

# A Statistical Model of Riemannian Metric Variation for Deformable Shape Analysis

Andrea Gasparetto and Andrea Torsello

Dipartimento di Scienze Ambientali, Informatica e Statistica  
Università Ca' Foscari Venezia - via Torino, 155 - 30172 Venice Italy

{andrea.gasparetto,torsello}@unive.it

## Abstract

*The analysis of deformable 3D shape is often cast in terms of the shape's intrinsic geometry due to its invariance to a wide range of non-rigid deformations. However, object's plasticity in non-rigid transformation often result in transformations that are not completely isometric in the surface's geometry and whose mode of deviation from isometry is an identifiable characteristic of the shape and its deformation modes. In this paper, we propose a novel generative model of the variations of the intrinsic metric of deformable shapes, based on the spectral decomposition of the Laplace-Beltrami operator. To this end, we assume two independent models for the eigenvectors and the eigenvalues of the graph-Laplacian of a 3D mesh which are learned in a supervised way from a set of shapes belonging to the same class. We show how this model can be efficiently learned given a set of 3D meshes, and evaluate the performance of the resulting generative model in shape classification and retrieval tasks. Comparison with state-of-the-art solutions for these problems confirm the validity of the approach.*

## 1. Introduction

The ability to retrieve similar 3D objects given a query object has become of great importance in several fields, like medical research, automatic information retrieval systems and copyright protection. The main challenge in 3D object retrieval algorithms is to define an invariant representation of a shape capable of capturing geometrical and topological properties of a shape [37, 7, 47]. A large number of methods for 3D shape retrieval have been proposed [35, 20, 44, 32, 3], but most of them are only suitable for the representation of rigid 3D shapes. Non-rigid 3D shape descriptors are more challenging to define. Indeed, those representations still need to capture the most distinctive properties of a shape, but they should be also insensitive to many other transformations (e.g. inelastic transformation,

acquisition resolution and noise *etc.*).

Various methods have been proposed to address the non-rigid shape retrieval problem. These can be mainly categorized according to two main directions: topology-based approaches against surface-based methods and the employment of shape descriptors that can be local or global. The first solution usually captures the global topological structure of the shape [31, 25, 34], while the surface-based methods usually exploit spectral shape analysis in order to build a spectral descriptor [46, 5, 42] to be used for the task. Finally, one can employ global descriptors in order to characterize the whole shape [46, 11] or define a set of local descriptors [13, 15] that characterize only parts of the object (e.g. segments or points).

### 1.1. Related works

The methods which define a shape descriptor (both local and global) to tackle the non-rigid 3D shape recognition are the ones we are most interested in. A popular approach to define shape descriptors is through spectral shape analysis. Research efforts have recently resulted in several spectral descriptors [46, 5, 40, 22] usually based on the spectral decomposition of the Laplace-Beltrami operator. In particular, the graph-Laplacian, the discrete counterpart of the Laplace-Beltrami operator, has been extensively used to provide spectral representations of structures [29]. Reuter *et al.* [39] suggest to use the Laplace-Beltrami spectra as fingerprints of surfaces, while Jain and Zhang [19] propose to use the eigenvalues of the geodesic distance matrix of a 3D object in order to build the associated shape descriptor. Huang *et al.* [18], on the other hand, build the signature directly over local features, selecting discriminative volumetric features over pre-aligned shapes.

The aggregation of local descriptors in order to build a global descriptor is a general thread in the literature. For this purpose, the *Bag-of-Features* (BoF) paradigm is quite popular and has been successfully applied to 3D shape description [10, 12, 23, 48]. Li and Hamza [27] used the BoF paradigm combining the exploitation of hierarchical

structures of the shape, such as pyramid matching [16] and spatial relationship [10, 12, 23]. They proposed to adopt the eigenfunction associated with the second-smallest eigenvector of the Laplace-Beltrami operator in order to build a global surface coordinate system which is insensitive to shape deformation, showing that the introduction of global spatial context could improve the effectiveness of their descriptor in 3D shape recognition. Spatial pyramid [30, 27, 24], is the term used to identify this approach. Other approaches inspired by text-analysis have been proposed. For instance, in [8, 17] the authors adopt higher-order models defining relations between ‘geometric words’.

Within the bag of features model, features quantization is generally performed through unsupervised learning. Departing from this approach, Litman *et al.* [28] recently proposed a new supervised BoF framework mapping the discriminative training directly into the dictionary construction step.

## 1.2. Main contribution

In this paper, we propose a new supervised technique to learn a statistical model build on the Riemannian metric variations on deformable shapes based on the spectral decomposition of the Laplace-Beltrami operator. To this end, we define a statistical framework that models a shape as two independent models for the eigenvectors and for the eigenvalues. The eigenvector and eigenvalue matrices of a set of discrete representations (*i.e.* meshes representing the shape in different poses) are assumed to be points of two separate smooth Riemannian manifolds. The invariant representation of a certain shape is computed as the centroid of such manifolds. The shape centroid acts as our invariant representation of the shape. One of the main contributions is indeed the definition of an efficient algorithm that compute the spectral decomposition of this centroid. In particular, we define it as a supervised data-driven optimization process. For this purpose, we define a computationally efficient intrinsic distance metric over the eigenvectors manifold. Finally, we define a statistical model over the variations of such metric, assuming that the squared geodesic distances follow a  $\Gamma$ -distribution. Moreover, in order to define a descriptor that is robust to small non-isometric perturbations, we assume that the eigenvalues are log-normally distributed for the same stability reasons presented by Aubry *et al.* [5].

## 2. Background

We model a shape as a surface which is invariant to rigid and non-rigid transformations, or, quoting D.G. Kendall [21], a shape is ‘what is left when the differences which can be attributed to translations, rotations, and dilatations have been quotiented out’. A representation of such shape is modelled as a smooth compact 2-manifold  $\mathcal{S}$  without boundary isometrically embedded in some Eu-

clidean space  $\mathbb{R}^3$  with geometry induced by the embedding. A 3D mesh is a discretization of a shape embedded in  $\mathbb{R}^3$ , and a discrete Laplacian is a discretization of the continuous Laplace-Beltrami operator on the mesh. There are various ways by which such discretization can be obtained from the mesh. Here we adopt the algorithm proposed by Belkin *et al.* [6] which offers point-wise convergence guarantees and was experimentally shown to be quite robust.

The spectral representation of the mesh can be obtained from the Laplacian through singular value decomposition. Given a Laplacian  $\mathcal{L}$ , its decomposition is  $\mathcal{L} = \Phi\Lambda\Phi^T$ , where  $\Lambda = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_{|V|})$  is the matrix whose diagonal contains the ordered eigenvalues, while  $\Phi = (\phi_1|\phi_2|\dots|\phi_{|V|})$  is the matrix whose columns are the ordered eigenvectors. This decomposition is unique up to a permutation of the vertices of the mesh, a change of sign of the eigenvectors, or a change of basis over the eigenspaces associated with a single eigenvalue, *i.e.*, the following properties hold:

$$\mathcal{L} \simeq PLP^T = P\Phi\Lambda(P\Phi)^T \quad (1)$$

$$\mathcal{L} = \Phi\Lambda\Phi^T = \Phi S\Lambda S\Phi^T \quad (2)$$

where  $\simeq$  indicates isomorphism of the underlying surfaces,  $P$  is a permutation matrix and  $S$  is a diagonal matrix with diagonal entries equal to  $\pm 1$ .

## 3. Statistical model definition

One of the main contribution of this paper is the definition of an efficient algorithm to compute the spectral decomposition of the manifold centroid from a set of points lying on it. In order to address the classification task, we study the variations of the metrics in the manifold, casting them into a statistical framework. So let  $M^S = \{M_1^S, M_2^S, \dots, M_N^S\}$  be a set of meshes which represent a discretization of the same underlying shape  $\mathcal{S}$  in different poses. In the proposed method, we assume that each component of the spectral decomposition of the Laplacian of each mesh  $M_i^S$ ,  $i = 1, \dots, N$ , lies on a Riemannian manifold. Let  $\Delta_{M_i^S}$  be the Laplace operator introduced in section 2 applied to the mesh  $M_i^S$ . The spectral decomposition of  $\Delta_{M_i^S}$  yields the eigenvectors  $\Phi_{M_i^S}$  and the eigenvalues  $\Lambda_{M_i^S}$  ( $\Phi_i$  and  $\Lambda_i$  in short) such that

$$\Delta_{M_i^S} = \Phi_{M_i^S} \Lambda_{M_i^S} \Phi_{M_i^S}^T \quad (3)$$

These components are used as a basis for our invariant representation of the shape  $\mathcal{S}$ . Let  $M_0^S = \{\Phi_0^S, \Lambda_0^S\}$  be such invariant representation. We treat the spectral components independently, building a model for the eigenvectors and a model for the eigenvalues. Since the Laplacian matrix is a positive-semidefinite matrix, the eigendecomposition

of such matrix yields an orthogonal basis of eigenvectors,  $\phi \in \mathcal{O}(n)$ .

The set of orthogonal matrix is composed by two disjoint subsets, one with matrices with determinant +1 (*i.e.* the special orthogonal group  $\mathcal{SO}(n)$ ) and one with matrices with determinant -1. In the computation of the geodesic distance, the two orthogonal matrices must belong to the same subset. However, the arbitrariness of the sign of the eigenvectors guarantee that we can always find representatives in the same connected component. In particular, choosing the sign-flip  $S$  that minimizes the Frobenius distance between the eigenvector matrices, we guarantee that we pick representatives that belong to the same connected component and which are closest to one another in terms of geodesic distance. This requires a pre-processing of the eigendecomposition of the Laplacians of the whole class in such a way that the direction of the eigenvectors are concordant. To this end, we take a reference mesh and flip the sign of the eigenvectors of the other meshes in such a way as to make the dot product of corresponding eigenvectors form the different meshes positive. Under this assumption, we define  $\Phi_0^S$  as the solution to the following minimization problem:

$$\operatorname{argmin}_{\Phi_0^S \in \mathcal{O}(n)} \sum_i^N d^2(\Phi_i^S, \Phi_0^S) \quad (4)$$

where  $d^2$  is the geodesic distance between the eigenvector matrices, while  $N$  is the number of meshes representing the same shape  $\mathcal{S}$ . Given two orthogonal matrices  $R_1 \in \mathcal{O}(n)$  and  $R_2 \in \mathcal{O}(n)$  belonging to the same connected component of  $\mathcal{O}(n)$ , we can define the geodesic distance as

$$d_g(R_1, R_2) = \|\log(R_1^T R_2)\|_F \quad (5)$$

where  $\|\cdot\|_F$  is the Frobenius norm. For orthogonal matrices belonging to the same connected component of  $\mathcal{O}(n)$ , we have

$$d_g^2(R_1, R_2) = \sum_i^n \Theta_i^2 \quad (6)$$

where  $\Theta_i$  are the angles of the rotation  $R_1^T R_2 \in \mathcal{SO}(n)$ .

Using Taylor's expansion of  $\cos \Theta$ , we obtain  $\Theta^2 = 2 - 2 \cos \Theta + O(\Theta^4)$ . Recalling that for  $A \in \mathcal{SO}(n)$   $\operatorname{Tr}(A) = \operatorname{Tr}(\frac{1}{2}(A + A^T)) = \sum_i^n \cos(\Theta_i)$ , we can approximate the geodesic distance as

$$d_g^2(R_1, R_2) = 2n - \operatorname{Tr}(R_1^T R_2) + O(\Theta_i^4) \quad (7)$$

resulting in the following approximation of 4:

$$\operatorname{argmin}_{\phi_0 \in \mathcal{O}(n)} 2Nn - 2\operatorname{Tr}\left(\left(\sum_i^N \phi_i^T\right)\phi_0\right) \quad (8)$$

which is equivalent to

$$\operatorname{argmax}_{\phi_0 \in \mathcal{O}(n)} \operatorname{Tr}\left(\left(\sum_i^N \phi_i^T\right)\phi_0\right) \quad (9)$$

For this work, we assume that the geodesic distances between eigenvector matrices of the Laplacian of the meshes that represent the same shape follow a  $\Gamma$ -distribution with shape parameter  $k$  and scale parameter  $\theta$ .

$$\Gamma(x; k, \theta) = \frac{1}{\theta^k \Gamma(k)} x^{k-1} e^{-\frac{x}{\theta}} \quad (10)$$

where  $N$  is the number of meshes involved in the model construction,  $x$  are the geodesic distances between the eigenvectors of the meshes and the eigenvectors centroid, and  $k$  and  $\theta$  are estimated using the maximum likelihood estimation. Since there is no closed form solution for the parameter  $k$ , we approximate it using [33] and computing the parameter  $s$  as

$$s = \log\left(\frac{1}{N} \sum_{i=1}^N \sqrt{x_i}\right) - \frac{1}{N} \sum_{i=1}^N \log(x_i) \quad (11)$$

where  $x_i$  in our instance is the geodesic distance between the eigenvectors of the Laplacian of the mesh  $i$  and the eigenvectors centroid of the class. Using  $s$ , we can approximate  $k$  as

$$k \approx \frac{3 - s + \sqrt{(s-3)^2 + 24s}}{12s} \quad (12)$$

Once the shape parameter is computed, we use it to compute the second parameter of the distribution,  $\theta$ .

$$\theta = \frac{1}{kN} \sum_{i=1}^N \sqrt{x_i} \quad (13)$$

Finally, thanks to 7, we are able to define the geodesic distance between two eigenvector matrices, which is computed as

$$d^2(\phi_i, \phi_0) = 2n - 2\operatorname{Tr}(\phi_i^T \phi_0) \quad (14)$$

The second component of the spectral decomposition of our manifold centroid is simply the exponentiation of the average of the logarithm of each  $j$ -th eigenvalue

$$\Lambda_{j0} = e^{\frac{1}{N} \sum_i^N \log \lambda_{ji}} \quad (15)$$

We assume that the eigenvalues follow a log-normal distribution. In particular, we assume that each  $i$ -th eigenvalue follows the same distribution, *i.e.* shares the same distribution parameters.

$$\log \mathcal{N}(x; \mu, \sigma) = \frac{1}{x\sqrt{2\pi\sigma}} e^{-\frac{(\log x - \mu)^2}{2\sigma^2}} \quad (16)$$

where  $x$  is an eigenvalue,  $\mu$  is the mean of the distribution and  $\sigma$  is the standard deviation. Let  $\lambda_{i0}$  be the  $i$ -th eigenvalue of the spectral decomposition of the manifold centroid. Then, the distribution mean is defined as

$$\mu_i = \log \lambda_{i0} \quad (17)$$

Let  $\lambda_{ij}$  be the  $i$ -th eigenvalue of the mesh  $j$ . We define the standard deviation  $\sigma$  as

$$\sigma_i = \sqrt{\frac{1}{N} \sum_{j=1}^N (\log \lambda_{ij} - \mu_i)^2} \quad (18)$$

Finally, we can define the density of the  $i$ -th eigenvalue of the  $j$ -th mesh as

$$\lambda_{d_{ij}} = \frac{1}{\lambda_{ij} \sqrt{2\pi\sigma_i}} e^{-\frac{(\log(\lambda_{ij}) - \mu_i)^2}{2\sigma_i^2}} \quad (19)$$

Once both densities are computed, it is possible to compute the density of a particular mesh as

$$p(j) = \left( \prod_i \lambda_{d_{ij}} \right) \Gamma(k, \theta)(g_{d_j}) \quad (20)$$

where  $g_{d_j}$  is the geodesic distance between the eigenvector matrix of the Laplacian of a mesh and the eigenvector centroid of the shape we are considering.

#### 4. Embedding, isometries and lack of correspondences

The use of the whole Laplacian eigendecomposition of a mesh is not necessary, nor encouraged, since we are usually more interested in the smallest eigenvalues and the associated eigenvectors. In fact, most of the structural information is stored in those eigenvalues and eigenvectors. And that is why we embed the Laplacian matrix into a lower dimensional space. Let  $p$  be the embedding dimension. In the formulation of the geodesic distances 7 and 14, the embedding dimension must be taken into account and  $p$  should be used in place of  $n$ .

The use of spectral shape descriptor entails several advantages like the simple representation, scale invariance and a very good performance for shape retrieval of non-rigid shapes [38]. But there are also several issues that raise from its employment. For example, the isometric embedding of a Riemannian manifold on a lower dimensional space is unique up to isometries. For this reason, we introduce a new rigid transformation  $\mathcal{R}_i$  which aligns the eigenvectors of the Laplacian of a mesh  $i$  with the eigenvector centroid of a certain shape. Hence, we can rewrite 9 introducing the rotation matrix  $\mathcal{R}_i$ , obtaining

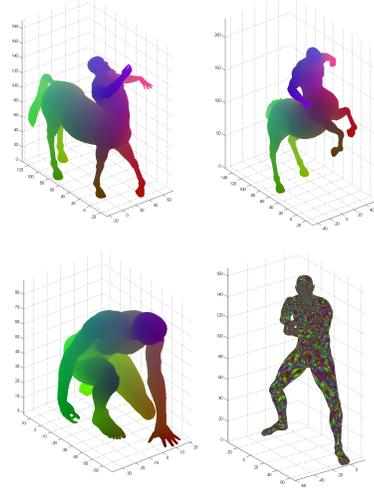


Figure 1. Example of known vertex correspondences between meshes which represent the same shape. These meshes, the Centaur (first row) and a human (second row), belong respectively to the SHREC'10 and SHREC'14 dataset. For the former dataset the correspondence ground-truth is known. In particular, we coloured the vertex accordingly to the correspondence map between the meshes. Thus, the same part (e.g. the head) takes the same color in both the representations if our map is consistent. The second row shows an example of two meshes which are not in correspondence.

$$\operatorname{argmax}_{\phi_0, \mathcal{R}_i \in \mathcal{SO}(p)} \operatorname{Tr} \left( \sum_i \mathcal{R}_i \phi_i^T \phi_0 \right) \quad (21)$$

The construction of our invariant representation of a shape is based on the assumption that the correspondences between the vertices of two meshes is known. But usually this kind of information is not available. Thus, the second issue we dealt with is related to the lack of correspondences between two meshes. There are several datasets available that provide the ground-truth for the correspondences (at least of the meshes that represent the same shape, i.e. meshes that belong to the same class), like TOSCA dataset [2] or the kids dataset [41]. But most of the datasets used in shape retrieval task do not provide such information. In order to make our method as flexible as possible, we cast the problem of finding a fine matching between two meshes to a maximum bipartite matching, solving the lack of correspondences as an assignment problem. So let  $\phi_i \in R^{n \times p}$  and  $\phi_j \in R^{m \times p}$  be the eigenvectors of the embedded Laplacian of two meshes. Let  $\mathcal{R} \in \mathcal{SO}(p)$  the rotation matrix which aligns the two meshes. We define a weight matrix  $W$  as

$$W = \phi_i \mathcal{R} \phi_j^T \quad (22)$$

This matrix is then used as input for the Hungarian algorithm [9]. The Hungarian algorithm yields a permuta-

tion matrix  $P$  that minimize the assignment cost. Since we want to maximize that cost, the input of the algorithm is the negation of the weight matrix  $W$ . Hence, we can define the introduced problem as the maximization problem

$$\operatorname{argmax}_{\mathcal{R}, P} \operatorname{Tr}(\phi_j \mathcal{R} \phi_i^T P) \quad (23)$$

while the geodesic distance between two eigenvector matrices is defined as

$$d^2(\phi_i, \phi_j) = 2n - 2\operatorname{Tr}(\phi_j \mathcal{R} \phi_i^T P) \quad (24)$$

## 5. Learning the invariant representation

The proposed method consists of two independent phase. In this section we present the first one, whose main purpose is to learn the invariant representation of each shape of a given dataset. In other words, given a shape  $S$ , characterized in the dataset by a set of meshes  $\{M_1, M_2, \dots, M_N\}$ , we want to solve the problem 9 in order to compute the eigendecomposition of the manifold centroid  $\{\Phi_0, \Lambda_0\}$  of the shape. For the eigenvectors component, we employ an iterative optimization process. An iterative approach is mandatory since we have to find the optimum eigenvector centroid  $\Phi_0$  while simultaneously optimizing each rotation matrix  $\mathcal{R}_i$ , one for each of the  $N$  meshes of the training set. The adoption of the iterative optimization process allows to split the problem defined in 21 in two parts, one for the eigenvector centroid, defined as

$$\Phi_0 = \operatorname{argmax}_{\Phi_0 \in \mathcal{O}(n)} \operatorname{Tr} \left( \left( \sum_i^N \mathcal{R}_i \Phi_i^T \right) \Phi_0 \right) \quad (25)$$

and one for the rotation matrix which align the  $i$ -th eigenvector matrix with  $\Phi_0$ .

$$\mathcal{R}_i = \operatorname{argmax}_{\mathcal{R}_i \in \mathcal{SO}(n)} \operatorname{Tr} \left( \left( \sum_i^N \Phi_i^T \Phi_0 \right) \mathcal{R}_i \right) \quad (26)$$

So, we define our iterative optimization process as

1. Computation of the mesh Laplacian of each mesh of the training set,  $\{\Delta_1, \Delta_2, \dots, \Delta_N\}$ , using Belkin *et al.* formulation [6].
2. Eigendecomposition of the Laplacians, which yields the set  $\{(\Phi_1, \Lambda_1), (\Phi_2, \Lambda_2), \dots, (\Phi_N, \Lambda_N)\}$ . The eigenvalues are used in the second part of the learning phase in order to compute the eigenvalues centroid  $\Lambda_0$ .
3. After the application of the rotation matrix  $\mathcal{R}_i$  to each  $\Phi_i$ , we sum the eigenvector matrices together  $\Sigma_\Phi = \sum_i^N \mathcal{R}_i \Phi_i$ . To estimate the eigenvector centroid  $\Phi_0$ , we compute the singular value decomposition  $\operatorname{svd}(\Sigma_\Phi) = ULV^T$ . We compute the eigenvector centroid as  $\Phi_0 = UV^T$ .

4. To estimate the rotation matrix  $\mathcal{R}_i$ , we compute the sum  $\Sigma_{\mathcal{R}} = \sum_i^N \Phi_i^T \Phi_0$ . We use again the *svd*, obtaining  $\operatorname{svd}(\Sigma_{\mathcal{R}}) = ULV^T$ . Finally, we compute the rotation matrix as  $\mathcal{R}_i = VU^T$ .

The steps 3 – 4 are repeated till convergence, which is usually reached after a low number of iterations (about 5 iterations).

Once the eigenvector centroid is obtained, we can proceed with the computation of the geodesic distances between each mesh of the training set and the centroid itself, using equation 14. This results in the vector  $d_g = \{d_1, d_2, \dots, d_N\}$ . As already introduced in section 3, we assume that the geodesic distances  $d_g$  follow a  $\Gamma$ -distribution. We use 12 and 13 to compute the distributions parameters  $k$  and  $\Theta$ .

The eigenvalues component of the invariant representation is computed in a more straightforward manner.  $\Lambda_0$  of a certain class is computed through 15. The log-normal distribution parameters  $\mu$  and  $\sigma$  for the eigenvalues component of the centroid are computed using 17 and 18.

To summarize, the learning phase of the proposed method applied to a training set with meshes that represent a shape  $S$  produces the manifold centroid  $\{\Phi_0, \Lambda_0\}$  and the distribution parameters  $(\mu_i^S, \sigma_i^S)$ ,  $i = 1, \dots, p$ , and  $(k^S, \Theta^S)$ . The described learning process is repeated for each class of the dataset. Figure 2 shows the distributions yielded by the learning phase introduced in this paragraph.

## 6. Models inference

In section 4, we introduced one of the problems which raises from the computation of a distance metric between two meshes, namely the lack of correspondences between these meshes. In the learning phase of the proposed method this was a minor issue, since the meshes that belong to the same class can be assumed to be quite isometric between them self. It is not the case for the meshes that represent different shapes. In this case, we cast the lack of correspondences problem into an assignment problem (see 23). In order to solve this problem, we make use of another iterative process where we are looking to optimize the parameters  $\mathcal{R}_*$  (a rotation matrix) and  $P_*$  (a permutation matrix). Let  $\Phi_*$  be the eigenvector matrix computed on the Laplacian applied to a mesh  $M_*$ . No prior knowledge about the shape represented by  $M_*$  is assumed. Let  $\Phi_0$  be the eigenvector centroid of a shape  $S$ . After a initialization step where we assign the identity matrix to the rotation matrix  $\mathcal{R}_*$ , we define the optimization process as

1. Computation of the weight matrix  $W$  to be used in our assignment problem,  $W = \Phi_* \mathcal{R}_* \Phi_0^T$
2. The weight matrix  $-W$  is then used as the input of the Hungarian algorithm, whose output is a cost matrix  $C$

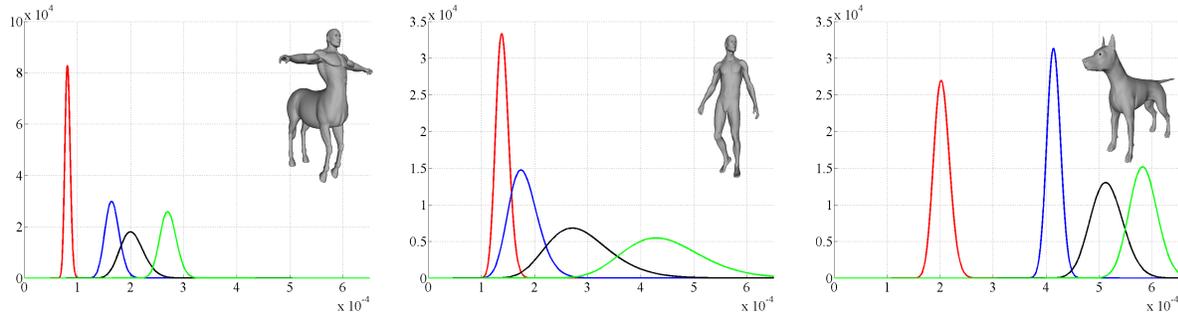


Figure 2. Distributions computed on the invariant representation of 3 shapes of the SHREC'10 dataset. The graphs in the second column of the image show the first four distributions computed on the eigenvalues centroid of the shape represented in the first row (see 19). In particular, the distributions refer to the first (red), second (blue), third (black) and fourth (green) eigenvalues of the corresponding centroids.

and a permutation matrix  $P_*$

3. The rotation matrix  $\mathcal{R}_*$  is the orthogonal matrix which most closely maps the permuted eigenvector matrix  $\Phi_*$  to the eigenvector centroid  $\Phi_0$ , which is the definition of the orthogonal Procrustes problem [43]. So, to estimate the rotation matrix  $\mathcal{R}_*$ , we use the *svd*,  $svd(\Phi_*^T P_* \Phi_0) = ULV^T$ . We compute the rotation matrix  $\mathcal{R}$  as  $\mathcal{R} = VU^T$ .

The process converges after a low number of iterations ( $\approx 3$ ). Once the parameters  $\mathcal{R}_*$  and  $P_*$  are computed, we use 24 to compute the geodesic distance  $d_*^S$  between the eigenvector matrix associated to  $M_*$  and the eigenvector centroid of the shape  $\mathcal{S}$ . Let  $k^S$  and  $\Theta^S$  be the  $\Gamma$ -distribution parameters computed as explained in section 5. The probability density of  $M_*$  with respect to the eigenvector model of the shape  $\mathcal{S}$  is

$$\phi_{d_{(*)\mathcal{S}}} = \Gamma(d_*^S; k^S, \Theta^S) \quad (27)$$

The eigenvalues of the mesh Laplacian are used straightforward after their computation. As introduced in section 3, we assume that all the  $i$ -th eigenvalues follow a log-normal distribution with parameters  $\mu_i$  and  $\sigma_i$ ,  $i = 1, 2, \dots, p$ , with  $p$  the number of eigenvalues used (embedding dimension). So, let  $\Lambda_*$  be the eigenvalues of the mesh  $M_*$ . Let  $\Lambda_0^S$  be the eigenvalue centroid of the shape  $\mathcal{S}$ . The probability density of each eigenvalue of  $M_*$  with respect to the eigenvalues model of the shape  $\mathcal{S}$  is  $\lambda_{d_{(*)\mathcal{S}}} = \{\lambda_{d_{(*)\mathcal{S}}_1}, \lambda_{d_{(*)\mathcal{S}}_2}, \dots, \lambda_{d_{(*)\mathcal{S}}_n}\}$ , where  $\lambda_{d_{(*)\mathcal{S}}_i}$  is defined as

$$\lambda_{d_{(*)\mathcal{S}}_i} = \log \mathcal{N}(\lambda_{*i}; \mu_i^S, \sigma_i^S) \quad (28)$$

Finally, we combine the afore-computed density using 20. For numerical stability, in place of the product of the densities, we compute the sum of the logarithms of the

densities. Hence, the probability density of a mesh  $M_*$  with respect to the models computed on a shape  $\mathcal{S}$  is given by

$$\log p(*|\mathcal{S}) = \log \phi_{d_{(*)\mathcal{S}}} + \sum_i^n \log(\lambda_{d_{(*)\mathcal{S}}_i}) \quad (29)$$

Once the combined density is computed, a decision rule is applied in order to assign the mesh to the most probable class.

It is interesting to note that the lack of correspondences does not affect the construction and use of the eigenvalues model. This is due to the fact that we treat each eigenvalue as an independent variable, while each  $i$ -th eigenvalue of a mesh belonging to the same shape is an independent observation. The only implicit assumption regards the dimension of the embedding  $p$ , which has to be consistent between all the meshes involved both in the learning and in the inference phase. Since the Laplacian eigendecomposition of a mesh is part of the pipeline, the former assumption holds as long as the chosen embedding dimension is smaller than the minimum number of vertex of each mesh involved in the process.

## 7. Experimental results

In this section we evaluate the performance achieved by our method, comparing our results with the current state-of-the-art. Our approach was implemented in MATLAB and the source code is available at the web page of the author <sup>1</sup>. Retrieval performance was evaluated using mean average precision. The proposed method has been applied to several popular datasets in the shape retrieval field.

**SHREC'10 ShapeGoogle** [10] dataset is the aggregation of three public domain collections: TOSCA shapes [2], Robert Sumner's collection of shapes [45] and Princeton

<sup>1</sup><http://www.dsi.unive.it/~gasparetto/publications.htm>

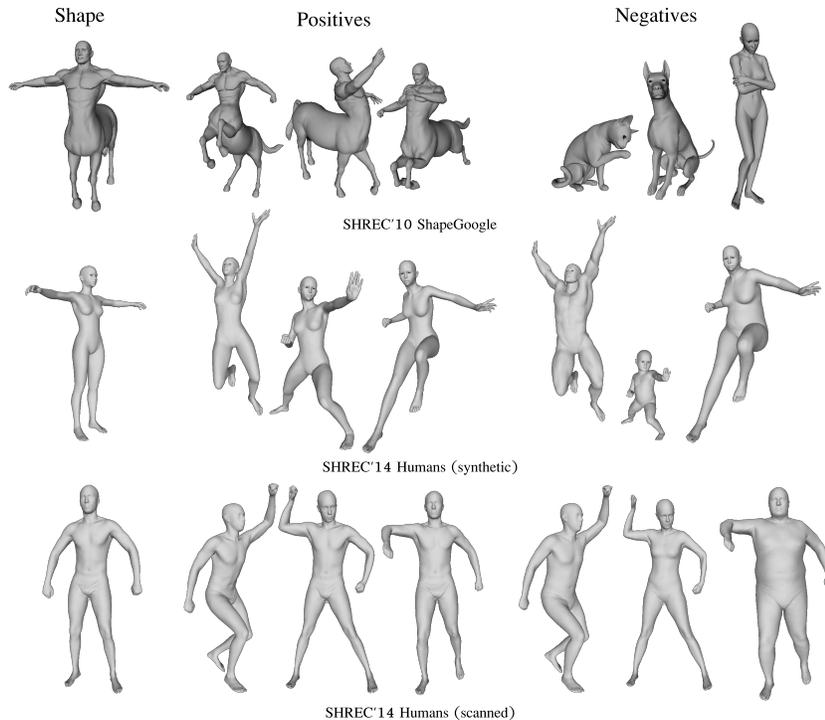


Figure 3. Example of the meshes from three datasets used in our experiments, from the easiest to the hardest (top to bottom). SHREC’10 ShapeGoogle dataset contains shapes of different bi and quadrupeds, SHREC’14 Humans (synthetic) dataset contains a collection of human models created through 3D modelling/animation software. The last row shows the ”scanned” dataset which contains human models built from the point-clouds contained within the CAESAR using the SCAPE method ([4]). In the first column, a template of the shape is shown. The central part shows meshes representing the same shape but in different poses, while the rightmost part shows some negatives (*i.e.*, meshes representing a different shape).

Transformation	VQ [10]	Sup. DL [28]	RMVM
Isometry	98.8	<b>99.4</b>	<b>99.4</b>
Topology	100	100	100
Isometry+Topology	93.3	95.6	<b>99.5</b>
Partiality	94.7	<b>95.1</b>	90.0
Triangulation	95.4	95.5	<b>96.5</b>

Table 1. Comparison of different retrieval methods, in terms of average precision on the SHREC’10 datasets, broken down according to different transformations.

shape repository [1]. It consists of 1184 synthetic meshes, out of which 715 meshes were obtained from 13 shape classes with simulated transformation (55 per shape) used as queries, while the 456 unrelated distractor shapes, which are usually treated as negatives, were not used. The transformations applied to the meshes show different levels of strength. An example of the shapes included in this dataset is shown in figure 3 in the first row. The results, which are shown in table 1, were obtained using a 10-fold test over 100 iterations.

In particular, we subdivided the meshes belonging to the same class in 10 randomly selected groups, using each one as a test set, while the remaining meshes were used as the training set in order to learn the invariant representation of a particular shape and the associated distribution parameters. This dataset contains a different number of representatives (*i.e.* meshes) for each class, hence the number of positives and negatives differs from class to class. For the smallest subset of meshes (representing the same shape), we had 3 positives and 63 negatives, while for the largest we tested our method against 13 positives and 54 negatives. Like in Litman *et al.* work [28], we removed the ‘don’t-care’ ground-truth labels used in the original benchmark (e.g., male and female shapes were considered the same class). Furthermore, in order to make the dataset more challenging, we re-scaled all the meshes to have the same size. We compared the performance of our method (which takes the name of RMVM, *i.e.* Riemannian Metric Variation Model) with respect to several methods that employ Bag-of-Features descriptors. As the table shows, the proposed method obtains a slight performance improvement (or be at least on par)

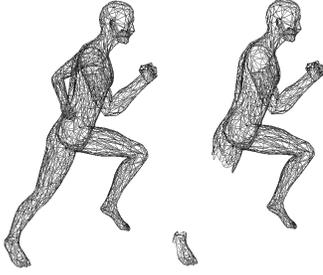


Figure 4. Example of partiality transformation on the SHREC’10 dataset. On the left there is the full mesh, on the right the same mesh after the application of a partiality transformation of strength 3.

Transformation	1	$\leq 2$	$\leq 3$	$\leq 4$
Partiality	100	100	97.5	76.7

Table 2. Average precision on the SHREC’10 datasets, broken down according to the strength of the transformation applied.

Method	Synthetic	Scanned
ISPM [26]	90.2	25.8
DBN [36]	84.2	30.4
R-BiHDM [49]	64.2	64.0
HAPT [14]	81.7	63.7
ShapeGoogle (VQ) [10]	81.3	51.4
Unsupervised DL [28]	84.2	52.3
Supervised DL [28]	95.4	79.1
RMVM	<b>96.3</b>	<b>79.5</b>

Table 3. Comparison of different retrieval methods in terms of mean average precision on the SHREC’14 Humans datasets.

in 4 out of 5 transformations with respect to the current state-of-the-art approaches. We achieved lower classification accuracy with meshes which represent only partially the underlying shape. This is due to the fact that our approach exploits the prior-knowledge about the correspondences between vertices of different meshes. The removal of some parts of the mesh makes our assumption weaker, and this leads to a lower accuracy in the retrieval task. This issue is clear once we analyse the accuracy variation broken down to the strength of the transformation. Table 2 shows the results obtained for each level of strength of the transformation, while figure 4 shows a partiality transformation applied to a mesh.

**SHREC’14 Humans** [36] consists of two different subsets. The first part (synthetic) contained 15 different human models created using DAZ Studio, each in 20 different poses (total of 300 models, figure 3, middle row). The second one (scanned) contained scans of 40 human subjects,

each in 10 different poses (400 shapes in total, figure 3, last row). All shapes were down-sampled to have about  $6 \times 10^3$  triangles. Both datasets are extremely challenging, as they contain geometrically similar human shapes. In particular, the scanned dataset contains human shapes very difficult to distinguish even for a human observer (see figure 3, third row). We employed again a 10-fold test over the datasets. After a random permutation of the datasets, we subdivided the meshes into a training set and a test set. For the training set we used 90% of meshes for each class in order to build our shape invariant representation. For the synthetic dataset test set, we employed 2 positives and 28 negatives per query, while for the scanned dataset test set, the number of positives was 1 and the number of negatives was 39. The process was repeated for each possible subset of the datasets and the whole test was run 100 times. In addition to the same Bag-of-Features approaches seen before, we compared our model with the most accurate methods that participate to the SHREC’14 benchmark. In particular, we compared our approach to works based on Histograms of Area Projection Transform (HAPT) [14], Deep Belief Network (DBN) [36], Intrinsic Spatial Pyramid Matching (ISPM) [26], and Reduced Bi-harmonic Distance Matrix (R-BiHDM) [49]. Table 3 shows the results yielded by those methods. The proposed method performs slightly better with respect to the current state-of-the-art, achieving high retrieval accuracies in both the synthetic and scanned datasets.

Unfortunately, we are not able to conduct exhaustive experiments with regards to the performance achieved by the compared methods, since the performance analysis of the different approaches is not available. Just to have an insight about the performance achieved by the proposed method, we compare with Litman’s approach [28], where the training time was reported to be approximately of 4 hours with a 3.2 Ghz CPU. On our 2.4 Ghz machine, we are able to learn the model of a whole class in about 5s (about 80s for the whole synthetic dataset). The prediction time is computed in 9s, which means that the computation of the density of a mesh with respect to a certain class requires about 0.6s.

## 8. Conclusion

In this work, we proposed a supervised data-driven method for the definition of an invariant representation of an arbitrary shape. Further, we defined a statistical framework based on such representation that can be used to address shape classification tasks. Moreover, we defined an efficient intrinsic metric to approximate the geodesic distance between points over a manifold. One of the limitations of the proposed method involves the knowledge of the correspondence ground-truth between meshes. We tested our method on several standard shape retrieval datasets. Experimental results show that the proposed approach is competitive with the current state-of-the-art for non-rigid 3D shape retrieval.

## Acknowledgements

We gratefully acknowledge Emanuele Rodolà (TU Munich), who provided insight and expertise that greatly assisted the research.

## References

- [1] The princeton shape benchmark. In *Proceedings of the Shape Modeling International 2004*, SMI '04, pages 167–178, Washington, DC, USA, 2004. IEEE Computer Society.
- [2] R. K. A. M. Bronstein, M. M. Bronstein. *Numerical geometry of non-rigid shapes*. Springer, 2008.
- [3] A. Albarelli, E. Rodolà, F. Bergamasco, and A. Torsello. A non-cooperative game for 3d object recognition in cluttered scenes. In *2011 International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission, 3DIMPVT 2011*, pages 252–259, 2011.
- [4] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis. Scape: shape completion and animation of people. *ACM Trans. Graph.*, 24:408–416, 2005.
- [5] M. Aubry, U. Schlickewei, and D. Cremers. The wave kernel signature: A quantum mechanical approach to shape analysis. In *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, pages 1626–1633, Nov 2011.
- [6] M. Belkin, J. Sun, and Y. Wang. Discrete laplace operator on meshed surfaces. In *Proceedings of the Twenty-fourth Annual Symposium on Computational Geometry, SCG '08*, pages 278–287, New York, NY, USA, 2008. ACM.
- [7] A. D. Bimbo and P. Pala. Content-based retrieval of 3d models. *ACM Trans. Multimedia Comput. Commun. Appl.*, 2(1):20–43, Feb. 2006.
- [8] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, Mar. 2003.
- [9] F. Bourgeois and J.-C. Lassalle. An extension of the munkres algorithm for the assignment problem to rectangular matrices. *Commun. ACM*, 14(12):802–804, Dec. 1971.
- [10] A. M. Bronstein, M. M. Bronstein, L. J. Guibas, and M. Ovsjanikov. Shape google: Geometric words and expressions for invariant shape retrieval. *ACM Trans. Graph.*, 30(1):1:1–1:20, Feb. 2011.
- [11] M. Bronstein and I. Kokkinos. Scale-invariant heat kernel signatures for non-rigid shape recognition. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 1704–1711, June 2010.
- [12] T. Darom and Y. Keller. Scale-invariant features for 3-d mesh models. *Image Processing, IEEE Transactions on*, 21(5):2758–2769, May 2012.
- [13] T. Funkhouser and M. Kazhdan. Shape-based retrieval and analysis of 3d models. *Communications of the ACM*, pages 58–64, 2005.
- [14] A. Giachetti and C. Lovato. Radial symmetry detection and shape characterization with the multiscale area projection transform. *Computer Graphics Forum*, 31(5):1669–1678, 2012.
- [15] B. Gong, J. Liu, X. Wang, and X. Tang. Learning semantic signatures for 3d object retrieval. *Multimedia, IEEE Transactions on*, 15(2):369–377, Feb 2013.
- [16] K. Grauman and T. Darrell. The pyramid match kernel: discriminative classification with sets of image features. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 2, pages 1458–1465 Vol. 2, Oct 2005.
- [17] T. Hofmann. Unsupervised learning by probabilistic latent semantic analysis. *Mach. Learn.*, 42(1-2):177–196, Jan. 2001.
- [18] Q.-X. Huang, H. Su, and L. Guibas. Fine-grained semi-supervised labeling of large shape collections. *ACM Trans. Graph.*, 32(6):190:1–190:10, Nov. 2013.
- [19] V. Jain and H. Zhang. A spectral approach to shape-based retrieval of articulated 3d models. *Comput. Aided Des.*, 39(5):398–407, May 2007.
- [20] M. Kazhdan, T. Funkhouser, and S. Rusinkiewicz. Rotation invariant spherical harmonic representation of 3d shape descriptors. In *Proceedings of the 2003 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing, SGP '03*, pages 156–164, Aire-la-Ville, Switzerland, Switzerland, 2003. Eurographics Association.
- [21] D. G. Kendall. Shape manifolds, procrustean metrics, and complex projective spaces. *Bulletin of the London Mathematical Society*, 16(2):81–121, 1984.
- [22] R. Houry, J.-P. Vandeborre, and M. Daoudi. Indexed heat curves for 3d-model retrieval. In *Pattern Recognition (ICPR), 2012 21st International Conference on*, pages 1964–1967, Nov 2012.
- [23] G. Lavou. Combination of bag-of-words descriptors for robust partial shape retrieval. *The Visual Computer*, 28(9):931–942, 2012.
- [24] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 2169–2178, 2006.
- [25] C. Li and A. Ben Hamza. Skeleton path based approach for nonrigid 3d shape analysis and retrieval. In J. Aggarwal, R. Barneva, V. Brimkov, K. Koroutchev, and E. Korutcheva, editors, *Combinatorial Image Analysis*, volume 6636 of *Lecture Notes in Computer Science*, pages 84–95. Springer Berlin Heidelberg, 2011.
- [26] C. Li and A. Ben Hamza. A multiresolution descriptor for deformable 3d shape retrieval. *Vis. Comput.*, 29(6-8):513–524, June 2013.
- [27] C. Li and A. Hamza. Intrinsic spatial pyramid matching for deformable 3d shape retrieval. *International Journal of Multimedia Information Retrieval*, 2(4):261–271, 2013.
- [28] R. Litman, A. Bronstein, M. Bronstein, and U. Castellani. Supervised learning of bag-of-features shape descriptors using sparse coding. In *Computer Graphics Forum*, volume 33, pages 127–136. Wiley Online Library, 2014.
- [29] R. Litman and A. M. Bronstein. Learning spectral descriptors for deformable shape correspondence. *IEEE Trans. Pattern Anal. Mach. Intell.*, 36(1):171–180, 2014.

- [30] R. Lpez-Sastre, A. Garca-Fuertes, C. Redondo-Cabrera, F. Acevedo-Rodrguez, and S. Maldonado-Bascn. Evaluating 3d spatial pyramids for classifying 3d shapes. *Computers and Graphics*, 37(5):473 – 483, 2013.
- [31] D. Macrini, K. Siddiqi, and S. Dickinson. From skeletons to bone graphs: Medial abstraction for object recognition. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8, June 2008.
- [32] A. Mademlis, P. Daras, D. Tzovaras, and M. G. Strintzis. 3d object retrieval using the 3d shape impact descriptor. *Pattern Recognition*, 42(11):2447 – 2459, 2009.
- [33] T. P. Minka. Estimating a gamma distribution, 2002.
- [34] W. Mohamed and A. Ben Hamza. Reeb graph path dissimilarity for 3d object matching and retrieval. *The Visual Computer*, 28(3):305–318, 2012.
- [35] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin. Shape distributions. *ACM Trans. Graph.*, 21(4):807–832, Oct. 2002.
- [36] D. Pickup, X. Sun, P. L. Rosin, R. R. Martin, Z. Cheng, Z. Lian, M. Aono, A. Ben Hamza, A. Bronstein, M. Bronstein, S. Bu, U. Castellani, S. Cheng, V. Garro, A. Giachetti, A. Godil, J. Han, H. Johan, L. Lai, B. Li, C. Li, H. Li, R. Litman, X. Liu, Z. Liu, Y. Lu, A. Tatsuma, and J. Ye. SHREC’14 track: Shape retrieval of non-rigid 3d human models. In *Proceedings of the 7th Eurographics workshop on 3D Object Retrieval, EG 3DOR’14*. Eurographics Association, 2014.
- [37] Z. Qin, J. Jia, and J. Qin. Content based 3d model retrieval: A survey. In *Content-Based Multimedia Indexing, 2008. CBMI 2008. International Workshop on*, pages 249–256, June 2008.
- [38] M. Reuter, F. erich Wolter, and N. Peinecke. Abstract laplacebeltrami spectra as shape-dna of surfaces and solids, 2005.
- [39] M. Reuter, F.-E. Wolter, and N. Peinecke. Laplace-spectra as fingerprints for shape matching. In *Proceedings of the 2005 ACM Symposium on Solid and Physical Modeling, SPM ’05*, pages 101–106, New York, NY, USA, 2005. ACM.
- [40] M. Reuter, F.-E. Wolter, and N. Peinecke. Laplace-beltrami spectra as ‘shape-dna’ of surfaces and solids. *Comput. Aided Des.*, 38(4):342–366, Apr. 2006.
- [41] E. Rodola, S. Rota Bulò, T. Windheuser, M. Vestner, and D. Cremers. Dense non-rigid shape correspondence using random forests. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 4177–4184, June 2014.
- [42] R. M. Rustamov. Laplace-beltrami eigenfunctions for deformation invariant shape representation. In *Proceedings of the Fifth Eurographics Symposium on Geometry Processing, SGP ’07*, pages 225–233, Aire-la-Ville, Switzerland, Switzerland, 2007. Eurographics Association.
- [43] P. Schnemann. A generalized solution of the orthogonal procrustes problem. *Psychometrika*, 31(1):1–10, 1966.
- [44] J.-L. Shih, C.-H. Lee, and J. T. Wang. A new 3d model retrieval approach based on the elevation descriptor. *Pattern Recognition*, 40(1):283 – 295, 2007.
- [45] R. W. Sumner and J. Popović. Deformation transfer for triangle meshes. *ACM Trans. Graph.*, 23(3):399–405, Aug. 2004.
- [46] J. Sun, M. Ovsjanikov, and L. Guibas. A concise and provably informative multi-scale signature based on heat diffusion. In *Proceedings of the Symposium on Geometry Processing, SGP ’09*, pages 1383–1392, Aire-la-Ville, Switzerland, Switzerland, 2009. Eurographics Association.
- [47] J. Tangelder and R. Velkamp. A survey of content based 3d shape retrieval methods. In *Shape Modeling Applications, 2004. Proceedings*, pages 145–156, June 2004.
- [48] R. Toldo, U. Castellani, and A. Fusiello. The bag of words approach for retrieval and categorization of 3d objects. *The Visual Computer*, 26(10):1257–1268, 2010.
- [49] J. Ye, Z. Yan, and Y. Yu. Fast nonrigid 3d retrieval using modal space transform. In *Proceedings of the 3rd ACM Conference on International Conference on Multimedia Retrieval, ICMR ’13*, pages 121–126, New York, NY, USA, 2013. ACM.