Multimed Tools Appl
DOI 10.1007/s11042-007-0181-0

A survey of content based 3D shape retrieval methods

Johan W. H. Tangelder · Remco C. Veltkamp

Abstract Recent developments in techniques for modeling, digitizing and visualizing 3D shapes has led to an explosion in the number of available 3D models on the Internet and in domain-specific databases. This has led to the development of 3D shape retrieval systems that, given a query object, retrieve similar 3D objects. For visualization, 3D shapes are often represented as a surface, in particular polygonal meshes, for example in VRML format. Often these models contain holes, intersecting polygons, are not manifold, and do not enclose a volume unambiguously. On the contrary, 3D volume models, such as solid models produced by CAD systems, or voxels models, enclose a volume properly. This paper surveys the literature on methods for content based 3D retrieval, taking into account the applicability to surface models as well as to volume models. The methods are evaluated with respect to several requirements of content based 3D shape retrieval, such as: (1) shape representation requirements, (2) properties of dissimilarity measures, (3) efficiency, (4) discrimination abilities, (5) ability to perform partial matching, (6) robustness, and (7) necessity of pose normalization. Finally, the advantages and limitations of the several approaches in content based 3D shape retrieval are discussed.

Keywords 3D shape retrieval · 3D shape matching · Dissimilarity measures · Meshes · Volume models

J. W. H. Tangelder
Prime Vision, Delft, The Netherlands
e-mail: h.tangelder@primevision.com

R. C. Veltkamp (✉)
Department of Information & Computing Sciences, Utrecht University,
Utrecht, The Netherlands
e-mail: remco.veltkamp@cs.uu.nl

1 Introduction

The advancement of modeling, digitizing and visualizing techniques for 3D shapes has led to an increasing amount of 3D models, both on the Internet and in domain-specific databases. This has led to the development of the first experimental search engines for 3D shapes, such as the Ephesus search engine at the National Research Council of Canada [98], the 3D model search engine at Princeton University [2, 85], the 3D model retrieval system at the National Taiwan University [1, 112], the Ogden IV system at the National Institute of Multimedia Education, Japan [92, 119], the 3D search system at the Informatics and Telematics Institute, Greece [4], the 3D model similarity search engine at the University of Konstanz [3, 126], and the 3D retrieval engine at Utrecht University [5, 120].

Laser scanning has been applied to obtain archives recording cultural heritage like the David of Michelangelo [71], and the Minerva of Arezzo [40]. Furthermore, archives containing domain-specific shape models are now accessible on the Internet. Examples are the National Design Repository, an online repository of CAD models [88, 104], ShapeSifter, a retrieval system for CAD databases [111, 118] and the Protein Data Bank, an online archive of structural data of biological macromolecules [14, 100].

Very recently, the first benchmarks for mesh models have been made available by Princeton University [2, 113], the University of Konstanz [3], Utrecht University [5, 120], and Purdue University [39, 55]. The Princeton Shape Benchmark database contains 1,814 models downloaded from the web, subdivided into a training set and a test set, containing 907 models each, classified into 90 and 92 classes respectively. The database at the University of Konstanz provides a test set containing 473 models classified into 55 classes, and 1,366 unclassified models. The database at Utrecht University consists of 512 models classified into six categories. The Purdue Engineering Shape Benchmark contains 1,391 models classified into solids of revolution, prisms, and flat-thin wall models.

Unlike text documents, 3D models are not easily retrieved. Attempting to find a 3D model using textual annotation and a conventional text-based search engine would not work in many cases. The annotations added by human beings depend on language, culture, age, sex, and other factors. They may be too limited or ambiguous. In contrast, content based 3D shape retrieval methods, that use shape properties of the 3D models to search for similar models, work better than text based methods [86].

Matching is the process of determining how similar two shapes are. This is often done by computing a distance. A complementary process is indexing. In this paper, indexing is understood to be the process of building a data structure to speed up the search. Note that also "indexing" is also used as a term for the identification of features in models, or multimedia documents in general. Retrieval is the process of searching and delivering the query results. Matching and indexing are often part of the retrieval process.

Recently, a number of researchers have investigated the specific problem of content based 3D shape retrieval. Also, an extensive amount of literature can be found in the related fields of computer vision, object recognition, geometric modelling, computer-aided design and engineering. Survey papers to this literature have been provided by Besl and Jain [15], Loncaric [75], Campbell and Flynn [25] and Mamic and Bennamoun [78]. For an overview of 2D shape matching methods we refer the reader to the paper by Veltkamp [124]. Unfortunately, most 2D methods do not generalize directly to 3D model matching. 3D shapes don't need to be segmented from a background, exhibit no projective deformation, and have no direct boundary parameterization.

The recent Ph.D. dissertations by Min [84], Kazhdan [64] and Vranić [127] focus on 3D model retrieval. They provide an introduction to 3D shape matching and a detailed description of their new shape retrieval and querying methods. Iyer et al. [57] provide an extensive overview of 3D shape searching techniques especially relevant for CAD and engineering. The survey by Cardone et al. [26] primarily focuses on shape similarity methods that are suitable to compare CAD models in the context of product design and manufacturing applications. Shilane et al. [113] compare 12 shape matching methods with respect to processing time, storage requirements and discriminative power using the Princeton Shape Benchmark. Bustos et al. [24] focus on feature based methods. Biasotti et al. [18] present a framework for evaluating 3D shape classification performances.

In contrast, this paper evaluates 3D shape retrieval methods with respect to the following requirements: (1) shape representation requirements, (2) properties of dissimilarity measures, (3) efficiency, (4) discrimination abilities, (5) ability to perform partial matching, (6) robustness, and (7) necessity of pose normalization. In Section 2 we discuss several aspects of 3D shape retrieval. The literature on 3D shape matching methods is discussed in Section 3 and evaluated in Section 4.

2 3D shape retrieval aspects

In this section we discuss several issues related to 3D shape retrieval.

2.1 3D shape retrieval framework

At a conceptual level, a typical 3D shape retrieval framework as illustrated by Fig. 1 consists of a database with an index structure created *offline*, and an *online* query engine. Each 3D model has to be identified with a shape descriptor, providing a compact overall description of the shape. To efficiently search a large collection online, an indexing data structure and searching algorithm should be available. The *online* query engine computes the query descriptor, and models similar to the query model are retrieved by matching descriptors to the query descriptor from the index structure of the database. The similarity between two descriptors is quantified by a dissimilarity measure. Three approaches can be distinguished for providing a query object: (1) browsing to select a new query object from the obtained results (for example the AIM@SHAPE shape repository [6]), (2) a direct query by providing a query descriptor (for example the search engine at the University of Konstanz [3]), (3) query by example by providing an existing 3D model, by creating a 3D shape query from scratch using a 3D tool, or by sketching 2D projections of the 3D model (for example the search engine at Princeton University [2]). Finally, the retrieved models can be visualized.

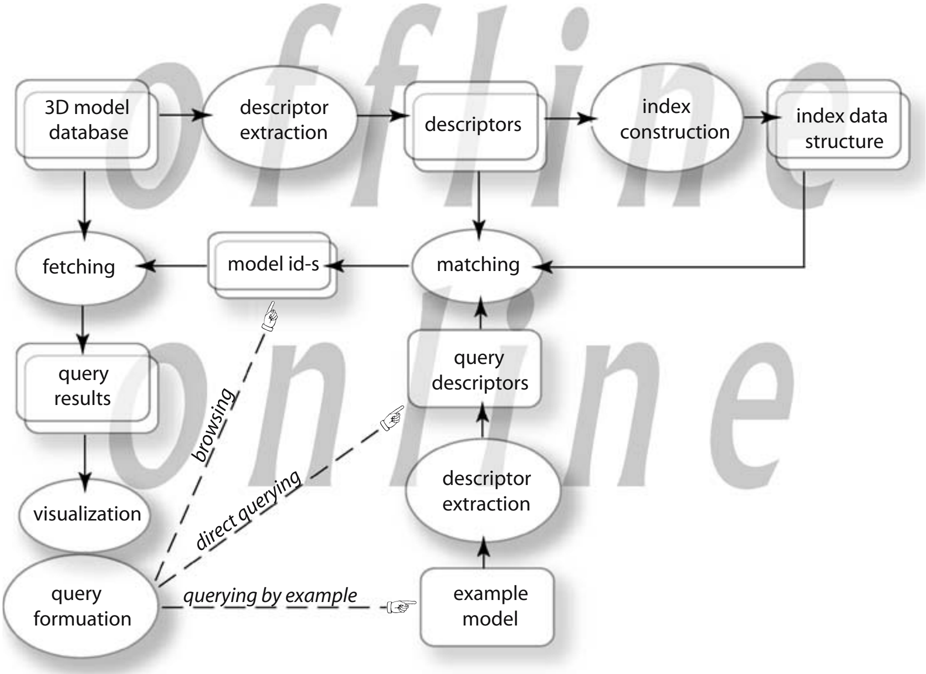

Fig. 1 Conceptual framework for shape retrieval

2.2 Shape representations

An important issue is the type of shape representation(s) that a shape retrieval system accepts. Most of the 3D models found on the World Wide Web are meshes defined in a file format supporting visual appearance. Currently, the most common format used for this purpose is the Virtual Reality Modeling Language (VRML) format [47]. Since these models have been designed for visualization, they often contain only geometry and appearance attributes. Often, they are represented by "polygon soups", consisting of unorganized sets of polygons. Also, in general these models are not "watertight" meshes, i.e. they do not enclose a volume. By contrast, for volume models, retrieval methods that critically depend on a proper volume can be applied.

There are many possible representation techniques for both surface and volume models: implicit surfaces, CSG (constructive solid geometry), BSP (binary space partitioning) trees, octrees, B-rep (boundary representation), free form surfaces, etc. Whenever it is relevant to a particular retrieval method, we will mention them in the next sections.

2.3 Measuring similarity

In order to measure how similar two objects are, it is necessary to compute distances between pairs of descriptors using a dissimilarity measure. Although the term similarity is often used, dissimilarity better corresponds to the notion of distance: small distance means small dissimilarity, and large similarity.

A dissimilarity measure can be formalized by a function defined on pairs of descriptors indicating the degree of their resemblance. Formally speaking, a dissimilarity measure d on a set S is a non-negative valued function $d : S \times S \to \mathbb{R}^+ \cup \{0\}$. Function d may have some or all of the following properties:

1. *Identity*: For all $x \in S$, $d(x, x) = 0$.
2. *Positivity*: For all $x \neq y$ in S, $d(x, y) > 0$.
3. *Symmetry*: For all $x, y \in S$, $d(x, y) = d(y, x)$.
4. *Triangle inequality*:
   For all $x, y, z \in S$, $d(x, z) \leq d(x, y) + d(y, z)$.
5. *Transformation invariance*: For a chosen transformation group G, for all $x, y \in S$, $g \in G$, $d(g(x), g(y)) = d(x, y)$.

The identity property says that a shape is completely similar to itself, while the positivity property claims that different shapes are never completely similar. This property is very strong for a high-level shape descriptor, and is often not satisfied. However, this is not a severe drawback, if the loss of uniqueness depends on negligible details.

Symmetry is not always wanted. Indeed, human perception does not always find that shape x is equally similar to shape y, as y is to x. In particular, a variant x of prototype y, is often found to be more similar to y than vice versa [123].

Dissimilarity measures for partial matching, giving a small distance $d(x, y)$ if a part of x matches a part of y, do not obey the triangle inequality.

Transformation invariance has to be satisfied, if the comparison and the extraction process of shape descriptors have to be independent of the location, orientation and/or scale of the object in its Cartesian coordinate system. If we want that a dissimilarity measure is not affected by any transformation on x, while not changing y, then we may use as alternative formulation for v:

5′. *Transformation invariance*: For a chosen transformation group G, for all $x, y \in S$, $g \in G$, $d(g(x), y) = d(x, y)$.

When properties (1)–(4) hold, the dissimilarity measure is called a *metric*. Other combinations are possible: a pseudo-metric is a dissimilarity measure that obeys (1), (3) and (4), while a semi-metric obeys only (1), (2) and (3). If a dissimilarity measure is a pseudo-metric, the triangle inequality can be applied to make retrieval more efficient [11, 125].

2.4 Efficiency

For large shape collections, it is inefficient to sequentially match all objects in the database with the query object. Because retrieval should be fast, efficient indexing search structures are needed to support efficient retrieval. Since for “query by example” the shape descriptor is computed online, it is reasonable to require that the shape descriptor computation is fast enough for interactive querying.

2.5 Discriminative power

A shape descriptor should capture properties that discriminate objects well. However, the judgement of the similarity of the shapes of two 3D objects is somewhat subjective, depending on the user preference or the application at hand. For example for CAD models often topological properties such as the numbers of holes in a model are more important than minor differences in shape. On the contrary, if a user searches for models looking roughly similar, then the existence of a small hole in the model may be of no importance to the user.

2.6 Partial matching

In contrast to global shape matching, partial matching finds a shape of which a part is similar to a part of another shape. Partial matching can be applied if 3D shape models are not complete, e.g. for objects obtained by laser scanning from one or two directions only. Another application is the search inside "3D scenes" containing an instance of the query object. Also, this feature can potentially give the user flexibility towards the matching problem, if parts of interest of an object can be selected or weighted by the user.

2.7 Robustness and sensitivity

It is often desirable that a shape descriptor is insensitive to noise and small extra features, and robust against arbitrary topological degeneracies, e.g. if it is obtained by laser scanning. Therefore, small changes in a shape should result in small changes in the shape descriptor. On the other hand, if large changes in the shape of the object result in very small changes in the shape descriptor, then the shape descriptor is considered not sensitive. Poor sensitivity will lead to poor discriminative abilities. Also, if a model is given in multiple levels-of-detail, representations of different levels should not differ significantly from the original model.

2.8 Pose normalization

In the absence of prior knowledge, 3D models have arbitrary scale, orientation, and position in the 3D space. Because not all dissimilarity measures are invariant under scale, translation, or rotation, one or more of the normalization procedures described below may be necessary. The normalization procedure depends on the center of mass, which is defined as the center of its surface points.

To normalize a 3D model for scale, the average distance of the points on its surface to the center of mass should be scaled to a constant. Note that normalizing a 3D model by scaling its bounding box is sensitive to outliers. To normalize for translation the center of mass is translated to the origin.

To normalize a 3D model for rotation usually the principal component analysis (PCA) method [99] is applied. It aligns the principal axes to the x-, y-, and z-axes of a canonical coordinate system by an affine transformation based on a set of surface points, e.g. the set of vertices of a 3D model. After translation of the center of mass to the origin, a rotation is applied so that the largest variance of the transformed points is along the x-axis. Then a rotation around the x-axis is carried out such that the

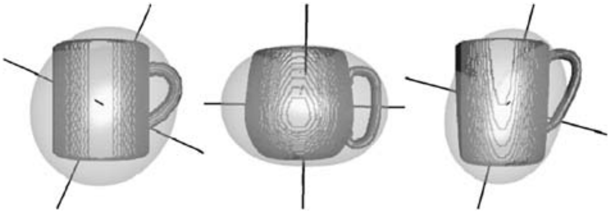

Fig. 2 Similar mugs oriented by principal axes in different ways [42]

maximal spread in the yz-plane occurs along the y-axis. A problem is that differing sizes of triangles are not taken into account which may cause very different results for models that are identical except for finer triangle resolution in some parts of the model. To address this issue, Vranić and Saupe [128] introduced appropriately chosen vertex weights, while Paquet et al. [98] applied centers of mass of triangles as points for the PCA weighted by the triangle areas. Later, Vranić and Saupe [129] generalized the PCA to the continuous PCA so that all of the (infinitely many) points on the mesh surface are equally relevant to compute the transformation aligning the principal axes to a canonical coordinate system. Finally, Novotni and Klein [90] use volumetric properties instead of surface properties to compute the PCA, but their approach is only applicable to watertight models.

Another problem with the PCA algorithm is that due to the lack of information about the direction of the principal axes, either the positive or the negative axes are moved to the x-, y- and z-axes. This results in four valid configurations that align the principal axes (four configurations are not valid because they do not define a proper coordinate system) [90]. Elad et al. [37] solve this ambiguity by choosing the direction of the axes such that the area of the model on the positive side of the x-, y- and z-axes is greater than the area on the negative side.

The PCA algorithm and its variants for pose estimation are fairly simple and efficient. However, if the eigenvalues are equal, principal axes may still switch, without affecting the eigenvalues. Similar eigenvalues may imply an almost symmetrical mass distribution around an axis (e.g. nearly cylindrical shapes) or around the center of mass (e.g. nearly spherical shapes). Figure 2 illustrates the problem.

2.9 Performance measures

The effectiveness of shape retrieval can be measured in performance measures such as precision, recall, bull's eye percentage, k-th tier, ROC curve, specificity, total performance, and relative error. Such performance measures depend on the chosen query, embedding database, and chosen ground truth. They are not an inherent property of the retrieval method, so it is difficult so give exact, objective numbers. Currently, there is no good test set that can serve a similar role as the MPEG-7 core experiment image shape retrieval test set, which can be used to compare 2D shape retrieval methods, see for example the SIDESTEP (Shape-based Image Delivery Statistics Evaluation Project) website [116]. However, a number of initiatives in that direction have recently been developed, such as the Princeton Shape Benchmark [113], the Purdue Engineering Shape Benchmark (ESB) [39], and the AIM@SHAPE Shape Retrieval Contest (SHREC) [114]. In Section 4 we discuss some evaluation studies.

3 Shape matching methods

In this section we discuss 3D shape matching methods. Based on the representation of the shape descriptor we divide shape matching methods into three broad categories: (1) feature based methods, (2) graph based methods and (3) geometry based methods. Figure 3 shows a more detailed categorization of shape matching methods. Note that the classes of these methods are not completely disjoint. For instance, a graph-based shape descriptor, may be extended to describe shape properties not related to topology. In these cases we categorized the method by the most characteristic aspect of its representation. Recently, a number of approaches to combine multiple shape matching methods have been introduced. Also, a number of techniques have been introduced that improve shape matching in general. We discuss these general techniques and approaches to combine methods in Section 4.1.

3.1 Feature based methods

In the context of 3D shape matching, features denote geometric and topological properties of 3D shapes. So, 3D shapes can be discriminated by measuring and

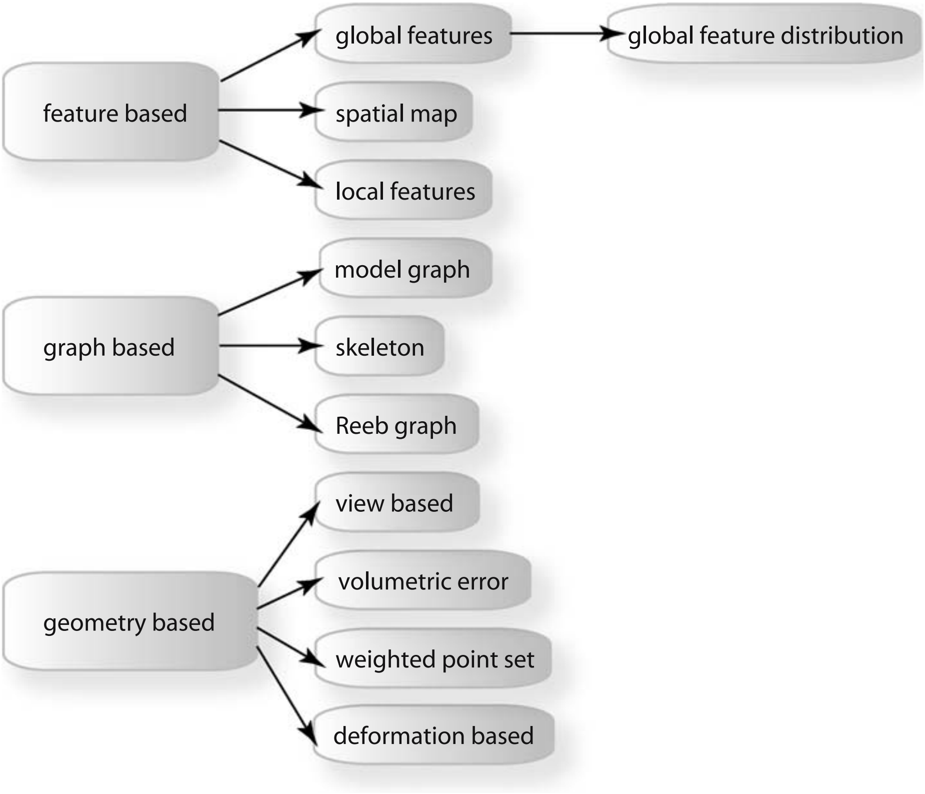

Fig. 3 Taxonomy of shape matching methods

comparing their features. Feature based methods can be divided into four categories according to the type of shape features used: (1) global features, (2) global feature distributions, (3) spatial maps, and (4) local features. Feature based methods from the first three categories represent features of a shape using a single descriptor consisting of a *d*-dimensional vector of values, where the dimension *d* is fixed for all shapes. The value of *d* can easily be a few hundred. The descriptor of a shape is a point in a high dimensional space, and two shapes are considered to be similar if they are close in this space. Retrieving the *k* best matches for a 3D query model is equivalent to solving the *k* nearest neighbors problem in a high-dimensional space. Although this problem is know to be hard in the worst case, matching feature descriptors can be done efficiently in practice by searching in multiple 1D spaces to solve the approximate *k* nearest neighbor problem as shown by Indyk and Motwani [51]. In contrast with the feature based methods from the first three categories, local feature based methods describe the 3D shape around a number of surface points. For this purpose, a descriptor for each surface point is used instead of a single descriptor.

3.1.1 Global feature based similarity

Global features characterize the global shape of a 3D model. Examples of these features are the statistical moments of the boundary or the volume of the model, volume-to-surface ratio, or the Fourier transform of the volume or the boundary of the shape.

Zhang and Chen [132] describe methods to compute global features such as volume, area, statistical moments, and Fourier transform coefficients efficiently.

Paquet et al. [98] apply bounding boxes, cords-based, moments-based and wavelet-based descriptors for 3D shape matching.

Corney et al. [31] introduce convex-hull based indices like hull crumpliness (the ratio of the object surface area and the surface area of its convex hull), hull packing (the percentage of the convex hull volume not occupied by the object), and hull compactness (the ratio of the cubed surface area of the hull and the squared volume of the convex hull).

For fast content-based search of VRML models Kolonias et al. [66] propose to use 3 global shape descriptors: the aspect ratio, a binary 3D shape mask, and the set of edge paths. The set of edge paths descriptor is introduced by Kolonias et al. to characterize the shape of a 3D model. The edge paths are derived from the VRML representation of the 3D models and the similarity between two sets of edge paths is defined as the mean similarity of the edge paths in two 3D models.

The following two methods represent 3D models by spherical functions. Horn [50] introduced the extended Gaussian image (EGI), that maps the surface normal orientation to the unit sphere, called the Gaussian sphere. The EGI is obtained by having each triangle vote on the bin corresponding to its normal direction, with a weight equal to the area of the triangle. Kang and Ikeuchi [59] generalized the EGI to the complex extended Gaussian image (CEGI), which stores for each bin also the normal distance of the surface points to the origin. Both the EGI and the CEGI require normalization for orientation.

Kazhdan et al. [60] describe a reflective (miroring) symmetry descriptor as a 2D function associating a measure of reflective symmetry to every plane (specified by 2 parameters) through the model's centroid. Every function value provides a measure of global shape, where peaks correspond to the planes near reflective symmetry, and valleys correspond to the planes near anti-symmetry. Recently, Kazhdan et al. [63] generalized the concept of reflective symmetry to *k*-fold symmetry. The experimental results of both papers show that the combination of the reflective symmetry descriptor with existing methods provides better results than using only an existing method.

Since only global features are used to characterize the overall shape of the objects, these methods are not very discriminative about object details, but their implementation is straightforward. Therefore, these methods can be used as an active filter, after which more detailed comparisons can be made, or they can be used in combination with other methods to improve results.

Global feature methods are able to support user feedback as illustrated by the following research. Zhang and Chen [133] applied features such as volume-surface ratio, moment invariants and Fourier transform coefficients for 3D shape retrieval. They improve the retrieval performance by an active learning phase in which a human annotator assigns attributes such as airplane, car, body, and so on, to a number of sample models. Elad et al. [37] use a moments-based classifier and a weighted Euclidean distance measure. Their method supports iterative and interactive database searching where the user can improve the weights of the distance measure by marking relevant search results.

3.1.2 Global feature distribution based similarity

The concept of global feature based similarity has been refined recently by comparing distributions of global features instead of the global features directly.

Osada et al. [97] introduce and compare shape distributions, which measure properties based on distance, angle, area and volume measurements between random surface points. They evaluate the similarity between two objects using a pseudo-metric that measures distances between distributions. In their experiments, the D2 shape distribution, which measures distances of random surface points (illustrated in Fig. 4), is most effective.

Ohbuchi et al. [95] investigate shape histograms that are discretely parameterized along the principal axes of inertia of the model. The shape descriptor consists of three

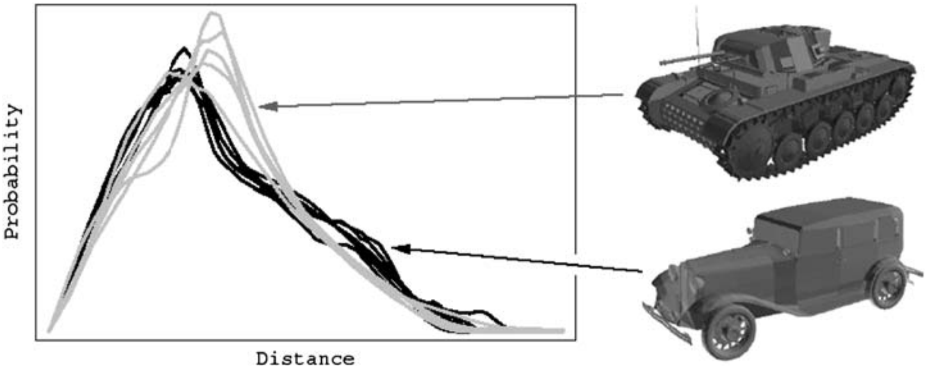

Fig. 4 D2 shape distributions of five tanks (*gray curves*) and six cars (*black curves*) [97]

shape histograms: (1) the moment of inertia about the axis, (2) the average distance from the surface to the axis, and (3) the variance of the distance from the surface to the axis. Their experiments show that the axis-parameterized shape features work only well for shapes having some form of rotational symmetry.

Ip et al. [52] investigate the application of shape distributions in the context of CAD and solid modeling. They refined Osada's D2 shape distribution function by classifying two random points as (1) IN distances if the line segment connecting the points lies completely inside the model, (2) OUT distances if the line segment connecting the points lies completely outside the model, (3) MIXED distances if the line segment connecting the points lies both inside and outside the model. They measure the dissimilarity of two objects a weighted combination of their dissimilarity for the D2, IN, OUT and MIXED distributions. Since their method requires that a line segment can be classified as lying inside or outside the model it is required that the model defines a volume properly. Therefore it can be applied to volume models, but not to polygonal soups. Recently, Ip et al. [53] extend this approach with a technique to automatically categorize a large model database, given a categorization on a number of training examples from the database.

Rea et al. [103] use the difference of the shape distribution of the object and the shape distribution of its convex hull as a descriptor characterizing the concavities of a shape. Ohbuchi et al. [93] investigate another extension of the $D2$ shape distribution function, called the Absolute Angle-Distance histogram, parameterized by the distance between two random points and by the angle between the surfaces on which two random points are located. The latter parameter is actually computed as an inner product of the surface normal vectors. In their evaluation experiment, this shape distribution function outperformed the D2 shape distribution function at about 1.5 times higher computational cost. Ohbuchi et al. [96] improved this method further by a multi-resolution approach computing a number of alpha-shapes at different scales, and computing for each alpha-shape their Absolute Angle-Distance descriptor. Their experimental results show that this approach outperforms the Angle-Distance descriptor at the cost of high processing time needed to compute the alpha-shapes. Rea et al. [102] describe a surface partioning spectrum distribution, which measures the number of maximal connected regions against a range of tolerance values between 0 and 360°, where two adjacent faces are connected if and only if the angle between their normals is less than the tolerance value. Liu et al. [73] propose another variation of a shape distribution function, the thickness histogram estimating thickness of a model from all directions. Pu et al. [101] describe a shape descriptor based on measuring the similarity between two 3D models by measuring the similarity of a series of 2D slices. They compare the similarity between two 2D slices using a two-dimensional D2 shape distribution function.

Shape distributions distinguish models in broad categories very well: aircraft, boats, people, animals, etc. However, they often perform poorly when having to discriminate between shapes that have similar gross shape properties but different detailed shape properties.

3.1.3 Spatial map based similarity

Spatial maps are representations that capture the spatial location of an object. The map entries correspond to physical locations or sections of the object, and are

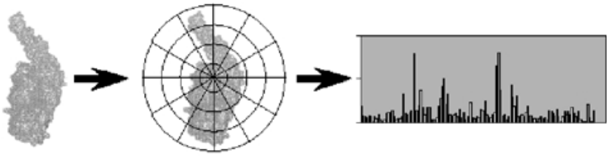

Fig. 5 Shape histogram of a molecular surface defined on concentric shells and sectors around a model's centroid [8]

arranged in a manner that preserves the relative positions of the features in an object. Spatial maps are in general not invariant to rotations, except for specially designed maps. Therefore, typically a pose normalization is done first.

Ankerst et al. [8] use shape histograms as a means of analyzing the similarity of 3D molecular surfaces. The histograms are not built from volume elements but from uniformly distributed surface points taken from the molecular surfaces. The shape histograms are defined on concentric shells and sectors around a model's centroid as illustrated by Fig. 5. The shape histograms are compared using a quadratic form distance measure taking into account the distances between the shape histogram bins.

Vranić et al. [129] introduce a ray-based descriptor that describes a surface by associating to each ray from the origin the distance to the last point of intersection of the ray with the model. For this spherical extent function its spherical harmonics are computed, which form a Fourier basis on a sphere much like the familiar sine and cosine do on a line or a circle. Their method requires pose normalization to provide rotational invariance. Also, Yu et al. [130] propose a descriptor similar to a spherical extent function and a descriptor counting the number of intersections of a ray from the origin with the model. In both cases the dissimilarity between two shapes is computed by the Euclidean distance of the Fourier transforms of the descriptors of the shapes. Their method requires pose normalization to provide rotational invariance.

Kazhdan et al. [61] present a general approach based on spherical harmonics to transform rotation dependent shape descriptors into rotation independent ones. Their method is applicable to a shape descriptor which is defined as either a collection of spherical functions or as a function on a voxel grid. In the latter case a collection of spherical functions is obtained from the function on the voxel grid by restricting the grid to concentric spheres. From the collection of spherical functions they compute a rotation invariant descriptor by (1) decomposing the function into its spherical harmonics, (2) summing the harmonics within each frequency, and computing the L_2-norm for each frequency component. The resulting shape descriptor is a 2D histogram indexed by radius and frequency, which is invariant to rotations about the center of the mass. This approach offers an alternative for pose normalization, because their method obtains rotation invariant shape descriptors. Their experimental results show indeed that in general the performance of the obtained rotation independent shape descriptors is better than the corresponding normalized descriptors obtained using the conventional PCA approach. Their experiments include the ray-based spherical harmonic descriptor proposed by Vranić et al. [129]. Finally, note that their approach generalizes the method to compute the voxel-based spherical harmonics shape descriptor, described by Funkhouser et al. [42], which is defined as a binary function on the voxel grid. Note that Kazhdan et al. [61] use the negatively exponentiated Euclidean distance transform of the surface of a 3D model to compute the 3D voxel grid.

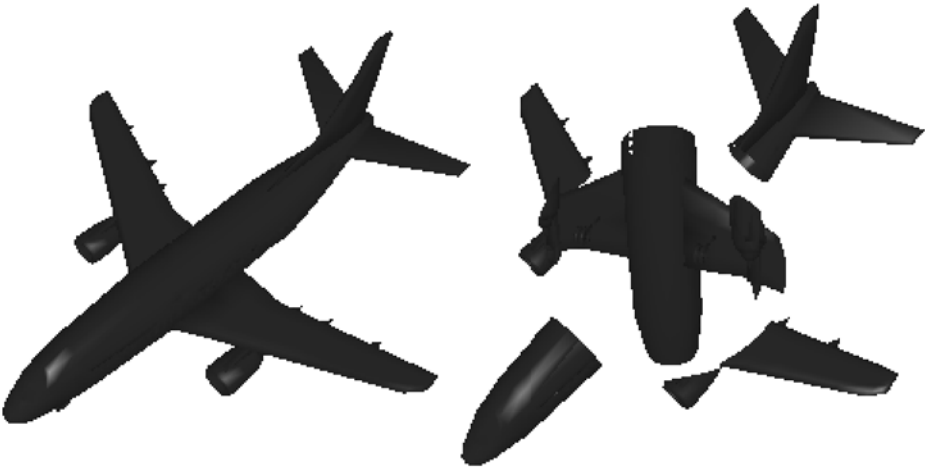

Fig. 6 Spherical harmonics do not distinguish models that differ by a rotation of an interior part [61]

Novotni and Klein [91] present a method to compute 3D Zernike descriptors from voxelized models as natural extensions of spherical harmonics based descriptors. 3D Zernike descriptors capture object coherence in the radial direction as well as in the direction along a sphere. Both 3D Zernike descriptors and spherical harmonics based descriptors achieve rotation invariance. However, by sampling the space only in the radial direction the latter descriptors do not capture object coherence in the radial direction, as illustrated by Fig. 6.

The limited experiments comparing spherical harmonics and 3D Zernike moments performed by Novotni and Klein show similar results for a class of planes, but better results for the 3D Zernike descriptor for a class of chairs.

Vranić [126] expects that voxelization is not a good idea, because many fine details are lost in the voxel grid. Therefore, he introduces the radial extent function as a spherical harmonics shape descriptor based on using functions defined on concentric shells around the centre of a model, and compares it with the voxel-based spherical harmonics shape descriptor proposed by Funkhouser et al. [42]. Also, Vranić et al. accomplish pose normalization using the so-called continuous PCA algorithm [129], which calculates sums of integrals over all triangles in a mesh model, instead of using the conventional PCA algorithm. The experimental results from the paper show that the continuous PCA is better than the conventional PCA and better than the weighted PCA, which takes into account the differing sizes of the triangles of a mesh. Also it is shown that the shell-based spherical harmonic descriptor outperforms the voxel-based spherical harmonic descriptor proposed by Funkhouser et al. [42]. Liu et al. [72] point out that Vranić's method is unstable, because noise can shift the centre of the concentric shells. As an alternative they propose a method based on spherical harmonics using Delta functions defined by the distance to the centre of the model. Their experimental results using their own database of 740 models show better performance for their method than the method by Vranić [126].

Kriegel et al. [68, 69] investigate similarity for voxelized models. They obtain a spatial map by partitioning a voxel grid into disjoint cells which correspond to the histograms bins. They investigate three different spatial features associated with the grid cells: (1) volume features recording the fraction of voxels from the volume in each cell, (2) solid-angle features measuring the convexity of the volume boundary in each cell, (3) eigenvalue features estimating the eigenvalues obtained by the PCA applied to the voxels of the model in each cell [69]. Moreover, they investigate a fourth method, using flexible partitioning of the voxels by cover sequence features, which approximate the model by unions and differences of cuboids, each containing a number of voxels [68]. Their experimental results show that the eigenvalue method and the cover sequence method outperform the volume and solid-angle feature method. Their method requires pose normalization to provide rotational invariance. Instead of representing a cover sequence with a single feature vector, Kriegel et al. [68] represent a cover sequence by a set of feature vectors. This approach allows an efficient comparison of two cover sequences, by comparing the two sets of feature vectors using a minimal matching distance.

Ricard et al. [105] introduce a 3D angular radial transform (ART) shape descriptor. The angular radial transform represents a 3D shape represented in spherical coordinates as the product of a radial basis function along the angular and two radial basis functions along the radial directions. The shape descriptor consists of an array of the ART coefficients of these basis functions. Since the ART coefficients are invariant to rotation around the z-axis, their method requires only alignment of the z-axis to the first principal axis computed by the PCA. Their experimental results show that for the Princeton Shape Benchmark, methods based on spherical harmonics are better, and for a database provided by Renault consisting of 5000 models of car parts, the results of both methods are similar.

The spatial map based approaches show good retrieval results. But a drawback of these methods is that partial matching is not supported, because they do not encode the relation between the features and parts of an object. Furthermore, these methods provide no feedback to the user about why shapes match.

3.1.4 Local feature based similarity

Local feature based methods provide various approaches to take into account the surface shape in the neighborhood of points on the boundary of the shape.

Shum et al. [115] use a spherical coordinate system to map the surface curvature of 3D objects to the unit sphere. By searching over a spherical rotation space, a distance between two curvature distributions is computed and used as a measure for the similarity of two objects. Unfortunately, the method is limited to objects which contain no holes, i.e. have genus zero.

Zaharia and Prêteux [131] describe the 3D Shape Spectrum Descriptor, which is defined as the histogram of shape index values, calculated over an entire mesh. The shape index, first introduced by Koenderink [65], is defined as a function of the two principal curvatures on continuous surfaces. They present a method to compute these shape indices for meshes, by fitting a quadric surface through the centroids of the faces of a mesh. Unfortunately, their method requires a non-trivial preprocessing phase for meshes that are not topologically correct or not orientable.

Chua and Jarvis [28] compute point signatures that accumulate surface information along a 3D curve in the neighborhood of a point. Johnson and Herbert [58] introduce spin images: 2D histograms of the surface locations around a point. They apply spin images to recognize models in a cluttered 3D scene. Due to the complexity of their representation [28, 58] these methods are very difficult to apply to 3D shape matching. Also, it is not clear how to define a dissimilarity function that satisfies the triangle inequality.

Körtgen et al. [67] apply 3D shape contexts for 3D shape retrieval and matching. 3D shape contexts are semi-local descriptions of object shape, centered at points on the surface of the object, and are a natural extension of 2D shape contexts introduced by Belongie et al. [13] for recognition in 2D images. The shape context of a point **p** is defined as a coarse histogram of the relative coordinates of the remaining surface points. The bins of the histogram are defined by the overlay of concentric shells around the centroid of the model and sectors emerging from the centroid. Matching consists of a local matching stage and a global matching stage. In the local matching stage, for all points **p** the best matching point **q** is found on the other shape. In the global matching stage, correspondences between similar sample points on the two shapes are found.

Compared to the methods presented in the previous sections of this paper, matching 3D shape contexts is less efficient, efficient indexing is not straightforward, and the obtained dissimilarity measure does not obey the triangle inequality.

3.2 Graph based methods

In general, the feature based methods discussed in the previous section take into account only the pure geometry of the shape. In contrast, graph based methods attempt to extract a geometric meaning from a 3D shape using a graph showing how shape components are linked together. Graph based methods can be divided into three broad categories according to the type of graph used: (1) model graphs, (2) Reeb graphs, and (3) skeletons. For an extensive discussion of Reeb graphs and skeletons we refer the reader to the paper of Biasotti et al. [19].

Efficient computation of existing graph metrics for general graphs is not possible: computing the edit distance is NP-hard [135] and computing the maximal common subgraph [45] is even NP-complete. Polynomial solutions can be obtained for directed acyclic graphs such as shock graphs. Sebastian et al. [108] describe an approach to compute a pseudo-metric between shock graphs. It is obtained by exhaustively searching for the optimal deformation path between two 2D shapes, and using the cost of this path as a distance between two shapes. But the computation time of this method is too high for practical application, and it is not straightforwardly generalized to 3D.

3.2.1 Model graph based similarity

Model graph based similarity methods are applicable to 3D solid models as produced by most CAD systems. The most dominant solid modeling representation methods are boundary representation (B-rep) and constructive solid geometry (CSG). A B-rep describes a model in terms of its vertices, edges and faces. By contrast to the facets in meshes, the faces of a B-rep may be represented as free-form surfaces. CSG describes a model in terms of a set of Boolean operations applied to primitive geometric entities such as cubes and cylinders. For an introduction to solid modeling representations we refer the reader to the book by Hoffmann [49]. For content based retrieval of solid models, researchers have investigated the application of graph-based data structures storing engineering features (machining features, form features, etc.). Elinson et al. [38], and Cicirello and Regli [29] investigate the application of model dependency graphs storing machining features. These approaches compare the similarity of solid models by comparing their associated manufacturing plans.

McWerther et al. [80–82], and El-Mehalawi and Miller [35, 36] apply model signature graphs that both model the topology of a shape model by a graph structure and map a number of engineering features to a high-dimensional feature vector. Model signature graphs represent the faces from the B-Rep data structure of the solid model as nodes and the boundary curves between the faces as edges. This approach allows comparison of shape models by comparing their topology using graphs and comparing their other properties using feature vectors in the same way as feature based methods. Therefore, McWerther et al. [80–82] apply approximate graph comparison using the spectrum of the graph. The graph spectrum, which consists of the sorted eigenvalues of the adjacency matrix of the graph, is strongly related to the structure of the graph, and hence to the topology of the shape model. El-Mehalawi and Miller [35, 36] apply approximate graph comparison based on local clique matching.

The model graph based approaches are especially relevant for the CAD/CAM community, but are difficult to apply to models of natural shapes like humans and animals. To the best of our knowledge only Zuckerberger et al. [136] applied an approach similar to model graphs to content based retrieval suitable for natural shapes. They decompose the surface of a model into patches classified as similar to a sphere, a cylinder, a cone or a plane, and identify adjacent patches to build a graph representation of the model.

3.2.2 Skeleton based similarity

Sundar et al. [117] use as a shape descriptor a skeletal graph that encodes geometric and topological information. After voxelization of a shape, the skeletal points are obtained by a distance transform-based thinning algorithm developed by Gagvani [44] using a thinness parameter. The skeletal points are connected in an undirected acyclic shape graph by applying the minimum spanning tree algorithm. Decreasing the thinness results in denser skeletal graphs. So, by using different values of the thinness parameter they obtain a hierarchical graph structure. Each node in the graph represents a segment of the original skeleton. With each node a geometrical signature vector is associated encoding the radial distribution about the segment. Also, with each node of the graph a topological signature vector is associated encoding the topology of the subtrees rooted at the node. This topological signature vector is defined recursively over the subgraphs of the node using eigenvalues of their adjacency matrices. Sundar et al. [117] match two shapes by approximate comparison of their hierarchical skeletal graphs using a greedy algorithm finding the maximum cardinality, minimum weight matching in a bipartite graph. Efficient indexing of a skeletal graph is supported by storing for each node its topological signature vector. Since a topological signature vector has a fixed size, the size of its shape descriptor is a constant multiplied by the number of nodes of the skeleton. Figure 7 illustrates their shape matching approach. Their method also supports matching of articulated objects, by taking into account only the topological signature vector as a descriptor,

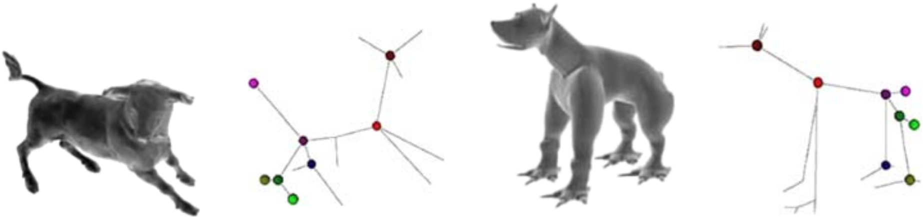

Fig. 7 Skeletal graph matching with colors showing the node-to-node correspondence based upon the topology and radial distance about the edge [117]

and partial matching, where a skeletal graph is allowed to match with a subgraph of another skeletal graph.

Iyer et al. [56, 76] use both global features and skeletal graphs to describe volume models, obtained by voxelizing solid models. They obtain a skeletal graph by a thinning algorithm iteratively eroding voxels until a one-voxel width skeleton is left. They match two shapes by an algorithm detecting graph/subgraph isomorphism using a decision-tree based approach developed by Messmer and Bunke [83]. Their algorithm indexes all the graphs in a database in the form of a decision tree using the various permutations of the adjacency matrix. Hence, this algorithm obtains a search time polynomial in the number of graph nodes, at the cost of exponential space requirements. Their results show the feasibility of their approach for relatively small volume models. Very recently, Iyer et al. [54] generalized the above method to a multi-scale hierarchical skeletal graph representation storing local shape information such as normalized entity lengths and curvature distribution. In contrast with their earlier work [56, 76] described above, they apply a heuristic-based genetic algorithm developed by Marchiori [79] to find the homomorphism yielding the highest similarity between two skeletal graphs. Some query examples on a database of 105 parts illustrate that using local shape information improves retrieval.

3.2.3 Reeb graph based similarity

Mathematically, the Reeb graph is defined as the quotient space of a shape S and a quotient function f. Biasotti et al. [20] compare Reeb graphs obtained by using different quotient functions f and highlight how the choice of f determines the final matching result. For instance, the integral geodetic distance as quotient function is especially suited for articulated objects, while the distance to the barycenter should be preferred if the aim is to distinguish different poses of an articulated object. Hilaga et al. [48] describe a topological matching method relevant especially for articulated objects. Their method uses Reeb graphs based on a quotient function defined by an integral geodesic distance. Bespalov et al. [16] investigate the application of Hilaga's method to solid models. They found that for solid models, minor changes in topology may result in significant differences in similarity. Since for solid models topological insensitivity is important, they conclude that the Reeb graph technique requires some improvements. Bespalov et al. [17] present preliminary research on a modification of Hilaga's method, which computes a scale-space decomposition of a shape, represented as a rooted undirected tree instead of a Reeb graph. This reduces the problem of comparing two 3D models to computing a matching among the corresponding rooted trees. Tung and Schmitt [122] propose to augment a multi-resolution Reeb graph with geometrical attributes like volume, cords and curvature of the surface part associated with a node of the Reeb graph. Also, their approach supports partial matching.

In summary, Reeb graphs defined by a geodesic distance are suited for matching articulated objects, but they are sensitive to topological changes. Also, they cannot be applied to arbitrary meshes, because topological problems like missing faces disturb the computation of geodesic distances.

3.3 Geometry based methods

Finally, we discuss a number of geometry based methods that are applied to shape matching.

3.3.1 View based similarity

The main idea of view based similarity methods is that two 3D models are similar, if they look similar from all viewing angles, similar to one of the models of human object recognition [106]. A natural application of this paradigm is the implementation of query interfaces based on defining a query by one or more sketches showing the query from different views. Löffler [74] applies view based similarity to retrieve 3D models using a 2D query interface. In the preprocessing phase, for each 3D model a descriptor is obtained consisting of a number of binary images. In the query phase, a sketch or a 2D image is used as a query to retrieve a number of 3D models, whose images match the query. Also, Funkhouser et al. [42] apply view based similarity to implement a 2D sketch query interface. In the preprocessing phase a descriptor of each 3D model is obtained by 13 thumbnail images of boundary contours of the 3D object as seen from 13 orthographic view directions. Then in the query phase the user defines a 3D shape query by drawing one or more sketches. 3D shape models are retrieved by comparing these sketches with the descriptors from the shapes in the database using image matching.

In the approach described by Cyr and Kimia [32], a query is specified by a view of a 3D object. A descriptor of a 3D object consists of a number of views of the 3D object. The number of views of each object is kept small by clustering views, and by representing each cluster with one view, which is represented by a shock graph. They recognize a 3D shape by comparing a view of the shape with all views of 3D objects using shock graph matching. However, they do not address the shock graph indexing problem, resorting to a linear search of all views in the database in order to retrieve an object.

Using shock graph matching, Macrini et al. [77] apply indexing using topological signature vectors to implement view based similarity matching more efficiently. Also, recently, view based similarity has been applied to retrieve 3D objects by Chen et al. [27, 112]. They consider two models to be similar,if they look similar from all viewing angles. A lightfield descriptor is introduced, which compares ten silhouettes of the 3D shape obtained from ten viewing angles distributed evenly on the viewing sphere, as illustrated by Fig. 8. Each silhouette is a 2D image, encoded by its Zernike moments and Fourier descriptors. The dissimilarity of two shapes is found as the minimal

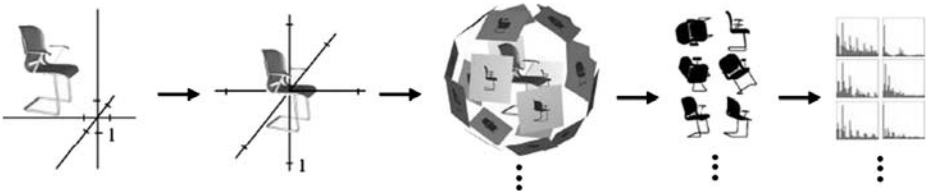

Fig. 8 Extraction of the lightfield descriptor for a chair model [112]

dissimilarity obtained by rotating the viewing sphere of one lightfield descriptor relative to the other lightfield descriptor. The running time of the retrieval process is reduced by a clever multi-step approach supporting early rejection of non-relevant models. The experimental results show that their approach obtains better retrieval results than the 3D harmonics approach proposed by Funkhouser et al. [42] using a test database containing 1,833 models, at the cost of much more processing time. In his Ph.D. thesis Vranić [127] describes a shape descriptor based on three silhouette images perpendicular to the *x*-, *y*- and *z*-axis of a canonical coordinate system. The shape feature for each silhouette image is a sequence of discrete Fourier coefficients. In contrast with the approach by Chen et al. [27], the approach by Vranić requires pose estimation.

Ohbuchi et al. introduce a shape descriptor based on the appearance of a set of depth-buffer images viewed from 42 viewpoints. The shape feature for each depth image is a rotationally invariant generic Fourier descriptor for 2D images developed by Zhang and Lu [134]. The set of these 42 feature vectors is the shape descriptor of the model. The dissimilarity between two shape models is computed by minimizing the distance between all the 42^2 possible combinations of two sets of feature vectors. The experimental results show that their approach obtains better retrieval results than the D2 shape distribution approach proposed by Osada et al. [97] using a test database containing 1,213 models, at the cost of much more processing time. Vranić [127] describes a descriptor based on the appearance of a set of depth-buffer images viewed from 6 viewpoints perpendicular to the *x*-, *y*- and *z*-axis of a canonical coordinate system. The shape feature for each depth image is a sequence of discrete Fourier coefficients. In contrast with the approach by Ohbuchi et al. [94] the approach by Vranić requires pose estimation.

Also, Ansary et al. [9] present an approach to search by a 2D view. To reduce the search space they select from a subset of 80 views equally spaced on the unit sphere a small subset of characteristic views. Also, they present a probabilistic approach to improve retrieval of a 3D object using a 2D view as a query. In their experiments they found that 2D Zernike moments outperformed curvature histograms and curvature scale space representations.

Other view based methods try to select those views that contain most information, for example using principal component [70].

3.3.2 Volumetric error based similarity

Novotni and Klein [90] describe a geometry similarity approach to 3D shape matching based on calculating a volumetric error between one object and a sequence of offset hulls of the other object. A drawback of their method is that their dissimilarity measure is not symmetric and does not obey the triangle inequality. Sánchez-Cruz and Bribiesca present a method [107] relating the volumetric error between two voxelized shapes to a transportation distance measuring how many voxels have to move, and how far, to change one shape into another. Since in general these voxelized shapes will have many voxels, the computation cost of this transportation distance will be high.

3.3.3 Weighted point set based similarity

Another approach is based on shape descriptors consisting of weighted 3D points. Dey et al. [34] present a method to obtain a descriptor of a shape, given by a point sample, by first decomposing the shape into its components. They obtain as shape descriptor a weighted point set by representing each component by a point with a weight denoting the volume of the component. They match weighted point sets by a measure which does not obey the triangle inequality. Tangelder and Veltkamp [120] use as shape descriptors weighted point sets consisting of points with a high curvature value. A measure for the curvature is used as a weight. They compare weighted point sets using a variant of the Earth mover's distance, the proportional transportation distance, which obeys the triangle inequality [46]. Shamir et al. [110] propose a shape descriptor consisting of a hierarchy of weighted point sets, representing spherical shape approximations. They utilize this multi-resolution approximation to implement an algorithm to simultaneously align and compare two shapes. Funkhouser et al. [41] developed a weighted point matching shape descriptor especially relevant for partial matching as part of an example based 3D modeling system. The system allows the user to select parts of an object as a query object and to search for objects containing parts similar to the query object. This shape descriptor is based on the sum of squared distances for models aligned in the same coordinate system. Partial matching is supported by associating weights to the points in the user selected part of the model. The efficient computation of the sum of squared distances is supported using squared Euclidean distance transforms of the models. Efficient retrieval is done using a standard singular value decomposition (SVD) technique to project the shape descriptors to a low-dimensional subspace. The experimental results show that for the Princeton Shape Benchmark, the shape descriptor outperforms the 3D harmonics approach proposed by Funkhouser et al. [42] and the radial extent function [126].

3.3.4 Deformation based similarity

A number of methods [12, 30] compare a pair of 2D shapes by measuring the amount of deformation required to register the shapes exactly. These methods depend on the natural arc length parameterization of their contours, which is not straightforwardly generalized to 3D. As a result, methods that use deformation for shape fitting [121] or shape evolution [33] are very difficult to apply to 3D shape matching.

4 Comparisons, combinations, and generic techniques

4.1 Experimental comparisons and mixed methods

Very recently available benchmark databases, containing models classified in different categories containing similar shapes, have been applied for experimental comparison of shape matching methods as well as the development of mixed methods. An example of such a benchmark is the Princeton Shape Benchmark, a publicly available database of 3D models, software tools, and a standarized set of experiments, for comparing 3D shape matching algorithms. The database contains 1,814 models collected from the World Wide Web and classified by humans according to function and form. Shilane et al. [113] applied the Princeton Shape Benchmark to compare 12 shape descriptors. Their results show that shape matching algorithms do not perform equally well on all object types. For instance, extended Gaussian images are good at discriminating between man-made and natural objects, but not that good at making detailed class distinctions. They also find that the lightfield descriptor [27] is the most discriminating between the 12 shape descriptors tested, but at higher storage and computational costs than most other descriptors. Vranić [127] compares a number of shape matching methods using the Princeton shape benchmark, the publicly-available database at the University of Konstanz, (containing 473 models classified into 55 classes and 1,366 unclassified models), and the MPEG-7 test set [87] (not publicly available). The experimental tests by Vranić show that from the descriptors proposed by other authors, the spherical harmonics approach using the negatively exponentiated Euclidean distance transform [61] significantly outperforms all other approaches. However, the lightfield descriptor [27] found best by the experiments described by Shilane et al. [113] has not been included in his experiments. Del Bimbo and Pala [21] present a comparative analysis of a few different approaches with respect to robustness against deformations, ability to capture an object structural complexity, and the resolution of the models. The results of the 3D Shape Retrieval Contest of 2006 and 2007 [114] are in line with the experience that view based methods perform well.

Vranić [127] also investigates a number of hybrid shape matching methods obtained by combining shape matching methods. In his framework similarity is measured using a weighted combination of the similarity of the separate methods, where the weight of each separate method is proportional to the dimension of its feature vector. His experiments show that for comparing overall performance, a combination of the depth buffer-based descriptor [127] and the silhouette-based descriptor [127], both described in Section 3.3.1, and the ray-based descriptor [129] described in Section 3.1.3, is the best. A similar study to combining features has been performed by Atmosukarto et al. [10].

Bustos et al. [22, 23] propose to use k-entropy impurity as a measure for the a priori estimation of the power of an individual method given a query. They experimented with a technique that applies the k-entropy impurity heuristic to select the best method and a technique that selects a weighted combination of methods. Their results show that the combination technique especially improves the effectiveness of shape retrieval.

4.2 Generic techniques

Nehab and Shilane [89] describe the stratified point sampling technique that generates evenly spaced point samples. They compare nine shape descriptors using stratified point sampling against using uniform sampling to compute the same shape descriptors. Their experimental results show that shape descriptors computed by stratified sampling outperform the ones computed by uniform sampling.

Kazhdan et al. [62] describe a new shape matching paradigm based on anisotropic scaling, which compares two models by (1) transforming each of them into isotropic models, (2) comparing the geometric similarity of the isotropic models, and (3) defining the measure of model similarity as a function of both the similarity of isotropic models, and the difference in their initial anisotropic scales. Their experimental results show that for most shape matching methods the application of anisotropic scaling improves the matching performance.

Akgül et al. [7] build a density based shape descriptor on a number of chosen local features, by estimating the probability density function (pdf). The final shape descriptor is the pdf values evaluated at a set of locations in the range of the local feature values.

5 Overview and conclusions

In this section we summarize our discussion on shape matching methods from the previous sections and indicate directions for further research.

Feature based methods, categorized into (1) global features, (2) global feature distributions, (3) spatial maps and (4) local features, characterize shapes by their feature values. The shape matching methods from the first three categories represent the feature values by a vector in a high d-dimensional vector space. Since the feature values are typically computed by sampling 3D shapes, no restrictions on the kind of shape model are imposed and in general the descriptor computation is fast. Because a feature vector is a point in a fixed d-dimensional space, two models can be compared fast by computing their distance in this space. Also, indexing is straightforward and retrieval can be implemented efficiently by nearest neighbour search. In general these methods are robust, because they are based on sampling. For most features normalization is required, e.g. using the PCA, or rotation invariant shape descriptors should be obtained (e.g. using Kazhdan's method [61]). The discriminative abilities of Osada's method [97] have been improved by further refinements of distribution methods as well as by several methods based on spatial maps [61, 91, 126]. If details of shapes are not taken into account, these methods distinguish shapes very well. Details may be taken into account using higher order moments, but this has not been verified by experiments. A drawback of these methods is that partial matching is not supported, because they do not encode the relation between the features and parts of an object. Furthermore, these methods provide no feedback to the user about why shapes match.

Local feature based methods compute feature value vectors for a number of surface points. The comparison of local feature methods to the methods from the first three categories, shows that for the former methods matching is less efficient,

efficient indexing is not straightforward, and the obtained dissimilarity does not obey the triangle inequality. But local feature based methods provide opportunities for partial matching. Therefore, more research on these methods is worthwhile.

Model graphs are extracted from solid model representations [49] used by most CAD systems. Therefore, the model graph based approaches are only applicable to solid models, while skeletal graph and Reeb graph approaches are applicable to volume models including models of natural shapes like humans and animals represented as volumes. The only exception is the model graph based approach by Zuckerberger et al. [136], which is also applicable to natural shapes.

For graph based descriptors the complexity of the exact computation of a metric obeying the triangle inequality prevents practical application. Hence, the efficient implementation of approximate matching methods is a current research issue. Pure graph based methods have a limited discriminating power, because only topology is taken into account. To improve discriminative abilities, most authors apply graph based matching in combination with other methods. For instance, Sundar et al. [117] match two skeletal graphs based upon the topology and radial distance about the edges of the skeletal graphs. For graph based methods, minor changes in topology may result in significant differences in similarity. Hence, these methods are less robust than feature based methods. Advantages of graphs based methods are that no pose normalization is required, and that a graph based structure is suited to implement partial matching.

The view based similarity approach recently implemented by Chen et al. [27] provides good retrieval results at the cost of processing time. It does not require pose normalization, because its similarity comparison is rotation independent. Since their experimental results, obtained with their own database, show better discriminative abilities than other methods, it would be interesting to compare these methods using other databases. The volumetric error approach implemented by Novotni et al. [90] allows fast matching, but uses a dissimilarity measure that does not obey the triangle inequality. In contrast, Sánchez-Cruz and Bribiesca [107] use a pseudo-metric dissimilarity measure which is expensive to compute. Among the weighted point set approaches only Tangelder et al. [120] and Funkhouser et al. [41] use a pseudo-metric to compare weighted point sets. The weighted point matching approach developed by Funkhouser et al. [41] efficiently supports part-in-whole matching. Deformation based methods which have been applied to 2D shapes, are too slow for 3D shape matching. Finally, mixed methods based on approaches such as weighted combinations of shape matching methods [127] or on a query-dependent a priori estimation of the power of each available shape matching method [22, 23] as well as general shape matching improving techniques such as anisotropic scaling [62] and stratified point sampling [89] are powerful tools to develop better shape matching methods. Table 1 summarizes the above discussion with respect to several requirements of content based 3D shape retrieval. In this table the efficiency is qualified as low, medium or high based on database retrieval efficiency, where high efficiency supports interactive retrieval, i.e. answering queries within a second, and medium efficiency supports almost interactive retrieval, i.e. answering queries within a few seconds, using a database containing thousands of models. We did not include a comprehensive study of algorithmic complexity, because for a number of techniques such as view based methods, model graphs, deformation based methods, weighted point set methods, it is difficult to determine algorithmic complexity.

Table 1 Comparison of shape matching methods

	References	Shape model	Triangle inequality	Efficiency	Discriminative power	Partial matching	Robustness	Normalization required
Global feature	[31, 37, 50, 59, 60, 63] [66, 98, 132, 133]	All models	Yes	Fast	Low	No	High	Only [50, 59]
Global feature distribution	[52, 53, 73, 93, 95] [96, 97, 101–103]	All models	Yes	Fast	Medium	No	High	Only [95, 101]
Spatial map	[8, 42, 61, 68] [69, 72, 91, 105] [126, 129, 130]	All models	Yes	Fast	High [61, 91, 129]	No	High	Except [42, 61, 91]
Local feature	[28, 58] [67, 115, 131]	Mesh	Unknown	Medium	Medium	Yes	Medium	No
Model graph	[29, 35, 36, 38] [80–82, 136]	Solid except [136]	Not Applicable in practice	Medium	Medium	Yes	Medium	No
Skeleton	[54, 56, 76, 117]	Volume	Not Applicable in practice	Medium	Medium	Yes	Medium	No
Reeb graph	[16, 17, 20, 48, 122]	Volume	Not Applicable in practice	Medium	Medium	Yes	Medium	No
View	[9, 27, 32, 42] [74, 77, 94, 127]	Mesh	[27]	Medium	High [27]	No	High	No
Volumetric error	[90, 107]	Volume	[107]	Fast [90]	Medium	No	High	Yes
Weighted point set	[34, 41, 110, 120]	Mesh	[41, 120]	Fast [41]	High [41]	[41]	High [41]	Yes
Deformation	[33, 121]	Mesh	Unknown	Slow	Medium	No	High	No
Mixed methods	[10, 22, 23, 127]	All Models	Yes	Fast	High	No	High	No

References indicate which papers provide the indicated property. If no reference is indicated the property is valid in general

We identify the following research issues:

- **Experimental evaluation using benchmarks** Comparison of different shape matching methods using publicly available benchmark databases, containing models classified in different categories containing similar shapes, has already led to valuable comparisons of existing shape matching methods. In future work, to evaluate the power of newly developed shape matching methods, researchers should compare their methods against other state-of-the art shape matching methods using these publicly available benchmarks. In 2006, the 3D Shape Retrieval Contest (SHREC) [114] consisted of a single track, retrieval from a database of polygonal models, in 2007 it consisted of five tracks: watertight models, partial matching, protein models, CAD models, and 3D face models.
- **Obtaining rotation invariance** To obtain rotation invariant dissimilarity measures, Vranić et al. [129] advocate the use of the continuous PCA method for pose normalization, while Kazhdan et al. [61], and Novotni and Klein [91], favor using rotation invariant shape descriptors that need no pose normalization. Further research is needed to improve and compare both approaches.
- **Efficient indexing** The vantage method [125] may be applied to compute an efficient index structure for pseudo-metrics that require much computing time. Also, clustering of *d*-dimensional feature vectors [13] can be applied for efficient indexing. For graph descriptors the development of an efficient indexing method is a major research topic. For instance, Sebastian et al. [109] describe a promising approach for indexing shock graphs.
- **Partial matching** Local feature based methods and graph based methods in general applicable to partial matching. The weighted point method developed by Funkhouser et al. [41], where the user selects a part of an object as a query, is a first powerful tool for part-in-whole matching. This allows a user to search for shape models containing an instance of the query object. However, for part-to-part matching, the part to be matched in the query object is not know beforehand. Based on local feature correspondences, Funkhouser and Shilane [43] perform partial matching by backtracking using a priority queue. Part-to-part matching would facilitate the automatic reconstruction of a 3D model from partial models obtained by e.g. laser scanning, where models may have to be reconstructed from thousands of partial models [71].
- **Combining shape matching methods** The mixed methods proposed so far [22, 23, 127] all combine feature based methods. Since the capabilities of feature based methods (fast computation, pseudo-metric, discriminative abilities, robustness) are orthogonal to the capabilities of graph based methods (partial matching, no normalization required), combining different approaches may produce more powerful shape matching methods. Also combining geometry and topology based approaches may produce better shape matching methods.
- **Domain-specific shape retrieval** In general, any fully-fledged system should apply as much domain knowledge as possible, in order to make shape retrieval effective. For example, it makes a lot of difference whether a user is looking for industrial mechanical parts, or molecular structures. Bioinformatics has become a large area of research, one part of which is the comparison of the protein structures based on their geometric shape, since their 'higher structure' is so important for their function. Industrial designers often want to start designing a new part by adapting an existing similar one, where the similarity partly depends

on functional or manufacturing aspects, which should play a role in the search process. In general, there are many domains where semantics and ontology play an important role, but which are only barely trodden until now.

Acknowledgements The Figs. 2, 4, 5, 6, 7, and 8 are extracted from [42, 97, 8, 61, 117, 112], respectively. We thank Patrick Min for his valuable comments on a draft of this paper, and the reviewers for their useful suggestions. This research was partially supported by the FP6 IST Network of Excellence 506766 AIM@SHAPE.

References

1. 3D model retrieval system. http://3d.csie.ntu.edu.tw/~dynamic/
2. 3D model search engine. http://shape.cs.princeton.edu
3. 3D model similarity search engine. http://merkur01.inf.uni-konstanz.de/CCCC
4. 3D search tool. http://3d-search.iti.gr/3DSearch
5. 3D shape retrieval engine. http://www.cs.uu.nl/centers/give/multimedia/3Drecog/3Dmatching.html
6. AIM@SHAPE shape repository. http://shapes.aim-at-shape.net/
7. Akgül CB, Sankur B, Schmitt F, Yemez Y (2007) Multivariate density-based 3D shape descriptors. In: Proceedings of shape modeling international, pp 3–12
8. Ankerst M, Kastenmuller G, Kriegel H-P, Seidl T (1999) 3D Shape histograms for similarity search and classification in spatial databases. In: Proc. symposium on large spatial databases, pp 207–226
9. Ansary TF, Vandeborre J-P, Mahmoudi S, Daoudi M (2004) A bayesian framework for 3D models retrieval based on characteristic views. In: Proc. 2nd international symposium on 3D data processing, visualization, and transmission
10. Atmosukarto I, Leow WK, Huang Z (2005) Feature combination and relevance feedback for 3D model retrieval. In: Proceedings of the 11th international multimedia modelling conference, pp 334–339
11. Barros J, French J, Martin W, Kelly P, Cannon M (1996) Using the triangle inequality to reduce the number of comparisons required for similarity-based retrieval. In: Proc. SPIE conference 267, pp 392–403
12. Basri R, Costa L, Geiger D, Jacobs D (1998) Determining the similarity of deformable shapes. Vis Res 38:2365–2385
13. Belongie S, Malik J, Puzicha J (2002) Shape matching and object recognition using shape contexts. IEEE Trans Pattern Anal Mach Intell 24(4):509–522
14. Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, Weissig H, Shindyalov IN, Bourrne PE (2000) The protein data bank. Nucleic Acids Res 28:235–242
15. Besl PJ, Jain RC (1985) Three-dimensional object recognition. Comput Surv 17(1):75–145
16. Bespalov D, Shokoufandeh A, Regli WC (2003) Reeb graph based shape retrieval for CAD. In: Proc. ASME design engineering technical conferences, computers and information in engineering conference
17. Bespalov D, Shokoufandeh A, Regli WC, Sun W (2003) Scale-space representation of 3D models and topological matching. In: Proc. solid modeling 2003, pp 208–215
18. Biasotti S, Giorgi D, Marini S, Spagnuolo M, Falcidieno B (2006) A comparison framework for 3D object classification methods. In: Multimedia content representation, classification and security, proceedings international workshop, MRCS 2006. Lecture notes in computer science, vol 4105, pp 314–321
19. Biasotti S, Marini S, Mortara M, Patané G (2003) An overview of properties and efficacy of topological skeletons in shape modelling. In: Proc. shape modeling international 2003, pp 245–254
20. Biasotti S, Marini S, Mortara M, Patané G, Spagnuolo M, Falcidieno B (2003) 3D shape matching through topological structures. In: Proc. discrete geometry for computer imagery, pp 194–203
21. Bimbo AD, Pala P (2006) Content-based retrieval of 3D models. ACM Trans Multi Commun App 2(1):20–43

22. Bustos B, Keim D, Saupe D, Schreck T, Vranić D (2004) Automatic selection and combination of descriptors for effective 3D similarity search. In: Proc. IEEE international workshop on multimedia content-based analysis and retrieval
23. Bustos B, Keim D, Saupe D, Schreck T, Vranić D (2004) Using entropy impurity for improved 3D object similarity search. In: Proc. IEEE international conference on multimedia and expo
24. Bustos B, Keim DA, Saupe D, Shrec T, Vranic DV (2005) Feature-based similarity search in 3D object databases. ACM Comput Surv 37:345–387
25. Campbell RJ, Flynn PJ (2001) A survey of free-form object representation and recognition techniques. Comput Vis Image Underst 81(2):166–210
26. Cardone A, Gupta SK, Karnik M (2003) A survey of shape similarity assessment algorithms for product design and manufacturing applications. J Comput Inf Science Eng 3:109–118
27. Chen D-Y, Tian X-P, Shen Y-T, Ouhyoung M (2003) On visual similarity based 3D model retrieval. Comput Graph Forum (EG 2003 Proceedings) 22(3):223–232
28. Chua SJ, Jarvis R (1997) Point signatures: a new representation for 3D object recognition. Int J Comput Vis 25(1):63–65
29. Cicirello V, Regli WC (2001) Machining feature-based comparisons of mechanical parts. In: Proc. solid modeling 2001, pp 176–185
30. Cohen I, Ayache N, Sulger P (1992) Tracking points on deformable objects using curvature information. In: Proc. 2nd European conference on computer vision
31. Corney J, Rea H, Clark D, Pritchard J, Breaks M, Macleod R (2002) Coarse filters for shape matching. IEEE Comput Graph Appl 22(3):65–74
32. Cyr CM, Kimia B (2001) 3D Object recognition using shape similiarity-based aspect graph. In: Proc. eighth international conference on computer vision, pp I:254–261
33. DeCarlo D, Metaxas D (1998) Shape evolution with structural and topological changes using blending. IEEE Trans Pattern Anal Mach Intell 20(11):1185–1205
34. Dey TK, Giesen J, Goswami S (2003) Shape segmentation and matching with flow discretization. In: Proc. workshop on algorithms and data structures, pp 25–36
35. El-Mehalawi M, Miller RA (2003) A database system of mechanical components based on geometric and topological similarity. Part I: representation. J Comput-Aided Des 35(1):83–94
36. El-Mehalawi M, Miller RA (2003) A database system of mechanical components based on geometric and topological similarity. Part II: indexing, retrieval, matching and similarity assessment. J Comput-Aided Des 35(1):95–105
37. Elad M, Tal A, Ar S (2001) Content based retrieval of VRML objects - an iterative and interactive approach. In: Proc. eurographics multimedia workshop, pp 97–108
38. Elinson A, Nau DS, Regli WC (1997) Feature-based similarity assessment of solid models. In: Proc. solid modeling '97, pp 297–310
39. Engineering shape benchmark. http://purdue.edu/shapelab
40. Fontana R, Greco M, Materazzi M, Pampaloni E, Pezzati L, Rocchini C, Scopigno R (2002) Three-dimensional modelling of statues: the Minerva of Arezzo. Cultural Heritage Journal 3(4):325–331
41. Funkhouser T, Kazhdan M, Shilane P, Min P, Kiefer W, Tal A, Rusinkiewicz S, Dobkin D (2004) Modeling by example. In: Proc. SIGGRAPH 2004
42. Funkhouser T, Min P, Kazhdan M, Chen J, Halderman A, Dobkin D, Jacobs D (2003) A search engine for 3D models. ACM Trans Graph 22(1):83–105
43. Funkhouser T, Shilane P (2006) Partial matching of 3D shapes with priority-driven search. In: Proceedings eurographics symposium on geometry processing, pp 131–142
44. Gagvani N, Silver D (1999) Parameter controlled volume thinning. Graph Models Image Process 61(3):149–164
45. Garey MR, Johnson DS (1979) Computers and intractability: a guide to the theory of NP-Completeness. Freeman
46. Giannopoulos P, Veltkamp RC (2002) A pseudo-metric for weighted point sets. In: Proc. 7th European conference on computer vision, pp 715–730
47. Hartman J, Wernecke J (1997) The VRML 2.0 handbook—building moving worlds on the web. Addison Wesley Developers Press
48. Hilaga M, Shinagawa Y, Kohmura T (2001) Topology matching for fully automatic similarity estimation of 3D shapes. In: Proc. SIGGRAPH 2001, pp 203–212
49. Hoffmann CM (1989) Geometric & solid modeling, an introduction. Morgan Kaufman
50. Horn B (1984) Extended Gaussian images. Proc of the IEEE 72(12):1671–1686
51. Indyk P, Motwani R (1998) Approximate nearest neighbors: Towards removing the curse of dimensionality. In: Proc. 30th symposium on theory of computing, pp 604–613

52. Ip CY, Lapadat D, Sieger L, Regli WC (2002) Using shape distributions to compare solid models. In: Proc. solid modeling ’02, pp 273–280
53. Ip CY, Sieger L, Regli WC, Shokoufandeh A (2003) Automated learning of model classifications. In: Proc. solid modeling ’03, pp 322–327
54. Iyer N, Janyanti S, Lou K, Kalyanaraman Y, Ramani K (2004) A multi-scale hierarchical 3D shape representation for similar shape retrieval. In: Proc. fifth international symposium on tools and methods of competitive engineering
55. Iyer N, Jayanti S, Ramani K (2005) An engineering shape benchmark for 3D models. In: Proceedings of the ASME international design engineering technical conference & computers and information in engineering conference, pp 24–28
56. Iyer N, Kalyanaraman Y, Lou K, Janyanti S, Ramani K (2003) A reconfigurable 3D engineering shape search system part I: shape representation. In: Proc. ASME design engineering technical conferences, computers and information in engineering conference
57. Iyer N, Lou K, Janyanti S, Kalyanaraman Y, Ramani K (2005) Three dimensional shape searching: state-of-the-art review and future trends. Comput Aided Des 37(5):509–530
58. Johnson AE, Hebert M (1999) Using spin images for efficient object recognition in cluttered 3D scenes. IEEE Trans Pattern Anal Mach Intell 21(5):635–651
59. Kang S, Ikeuchi K (1991) Determining 3-D object pose using the complex extended Gaussian-image. In: Proc. IEEE conference on computer vision and pattern recognition, pp 580–585
60. Kazhdan M, Chazelle B, Dobkin D, Funkhouser T, Rusinkiewicz S (2004) A reflective symmetry descriptor for 3D models. Algorithmica 38(1):201–225
61. Kazhdan M, Funkhouser T, Rusinkiewicz S (2003) Rotation invariant spherical harmonic representation of 3D shape descriptors. In: Proc. Symposium on geometry processing
62. Kazhdan M, Funkhouser T, Rusinkiewicz S (2004) Shape matching and anisotropy. In: Proc. SIGGRAPH 2004
63. Kazhdan M, Funkhouser T, Rusinkiewicz S (2004) Symmetry descriptors and 3D shape matching. In: Proc. eurographics symposium on geometry processing
64. Kazhdan MM (2004) Shape representations and algorithms for 3D model retrieval. PhD thesis, Princeton University
65. Koendering J (1990) Solid shape. The MIT Press
66. Kolonias I, Tzovaras D, Malassiotis S, Srintzis MG (2001) Fast content-based search of VRML models based on shape descriptors. IEEE Trans Multimedia 2:133–136
67. Körtgen M, Park G-J, Novotni M, Klein R (2003) 3D shape matching with 3D shape contexts. In: Proc. 7th central European seminar on computer graphics, April
68. Kriegel H-P, Brecheisen S, Kröger P, Pfeile M, Schubert M (2003) Using sets of feature vectors for similarity search on voxelized CAD objects. In: Proc. SIGMOD 2003
69. Kriegel H-P, Kröger P, Mashael Z, Pfeile M, Pötke M, Seidl T (2003) Effective similarity search on voxelized CAD objects. In: Proc. 8th international conference on database systems for advanced applications
70. Kuo C-T, Cheng S-C (2007) 3D model retrieval using principal plane analysis and dynamic programming. Pattern Recogn 40:742–755
71. Levoy M, Pulli K, Curless B, Rusinkiewicz S, Koller D, Pereira L, Ginzton M, Anderson S, Davis J, Ginsberg J, Shade J, Fulk D (2000) The digital Michelangelo project: 3D scanning of large statues. In: Proc. SIGGRAPH 2000, pp 131–144
72. Liu Y, Pu J, Guyu X, Hongbin Z, Liu W, Uehara Y (2004) A robust method for shape-based 3D model retrieval. In: Proc. Pacific Graphics 2004
73. Liu Y, Pu J, Zha H, Liu W, Uehara Y (2004) Thickness histogram and statistical Harmonic Representation for 3D model retrieval. In: Proc. 2nd international symposium on 3D data processing, visualization, and transmission
74. Löffler J (2000) Content-based retrieval of 3D models in distributed web databases by visual shape information. In: Proc. international conference on information visualisation
75. Loncaric S (1998) A survey of shape analysis techniques. Pattern Recogn 31(8):983–1001
76. Lou K, Janyanti S, Iyer N, Kalyanaraman Y, Prabhakar S, Ramani K (2003) A reconfigurable 3D engineering shape search system part II: database indexing, retrieval and clustering. In: Proc. ASME design engineering technical conferences, computers and information in engineering conference
77. Macrini D, Shokoufandeh A, Dickenson S, Siddiqi K, Zucker S (2002) View-based 3-D object recognition using shock graphs. In: Proc. international conference on attern recognition
78. Mamic G, Bennamoun M (2002) Representation and recognition of 3D free-form objects. Digit Signal Process 12:47–67

79. Marchiori E (1998) A simple heuristic based genetic algorithm for the maximum clique problem. In: Proc. ACM symposium on applied computing, pp 366–373
80. McWherter D, Peabody M, Regli WC, Shokoufandeh A (2001) An approach to indexing databases of graphs. Technical Report DU-MCS-01-03, Drexel University
81. McWherter D, Peabody M, Regli WC, Shokoufandeh A (2001) Transformation invariant shape similarity comparison of solid models. In: Proc. ASME design engineering technical conferences, 6th design for manufacturing conference
82. McWherter D, Peabody M, Shokoufandeh A, Regli WC (2001) Database techniques for archival of solid models. In: Proc. solid modeling '01, pp 78–87
83. Messmer BT, Bunke H (1999) A decision tree approach to graph and subgraph isomorphism detection. Pattern Recogn 32:1979–1998
84. Min P (2004) A 3D model search engine. PhD thesis, Princeton University
85. Min P, Halderman JA, Kazhdan M, Funkhouser TA (2003) Early experiences with a 3D model search engine. In: Proc. web3D symposium, pp 7–18
86. Min P, Kazhdan M, Funkhouser T (2004) A comparison of text and shape matching for retrieval of online 3D models. In: Proc. European conference on digital libraries, pp 209–220
87. MPEG-7 Video Group (2000) Description of core experiments for motion and shape. Technical report, ISO/IEC N3397, MPEG-7
88. National design repository. http://www.designrepository.org/
89. Nehab D, Shilane P (2004) Stratified point sampling of 3D models. In: Proc. eurographics symposium on point-based graphics
90. Novotni M, Klein R (2001) A geometric approach to 3D object comparison. In: Proc. shape modeling international 2001, pp 154–166
91. Novotni M, Klein R (2003) 3D zernike descriptors for content based shape retrieval. In: Proc. solid modeling 2003
92. Ogden IV system. http://www.nime.ac.jp/~motofumi/Ogden/
93. Ohbuchi R, Minamitani T, Takei T (2003) Shape-similarity search of 3D models by using enhanced shape functions. In: Theory and practice of computer graphics 2003, pp 97–104
94. Ohbuchi R, Nakazawa M, Takei T (2003) Retrieving 3D models shapes based on their appearance. In: Proc. 5th ACM SIGMM workshop on multimedia information retrieval (MIR 2003), pp 39–46
95. Ohbuchi R, Otagiri T, Ibato M, Takei T (2002) Shape-similarity search of three-dimensional models using parameterized statistics. In: Proc. Pacific graphics 2002, pp 265–274
96. Ohbuchi R, Takei T (2003) Shape-similarity comparison of 3D models using alpha shapes. In: Proc. Pacific Graphics 2003
97. Osada R, Funkhouser T, Chazelle B, Dobkin D (2002) Shape distributions. ACM Trans Graph 21(4):807–832
98. Paquet E, Murching A, Naveen T, Tababai A, Rioux M (2000) Description of shape information for 2-D and 3-D objects. Signal Process Image Commun 16:103–122
99. Petrou M, Bosdogianni P (1999) Image processing: the fundamentals. John Wiley
100. Protein data bank. http://www.rcsb.org/pdb/
101. Pu J, Liu Y, Guyu X, Hongbin Z, Liu W, Uehara Y (2004) 3D model retrieval based on 2D slice similarity measurements. In: Proc. 2nd international symposium on 3D data processing visualization, and transmission
102. Rea HJ, Corney JR, Clark DER, Taylor NK (2004) A surface partitioning spectrum (SPS) for retrieval and indexing of 3D CAD models. In: Proc. 2nd international symposium on 3D data processing, visualization, and transmission
103. Rea HJ, Sung R, Corney JR, DC (2003) Identifying 3D object features using shape distributions. In: Proc. 18th international conference on computer aided production engineering
104. Regli WC, Cicirello VA (2000) Managing digital libraries for computer-aided design. J Comput-Aided Des 32(2):119–132
105. Ricard J, Couerjolly D, Baskurt A (2004) Generalizations of angular radial transform for 2D and 3D shape retrieval. Technical report, laboratoire LIRIS, Claude Bernard University, Lyon
106. Riesenhuber M, Poggio T (2000) Computational models of object recognition in cortex: a review. Technical report CBCL paper no. 1695, MIT
107. Sánchez-Cruz H, Bribiesca E (2003) A method of optimum transformation of 3D objects used as a measure of shape dissimilarity. Image Vis Comput 21(12):1027–1036
108. Sebastian TB, Klein PN, Kimia BB (2001) Recognition of shapes by editing shock graphs. In: Proc. eighth international conference on computer vision, pp 755–762

109. Sebastian TB, Klein PN, Kimia BB (2002) Shock-based indexing into large shape databases. In: Proc. 7th European Conference on computer vision, pp 731–746
110. Shamir A, Sotzio A, Cohen-Or D (2004) Enhanced hierarchical shape matching for shape transformation. Int J Shape Model IJSM 9(2):203–222
111. ShapeSifter. http://www.shapesearch.net/index.html
112. Shen Y-T, Chen D-Y, Tian X-P, Ouhyoung M (2003) 3D model search engine based on lightfield descriptors. In: Proc. eurographics 2003
113. Shilane P, Kazhdan M, Min P, Funkhouser T (2004) The princeton shape benchmark. In: Proc. shape modeling international 2004, pp 157–166
114. SHREC—Shape retrieval contest. http://www.aim-at-shape.net/
115. Shum H-Y, Hebert M, Ikeuchi K (1996) On 3D shape similarity. In: Proc. IEEE computer vision and pattern recognition, pp 526–531
116. SIDESTEP—Shape-based Image Delivery Statistics Evaluation Project. http://give-lab.cs.uu.nl/sidestep/
117. Sundar H, Silver D, Gagvani N, Dickenson S (2003) Skeleton based shape matching and retrieval. In: Proc. shape modeling international 2004, pp 130–139
118. Sung R, Rea HJ, Corney JR, Clark DER, Pritchard J, Breaks ML, Macleod RA (2002) Shape-Sifter: a retrieval system for databases of 3D engineering data. New Rev Inf Netw 8:33–54
119. Suzuki MT (2001) A search engine for polygonal models to support development of 3D e-learning applications. In: The tenth international world wide web conference poster proceedings, pp 182–183
120. Tangelder JWH, Veltkamp RC (2003) Polyhedral model retrieval using weighted point sets. Int J Image Graphics 3(1):209–229
121. Terzopoulus D, Metaxas D (1991) Dynamic models with local and global deformations: deformable superquadrics. IEEE Trans Pattern Anal Mach Intell 13(7):703–714
122. Tung T, Schmitt F (2004) Augmented reeb graphs for content-based retrieval of 3D mesh models. In: Proc. shape modeling international 2004, pp 157–166
123. Tversky A (1977) Features of similarity. Psychol Rev 84(4):327–352
124. Veltkamp RC, Hagedoorn M (2001) State-of-the-art in shape matching. In: Lew M (ed) Principles of visual information retrieval, pp 87–119
125. Vleugels J, Veltkamp RC (2002) Efficient image retrieval through vantage objects. Pattern Recogn 35(1):69–80
126. Vranić DV (2003) An improvement of rotation invariant 3D shape descriptor based on functions on concentric spheres. In: Proc. IEEE international conference on image processing
127. Vranić DV (2004) 3D model retrieval. PhD thesis, University of Leipzig
128. Vranić DV, Saupe D (2000) 3D model retrieval. In: Proc. spring conference on computer graphics and its applications, pp 89–93, May
129. Vranić DV, Saupe D, Richter J (2001) Tools for 3D-object retrieval: Karhunen–Loeve-transform and spherical harmonics. In: Proc. IEEE 2001 workshop on multimedia signal processing
130. Yu M, Atmosukarto I, Leow WK, Huang Z, Xu R (2003) 3D model retrieval with morphing-based geometric and topological feature maps. In: Proc. IEEE conf. on computer vision and pattern recognition, pp II: 656–661
131. Zaharia T, Prêteux F (2001) 3D shape-based retrieval within the MPEG-7 framework. In: Proc. SPIE conference 4304, pp 133–145
132. Zhang C, Chen T (2001) Efficient feature extraction for 2D/3D objects in mesh representation. In: IEEE international conference on image processing
133. Zhang, C., Chen T: (2001) Indexing and retrieval of 3D models aided by active learning. In: Proc. ACM Multimedia, pp 615–616
134. Zhang DS, Lu G (2002) Shape-based image retrieval using generic Fourier descriptor. Signal Process Image Commun 17(10):825–848
135. Zhang K, Whang JTL, Shasha D (1996) On the editing distance between undirected acyclic graphs. Int J Foundations Comp Science 7(1):43–57
136. Zuckerberger E, Tal A, Shlafman S (2002) Polyhedral surface decomposition with applications. Comp Graphics 26(5):733–743

Johan W. H. Tangelder obtained his Ph.D. in 1998 from Delft University of Technology on the Sculpturing Robot project at Faculty of Industrial Design Engineering. After joining the Faculty of Geodetic Engineering at the same university, he joined the Multimedia Group at Utrecht University, working on 3D shape retrieval, 3D scanning and approximate spatial searching. He is now affiliated with Prime Vision.

Remco C. Veltkamp is associate professor at Utrecht University, and principal investigator of the Multimedia Group. His current research focuses on the algorithmic aspects of multimedia retrieval, like the algorithmic design and analysis, and experimental verification. He is editor of the International Journal on Shape Modelling, the journal Pattern Recognition, and is organiser of the Dagstuhl Seminars series on Content-Based Retrieval.