Shape Retrieval using 3D Zernike Descriptors

Marcin Novotni Reinhard Klein

University of Bonn,
Institute of Computer Science II.
Römerstr. 164, D-53117 Bonn, Germany
{marcin,rk}@cs.uni-bonn.de

Abstract

We advocate the usage of 3D Zernike invariants as descriptors for 3D shape retrieval. The basis polynomials of this representation facilitate computation of invariants under rotation, translation and scaling. Some theoretical results have already been summarized in the past from the aspect of pattern recognition and shape analysis. We provide practical analysis of these invariants along with algorithms and computational details. Furthermore, we give a detailed discussion on influence of the algorithm parameters like the conversion into a volumetric function, number of utilized coefficients, etc. As is revealed by our study, the 3D Zernike descriptors are natural extensions of recently introduced spherical harmonics based descriptors. We conduct a comparison of 3D Zernike descriptors against these regarding computational aspects and shape retrieval performance using several quality measures and based on experiments on the Princeton Shape Benchmark.

1 Introduction

It can be observed that the proliferation of a specific digital multimedia data type (e.g. text, images, sounds, video) was followed by emergence of systems facilitating their retrieval. With the recent advances in 3D data acquisition techniques, graphics hardware and modeling methods, there is an increasing amount of 3D objects spread over various archives: general objects commonly used e.g. in games or VR environments, etc. On the other hand, modeling of high fidelity 3D objects is a very cost and time intensive process – a task which one can potentially get around by reusing already available models. Another important issue is the efficient exploration of scientific data represented as 3D entities. Such archives are becoming increasingly popular in the areas of Biology, Chemistry, Anthropology and Archeology to name a few. Therefore, since recently, concentrated research efforts are being spent on elaborating techniques for efficient retrieval of 3D objects.

It has to be mentioned that in the last three decades hundreds of millions of 3D CAD data was created by the industry, and there is a large body of work to elaborate retrieval systems for such data. The method described in this paper is a result of efforts going in a slightly different direction. The ultimate goal of the presented work is to be able to

retrieve 3D models from repositories containing 3D models of general categories (e.g. cars, airplanes, humans, etc.) that can be used for instance in computer games or VR environments. This imposes a notion of similarity that is hard to define precisely a priori, as there is no single classification of objects that would satisfy every query. In fact, even inconsistent classifications are conceivable, e.g. one classifying cars by brands, i.e. BMW, Mercedes, etc. and the other classifying by type, i.e. limousine, SUV, race car, etc. It is important to note that for the domains where the object similarity is better defined and the representation more uniform – as in case of most of CAD, chemistry or archaeological databases – more specialized and powerful descriptors and matching methods may presumably be built, possibly by extending the descriptors presented in this paper.

One of major the challenges in the context of data retrieval is to elaborate a suitable canonical characterization of the entities to be indexed. In the following, we will refer to this characterization as a descriptor. Since the descriptor serves as a key for the search process, it decisively influences the performance of the search engine in terms of computational efficiency and relevance of the results. A simple approach is to annotate the entities with keywords, however, due to the inherent complexity and multitude of possible interpretations this proved to be incomplete, insufficient and/or impractical for almost all data types, cf. [34, 16].

Guided by the fact that for a vast class of objects the shape constitutes a large portion of abstract object information, we focus in this paper on general shape based object descriptors. We now can state some requirements that a general shape based descriptor should obey:

1. **Descriptive power** - the similarity measure based on the descriptor should deliver a similarity ordering that is close to the application driven notion of resemblance.

2. **Conciseness and ease of indexing** - the descriptor should be compact in order to minimize the storage requirements and accelerate the search by reducing the dimensionality of the problem. Very importantly, it should provide some means of indexing and thereby structuring the database in order to further accelerate the search process.

3. **Invariance under transformations** - the computed descriptor values have to be invariant under an application dependent set of transformations. Usually, these are the similarity transformations, however, some applications like e.g. retrieval of articulated objects may additionally demand invariance under certain deformations, etc.

A list of additional requirements may be given, which apply in case of search for general 3D objects that may be found in various archives on the World Wide Web: insensitiveness to noise and small extra features, independence of 3D object representation, tessellation, or genus, robustness against arbitrary topological degeneracies.

In this paper we present a 3D shape retrieval method relying on 3D Zernike moments. These moments are computed as a projection of the function defining the object

onto a set of orthonormal functions within the unit ball – the 3D Zernike polynomials introduced by Canterakis [10]. From these Canterakis has derived affine invariant features of 3D objects represented by a volumetric function. To our knowledge, these results have not been applied to the retrieval of 3D objects so far. Apart from providing practical details on this technique, we perform a comparison with previous description methods.

To this end we apply our method to the Princeton Shape Benchmark database [5] of general polygonal models, e.g. chairs, cars, airplanes, etc. collected from the WWW. We expect this recently created database to become a standard benchmark for shape descriptors like those presented in this paper.

We compared the retrieval results with those yielded by the recently introduced spherical harmonic descriptors (SH descriptors) [16], which are reported to be among the most powerful at present with respect to general object classes similar to those mentioned above. As it turns out, the construction of SH and 3D Zernike descriptors is closely related, which enables a detailed comparison regarding the structure and performance.

To summarize, the contribution of the paper is threefold: i) we attempt to present the previous theoretical results of Canterakis [10] on 3D Zernike moments in a self-contained and accessible way; ii) we present algorithmic solutions leading to a practical implementation of descriptors and iii) we describe experimental results on performance of the 3D Zernike descriptors based on the Princeton Shape Benchmark, and finally we give details on comparison with the SH descriptors.

The outline of the rest of the paper is as follows: in the next section we shortly review the relevant previous work. In Section 3 we present a general theoretical framework for the computation of rotationally invariant descriptors and delineate the 3D Zernike descriptors in this framework. We also examine the 3D Zernike descriptors for accordance with the above criteria and 3D shape retrieval performance. Section 4 gives a discussion on practical issues concerning the implementation of 3D Zernike descriptors paying special attention to numerical stability of computations. In Section 5 we present our results and conclude in Section 6.

2 Previous Work

2.1 Systems

To date numerous systems for 2D image retrieval have been introduced. To gain a good overview over the state-of-the-art in this area we refer to the survey papers [34, 17, 31]. As for retrieval of general 3D objects described in the previous section, the first system was introduced in [29], which was followed by [35]. A very recent result is presented in [16, 26]. Considering systems covering narrower domains, [1] deals with anthropological data, [3, 12] facilitate the retrieval of industrial solid models, [6] explores protein databases. It should be noticed that for the narrower domains where the notion of similarity is better specified usually other methods for descriptor construction apply than for broad domains (see [34]). Thus, often better performance is achieved in systems aiming at narrow domains.

2.2 Descriptors

In our classification of shape description techniques we adopt a grouping similar to that of [24] into space domain and scalar transform methods.

2.2.1 Space domain

The space domain shape analysis methods yield non-numeric results, usually an attributed graph, which encodes the spatial and/or topological structure of an object. Notably, in his seminal work Blum introduced the Medial Axis Transform (MAT) [8], which was followed by a number of extensions like shock graphs, see e.g. [33], shock scaffold [23], etc. Forsyth et al. [15] represent 2D image objects by spatial relationships between stylized primitives, [30] uses a similar approach. A further technique having a long tradition is the geon based representation [7]. As for 3D industrial solid models, [11, 25] capture geometric and engineering features in a graph, which is subsequently used for similarity estimation. Tangelder and Veltkamp [37] describe an approach representing the polyhedral objects as weighted point sets. Hilaga et al. [18] presented a method for general 3D objects utilizing Reeb graphs based on geodesic distances between points on the mesh, which enabled a deformation invariant recognition. The methods in this class are attractive since they capture a high level structure of objects and provide rich descriptors. Unfortunately though, they are computationally expensive, the MAT based techniques suffer from sensitivity to small perturbations of the object boundary. Furthermore, the underlying graph representation that is typical for space domain methods makes the indexing and comparison of objects difficult.

2.2.2 Scalar transform

The scalar transform techniques capture global properties of the objects generating numbers (scalars or vectors) as shape descriptors.

Histograms A number of methods in 2D rely on color histograms measuring the color distribution in an image [36]. In [29] this is generalized to constructing 3D histograms of normal vector, color, material, etc. distributions. Ankerst et al. [6] subdivide the space into spherical shells and sectors around the center of gravity of an object, the resulting partitions correspond to the bins of the 3D shape histogram. Kazhdan et al. describe a reflective symmetry descriptor in [21], Osada et al. [28] compute histograms based on geometric statistics of the boundary of 3D objects. Unfortunately, these descriptors usually provide an insufficient discrimination between objects.

Projection based techniques Some techniques both in 2D and in 3D are based on coefficients yielded by compression transforms like the cosine [32] or wavelet transform e.g. in [20]. Fourier descriptors [42] have been applied in 2D, however, these are hard to generalize to 3D due to the difficulties in parameterizing 3D object boundaries.

Moments can generally be defined as projections of the function defining the object onto a set of functions characteristic to the given moment. Since Hu [19] popularized the usage of image moments in 2D pattern recognition, they have found numerous

applications. Teague [38] was first to suggest the usage of orthogonal functions to construct moments. Subsequently, several 2D moments have been elaborated and evaluated [39]: geometrical, Legendre, Fourier-Mellin, Zernike, pseudo-Zenike moments. For 3D objects geometrical moments have been used in [14, 27], and a spherical harmonic decomposition was used by Vranic and Saupe [41]. The main drawback of these methods is that prior to computations a canonical pose of objects has to be determined, which often proves to be unstable, as discussed in [16]. Funkhouser et al. [16] profit from the invariance properties of spherical harmonics and present a translation and rotation invariant descriptor. The main idea behind this is to decompose the 3D space into concentric spherical shells and compute rotationally invariant representations of these subspaces. In this way a descriptor was constructed which was experimentally shown to be superior over other 3D techniques with regard to shape retrieval performance. In [39], 2D Zernike moments were found to be superior over others in terms of noise sensitivity, information redundancy and discrimination power. Guided by this, Canterakis [10] generalized the classical 2D Zernike polynomials to 3D, however, in his work Canterakis considered mostly theoretical aspects.

3 3D Zernike Moments and Descriptors

This section gives a systematic construction of 3D Zernike moments and descriptors. We attempt to describe a framework providing a general approach to construct rotation invariant descriptors. We also recall the relevant results of Canterakis [10] and describe our improvements.

3.1 Moments

Moments in the context of shape analysis are defined as projections of the (square integrable) object function $f \in L^2$ onto a set of functions $\Psi = \{\psi_i\},\ i \in \mathbb{N}$ over the domain Ω. The projection is computed as a dot product defined on the Hilbert space of finite energy functions L^2:

$$\mu_i = \langle f, \psi_i \rangle.$$

The behavior and properties of a particular moment based representation are therefore determined by the set of functions Ψ.

We now consider the desirable properties of a descriptor based on moments and subsequently give a general formula for computation of moments obeying these properties for the two and three dimensional case.

1. **Invariance.** Let $\mathscr{F}(f)$ be a set of descriptors computed on the function f defining the object, and let G be a group of transformations. The invariance of $\mathscr{F}$ under the action of G can be defined as follows:

   $$\mathscr{F}(gf) = \mathscr{F}(f),$$

   where $g \in G$. A typical requirement is the invariance under the action of similarity transformations, i.e. uniform scaling, reflection, translation and rotation.

2. **Orthonormality.** The collection of functions Ψ is orthonormal, if

$$\langle \psi_i, \psi_j \rangle = \delta_{ij},$$

where $\psi_i, \psi_j \in \Psi$ and δ_{ij} is the Kronecker delta.

3. **Completeness.** The set of functions Ψ forms a complete system if for any $f \in L^2$,

$$\lim_{n\to\infty} ||f - \sum_{i=0}^{n} \langle f, \psi_i \rangle \psi_i||^2 = 0,$$

where $||.||$ denotes the L_2-norm. Complete orthonormal function collections are said to form a basis of the function space on the domain Ω.

Concerning the invariance, most approaches transform the object into a canonical pose: translate the center of gravity of the object into the origin and normalize the area/volume or radius of the bounding circle/sphere. The rotation invariance may subsequently be achieved by aligning the principal axes of the object with the coordinate system axes. However, as has been investigated by [16], this last step is often unreliable and leads to reduced retrieval performance. Based on these observations, in our choice of Ψ we will favor representations yielding a more stable rotation invariance.

The orthogonality of our function collection, i.e. the mutual independence of computed features is an important property, since it implies that a set of features will not contain redundant information. The non-orthogonality (as in the case of geometrical moments based on monomials) means that some characteristics of the objects will be over-represented during the comparison. The classical 2D Zernike polynomials are orthonormal within the unit circle. They therefore deliver independent features, and are shown to be superior over the geometrical moments in terms of retrieval performance. The additional normalization is essentially a convenience criterion, as this property allows for a canonical formulation of projections of functions.

The completeness property implies that we are able to reconstruct approximations of the original object from moments. The approximations are getting finer with increasing number n of moments and converge to the original object at infinity. This is of considerable practical importance, since the ability to reconstruct allows us to infer a higher bound on the amount of object information encoded by a given number of moments. Another related practical property that may be inferred from reconstructions is the multiscale nature of the descriptors allowing an efficient coarse-to-fine search. See Section 5.3 for a reconstruction experiment.

3.2 Selection of basis functions

To sum up, we are looking for sets of functions forming complete orthogonal systems and allowing for construction of moments that are invariant under rotation transformations. As it turns out, a straightforward solution is essentially a tensor product formulation and consists of two ingredients. First, one has to choose an angular function set $\{S_l(\varphi)\}$ or $\{S_l^m(\varphi, \vartheta)\}$ defined on the unit circle or sphere, respectively, that is orthogonal and has subspaces invariant under the action of the rotation group. Second, the

circular or spherical function is modulated by a suitable radial function $R_{nl}^m(r)$ while maintaining the orthonormality. Note that in general a particular function R may depend on indices l and m, which implies a dependency on the angular function. The radial polynomials of both the classical 2D and the new 3D Zernike functions depend on n and l.

Let B^2 and B^3 denote the unit disc and unit ball, respectively. The general formulas in polar and spherical coordinates for the generation of moments μ possessing the above properties are:

$$\mu_{ln} = \langle f, R_{nl} S_l \rangle_{B^2} = \int_0^1 \int_0^{2\pi} f(r,\varphi) \overline{R_{nl}(r) S_l(\varphi)} r d\varphi dr$$

and

$$\mu_{ln}^m = \langle f, R_{nl}^m S_l^m \rangle_{B^3} = \int_0^1 \int_0^{\pi} \int_0^{2\pi} f(r,\varphi,\vartheta) \overline{R_{nl}^m(r) S_l^m(\varphi,\vartheta)} \sin(\vartheta) d\vartheta d\varphi dr$$

for the two and three dimensional case, respectively. The choice of an appropriate angular function seems to be crucial, therefore we first summarize some observations that have been made in the 2D case and then move on to 3D.

3.2.1 2D Zernike moments

In 2D, a suitable angular function has proven to be:

$$S_l(\varphi) = e^{il\varphi}, \tag{1}$$

which is essentially the familiar Fourier basis function. It has been shown e.g. by Khotanzad and Hong [22] that for such functions the following relation applies:

$$|\langle f(\varphi + \varphi_0), e^{il\varphi} \rangle_{S^1}| = |\langle f(\varphi), e^{il\varphi} \rangle_{S^1}|.$$

This implies that by projecting a function f defined on the circle onto a basis of above functions (Eqn. 1), and computing the norms of these projections, we obtain descriptors of f that are invariant under the action of 2D rotations. The radial polynomial R_{nl} for the 2D Zernike functions $Z_{nl}(r,\varphi) = R_{nl}(r) e^{il\varphi}$ is defined so that the resulting basis Z_{nl} is orthonormal, refer e.g. to [22] for exact definition of the radial polynomials.

3.2.2 3D Zernike moments

Using the general construction rule derived above, we now derive the 3D Zernike moments (see [10] and [9] for details).

Spherical harmonics Motivated by the facts summarized in the previous subsection and recalling that spherical harmonics on the sphere have properties similar to the functions of Eqn. 1, we continue with the description of spherical harmonics.

Spherical harmonics form a Fourier basis on a sphere much like the familiar sines and cosines do on a line or a circle. Spherical harmonics Y_l^m are given by:

$$Y_l^m(\vartheta,\varphi) = N_l^m P_l^m(\cos\vartheta) e^{im\varphi},$$

where N_l^m is a normalization factor

$$N_l^m = \sqrt{\frac{2l+1}{4\pi}\frac{(l-m)!}{(l+m)!}},$$

and P_l^m denotes the associated Legendre functions.

Invariance properties The vector of spherical harmonics

$$\mathbf{Y}_l = (Y_l^l, Y_l^{l-1}, Y_l^{l-2}, \ldots, Y_l^{-l})^t \tag{2}$$

for a given l forms the basis for a $(2l+1)$-dimensional subspace which is invariant under the operations of the full rotation group[1]. This can be formulated as

$$\mathbf{Y}_l(\vartheta+\vartheta_0, \varphi+\varphi_0) = \mathbf{o}_l(\vartheta_0, \varphi_0)\mathbf{Y}_l(\vartheta, \varphi), \tag{3}$$

where $\mathbf{o}_l$ is a unitary matrix referred to as l-th representation of the three dimensional rotation group $SO(3)$. Furthermore, this subspace is irreducible that is, it cannot be split into smaller subspaces which are also invariant under the rotation group. Since rotations do not change the norm of functions, in consequence of Eqn. 3, after projecting a function f defined on the unit sphere S^2 onto the functions of the vector $\mathbf{Y}_l$, we obtain invariant features μ_l of f by computing the norms of the so computed vectors:

$$\mu_l = \left\| \begin{array}{c} \langle f, Y_l^l(\vartheta+\vartheta_0, \varphi+\varphi_0)\rangle_{S^2} \\ \langle f, Y_l^{l-1}(\vartheta+\vartheta_0, \varphi+\varphi_0)\rangle_{S^2} \\ \vdots \\ \langle f, Y_l^{l-1}(\vartheta+\vartheta_0, \varphi+\varphi_0)\rangle_{S^2} \end{array} \right\| = \left\| \begin{array}{c} \langle f, Y_l^l(\vartheta, \varphi)\rangle_{S^2} \\ \langle f, Y_l^{l-1}(\vartheta, \varphi)\rangle_{S^2} \\ \vdots \\ \langle f, Y_l^{l-1}(\vartheta, \varphi)\rangle_{S^2} \end{array} \right\| \tag{4}$$

As a next step, we have to augment this representation to cover the three dimensional space.

Harmonic polynomials Canterakis based his derivations on harmonic polynomials which finally enabled him to formulate the 3D Zernike polynomials as homogenous polynomials in the Cartesian coordinates x, y and z.

Let us define the conversion between Cartesian and spherical coordinates by $\mathbf{x} = |\mathbf{x}|\xi = r\xi = r(\sin\vartheta\cos\varphi, \sin\vartheta\sin\varphi, \cos\varphi)^T$. The harmonic polynomials e_l^m are defined as

$$e_l^m(\mathbf{x}) = r^l Y_l^m(\vartheta, \varphi).$$

Using the integral formula for associated Legendre functions [13] and converting into Cartesian coordinates, we can express the harmonic polynomials as

$$\begin{aligned} e_l^m(\mathbf{x}) &= c_l^m r^l \left(\tfrac{ix-y}{2}\right)^m z^{l-m} \\ &\quad \cdot \Sigma_{\mu=0}^{\lfloor \frac{l-m}{2} \rfloor} \left(\begin{array}{c} l \\ \mu \end{array} \right) \left(\begin{array}{c} l-\mu \\ m+\mu \end{array} \right) \left(-\tfrac{x^2+y^2}{4z^2}\right)^\mu, \end{aligned} \tag{5}$$

[1] A set $\{\psi_i\}$ of vectors is said to span an invariant subspace V_s under a given set of group operations $\{g_j\}$ if $g_j\psi_i \in V_s \ \forall i, j$.

where c_l^m are normalization factors:

$$c_l^m = c_l^{-m} = \frac{\sqrt{(2l+1)(l+m)!(l-m)!}}{l!}.$$

The above formula yields homogenous polynomials for $m > 0$. For $m < 0$ the following symmetry relation is used:

$$e_l^{-m}(\mathbf{x}) = (-1)^m \overline{e_l^m(\mathbf{x})}, \tag{6}$$

which yields homogenous polynomials in this case as well. It is easy to see that an invariance relation similar to that of Eqn. 4 applies for the harmonic polynomial.

Derivation of 3D Zernike moments The 3D Zernike functions Z_{nl}^m are defined as

$$Z_{nl}^m(\mathbf{x}) = R_{nl}(r) \cdot Y_l^m(\vartheta, \phi)$$

while restricting l so that $l \leq n$ and $(n-l)$ be an even number. The above equation can be rewritten in Cartesian coordinates using the harmonic polynomials e_l^m:

$$Z_{nl}^m(\mathbf{x}) = \sum_{\nu=0}^{k} q_{kl}^{\nu} |\mathbf{x}|^{2\nu} e_l^m(\mathbf{x}), \tag{7}$$

where $2k = n-l$ and the coefficients q_{kl}^{ν} are determined to guarantee the orthonormality of the functions in the unit ball:

$$q_{kl}^{\nu} = \frac{(-1)^k}{2^{2k}} \sqrt{\frac{2l+4k+3}{3}} \binom{2k}{k} (-1)^{\nu} \cdot \frac{\binom{k}{\nu} \binom{2(k+l+\nu)+1}{2k}}{\binom{k+l+\nu}{k}}.$$

The radial polynomials used to generate Eqn. 7 expressed explicitly are of the following form:

$$R_{nl}(r) = r^l \sum_{\nu=0}^{k} q_{kl}^{\nu} r^{2\nu}.$$

The orthonormality relation reads as follows:

$$\frac{3}{4\pi} \int_{||\mathbf{x}|| \leq 1} Z_{nl}^m(\mathbf{x}) \cdot \overline{Z_{n'l'}^{m'}(\mathbf{x})} d\mathbf{x} = \delta_{nn'} \delta_{ll'} \delta^{mm'}$$

In case of the 3D Zernike functions the same invariance relation applies as in case of spherical harmonics. If we collect the functions into $(2l+1)$-dimensional vectors $\mathbf{Z}_{nl} = (Z_{nl}^l, Z_{nl}^{l-1}, Z_{nl}^{l-2}, \ldots, Z_{nl}^{-l})^t$ for each l, for an arbitrary rotation $\mathbf{P}$ we obtain the relation

$$\mathbf{Z}_{nl}(\mathbf{Px}) = \mathbf{o}_l(\mathbf{P}) \mathbf{Z}_{nl}(\mathbf{x}). \tag{8}$$

We are now able to define the 3D Zernike moments Ω_{nl}^m of an object defined by f as

$$\Omega_{nl}^m := \frac{3}{4\pi} \int_{|\mathbf{x}|\leq 1} f(\mathbf{x}) \overline{Z_{nl}^m(\mathbf{x})} d\mathbf{x}.$$

It is worthwhile noting that due to the symmetry relation of Eqn. 6, a similar relation holds for the Zernike moments:

$$\Omega_{nl}^{-m}(\mathbf{x}) = (-1)^m \overline{\Omega_{nl}^m(\mathbf{x})}. \tag{9}$$

It is important to notice that the 3D Zernike moments Ω_{nl}^m are not invariant under rotations. In order to achieve invariance, we apply the approach followed in case of spherical harmonics (cf. Eqn. 4): we collect the moments into $(2l+1)$-dimensional vectors $\Omega_{nl} = (\Omega_{nl}^l, \Omega_{nl}^{l-1}, \Omega_{nl}^{l-2}, \ldots, \Omega_{nl}^{-l})^t$ and define the rotationally invariant 3D Zernike descriptors F_{nl} as norms of vectors Ω_{nl}:

$$F_{nl} := ||\Omega_{nl}||. \tag{10}$$

Reconstruction Since the functions Z_{nl}^m form a complete orthonormal system, it is possible to approximate the original function f by a finite number of 3D Zernike moments Ω_{nl}^m:

$$\hat{f}(\mathbf{x}) = \sum_n \sum_l \sum_m \Omega_{nl}^m \cdot Z_{nl}^m(\mathbf{x}). \tag{11}$$

Here, we sum over $n \in [0,N]$, $l \in [0,n]$ such that $(n-l)$ be an even number and $m \in [-l,l]$. We use the reconstruction to verify how much of the original object information is included in a set of 3D Zernike moments up to a given order $n = N$.

4 Computation of 3D Zernike descriptors

We now consider the computational details. First, let us expand Z_{nl}^m of Eqn. 7 using Eqn. 5:

$$\begin{aligned} Z_{nl}^m(\mathbf{x}) \quad = \quad & c_l^m 2^{-m} \sum_{\nu=0}^{k} q_{kl}^{\nu} \\ & \cdot \sum_{\alpha=0}^{\nu} \binom{\nu}{\alpha} \sum_{\beta=0}^{\nu-\alpha} \binom{\nu-\alpha}{\beta} \\ & \cdot \sum_{u=0}^{m} (-1)^{m-u} \binom{m}{u} i^u \\ & \cdot \sum_{\mu=0}^{\lfloor \frac{l-m}{2} \rfloor} (-1)^{\mu} 2^{-2\mu} \binom{l}{\mu} \binom{l-\mu}{m+\mu} \\ & \cdot \sum_{\nu=0}^{\mu} \binom{\mu}{\nu} \end{aligned}$$

$$\cdot x^{2(\nu+\alpha)+u}$$
$$\cdot y^{2(\mu-\nu+\beta)+m-u}$$
$$\cdot z^{2(\nu-\alpha-\beta-\mu)+l-m}.$$

Substituting $r = 2(\nu+\alpha)+u$, $s = 2(\mu-\nu+\beta)+m-u$ and $t = 2(\nu-\alpha-\beta-\mu)+l-m$ and setting

$$\begin{aligned}
\chi_{nlm}^{rst} &= c_l^m 2^{-m} \sum_{\nu=0}^{k} q_{kl}^{\nu} \\
&\quad \cdot \sum_{\alpha=0}^{\nu} \binom{\nu}{\alpha} \sum_{\beta=0}^{\nu-\alpha} \binom{\nu-\alpha}{\beta} \\
&\quad \cdot \sum_{u=0}^{m} (-1)^{m-u} \binom{m}{u} i^u \\
&\quad \cdot \sum_{\mu=0}^{\lfloor \frac{l-m}{2} \rfloor} (-1)^{\mu} 2^{-2\mu} \binom{l}{\mu} \binom{l-\mu}{m+\mu} \cdot \sum_{\nu=0}^{\mu} \binom{\mu}{\nu},
\end{aligned}$$

Z_{nl}^m can be written in a more compact form as a linear combination of monomials of order up to n

$$Z_{nl}^m(\mathbf{x}) = \sum_{r+s+t \le n} \chi_{nlm}^{rst} \cdot x^r y^s z^t \tag{12}$$

Finally, let us observe that using Eqn. 12, the 3D Zernike moments Ω_{nl}^m of an object can be written as a linear combination of geometrical moments of order up to n:

$$\Omega_{nl}^m = \frac{3}{4\pi} \sum_{r+s+t \le n} \overline{\chi_{nlm}^{rst}} M_{rst}, \tag{13}$$

where M_{rst} denotes the geometrical moment of the object scaled to fit in the unit ball:

$$M_{rst} := \int_{|\mathbf{x}| \le 1} f(\mathbf{x})\, x^r y^s z^t d\mathbf{x}, \tag{14}$$

where $\mathbf{x} \in \mathbb{R}^3$ denotes the vector $\mathbf{x} = (x,y,z)^t$. An important fact implied by Eqn. 13 is that in order to compute the 3D Zernike functions we only have to compute the geometrical moments instead of evaluating the complex exponential and associated Legendre function of spherical harmonics. Further important implication is that using such formulation we may perform the computations on a rectangular grid instead of having to resample the volumetric object function to get a spherical grid.

4.1 Algorithm

The above observations lead to the following algorithm to compute the 3D Zernike descriptors F_{nl}^m. The computations have to be conducted for all n,l,m index combinations for $n \in [0,N]$, $l \in [0,n]$ such that $(n-l)$ be an even number and $m \in [-l,l]$.

The values χ_{nlm}^{rst} for $r+s+t \le n$ have to be determined before starting the algorithm. Note that this step is independent of a particular object and may be done offline. Since

for an (n,l,m) triple, there will typically be a lot of zero coefficients, we store the values of these together with the indices r,s,t indexing the corresponding geometrical moment in a list $List_{\chi}^{nlm}$.

We now give the steps needed to compute the 3D Zernike moments and descriptors:

1. **Normalization.** Compute the center of gravity of the object, transform it to the origin, and scale the object so that it will be mapped into the unit ball.

2. **Geometrical moments.** Compute all geometrical moments M_{rst} for each combination of indices, such that $r,s,t \geq 0$ and $r+s+t \leq N$. Refer to the next subsection for details on this computation.

3. **3D Zernike moments.** Compute all Zernike moments Ω_{nl}^{m} according to Eqn. 13. Note that the summation has to be conducted only for the nonzero coefficients χ_{nlm}^{rst} stored in the list $List_{\chi}^{nlm}$. Also note that for $m \leq 0$, Ω_{nl}^{m} may be computed using the symmetry relation of Eqn. 9.

4. **3D Zernike descriptors.** Compute all F_{nl} according to Eqn. 10.

4.2 Geometrical moments

The computation of the geometrical moments is of central importance with respect to the overall computational efficiency end numerical accuracy of our method.

When working with 2D images or 3D voxel grids, it is usually tempting to use integral coordinates at grid points and sample the functions accordingly at these point. Thus, a typical approach to compute the geometrical moments of an object represented by a 2D or 3D image is the following:

1. Fix a coordinate system with origin at a corner of the grid and axes aligned with the grid axes. Subsequently, sample all monomials of order up to P at the grid point positions.

2. Compute the geometrical moments according to Eqn. 14 by summing over the whole voxel grid. In 3D this may be formulated as follows:

$$M_{rst} = \sum_{i=0}^{L-1} \sum_{i=0}^{M-1} \sum_{i=0}^{N-1} i^r j^s k^t f(i,j,k),$$

   where $L,M,N \in \mathbb{N}$ denote the grid size in respective dimensions

3. Transform the geometrical moments according to the normalization transformation of the object. This can easily be accomplished, since scaling can be achieved by scaling the moments, the moments of the translated object can be represented in terms of a linear combination of original moments of not greater order.

The first two steps introduce numerical problems. First, the sampling at grid points implies that we treat the monomial as a function having a constant value within a voxel, which is determined by the value of the monomial e.g. in the center of the voxel. For rapidly changing functions, like the monomials of high order, this results in inaccuracy.

Second, for a 64^3 grid for instance, the precision of the double precision floating point number is exceeded already at the order of 9. According to our experience, moments up to order of 20 are needed to provide a good descriptor.

We treat the first issue by computing the geometrical moments in terms of monomials integrated over the voxels. Since for high orders the 3D Zernike descriptors seem to discard the values of voxels close to the origin, we normalize the object prior to computation of moments, thus obtaining considerably better numerical accuracy and providing a cure to the second problem. These procedures are described in the remainder of this section.

4.2.1 Integration.

Let us first consider the 1D case. The function f is sampled at the sample points $\{x_i\}$, $0 \le i \le N-1$. We treat f as having constant values f_i within intervals $[x_i, x_{i+1})$:

$$\begin{aligned} M_p &= \int f(\alpha)\alpha^p \, d\alpha \\ &= \sum_{i=0}^{N-1} f_i \int_{x_i}^{x_{i+1}} \alpha^p d\alpha \\ &= \sum_{i=0}^{N-1} f_i \frac{x_{i+1}^{p+1} - x_i^{p+1}}{p+1}. \end{aligned}$$

The computation of geometrical moments of order p for $0 \le p \le P$ can be formulated in matrix form:

$$\underbrace{\begin{bmatrix} M_0 \\ M_1 \\ \vdots \\ M_P \end{bmatrix}}_{\mathbf{M}} = \begin{bmatrix} \frac{1}{2} \\ \frac{1}{3} \\ \vdots \\ \frac{1}{P+1} \end{bmatrix} \underbrace{\begin{bmatrix} x_0 & x_1 & \cdots & x_N \\ x_0^2 & x_1^2 & \cdots & x_N^2 \\ \vdots & \vdots & \ddots & \vdots \\ x_0^P & x_1^P & \cdots & x_N^P \end{bmatrix}}_{\mathbf{X}} \cdot$$

$$\cdot \underbrace{\begin{bmatrix} -1 & & & & \\ 1 & -1 & & & \\ & 1 & \cdots & & \\ & & \cdots & -1 & \\ & & & 1 & \end{bmatrix}}_{\mathbf{D}} \underbrace{\begin{bmatrix} f_0 \\ f_1 \\ \vdots \\ f_{N-1} \end{bmatrix}}_{\mathbf{F}}$$

We note that $\mathbf{X}$ is a Van der Monde with dimensions $(P+1) \times (N+1)$, matrix $\mathbf{D}$ is of dimensions $(N+1) \times N$. During the computation, we first conduct the multiplication $\mathbf{DF}$ yielding a vector $\mathbf{F}'_0$ of differences:

$$\mathbf{F}'_0 = \mathbf{DF} = \begin{bmatrix} f'_{0,0} \\ \vdots \\ f'_{0,N-1} \\ f'_{0,N} \end{bmatrix} = \begin{bmatrix} -f_0 \\ f_0 - f_1 \\ \vdots \\ f_{N-2} - f_{N-1} \\ f_{N-1} \end{bmatrix}.$$

Subsequently, we generate the vectors $\mathbf{F}'_i$ by successively multiplying componentwise with the vector of samples $\mathbf{S} = [x_0, x_1, \ldots, x_N]^T$:

$$\mathbf{F}'_{n+1} = \begin{bmatrix} f'_{n+1,0} \\ \vdots \\ f'_{n+1,N-1} \\ f'_{n+1,N} \end{bmatrix} = \begin{bmatrix} x_0 f'_{n,0} \\ \vdots \\ x_{N-1} f'_{n,N-1} \\ x_N f'_{n,N} \end{bmatrix}$$

The 1D geometrical moments M_p can thus be computed by adding up the components of $\mathbf{F}'_p$ and multiplying by a factor:

$$M_p = \frac{1}{p+1} \sum_{l=0}^{N} f'_{p,l}.$$

The 3D geometrical moments M_{pqr} on a N^3 grid can be written as:

$$M_{pqr} = \sum_{i=0}^{N-1} \frac{x_{i+1}^{p+1} - x_i^{p+1}}{p+1} \cdot \sum_{j=0}^{N-1} \frac{y_{j+1}^{q+1} - y_i^{q+1}}{q+1} \cdot \sum_{k=0}^{N-1} \frac{z_{k+1}^{r+1} - z_i^{r+1}}{r+1} f_{ijk}.$$

In the formula we have a tensor product, which already suggests that the three dimensional case can be split into 1D cases and may thus be computed in exactly same manner as above.

4.2.2 Pre-scaling

The high order radial polynomials of the 3D Zernike function tend to have small values near the origin. As a consequence, the projections of the object function f in the vicinity of the origin are suppressed – these values have relatively small impact on the final value of a high order 3D Zernike moment Ω_{nl}^m. On the other hand, scaling the object to fit into the unit ball means that we shift the most severe numerical inaccuracy caused by the floating point representation to the vicinity of the origin, since high order monomials have values with high negative exponent in this area.

Consequently, in order to obtain an improved numerical accuracy, as a first step we translate and scale the object according to the normalization transformation discussed above, and compute the geometrical moments afterwards. As is proven by our results in Section 6, this considerably improves the numerical accuracy of the final 3D Zernike descriptors.

5 Results

In this section we describe some practical results of our approach. We first describe our experimental setting and demonstrate the numerical accuracy of the algorithm and the reconstruction ability of 3D Zernike moments based representation. We subsequently specify the quality measures we used to quantify the performance of the descriptors. Furthermore, we discuss the dependency of the Zernike- as well as the SH descriptor retrieval performance on the voxelization method and number of used invariants. Finally, we present the results on performance of both types of descriptors.

5.1 Experimental setting

In our experiments, the discrete volumetric object functions were generated by voxelizing the polygonal boundary representations of geometric models. To this end we used the vxt software library [2]. See Section 5.4 for more details on voxelization.

From the volumetric representation of an object we generate the Zernike descriptors consisting of vectors of respective invariants and store them as search keys. To determine the similarity between two 3D shapes we compute Euclidean distances between the corresponding vectors. Our experimental 3D shape retrieval system generates a similarity ranking of the objects in database to a given query object. Hence, the presented system is a *query-by-example* system.

Test data An important problem we already mentioned in Section 1 is the notion of similarity, or ground truth classification, which can be used as a reference to measure against. To obtain results on a widely available standard data set we resort as a solution to the pre-classifications delivered by the Princeton Shape Benchmark (PSB) [5], however, it has to be emphasized that other classifications are possible as well. The current version of PSB contains 1814 models of various categories (animals, plants, furniture, etc.). We used the base classification to assess the descriptors. It splits the models into a training set containing 907 objects sorted into 90 classes and a test set consisting of 907 objects and 92 classes. We used the training set to obtain an optimal parameter configuration which is then utilized for final evaluation on the test data set. Note that the base classification of PSB defines hierarchies of nested classes, however, in our investigations we used only the leaf classes.

As for numerical experiments, we present results generated using 5 objects downloaded from www.3dcafe.com.

Spherical harmonic descriptors We compare the performance of 3D Zernike descriptors against the spherical harmonic descriptors introduced very recently by Funkhouser et al. [16]. These descriptors essentially fit into the model of rotationally invariant descriptor construction we presented in the Section 3, with the difference that the authors do not use radial polynomials to modulate the spherical harmonics, but sample the three dimensional space as concentric shells, where the shells are defined by equal radial intervals. Subsequently, they discretize the shells into equiangular bins, and define a binary spherical function defined as 1 if there is an object point in such a bin and 0 otherwise. Their object representation consists of a spherical harmonic decomposition for each shell.

The objects are first voxelized into a 64^3 grid after a normalization transformation similar to that described in the previous section. The authors use 32 concentric shells to define the spherical functions and 16 rotationally invariant spherical harmonic descriptors for each shell, this gives a vector of 512 scalar values for a single object. In order to obtain a fair comparison, similarly to the 3D Zernike descriptors, we conducted a parameter tuning to determine how many components yield the best retrieval performance, see Section 5.4. For the descriptor computation we use 32 concentric shells to generate the spherical functions, as suggested by Funkhouser et al., however, we compute 32 instead of 16 coefficients for each such shell to start with. We then choose

the count of coefficients which gives optimal retrieval performance. During the search in a database, the similarity of objects is calculated as Euclidean distance between the vectors of coefficients.

5.2 Numerical accuracy

Figure 1: The objects used to generate the results of Table 1.

In order to verify the numerical accuracy of our method, we used the GNU MP arbitrary precision arithmetic library (http://www. swox.com/gmp/) to generate an accurate reference. As already mentioned in the Section 4.2, the accuracy of computations is decisively influenced by the numerical stability of geometrical moments. We have implemented three versions of our software: with pre-scaling, with integration and with both scaling and integration. The results for the set of objects[2] depicted in the Fig. 1 are presented in Table 1. We obtained these results by computing the 3D Zernike descriptors from Zernike moments of order n up to 20 and calculating the L_2 norm of the difference between a vector of invariants yielded by the respective version and the precise values computed using the GNU MP. In order to be able to assess the values of Table 1 we note that the average distance between the objects in the database (see Section 5.1) is on the order of 0.25, the maximal distance is 0.6. This implies that the inaccuracy caused by incautious computation of geometrical moments render the approach unusable. Therefore, both the scaling and the integration are important components of the numerical calculations.

	Integrated	Pre-scaled	Integrated, Pre-scaled	L_2 norm
GUITAR	484.5	22.7	0	0.1359
T_INVADER	85215.7	796.2	0	0.269
INVCHAIR	874.4	997.2	1.6e-17	0.2812
747	2.6e5	56.7	1.3e-17	0.2039
BALL	1.7e7	2289.7	3.7e-9	0.2141

Table 1: The L_2 error in dependence of the geometrical moment calculation method and the norm of the descriptors representing the objects.

We note that the computations using the GNU MP library were about two orders of magnitude slower compared to those using the built-in double precision arithmetic.

[2] www.3dcafe.com/models/{747.zip, bball.zip, guitar.zip, wheelchr.zip, t-invadr.zip}

5.3 Reconstruction

Figure 2 demonstrates the reconstruction property of the 3D Zernike moments. We reconstructed the volumetric discrete representation object using Eqn. 11 on a 24^3 grid, we subsequently extracted the iso-surfaces to be able to visualize the objects. As it can be seen, the moments of order up to 20 allow for reconstructing the main object characteristics while discarding small details.

We note that in case of a large database of 3D objects, the underlying frequency metaphor may be used to accelerate the search process. We recall that the spherical harmonics form essentially a Fourier basis on the sphere and the radial polynomials may also be interpreted analogously in terms of their order. This allows us to generate a natural hierarchy of multiscale representations and enables the utilization of an efficient hierarchical search algorithm.

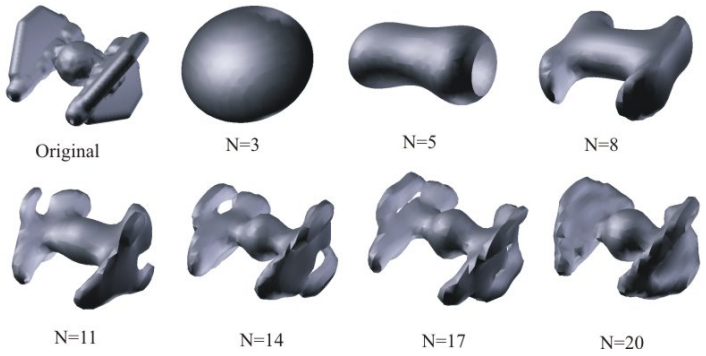


Figure 2: Reconstruction of a spaceship shown as isosurfaces of the reconstructed volumetric function. The upper left is the voxelized original object. The numbers below the images indicate the number of Zernike moments that have been used for the reconstruction.

5.4 Tuning the descriptors

We investigated the influence of several parameters of the descriptor computation on the retrieval performance, i.e. the voxelization method and number of invariants used as object characterization. Before we report on these experiments, we define the quality measures we used to assess a particular setting.

5.4.1 Retrieval quality measures

The goal of a retrieval system is to retrieve relevant documents while holding back non-relevant one. Measuring the effectiveness of retrieval systems is an open research area in Information Retrieval (see [40] for more details on this topic).

Let C denote a class of objects and $|C|$ the number of its elements. We quantify our results based on the following measures (see also [5] for more details):

- **Precision and recall.** Having the top k matches delivered by the retrieval system with n being the count of relevant objects among these matches, the recall $R_k = \frac{n}{|C|}$ indicates the ratio of the already retrieved elements of C. The precision $P_k = \frac{R_k}{k}$ denotes the ratio of this number to k. Informally, high precision values indicate that most of the retrieved objects are relevant, which, combined with a high recall, means that the system has already found most of the desired objects. Hence, a popular way to describe the performance of retrieval is to generate precision vs. recall diagrams. However, to compare retrieval results for different methods, scalar quality measures are more useful.

- **Average precision.** Intuitively, the higher the precision-recall curve, the better the retrieval performance. Therefore, the precision averaged over recall values $P^C_{avg} = 1/|C| \sum_{k=1}^{|C|} P_k$ may be used as a single scalar quality measure.

- **First and second tier** measure the recall for the top $|C|$ and $2|C|$ matches, respectively.

- **E and F measures.** The user is usually most interested in the first k matches, e.g. fitting onto the first result page. The E and F measures incorporate both the precision and recall computed for a fixed number of top k matches:

$$F = \frac{2}{1/P_k + 1/R_k}, E = \frac{b^2 P_k R_k + P_k R_k}{b^2 P_k + R_k}.$$

  b indicates the relative importance of precision and recall. We set $k = 32$ and $b = 0.5$, i.e. we assume that 32 thumbnail images representing the objects appear on a result page and precision and recall are weighted equally.

- **Discounted cumulative gain.** The intuition behind this measure is that the correct shapes retrieved sooner are more valuable, as often the user is unwilling to investigate the objects down the ranking. According to this, the discounted cumulative gain is computed by setting $a = 0$ and accumulating $a = a + 1/lg(i)$ if the ith object in the ranking is relevant until recall $= 100\%$. Finally, the accumulated gain is normalized by the best possible value, so that the discounted cumulative gain is $dcg = \frac{1+a}{1+1/\sum_{i=2}^{|C|} lg(i)}$.

For all the measures the maximal value is 1.0 and higher values imply better performance. The above quantities apply always for a particular query. In order to get overview results over individual classes, we average the measures over query objects from these classes. To get macro level results, we further average the quantities over all classes in the classification.

5.4.2 Parameter choice

We analyzed the influence of the following parameters:

- **Voxelization method**: we voxelized the polygonal boundary of the objects using radial linear, Gaussian and binary kernels with varying widths. The support

Measure	Max. value	#Coeffs
dcg	0.49	25
P_{avg}	0.30	29
E	0.128	29
F	0.164	29
First Tier	0.288	21
Second Tier	0.349	30

SH Descriptors

Measure	Max. value	#Coeffs
dcg	0.51	135
P_{avg}	0.333	152
E	0.136	116
F	0.175	116
First Tier	0.312	154
Second Tier	0.381	150

3D Zernike descriptors

Table 2: The maxima of macro-averaged retrieval measure values for both types of descriptors.

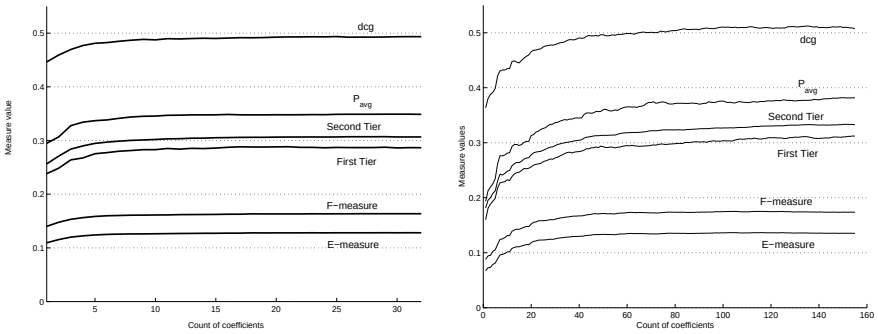
width of the kernel indicates the distance in voxel units where the kernel values decrease to zero. The voxelization rasterizes the polygonal boundary of the object; the kernel function determines the thickness and functional profile of the corresponding volumetric representation of the boundary. We investigated several resolutions of the voxel grid as well.

- **Number of coefficients**: we recall that the invariant description of objects consists of vectors of F_{nl} (Eqn. 10) for the Zernike descriptor and μ_l (Eqn. 4) for each concentric shell of the SH descriptors. To investigate the impact of the number of coefficients we successively added coefficients of higher order to the vectorial representation of each descriptor. Note that in case of SH descriptors we simultaneously add a coefficient of incremented l for each concentric shell, while in case of the Zernike descriptors we add one coefficient each in order $F_{00}, F_{11}, F_{20}, F_{22}, F_{31}, F_{33}, \ldots$ (the value of $(n-l)$ must be even, Eqn. 7 and 10). Utilizing low number of coefficients we discard the high frequencies of the objects, while for high values detail information is incorporated into the comparison as well.

As for voxelization we found the binary kernel and resolution of 64^3 to yield the best results. Moreover, for all measures and for both descriptors the unit kernel width resulted in best performance macro-averaged over the leaf classes of the base training classification of PSB. The maximal macro-averaged measure values for the six scalar measures and the corresponding coefficient counts are summarized in Table 2. The plots depicting the change in macro-averaged measure values caused by successively adding coefficients to the representations can be found in Figure 4.

As it can be seen, the best performance is obtained for different coefficient counts in case of both descriptors. However, the plots in Figure 4 indicate that after reaching a certain count of coefficients, the change of values is not significant, when adding further coefficients. Thus, as a simple solution one could choose a coefficient count of about 30 and 152 for the SH and 3D Zernike descriptors, respectively. To make the choice deterministic, we use the following straightforward procedure for each descriptor type: we normalize the plots by scaling the maxima to an unit and sum up the resulting scaled measure values. Subsequently, we choose the coefficient count corresponding

to the maximum of this sum. The coefficient counts obtained by this approach are 29 and 154 for the SH- and 3D Zernike descriptors, respectively.



SH 3D Zernike

Figure 3: Plots of the macro-averaged retrieval performance measure values against increasing coefficient count. Beside measuring the performance, the plots confirm the multiscale nature of both descriptors, i.e. that by adding coefficients of higher order to the representation, the overall performance of the system improves. This fact may be used to accelerate the search process in large databases using a coarse-to-fine search strategy.

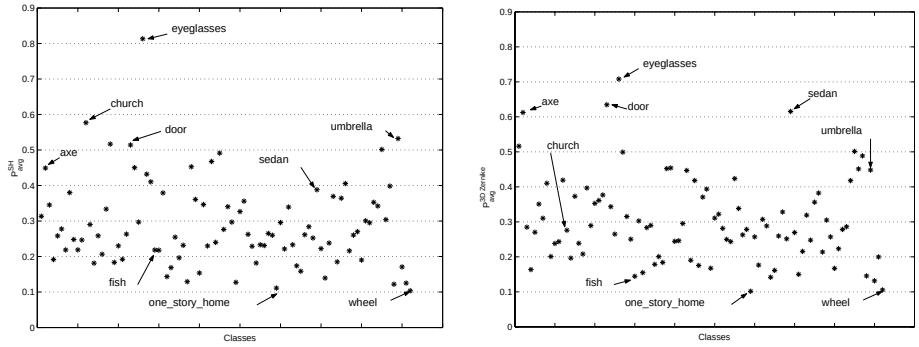
5.5 Retrieval performance

To summarize the results of the previous subsection, the best performance on the training base classification of PSB were obtained using a unit kernel width binary voxelization in a 64^3 voxel grid. Furthermore, in what follows, we will use coefficient counts of 29 and 154 for the SH and 3D Zernike descriptor generation, respectively. As the coefficients have to be computed for each of the 32 concentric shells in case of the SH descriptors (see Section 5.1), $32 \cdot 29 = 928$ floating point values are needed to store these descriptors without compression. For the 3D Zernike descriptors only 154 floats are necessary. This means that even if one discards some number of spherical harmonic coefficients to trade retrieval performance for less storage, the 3D Zernike descriptors are considerably more compact. In consequence, the dimensionality of the search problem is reduced, which supports the effectiveness of the search process, and there is less storage overhead for an object.

Both 3D Zernike and spherical harmonic descriptors achieve rotation invariance by exploiting the invariance properties of the spherical harmonics. However, by merely sampling the space in radial direction, the latter descriptor does not capture object ”frequencies” or coherence in this direction, thereby not incorporating such object characteristic information. As shown in Table 3, the 3D Zernike descriptors perform better on average in spite of their compactness. A plot of P_{avg} for each class corresponding to both types of descriptor can be seen in Figure 4. We note that for other measures the relations in plots are very similar.

	SH	3D Zernike
dcg	0.4635	0.4808
P_{avg}	0.2879	0.3028
E	0.1237	0.1320
F	0.1578	0.1688
First Tier	0.2675	0.2808
Second Tier	0.3239	0.3417

Table 3: The macro-averaged retrieval performance measures based on the leaf classes of the base test classification of PSB



SH 3D Zernike

Figure 4: Plots of P_{avg} values averaged over the leaf classes of the base test classification of PSB.

Although in average sense the Zernike descriptors perform better, there is a certain disparity in the results: for some classes the SH descriptor performs better, for some the 3D Zernike descriptor. To visualize this phenomenon, we subtracted the vectors of retrieval performance measures corresponding to the SH- and 3D Zernike descriptors, where each element of the vectors correspond to the measure averaged over the corresponding class. We investigated these quantities for all measures separately, and it turned out that the results were qualitatively almost the same for each type of measure. Therefore, in order to obtain a clear picture, we illustrate the values in plot of Figure 5 only for P_{avg}. On the x-axis the classes are represented, the negative and positive y-values indicate the superiority of the SH- and 3D Zernike descriptor, respectively.

We indicate some of the classes for selected extrema in both directions, as well as some classes for which the performance was very similar. The detailed data on the performance can be found at [4]. It is indeed apparent that for some classes the difference in performance is considerable. Given the similar structure of the descriptors, this raises the question of what causes this disparity and how the descriptors can be modified to improve the retrieval performance. Analysis of this question is certainly a central avenue for future work.

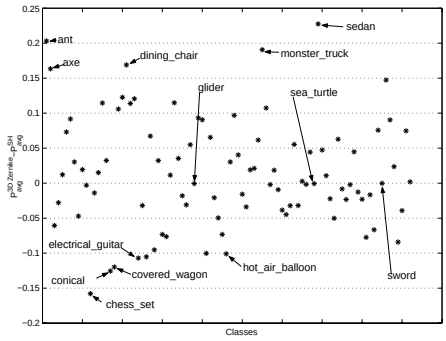


Figure 5: Per class averaged difference of P_{avg} between the measure value for the SH- and 3D Zernike descriptors. On the x-axis the classes are represented, the negative and positive y-values indicate the superiority of the SH- and 3D Zernike descriptor, respectively.

Similarly to the spherical harmonic descriptors, the representation as 3D Zernike descriptors is insensitive to geometric and topological artifacts common to freely available objects.

Finally, we mention some timing data measured on a 3.06 GHz Pentium 4 with 1 GB RAM; our experimental implementation may be downloaded from [4]. The voxelization of objects takes from 0.2s to 12s, depending on the polygon count, the average voxelization time was about 5s. The computation of 3D Zernike descriptors consisting of 156 coefficients takes 0.25s on a 64^3 grid, the SH descriptor computation on the same grid generating 32 coefficients for 32 concentric shells takes 0.2s. Finally, a retrieval from a database of 907 objects lasts approximately 0.05s and 0.1s for the 3D Zernike descriptors and SH descriptors, respectively. We once again mention that for larger database the performance can considerably be improved exploiting the multiscale nature of the representation confirmed by our experiments – see Section 5.3 and Figure 4.

6 Conclusions and future work

In this paper we utilized the 3D Zernike descriptor for the purpose of retrieval of 3D objects. We discussed some general rules for the construction of affine invariant object descriptors and derived the 3D Zernike descriptors within this framework. We furthermore considered the implementational issues: the severe instability of geometrical moments and hence the 3D Zernike descriptors in case of high orders. As a cure to this problem, we applied analytical integration within each voxel and scaled the object prior to computations, thereby achieving high accuracy even for high orders of Zernike moments. The quality of the descriptor regarding the retrieval performance was analyzed and verified also with respect to another related recent technique: the spherical harmonic descriptor. As it turns out, the 3D Zernike descriptors compare favorably to

recent descriptors for general 3D objects in terms of retrieval performance and robustness against topological and geometrical artifacts plaguing a most of freely available models.

As for short term future work we plan to investigate the usage of further radial functions: a wavelet based function seems to be promising, as such basis would allow for a multi-resolution radial localization of frequencies. As already mentioned, the closer analysis of the relation of the compared SH- and 3D Zernike descriptors is a task we will tackle in the near future, as well. Moreover, we intend to elaborate a new distance function between the descriptors, as we suspect that different coefficients contribute to a different extent to the overall shape information.

7 Acknowledgments

We thank Roland Wahl, Patrick Degener and Gabriel Zachmann for the insightful discussions. Special thanks go to Szabolcs Hódossy for his invaluable support as well. We are grateful to Misha Kazhdan for his kind help on the SpharmonicKit. We appreciate very much the valuable comments of the anonymous reviewers which helped to improve this paper. This work was partially funded by the German Research Foundation (DFG) within the initiative V^3D^2 "Distributed Processing and Delivery of Digital Documents".

References

[1] 3D Knowledge, `http://3dk.asu.edu`.

[2] Milos Sramek's volume visualization page, `http://www.viskom.oeaw.ac.at/~milos/`.

[3] National design repository, `http://edge.mcs.drexel.edu/repository/frameset.html`.

[4] 3D Shape Rertieval at the University of Bonn, `http://www.cg.cs.uni-bonn.de/project-pages/3dsearch/`, 2003.

[5] Princeton Shape Benchmark, `http://shape.cs.princeton.edu/benchmark/index.cgi`, 2003.

[6] M. Ankerst, G. Kastenmuller, H.-P. Kriegel, and T. Seidl. 3D shape histograms for similarity search and classification in spatial databases. In *Symposium on Large Spatial Databases*, pages 207–226, 1999.

[7] I. Biederman. Recognition-by-components: A theory of human image understanding. *Psychological Review*, 94:115–147, 1987.

[8] H. Blum. Biological shape and visual science. *Journal of Theoretical Biology*, 38:205–287, 1973.

[9] N. Canterakis. Vollständige Invarianten und Lageberechnung für allgemeine Drehspiegelungen von 3D Objekten. Technical report, Technische Universität Hamburg-Harburg, 1995.

[10] N. Canterakis. 3D Zernike moments and zernike affine invariants for 3D image analysis and recognition. In *11th Scandinavian Conf. on Image Analysis*, 1999.

[11] Vincent Cicirello and William C. Regli. Machining feature-based comparisons of mechanical parts. pages 176–185. Int'l Conf. on Shape Modeling and Applications, May 2001.

[12] J. Corney, H. Rea, D. Clark, J. Pritchard, M. Breaks, and R. MacLeod. Coarse filters for shape matching. *IEEE Computer Graphics and Applications*, 22(3):65–74, 2002.

[13] H. Dym and H.P. McKean. *Fourier series and integrals*. Academic Press, 1972.

[14] M. Elad, A. Tal, , and S. Ar. Content based retrieval of VRML objects - an iterative and interactive approach. In *Eurographics Multimedia Workshop*, pages 97–108, 2001.

[15] David Forsyth, Jitendra Malik, Margaret Fleck, and Jean Ponce. Primitives, perceptual organization and object recognition. Technical report, Computer Science Division, University of California at Berkeley, Berkeley, CA 94720, 1997.

[16] Thomas Funkhouser, Patrick Min, Michael Kazhdan, Joyce Chen, Alex Halderman, David Dobkin, and David Jacobs. A search engine for 3D models. *ACM Transactions on Graphics*, 22(1), 2003.

[17] Amarnath Gupta and Ramesh Jain. Visual information retrieval. *Communications of the ACM*, 40(5):70–79, 1997.

[18] M. Hilaga, Y. Shinagawa, T. Kohmura, and T. L. Kunii. Topology matching for fully automatic similarity estimation of 3D shapes. In *Proceedings of ACM SIGGRAPH*, 2001.

[19] M. K. Hu. Visual pattern recognition by moment invariants. *IRE Trans. Information Theory*, 8(2):179–187, 1962.

[20] Charles E. Jacobs, Adam Finkelstein, and David H. Salesin. Fast multiresolution image querying. In *Proceedings of SIGGRAPH '95*, pages 277–286, 1995.

[21] Michael Kazhdan, Bernard Chazelle, David Dobkin, Thomas Funkhouser, and Szymon Rusinkiewicz. A reflective symmetry descriptor for 3D models, 2003. To appear in Algorithmica.

[22] A. Khotanzad and Y.H. Hong. Invariant image recognition by Zernike moments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(5), 1990.

[23] F. Leymarie and B. Kimia. The shock scaffold for representing 3d shape. In *Proc. of 4th International Workshop on Visual Form (IWVF4)*, 2001.

[24] S. Loncaric. A survey of shape analysis techniques. *Pattern Recognition*, 31(8):983–1001, 1998.

[25] David McWherter, Mitchell Peabody, William C. Regli, and Ali Shokoufandeh. Transformation invariant shape similarity comparison of solid models. ASME Design Engineering Technical Confs., 6th Design for Manufacturing Conf. (DETC 2001/DFM-21191), Sep 2001.

[26] Patrick Min, John A. Halderman, Michael Kazhdan, and Thomas A. Funkhouser. Early experiences with a 3D model search engine. In *Proc. Web3D Symposium*, 2003.

[27] Ryutarou Ohbuchi, Tomo Otagiri, Masatoshi Ibato, and Tsuyoshi Takei. Shape-similarity search of three-dimensional models using parameterized statistics. In *Pacific Graphics*, 2002.

[28] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin. Matching 3D models with shape distributions. In *International Conference on Shape Modeling and Applications*, 2001.

[29] E. Paquet and M. Rioux. A content-based search engine for VRML databases. In *CVPR Proceedings*, pages 541–546, 1998.

[30] E. Petrakis. Design and evaluation of spatial similarity approaches for image retrieval. *Image and Vision Comp.*, 20(1):59–76, 2002.

[31] Yong Rui, Thomas S. Huang, and Shih-Fu Chang. Image retrieval: Past, present, and future. In *International Symposium on Multimedia Information Processing*, 1997.

[32] M. Schneier and M. Abdel-Mottaleb. Exploiting the jpeg compression scheme for image retrieval. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 18(8):849–853, 1996.

[33] Kaleem Siddiqi, Ali Shokoufandeh, Sven J. Dickinson, and Steven W. Zucker. Shock graphs and shape matching. In *ICCV*, pages 222–229, 1998.

[34] A.W.M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12):1349–1380, 2000.

[35] Motofumi T. Suzuki, Toshikazu Kato, and Nobuyuki Otsu. A similarity retrieval of 3D polygonal models using rotation invariant shape descriptors. In *IEEE International Conference on Systems, Man, and Cybernetics (SMC2000)*, pages 2946–2952, 2000.

[36] M.J. Swain and D.H. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1):11–32, 1991.

[37] Johan W.H. Tangelder and Remco C. Veltkamp. Polyhedral model retrieval using weighted point sets. *International Journal of Image and Graphics*, 3(1):1–21, 2003.

[38] M.R. Teague. Image analysis via the general theory of moments. *Journal Optical Society of America*, 70(8):920–930, 1980.

[39] C.-H. Teh and R. T. Chin. On image analysis by the methods of moments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(4):496–513, 1988.

[40] C. J. Van Rijsbergen. *Information Retrieval, 2nd edition*. Dept. of Computer Science, University of Glasgow, 1979.

[41] D. V. Vranic and D. Saupe. Description of 3D-shape using a complex function on the sphere. In *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2002)*, pages 177–180, 2002.

[42] C. T. Zahn and R. Z. Roskies. Fourier descriptors for plane closed curves. *IEEE Transactions on Computers*, 21:269–281, 1972.