of target scene with information in hash table bins based on predetermined tolerance ranges. This section will explain how this threshold or tolerance values are determined and how behavior of the recognition process is changed according to different threshold values. As mentioned in the geometric hashing section, specific parameters which are type of basis points, splash angles of basis points, internal angles of basis and distances of remaining points according to the basis are available for scene representation. When we have accessed to the hash table with type of basis points, there are three other parameters to check. Therefore, during the verification, we have used three threshold values which are splash angle threshold, internal angle threshold and distance threshold. Instead of looking splash and internal angles one by one we have evaluated them as one parameter like splash parameter and internal angle parameter. So threshold values are checked by sum of absolute differences (SAD) of parameters. For instance, a basis is determined for checking three parameters for splash and three parameters for internal angle will be available for this basis.

The parameters coming from hash table which are also three parameters for splash and internal angles. Differences of groups are found by simple subtraction and their absolute values are obtained. Then these absolute values are summed to obtain SAD value.

SAD value for splash angles and internal angles are checked to see if it is below or equal to a determined threshold value. If it is less than the threshold, this means that the basis we are looking for is found in the hash table.

5.4. RESULTS FOR TEST MODELS

In this section, recognition rates for few models are shown and compared. By applying different threshold values, the effect of the threshold on the recognition rates is observed. Five models from the model set are used for this test. Overall recognition rate for all model set is also given. Results are displayed in Hinton diagrams which provide qualitative display of confusion matrix. Each recognition rate is displayed with a box and color of the box in gray level is associated with the recognition rate. For instance, black shows the maximum recognition rate and white is for minimum recognition rate. Each recognition rate is displayed in a way to obtain cross comparisons.

From this point on, we will call all threshold values with their names. Threshold for splash angles is called ***thr_splash,*** threshold for internal angles is called ***thr_int_ang*** and threshold of distances is called ***thr_dist.***

Thresholds for SAD results are obtained by using fixed values for internal angle threshold and distance threshold. Then by varying splash angle threshold we have observed the change in recognition rate. For three values, the recognition rates are shown in Figure 33.

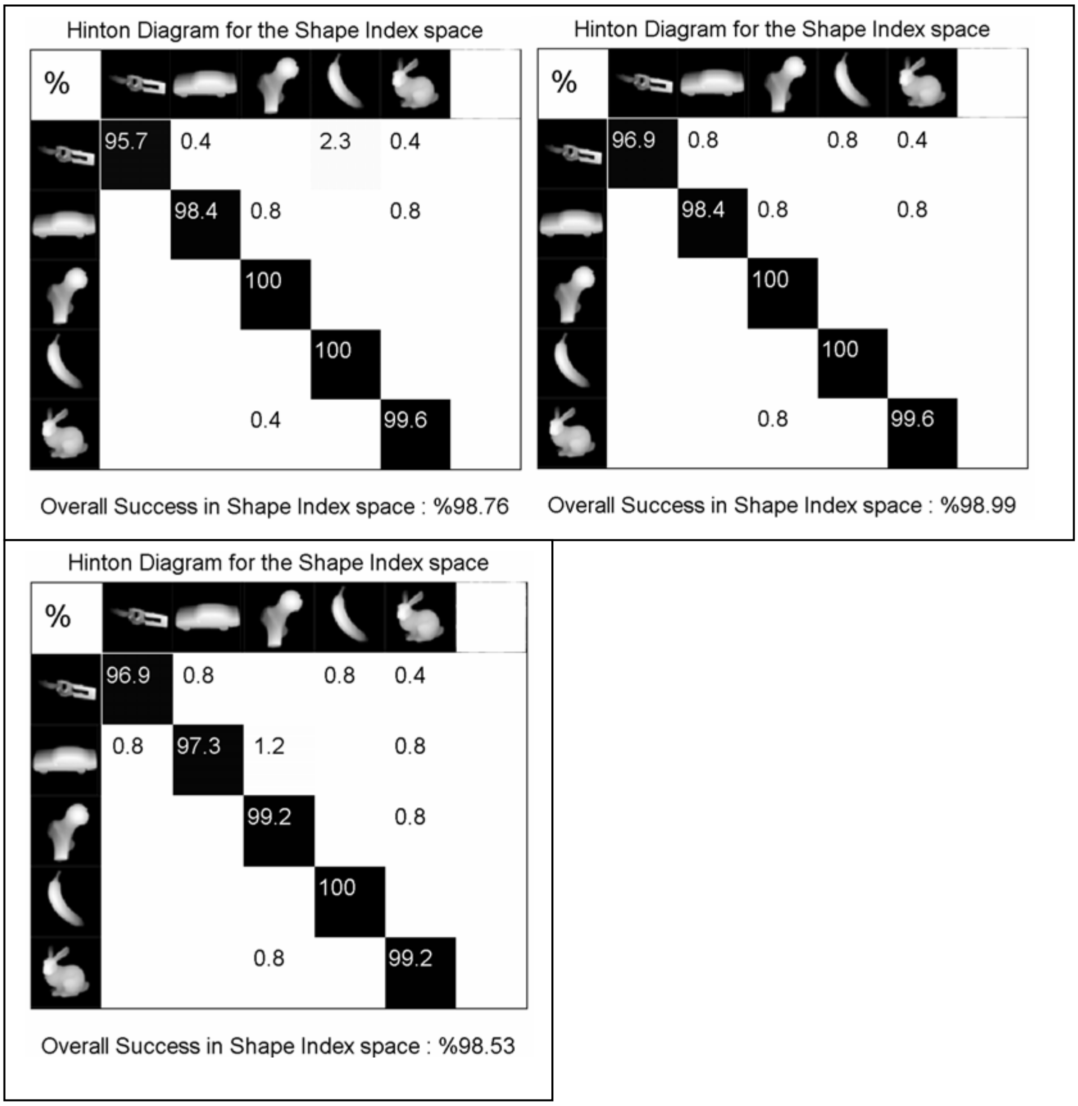


Figure 33. Results for various splash angle thresholds when internal angle and distance threshold are fixed : *thr_int_ang = 25, thr_splash = 20, 25, 30, thr_dist = 20*

The results show the effect of changing recognition rate against various splash angle threshold values while internal angle and distance threshold values are fixed to 25 and 20 respectively. The top left hinton diagram shows for splash threshold of 20 units, top right one is for threshold value of 25 and the second row hinton shows the result for splash angle threshold of 30 degrees. The five models in these hinton diagrams are selected from the top five models of alphabetical order of the model names. There is no specific selection method applied for the given models. The threshold values are used during SAD calculation between target scene and hash table bin content.

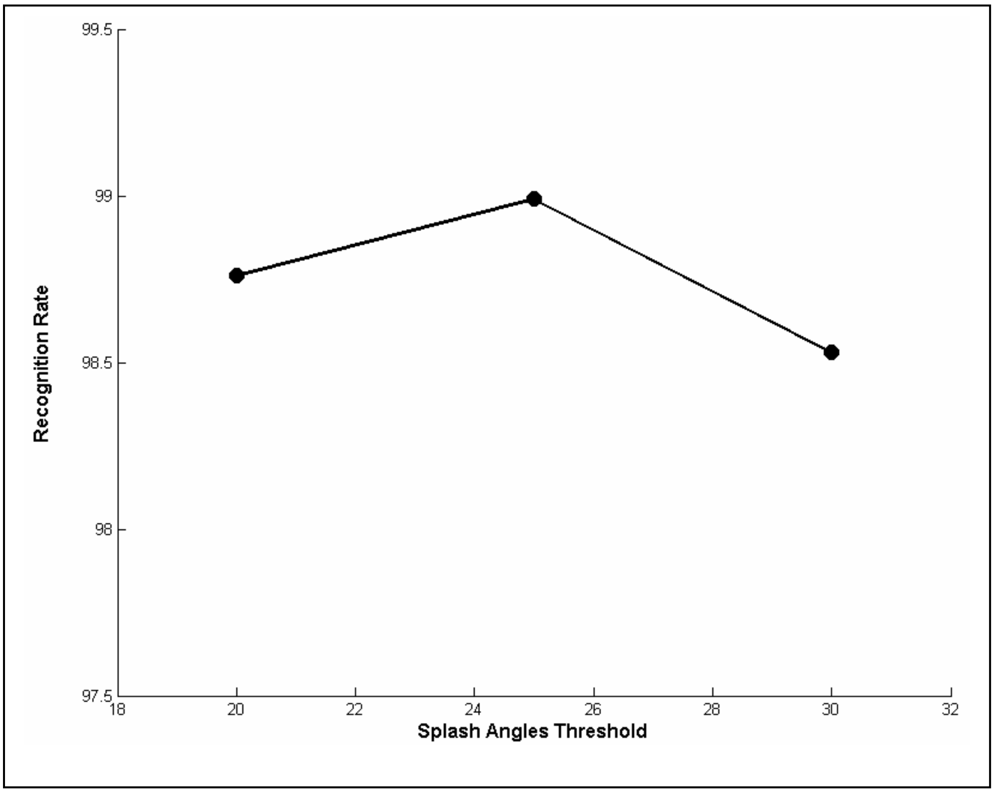


Figure 34. Effect of Changing Splash Angle Parameter

The grahic above shows that recognition rate is raised with increasing splash angle threshold but after a point, the rate is decreased. The optimum value for the splash angle threshold is determined as 25 units.

Thresholds for SAD results are obtained by using fixed values for internal angle threshold and splash angle threshold. Then by varying distance threshold we have observed the change in recognition rate. For three different values of distance threshold, the recognition rates are shown in Figure 35.

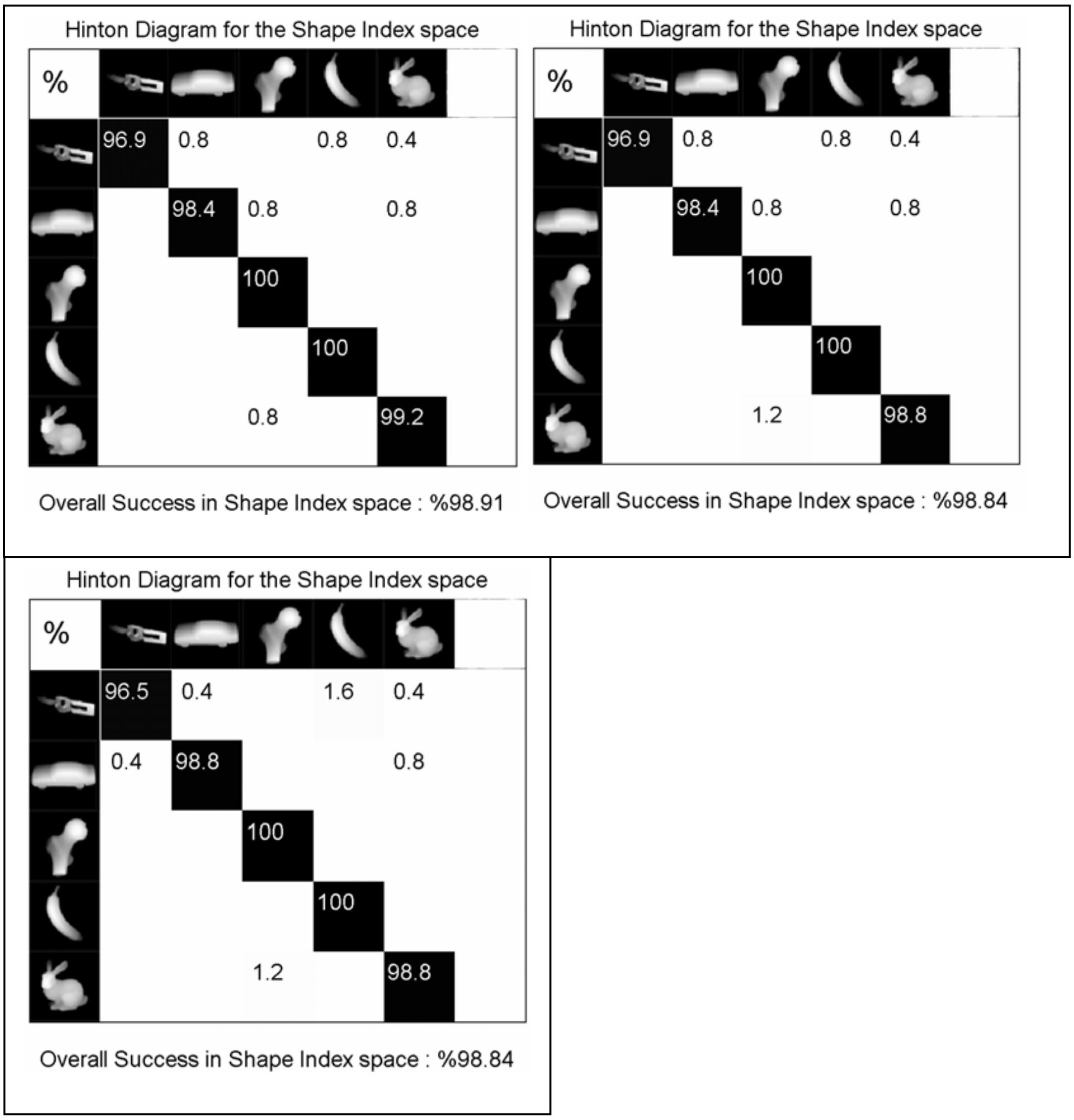


Figure 35. Results for various distance thresholds when internal angle and splash angles are fixed : *thr_int_ang = 25, thr_splash = 25, thr_dist = 20, 30, 40*

Another experiment is conducted to see how the recognition rate is changed with various Euclidean distance threshold values. The splash and internal angles are fixed to predetermined values and then distance threshold is changed. The top left hinton diagram shows the five model results with distance threshold of 20, top right shows the results when the distance threshold is 30 and the last one is the result of threshold value as 40. As seen from the rates, threshold with 20 units provides the highest result.

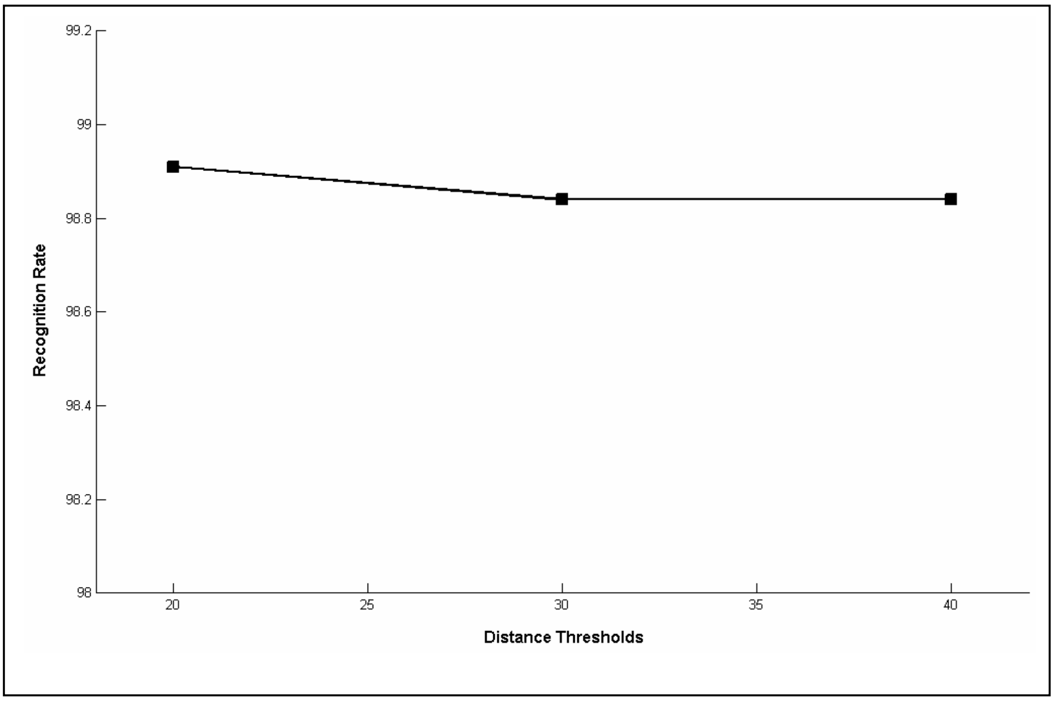


Figure 36. Effect of Changing Distance

The graphic shows how the recognition rate is changed with varying distance threshold. It is also seen that with increasing distance threshold, the rate is decreasing. From the experiments optimum value for the distance threshold is obtained as 20 units.

Thresholds for SAD results are obtained by using fixed values for splash angle threshold and distance threshold. Then by varying internal angle threshold we have observed the change in the recognition rate. For three different values for internal angle, the recognition rates are shown in Figure 37. Top left hinton diagram is obtained when the splash angle threshold is fixed to 25, distance threshold is fixed to 20 and internal angle is used as 25. Top right is the internal angle with 30 units and the last one with 40 units.

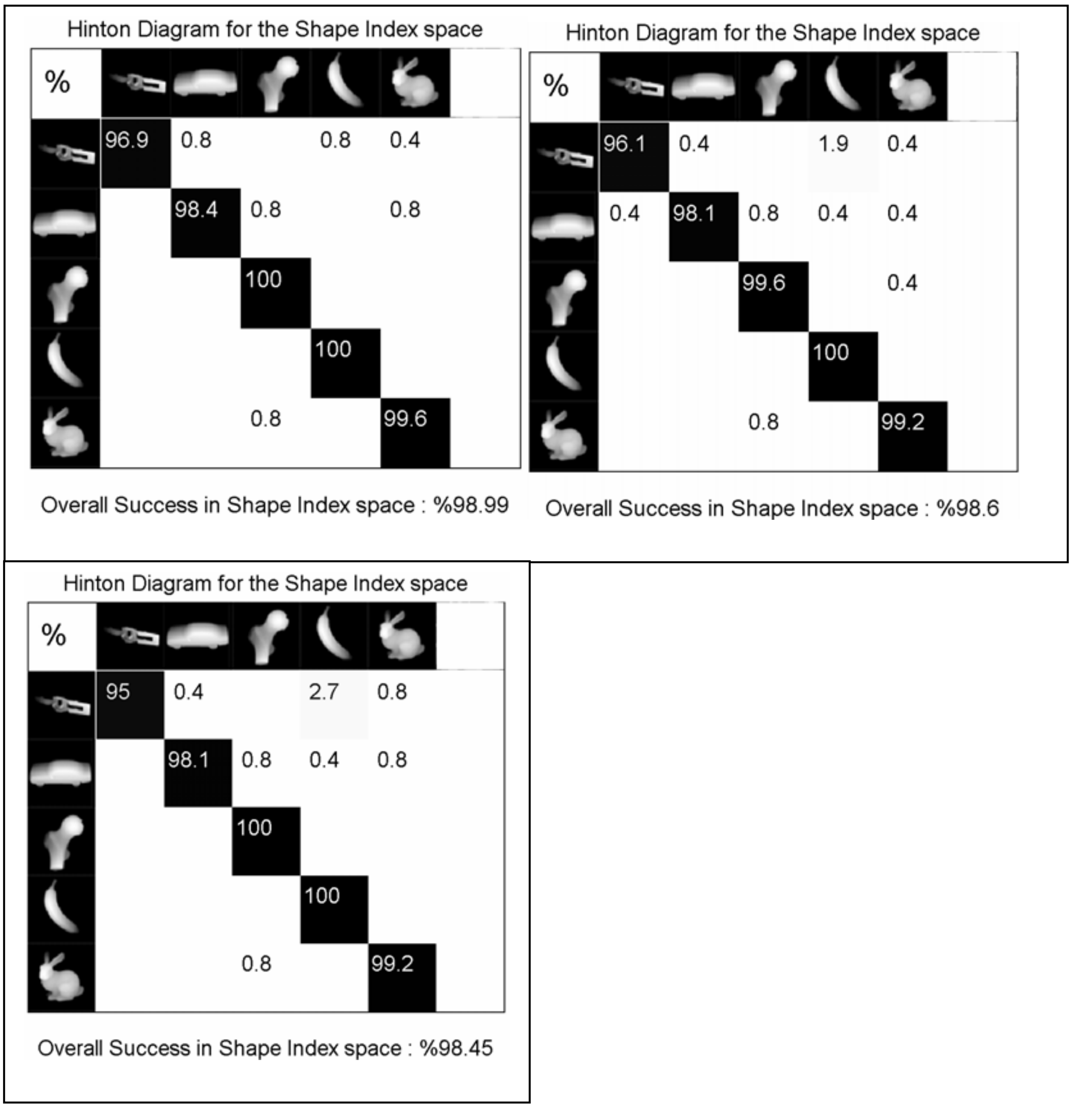


Figure 37. Results for various internal angle thresholds when distance and splash angles are fixed : *thr_int_ang = 25, 30, 40, thr_splash = 25, thr_dist = 20*

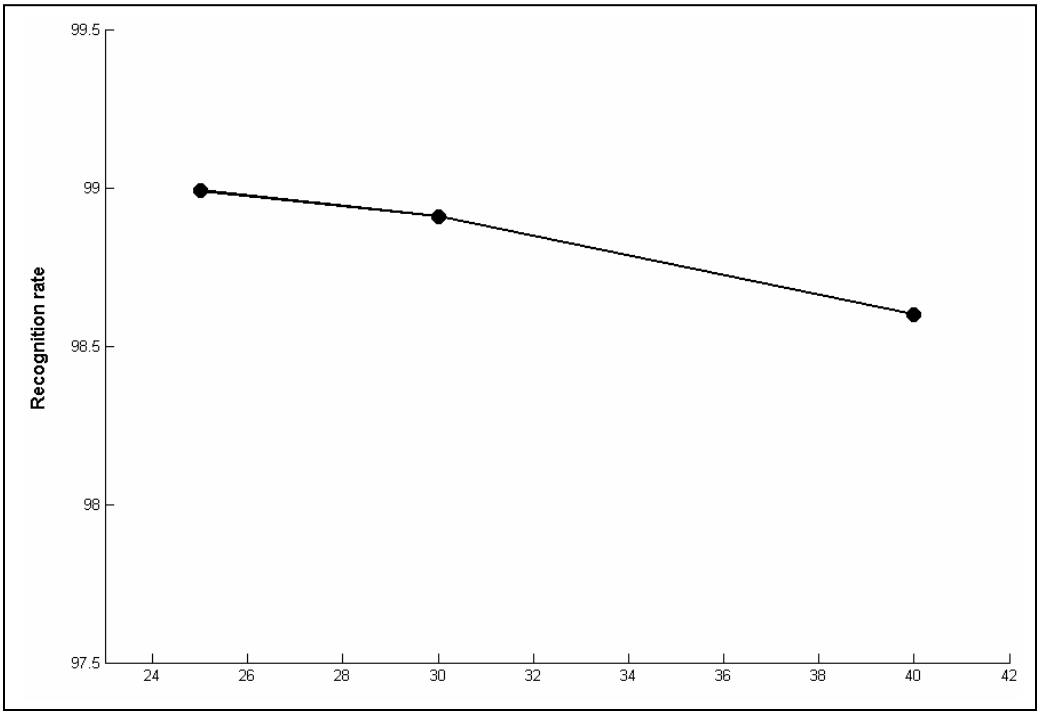


Figure 38. Effect of Changing Internal Angle

Last graphic shows the recognition rate against varying internal angle threshold. Increasing internal angle threshold has not increased recognition rate. It causes the rate to fall down. For internal parameter optimum value is found as 25 units.

Results obtained from the recognition process are seen in Figure 37. As seen from hinton diagrams, various threshold values result different recognition rates. We have tried several threshold values to obtain a stable and high recognition rates. It is also seen from the diagrams that the highest rate is obtained with thresholds (***thr_int_ang, thr_splash, thr_dist) = (25, 25, 20)*** in the overall recognition. Beside various splash angles, internal angles and distance threshold trials, we have tried different splash radius values while extracting splash profile of the basis. Also, varying splash radii have affected the recognition rate. In the experiments above splash radii of (10, 15, 20) pixels are used. By using three different radii, three super splash profiles are obtained. Then by changing splash radii, new results have been obtained and compared. Also these results have allowed us to determine stable threshold parameters for the algorithm.

Results for different splash radii are given in Figure 39 where threshold parameters are fixed to ***(thr_int_ang, thr_splash, thr_dist) = (25, 25, 20)*** for which the highest recognition rate was achieved. Therefore only splash radii differences have been analyzed.

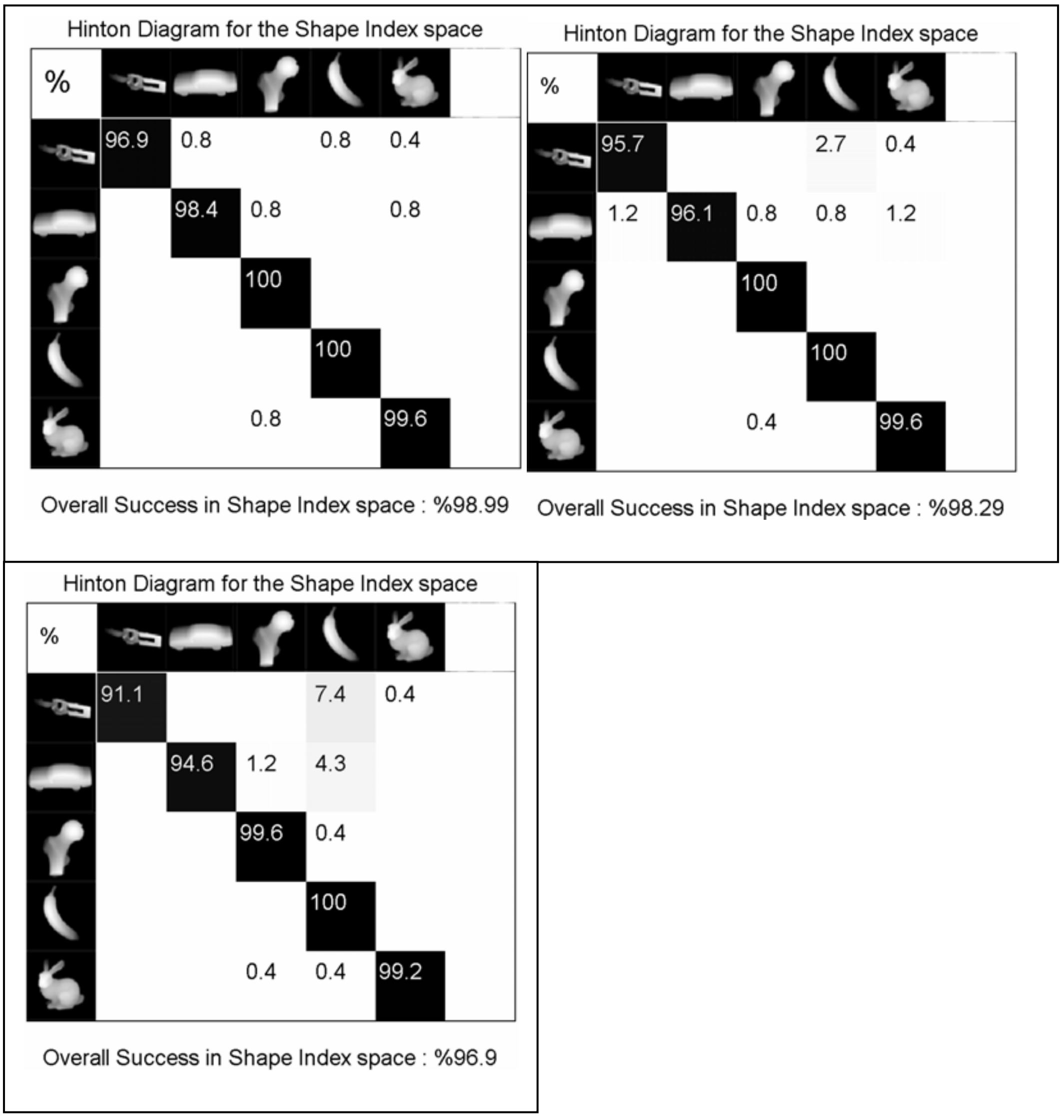


Figure 39. Recognition rates for different splash radii

Recognition rates with splash radii for (10, 15, 20), (15, 20, 25), (25, 30, 35) are shown in Figure 40.

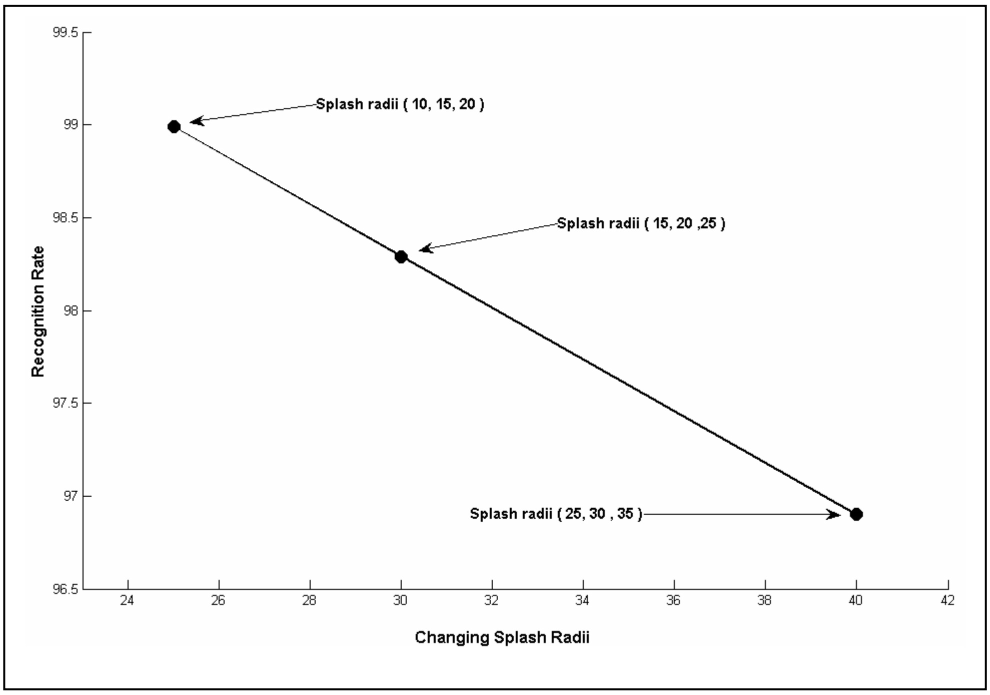


Figure 40. Effect of Changing Splash Radius

In the graphic above, it is obvious that increasing radius of splash region has decreased the recognition rate. The reason of decreasing rate with increased radius is that the splash profile has lost its discriminative property as the radius is increased. Since the splash parameter is obtained by relating surface normals of point with its surrounding region, the obtained splash samples far away from the given point do not define the surface topology well. We always try to describe the region closest to the given point. By sampling irrelevant points, common features of models have raised so the discriminative property of the splash has been lost. System could not distinguish models one from another and votes for wrong models has been increased and recognition rate is decreased.

System has satified % 91.03 recognition rate for 42 models. The elapsed time for the whole recognition process including the preprocessing and the recognition times is 42 hours for 42 models with 66 training and 258 test scene by using dual core 2.2 GHz processor. The hinton diagram described recognition rates depending on gray level (Figure 41). The algorithm is stable against different models. Extraction process has obtained robust results by using shape index values. Curvature of the local surfaces has given dicriminative results to describe the global shape of the object. Splash approach has added extra information about the surrounding region of the feature points. The surface topology is obtained with a number of splash regions with different radii.

Geometric hashing is fast and reliable method to obtain relevant information from the database. It obviated the unnecessary checks for matching process. Object centered coordinate frame definition is difficult to obtain but serves robust representation of objects even if rigid transformation is applied to the object.

Table 3 has shown different methods used for object recognition in the past. They have attractive results considering recognition rates. But model database consists few number of models (at most 15 models).

Table 3 Recognition rates for previously applied recognition techniques.[27]

Recognition Technique	Features	Database Size (Objects)	Recognition Rate (%)
Splash [1]	Normals (structural indexing)	9	NA
Point signature [26]	Distance	15	100
Spin image [24]	Surface histogram	4	100

Overall recognition rates are shown in Figure 41. Hinton diagram displays cross recognition rates between models. Gray level determines the rate of the recognition. Black is for high and white is for low recognition rate.

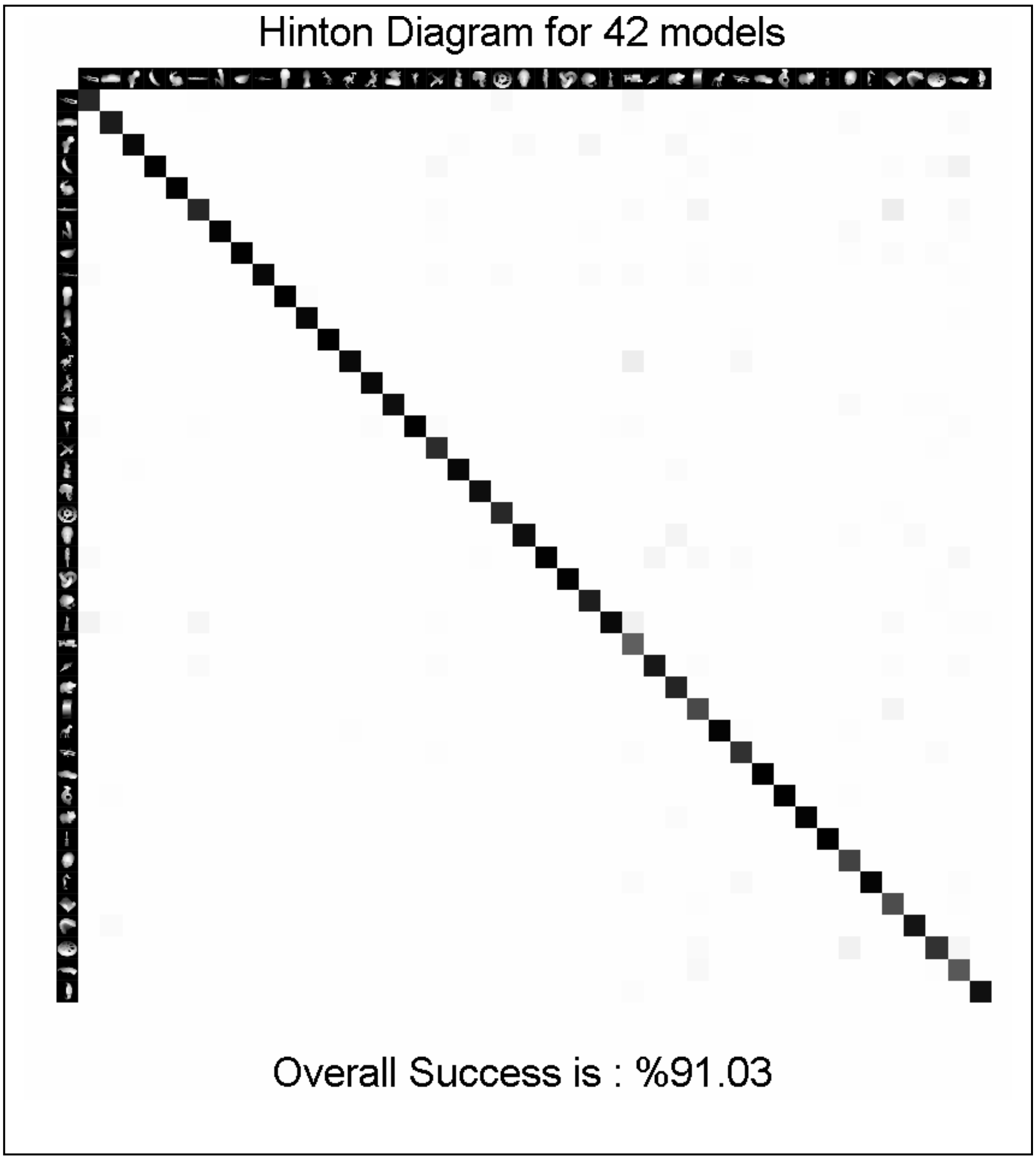
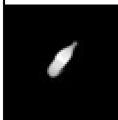
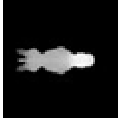
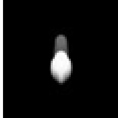
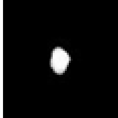
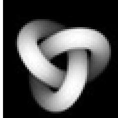
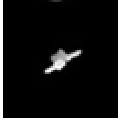
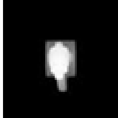
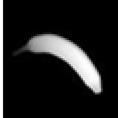
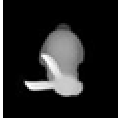
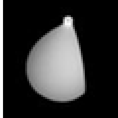
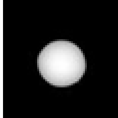
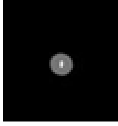


Figure 41.Hinton Diagram for 42 Models.

In our experiments we have obtained recognition rates for each model separately. We have observed that some models had quite high recognition rates, while some of them had relatively low. Models which had the highest recognition rates are banana, bunny, chicken, club, deoflach, dino, female, Isis, knot, pittbull, rocker, screwdriver and seahorse. The models having relatively low recognition rates are vette, machine and pedaltop.The difference in recognition rates among models are because of distinctivity strength of the extracted features. In some of the models the extracted features and their splash and geometric relations are so distinctive that we can distinguish them from the other models easily. But as the number of distinctive features of a model is decreasing, similarity among the models is rising. That makes the recognition system assume some features as if they are belonging to other models. That is why relatively low rates are obtained for some of the models.

In Figure 42 models with highest and lowest recognition rates are shown. These results show the performance of the algorithm for selected objects among 42 models. In Figure 42 left column shows the models having recognition rates higher than % 98 and the right column shows the models having the lowest recognition rates.

Models		Recognition Rates %
	seahorse	98
	pitbull	98,45
	chicken	98,45
	Isis	98,8
	knot	98,8
	female	99,2
	rocker	99,2
	banana	99,2
	bunny	99,2
	club	99,2
	deoflach	99,2
	dino	99,6
	screwdriver	100

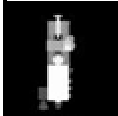
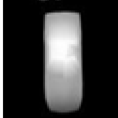
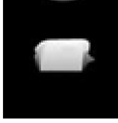
Models		Recognition Rates %
	machine	62,8
	vette	65,5
	pedaltopf	71,3

Figure 42. Models with the highest (left column) and the lowest (right column) recognition rates

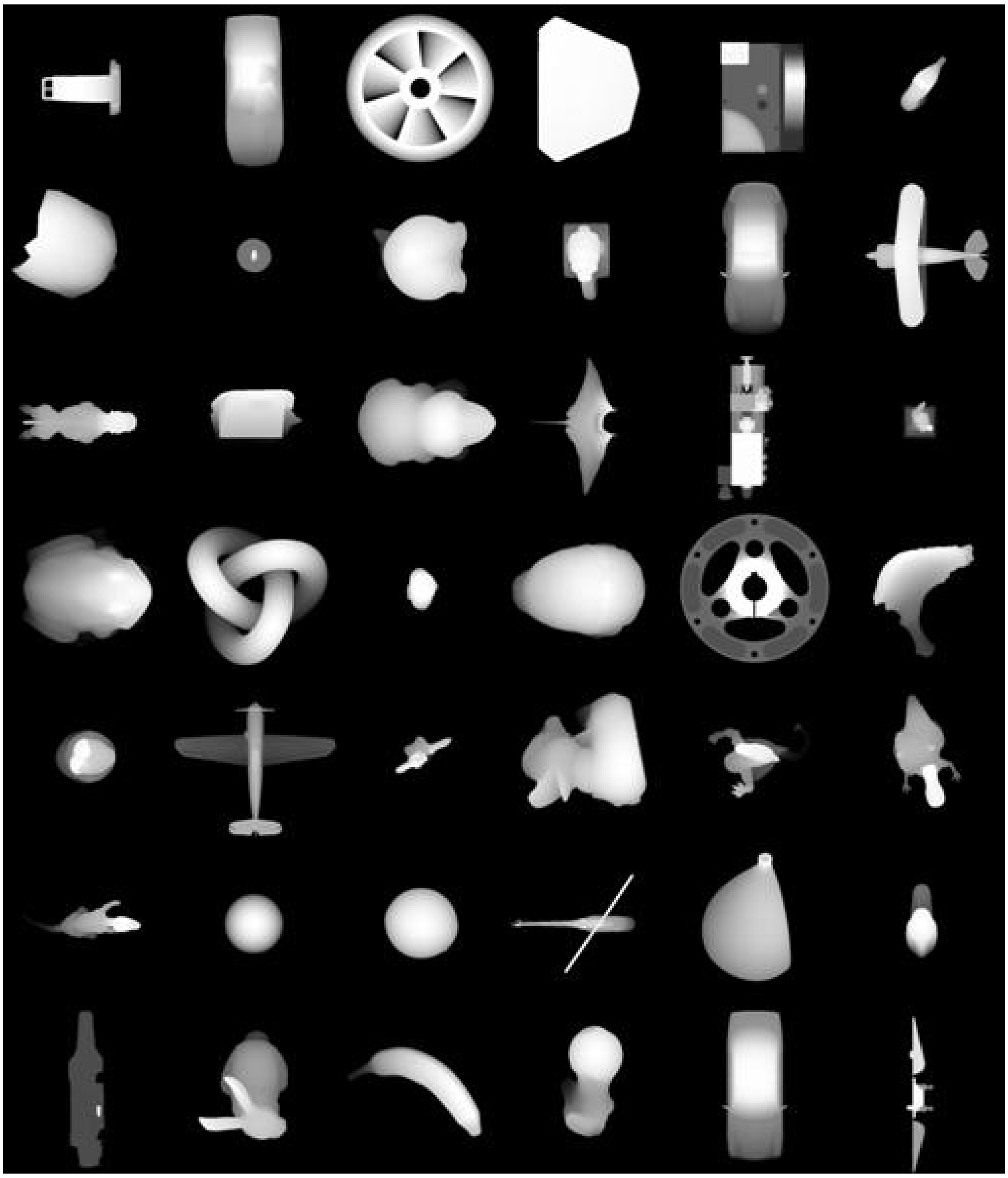


Figure 43. Dataset Models Used for Recognition

CHAPTER 6

CONCLUSION

This thesis aim is to work on 3D object recognition techniques and develop more robust and efficient methods based on algorithms in the literature. We have used different approaches to establish a more robust system. The whole process comprised three basic stages.First is the feature extraction stage, where the quality of object representation affects the performance of the recognition process directly. Second one is the feature description stage. Meaningful information is obtained by relating and describing points with eachother. Last one is the matching stage. The final step of the whole process checks if the target object exists in the model library or not.

In our implementation we have used local surface curvatures for the feature extraction stage. The shape index which is a dimensionless, scale invariant parameter is used to extract features from the object surface. Local patches are extracted as interest regions. Extracted salient regions are represented with their center points and all features have been served as a point cloud. This data extraction process provides transformation invariant point set to the next stages.

Feature description stage of the algorithm is also very important. The point dataset obtained from the extraction stage is processed and used for description of the model. The splash approach is used to obtain the surface topology from the given points. Also the splash information provides additional information to the surface curvature points.

All of the computed data is used in the geometric hashing algorithm for the purpose of the object recognition.3D geometric hashing is fast and reliable method which provides easy and fast access to the relevant information and obviates irrelevant information for fast computation. Although definition of the object centered coordinate frames is difficult representing the point dataset by an object centered frame is transformation invariant and that is why this method is robust enough to the partial occlusion.

Object recognition researches have always considered two important performance criteria which are computation cost and recognition accuracy. The trade off between these concepts is always analysed and system requirements determined the better choice. In this thesis, both concepts have been evaluated so that high recognition accuracy with low computation cost is achieved as much as possible. By our hybrid solution for 3D object recognition, we obtained promising results. Experiments on the database with 42 models satisfy % 91.03 recognition rate. When we compared our results with some previously implemented studies, we can clearly state that thesis provides precedence over latest researches.

REFERENCES

[1] 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

[2] B. Hudson, Arts 101 Lecture web page http://facstaffwebs.umes.edu/bphudson/pixs/Edgerton,MilkSplash.jpg, (accessed December 22 , 2008)

[3] J.J. Koenderink, A.J. van Doorn, Surface shape and curvature scales, Image and Vision Computing 10 (8) (1992) 557–565.

[4] J.J. Koenderink, A.J. van Doorn, Local structure of Gaussian texture, IEICE Transactions on Information and Systems 86 (7) (2003) 1165–1171.

[5] M. Lillholm, L.D. Griffin, Statistics and category systems for the shape index descriptor of local 2nd. Order natural image structure.., Image Vis. Comput. (2008), doi:10.1016/j.imavis.2008.08.003

[6] H. Ragheb, Edwin R. Hancock, Darboux smoothing for shape-from-shading. Pattern Recognition Letters 24(1-3): 579-595 (2003)

[7] Chu-Song Chen, Yi-Ping Hungm, Ja-Ling WuEJ, Model-Based Object Recognition Using Range Images by Combining Morphological Feature Extraction and Geometric Hashing, Pattern Recognition, 1996., Proceedings of the 13th International Conference (1996) , vol.1 565-569

[8] V. A. Toponogov, Differential Geometry of Curves and Surfaces, A Concise Guide (2006), ISBN-10 0-8176-4384-2

[9] S.Mann, Gaussian Curvature, (http://www.cgl.uwaterloo.ca/~smann/Research/gaussiancurvature.html) (accessed 25 December, 2008)

[10] P.C.J. Hoogenboom, Classification of Shell Surfaces, (http://www.mechanics.citg.tudelft.nl/~pierre/CT5143_chap2-3.pdf), (accessed January 4 , 2009)

[11] 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.

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

[13] Robert B. Fisher, School of Informatics University of Edinburgh, http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/CANTZLER1/cvonline.htm (accessed December 28, 2008)

[14] R. Fisher, S. Perkins, A. Walker and E. Wolfart ,School of Informatics University of Edinburgh, 2003, http://homepages.inf.ed.ac.uk/rbf/HIPR2/label.htm (accessed December 24, 2008)

[15] John J. Craig, Introduction to Robotics Mechanics and Control 2nd Edition, Addison Wesley Longman Press ,1989,ISBN : 0-201-09528-9

[16] Weisstein, Eric W. "Collinear." From MathWorld--A Wolfram Web Resource. http://mathworld.wolfram.com/Collinear.html (accessed December 28, 2008)

[17] Weisstein, Eric W. "Rotation Matrix." From MathWorld--A Wolfram Web Resource. http://mathworld.wolfram.com/RotationMatrix.html

[18] H. J. Wolfson and I. Rigoutsos (1997). Geometric Hashing: An Overview, IEEE Computational Science and Engineering, 4(4), 10-21.

[19] F. Stein and G. Medioni, Efficient Two-Dimensional Object Recognition, Proc. Int'l Conf. Pattern Recognition, IEEE Computer Society, 1990, pp. 596–600.

[20] Y. Lamdan and H. J. Wolfson Geometric hashing: A general and efficient model-based recognition scheme, In Proceedings of the Second International Conference on Computer Vision, (1988) pages 238-249.

[21] R.Fisher ,School of Informatics University of Edinburgh, http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/CANTZLER2/range.html (accessed December 24 2008)

[22] The University of Stuttgart. Stuttgart range image database, 2001. http://range.informatik.uni-stuttgart.de/htdocs/html/. (accessed January 2, 2009)

[23] F.Arman,J.K. Aggarwal, Model Based Object Recognition in Dense Range Images- A Review, ACM Computing Surveys, Vol. 25, No. 1, March 1993

[24] A. Johnson, Spin-Images: A Representation for 3-D Surface Matching, doctoral dissertation, The Robotics Institute, Carnegie Mellon Univ., 1997.

[25] A. Johnson, M. Hebert, Using Spin Images for Efficient Object Recognition in Cluttered 3D Scenes, IEEE Transactions on Pattern Analysis and Machine Intelligence, May 1999, Vol. 21, NO. 5,

[26] Chin S Chua, Ray Jarvis, Point Signatures: A New Representation for 3D Object Recognition, International Journal of Computer Vision, Vol. 25, No. 1. (1 October 1997), pp. 63-85.

[27] R.J. Campbell ,P.J. Flynn A survey of free-form object representation and recognition techniques Computer Vision and Image Understanding Volume 81 , Issue 2 (February 2001) Pages: 166 – 210 ,2001, ISSN:1077-3142

[28] P. V.Miranda, Alexandre X. Falcao, A. Rocha, and F. Bergo, Object Delineation by K-Connected Components, EURASIP Journal on Advances in Signal Processing Volume 2008, Article ID 467928, 14 pages doi:10.1155/2008/467928

[29] D.G.Lowe, Distinctive image features from scale-invariant keypoints, International Journal of Computer Vision, 2004, Vol.60, pp 91-110

[30] D.G.Lowe, Object recognition from local scale-invariant features, International Conference on Computer Vision, Corfu, Greece (September 1999), pp. 1150-1157