Eurographics Workshop on 3D Object Retrieval (2014)
B. Bustos, H. Tabia, J.-P. Vandeborre, and R. Veltkamp (Editors)

TreeSha: 3D shape retrieval with a tree graph representation based on the autodiffusion function topology

Valeria Garro and Andrea Giachetti

Department of Computer Science, University of Verona, Italy

Abstract

In this paper we present a new method for shape description and matching based on a tree representation built upon the scale space analysis of maxima of the Autodiffusion function (ADF). The use of the Heat Kernel based approach makes the method invariant to articulated deformations. By coupling maxima of the Autodiffusion function with the related basins of attraction, it is possible to link the information at different scales encoding spatial relationships in a tree structure. Furthermore, texture information can be easily included in the descriptor by adding regional color histograms to the node attributes of the tree. Dedicated graph kernels have been designed to evaluate shape dissimilarity from the obtained representations using both structural, geometric and color information. Preliminary experiments performed on the SHREC 2013 non-rigid textured dataset showed very good retrieval performances.

Categories and Subject Descriptors (according to ACM CCS): H.3.1 [Computer Graphics]: Information Storage and Retrieval—Content Analysis and Indexing

1. Introduction

3D shape retrieval is a hot topic in Computer Graphics and Computer Vision due to the fast growing availability of acquired 3D data and the large number of related applications, for example, automatic indexing, object recognition and medical diagnosis. Several approaches have been proposed for this task, and many of these methods demonstrate good performances in dedicated benchmarks, but there are still open issues related to the development of methods that are effective in different specific contexts. For example, several algorithms demonstrated good performances for the retrieval of shapes modified by articulated deformations [LGB*11], but they may be not similarly effective when the deformations relating elements of the same class are not isometric (e.g. partial dilation and stretching, body type variations). Many of the existing methods also do not handle directly the use of texture information, and specific benchmark for the comparisons of textured shape retrieval have been proposed only recently [CBA*13].

In this paper we present a novel framework for 3D object retrieval that tries to combine the two main approaches used to compare non-rigid shapes, i.e. spectral based and graph based, allowing an easy integration of color information in the surface descriptor. Our approach is based on the Laplace-Beltrami decomposition, but instead of comparing directly the spectral components it describes shapes through the topology of the scale space of the Autodiffusion function [SOG09, GBAL09]. Analyzing the evolution of salient points (maxima of ADF) across scales together with the associated basins of attraction, we can generate a tree structure (TreeSha) that encodes the relationships between structure with different level of detail in different part of the object. Texture information is naturally handled associating a texture descriptor for each basin of attraction. Preliminary results demonstrate that the proposed approach is reasonable and it provides retrieval scores similar to state-of-the-art methods on existing non-rigid shape retrieval benchmark.

2. Related Work

There is a huge amount of literature related to shape retrieval. To have a general overview of the problem and the related approaches, we can refer to surveys like [TV08] or to the proceedings of dedicated workshops (e.g. SHREC). Due to limited space it is not possible to comprehensively discuss the huge amount of methods recently proposed, instead we try to summarize some of the most relevant contributions re-

DOI: 10.2312/3dor.20141045

lated to our technique and to the specific task of deformable and textured shape retrieval.
To obtain shape retrieval methods that are robust against articulated deformations two main classes of algorithms have been proposed, one based on spectral descriptors and the other one on graph representations.

Methods based on spectral decomposition are usually based on the analysis of the eigenvalues of the Laplace-Beltrami spectrum. ShapeDNA [RWP06] is probably the best known global descriptor based on this approach, using simply the normalized truncated sequence of eigenvalues. The descriptor is invariant against isometries, but inheriting the properties by the spectrum itself it may be not necessarily robust against body type variations. A method based on spectral decomposition of geodesic distance matrix is described in [SHVS12], this approach, coupled with a local descriptor, performed very well in the SHREC comparison [LGB*12]. Point descriptors based on Laplace-Beltrami have been used to characterize globally 3D shapes using the "bag of words" paradigm [Lav12]. With this approach it is possible to handle partial matching, but information on spatial point distribution is lost. A framework to include texture information in a global spectral description is presented in [KRB*13].

A relevant number of non-rigid shape retrieval methods is actually based on medial representations or graph representations. Sundar et al [SSGD03] demonstrated the possibility of matching shapes using the curve skeleton, and a practical approach for 3D shape matching has been presented in [LH13]. In [APP*09] attributed relational graphs are based on shape segmentation and matched using the earth mover's distance. In [GL12] shape retrieval is performed using histograms of a volumetric medial representation. The method has been extended also to include color information [CBA*13].
Reeb Graphs have often been used as descriptor for shape retrieval [HSKK01]. In [TS05], multiresolution Reeb graphs based on average geodesic distances have been enhanced also with texture information (color histograms). A problem with this approach is to include geometric information in the topological setting and to link information at different scales. In [BB13] extended Reeb Graphs are matched using optimized kernels. In [LMS13] a 3D shape is represented as a graph interconnecting parts that share some spatial relationships, allowing semantic correspondence and classification of objects' parts.

An effort towards the optimization of textured shape retrieval has been done with the organization of a dedicated SHREC 2013 track. Retrieval methods performing well on the proposed dataset are described in [CBA*13] and in some case they characterize separately color and shape. On the same dataset, very good results have been obtained with an approach using a relevant number of scalar functions defined on the surface (including color information), encoded in a persistence homology setting [BCGS13].

In this paper we propose a shape description and matching technique that is based on the combination of a graph based and a spectral based approach. The method builds graphs by analyzing the scale space of the autodiffusion function. The use of topological information related to the autodiffusion at different scales should increase the robustness against "body type" variations, modifying selected spectral components. Coupling salient points with their basins of attraction and linking them across scales, we are also able to include naturally geometrical information (spatial relationships between subparts) and texture information in the tree structure. The graph-based approach, that relies on salient points, could be applied to partial matching as well.

3. Method

The main idea of the proposed method is to define a shape representation based on a graph that encodes the spatial relationships between local maxima of the ADF extracted at different time scales in its topology and that associates to the nodes local chromatic and geometric features. In this section we explain in details the algorithm steps and we briefly describe the theoretical foundations behind them.

3.1. Salient Point Extraction

Modeling a shape as a compact Riemannian manifold M, the heat diffusion equation is defined as

$$(\Delta_M + \frac{\partial}{\partial t})u(x,t) = 0, \tag{1}$$

and it describes the heat propagation over the manifold M. Here, Δ_M is the Laplace-Beltrami operator of M, a generalization of the Laplace operator applied on Riemannian manifolds, and $u(x,t)$ denotes the heat distribution at point x in M at time t.
The fundamental solution $K(x,y,t)$ to equation 1 with point heat distribution $u_0 = \delta(x-y)$ is called heat kernel. The heat kernel function $K(x,y,t) : M \times M \times \mathbb{R}_0^+ \rightarrow \mathbb{R}$ expresses the amount of heat transferred from the point y to the point x of the manifold M after time t.
Using the spectral decomposition theorem the heat kernel can be defined as

$$K(x,y,t) = \sum_{l=0}^{\infty} e^{-\lambda_l t}\phi_l(x)\phi_l(y), \tag{2}$$

where λ_l and ϕ_l are respectively the l^{th} eigenvalue and eigenfunction of Δ_M, so the heat kernel function can be computed using the Laplace-Beltrami operator of the corresponding mesh. In practice, dealing with surfaces with triangular faces, we applied a well established method to estimate the discretization of the Laplace-Beltrami operator Δ_M called cotangent scheme [PP93].

As formalized in detail in [SOG09], the heat kernel has several relevant properties for the shape analysis. It is invariant under isometry transformations, it is stable under local perturbations and it has a multi scale behavior: for small values of t $K(x,x,t)$ characterizes the shape properties of a small region centered in x, the size of the x neighborhood increases proportionally to the t value meaning that for larger values of t the function describes more global information of M still from the point of view of x.
The kernel function $K(x,x,t)$ computed on the same point x at time t is also known as the Autodiffusion function $ADF(x,t)$. Its behavior at different t in a normalized scale is typically used to characterize local shape features that can be used for point recognition and matching or also to create global descriptors [SOG09, GBAL09, Lav12].

3.2. Tree Representation

The evolution of the salient points of the Autodiffusion function at different time values can be analyzed using the classical scale space theory [Koe84, Lin94].
Our basic assumption is that increasing the value of time t some local maxima will disappear because they are created by local high-frequency details (according to Morse theory [Mil63], there is a "catastrophe" and maxima and saddles are annihilated). Other one are preserved (even if they can move along the shape) and this means that we can consider them more "topologically persistent" [ELZ02]. The evolution of the points gives us information about the shape, but also the spatial relationships between the points can be important for shape retrieval. Our idea is therefore to encode these relationships by connecting disappearing points with surviving ones, thus creating a graph structure. This structure can be built by considering that a critical point that is disappearing for increasing t is spatially located, immediately after the catastrophe, inside the basin of attraction of another more persistent critical point. We build the graph structure by simply linking the disappearing point with the spatially corresponding persistent one.
Of course, we can not evaluate the salient points evolution for continuously varying t. We create therefore our graph structure by joining salient points computed at a few discrete time samples using, however, the same rule.
In our implementation, we compute the $ADF(x,t)$ for each vertex x of the input triangulated mesh at ten different values of t uniformly distributed in the logarithmic scale over the time interval $[t_{\min}, t_{\max}]$ where $t_{\min} = 0.4ln(10)/\lambda_{300}$ and $t_{\max} = 4ln(10)/\lambda_2$. For $t > t_{\max}$ there is no appreciable variability in the function values, as they are mainly influenced by the smallest eigenvalue not equal to zero λ_2 and its corresponding eigenvector ϕ_2. For each $t_j \in \{t_1 = t_{\min}, t_2, \ldots, t_{10} = t_{\max}\}$,we extract the local maxima $\{m_k\}_{t_j}$ comparing the ADF of each vertex to the corresponding values in its 1-ring neighborhood.
We exploit homological persistence [ELZ02] to detect and delete those local maxima that are due to noise fluctuations and to compute the basins of attraction of the remaining reliable local maxima $\{s_k\}_{t_j} \subset \{m_k\}_{t_j}$. We use the technique presented in [SOCG10] that is claimed to be stable under near-isometric deformations and provides a tunable parameter to define the desired noise threshold. These basins of attraction create a disjoint partitioning of the mesh associating a surface patch to each salient point and that can be used, for example, to give a characterization of the points related to texture.
Our graph representation is then easily created as follows:

- our nodes are the salient points $\{s_k\}_{t_j}$ for $j = \{1, \ldots, 10\}$.
- Two nodes are connected with an edge only if:
  - their corresponding salient points are extracted at two consecutive time values t_{j-1} and t_j for $j \in \{2, \ldots, 10\}$.
  - the salient point extracted at the smaller time value, t_{j-1}, is located inside the basin of attraction of the one at t_j.

Nodes and edges define a structure characterizing the shape an that can be easily enriched with several local attributes.
Before describing in more detail the further processing steps, in order to clarify the presentation, we briefly summarize some essential notions of graph theory.
A *graph* $G = (V,E)$ is composed by a set V of elements called vertices or nodes and a set E of edges connecting some pairs of nodes. In case the nodes or the edges (or both) are associated to a weight or a label G is called *attributed graph*.
A *walk* w of length l is a sequence of nodes $v_1, v_2, \ldots, v_{l+1}$ where $\exists e = (v_i, v_{i+1}) \in E$ for $1 \leq i \leq l$. The length l is equal to the number of edges in the walk.
A *path* is a walk with no cycles, $v_i \neq v_j$ with $i \neq j \forall i, j \in \{1, \ldots, l+1\}$.
If there exists a walk from any node to any other node in the graph it is said to be *connected*. If a connected graph has no cycles is called *tree*.
The graph created as described from the multiscale salient points extraction is then preprocessed in order to simplify the structure and better encode the relevant information. First of all, some nodes are removed with a pruning strategy. We recursively search for what we call the “one-child configuration” (namely when a node both has only one child and one father and it is the only child of its father) and we delete that node from the graph. In case of one-child configuration (illustrated in Figure 1 by nodes 4, 7 and 9) the involved nodes represent the same local maximum that is extracted at consecutive time values and it is not merged with any other maxima.
After that, we define a weight for each edge, more precisely let e_{ab} be the edge that connects node n_a extracted at time t_i and node n_b extracted at time t_j, the weight $w(\cdot)$ of edge e_{ab} is defined as the difference between the corresponding

time values extraction of n_a and n_b : $w(e_{ab}) = |i - j|$. In case $w(e_{ab}) = 1$, there has not been a pruning between nodes n_a and n_b, meaning that node n_b has disappeared at time t_i and its basin of attraction has been merged with the basin of n_a. On the other hand, if $w(e_{ab}) > 1$ n_a and n_b represent the same local maximum and $w(e_{ab})$ indicates a measure of persistence of the implicated local maximum trough the scale space.

Finally, the salient points extracted at the highest time value t_{10} are connected to a new artificial node that represents the entire mesh, in this way each mesh is actually represented with a tree.

The steps of the graph creation are illustrated in a toy example in Figure 1 using a one dimensional function exemplifying the autodiffusion function and only tree time scale values.

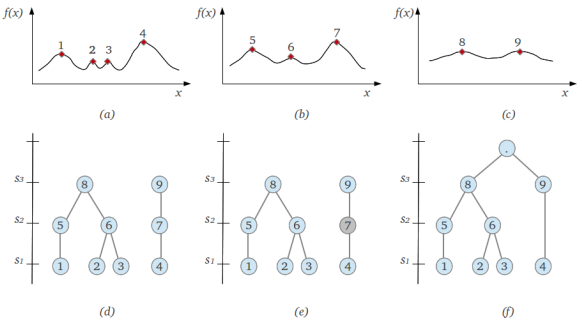

Figure 1: *Top: sketch of the scale-space representation of a one dimensional function $f(x)$, (a) f(x) computed at the initial scale s_1, (b) f(x) computed at an intermediate scale s_2 and (c) f(x) computed at the final scale s_3, with $s_1 < s_2 < s_3$. The local maxima are enumerated and indicated with diamond markers. Bottom: The three steps of the graph definition: (d) creation of the nodes and the edges, the number inside each node indicates its corresponding local maximum; (e) node pruning, node number 7 has a one-child configuration and it will be deleted; (f) final insertion of the artificial root node.*

3.3. Graph Matching

A mesh is thus modeled as an attributed tree and we have to define a metric to compare their graph representations.

There exist several techniques to measure the similarity between two graphs, they can be divided into three main categories: methods based on graph spectra, methods based on graph edit distance and methods based on graph kernels. For a clear general introduction on this topic the reader can refer to [LCT*12].

Kernel based methods are well established classification procedures in the machine learning community. Recently they have become popular also in shape analysis, both 2D [SRB07, DB08] and 3D [PFJFG10, BB13] and other branches of the computer vision. Generally speaking, graph kernels estimate the similarity of a pairs of graphs by comparing some graph sub-structures using local kernel functions. A local kernel function expresses the similarity of two sub-structures by comparing each pairs of elements. These sub-structures can have different nature, for example they can be walks [Gär02], shortest paths [BK05] or cyclic patterns [HGW04].

We adopt a technique similar to [BK05] in which a graph is efficiently represented by its complete set of shortest paths which can be computed in polynomial time using the well known Dijkstra [Dij59] or Floyd-Warshall [Flo62] algorithms. As we are dealing with trees, we can exploit the particular configuration of these graphs in order to improve the computational efficiency. We decided to compare only a subset of the shortest paths, namely those starting from the node root and ending to each node leaf of the trees. In the following, this set will be called P_T.

Graph kernel definition. Let T_1 and T_2 be two TreeSha graphs. We define our *root-leaves shortest path graph kernel* as:

$$K(T_1, T_2) = \sum_{p_i \in P_{T_1}} \sum_{p_j \in P_{T_2}} k_{\text{path}}(p_i, p_j) \tag{3}$$

where p_i and p_j are the shortest paths from the root to the leaves of tree T_1 and T_2 respectively, and the path kernel function $k_{\text{path}} : P_{T_1} \times P_{T_2} \rightarrow \mathbb{R}_0^+$ is a symmetric and positive definite function.

Path Kernels. Measuring the similarity between two paths consists in a sequence of comparisons of corresponding basic elements of the paths, i.e., nodes and edges. For two paths $p_i = (v_1, v_2, \ldots, v_{l+1})$ and $p_j = (u_1, u_2, \ldots, u_{l+1})$ of equal length l, let e_k and f_k be the edges connecting respectively nodes (v_k, v_{k+1}) and (u_k, u_{k+1}), the kernel function comparing the paths is defined as:

$$k_{\text{path}}(p_i, p_j) = \left(\left(\prod_{k=1}^{l-1} k_{\text{nd}}(v_k, u_k) k_{\text{ed}}(e_k, f_k) \right) k_{\text{nd}}(v_l, u_l) \right)^{\frac{1}{l}} \tag{4}$$

where k_{nd} is the kernel that measures the similarity between two nodes and k_{ed} is the one applied to edges. In order to have equal contributions when comparing paths of different lengths, the resulted product is raised to $1/l$.

Instead, for a pair of paths p_i and p_j with different length the similarity is set to zero:

$$k_{\text{path}}(p_i, p_j) = 0 \qquad \text{for} \quad l_i \neq l_j \tag{5}$$

Below, we explain in details the complete set of kernel functions applied to the nodes and the edges of the paths.

Kernel functions on nodes. In our representation, the nodes of the tree correspond to the salient points extracted from the mesh. To compare nodes, we use different functions that characterize the geometry and the texture of these salient points and their related basins of attraction. Let (v_k, u_k) be our pair of nodes to be compared, $k_{\text{nd}}(v_k, u_k)$ will be the product of four different kernels measuring various attributes

related to the salient points:

$$k_{\text{nd}} = k_{\text{deg}} \cdot k_{\text{area}} \cdot k_{\text{hLab}} \cdot k_{\text{mLab}}. \quad (6)$$

The first is the Degree kernel (D), that analyzes the local structure of the graphs comparing the number of edges incident to the node, i.e. the node's degree $\deg(\cdot)$:

$$k_{\text{deg}}(v_k, u_k) = e^{\frac{(\deg(v_k)-\deg(u_k))^2}{2\sigma_d}} \quad (7)$$

By construction each node has only one node-father, therefore the $\deg(\cdot)$ feature indicates the number of node-children, corresponding to the number of local maxima extracted at a lower t value that are located inside the node's basin of attraction.
The second kernel is the Area kernel (A), that compares the geometry of the shapes by measuring the difference between the areas of the two basins of attraction associated to the two nodes, indicated by $a(\cdot)$:

$$k_{\text{area}}(v_k, u_k) = e^{\frac{(a(v_k)-a(u_k))^2}{2\sigma_a}} \quad (8)$$

The area values are normalized by the total area of the corresponding shape.
The last two kernels exploit the photometric information given by the texture of the shape. For a colored mesh, each vertex is associated to a color value, usually RGB, we convert this data to the CIELab color representation [WS00]. The advantage of using this color space is that the euclidean distance between two points in CIELab is linearly proportional to the difference of the corresponding colors perceived in human vision. For each basin of attraction we compute two color features: a normalized color histogram, that characterizes the color distribution of that particular shape's region, and the mean color value $m(v_k)$. In the CIELab representation, formed by three components (L^*, a^*, b^*), the lightness described by L^* is separated to the cromaticity values (a^*, b^*). In our test, the color quantization is set to 4 bins for the L^* component and 8 bins for both a^* and b^*. As a result we have a histogram $H(v_k)$ of size $4 \times 8 \times 8$ for each basin of attraction. The distance between two histograms is measured using the earth mover's distance [RTG98], the comparison between two mean color values is done trough euclidean distance.
The first photometric kernel is therefore the Histogram kernel (H), defined as follows:

$$k_{\text{hLab}}(v_k, u_k) = e^{\frac{d_{EMD}(H(v_k)-H(u_k))^2}{2\sigma_h}} \quad (9)$$

The second color-based kernel, namely the Mean color kernel (M), measures only distances of average colors in the corresponding regions:

$$k_{\text{mLab}}(v_k, u_k) = e^{\frac{\|(m(v_k)-m(u_k)\|^2}{2\sigma_m}} \quad (10)$$

Edge Kernel function. We formulate also a further kernel function that involves the weights of the edges of the graph. As explained in Section 3.2, the weight $w(\cdot)$ of an edge represents the persistence value of the related local maximum

Figure 2: *Examples of shapes from the SHREC'13 dataset. Isometric and non isometric transforms are applied with highly different strengths to base models*

according to the scale space t. This kernel, defined as Edge kernel (E), compares the weights of two edges simply their Euclidean distance.

$$k_{\text{ed}}(e_k, f_k) = e^{\frac{\|(w(e_k)-w(f_k)\|^2}{2\sigma_e}} \quad (11)$$

All the kernel functions described above are gaussian kernels, measuring the similarity in the $(0,1)$ interval and proved to be positive definite [SS01].

4. Experimental results

We tested our shape description and comparison techniques on the dataset created for the SHREC'13 track retrieval on Textured 3D Models. The dataset is composed by 240 watertight textured meshes, with 10 shape classes created from two base models modified with four spatial transforms (noise addition, articulated deformation and two kind of non metric deformations). Each shape class is represented with different textures so that 33 classes differing for shape and texture can be identified, 6 classes formed by 4 models and the remaining one formed by 8 models. Similar textures are applied to different shapes in order to avoid an easy retrieval based only on color information.
On these models we computed the TreeSha representation based on the salient points extracted at the selected scales. Figure 3 shows salient points with related basins of attraction computed on two example models and the corresponding tree structures. The structural representation should be rather robust against noise and deformation, even if the segmentation of the basins of attraction is rough. However, the use of a very small number of sample values t creates some problem in the tree structure, as, for example, a few paths are unexpectedly ended at $t > 1$. Using a finer time sampling probably these problems would be removed, at the cost of an increased computational complexity. The results of the shape retrieval tests, however, demonstrate that, despite the

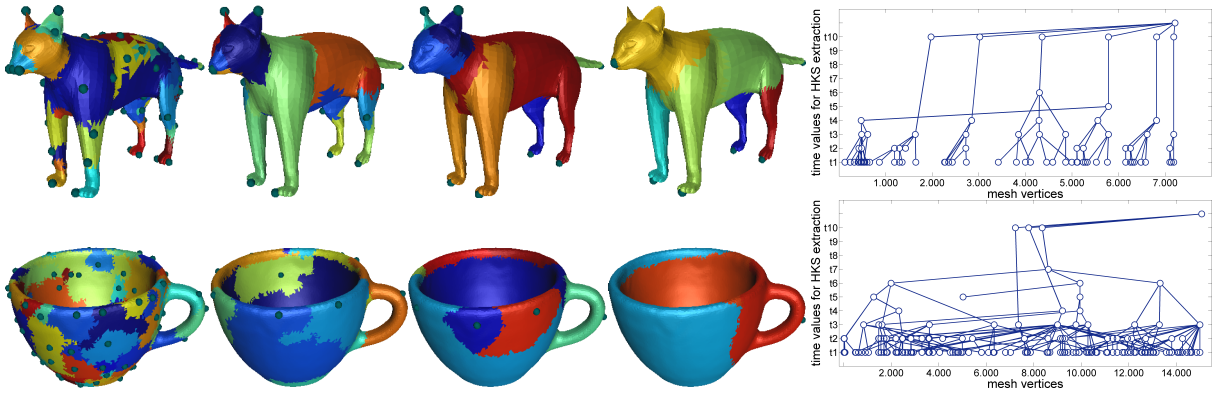

Figure 3: *Left: salient points and related basins of attractions at four selected time values samples (t_1,t_3,t_5,t_7) on two example models of the SHREC'13 dataset. Right: the corresponding TreeSha graphs.*

rough discretization, the TreeSha structure encodes well the relevant information.

We compared the results of our retrieval tests with those of the best performing methods of the SHREC13 contest and with the results presented in [BCGS13]. The evaluation measures compared are the classical Nearest Neighbor (NN), First Tier (FT), Second Tier (ST) related to the ratio of models in the query class that also appear within the top matches (see [SMKF04]), considering only "highly relevant" items, e.g. objects with same shape label and same textures; and a measure evaluating also the retrieval of correct shapes with wrong texture ("marginally relevant items") even if with lower weight: the Average Dynamic Recall (ADR, see [CBA*13]).

Table 1 shows the results obtained with our descriptor and a graph matching procedure including all the graph kernels introduced in Section 3. Our method provides the best results in all the scores, showing a good ability to discriminate both shape and texture.

In this experiment the values of the sigma parameters for each kernel have been set to values that we considered reasonable in order to obtain a good trade-off between strict selection of similar feature and noise tolerance. The values chosen were: $\sigma_d = 3$, $\sigma_a = 256$, $\sigma_h = 256$, $\sigma_m = 64$ and $\sigma_e = 3$. Different choices of the parameters (as well as of the kernel functions) can clearly change the results, and the selection of kernel types and parameters for our attributed graphs is surely an interesting topic for further research.

An interesting aspect of the proposed method is that we actually used a limited number of features (and related graph kernels) to describe the textured shapes: Histogram (H), Area (A), Mean color (M), Degree (D), Edge (E).

The Histogram kernel (H) is, as expected, the one providing the best retrieval scores if used independently, encoding both structural and color information. The use of the other kernels, however, improves the performances and reaches the scores reported in Table 1. Figure 4 shows the different

Run	NN	FT	ST	ADR
A2	0.508	0.561	0.730	0.380
Gi	0.788	0.658	0.748	0.470
G2	0.898	0.733	0.893	0.508
V2	0.879	0.764	0.904	0.520
PHOG	0.951	0.773	0.899	0.534
TreeSha	**0.958**	**0.791**	**0.906**	**0.601**

Table 1: *Retrieval performances obtained with the TreeSha representation and the complete graph kernel described in Section 2, compared with the scores of the best performing methods in the SHREC '13 contest (A2,Gi,G2,V2) and with the PHOG method.*

scores obtained by combining differently the tested kernels. It is clear that different functions could have been tested and that specific kernels and kernel combinations could be developed and customized for specific applications. For example, the spectral signatures on critical points (or averaged over basins of attraction), the geodesic distances on paths, the scalar values on geodesic paths could be used as well, and we plan to test their use as future work.

The selection of the optimal kernel for the specific tasks

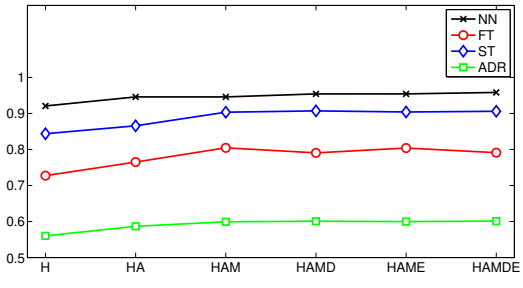

Figure 4: *Retrieval scores (Nearest Neighbor, First Tier, Second Tier, Average Dynamic Recall) obtained with different combinations of the kernels: Histogram (H), Area (A), Mean (M), Degree (D), Edge(E).*

depends on the kind of shape and texture discriminations required by the application. To understand, for example, where our kernels perform well, we can have a look on the results obtained on different shape classes. Figure 5 shows the average precision-recall plot and a few selected precision-recall plots for specific shape/texture categories. It is possible to see that the method is less effective for shapes with flat parts where the regions extracted with the basins of attraction are less stable. This behavior is visible also in the tier matrix represented in Figure 6, where Nearest Neighbors (black), First Tier results (red) and Second Tier results (blue) are plotted against query objects.

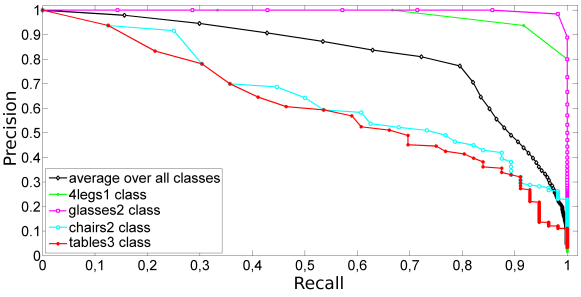

Figure 5: *Precision vs. recall plots for four example single classes and averaged on all classes.*

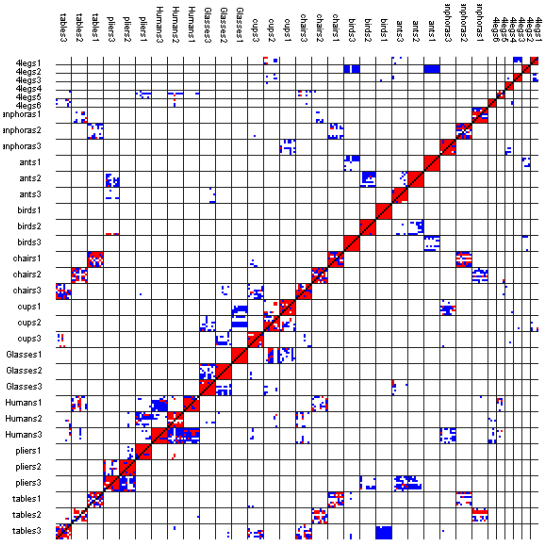

Figure 6: *Tier image representing retrieved objects for each query object. Nearest Neighbor (NN) are represented in black, First Tier in red, second Tier in blue. Tables and chairs are the maximally critical shape classes for our method.*

The proposed method has been implemented in MATLAB, the experiments have been run on a Intel QuadCore with 2, 8Ghz. During the creation of the TreeSha graph for a single mesh the most time consuming steps of the algorithm are the salient points extraction and the computation of the basins of attraction that together take about 300 sec. for a mesh with 15000 vertices. On average, the obtained TreeSha graphs have 160 nodes of which 100 are leaves nodes, the mean time for the computation of the Graph Kernel for a pair of shapes is 0.48 sec. , for the whole dataset it took about 235 min.

5. Conclusions

We presented a novel method to characterize 3D textured shapes based on a combination of spectral and graph-based techniques. The advantage of our method with respect to other techniques using multiscale graph representation is that our graph links spatially the structures extracted at different levels of detail in a natural way. Being based on the Laplace-Beltrami spectrum, our method inherits its good features (isometric invariance) and limitations (sensitivity to topological noise).
The creation of the TreeSha tree structure is clearly subject to noise and problems related to the effect of non metric transforms, and to the quite coarse discretization of the time interval done in our test. However, being the information encoded quite rich and using optimally selected kernel functions to match the graphs, the results obtained in a challenging shape retrieval test are quite good. These results could be probably improved by using a finer sampling of the scale space and by adding other attributes to both nodes and edges of the TreeSha structure together with the corresponding graph kernels for comparison. We plan to test these improvements as future work and to apply the same technique to different problems for which we expect it should be suitable, such as partial matching (in this case, however, the method should be changed to preserve scale invariance [BK10]).

Acknowledgment Thanks to Davide Boscaini and Umberto Castellani for useful discussions.

References

[APP*09] AGATHOS A., PRATIKAKIS I., PAPADAKIS P., PERANTONIS S., AZARIADIS P., SAPIDIS N.: Retrieval of 3d articulated objects using a graph-based representation. In *Proceedings of the 2nd Eurographics conference on 3D Object Retrieval* (2009), Eurographics Association, pp. 29–36. 2

[BB13] BARRA V., BIASOTTI S.: 3d shape retrieval using kernels on extended reeb graphs. *Pattern Recognition 46*, 11 (2013), 2985–2999. 2, 4

[BCGS13] BIASOTTI S., CERRI A., GIORGI D., SPAGNUOLO M.: Phog: Photometric and geometric functions for textured shape retrieval. In *Computer Graphics Forum* (2013), vol. 32, Wiley Online Library, pp. 13–22. 2, 6

[BK05] BORGWARDT K., KRIEGEL H.-P.: Shortest-path kernels on graphs. In *Data Mining, Fifth IEEE International Conference on* (2005), pp. 8 pp.–. 4

[BK10] BRONSTEIN M. M., KOKKINOS I.: Scale-invariant heat kernel signatures for non-rigid shape recognition. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on* (2010), IEEE, pp. 1704–1711. 7

[CBA*13] CERRI A., BIASOTTI S., ABDELRAHMAN M., ANGULO J., BERGER K., CHEVALLIER L., EL-MELEGY M., FARAG A., LEFEBVRE F., GIACHETTI A., ET AL.: Shrec'13 track: Retrieval on textured 3d models. In *Eurographics 2013 Workshop on 3D Object Retrieval* (2013), The Eurographics Association, pp. 73–80. 1, 2, 6

[DB08] DUPÉ F.-X., BRUN L.: Hierarchical bag of paths for kernel based shape classification. *CoRR abs/0810.3579* (2008). 4

[Dij59] DIJKSTRA E. W.: A note on two problems in connexion with graphs. *Numerische Mathematik 1* (1959), 269–271. 4

[ELZ02] EDELSBRUNNER H., LETSCHER D., ZOMORODIAN A.: Topological persistence and simplification. *Discrete & Computational Geometry 28*, 4 (2002), 511–533. 3

[Flo62] FLOYD R. W.: Algorithm 97: Shortest path. *Commun. ACM 5*, 6 (June 1962), 345–. 4

[Gär02] GÄRTNER T.: Exponential and geometric kernels for graphs. In *NIPS Workshop on Unreal Data: Principles of Modeling Nonvectorial Data* (2002). 4

[GBAL09] GĘBAL K., BÆRENTZEN J. A., AANÆS H., LARSEN R.: Shape analysis using the auto diffusion function. In *Proceedings of the Symposium on Geometry Processing* (Aire-la-Ville, Switzerland, Switzerland, 2009), SGP '09, Eurographics Association, pp. 1405–1413. 1, 3

[GL12] GIACHETTI A., LOVATO C.: Radial symmetry detection and shape characterization with the multiscale area projection transform. *Comp.Graph.Forum 31*, 5 (2012), 1669–1678. 2

[HGW04] HORVÁTH T., GÄRTNER T., WROBEL S.: Cyclic pattern kernels for predictive graph mining. In *Proceedings of the 10t ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2004), August 22-25, 2004, Seattle, WA, USA* (2004), Kim W., Kohavi R., Gehrke J., DuMouchel W., (Eds.), ACM Press, New York, NY, USA, pp. 158–167. 4

[HSKK01] HILAGA M., SHINAGAWA Y., KOHMURA T., KUNII T. L.: Topology matching for fully automatic similarity estimation of 3d shapes. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques* (2001), ACM, pp. 203–212. 2

[Koe84] KOENDERINK J. J.: The structure of images. *Biological Cybernetics 50*, 5 (1984), 363–370–370. 3

[KRB*13] KOVNATSKY A., RAVIV D., BRONSTEIN M. M., BRONSTEIN A. M., KIMMEL R.: Geometric and photometric data fusion in non-rigid shape analysis. *Numerical Mathematics: Theory, Methods & Applications 6*, 1 (2013). 2

[Lav12] LAVOUÉ G.: Combination of bag-of-words descriptors for robust partial shape retrieval. *The Visual Computer 28*, 9 (2012), 931–942. 2, 3

[LCT*12] LEE W.-J., CHEPLYGINA V., TAX D. M. J., LOOG M., DUIN R. P. W.: Bridging structure and feature representations in graph matching. *IJPRAI 26*, 5 (2012). 4

[LGB*11] LIAN Z., GODIL A., BUSTOS B., ET AL.: Shrec'11 track: shape retrieval on non-rigid 3d watertight meshes. In *Proc. Eurographics/ACM SIGGRAPH Symposium on 3D Object Retrieval* (2011), pp. 79–88. 1

[LGB*12] LIAN Z., GODIL A., BUSTOS B., DAOUDI M., HERMANS J., KAWAMURA S., KURITA Y., LAVOUÉ G., VAN NGUYEN H., OHBUCHI R., ET AL.: A comparison of methods for non-rigid 3d shape retrieval. *Pattern Recognition* (2012). 2

[LH13] LI C., HAMZA A. B.: Symmetry discovery and retrieval of nonrigid 3d shapes using geodesic skeleton paths. *Multimedia Tools and Applications* (2013), 1–21. 2

[Lin94] LINDEBERG T.: *Scale-Space Theory in Computer Vision*. Kluwer Academic Publishers, Norwell, MA, USA, 1994. 3

[LMS13] LAGA H., MORTARA M., SPAGNUOLO M.: Geometry and context for semantic correspondences and functionality recognition in man-made 3d shapes. *ACM Transactions on Graphics (TOG) 32*, 5 (2013), 150. 2

[Mil63] MILNOR J.: *Morse Theory*, vol. 51 of *Annals of Mathematics Studies*. Princeton University Press, 1963. 3

[PFJFG10] 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* (New York, NY, USA, 2010), 3DOR '10, ACM, pp. 69–74. 4

[PP93] PINKALL U., POLTHIER K.: Computing discrete minimal surfaces and their conjugates. *Experimental mathematics 2*, 1 (1993), 15–36. 2

[RTG98] RUBNER Y., TOMASI C., GUIBAS L.: A metric for distributions with applications to image databases. In *Computer Vision, 1998. Sixth Int. Conference on* (1998), pp. 59–66. 5

[RWP06] REUTER M., WOLTER F.-E., PEINECKE N.: Laplace–beltrami spectra as shape-dna of surfaces and solids. *Computer-Aided Design 38*, 4 (2006), 342–366. 2

[SHVS12] SMEETS D., HERMANS J., VANDERMEULEN D., SUETENS P.: Isometric deformation invariant 3d shape recognition. *Pattern Recognition 45*, 7 (2012), 2817–2831. 2

[SMKF04] SHILANE P., MIN P., KAZHDAN M., FUNKHOUSER T.: The princeton shape benchmark. In *Shape Modeling Applications, 2004. Proceedings* (2004), IEEE, pp. 167–178. 6

[SOCG10] SKRABA P., OVSJANIKOV M., CHAZAL F., GUIBAS L.: Persistence-based segmentation of deformable shapes. In *CVPR Workshop on Non-Rigid Shape Analysis and Deformable Image Alignment* (June 2010). 3

[SOG09] SUN J., OVSJANIKOV M., GUIBAS L. J.: A concise and provably informative multi-scale signature based on heat diffusion. *Comput. Graph. Forum 28*, 5 (2009), 1383–1392. 1, 3

[SRB07] SUARD F., RAKOTOMAMONJY A., BENRSHRAIR A.: Kernel on bag of paths for measuring similarity of shapes. In *ESANN* (2007), pp. 355–360. 4

[SS01] SCHOLKOPF B., SMOLA A. J.: *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge, MA, USA, 2001. 5

[SSGD03] SUNDAR H., SILVER D., GAGVANI N., DICKINSON S.: Skeleton based shape matching and retrieval. In *Proc. Shape Modeling International* (Washington, DC, USA, 2003), IEEE Computer Society, pp. 130–. 2

[TS05] TUNG T., SCHMITT F.: The augmented multiresolution reeb graph approach for content-based retrieval of 3d shapes. *International Journal of Shape Modeling 11*, 01 (2005), 91–120. 2

[TV08] TANGELDER J. W., VELTKAMP R. C.: A survey of content based 3d shape retrieval methods. *Multimedia tools and applications 39*, 3 (2008), 441–471. 1

[WS00] WYSZECKI G., STILES W. S.: *Color Science: Concepts and Methods, Quantitative Data and Formulae (Wiley Series in Pure and Applied Optics)*, 2 ed. Wiley-Interscience, Aug. 2000. 5