Bag of Feature Graphs: A New Method for Non-rigid 3D Shape Retrieval

A Thesis Presented

by

Xiaohua Hou

to

The Graduate School

in Partial Fulfillment of the

Requirements

for the Degree of

Master of Science

in

Computer Science

Stony Brook University

December 2011

Stony Brook University

The Graduate School

Xiaohua Hou

We, the thesis committee for the above candidate for the

Master of Science degree, hereby recommend

acceptance of this thesis.

Hong Qin – Thesis Advisor
Professor, Computer Science Department

Xianfeng Gu – Second Reader
Associate Professor, Computer Science Department

Klaus Mueller – Third Reader
Associate Professor, Computer Science Department

This thesis is accepted by the Graduate School

Lawrence Martin
Dean of the Graduate School

Abstract of the Thesis

Bag of Feature Graphs: A New Method for Non-rigid 3D Shape Retrieval

by

Xiaohua Hou

Master of Science

in

Computer Science

Stony Brook University

2011

This thesis presents a new paradigm for non-rigid 3D shape retrieval, which is also called *Bag of Feature Graphs* (BoFG). The main idea is to connect only the features on the shape to construct the graphs so that the number of points involved in the computation is greatly reduced. Given a vocabulary of geometric words, the BoFG approach generates a graph that preserves the spatial information among features for each word. The spatial information is weighted by its similarities to each word so that points unlike the word category are eliminated. And the graphs are captured by the affinity matrices of *Weighted Heat Kernels* (WHK) whose eigenvalues form a shape descriptor. Also, the BoFG approach can supports partial 3D shape retrieval by coupling with graph matching techniques and comparing only sub graphs that represent common parts of the shape. Finally, experiments are conducted and show that the proposed BoFG method is faster to compute and the retrieval performance is also competitive compared with other state-of-the-art methods.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

BoFG	Bag of Feature Graphs
WHK	Weighted Heat Kernels
BoF	Bag of Features
SS-BoF	Spatial-Sensitive Bag of Features
SPM	Spatial Pyramid Matching
HK	Heat Kernel
HKS	Heat Kernel Signature
SIFT	Scale Invariant Feature Transform
SHD	Spherical Harmonic Descriptor
LFD	Light Field Descriptor
SI-HKS	Scale-invariant Heat Kernel Signature
VHKS	Volumetric Heat Kernel Signatures
ANN	Approximate Nearest Neighbor
MDS	Multi-Dimensional Scaling
FEM	Finite Element Methods
PR	Precision Recall
mAP	Mean Average Precision

Acknowledgements

This thesis would not be possible without the support of many people.

In the first place, I would like to record my gratitude to Professor Hong Qin, who is my thesis advisor, for his supervision and advice from the very start of my research. Professor Qin introduced me to the field of 3D shape retrieval and shared me extraordinary experiences throughout my work. Without his continuous support and encouragement, I would say it is impossible for me to complete this endeavor.

I also gratefully acknowledge Tingbo Hou, who is right now a senior PhD candidate in our lab, for his advice and guidance. Tingbo shared his knowledge with me and helped me fully understand a lot of important concepts in the field of 3D shape retrieval which are the keys to my work.

Finally, I would like to give my special thanks to all my lab mates for the patient science discussion every day. They made our lab a really nicer place to work. I will always be indebted to all my family members for believing in me. It is to them that I dedicate my work. Moreover, I would like to thank everybody who made contributions to the realization of this thesis, as well as expressing my apology that I could not mention personally one by one.

Chapter 1 Introduction

Since recent years, with the advances in 3D computer modeling, 3D scanning and other related 3D technologies, a lot of non-rigid 3D shapes have been created and used for various purposes. Shapes of mechanical parts are commonly stored in 3D form for manufacturing and assembling. Molecular structures like protein bonding are also represented by 3D shapes, etc. Because of the high demand for large number of 3D shapes, large 3D model repositories are formed and have already been widely available on the public domain. Therefore, given a shape as a standard input, an operation of retrieving similar or related 3D shapes in one or more 3D shape repositories becomes very necessary and popular. Such operation is also commonly called as “3D shape retrieval” in the field of researches, which has received considerable attention in recent years. Also, because of its popularity, it has led to the development of several 3D shape search engine such as the 3D model retrieval system at the National Taiwan University [56, 21], 3D model search engine at Princeton University [55, 27], the Ogden IV system at the National Institute of Multimedia Education, Japan [51, 39], etc. There are various kinds of approaches for 3D shape retrieval, among those, more researchers have put their interest on the feature based approach based on the *Bag of Features* (BoF) [33] framework recently, which originally comes from the 2D image retrieval fields, due to its simplicity and good performance in practice. This chapter will first give the problems and challenges we met in the field of 3D shape retrieval then introduce the main contributions of our new proposed approach in this thesis and will give the thesis structure finally.

1.1 Problems and Challenges

Non-rigid 3D shape retrieval has been a must for large 3D shape repositories available on the public domain and is a challenging problem in the computer vision and recognition area. While there are various kinds of approaches for the non-rigid 3D shape retrieval, the retrieval performance of them is not generally good for all. Few methods can reach a very high retrieval performance. Lately, the *Heat Kernel Signature* (HKS) [38] provides a way of informative data representations in computer graphics and making itself very suitable for non-rigid 3D shape retrieval. The recent state-of-the-art accomplishment is achieved by the "Shape Google" [11, 10, 7], which utilized the HKS as the feature descriptors and combined with the BoF framework. However, the "Shape Google" related methods integrate word similarities over the entire shape, which excessively compress the geometry and costs a lot of computation. Furthermore, another disadvantage is that it totally loses the spatial information because the approach follows the BoF pipeline. And, it is also unsuitable for partial shape retrieval.

These issues are of great importance to the research of 3D shape retrieval. First of all, spatial information may provide the better power of discrimination among shapes. Totally ignoring such kind of information greatly affects the retrieval performance. Secondly, less computation effectively guarantees that such the retrieval process is on-line and reduces lots amount of time for the off-line pre-computation. Finally, for partial shape retrieval, it has already become a high demand in the practical use. Therefore, one good and complete 3D shape retrieval method has to support the partial case of the retrieval.

There comes the challenges that how can we devise a new approach of 3D shape retrieval that incorporate the spatial relation information and improve the time efficiency while still keep a high retrieval performance that an original BoF framework does. The difficulty here is that how we can extract the key spatial information to improve the retrieval performance and rely on fewer points on the shape in order to reduce the time complexity. Furthermore, the new approach should still be invariant or robust to different kinds of non-rigid 3D shape transformations. Also it should easily support the case of partial 3D shape retrieval.

The challenges here are greatly related to the paradigms of 3D shape representations. An intrinsic approach is to represent the data of the 3D shape by connecting all the points on the shape and constructing a graph. The left graph in Figure 1.1 shows the paradigm; such a graph faithfully characterizes the shape by using an affinity matrix such as geodesic matrix, diffusion matrix and heat kernel matrix, etc. However it is not a concise representation and is dominated by the non-salient points on the shape, which downgrades the discrimination power of the shape retrieval.

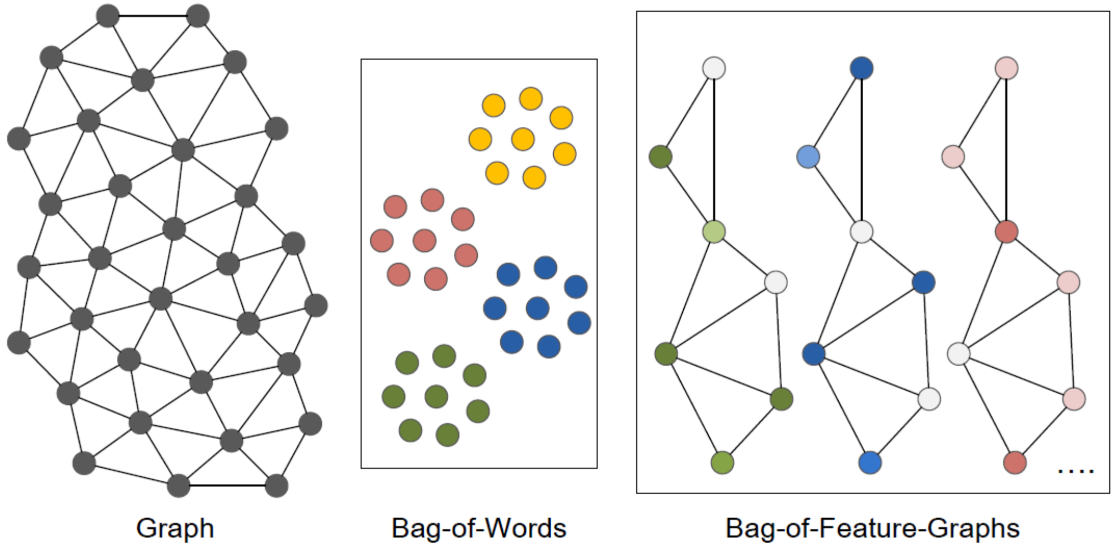


Figure 1.1 Shape representation paradigms

Recently, the feature based 3D shape retrieval method based on BoF framework, which is also called "Bag of Words", dominates this research field due to its generality and relative high performance. "Shape Google" uses this paradigm as the middle graph shown in Figure 1.1. It groups the points to different geometric words given a geometry vocabulary. The shape descriptor is formed by histograms that represent the frequencies of the geometric words associated on the shape. As mentioned before, the BoF loses the spatial information in the shape comparison. For the 3D shape retrieval, considering the spatial information may improve the descriptive power and have a greatly impact on the retrieval performance. In 2D field, one of the successful approaches is called Spatial Pyramid Matching (SPM) [35]. Though, in [11], a paradigm called *Spatial-Sensitive Bag of Features* (SS-BoF) is introduced and it compresses the spatial information by counting frequencies of the word pairs. As a result, the SS-BoF severely increases the time complexity of the computation because of the nature of BoF framework that it is based on the distribution of large number of samples and also because of the point to point computation.

1.2 Contributions

In the previous section, we have discussed the problems and challenges that we are trying to solve in the field of the 3D shape retrieval and are going to present our new approach in Chapter 4 in details. Our motivation is to provide a concise and spatially-informative 3D representation paradigm for non-rigid 3D shape retrieval. The main contribution of this thesis is called BoFG. The key concept is to construct graphs of a 3D shape based on only features on that shape, as the right graph shown in Figure 1.1. The BoFG builds the graphs associated with the geometric words, which capture geometric relation of only features weighted by their similarities to the words. Here, for the purpose of this thesis, we first give an overview of the contributions.

- It is fast to compute by reducing the number of points on the shape involved in the representation. Therefore it is much more concise.

- Sspatial information is considered by constructing diffusion matrix only among feature points, which is a key factor that influences the retrieval performance.

- It is representative because of the nature of the feature points that they are salient points on the 3D shape and contain important information.

- Graphs have different dominating features associated with corresponding geometric words respectively, which improve the accuracy of shape retrieval greatly.

- It supports the partial shape retrieval by first applying the graph matching technique. And then select the corresponding rows and columns from the BoFG matrices.

1.3 Thesis Structure

About this thesis, in the following Chapter 2, we will give a general overview of the related work that recently researchers did in the area of 3D shape retrieval and introduce some state-of-the-art methods. And then, in Chapter 3, we introduce our proposed new approach in details for the same research topic. Chapter 4 shows the experiments and results we did and gives a comparison and evaluation between our method and the state-of-the-art “Shape Google” related approaches that published before. Chapter 5, which is the last chapter, finally will summarize our work and suggest some points that in future we can improve.

Chapter 2 Related Work

Though 3D shape retrieval is a new and challenging research topic, a lot of work has been done by others. The main goal of the 3D shape retrieval is to find a discriminative and effective representation as a signature of the shape, and then the similarity measure is carried on. In particular, for recently proposed content based 3D shape retrieval methods, different kinds of features and descriptors are used to represent the shapes and results vary. In Chapter 2, we present the related work by first giving a general background of the 3D shape retrieval and introducing the different kinds of content based 3D shape retrieval methods in broad categories then. Also we will put our focus on some state-of-the-art 3D shape retrieval approaches and the *Heat Kernel* (HK) related method, which is called "Shape Google" and has been proved to be better recently. Besides that, we also use the *Heat Kernel Signature* (HKS) as the feature descriptors in our new approach, which will be discussed in Chapter 3.

2.1 3D Shape Retrieval Background

The retrieval process for 3D shapes is very similar to our searching on the "Google" website. Except that the input of the query is a 3D object instead of a string. However, compared to the original text searching, which has already been a mature technology, the technology of 3D shape retrieval is much newer and more challenging. Trying to find a 3D shape using the textual information and use the traditional text-based engine could not work in a lot of cases. Therefore, 3D shape retrieval based on the properties of the 3D shape itself to search similar shapes [28] has been a very popular research area in the last decade. In Section 2.1, we generally present some ways people did the 3D shape retrieval before. And we introduce the general 3D shape retrieval framework and the feature based 3D shape retrieval approach as a background.

For the basics of the 3D shapes, for example the 3D shape representation, the 3D file format used in our experiments, we can refer the Appendix A in details.

2.1.1 3D Shape Retrieval Framework

Here we refer the Figure in [42] showed here as Figure 2.1, which describes the conceptual framework for the 3D shape retrieval. The whole retrieval framework is formed by a 3D model database, the indexed data structure and an online query engine. Each 3D model is pre-computed and identified as a descriptor in the database. The online query engine accepts the query shape as the input and computes the descriptor. Finally, shapes in the database are retrieved by the dissimilarity measure of the descriptors and retrieved shapes can be visualized. Three approaches can be distinguished [42]: (1) browsing to select a new query from model id, (2) query by a descriptor, and (3) query by an existing 3D shape.

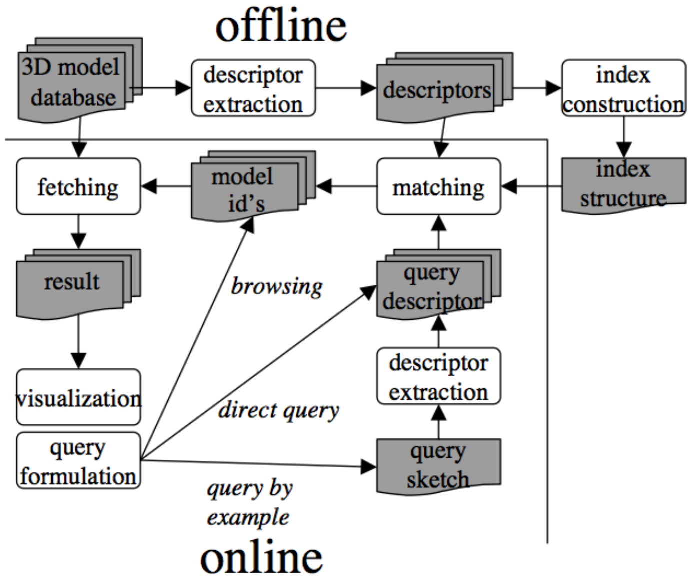


Figure 2.1 The conceptual framework for 3D shape retrieval (Referred from [42])

2.1.2 Earlier 3D Shape Retrieval Approaches

There are a lot of earlier ways to accomplish the 3D shape retrieval process. We are going to discuss some of them blow.

An earliest and very easy method of doing 3D shape retrieval is to compare the geometric information of the points one by one. This information includes the relative positions among all the points, etc. However, it is inefficient in computation because, for a particular model, the totally number of points is hundreds to thousands and we need to compare all these points. Furthermore, this method is variant to transformations like isometric deformation, scaling, etc. For a 3D shape, it is always subjected to different kinds of isometric deformation. These will result the geometric value changes of the points. Therefore, with this method, the retrieval usually will not return correct results, especially for non-rigid 3D shapes.

Another relatively easier way of the 3D shape retrieval is to tag the 3D shapes by the text. In that case, the advanced text searching technologies can be applied to the 3D shape retrieval.

This idea is exactly the same that we search videos on the "YouTube", where all videos are tagged by text for searching. However, this method has lots of disadvantages. First of all, for the same objects, the tags might be different because the meaning of the objects might be different to different people because the annotation added by human beings depend on language, culture, age, sex, and other factors. They may be too limited or ambiguous [42]. And the tag itself is misleading sometimes. Secondly, manually tagging the whole 3D shape repository is time consuming. Therefore, the above tagging approach for 3D shape retrieval is not a reasonable choice from recent points of view.

2.1.3 Recent 3D Shape Retrieval Approaches & Bag of Features

Recently, the feature based approach combined with the BoF framework, which is originally from the 2D image retrieval, is getting much more focus for the 3D shape retrieval. In the 2D image retrieval, traditional images are represented by various kinds of features. These features have important information and can be found in transformed versions of the same image. Features in an image can vary from methods to methods. Some famous ones are SIFT [25], MSER [26], SURF [2], etc. For a general overview of 2D image retrieval, we refer people to read the paper proposed by Veltkamp and Hagedoom [45]. Same in the area of 3D shape retrieval, the features on each shape are extracted and represent the shape. These features are usually stable on the shape and invariant under variants of 3D transformations. Different kinds of features can be applied to the shapes and the retrieval performance varies.

The BoF concept was also originated from the 2D image retrieval, and now is widely used in 3D shape retrieval scenarios. The way that the BoF works is that it first extracts the features from the objects, learns the "vocabularies" based on the typical features on different shapes, and quantizes features on each using these vocabularies generated. Finally, it represents objects by frequencies of different vocabularies and does the comparison between shapes. Further explaining the step, which is called "Learning the vocabularies", a set of "vocabularies" is a collection of features after clustering the typical features from the sample shapes in the database. These collections of features considered as vocabularies are then used as a standard to quantize the objects and generate the histogram of each object. One key point of the step is the choices of clustering algorithm. Currently, there are lots of clustering algorithms available, for example, K-means [36], spectral clustering, etc. For the most cases, people use K-means because of its simplicity and high performance. Although it has already been widely used in the BoF framework of the 2D images field, there are still some issues of "vocabularies" under research, for example, "How to choose the vocabulary size?", "Use generative of discriminative learning" and "computational efficiency", etc.

Another thing in BoF model we need to consider is the representation of the shapes. Currently, the histogram, both soft and hard version [11] [7], of the occurrence of the "geometric vocabularies" is widely used. One method of generating the histogram or "*Bag of Features*" that proposed in "Shape Google" methods [7] is to integrate the feature distribution over the entire shape and yield a $V \times 1$ vector, as shown in top of the Figure 2.3,

$$f(X) = \int_X \theta(x) d\mu(x) \tag{2.1}$$

where $\theta(x)$ is the feature distribution of vertex *x* on Shape *X*. Details can be referred in [7].

Using this representation, we can define a distance between two shapes *X* and *Y*:

$$d_{Bof}(X, Y) = \left\| f(X) - f(Y) \right\| \tag{2.2}$$

Shapes have the shortest distance that match the most and are most related.

Recent 3D shape retrieval approaches that using the shape itself as a query and based on the comparison of the geometric and topological properties of shapes are complicated by the fact the many 3D shapes manifest rich variability, and shape retrieval must often be invariant under different class of transformations [9]. As an illustration, the following Figure 2.2 referred from "Shape Google" [11] shows the whole pipeline for a particular recent feature based 3D shape retrieval approach based on a BoF framework.

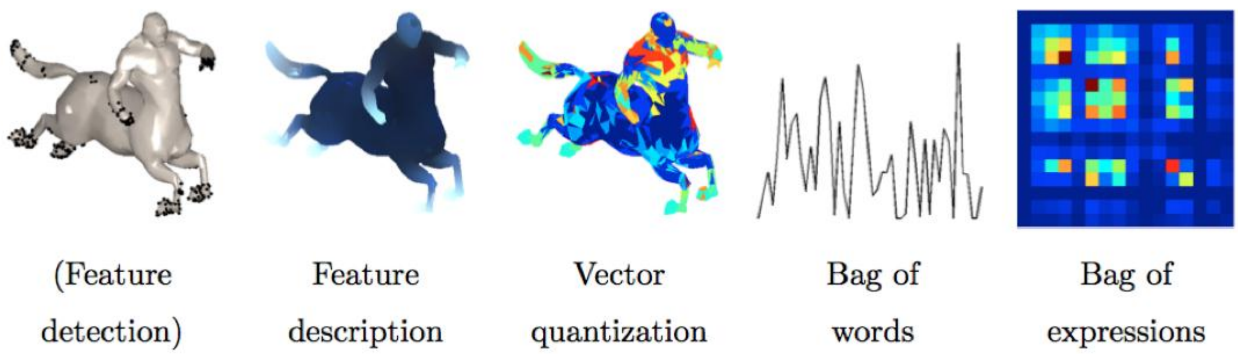


Figure 2.2 The local feature based 3D shape retrieval pipeline (Referred from [11])

As depicted in Figure 2.2, the first step is that features are detected on the shape and the shape is then represented as a collection of local feature descriptors. Next, given a "geometric vocabulary", the shape's descriptors are quantified to the closest words in the "geometric vocabulary". After that, the histogram of the frequency of occurrence of "geometric vocabulary" is generated as the final representation of the shape. By comparing the histograms of the different shapes, shapes with the minimum difference of the histograms are retrieved.

One of the most significant disadvantages of the above mentioned BoF is the fact that they consider only the distribution of the "geometric vocabularies" and only use the plain histogram as the representation. Therefore, the spatial information is totally lost among these vocabularies. However, the spatial information plays an important role in improving the retrieval performance. Researchers have done some improvements that partially or completely preserve the spatial information. For instance, the "Shape Google" [11] also introduces the SSBoF (as shown in right bottom of the Figure 2.3), which accounts not only for the frequency but also partially for the spatial relation information among all the points on one shape, etc. and yields a $V \times V$ matrix that counts the frequencies of word pairs, given by

$$f(X) = \int_{X \times X} \theta(x)\theta(y)^T K_t(x, y) d\mu(x) d\mu(y) \qquad (2.3)$$

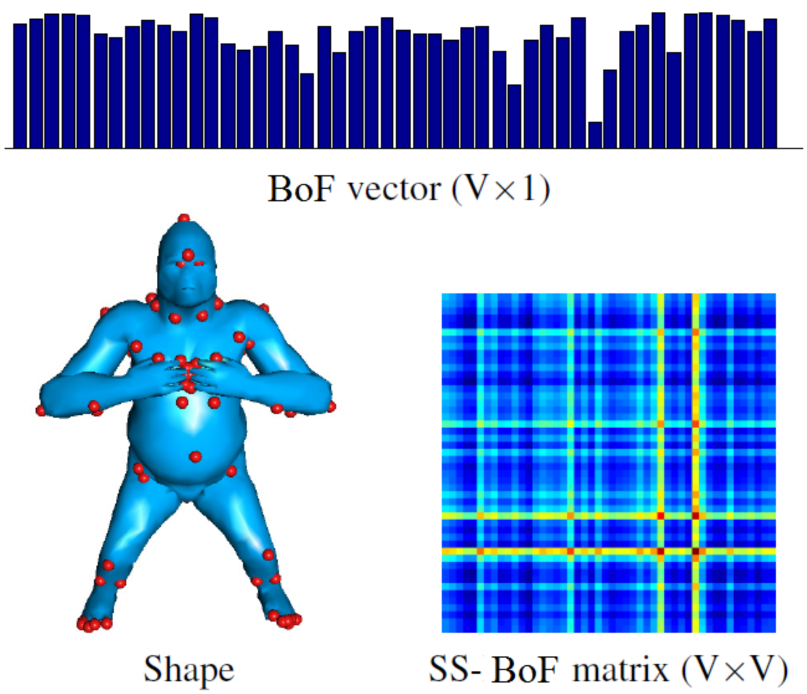


Figure 2.3 Shape representations of BoF and SS-BoF

2.2 Content Based 3D Shape Retrieval Methods

Recent 3D shape retrieval methods are more or less content based. In this section, we will discuss the content based 3D shape retrieval methods according to the following categories. However, we need to note that these classes of the methods are not completely disjoined. For further review, more related work can be found in the survey [42].

2.2.1 Feature Based Methods

Features on a shape denote the geometric properties of that shape. According to the type features used in the methods can be classified into global features and local features.

The global shape descriptor has been widely used in 3D shape retrieval, due to the efficiency of computing. It uses a high-dimensional single descriptor vector, where the dimension is fixed for all shapes, and computes the Euclidean distance to match the shapes. However, it is not discriminative, especially about the shapes’ details. In [47], the author composes a global feature using volume, area; statistical moments and Fourier transform coefficients. And the performance is further improved by introducing an active learning phase in [48]. J. Corney et al [14] introduces convex-hull based descriptor using the ratio of the shape's surface area and the surface area of its convex hull, the percentage of the convex hull part not occupied by the shape, etc. In [30], the shape distribution was introduced based on the measurements of distance, angle, and area between random points on the surface. In [46], geometric and topological feature maps were used for comparing shapes, which capture the amount of effort required to morph a 3D object into a canonical sphere.

The local feature based method captures the local geometry information of the surface shape. Therefore, it has more power of discrimination among similar shapes. Also, it can easily fit the BoF framework, which is used by most of the researchers recently. Local feature descriptors vary a lot. Laplace-spectra [33], eigenvalues of the Laplace operator, are used as fingerprints for shape matching. And “Shape Google” [7] presents a method to a non-rigid shape retrieval using the local heat kernel signatures as the feature descriptor. In “Shape Topics” [22], the spin image is introduced as local features after sampling the mesh vertices. Furthermore, inspired by the *Scale Invariant Feature Transform* (SIFT), [17] describes invariant points on a 3D voxelized model and calculates invariant 3D local feature descriptors at these key points.

2.2.2 Graph Based Methods

Graph based methods try to capture the geometric relation of a 3D shape using the graph representing how the components of a shape are linked. For instance, Reeb graphs and skeleton are traditional graph based methods. In [41], skeleton is composed by the topological and geometric features and is used as a representation of the 3D shape for deformable model retrieval. Three types of geometric information is then computed and associated to the topological features. In [37], the skeleton is also used as a shape descriptor and the match is done by the shapes hierarchical skeleton graph. In [43], Reeb graph has been adopted for the partial shape retrieval. Partial similarity between two shapes is evaluated by computing a variant of their maximum common sub-graph. We refer [4] for further discussion.

2.2.3 View Based Methods

The basic idea behinds this type of methods is that two 3D shapes are same if they look similar from all places. One approach proposed by Cyr and Kimia [15] composes the descriptor of a 3D shape by clustering different views of the shape and representing each cluster using one view. Then a shock graph matching is adopted to do the shape comparison. Another direct approach is to project the 3D model to 2D planes at different views [13]. Zernike moments and Fourier descriptors of the projected silhouettes are adopted as representations for retrieval. And this method achieves a better retrieval performance than the 3D harmonics method in [16], though it costs more processing time. The similar idea can be found in [21], where the aspect graph representation was employed as a shape descriptor.

2.3 Some State-of-The-Art 3D Shape Retrieval Methods

In this section, we present some state-of-the-art 3D shape retrieval methods in this research area. While they use different kinds of features and descriptors for the 3D shape retrieval, all related new methods follow the very traditional 3D shape retrieval pipeline using the BoF approach and its extensions due to BoF's relatively good performance.

2.3.1 Local Visual Patch for 3D Shape Retrieval

The main novelty of this approach [40] is that it proposed a new descriptor based on an indexed collection of closed curves in R^3 on the 3D shape surface. It starts by extracting feature points using Tierny's method [43]. The extraction is said to be invariant to isometric deformation and robust to the local surface noises. Then, the proposed closed curves are extracted around each feature points. These closed curves are designed to capture the local geometry of the 3D surface patch using the geometries of the associated curves. Such descriptor also has the advantage of being invariant to different transformations, though it is sensitive topological ones because that the features extracted are geodesic related. Finally, BoF framework is applied to get the results.

In [40], the method is compared with two other state-of-the-art methods:

- Spherical Harmonic Descriptor (SHD) [19]. In this method, the feature vector is composed by extracting the spherical harmonic coefficients from the spherical functions, which the maximal distance from center of mass is given.

- Light Field Descriptor (LFD) [13]. This is a very famous method, which is a view-based method for 3D shape retrieval. Ten silhouettes from different viewpoints are used to describe the shape. By analyzing these views, the 3D shape can be recognized.

We refer the figure (shown here as Figure 2.4) from [40] and show that this novel result achieves a better performance than the other two on TOSCA dataset [5].

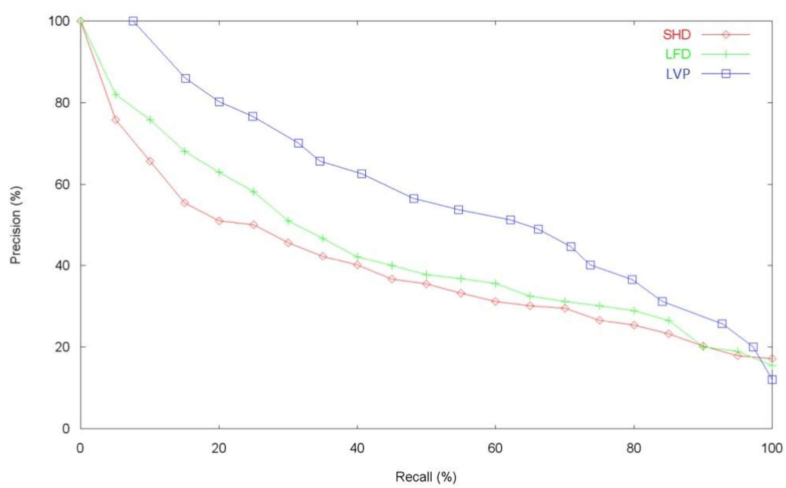


Figure 2.4 Precision Recall plots for LVP, SHD and LFD (Referred from [40])

2.3.2 Spatially Enhanced Bags of Words for 3D Shape Retrieval

As mentioned before, one of the disadvantages of the original BoF is that it loses the spatial information. In this approach [23], though still following the whole pipeline using the spin image as the low level features, a weak spatial constraint is applied to improve the descriptive capability compared to the original BoF also a new similarity metric is designed that accounts not only for appearance but also for geometry [23]. Figure 2.5, which is referred from [23], shows the whole concept and pipeline of the method.

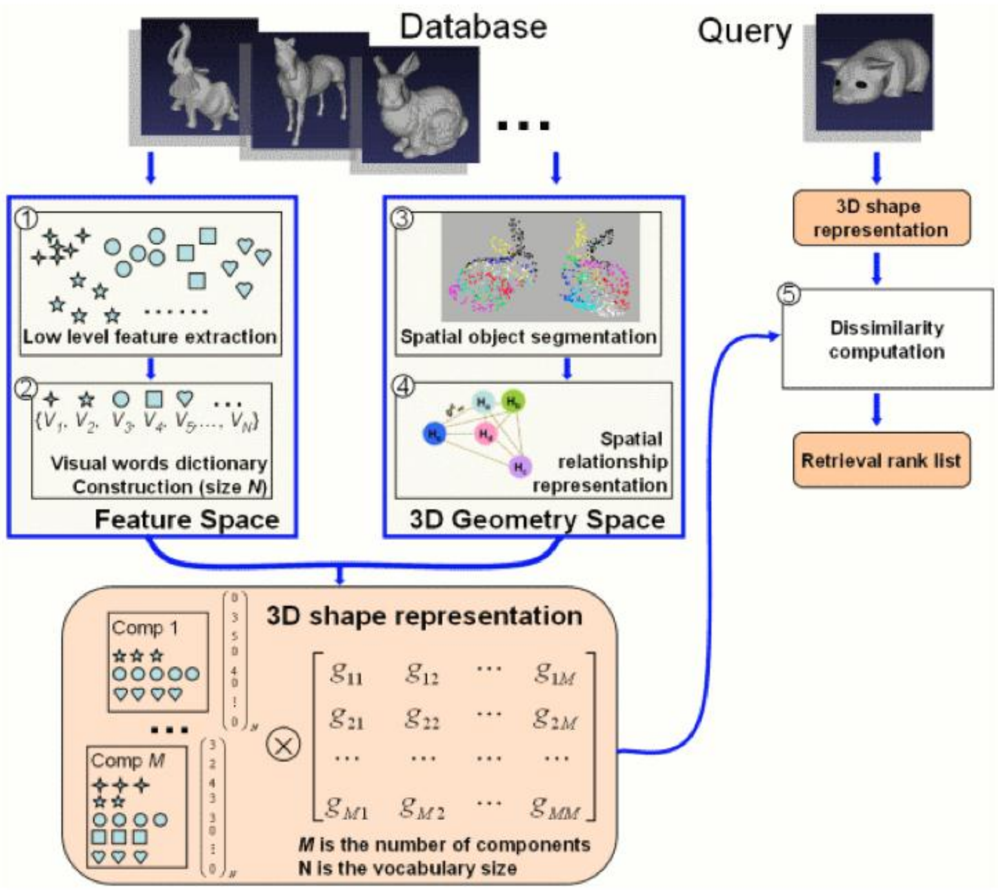


Figure 2.5 Spatially Enhanced Bags-of-Words method for 3D shape retrieval (Referred from [23])

The experiment was conducted on the Princeton Shape Benchmark (PSK) [53], the method was compared with the original BoF and results show that the nearest neighbor of the retrieval statistics is improved though the whole performance improvement is not large due to the method is sensitive to the K-means segmentation method. And the author claimed that further researches will focus on using a more sophisticated segmentation method.

2.3.3 Shape DNA

"Shape DNA" is considered a much more advanced method recently. The key point of the method [33] is that it uses the Laplace-Beltrami spectra as the shape descriptor. The author noted that the spectrum is deformation invariant due to the intrinsic nature of the Laplace-Beltrami operator. Using the spectrum as a descriptor is a very novel approach in the field of the 3D shape retrieval. Since the spectrum is isometry invariant to the respective object, this kind of descriptor is also independent of the spatial position. In addition, the eigenvalues can be normalized so that scaling factors for the geometric object can be obtained easily [33]. For the application of 3D shape retrieval combined with the BoF framework, only a comparison of their spectra is need. And experiments on TOSCA dataset [5] show that the retrieval accuracy can reach 91.11% on average at the best case and 68.60% on average at the worst case.

2.4 Heat Kernel Signature & "Shape Google"

In section 2.4, we emphasize on the HKS and the "Shape Google" related methods because we also completely adopt the HKS for the future descriptor and HK matrices for graph representation in our study.

2.4.1 Heat Kernel Signature

Recently, there has been increased interest in the use of diffusion geometry for shape analysis. A lot of works suggest using the diffusion geometry method for feature detection and description. People use that because it has a lot of advantages over others. HK is one of them. It arises from the heat equation,

$$(\Delta x + \frac{\partial}{\partial t})u = 0 \tag{2.4}$$

which governs the conduction of heat on the surface. Here, Δx means the Laplace-Beltrami operator. The fundamental solution $K_t(x, y)$ is called HK. By theorem of the spectral decomposition, it can also me wirrten as,

$$K_t(x, y) \approx \sum_{l=0}^{\infty} e^{-\lambda_l t} \phi_l(x) \phi_l(y) \tag{2.5}$$

where λ_l are the eigenvalues of the Laplace-Beltrami operator and ϕ_l are the corresponding eigenfunctions [9]. In addition, the HK, which is intrinsically relevant to the partial differential equation and random walks, is invariant to isometric deformations and resilient to noise.

Based on this, Jian Sun, etc [38] proposed HKS in 2009, which is the diagonal of the HK at different scales,

$$K_t(x, x) \approx \sum_{l=0}^{\infty} e^{-\lambda_l t} \phi^2{}_l(x) \tag{2.6}$$

The HKS $K_t(x, x) : R^+ \times M \rightarrow R$ is also defined as the amount of heat transferred a point *x* to itself at time *t*. As showed in [38] the HKS is a concise and informative representation, which has the properties of intrinsic, multi-scale, etc, and preserves all of the information about the intrinsic geometry of the shape. Thus, the HKS can have almost all types of invariance and can be applied on any representation. For feature finding, the local maxima of the HKS for a large time parameter can be used as a stable feature, which is also proposed by Sun, etc in [38].

Later on, there are some other special heat kernel signatures, which are designed for dealing with special situations, for example, *Scale-invariant Heat Kernel Signature* (SI-HKS) [10], *Volumetric Heat Kernel Signatures* (VHKS) [32], etc.

2.4.2 Shape Google Revisit

The recently "Shape Google" method [11] proposed by Bronstein, shows much better performance than others for the non-rigid 3D shape retrieval. Although it also basically follows the original BoF paradigm, in "Shape Google", HKS values on different time scale are used to compose the feature descriptor. And this kind of descriptor is intrinsic also captures the local geometric differential information around point x on a shape at different scales. For each point x on the shape, its HKS descriptor is an n-dimensional vector

$$p(x) = c(x)(K_{t1}(x,x),\ldots\ldots K_{tn}(x,x)) \tag{2.7}$$

where $c(x)$ is chosen in such a way that $\|p(x)\|_2 = 1$, $K_{tn}(x,x)$ is the HKS on time tn [11].

The geometric vocabulary $W = \{W_1, \cdots, W_v\}$, where W_i represents the geometric word in the vocabulary and *V* indicates the size of the vocabulary, is composed by clustering the HKS vectors in descriptors space by vector quantization using the K-means algorithm. "Shape Google" uses a dense sampling approach, which means HKS descriptors are computed on each point of the 3D shape. Therefore, for each point *x*, the geometric word distribution is given by

$$\Theta(x) = [\theta_1(x), \cdots\cdots, \theta_V(x)]^T \tag{2.8}$$

$$\theta_i(x) = c(x)e^{-\frac{\|K(x)-W_i\|^2}{2\sigma^2}} \tag{2.9}$$

where $c(x)$ is a constant that lets $\|\Theta(x)\|_1 = 1$ and σ is the parameter. The BoF descriptor, which is the histogram, is computed by Eq. 2.1 as mentioned before. For the ease of comparison, the "Shape Google" largely compresses the geometry information and loses the spatial information by only counting the frequencies of geometric words on the entire shape. The whole process of the BoF is time consuming because all points on the shape are involved in the computation. For a shape with N points, the time complexity of the algorithm used in “Shape Google” is $O(ND)$ where $O(D)$ is the time complexity for computing one HKS descriptor. In addition, we can also refer that SS-BoF, which is also introduced in “Shape Google”, is extremely slow and has a time complexity of $O(N^2D)$.

Chapter 3 New Approach: Bag of Feature Graphs

In the research area of 3D shape retrieval, recently most of the methods follows the BoF framework, which is originally from the 2D image retrieval. We have already discussed the disadvantages that the original BoF framework has. In this chapter, we present our new approach for the non-rigid 3D shape retrieval, which is called *Bag of Feature Graphs* (BoFG). We also adopt the recent HKS as the feature descriptors because we have seen the advantages of the HKS used in the non-rigid 3D shape retrieval area. Here, we emphasize the advantages of our approach again, which has already described in Chapter 1 in advance:

- It is fast to compute by reducing the number of points on the shape involved in the representation. Therefore it is much more concise.

- Sspatial information is considered by constructing diffusion matrix only among feature points, which is a key factor that influences the retrieval performance.

- It is representative because of the nature of the feature points that they are salient points on the 3D shape and contain important information.

- Graphs have different dominating features associated with corresponding geometric words respectively, which improves the accuracy of shape retrieval greatly.

- It supports the partial shape retrieval by first applying the graph matching technique. And then select the corresponding rows and columns from the BoFG matrices.

3.1 Bag of Feature Graphs

In order to reduce the complexity of the "Shape Google", one intuitive thinking is to reduce the number involved in the computation for representing the 3D shape. It naturally leads to first select the feature points on the shape because the feature points usually preserve more information of the shape than other non-salient points. Because the HKS has the property of multi-scale, HKS features have the information from points in small scales to the entire shape in large scales. However, the original BoF approach is based on the distribution of large number of points that represent the shape. Therefore, one concern is that the reduced number of feature points may not be sufficient to faithfully represent the shape. More geometry information must be added to the shape representation.

To resolve the concern, we devise the graphs information on the detected features. Such graphs encode spatial relations in among features. This leads to the concept of BoFG.

3.1.1 Feature Classification

Given a shape *X* with a set of features *F*, we only need to compute the geometric word distributions $\Theta(x)$, where $x \in F$. In this way, a lot of computation is saved. These features are quantized by assigning $\theta_i(x)$ portion of similarity to w_i in the distribution of the feature point x. The distribution $\Theta(x)$ is computed according to Eq. 2.9, where σ is set to be a quarter of the average distance of words in the vocabulary. Since in our BoFG method, the classification serves for eliminating the unlikely points in a comparison, the vocabulary is not necessary to be very large.

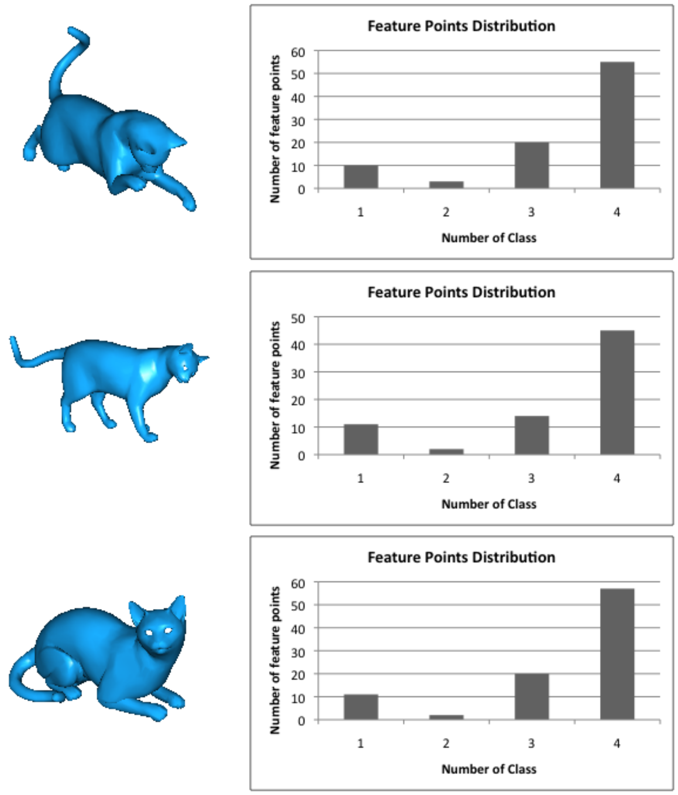


Figure 3.1 Feature points' distribution of cat shapes

Figure 3.1 shows the classification based on a hard quantization of feature points on the cat shapes with a vocabulary of four geometric words. And Figure 3.2 shows the same thing of the centaur shapes. We note that the feature points distributions of the cat or centaur shapes are very same. Compare Figure 3.1 with Figure 3.2, we note the difference of distribution between centaur shapes and cat shapes. It implies that such a scheme based on the distribution of the feature points can further enhance the power of discriminating the different shapes.

In order to avoid the misclassification in a hard quantization, we further develop a fuzzy classification, which is based on a soft quantization also reduces the ambiguities in graph comparison and eliminates the unlikely points in a comparison. Figure 3.3 shows a fuzzy classification also with four geometric words. The color of the feature points are coded by a linear combination of the colors of the geometric words, according to its similarities to the geometric words.

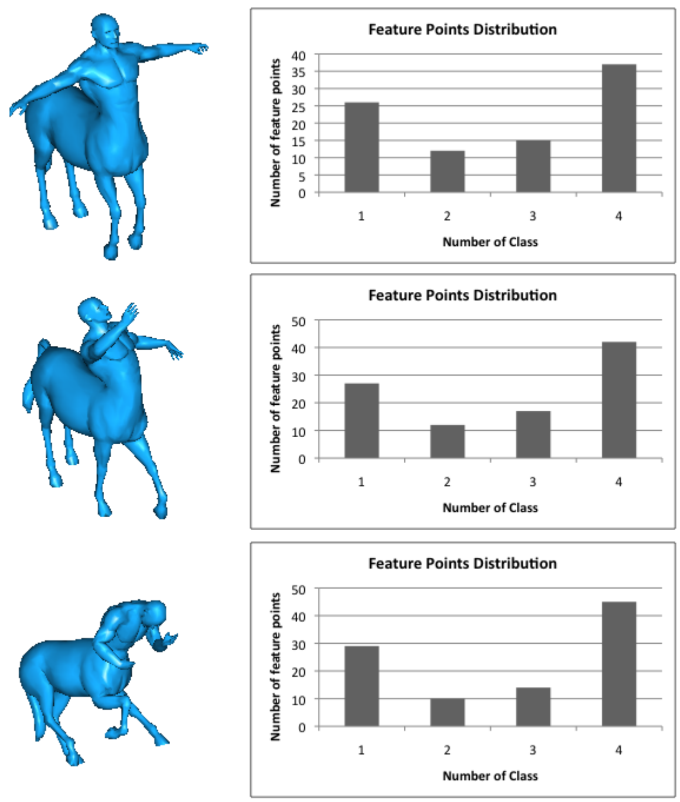


Figure 3.2 Feature points' distribution of centaur shapes

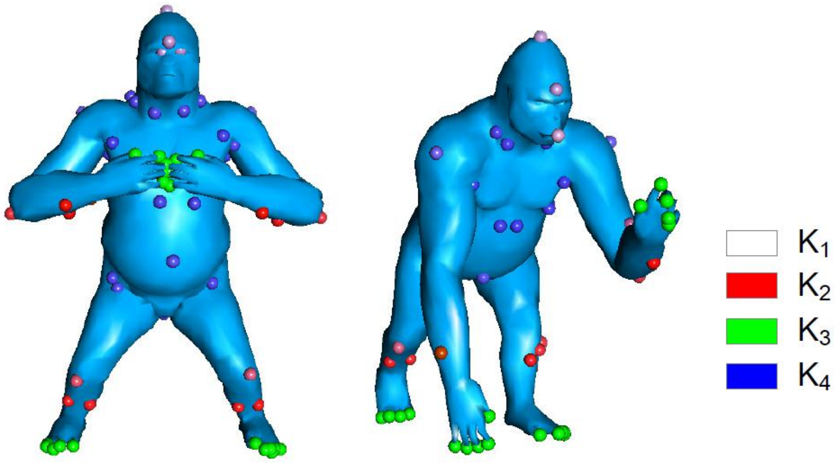


Figure 3.3 The fuzzy classification of feature points. The feature points are colored by a linear combination of word-colors based on its similarities to the geometric words respectively.

3.1.2 Weighted Heat Kernel Matrix

The structures of graphs are captured by the *Weighted Heat Kernel* (WHK) matrices. For each geometric word W_i, we compose a matrix $G_i(x, y)$, where $(x, y) \in F \times F$. The matrix is given by

$$G_i(x, y) = \theta_i(x)\theta_i(y)K_t(x, y) \tag{3.1}$$

, which is the HK between *x* and *y* weighted by the similarities to the geometric word W_i. Finally, the set of matrix $G(X) = \{G_i, \cdots\cdots, G_V\}$ is considered as a representation of shape *X*. Figure 3.4 shows the proposed BoFG representation of a shape. It contains a series of $|F| \times |F|$ matrices that preserve the spatial information of features assigned to different geometric word categories and contain the geometric information of features in multi-scale, which characterizes the shape faithfully. Note that the complexity for the BoFG representation is $O(|F|^2 D)$, which is much faster than the original "Shape Google" method, since the number of feature points on the shape is much less than the number of all points on the shape.

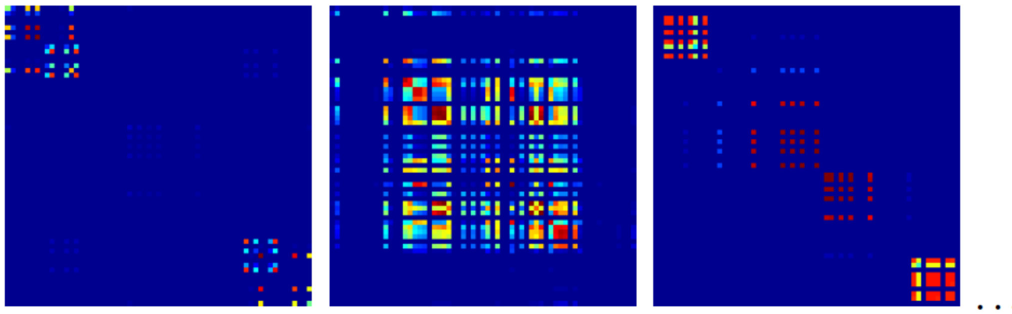


Figure 3.4 The BoFG representation of a shape.

3.1.3 Shape Retrieval & Similarity Comparison

For the goal of 3D shape retrieval, we need to build a concise BoFG descriptor of all the shapes in the database offline in advance. Given a query shape, the related shapes are retrieved by the *Approximate Nearest Neighbor* (ANN) search online. We use the significant eigenvalues of the BoFG matrices to compose the BoFG descriptor. Since each G_i is a real symmetric matrix, its eigenvalues are all real and eigenvectors are perpendicular to each other. For each G_i, the six largest eigenvalues $S_i(X)$ are chosen to build the BoFG descriptor $[S_1(X), \cdots \cdots, S_V(X)]^T$. The *Multi-Dimensional Scaling* (MDS) [6] is used to reduce the dimension. Figure 3.5 shows some samples of non-rigid shapes and their BoFG shapes. We note that the cat models have a very similar BoFG descriptor. However, the horse model has a different one instead.

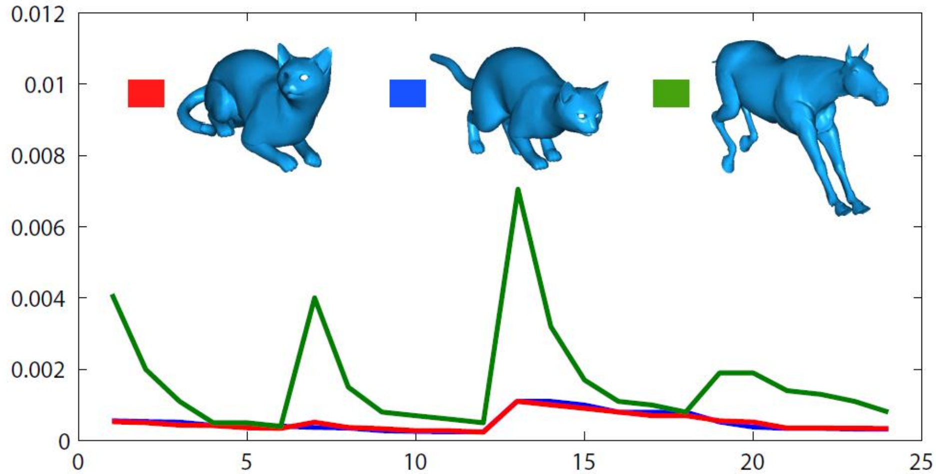


Figure 3.5 Non-rigid shapes and their BoFG descriptors.

In order to retrieve the shapes from the 3D shape database given a query shape, the similarity distance must be calculated and the shapes with the minimum distances are retrieved. We define the distance between shape X and shape Y in our method as the following,

$$d(X, Y) = \sum_{i=1}^{V} \left\| S_i(X) - S_i(Y) \right\|_2 \quad (3.2)$$

Need to note that such distance above is based on only one scale HK and we can easily further extends it to a multi-scale version with different values of time in order to improve the accuracy.

3.1.4 Partial Shape Retrieval

The original BoF paradigm, such as the “Shape Google” method, cannot support the partial shape retrieval that the given query shape is only part of the complete shape. The reason is that the original BoF is based on the histogram of geometric word frequencies. However, a partial shape always has different geometric word frequencies compared with complete one.

One of our BoFG paradigm's advantages is that it immediately supports the partial shape retrieval. In BoFG, the partial shape is represented by a feature sub-graph of the one for a complete shape. We first apply the sub-graph matching and use the graph matching method described in [18]. Then we can select the corresponding rows and columns from BoFG matrices to measure the distance among shapes. Figure 3.6 shows the example for sub-graph matching for partial shape retrieval.

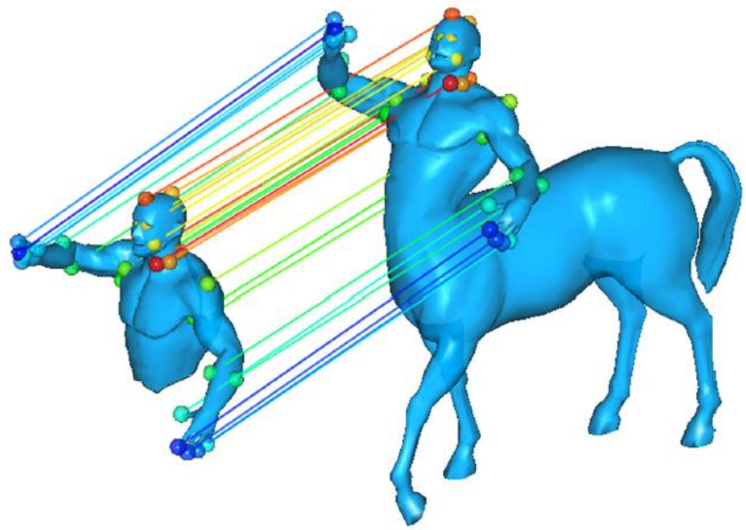


Figure 3.6 Sub-graph matching for partial shape retrieval

3.2 Implementation

3.2.1 Numerical Computation of HK

For the numerical computation of HK, the key is that the discrete eigenfunctions and eigenvalues of the Laplace-Beltrami operator should be used instead of continuous ones. We can use different kinds of approaches to simulate the Laplace-Beltrami operator on the discrete domain. For the 3D shapes with point clouds representation, the approximation can be done by [3]. Because our 3D data are all triangular mesh, though using the *Finite Element Methods*

(FEM) [34] to approximate the Laplace-Beltrami operator can lead to a more accurate result, we use the recently most popular cotangent weight approach [31]. For any function f defined on the mesh,

$$(\Delta_X f)_i = \frac{1}{a_i}\sum_j w_{ij}(f_i - f_j) \tag{3.3}$$

where $w_{ij} = \cot\alpha_{ij} + \cot\beta_{ij}$ if j is the one ring neighborhood of vertex i, otherwise w_{ij} equals zero. We know from [31] that this kind of discretization preserves lots of properties that the continuous Laplace-Beltrami operator holds. Eq. 3.3 can also be written as

$$\Delta_X f = A^{-1}Lf \tag{3.4}$$

where $A = diag(a_i)$ and $L = diag(\sum_{l \neq i} w_{il}) - (w_{ij})$. According to Eq. 3.4, the discrete HK can be calculate by solving the *generalized eigendecomposition* problem [20], which is

$$K_t(x,y) \approx \sum_{l=0}^{k} e^{-\lambda_l t}\phi_l(x)\phi_l(y) \tag{3.5}$$

In practice, we only compute the finite number of eigenvalues and eigenvectors of the Laplace-Beltrami operator, which is 200 in our implementation. Figure 3.7 visualizes the heat diffusion on a 3D shape using the numerical computation of HK. We can see that the source of the heat is put one finger of the hand, which is red. And, the heat is transferred to the whole shape.

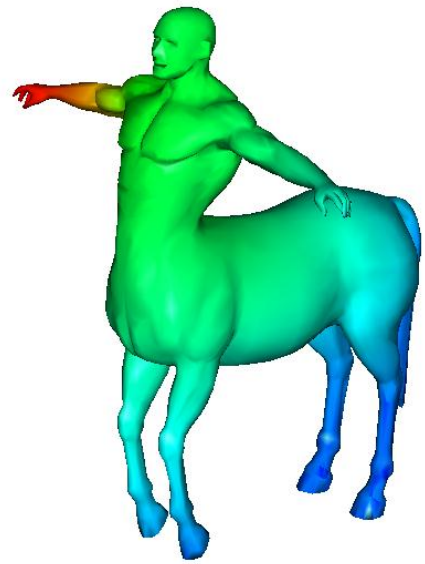


Figure 3.7 Visualization of heat diffusion (numerical computation, *t=40*) on a 3D shape, red means a higher heat value and blue means a lower heat value. The heat is put on the one finger and transferred to the whole area.

3.2.2 Resolution-Adaptive HKS

The original HK has the scale problem, which can be solved by eigenvalue [38], Fourier transform [10], or area [44]. The area scheme eliminates the scale factor by letting the average vertex area be unit one. However, the unit area will cover areas of different sizes when the resolution changes, which will cause a shift and a scale for the original HK as shown in Figure 3.8. In order to compute the HKS values on scaled and resolution changed shapes, we design a resolution-adaptive area scheme, which is invariant to both scale and resolution changes. Assume the surface has α times points, according the area scheme [44], it shifts logarithm of t by $\log \alpha$ and scales the HK by $1/\alpha$. Therefore, as the right plot shown in Figure 3.8, the proposed resolution-adaptive HKS fixes the problem.

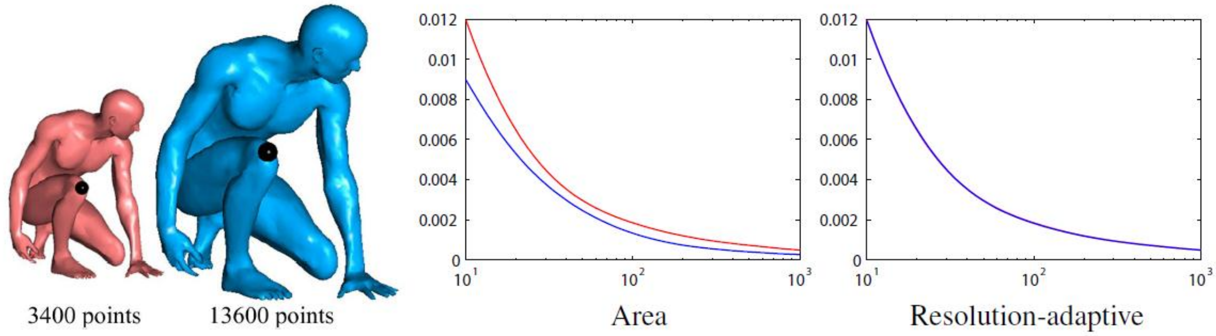


Figure 3.8 HKS curves computed on scaled and resolution changed shapes

Since the HKS exponentially decreases along the time *t*, the portions at small values of *t* dominate the HKS descriptor. Therefore, it is reasonable to balance the HKS at different time scale. In our approach, we adopt the domain-independent normalization described in [18].

$$K_t'(x) = \log(4\pi t K_t(x)) \tag{3.6}$$

From previous work, we know that the HKS is $(4\pi t)^{-1}$ at *t* in R^2 and is equal everywhere. Therefore, such normalization shows how the surface is bended from the plane and has a better power of discrimination. Positive values mean slower heat diffusion than a plane and negative values imply faster heat diffusion. Figure 3.9 shows the proposed normalized HKS curves at different time scales. In our HKS descriptors, we sample the time *t* logarithmically in a range from 5 to 640.

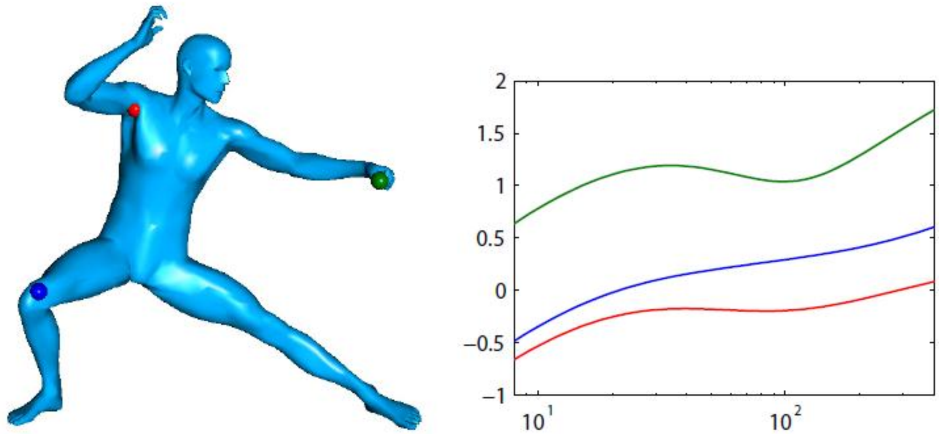


Figure 3.9 Normalized HKS curves

Chapter 4 Experiments

In this chapter, we are going to present and discuss the experimental results of the proposed method in the previous chapter, also mainly compare our results with “Google Shape” methods, which are considered to be the best in the field of non-rigid 3D shape retrieval recently. Our experiment shows that our method can also achieve a quite competitive result and work relativly faser under the same conditions of others. The whole simulation was implemented mainly using C++ and some MATLAB work on 3D models with Wavefront File Format (.obj) [54]. The whole framework includes an off-line feature detection and extraction, an off-line construction of geometric dictionary, calculating the descriptor for each model in the database which is also off-line. Finally, an on-line 3D shape retrieval process is executed in this approach. The following part of this chapter will describe the experiment in details and evaluate the method.

4.1 Dataset

In order to evaluate our method, we performed our shape retrieval experiments on the database that composed of shapes from TOSCA proposed by Bronstein et al. The database contains a total of 148 objects, including 9 cats, 11 dogs, 3 wolves, 17 horses, 15 lions, 21 gorillas, 1 shark, 6 centaurs, 6 seahorses, 24 female figures, and two different male figures, containing 15 and 20 poses. Each object contains approximately 3000 vertices [5]. Figure 4.1 shows some shape examples from the dataset.

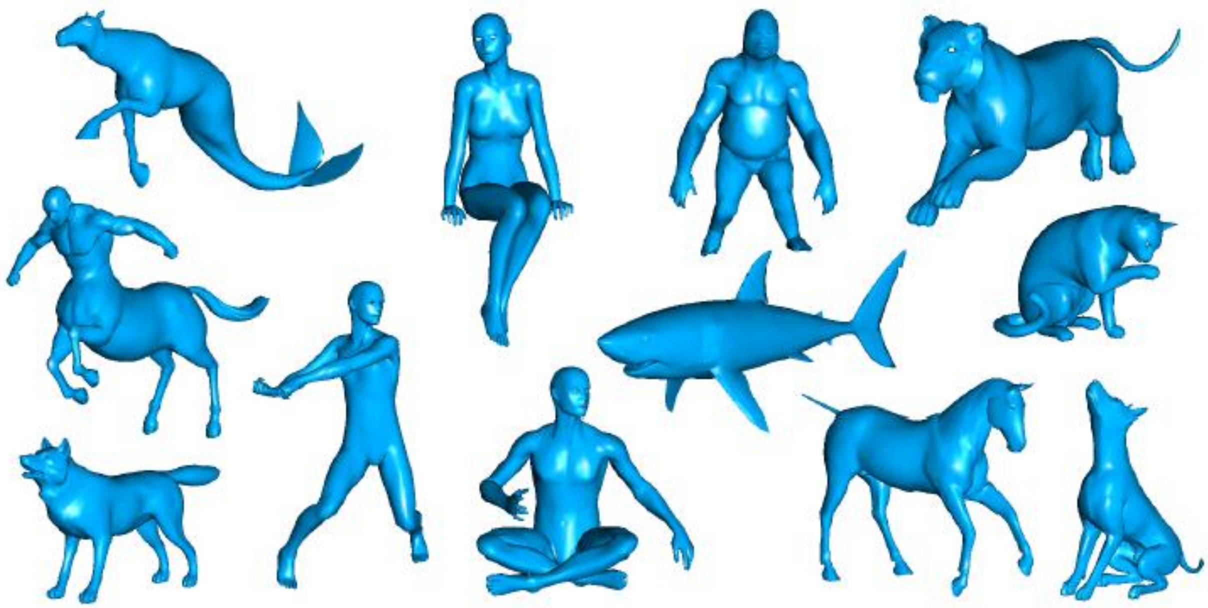


Figure 4.1 Sample shapes used in the TOSCA dataset

4.2 Time Performance

For a single query, the first step is to generate the shape descriptor to initiate the retrieval. In this experiment, we compare the time performance of computing the descriptor for a single shape based on the BoF, SS-BoF and BoFG respectively (Not including the time of computing the discrete Laplacian matrix). And the comparison time of two shapes' descriptor is negligible. Two models with 3k vertices and 30k vertices are used as the inputs. Figure 4.2 shows one 3D shape with 30k vertices that we used as the input for the experiment. The experiment was run on the laptop with a 2.4GHz Intel Pentium Core2 Duo CPU and 2GB of RAM. Figure 4.3 depicts time performance of three shape descriptors for the input models respectively. The feature numbers involved in BoFG for two shapes are 42 and 98. Since the SS-BoF runs extremely slow, we also use the same number of features in the implementation, denoted as FSS-BoF. And from the figure, we note that by reducing the number of points involved in the computation of BoFG, the time performance is significantly improved and better improvement can be achieved when the ratio of the points to features is greater.

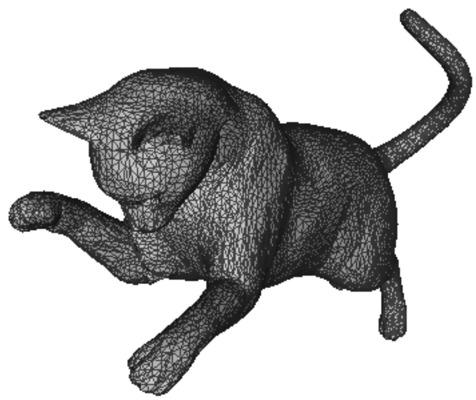


Figure 4.2 The input 3D shape with 30k vertices

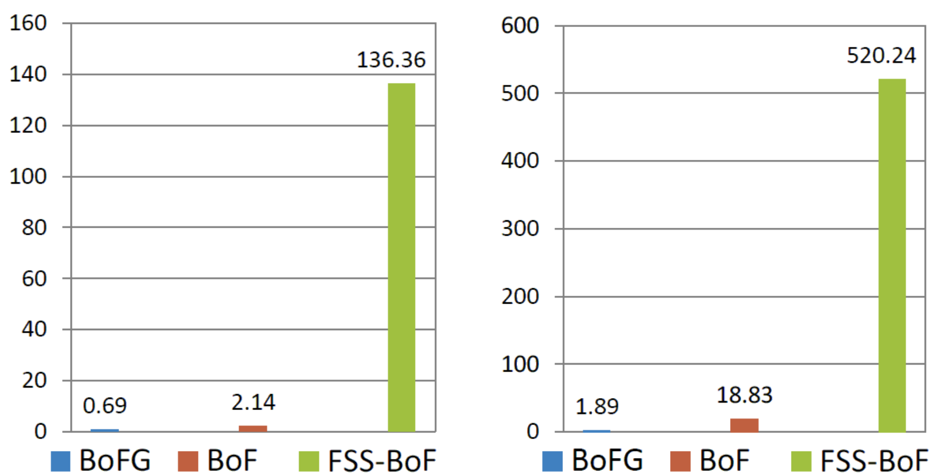


Figure 4.3 Time performance of three descriptors on a shape with 3K vertices (Left) and 30K vertices (Right)

However, we need to point out that though we may further adjust the number of feature points to get a faster speed, too less feature points might have a negative impact on the retrieval accuracy of BoFG.

4.3 Queries and Results

The query set is consisted by selecting one shape of each kind from the dataset. In the experiment, each time, we choose one shape from the query set as an input to the system and retrieve the most relevant shapes from the database. In the experiment, we use the cotangent weight approximation of the Laplace-Beltrami operator to discretely compute the heat kernels. For the feature detection, we define the feature points as the local extreme on different time scales. And compose the HKS feature descriptor in a time scale range of [5, 640] logarithmically. In order to test our method under a challenging condition, we apply some transformations to the original query shapes. Besides the null (no transformation), we apply scale change, resolution change (resampling with different number of points), hole (topological change and missing information), and partial shape (cut a part from a complete shape). Figure 4.4 to Figure 4.8 visualize the selected results in each transformation group of experiments. The shapes in the leftmost column are used as the query. And we showcase its top-six retrieved shapes in right part of these figures.

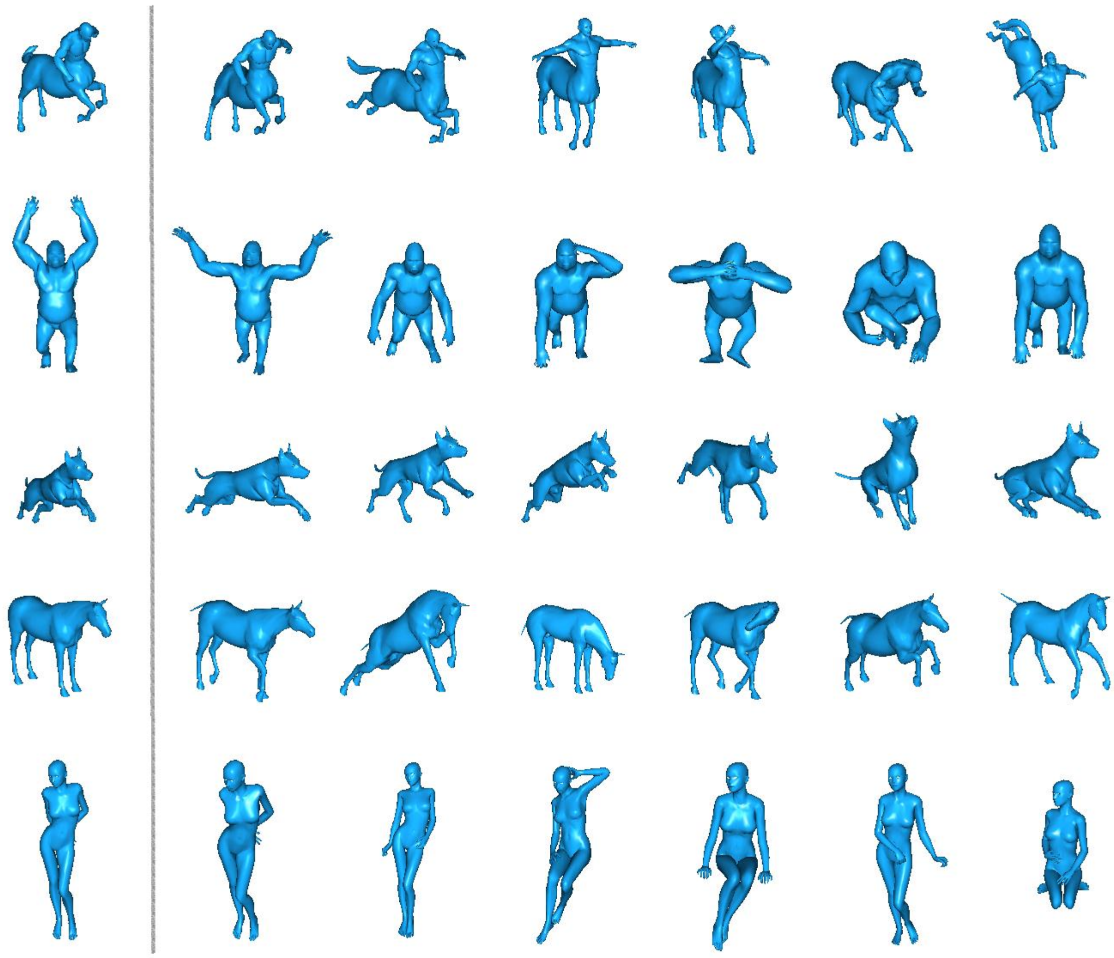


Figure 4.4 Selected results in the null group of experiments

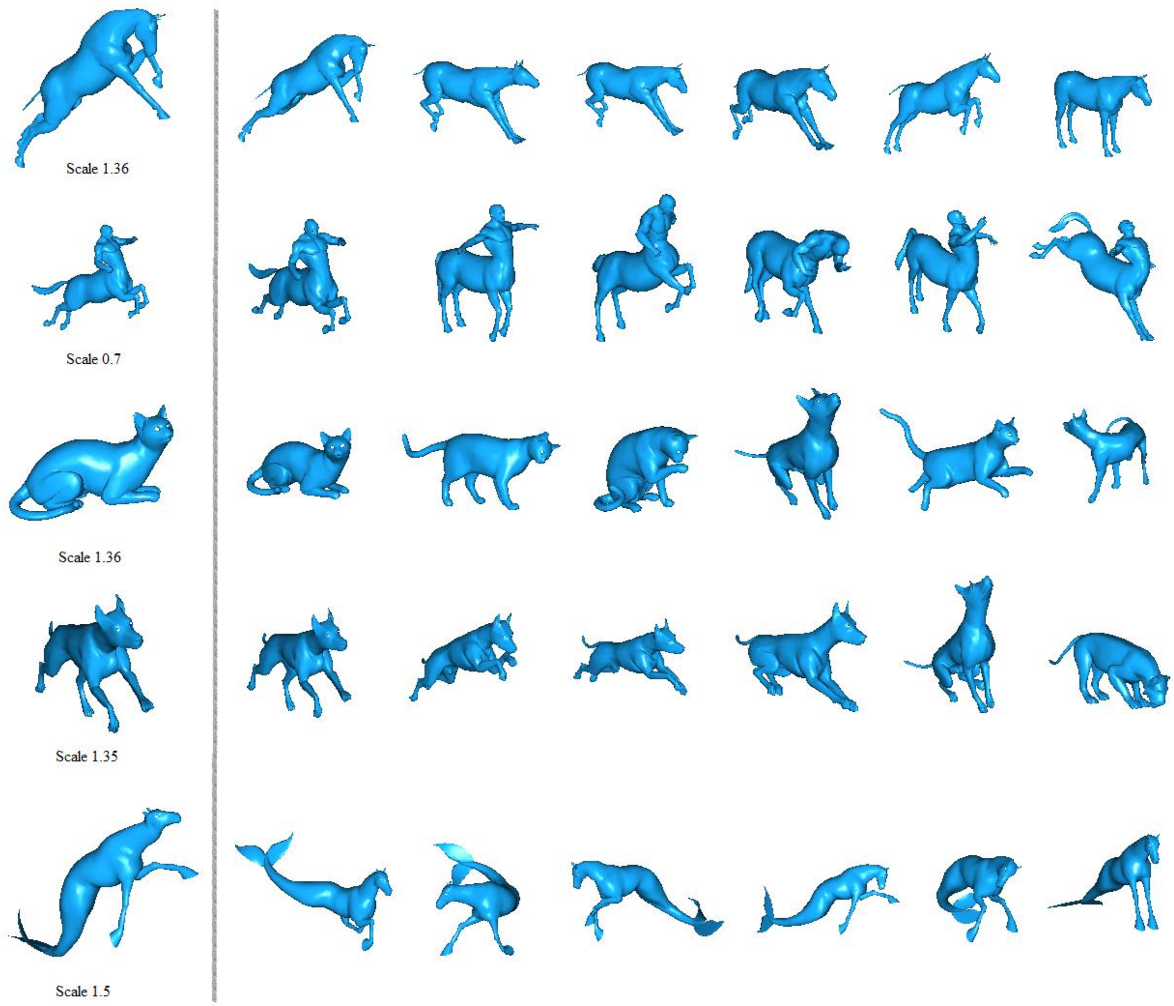


Figure 4.5 Selected results in the scale change group of experiments

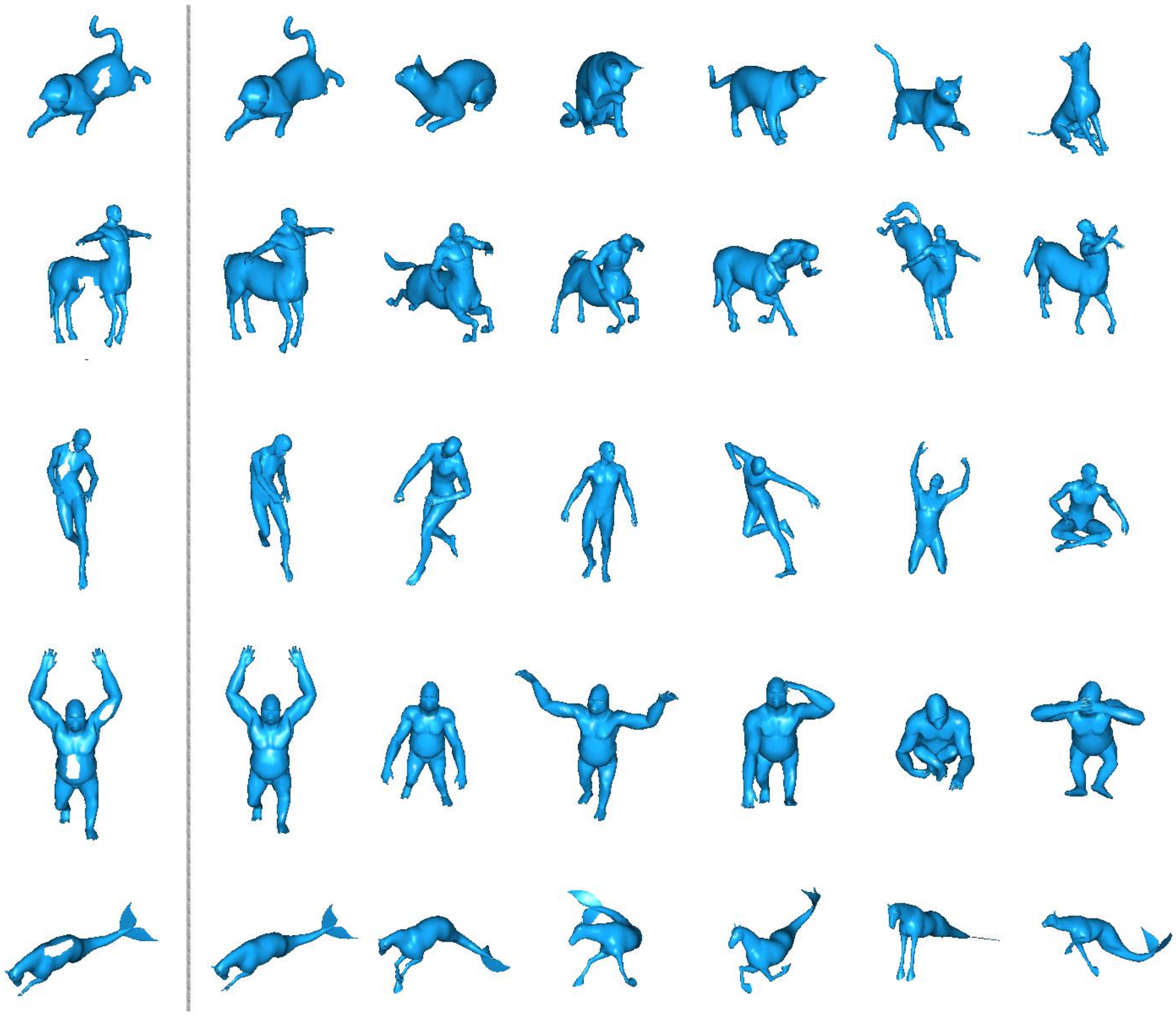


Figure 4.6 Selected results in the hole group of experiments

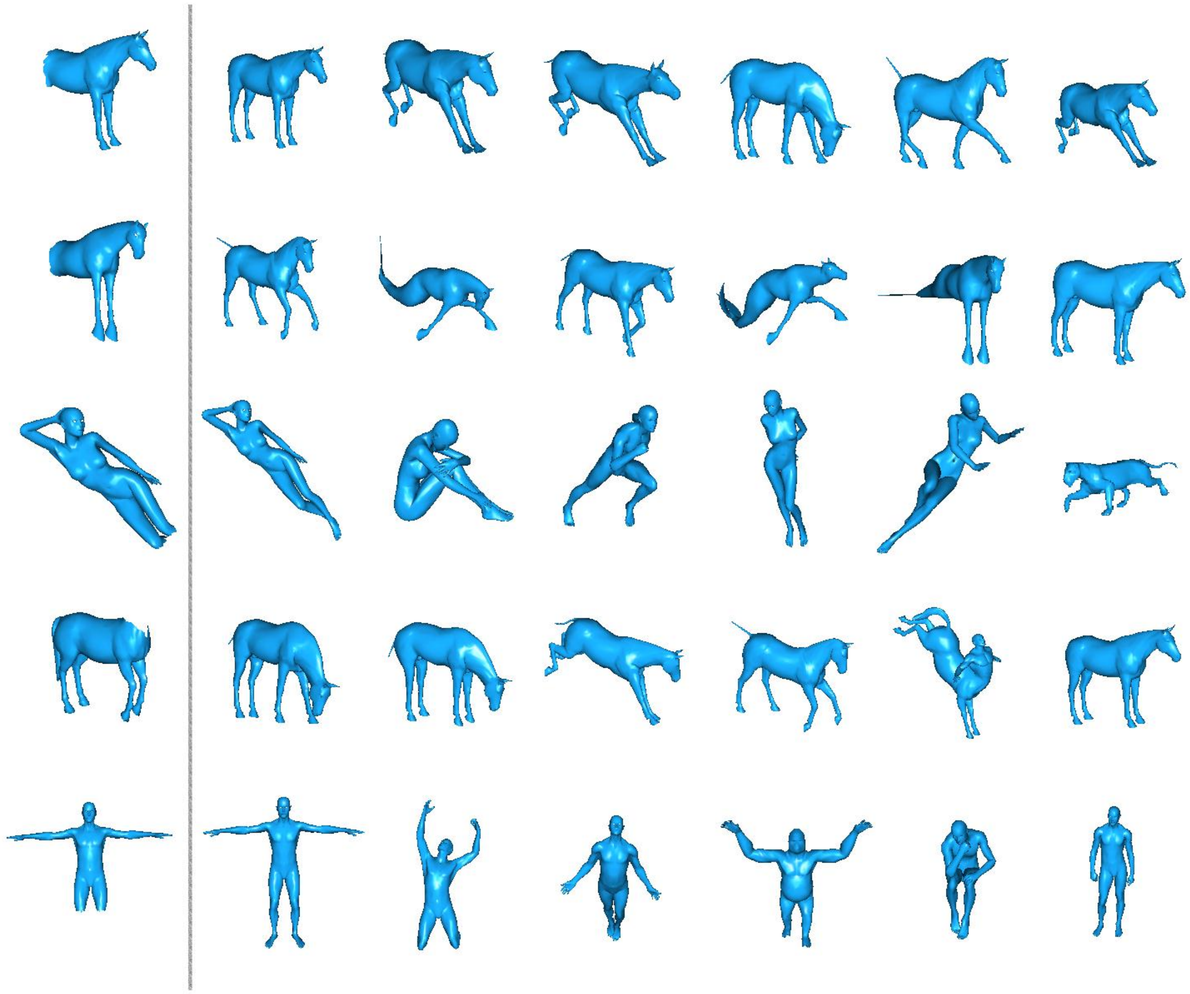


Figure 4.7 Selected results in the partial group of experiments

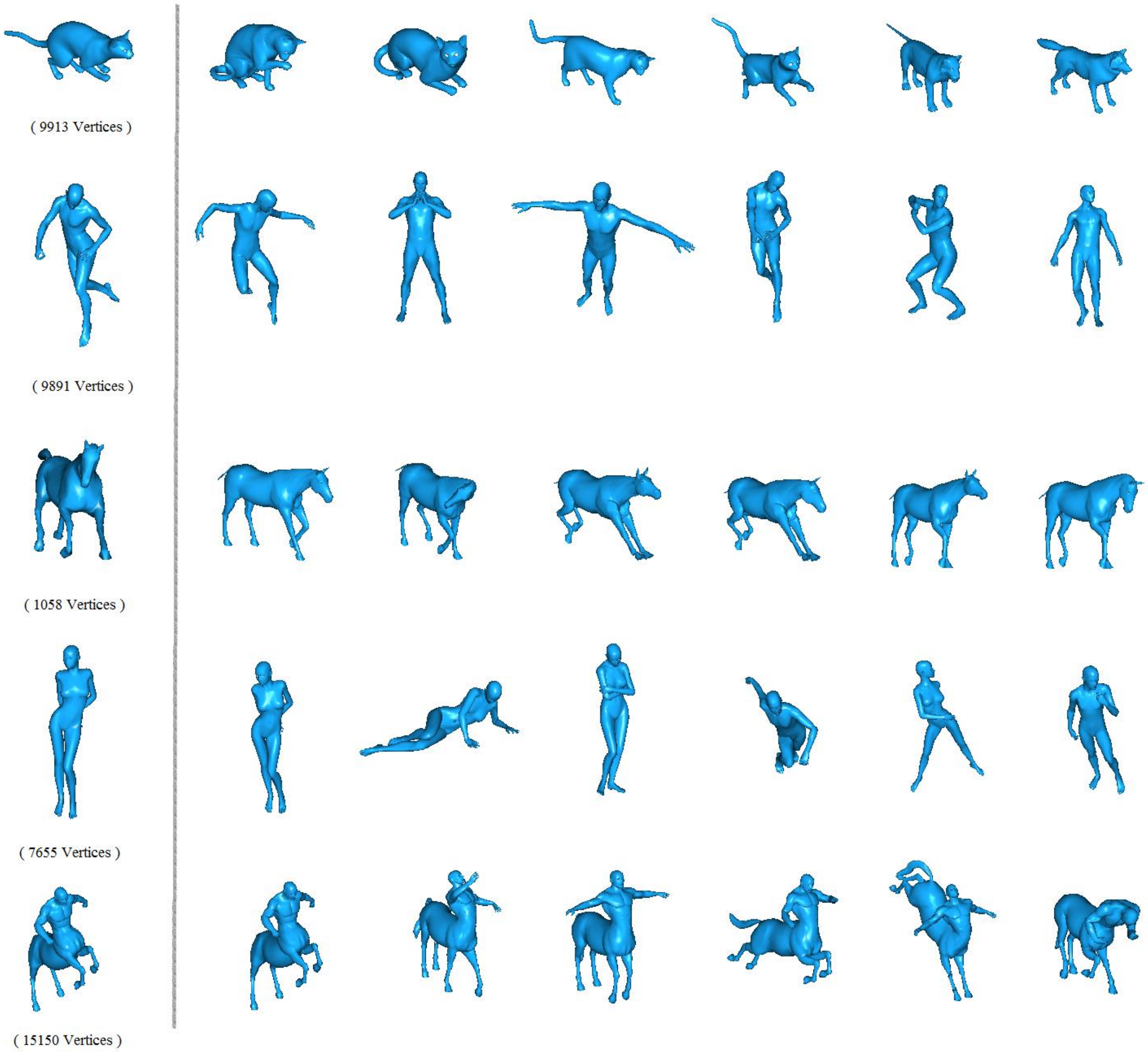


Figure 4.8 Selected results in the resolution change group of experiments

Especially, in Figure 4.6, we can see that our approach is insensitive to large holes or data missing on the shape. And in Figure 4.7, we note that shapes of horses and centaurs are retrieved from the query shape cut from the centaur shape, which are considered as correct retrieval results. Overall, these figures demonstrate a relatively meaningful query results.

4.4 Comparison & Evaluation

For the task of comparison, we use a more quantitative approach, which is called the *precision-recall* (PR) curve. The PR curve is usually used to measure various kinds of retrieval performance. It plots the tradeoff between *precision* (ratio of the number of relevant shapes retrieved and the total number of shapes retrieved) and *recall* (ratio of the number of relevant shapes retrieved and the total number of existing relevant shapes that could be ideally retrieved) [8].

To further illustrate our method, we compose an experiment by comparing our method to the "Shape Google" related methods including the BoF, SS-BoF and SI-HKS paradigm. As before, we use FSS-BoF instead of the SS-BoF in the implementation. The size of "geometric vocabulary" for the "Shape Google" related methods is set to be 48, while our vocabulary size is 4.

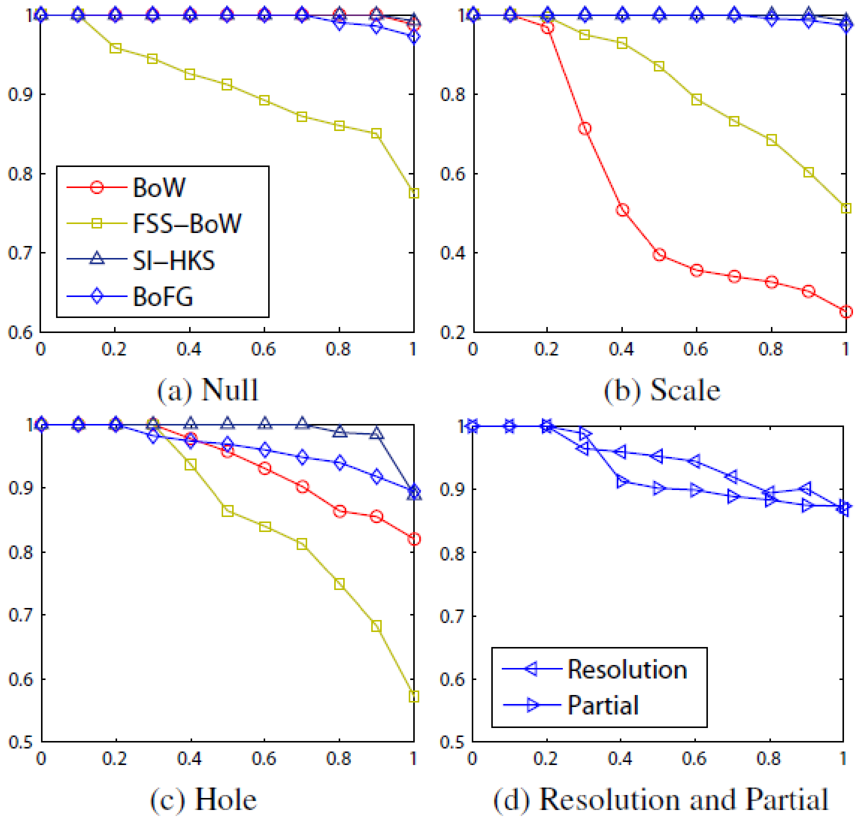


Figure 4.9 PR Curves of evaluated methods

Figure 4.9 shows the PR curves of our method compared with other “Shape Google” related methods. We can note that our BoFG has competitive results with the state-of-the-art “Shape Google” methods in categories of null (a), scale change (b) and hole (c). And in the category of resolution change and partial (d), we only show the results of our method since the “Shape Google” related methods are not designed to solve the problem. In the implementation of the partial shape retrieval, we first filter the complete shapes in the database that have less than 25 matches with the partial shape, which means they are not similar to the partial shape. Then, construct the BoFG descriptors for both the partial shape and complete shapes.

To further evaluate our methods’ performance, we also adopt a single measure of performance, which is called the *Mean Average Precision* (mAP) and defined in [8]. Average precision is computed as the area below the PR curve for each query, and the mAP is the average of AP over all queries [10]. Table 4.1 shows the mAPs of all the evaluated methods and our method still maintains a high mAPs throughout the all the challenging experiments.

Method	Null	Scale	Resolution	Hole	Partial
BoW	1.00	0.52	N/A	0.93	N/A
FSS-BoW	0.90	0.80	N/A	0.84	N/A
SI-HKS	1.00	0.98	N/A	0.98	N/A
BoFG	0.99	0.99	0.94	0.96	0.92

Table 4.1 Mean average precisions of evaluated methods

Chapter 5 Conclusion and Future Work

5.1 Conclusion

In this thesis, we present a novel paradigm, which is called BoFG for non-rigid 3D shape representations. Such BoFG provides a concise and faithful shape representation for the non-rigid 3D shape comparison and retrieval. Because the properties of the feature based approach, the retrieval performance greatly relies on the feature detection. Therefore, BoFG is designed to couple with stable and multi-scale method, such as the original HKS for the feature extraction on the non-rigid 3D shape. We adopt the HKS and further develop our resolution-adaptive HKS to solve the issue of scale change and resolution change of the 3D shapes. Also the BoFG can be expediently combined with other advanced techniques, such as graph matching, MDS and graph spectrum. By applying the graph matching to the BoFG, our method immediately supports the partial shape retrieval that previous state-of-the-art “Shape Google” related methods are not supposed to address.

Besides that, we conducted our non-rigid 3D shape retrieval experiments on th TOSCA database and presented the experiment results in Chapter 4. The results clearly demonstrate the effectiveness and efficiency of our approach under the challenging conditions, which show that the BoFG approach is robust to non-rigid isometry deformations, scale changes, resolution changes and insensitive to large holes.

For the partial shape retrieval, we need to point out that the sub-graph matching must be taken on-line. The graph matching process has to be done first between the partial query shape and all the complete shapes in the database. Such process usually takes only less than one second. And, the BoFG feature descriptor can still be pre-computed off-line in advance.

5.2 Future Work

For the immediate future work, there are something need to be discussed and may help further improve the results of our approach.

- **Sub-matrix comparison.** For the partial shape retrieval, a possible direction is to design a faithful sub-matrix comparison method that can directly compare to sub-matrix without building the correspondences between shapes. Therefore, we can eliminate the need for the graph matching process before we generate the feature descriptors and can further improve the efficiency of partial shape retrieval.

- **Patch based description on different levels.** Patch based description has been proved to be better than only using the point based description in 3D shape retrieval. Because it would contain more information than another. In our method, the construction of the BoFG heat kernel matrix might be considered as a patch based description for a shape in global and in different categories. We may improve that by constructing such kinds of patch description in more detail levels such as local BoFG descriptors. The spatial relation or diffusion distance among feature points can be further grouped together in a local area. Therefore, information on different levels for a shape can be captured not only the golable one.

- **Further utilize the spatial information among features.** Compared to the 2D images retrieval area, where spatial information is fully considered, for example, the methods described in [12], [49], such spatial information is much limited for 3D shape retrieval. This is due to the fact that segmentation of 3D shape is much harder than that of 2D images. "Spatially Enhanced Bags of Words for 3D Shape Retrieval"[23] takes the spatial information into account by clustering the points on the shape into groups based on the relative coordinates and generate a graph based connection. Though the result of the method did not greatly improve the 3D shape retrieval performance, similar ideas can be considered to improve our method.

- **Advanced clustering and classifying algorithms.** One of our proposed methods' deficient is that it might be sensitive to the classification of the feature points. In the method, we use the basic K-means clustering and then classifying the feature points softly. Though it is usually used in other methods, K-means might be not good enough for our approach. We can try to apply some other advanced clustering and classifying algorithms on our approach, for example, hierarchical K-means [1], etc.

Finally, one thing needs to be further improved is that our BoFG approach is completely based on the feature points one shape. Under some undesirable conditions, for example shapes with lots of holes or data missing on the area of feature points, our BoFG paradigm might not work well as usual while the original "Shape Google" methods might have better results because these methods are based on a more statistical approach instead of heavily depending on the feature points. This issue also needs to be considered and solved in future work.

References

[1] K. Arai, A. R. Barakbah. Hierarchical K-means: an algorithm for centroids initialization for k-means. Reports of the Faculty of Science and Engineering, Saga University, Japan, Vol. 36(1), 2007.

[2] H. Bay, T. Tuytelaars, L. VanGool. Surf: Speeded up robust features. Lecture Notes in Computer Science (LNCS), Vol. 3951(2006), pp. 404-417, 2006.

[3] M. Belkin, J. Sun, Y. Wang. Constructing Laplace operator from point clouds in R^d . In Proc. ACM-SIAM Symposium on Discrete Algorithms, pp.1031-1040, 2009.

[4] S. Biasotti, S. Marini, M. Mortara, G. Patané. An overview of properties and efficacy of topological skeletons in shape modeling. In Proc. Shape Modeling International (SMI), pp. 245-254, 2003.

[5] A. M. Bronstein, M. M. Bronstein, R. Kimmel. Efficient computation of isometry-invariant distances between surfaces. SIAM J. Scientific Computing, Vol. 28(5), pp. 1812-1836, 2006.

[6] A. M. Bronstein, M. M. Bronstein, R. Kimmel. Generalized multidimensional scaling: A framework for isometry-invariant partial surface matching. In Proc. Nat'l Academy of Science (PNAS), pp. 1168-1172, 2006.

[7] A. M. Bronstein, M. M. Bronstein, M. Ovsjanikov, and L. J. Guibas, Shape Google: a computer vision approach to invariant shape retrieval, In Proc. Workshop on Non-rigid Shape Analysis and Deformable Image Alignment (NORDIA), 2009.

[8] A. M. Bronstein, M. M. Bronstein, U. Castellani, B. Falcidieno, A. Fusiello, A. Godil, L. J. Guibas, I. Kokkinos, Z. Lian, M. Ovsjanikov, G. Patané, M. Spagnuolo, R. Toldo. Shrec 2010: robust large-scale shape retrieval benchmark. In Proc. Eurographics Workshop on 3D Object Retrieval (3DOR), 2010.

[9] A. M. Bronstein, M. M. Bronstein, M. Ovsjanikov. 3D features, surface descriptors, and object descriptors. Sep 2010.

[10] M. M. Bronstein, I. Kokkinos. Scale-invariant heat kernel signatures for non-rigid shape recognition, In Proc. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1704-1711, 2010.

[11] A. M. Bronstein, M. M. Bronstein, M. Ovsjanikov, L. J. Guibas. Shape Google: geometric words and expressions for invariant shape retrieval. ACM Transactions on Graphics (TOG), Vol. 30(1), pp. 1-20, 2011.

[12] Y. Cao, C. Wang, Z. Li, L. Zhang, L. Zhang. Spatial-bag-of-features. In Proc. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3352-3359, 2010.

[13] D. Y. Chen, X. P. Tian, Y. T. Shen, M. Ouhyoung. On visual similarity based 3D model retrieval. Eurographics Computer Graphics Forum, Vol. 22(3), pp. 223-232, 2003.

[14] J. Corney, H. Rea, D. Clark, J. Pritchard, M. Breaks, R. Macleod. Coarse filters for shape matching. IEEE Computer Graphics and Applications, Vol. 22(3), pp. 65-74, 2002.

[15] C. M. Cyr, B. Kimia. 3D object recognition using shape similarity-based aspect graph. In Proc. IEEE International Conference on Computer Vision (ICCV), Vol. 1, pp. 254-261, 2001.

[16] T. Funkhouser, P. Min, M. Kazhdan, J. Chen, A. Halderman, D. Dobkin, D. Jacobs. A search engine for 3D models. ACM Transactions on Graphics (TOG), Vol. 22(1), pp. 83-105, 2003.

[17] A. Godil, A. Wagan. Salient local 3D features for 3D shape retrieval. In Proc. The International Society for Optics and Photonics (SPIE), Vol. 7864, 78640S, 2011.

[18] T. Hou, H. Qin. Robust Dense Registration of Partial Non-rigid Shapes. IEEE Transactions on Visualization and Computer Graphics (TVCG), to appear.

[19] M. Kazhdan, T. Funkhouser, S. Rusinkiewicz. Rotation invariant spherical harmonic representation of 3D shape descriptors. In Proc. Eurographics/ACM SIGGRAPH Symposium on Geometry Processing (SGP), 2003.

[20] B. Lévy. Laplace-Beltrami eigenfunctions towards an algorithm that "understands" geometry. In Proc. IEEE International Conference on Shape Modeling and Applications, pp. 13-13, 2006

[21] S. Lee, S. Yoon, I. D. Yun, D. H. Kim, K. M. Lee, S.U. Lee. A new 3-d model retrieval system based on aspect-transition descriptor. In Proc. European Conference on Computer Vision (ECCV), pp. 543-554, 2006.

[22] Y. Liu, H. Zha, H. Qin. Shape topics: A compact representation and new algorithm for 3D partial shape retrieval. In Proc. International Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2025-2032, 2006.

[23] X. L. Li, A. Godil, A. Wagan. Spatially enhanced bags of words for 3D shape retrieval, Advances in Visual Computing, Lecture Notes in Computer Science (LNCS), Vol. 5358, pp. 349-358, 2008.

[24] Z. Lian, A. Godil, X. Sun. Visual similarity based 3D shape retrieval using bag-of-features. In Proc. Shape Modeling International (SMI), pp.25-36, 2010.

[25] D. Lowe. Distinctive image features from scale-invariant key point. International Journal of Computer Vision (IJCV), Vol. 60(2), pp. 91-110, 2004.

[26] J. Matas, O. Chum, M. Urban, T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. Image and Vision Computing (IVC), Vol. 22(10), pp. 761-767, 2004.

[27] P. Min, J. A. Halderman, M. Kazhdan, T. A. Funkhouser. Early experiences with a 3D model search engine. In Proc. International conference on 3D Web technology (Web3D), pp. 7-18, 2003.

[28] P. Min, M. Kazhdan, T. Funkhouser. A comparison of text and shape matching for retrieval of online 3D models. In Proc. European Conference on Digital Libraries (ECDL), pp. 209-220, 2004.

[29] S. Naik, B. Prabha. 3D Shape Retrieval. Master Thesis, Dec 2006.

[30] R. Osada, T. Funkhouser, B. Chazelle, D. Dobkin. Shape distributions. ACM Transactions on Graphics (TOG), Vol. 21(4), pp. 807-832, 2002.

[31] U. Pinkall and K. Polthier. Computing discrete minimal surfaces and their conjugates. Experimental Mathematics, Vol. 2(1), pp. 15-36, 1993.

[32] D. Raviv, M. M. Bronstein, A. M. Bronstein, and R. Kimmel. Volumetric heat kernel signatures, In Proc. ACM Multimedia Workshop on 3D Object Retrieval (3DOR), 2010.

[33] M. Reuter, F. Wolter, N. Peinecke. Laplace-spectra as fingerprints for shape matching. In Proc. ACM Symposium on Solid and Physical Modeling (SPM), pp. 101-106, 2005.

[34] M. Reuter, S. Biasotti, D. Giorgi, G. Patané, M. Spagneuolo. Discrete Laplace-Beltrami operators for shape analysis and segmentation. Computers & Graphics (CG), Vol. 33(3), pp. 381-390, 2009.

[35] C. Schmid, Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories, In Proc. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2169-2178, 2006.

[36] J. Sivic, A. Zisserman. Video google: A text retrieval approach to object matching in videos. In Proc. IEEE International Conference on Computer Vision (ICCV), Vol. 2, pp.1470-1477, 2003.

[37] H. Sundar, D. Silver, N. Gagvani, S. Dickenson. Skeleton based shape matching and retrieval. In Proc. Shape Modeling International (SMI) pp. 130-139, 2003.

[38] J. Sun, M. Ovsjanikov, L. Guibas. A concise and provably informative multi-scale signature based on heat diffusion. In Proc. Symposium on Geometry Processing (SGP), 2009.

[39] M. T. Suziki. A search engine for polygonal models to support development of 3D e-learning applications. In Proc. International World Wide Web Conference Poster, pp. 182-183, 2001.

[40] H. Tabia, M. Daoudi, J. P. Vandeborre, O. Colot. Local visual patch for 3D shape retrieval. In Proc. Eurographics Workshop on 3D Object Retrieval (3DOR), 2010.

[41] G. K. L. Tam, R. W. H. Lau. Deformable model retrieval based on topological and geometric signatures. IEEE Transactions on Visualization and Computer Graphics (TVCG), Vol. 13(3), pp. 470-482, 2007.

[42] J. W. H. Tangelder, R. C. Veltkamp. A survey of content based 3D shape retrieval methods. In Proc. Shape Modeling International (SMI), pp. 145-156, 2004.

[43] J. Tierny, J. P. Vandeborre, M. Daoudi. Partial 3D shape retrieval by reeb pattern unfolding. Eurographics Computer Graphics Forum, Vol. 28(1), pp. 41-55, 2009.

[44] A. Vaxman, M. Chen, C. Gotsman. A multi-resolution approach to heat kernels on discrete surfaces. ACM Transactions on Graphics (TOG), Vol. 29(4), 2010.

[45] R. C. Veltkamp, M. Hagedoorn. State-of-the-art in shape matching. In Principles of Visual Information Retrieval, pp. 87-119, 2001.

[46] M. Yu, I. Atmosukarto, W. K. Leow, Z. Huang, R. Xu. 3D model retrieval with morphing-based geometric and topological feature maps. In Proc. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), pp. 656-661, 2003.

[47] C. Zhang, T. Chen. Efficient feature extraction for 2D/3D objects in mesh representation. In Proc. International Conference on Image Processing (CIP), Vol. 3, pp. 935-938, 2001.

[48] C. Zhang, T. Chen. Indexing and retrieval of 3D models aided by active learning. In Proc. ACM international conference on Multimedia, pp. 615-616, 2001.

[49] E. Zhang, M. Mayo. Improving bag-of-words model with spatial information. The 25th International Conference of Image and Vision Computing New Zealand (IVCNZ), pp. 1-6, 2010.

[50] Object File Format (.off): http://cmp.felk.cvut.cz/~pajdla/Doc/3D.formats.html-off.

[51] Ogden IV system, http://www.nime.ac.jp/˜motofumi/Ogden/.

[52] Polygon File Format (.ply): http://en.wikipedia.org/wiki/PLY_%28file_format%29.

[53] The Princeton Shape Benchmark, In Proc. Shape Modeling International (SMI), pp. 167-178, 2004.

[54] Wavefront File Format (.obj): http://en.wikipedia.org/wiki/Wavefront_.obj_file.

[55] 3D model search engine, http://shape.cs.princetion.edu.

[56] 3D model retrieval system, http://3d.csie.ntu.edu.tw/˜dynamic/.

Appendix

A 3D Shapes Basics

A.1 3D Shape Representation

A 3D shape is usually a subset of Euclidean space, $\Omega \in R^3$. In some applications, they use a volumetric representation of 3D shapes, which divides a solid model into thousands of small voxels. In this way, one shape can be represented by a 3D matrix. However, most of the researches focus on the surface representation of one 3D shape (the boundary $\partial\Omega$ of the shape) because we can only access the boundary of one object in most cases.

For the surface of one 3D shape, there are mainly two representations: *Point Cloud* and *Mesh*. If X donates the object, the *Point Cloud* represents the surface by a set of discretely sampled points $\{x1, x2, ..., xn\} \subset X$. Only geometric co-ordinates are provided in this representation. If information of connectivity consisting of a set of edges $\{x_i, x_j\} \in E$ and faces $\{x_i, x_j, x_k\} \in F$ is also added, this kind of representation is called *Mesh*. Figure A.1 shows an airplane 3D shape with representation of *Point Cloud* and *Mesh* respectively [29]. In this thesis, we do our analysis and experiments on the *Mesh* surface representation of the 3D shapes; however, the proposed approach can also apply to *Point Cloud* representation.

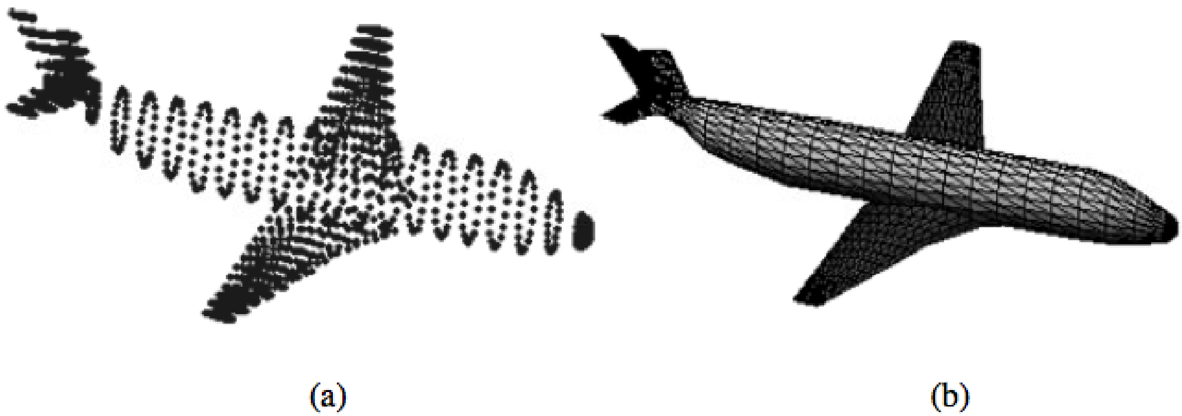


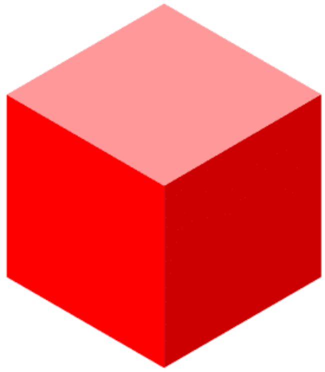
Figure A.1 3D shape of an airplane. (a) Cloud Point (b) Mesh (Referred from [29])

A.2 3D File Format

There are lots of well-known file formats for representing a 3D model mesh structure, such as .ply [52], .off [50], .obj [54], etc. They are all very simple and similar files describing the information of the geometry and connectivity of a 3D shape and can easily be transformed from one to another. In our experiments, we all use the Wavefront File Format (.obj) [54] as our data of 3D shapes while other file formats can also be applied to our method.

Figure A.2 is a sample of a cube model of the Wavefront File Format (.obj) file:

- V x y z defines a vertex at position (x, y, z).
- F i_1 i_2 i_3 defines a triangle with vertices indexed by i_1, i_2, and i_3 where the vertex identifiers correspond to the order of previously defined vertices in the file.



```
# cube.obj
v -0.25 -0.25 0.25
v 0.25 -0.25 0.25
v -0.25 0.25 0.25
v 0.25 0.25 0.25
v -0.25 0.25 -0.25
v 0.25 0.25 -0.25
v -0.25 -0.25 -0.25
v 0.25 -0.25 -0.25
f 1 2 4
f 1 4 3
f 3 4 6
f 3 6 5
f 5 6 8
f 5 8 7
f 7 8 2
f 7 2 1
```

Figure A.2 Sample .obj file format