Vis Comput (2014) 30:1247–1259
DOI 10.1007/s00371-014-0926-5

ORIGINAL ARTICLE

3D shape retrieval and classification using multiple kernel learning on extended Reeb graphs

Vincent Barra · Silvia Biasotti

Published online: 15 March 2014

Abstract We propose in this article a new 3D shape classification and retrieval method, based on a supervised selection of the most significant features in a space of attributed extended Reeb graphs encoding different shape characteristics. The similarity between pairs of graphs is addressed through both their representation as set of bags of shortest paths, and the definition of kernels adapted to these descriptions. A multiple kernel learning algorithm is used on this set of kernels to find an optimal linear combination of kernels for classification and retrieval purposes. Results on classical data sets are comparable with the best results of the literature, and the modularity and flexibility of the kernel learning ensure its applicability to a large set of methods.

Keywords 3D Object retrieval · Classification · Extended Reeb graph · Kernel learning

1 Introduction

The fast advancement of tools for acquisition and storage of 3D models has led to a rapid increase in the number and size of the models available on the internet and domain-specific databases. Organizing these 3D models is becoming an acute issue for numerous applications, including CAD, medical imaging, molecular biology, architecture or game design.

In this scenario, one fundamental problem is how to select and combine different features. If a-priori information on the object type and its description context is given, the selection of the features is facilitated. In addition, the intuition about the relation between the elements belonging to the same class of objects and their most salient features fades away with the increase of the shape complexity: the challenge is then to recognize from a large set of features, the subset that better characterizes the class itself. Also, shape features are often of different scales and therefore their combination is not a matter of linear combination.

In its generality, the question above cannot be answered because of the infinite number of features and contexts. In this paper we propose an effective 3D shape classification and retrieval method that selects the most significant shape features from a larger set considering as shape context the elements belonging to the same class. The construction of the representative set can be regarded as a machine learning task that uses a supervised learning technique to capture the the shape characteristics shared by the object classes. We therefore select those features that give an information content which is relevant for shape classification, without jeopardizing the retrieval performance of the method [1]. To deal with a semantic representation able to couple global and local features, we adopt, similarly to [2], a shape description that combines the overall shape structure (coded in a topological graph) with a local geometric description (the spherical harmonic indices of the shape parts) [3].

The 3D shape classification and retrieval problem is addressed as a search in a space of attributed graphs encoding different shape characteristics through a similarity measure able to handle both the graph structure and the geometric attributes associated to nodes and edges. The most efficient

V. Barra (✉)
LIMOS, Clermont-Université, Université Blaise Pascal,
BP 10448, 63000 Clermont–Ferrand, France
e-mail: vincent.barra@isima.fr

V. Barra
LIMOS, CNRS, UMR 6158, 63173 Aubiere, France

S. Biasotti
CNR, Istituto di Matematica Applicata e Tecnologie
Informatiche 'E. Magenes', Genova, Italy
e-mail: silvia@ge.imati.cnr.it

linear combination of features is then computed using supervised learning. More precisely, 3D models are represented by bags of shortest paths defined over a set of extended Reeb graphs (ERGs) computed from a set of functions. For each of these functions, the similarity between pairs of corresponding ERGs are computed using a so called kernel, implicitly defining the similarity between the models. Given this set of kernels, a Multiple Kernel Learning algorithm is used to find an optimal linear combination for classification and retrieval purposes. This linear combination can be class-specific or computed for a whole database. This paper extends the work presented in [4], exhibiting more results on a higher number of data sets and kernels. In addition, the retrieval scores are added and compared with the existing benchmarks, showing that the results we achieve are comparable with the literature, and the modularity and flexibility of the kernel learning ensure its applicability to a large set of methods.

The paper is organized as follows: Sect. 2 reviews 3D shape classification and feature selection methods. Section 3 introduces the elements of our method, i.e. ERGs and kernel on graphs, while Sect. 4 develops the method we propose, based on multiple kernel learning. Section 5 presents and analyses the classification and retrieval performance of the optimal kernel over two benchmark data sets [5,6].

2 Prior work

For efficient comparison and similarity estimation, 3D models can be represented with a set of meaningful descriptors that encode the salient geometric and topological characteristics of their shapes. The objects in the database are then ranked according to their distance to the descriptors of the query model, see [7–9] for an overview of 3D shape retrieval methods, or e.g [10] who exploited the structure of Reeb graphs associated with geometrical signatures to retrieve part-assembled 3D objects. The use of data-driven approaches to learn the salient features of a 3D model and the similarity measure that is suitable to the data set can improve significantly the performance of classification and retrieval algorithms. Even though several methods for shape comparison have been proposed, only few methodologies address the issue of identifying descriptions that capture the shape features shared by a class of models [11,12]. On one hand, a solution to learn inter-class distances is to manually feed the retrieval system with relevance feedback techniques [13–15]. On the other hand, to automatically address the selection of class features, machine learning techniques like boosting and support vector machines, have been adopted either using images [16] or shape descriptions [17–19]. For instance the AdaBoost algorithm [20] has been used in [11] to select relevant views of 3D objects with respect to the light field descriptor.

Other classifiers based on semi-supervised learning, dimensionality reduction, and probability have been successfully exploited for shape classification. For instance, in [17] Support Vector Machine is used to cluster 3D models with respect to semantic information. In Ohbuchi and Kobayashi [21] shape classifiers are obtained as a linear combination of individual classifiers and using non-linear dimensionality reduction. In Shilane and Funkhouser [22], relevant local shape descriptors are selected through a multivariate Gaussian distribution and collected to define a priority-driven search for shape retrieval. Marini et al. [23] use Adaboost and SVM as tools to automatically select the frequencies of the Laplacian spectrum of 3D shapes that are relevant for classification. There a selection of the eigenvalues is used to represent each class by means of those features that characterize the class members and that are discriminative with respect to non-member 3D objects. Unfortunately, there is not an explicit correspondence between the eigenvalues and the geometric meaning of the features selected by the boosting algorithm.

Recently, Tabia et al. [24] approached 3D classification using a graph-based representation to drive the decomposition of the shape into significant parts. In this work, parts are extracted from training objects and grouped into part classes. A belief function is to defined to map the part classes into the object classes. New 3D objects are classified mapping their parts into the learned part classes and inheriting the object classification from the learned belief function.

Moreover, the introduction of learning schemes for the construction of optimized spectral descriptors and metric learning [25] and the formulation of the shape differences based on the concept of shape maps [26–28] translates the problem of explicitly locating and describing the differences among two shapes in terms of a linear algebraic framework. With this aim, also the comparison of shape properties represented as real functions and their automatic grouping via clustering approaches has been recently explored [29], selecting a set of significant maps that would act as the filter to build further shape descriptions.

Differently from the previous approaches, our aim is to automatically build a new composite description from a set of elementary kernels. In recent years, several methods have been proposed to combine multiple kernels instead of using a single one. These different kernels may correspond to using different notions of similarity or may be using information coming from multiple sources (different representations or different feature subsets). In particular [30] provides a taxonomy of existing methods for Multiple Kernel Learning and reviews several algorithms.

3 ERG-based shape description and comparison

In this section we overview how to extract a graph-based description from a 3D shape, namely an ERG, and to

define kernels on these descriptions. In the following, $G = (V, E, \delta_V, \delta_E)$ is an undirected, labelled graph of nodes V and edges E where $\delta_V : V \to \mathbb{R}^p$, $\delta_E : E \to \mathbb{R}^d$ associate to every node and edge numerical attributes. We assume that G does not have multiple edges from one node to another and and does not contains no negative cycles.

A path p is a sub-graph of G defined by a sequence of l nodes $p = (v_1 \dots v_l)$ such that for each i, $(v_i v_{i+1}) \in E$. The length of p is defined as $l(p) = f(\delta_V(v_i), v_i \in p, \delta_E(e), e \in p)$, where f is a real function defined on the set of nodes and edges. p is said to be the shortest path between v_i and v_j if for all path p' between these two nodes, $l(p) < l(p')$.

3.1 Extended Reeb graph

The ERG is a 3D shape descriptor that fulfils the graph requirements on G of being an undirected and labelled graph [3]. The ERG represents a surface with or without boundary through a finite set of contours of a function f defined on the surface. Like other methods based on the Reeb graph [31], the ERG reflects the properties of the real function f, see [32] for an overview of Reeb graph-based descriptions.

In practice, the ERG approximates the Reeb graph through a set of isocontours of f. On the basis of these samples the surface is characterized in terms of *critical* (maxima, minima and saddles) and *regular* areas: the nodes of the ERG correspond to the critical areas while its arcs are detected tracking the evolution of the level sets [3]. To convey the geometrical properties in the description, the pure topological graph is coupled with attributes that describe the shape parts that correspond to nodes and edges of the graph. The coding strategy adopted in this paper is the detailed in [1]. Figure 1 depicts the ERG and the model parts associated to nodes and edges.

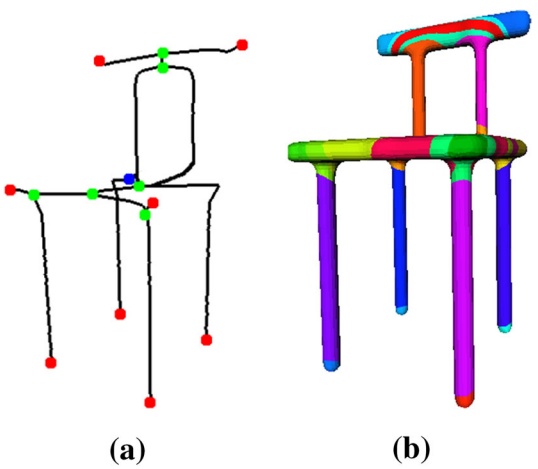

Fig. 1 An ERG (**a**) and the corresponding object partition (**b**). *Blue* = minimum, *red* = maxima and *green* = saddles. *Different colors* in the model (**b**) represent the different parts *(each part corresponds to a node or an edge)*

From the storage point of view, each node v of G corresponds to one critical area (a maximum, a minimum or a saddle-like area) and the edge $e = (v_1, v_2)$ is associated to the surface portion bounded by the regions r_1 and r_2 associated to v_1 and v_2, respectively. For each node, the attributes $\delta_V(v)$ and $\delta_E(e)$ correspond to the spherical harmonic values [33] of the related parts. If the regions r_1 and r_2 corresponding to v_1 and v_2 are adjacent (i.e., r_1 and r_2 share a boundary element), $\delta_E(e) = 0$. In case of multiple edges $(e_1, \dots, e_n)$ between v_1 and v_2, the attributes $\delta_E(e_i)$ codes the spherical harmonic indices related to the surface portions (e.g., the rear segments of the chair in Fig. 1b).

The most interesting aspect of the ERG is its parametric nature with respect to f: changing f allows different descriptions of the same surface that highlight different "local" features while preserving the "global" topological structure of the surface. Here, we focus on ten scalar functions, see Sect. 4.1 that capture different geometric properties, namely extrinsic (based on Euclidean metric) and intrinsic (based on Riemannian metric) invariants of the shape. These functions (and the derived kernel) are the possible "words" we can adopt to describe an object; then we combine kernels with the context (in our case the classes of the data set) to infer a new description (kernel) that keep the functions that better capture the intra-class variability, and therefore the semantics associated to that class.

3.2 Graph similarity and kernels on graphs

Given two ERG graphs G and G', the aim here is to find a similarity measure that can be geometric, topological or can manage both of these notions, provided that sufficient information is carried out in the graphs. If numerous methods derived from graph matching techniques are available, several authors recently proposed to use kernels [34,35] either for retrieval and indexing [36], classification [37] or scene analysis [38]. We herein base the similarity of graphs on the similarity of their representations by bag of paths.

The idea of building kernel between graphs originated from [39] and was then extended by [40] and [34]. It mainly consists in defining a specific kernel for both valued nodes and edges, and to gather the corresponding results into a high level kernel, managing the graph structure, and defining a dot product between graphs.

The kernel formulation is expressed in terms of bag of paths [41]. Two graphs $G = (V, E, \delta_V, \delta_E)$ and $G' = (V', E', \delta_{V'}, \delta_{E'})$ are described as bag of paths H and H', and the similarity $K(G, G')$ between G and G' is assessed through the similarity between H and H', using a predefined

kernel K_c on paths:

$$K(G, G') = \sum_{h \in H} \sum_{h' \in H'} K_c(h, h') P(h|G) P(h'|G') \quad (1)$$

where $P(h|G)$ (respectively $P(h'|G')$) is the probability of walking along h (resp. h') on G (resp. G'). The K_c function is computed from two basic kernels K_V and K_E defined on V and E. In the case of Gaussian kernels:

$$\forall (v, v'),\ K_V(v, v') = e^{-\frac{1}{2}\left(\delta_V(v) - \delta_{V'}(v')\right)^T \Sigma_V \left(\delta_V(v) - \delta_{V'}(v')\right)}$$
$$\forall (e, e'),\ K_E(e, e') = e^{-\frac{1}{2}\left(\delta_E(e) - \delta_{E'}(e')\right)^T \Sigma_E \left(\delta_E(e) - \delta_{E'}(e')\right)}$$

where $\Sigma_V = diag\left(\frac{1}{\sigma_i}\right)$ (resp. Σ_E) is diagonal, and σ_i is the bandwidth of the i^{th} label of v (resp e).

Then if $h = (v_1 \dots v_l)$, $h' = (v'_1 \dots v'_l)$, with $v_{i-1}v_i = e_i$ and $v'_{i-1}v'_i = e'_i$:

$$K_c(h, h') = K_V(v_1, v'_1) \prod_{i=2}^{l} K_V(v_i, v'_i) K_E(e_i, e'_i)$$

The probabilities $p(h|G)$ can easily be computed with $P(h|G) = p_S(v_1) \prod_{i=2}^{l} p_T(v_i|v_{i-1}) p_E(v_l)$, where p_S is the probability of initial visit, p_E is the ending probability and p_T is the transition probability between nodes in G. It is thus theoretically possible to compute probability values even for a large number of nodes and thus paths (see e.g. [34]). Nevertheless, some of the paths in the graphs are not relevant and do not need to be explored. We propose then to filter out the set of all paths and use only the shortest paths of length less or equal to a maximal length $L_{\max}$. We thus compute K by only retaining the best comparison values for all shortest paths, leading to the max matching kernel formulation [42]:

$$K(G, G') = \frac{1}{2}\left[\hat{K}(H, H') + \hat{K}(H', H)\right] \quad (2)$$

where $\hat{K}(H, H') = \frac{1}{|H|} \sum_{h \in H} \max_{h' \in H'} K_c(h, h')$, and H (resp. H') is the set of shortest paths in G (resp. G') of maximal length $L_{\max}$. But since this definition does not induce a definite positive kernel, we use the approximation proposed in [43]:

$$\max_{h' \in H'} K_c(h, h') \approx \sum_{h' \in H'} K_{dc}(h, h') \quad (3)$$

with $K_{dc} = exp\left(-\frac{d_c(h,h')^2}{2\sigma^2}\right)$, a kernel defined by the distance d_c induced by $K_c, \sigma > 0$. The resulting kernels are easy to compute, retain expressivity and are still positive definite. Finally, considering shortest paths between vertices naturally prevents from tottering phenomenon.

4 Learning kernels from functions

Our approach automatically correlates kernels with respect to the classes of a database. If we think to each kernel as an user's filter of the features of a data set, the selection of the kernels that better hold retrieval implicitly defines the class complexity and the invariants that characterize it. In our framework it is thus possible to compute a class-specific function (a kernel) and for each class to learn a kernel using MKL and a one class SVM. Therefore, similarly to previous approaches [2,11], our solution is data dependent, building upon data set and classification, and includes the context in the learned kernel. The set of descriptions is not necessarily orthogonal and admit redundancy and overlapping in the feature space [16].

4.1 Real functions

The role of a single function f is to convey the most significant geometric information and act as a filter of the features that will be stored in the shape descriptor [44]. The set $F = \{f : S \to \mathbb{R} | f continuous\}$ is infinite, anyway an appropriate selection of the functions is necessary to make the descriptor suitable for shape matching issues: for instance, the function f has to be invariant from object rotation, translation and scaling. In the large number of functions available in the literature, we are considering a subset of functions that is significant for shape retrieval [29]:

- the distance from the barycentre B of the object, $Bar(p) = d_E(p, B)$, $p \in S$ and d_E represents the usual Euclidean distance (Fig. 2a);
- the distance from the axis $\boldsymbol{\alpha}$, $MSA(p) = d_E(p, \boldsymbol{\alpha})$ where $\boldsymbol{\alpha} = \frac{\sum_{p \in S}(p-B)\|p-B\|}{\sum_{p \in S}\|p-B\|^2}$ [45], where B is the barycentre of S and α is the principal vector, i.e. it points towards the main shape direction (Fig. 2b);
- the function $MSANorm(p) = \|\boldsymbol{\alpha} \times (p - B)\|$, $p \in S$, $\boldsymbol{\alpha}$ and B are the same as above (Fig. 2c);
- an average of the geodesic distances, $Geodesic(p) = \frac{\sum_{v \in S} g(v,p)}{\max_{v \in S} g(v,p)}$, where $g(v, p)$ is the geodesic distance between a sampled vertex v and p, in the experiments we consider a uniform re-sampling R of S made of 256 vertices and $v \in R$, [46] (Fig. 2d);
- the first three (ranked with respect to the increasing eigenvalues), non-constant eigenfunctions of the Laplace-Beltrami operator of the mesh computed according to [47], $LAPL_i, i = \{1, 2, 3\}$, (Fig. 2e–g);
- a mix of the first three eigenfunctions of the Laplace-Beltrami operator obtained according to the rule: $MIX_{i+j-2} = (LAPL_i)^2 - (LAPL_j)^2$, $i = \{1, 2\}$, $j = \{2, 3\}$, $i \neq j$ (Fig. 2h–j).

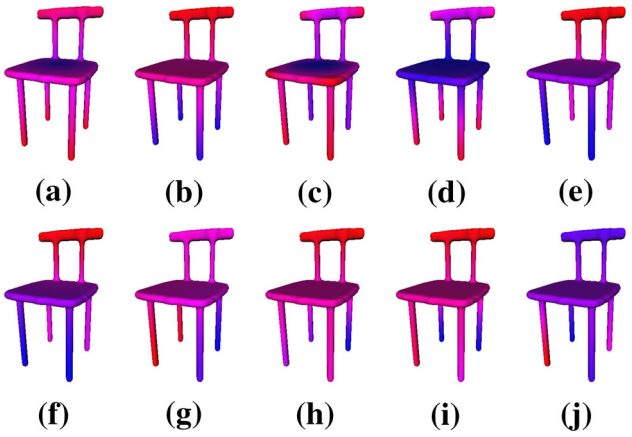

Fig. 2 The set of real functions in our framework. *Colors* represent the function, from low (*blue*) to high (*red*) values

The choice of these functions is motivated by the different behaviour they have with respect to different geometric aspects, each one reflecting either intrinsic or extrinsic shape features. For instance, the distance from the barycentre highlights the distribution of the object with respect to its barycentre. Therefore such a function is rotation invariant with respect to rotations around the barycentre but sensitive to pose variations. Similarly the distances *MSA* and *MSANorm* are independent of rotations and symmetries with respect to the main shape direction $\boldsymbol{\alpha}$. On the contrary the geodesic-based and the Laplacian-based functions are isometry-invariant and therefore pose invariant because they approximate the intrinsic Riemannian metric of the surface [48]. In this way, protrusions and hollows are emphasized, also at different scales in the case of Laplacian-based functions, and the graph representation is independent of the different articulations of the objects.

As quantitatively inferred in [29], these functions provide a significant "shape basis" and a mix of the different properties (rigid or isometry invariant) that guarantees that different shape features are coded. In addition, we highlight that the insertion in the loop of new functions simply influences the type of invariance detected from the method without modifying the global framework of our method.

4.2 The simpleMKL algorithm

We think as in [49] that "no single descriptor is capable of providing fine grain discrimination required by prospective 3D search engines". We then design the algorithm such that several structural/feature based kernels can be combined to enhance the retrieval result and to include the context (i.e. the intra-class correlation) in the selection of the most significant kernels. In this study, the combination is a simple convex combination of elementary kernels, and the corresponding combination method is a multiple kernel Learning algorithm (*simpleMKL*, [50]). The advantage of this algorithm, compared to other Multiple Kernel Learning algorithms [30] is that training time complexity is low, the number of stored support vectors is small and the solution is sparse (defined by the number of used kernels).

Multiple Kernel Learning (MKL) was first introduced in [51], and was then enhanced and refined (e.g [50,52]). The idea behind MKL is to look for a different solution of the learning problem: classically the solution is written in the form $f(x) = \sum_{i=1}^{l} \alpha_i^* K(x, x_i) + b^*$, where α_i^* and b^* are some coefficients to be learned from examples x_i and $K(.,.)$ is a given definite kernel associated with a reproducing kernel Hilbert space (RKHS) H. Lanckriet et al. [51] proposed to consider that $K(.,.)$ is a convex combination of basis kernels K_m

$$\begin{aligned} &K(x, y) = \sum_{m=1}^{M} d_m K_m(x, y) \\ &\text{with } d_m \geq 0, \sum_{m=1}^{M} d_m = 1, \end{aligned} \tag{4}$$

where M is the total number of elementary kernels. MKL learnt both the α_i^* and d_m in a single optimization problem, and [50] proposed to solve the equivalent problem:

$$\begin{aligned} \underset{\{f_m\},b,\xi,d}{\text{Minimize}} \quad & \frac{1}{2} \sum_m \frac{1}{d_m} \parallel f_m \parallel^2_{\mathrm{H}_m} + C \sum_i \xi_i \\ \text{subject to} \quad & y_i \sum_m f_m(x_i) + y_i b \geq 1 - \xi_i, \forall i, \\ & \xi_i \geq 0, \forall i, \\ & \sum_m d_m = 1, d_m \geq 0, \forall m, \end{aligned} \tag{5}$$

where H_m are the RKHS associated to each $K_m(.,.)$ and each f_m belongs to a different H_m. It is assumed that one looks for a decision function of the form $f(x) = \sum_m f_m(x)$. This problem can be extended to a one class problem, and we used this approach in the following using as K_m the kernels on ERG computed from the functions detailed in Sect. 4.1.

5 Results

The results shown in this Section mainly focus on the capability of the method of learning the most relevant features of each class rather than in the absolute retrieval and classification performance of this specific technique.

5.1 Data sets

We evaluate the performance of our method on two data sets:

- The SHREC'07 database of the SHape REtrieval Contest [5]. The collection is composed of 400 watertight mesh models, subdivided into 20 classes of 20 elements each. Ground truth was manually established so that the classes exhibit sufficient and diverse variation, from pose change to shape variability in the same semantic group.
- The SHREC'08 database of 646 watertight models of the [6] contest. These models were classified, and released as training, testing and query data. To reflect both functional and geometric similarity this data set has three different levels of categorization: from coarse to fine. At the *coarse* level, objects are classified according to semantic criteria, besides their shape; at the *intermediate* level, the classes are subdivided according to both functionality and shape features; while at the *fine* level, the classes are further partitioned on the basis of the geometric similarity. For instance, at the coarse level some objects were classified into the furniture class. At the intermediate level, these same objects were further divided into tables, seats and beds. At the fine level, the objects were classified into chairs, armchairs, stools, sofa and benches.

ERG graphs were extracted from these models using functions defined in Sect. 4.1 and all labelled using the spherical harmonics on both nodes and edges. The ERGs are computed automatically dividing the image of the function f in 16 intervals uniformly distributed. The kernel similarity measure was applied on these labelled graphs for the retrieval process. In the following, each model in each data set was used in turn as a query against the remaining part of the database. For a given query, the goal of the track was to retrieve the most similar objects.

5.2 Quantitative comparison

We propose to compare the results of our method, we call MKLERG in the following, with the state of the art. For this, several standard measures were used:

- Precision/recall curves [53]. Precision is the fraction of the relevant retrieved objects to a given query, and recall is the fraction of the relevant objects which have been retrieved from the database.
- Average dynamic recall $ADR = \sum_{i=1}^{q} \frac{R_i}{i}/q$, where q is be the number of matches in the ground truth and R_i is the number of retrieved relevant items within the first i retrieved items [54] .
- Percentage of success for the first (PF), sometimes referred to as the Nearest Neighbor (NN), is the probability of the first element in the return list to be relevant to the given query, and average it over the whole set of queries or over each class.
- The mean average ranking (MR), computed by averaging the retrieval ranking of all relevant items when considering a particular query.
- The last place ranking (LR), defined as $L = 1 - \frac{\rho - i}{N - i}$ where ρ is the rank at which the last relevant object is found, i is the number of relevant items and N the size of the database.
- First-tier (FT) and Second-tier (ST) are the percentages of models in the query's class that appear within the top $|C| - 1$ (FT) or $2(|C| - 1)$ (ST) matches, where $|C|$ is the cardinality of the class.
- Relaxed classification accuracy (RCA), defined as the number of correct classifications as a function of the class rank. The RCA at rank k is defined as $RCA(k) = \sum_{i=1}^{k} n(i)/N$ where $n(i)$ is the number of correct classifications at eht i-th rank, and N the total number of queries;
- Sensitivity (also known as the true positive rate, or the recall rate) measures the proportion of actual positives which are correctly identified as such;
- Specificity (also called the true negative rate) measures the proportion of negatives which are correctly identified as such.

5.3 Parameters estimation

The ERG graphs we processed have a maximal number of nodes less than 100. In order to find an appropriate $L_{\max}$

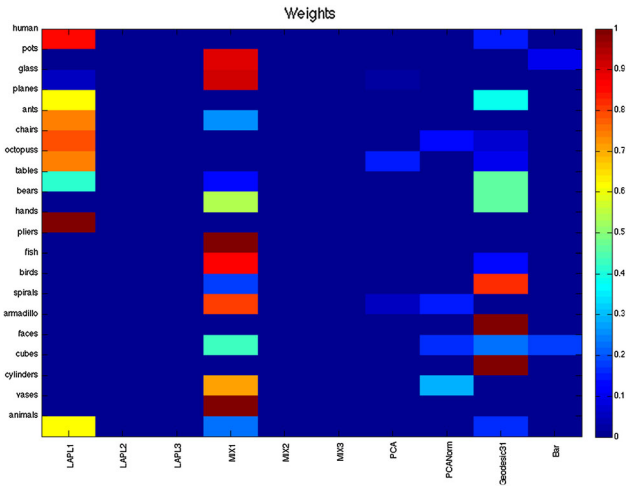

Fig. 3 Weights of the elementary kernels for each class of the SHREC'07 database

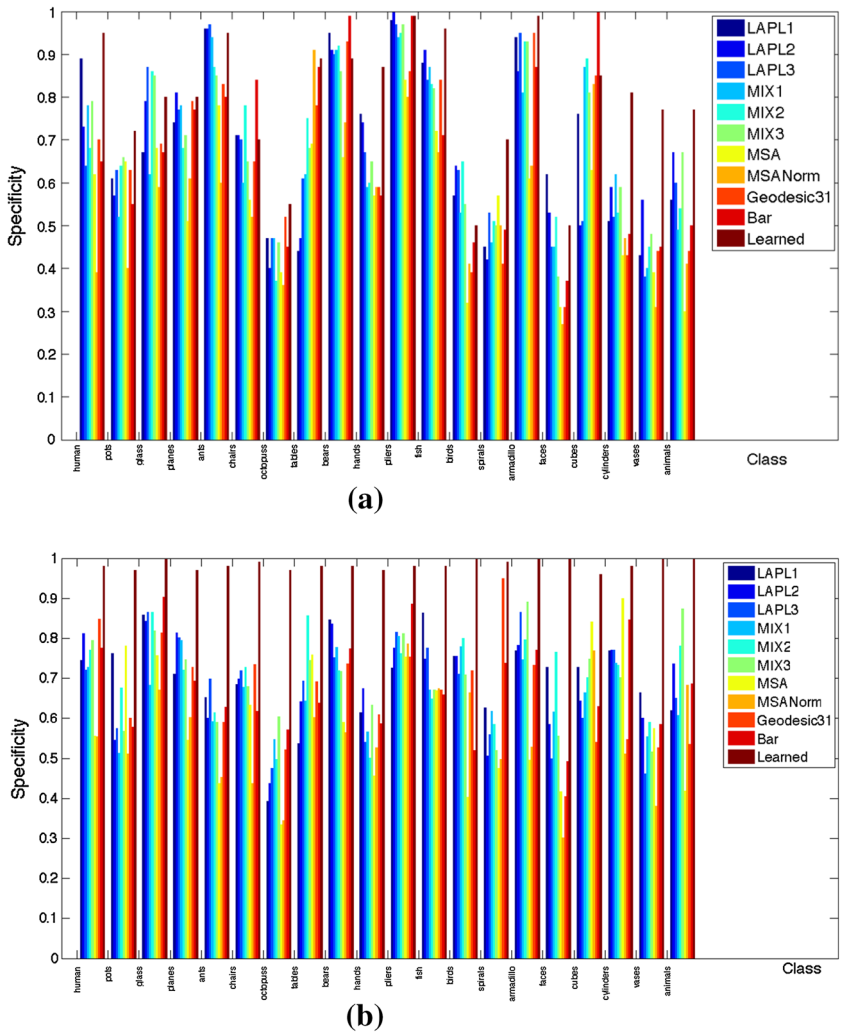

Fig. 4 Sensitivity (**a**) and specificity (**b**) on the SHREC'07 database. Results for the elementary and the learned kernels

value, we used a leave-one-out, cross-validation procedure on the SHREC databases. The range of possible values was from 0 (only nodes are considered) to the number of nodes in the graph. As the considered path length increased (a structural information was added in the graph description), the retrieval rate and the quantitative indexes increased then decreased when a given path length was reached. Any value of $L_{\max}$ ranging in $[\![3 \cdots 7]\!]$ gave both good retrieval results and low complexity, and we performed all our experiments with $L_{\max} = 5$. The other parameters that have to be estimated are related to the matrices Σ_E and Σ_V that appears in the definition of K_V and K_E in Sect. 3.2. These parameters express the bandwidths σ_i of the i^{th} spherical harmonics of v (resp. e). Accurately defining these bandwidths is crucial since they allow to define the intensity of the corresponding kernel. Several methods can be used to determine these values. However, to be database-specific, we choose a cross-validation approach and we apply a leave-one-out procedure data set to determine the best values for the bandwidths.

Several experiments were performed using as training set a percentage ranging from 10 to 50 % of the models in each class. We present in this article the results obtained using half

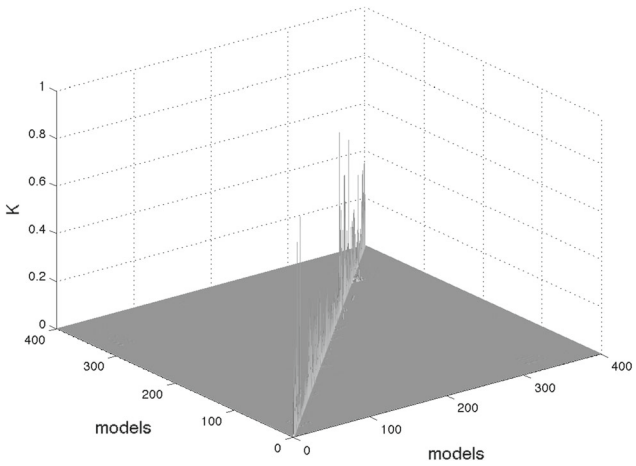

Fig. 5 Kernel matrix on the SHREC'07 database

Table 1 ADR, mean average (MR) and last (LR) ranking indexes, PF, PS and comparison to the state of the art—SHREC'07 database

Ideal	ADR	MR	LR	FT	ST
	1	10.5	1	100	100
MKLERG	0.7791	21.07	0.904	100.00	92.01
[55]	0.7931	45.90	0.60	100	93.97
[56]	0.7206	56.97	0.50	100	92.81
[57]	0.7795	51.93	0.60	100	95.87
[58]	0.7546	53.59	0.55	100	93.33
[59]	0.8577	30.31	0.74	100	97.68
[1]	0.7931	20.49	0.892	100	100

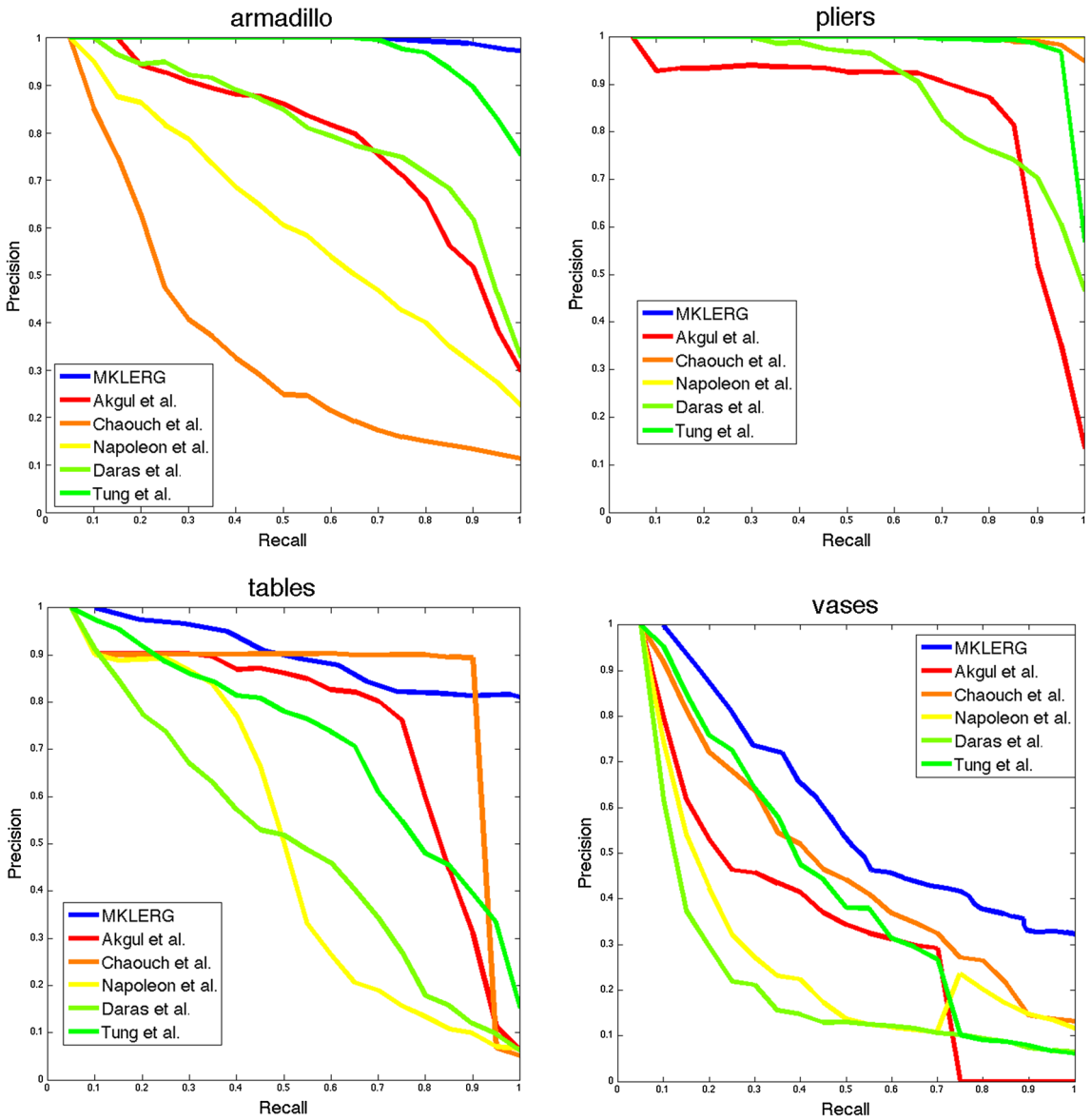

Fig. 6 Precision/Recall graphs for 4 of the 20 classes of SHREC'07 watertight database. Comparison to the state of the art

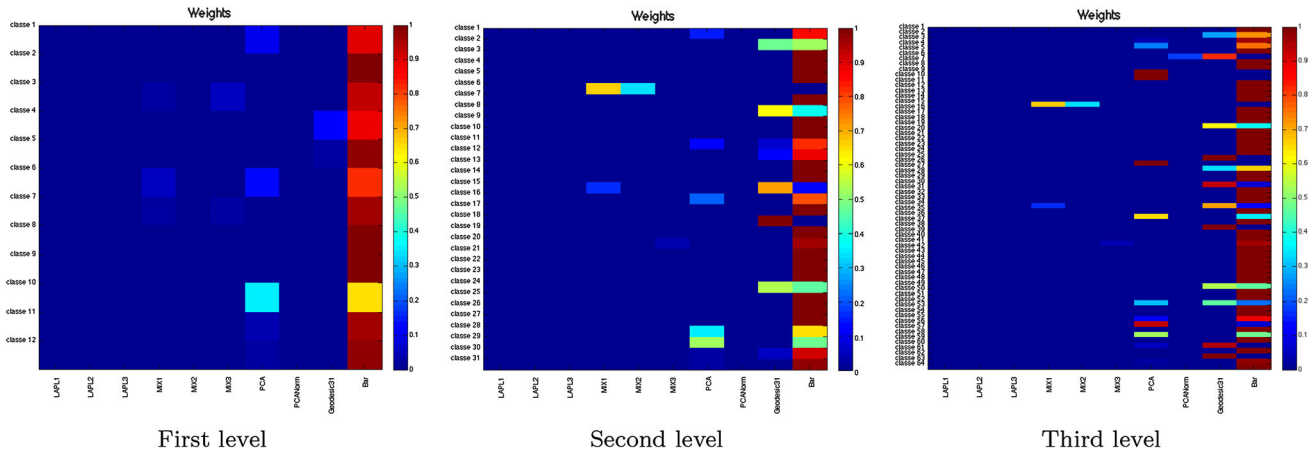

Fig. 7 Weights of the elementary kernels for each class of the SHREC'08 database for the 3 levels of categorization

of the elements in each class but roughly the same results were obtained with the 20 % of the models. With the 25 % of the class, there was no overfitting and most of the other 75 % of models was successfully retrieved thus having a good generalization capacity.

5.4 Results on SHREC'07

Each class of the SHREC'2007 database was processed separately using a one class SVM inside MKL. MKL thus faces a 2-class classification problem, and outputs the corresponding optimal convex combination of elementary kernels. In the experiments described in this section, the learning step was performed using 10 models per class (out the 20 available for each class). Figure 3 presents the weights of this convex combination for all classes.

The choice of the weights mostly depends on the intra-class variability. Anyway, LAPL1 and MIX1 are the most present functions for all the classes because many non-rigid deformations are present in the classes. Rigid functions such as barycentre and MSA are more discriminative when the intra-class variability is smaller, e.g. armadillo, faces, etc..

The experimental results show that the optimal kernel is based on a convex combination of a few single kernels. The comparison of the performance of the optimal kernel with respect to the single ones was done analyzing the sensitivity and the specificity over each class. Figure 4a presents the sensitivity computed for both each elementary kernel and for the learned one (red bold curve). Figure 4b shows the same results for the specificity. These two quantitative indexes were improved for almost every class, with a dramatic improvement for the specificity.

Figure 5 shows the optimal kernel matrix provided by the weights learned. The overall performance in terms of retrieval and classification rates over the whole data set is comparable or overcome the results on the literature [5] and

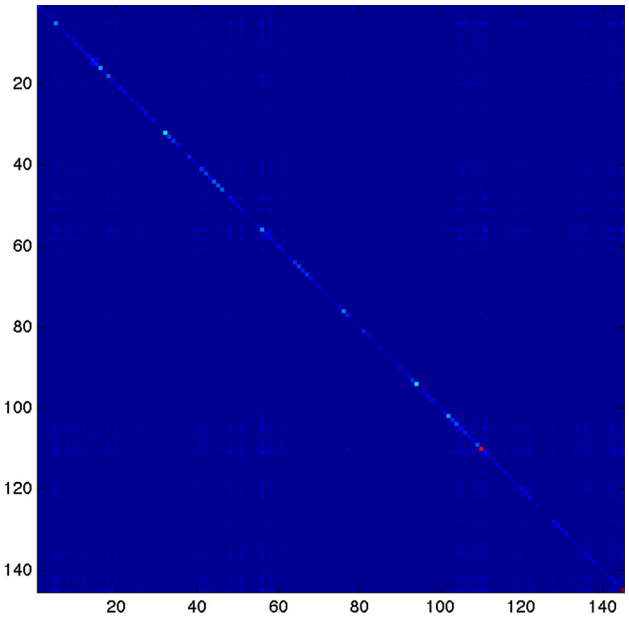

Fig. 8 Optimal kernel of the SHREC'2008 data set—third level of categorization on the training set

a flat averaging of the functions [1]. More precisely, in order to compare with the state of the art, we processed quantitative indexes of the retrieval. Table 1 presents the values of ADR, MR, LR and PF obtained on the SHREC'07 database by state-of-the-art methods [1,5] and ours (MKLERG). Note that for this latter comparison, less information was used as inputs of the algorithm, giving roughly the same performances than [1].

Figures 6 presents the Precision/Recall graphs 4 of the 20 classes of the database (to be comparable with [5]), and superimposes the curves obtained using MKLERG with the five state-of-the-art methods. Curves shifted upwards and to the right indicate a superior performance.

We wanted to make consideration not only on the average performance on the whole database, but also on the spe-

cific class results. MKLERG gave better results than the best one obtained in [5], for all classes. In particular, best results were obtained for ants, bears, mechanical parts, airplanes, armadillo and pliers where the P/R curves were almost ideal (see Fig. 6 for an example). Worst results were obtained for uniformly difficult classes vases, springs and faces, but were better than the state of the art also on these classes. Note that these results also slightly outperform the one obtained in [1], where the authors used the same 3D shape descriptors but aggregated the kernel results using a simple voting scheme.

5.5 Results on SHREC'08

The results in Fig. 7 show the choice of the weights for the optimal kernel on the SHREC'2008 data set. In general, we see that the distance from the barycenter is the most relevant in all the three levels of categorization. Anyway, as far as the classification is finer also the distance from the principal vector (MSA), the average geodesic distance and the mixed Laplacian-based functions come into the play. From our point of view, this fact is explained by the high intra-class variability of the shapes at the coarse level: in this case

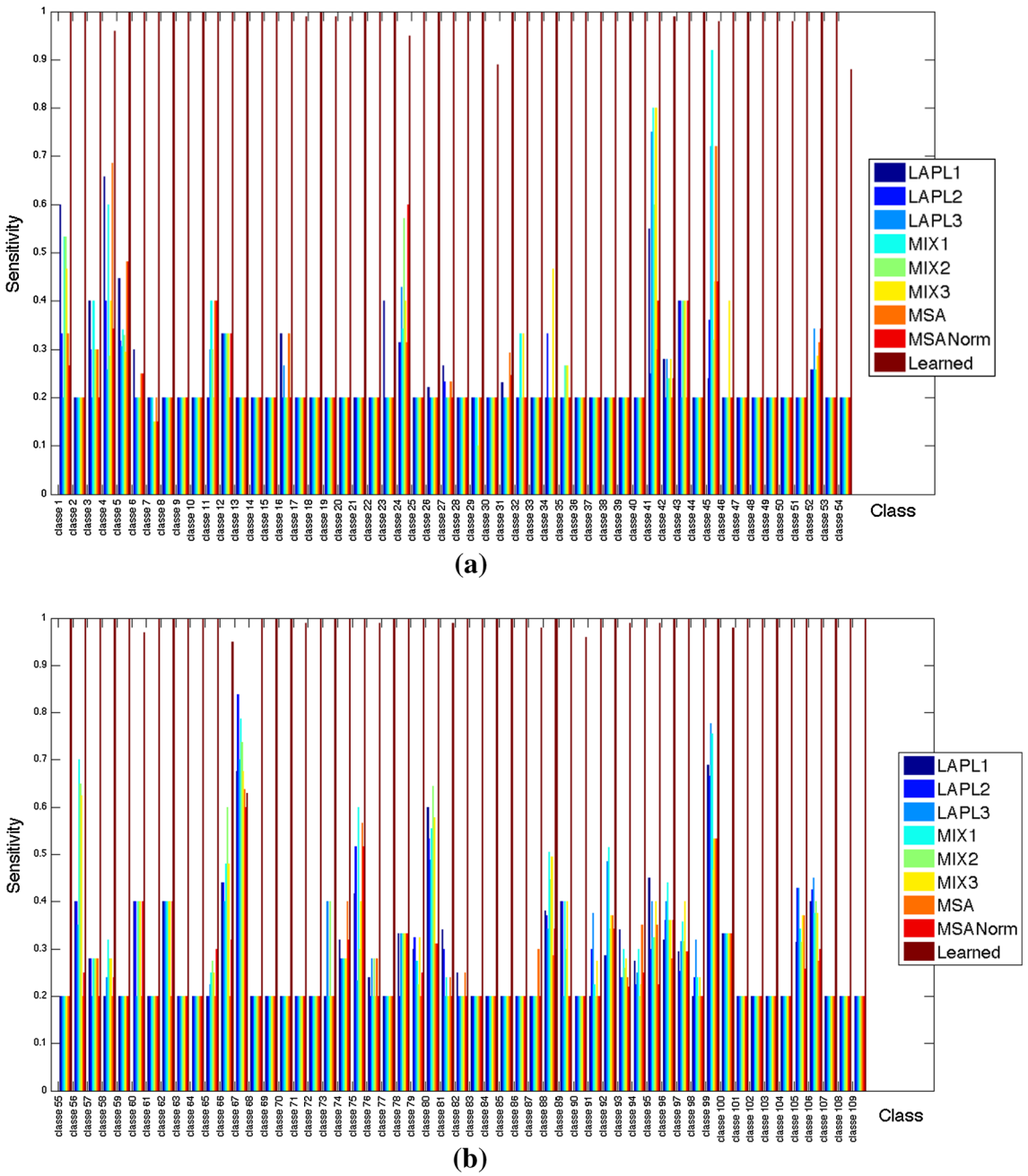

Fig. 9 Specificities of the SHREC'08 data set—third level of categorization for the first 54 (**a**) and the others 55 (**b**) classes

Table 2 RCA and comparison to the state of the art—SHREC'08 database

	RCA(1)	RCA(2)	RCA(3)
Coarse level	0.67 (0.67)	0.78 (0.79)	0.84 (0.83)
Intermediate level	0.63 (0.61)	0.68 (0.66)	0.69 (0.67)
Fine level	0.54 (0.50)	0.57 (0.55)	0.58 (0.57)

large geometric variations are admitted and isometric invariants are not enough to characterize the class. On the contrary at the finest level of detail, a class is often made by a single shape and its non-rigid deformations: in this case, the descriptions based on eigenfunctions or geodesics are invariant to these shape variations and better characterize the intra-class variability.

These results confirm that the kernel selection proposed by our approach considers context and user-defined classification. For instance, let us consider the first class of the data set at the coarse level: it contains human and humanoid models; at the second level, this class splits into two sub-classes: humans and humanoids; at the third level the granularity of the classification increases and models are distinguished between standing-up, walking, running and sitting models. As far as our method is concerned, the kernels that better capture the large variability of the class are those related to the coding of the spatial position: namely the distance from the barycenter and the distance from the principal axis that mainly encodes the shape axial symmetry. As soon as the distinction between humans and humanoids (like the armadillo-model) is introduced, we see that the kernel from the average geodesic function becomes relevant, at least for the humanoid class. Finally, at the third level of classification, when two objects belonging to a class share also the same pose, both the average geodesic function and the symmetry with respect to an "horizontal" plane become important. A similar reasoning holds for the whole database; in particular we highlight class 3 (coarse level) that corresponds to hand-made objects (small statue, chesses, etc.) with a large variability of the number of handles in the objects. In this case, when the classification take into account also the shape structure (level 2) the mixed eigenfunctions play an important discriminative role.

The optimal kernels can also be computed and Fig. 8 presents the kernel obtained from the third level on the training set.

In the experiments on the specificity and sensitivity of the learned kernels with respect to the single ones on the SHREC'2008 data set we noticed a significant improvement in the quantitative results; Fig. 9 presents the specificity curves of the single kernels and the learned one (red bold curve).

Optimal kernels for the three levels of categorization were computed (see e.g. Fig. 8), and used for classification purposes on the corresponding training sets. Classification performances were assessed and compared to the one provided in [6]. Table 2 presents the values of $RCA(k)$ for $k = 1, 2, 3$ computed with our method (bold) on the one hand and by [60] (in parenthesis) on the other hand.

Results slightly improve the state of the art, especially at the fine level, more particularly addressing shape similarity issues.

6 Conclusions

Combining several information in order to enhance the result of an algorithm is a quite common strategy in computer science. Examples include combination of classifiers (boosting methods) or combination of heterogeneous data (information fusion) in several domains (e.g. medical imaging, remote sensing). We adopt here this kind of strategy by computing several functions on model meshes, that provide complementary and redundant information for the retrieval process. We then deduce similarity measures using kernels on the ERGs, and combine them to allow a final decision function.

Such a strategy offers several advantages. It is evolutive (it allows other functions to be included in the processing pipeline), modular (only informative subsets of functions can be used) and adaptive (the aggregation rule can be changed, and even learned using a learning set extracted from the original database). Differently from an aggregation of the kernels based on averaging or voting rules like that in [1], learning through the MKL strategy explicitly yields a selection of the most relevant features without jeopardizing the retrieval and classification performances. These are good with respect to the reference methods, and most of the time outperform the state-of-the-art, with the advantage of being evolutive and modular. In addition, the use of kernel similarity measures based on the representation of graphs in terms of a bag of shortest paths with bounded maximal length allows an efficient computation.

Finally, our experiments highlight that only a fraction of the kernels is necessary to effectively retrieve and classify a shape and this selection is database dependent, varying according the intra- and extra-class variability of the shapes. We plan to further experiment the method on larger and heterogeneous data sets, in order to experimentally validate the assumption that some kernels better capture the "semantic" aspects of the class and others, e.g., geodesics, are better suited to detect intra-class variability aspects.

Acknowledgments This work has been partially developed in the CNR research activity ICT.P10.009 and the EU projects VISIONAIR (EU FP7-INFRASTRUCTURES-262044) and IQmulus (EU FP7-ICT-2011-318787).

References

1. Barra, V., Biasotti, S.: 3D shape retrieval using Kernels on extended reeb graphs. Pattern Recognit. **46**(11), 2985–2999 (2013). doi:10.1016/j.patcog.2013.03.019
2. Laga, H., Mortara, M., Spagnuolo, M.: Geometry and context for semantic correspondences and functionality recognition in man-made 3d shapes. ACM Trans. Graph. **32**(5), 150:1–150:16 (2013)
3. Biasotti, S.: Topological coding of surfaces with boundary using Reeb graphs. Comput. Graph. Geom. **7**(1), 31–45 (2005)
4. Barra, V., Biasotti, S.: Learning kernels on extended reeb graphs for 3d shape classification and retrieval. In: 3DOR, pp. 25–32 (2013)
5. Giorgi, D., Biasotti, S., Paraboschi, L.: Watertight models track. Tech. Report 09, IMATI, Genova (2007)
6. Giorgi, D., Marini, S.: Shape retrieval contest 2008: classification of watertight models. In: Shape Modeling International 2008, pp. 219–220. IEEE (2008)
7. Bustos, B., Keim, D., Saupe, D., Schreck, T., Vranić, D.: Feature-based similarity search in 3D object databases. ACM Comput. Surv. **37**(4), 345–387 (2005)
8. Bimbo, A.D., Pala, P.: Content-based retrieval of 3d models. ACM Trans. Multimed. Comput. Commun. **2**(1), 20–43 (2006)
9. Tangelder, J.W., Veltkamp, R.C.: A survey of content based 3d shape retrieval methods. Multimed. Tools Appl. **39**, 441–471 (2008)
10. Tierny, J., Vandeborre, J.P., Daoudi, M.: Partial 3D shape retrieval by reeb pattern unfolding. Comput. Graph. Forum Eurograph. Assoc. **28**(1), 41–55 (2009)
11. Laga, H.: Semantics-driven approach for automatic selection of best views of 3d shapes. In: 3rd EG Conference on 3D Object Retrieval, pp. 15–22. Eurographics Association, Aire-la-Ville (2010)
12. Mateus, D., Cuzzolin, F., Horaud, R., Boyer, E.: Articulated shape matching using locally linear embedding and orthogonal alignment. In: Work. Non-rigid Registration and Tracking through Learning, pp. 1–8. IEEE Computer Society Press, Rio de Janeiro (2007)
13. Nguyen, G., Worring, M., Smeulders, A.: Interactive search by direct manipulation of dissimilarity space. Multimed. IEEE Trans. **9**(7), 1404–1415 (2007)
14. Giorgi, D., Frosini, P., Spagnuolo, M., Falcidieno, B.: 3d relevance feedback via multilevel relevance judgements. Visual Comput. **26**(10), 1321–1338 (2010)
15. Akgul, C.B., Sankur, B., Yemez, Y., Schmitt, F.: Similarity learning for 3d object retrieval using relevance feedback and risk minimization. Int. J. Comput. Vis. **89**(2—-3), 392–407 (2010). doi:10.1007/s11263-009-0294-1
16. Tieu, K., Viola, P.: Boosting image retrieval. Int. J. Comput. Vis. **56**, 17–36 (2004)
17. Hou, S., Lou, K., Ramani, K.: SVM-based semantic clustering and retrieval of a 3D model database. Computer Aided Des. Appl. **2**(1–4), 155–164 (2005)
18. Laga, H., Nakajima, M.: A boosting approach to content-based 3d model retrieval. In: Andrew, Rohl. (ed.) Proceedings of the 5th International Conference on Computer Graphics And Interactive Techniques in Australia and Southeast Asia. GRAPHITE'07, pp. 227–234. ACM, Perth (2007)
19. Laga, H., Nakajima, M.: Supervised learning of similarity measures for content-based 3d model retrieval. In: Large-Scale Knowledge Resources. Construction and Application pp. 210–225 (2008)
20. Freund, Y., Schapire, R.E.: A decision-theoretic generalization of on-line learning and an application to boosting. In: 2nd European Conference on Computational Learning Theory, pp. 23–37. Springer, London (1995)
21. Ohbuchi, R., Kobayashi, J.: Unsupervised learning from a corpus for shape-based 3d model retrieval. In: Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval, MIR'06, pp. 163–172. ACM, New York, NY (2006)
22. Shilane, P., Funkhouser, T.: Selecting distinctive 3d shape descriptors for similarity retrieval. In: Shape Modeling International 2006, SMI'06, pp. 18. IEEE Computer Society, Washington, DC (2006)
23. Marini, S., Patanè, G., Spagnuolo, M., Falcidieno, B.: Spectral feature selection for shape characterization and classification. Visual Comput. **27**(11), 1005–1019 (2011)
24. Tabia, H., Daoudi, M., Vandeborre, J.P., Colot, O.: A parts-based approach for automatic 3d shape categorization using belief functions. ACM TIST **4**(2), 33 (2013)
25. Litman, R., Bronstein, A.M.: Learning spectral descriptors for deformable shape correspondence. IEEE Trans. Pattern Anal. Mach. Intell. **99**(PrePrints), 1 (2013)
26. Ovsjanikov, M., Ben-Chen, M., Solomon, J., Butscher, A., Guibas, L.: Functional maps: a flexible representation of maps between shapes. ACM Trans. Graph. **31**, 30:1–30:11 (2012)
27. Pokrass, J., Bronstein, A.M., Bronstein, M.M., Sprechmann, P., Sapiro, G.: Sparse modeling of intrinsic correspondences. Comput. Graph. Forum **32**(2pt4), 459–468 (2013)
28. Rustamov, R.M., Ovsjanikov, M., Azencot, O., Ben-Chen, M., Chazal, F., Guibas, L.: Map-based exploration of intrinsic shape differences and variability. ACM Trans. Graph. **32**(4), 72:1–72:12 (2013). doi:10.1145/2461912.2461959
29. Biasotti, S., Spagnuolo, M., Falcidieno, B.: Grouping real functions defined on 3d surfaces. Comput. Graph. **37**(6), 608–619 (2013). Shape modeling international (SMI) conference 2013. doi:10.1016/j.cag.2013.05.007
30. Gönen, M., Alpaydin, E.: Multiple kernel learning algorithms. JMLR **12**, 2211–2268 (2011)
31. Reeb, G.: Sur les points singuliers d'une forme de Pfaff complètement intégrable ou d'une fonction numérique. Comptes Rendus Hebdomadaires des Séances de l'Académie des Sciences **222**, 847–849 (1946)
32. Biasotti, S., Giorgi, D., Spagnuolo, M., Falcidieno, B.: Reeb graphs for shape analysis and applications. Theor. Comput. Sci. **392**(1–3), 5–22 (2008)
33. Funkhouser, T., Min, P., Kazhdan, M., Chen, J., Halderman, A., Dobkin, D., Jacobs, D.: A search engine for 3D models. ACM Trans. Graph. **22**(1), 83–105 (2003)
34. Kashima, H., Tsuda, K., Inokuchi, A.: Marginalized kernels between labeled graphs. In: 20th International Conference on Machine Learning, pp. 321–328. AAAI Press, Washington, DC (2003)
35. Vishwanathan, S.V.N., Schraudolph, N.N., Kondor, R., Borgwardt, K.: Graph kernels. J. Mach. Learn. Res. **11**, 1201–1242 (2010)
36. Philipp-Foliguet, S., Jordan, M., Fuzier, M., Gosselin, P.H.: Indexing of 3d models based on graph of surfacic regions. In: Proceedings of the ACM workshop on 3D object retrieval, pp. 69–74. ACM, New York, NY (2010)
37. Assfalg, J., Borgwardt, K.M., Kriegel, H.P.: 3dstring: a feature string kernel for 3d object classification on voxelized data. In: Yu, Philip S., Tsotras, Vassilis J., Fox, Edward A., Liu, Bing. (eds.) Proceedings of the 15th ACM International Conference on Information and Knowledge Management, pp. 198–207. ACM, Arlington, Virginia (2006)
38. Fisher, M., Savva, M., Hanrahan, P.: Characterizing structural relationships in scenes using graph kernels. ACM Trans. Graph. **30**, 34:1–34:12 (2011)
39. Gärtner, T., Flach, P., Wrobel, S.: On graph kernels: Hardness results and efficient alternatives. In: Proceedings of the 16th Conference on Computational Learning Theory, LNCS, vol. 2777, pp.

129–143. Springer, Berlin, Heidelberg, Germany, Washington, DC (2003)
40. Borgwardt, K.M., Kriegel, H.P.: Shortest-path kernels on graphs. In: Proceedings of the 5th International Conference on Data Mining, pp. 74–81. IEEE Computer Society, Washington, DC (2005)
41. Kondor, R., Jebara, T.: A kernel between sets of vectors. In: Tom, Fawcett., Nina, Mishra. (eds.) International Conference on Machine Learning. AAAI Press, Washington, DC (2003)
42. Suard, F., Rakotomamonjy, A., Benrshrair, A.: Kernel on bag of paths for measuring similarity of shapes. In: 15th European Symposium on Artificial, Neural Networks, pp. 355–360 (2007)
43. Haasdonk, B., Bahlmann, C.: Learning with distance substitution kernels. In: 26th DAGM Symposium on Pattern Recognition, LNCS, vol. 3175, pp. 220–227. Springer, Tübingen, Germany (2004)
44. Biasotti, S., De Floriani, L., Falcidieno, B., Frosini, P., Giorgi, D., Landi, C., Papaleo, L., Spagnuolo, M.: Describing shapes by geometrical-topological properties of real functions. ACM Comput. Surv. **40**(4), 1–87 (2008)
45. Biasotti, S., Cerri, A., Frosini, P., Giorgi, D., Landi, C.: Multidimensional size functions for shape comparison. J. Math. Imaging Vis. **32**, 161–179 (2008)
46. Hilaga, M., Shinagawa, Y., Kohmura, T., Kunii, T.L.: Topology matching for fully automatic similarity estimation of 3D shapes. In: 28th Conference on Computer Graphics and Interactive Techniques, SIGGRAPH'01, pp. 203–212. ACM Press, Los Angeles, CA (2001)
47. Belkin, M., Sun, J., Wang, Y.: Discrete laplace operator for meshed surfaces. In: Symposium on Computational Geometry, pp. 278–287. ACM, University of Maryland, USA (2008)
48. Bronstein, A.M., Bronstein, M.M., Kimmel, R.: Efficient computation of isometry-invariant distances between surfaces. SIAM J. Sci. Comput. **28**(5), 1812–1836 (2006)
49. Akgul, C.B., Sankur, B., Yemez, Y., Schmitt, F.: Similarity learning for 3d object retrieval using relevance feedback and risk minimization. Int. J. Comput. Vis. Special issue on 3D object retrieval **89**, 392–407 (2010)
50. Rakotomamonjy, A., Bach, F., Canu, S., Grandvalet, Y.: Simplemkl. J. Mach. Learn. Res. **9**, 2491–2521 (2008)
51. Lanckriet, G.R.G., Cristianini, N., Bartlett, P., El Ghaoui, L., Jordan, M.I.: Learning the kernel matrix with semidefinite programming. J. Mach. Learn. Res. **5**, 27–72 (2004)
52. Sonnenburg, S., Rätsch, G., Schäfer, C., Schölkopf, B.: Large scale multiple kernel learning. J. Mach. Learn. Res. **7**, 1531–1565 (2006)
53. Baeza-Yates, R.A., Ribeiro-Neto, B.: Modern Information Retrieval. Addison-Wesley Longman Publishing Co., Inc., Boston, MA (1999)
54. Typke, R., Veltkamp, R.C., Wiering, F.: A measure for evaluating retrieval techniques based on partially ordered ground truth lists. In: Proceedings of the 2006 IEEE International Conference on Multimedia and Expo. ICME 2006, July 9–12, pp. 1793–1796. IEEE, Toronto, ON (2006)
55. Akgul, C.B., Sankur, B., Yemez, Y., Schmitt, F.: Model retrieval using probability density-based shape descriptors. IEEE Trans. PAMI **31**(6), 1117–1133 (2009)
56. Chaouch, M., Verroust-Blondet, A.: 2d/3d descriptor based on depth line encoding. In: Proceedings of the SHREC'07—Shape Retrieval Contest 2007: Watertight Models Track and CAD Models Track, Presented at a Special Session of SMI'07, Shape Modeling International 2007, pp. 31–32. IEEE Computer Society, Washington, DC (2007)
57. Napoleon, T., Adamek, T., Schmitt, F.: O'Connor., N.: Multiview 3d retrieval using silhouette intersection and multi-scale contour representation. In: Proceedings of the SHREC'07—Shape Retrieval Contest 2007: Watertight Models Track and CAD Models Track, Presented at a Special Session of SMI'07, Shape Modeling International 2007, pp. 33–37. IEEE Computer Society, Washington, DC (2007)
58. Daras, P., Tzovaras, D., Axenopoulos, A., Zarpalas, D., Mademlis, A., Strintzis, M.G.: The spherical trace transform. In: Proceedings of the SHREC'07—Shape Retrieval Contest 2007: Watertight Models Track and CAD Models Track, Presented At A Special Session of SMI'07, Shape Modeling International 2007, p. 38. IEEE Computer Society, Washington, DC (2007)
59. Tung, T., Schmitt, F.: Shrec'08 entry: shape retrieval of noisy watertight models using amrg. In: Proceedings of the IEEE International Conference on Shape Modeling and Applications 2007, pp. 229–230. IEEE Computer Society, Washington, DC (2007)
60. Goldfeder, C., F. H., Allen, P.: Shrec'08 entry: training set expansion via autotags. In: Shape Modeling, International (2008)

Vincent Barra received the Ph.D. in image processing from the University of Clermont-Ferrand, France, in 2001. He is currently a full professor in Computer Science at the University Blaise Pascal of Clermont-Ferrand, and head of an engineering school (ISIMA). His research interests include image processing, statistical learning and 3D shape analysis. He has published more than 30 papers in the field of image analysis.

Silvia Biasotti is a researcher at IMATI-CNR. She graduated in Mathematics (1998), received a Ph.D. in Mathematics and Applications (2004) and a Ph.D. in ICT (2008), all from the Univ. of Genoa. Her research interests include the study of topological-geometrical descriptions of 2D and 3D models and shape similarity based on structural descriptions . She also is involved in the creation of benchmarks for 3D object retrieval and geospatial data analysis. In her career she authored more than 80 scientific peer-reviewed contributions and delivered specialistic courses at several international venues, among them SIGGRAPH2012, SERA2012 and SGP2013. She was the programme chair of the 3DOR'13 and is a member of programme committee of SMI'14, 3DIPM'14, 3DOR'14, PATTERNS2014, ICIAP 2015.