3D articulated object retrieval using a graph-based representation

Alexander Agathos, Ioannis Pratikakis, Panagiotis Papadakis, Stavros Perantonis, Philip Azariadis, Nickolas S. Sapidis

The Visual Computer manuscript No.
(will be inserted by the editor)

Alexander Agathos · Ioannis Pratikakis · Panagiotis Papadakis · Stavros Perantonis · Philip Azariadis · Nickolas S. Sapidis

3D Articulated Object Retrieval using a graph-based representation

Abstract In this paper, a retrieval methodology for 3D articulated objects is presented that relies upon a graph-based object representation. The methodology is composed of a mesh segmentation stage which creates the Attributed Relation Graph (ARG) of the object along with a graph matching algorithm which matches two ARGs. The graph matching algorithm is based on the Earth Movers Distance (EMD) similarity measure calculated with a new ground distance assignment. The superior performance of the proposed retrieval methodology against state of the art approaches is shown by extensive experimentation that comprise the application of various geometric descriptors representing the components of the 3D objects that become the node attributes of the ARGs as well as alternative mesh segmentation approaches for the extraction of the object parts. The performance evaluation is addressed in both qualitative and quantitative terms.

Keywords 3D Articulated Object Retrieval · Mesh segmentation · Graph matching

A. Agathos · P. Papadakis · S. Perantonis
Computational Intelligence Laboratory
Institute of Informatics and Telecommunications
NCSR 'Demokritos', Greece
E-mail: {agalex,ppapadak,sper}@iit.demokritos.gr

I. Pratikakis
Department of Electrical and Computer Engineering
Democritus University of Thrace, Greece
E-mail: ipratika@ee.duth.gr

P. Azariadis
Department of Product and Systems Design Engineering
University of the Aegean, Greece
E-mail: azar@aegean.gr

N. Sapidis
Department of Mechanical Engineering
University of Western Macedonia, Greece
E-mail: nsapidis@uowm.gr

1 Introduction

Recent advances in 3D object digitization have created a plethora of 3D objects available for processing in various contexts like game industry, cad, medicine, cultural heritage, etc. The wide availability and continuous increase of bandwidth to access the Internet is making feasible to widely share these objects leading to a tendency towards constructing large 3D databases. The continuous increase of those databases' size have made a necessity the construction of retrieval algorithms that enable efficient and effective 3D object retrieval from either public or proprietary 3D databases. 3D object retrieval is the process which retrieves 3D objects from a database in a ranked order so that the higher the ranking of an object the better the match to a 3D object query is by using a measure of similarity. Most of the approaches which address this problem use descriptors which express the object's global shape [7, 10, 11, 14, 17, 21–24, 30]. However, most of these approaches fail to consistently compensate for the intra-class variability of articulated objects. This occurs because it is not evident how a global descriptor will become invariant to non-rigid transformations like bending or stretching, thus, resulting in an erroneous matching.

In this paper, a retrieval methodology is presented which is based upon a graph-based representation that is built after a 3D mesh segmentation. The motivation of this approach originates from object recognition where the object is described in terms of its components that are characterized by geometric features and relational connections with each other. This description is referred to as the *structural description* of the object [5]. In order to recognize an object, its structural description is compared with the structural descriptions of already classified objects and the object is classified to the class of the best match. This recognition process can be naturally adopted for 3D object retrieval. Meaningful components of the object can be extracted using a segmentation algorithm. The structural description of the object is created by using the *Attributed Relational Graph* (ARG) con-

cept, i.e. the components of the object are represented as the nodes of a graph and the relationship of the components with each other are represented as the edges of the graph. To each node *unary* attributes are assigned which describe the geometric characteristics of the component and to each edge *binary* attributes are assigned which describe the relationship of the connected nodes.

Eventually, the problem of matching a query object with the objects stored in the database is transformed into the problem of matching their ARGs [18,27]. The proposed graph matching algorithm is based on the Earth Mover's Distance (EMD) similarity measure.

In this paper, the contribution consists of a complete methodology for retrieval of 3D articulated objects that relies upon a graph-based representation which is produced after a meaningful new mesh segmentation as well as a similarity measure that is based on EMD for which a new ground distance assignment is introduced.

The paper is organized as follows. Section 2 discusses the related work. Section 3 is dedicated to the detailed description of the proposed methodology. In Section 4, the experimental evaluation is presented while in Section 5 conclusions are drawn.

2 Related Work

Among the existing 3D object retrieval methods, two main categories can be distinguished :

i. Methods with global shape representations;
ii. Methods with graph-based shape representations.

The first category can be further classified according to the spatial dimensionality of the information used for retrieval, i.e. 2D, 3D and their combination.

Methods that use 2D information for retrieval use descriptors that are generated from image-projections that may be contours, silhouettes, depth buffers, etc. Chen *et al.* [8] introduce the light field descriptor. This descriptor is constructed by combining a region shape descriptor and a contour shape descriptor computed on a set of orthogonal projections of the model with viewpoints taken on the vertices of a dodecahedron enclosing the object. Retrieval is achieved by comparing the descriptors of all pair of images generated by the different projections of the query's object with the ones of each of the object stored in a database. Vranic [30] proposed a shape descriptor that is constructed by calculating the Fourier coefficients on the depth buffers derived by projecting the object on the four sides of the cube which surrounds the 3D object. Similarity between the query's object and each of the object stored in the database is judged by comparing their corresponding descriptor fourier coefficients with a suitable metric. In the method proposed by Ohbuchi *et al.* [21] multiscale features are computed from a set of projections that are taken from the vertices of a polyhedron enclosing the object. All the features of the objects in the Database construct a visual codebook using k-means. The descriptor of the object is derived by quantizing all the features of the object using the visual codebook into a vector containing the frequencies of the visual words. Retrieval is achieved by computing the Kullback-Leibler divergence between the descriptors of the objects. Passalis *et al.* [24] constructed a descriptor by calculating and weighting appropriately the Fourier coefficients derived from the depth buffers acquired after projecting the object on the four sides of the cube which surrounds the 3D object.

Methods that use 3D information derive their descriptors from the geometry of the 3D object. Vranic [30] introduced a descriptor which describes an object by a spherical extent function which captures the furthest intersection points of the object's surface with rays emanating from the origin of the sphere enclosing the object. The spherical extent is represent by spherical harmonics in the frequency domain. Jain and Zhang [14] created a descriptor which is based on spectral analysis using geodesic and Euclidian distances. The spectral analysis creates a set of eigenvalues for each object. In their retrieval process the querys eigenvalues are compared against the eigenvalues of each of the model stored in the database. In [11] Gal *et al.* constructed a density function using a pose oblivious shape diameter function and is combined with the centricity function in order to construct histograms which describe the shape of the object. Ben-Chen and Gotsman [3] introduced a discrete conformal scaling factor which identifies the extrusions of the object. In this work, the histogram of the conformal map is used as the descriptor of the mesh which was shown to be pose invariant. Bronstein *et al.* [6] uses intrinsic and extrinsic metrics in order to calculate the distance of two surfaces. The extrinsic metric calculates the rigid difference of two surfaces while the intrinsic metric expresses the similarity of two object disregarding the articulations that the different part perceive. Papadakis *et al.* [22] introduces a volumetric spherical function based representation of the object which is expressed by spherical harmonics. Methods that combine both 2D and 3D information have also been developed in order to improve the retrieval performance [7,10,23,30,29].

In the second category of retrieval methods, a descriptor is constructed based on the structural description of the object which in most of the cases is represented by a graph structure. Hilaga *et al.* [12] proposed a descriptor based on Reeb graph theory, specifically the object is described by a multiresolution reeb graph stucture and matching is achieved by the comparison of the reeb graph structures on different resolution levels. Tung and Schmitt [28] enhanced the retrieval performance of [12] by augmenting the multiresolution Reeb graph structure with geometrical and visual information. Biasotti *et al.* [4] constructed a descriptor based also on Reeb graph theory with the difference of being created by a finite set of contour levels. They call their representation Ex-

tended Reeb Graph with the aid of which they create a directed acyclic graph structure attributed with the geometric properties of each of the patches that each of the nodes represent. Retrieval is achieved by matching the directed acyclic graphs. Cornea *et al.* [9] extract the skeletons of the 3D objects from their volumetric representations using a generalized potential field generated by charges placed on the surface of the object. Retrieval is achieved by matching the skeletal graphs using an extension of the EMD similarity measure. Sundar *et al.* [26] extract also the skeletons of the 3D objects from their volumetric representation using a volumetric thinning approach. Using information from their volumetric thinning they direct the skeletons creating by this way directed graphs. Retrieval is achieved by matching the directed graphs using a recursive, depth first formulation of bipartite graph matching. In [18], the object is first voxelized and then segmented using a morphological structure. The extracted components create an Attributed Relational Graph. The query's ARG is matched against the ARGs stored in the database using an EMD-based approach. In [27] the mesh is decomposed into its meaningful components and the ARG of the object is constructed based on their decomposition. Retrieval is achieved by matching the query's ARG with the ARGs of the objects stored in the database using an error correcting graph isomorphism algorithm. In [20], the structural description of the object in the form of a graph is also used. Their methodology comprises two steps: first, they compute a common subgraph for each class of the database and then they define a set of editing operations based on the subgraph. These two steps allow them to construct a prototype for each class to which the query object is matched.

Considering the retrieval of articulated objects few algorithms that belong in the first category can provide efficient results [14,21,11,3,6]. On the other hand, algorithms that belong to the second category can efficiently handle articulated objects since the representation used to describe them is pose invariant in most of the cases. The only drawbacks that the latter algorithms exhibit are that in some cases complicated graph structures are constructed with the consequence of making their matching complexity high thus decreasing the time efficiency of retrieval, also the graph structures in some cases are susceptible to geometrical or topological noise.

The proposed retrieval algorithm belongs to the second category. A new meaningful mesh segmentation algorithm extracts the main components of the 3D object creating its ARG. The retrieval is going to be accomplished by matching the ARGs with an EMD-based matching algorithm.

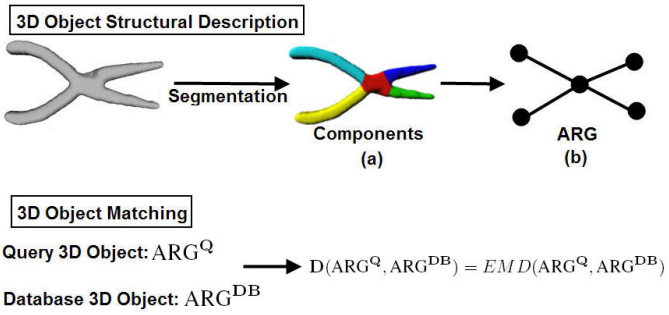


Fig. 1 The stages of the proposed retrieval methodology

3 The proposed methodology

The proposed retrieval methodology comprises three distinct stages, as shown in Fig. 1.

i. The query object is segmented into its constituent meaningful components using the proposed 3D mesh segmentation methodology (Fig. 1(a))
ii. The segmented components of (i) are used to build the ARG of the query's object (Fig. 1(b));
iii. The query's ARG is compared against each ARG of the distinct 3D objects that comprise the Database using an EMD-based graph matching algorithm.

It should be noted that the ARGs of the database are constructed in the same manner as the ARG of the query model in an off-line stage. The matching between the query's ARG and the ARG of an object in the database provides a distance measure (denoted as D in Fig. 1) which measures the similarity of the two objects and is computed based on the EMD.

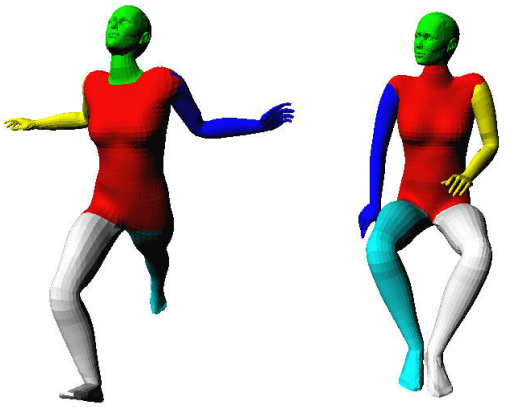
A detailed description of all the aforementioned stages will be given in the sequel.

3.1 3D Mesh Segmentation

In this section, the basic principles of the first stage in the proposed retrieval methodology will be given, i.e. the 3D mesh segmentation stage where the object is segmented into its constituent meaningful components. This is a critical stage since the components extracted from the segmentation algorithm define the ARG of the object. A detailed description of the mesh segmentation scheme used in this paper is given in [1].

When dealing with articulated objects, an efficient segmentation algorithm should be insensitive to the various poses that the mesh may take. The proposed segmentation algorithm can meet this requirement. An example is shown in Fig. 2, wherein although a 'human' 3D object takes different poses, the acquired segmentation in both cases is compatible, i.e. the segmentation algorithm is consistent in always segmenting the human object into its main body, legs, arms and head.

The proposed segmentation algorithm is based on the premise that the 3D object consists of a main (core) body and its constituent protrusible components. It can be summarized in the following stages. Initially the *salient*



(a) (b)

Fig. 2 (a), (b) Example of the proposed segmentation of a 'human' 3D object at different poses

points of the mesh which characterize the protrusions of the mesh are extracted. These points are further clustered according to their geodesic proximity where each cluster represents a main component of the object and each of them is assigned a unique representative point. In the next stage, the core (main body) of the mesh is approximated using the minimum cost paths that the aforementioned representatives create with each other. In the sequel, the boundary between the core and each of the protrusions (Partitioning Boundary) is approximated using closed boundaries which span the area containing the partitioning boundary. Finally the approximated partitioning boundary is refined using the minimum cut algorithm of Katz *et al.* [16].

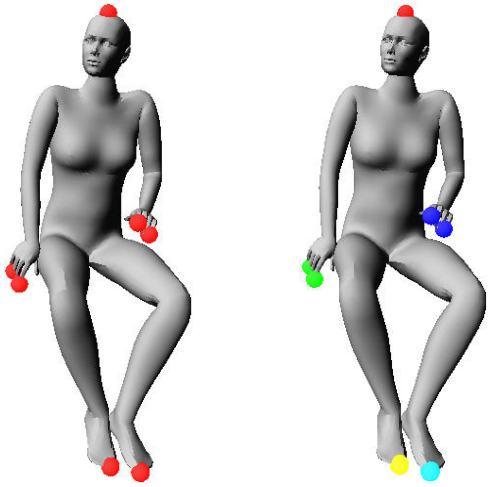
All of the stages of the proposed segmentation methodology will be detailed in the following sections.

3.1.1 Salient Points Extraction and Clustering stage

In this section, the salient points of the mesh will be extracted and a clustering methodology to group them into clusters representing a main protrusion of the mesh will be presented.

Intuitively, the salient points of the mesh should reside on the tips of its protrusions. A possible solution for finding them is to use a function which takes high values at the protrusions of the mesh and its local maxima are the tips of the protrusions.

A function which can achieve the requirements set above was first introduced by Hilaga *et al.* [12] and is



(a) (b)

Fig. 3 Example of the 'human' 3D mesh with its corresponding salient points at the (a) extraction stage (red dots) and (b) clustering stage - each color represents a different cluster

defined for each point v of the surface S of a 3D object as:

$$pf(v) = \int_{\mathrm{p}\in\mathrm{S}} g(v,\mathrm{p})\,\mathrm{dS} \tag{1}$$

where $g(v,\mathrm{p})$ denotes the geodesic distance between v, p. This function is called in [2] *protrusion function, pf()*.

From the function's definition it can be observed that small values correspond to points of the mesh which are near the center of the mesh while large values correspond to points that are at the protrusions of the mesh. Thus, the protrusion function meets the necessary requirements for the calculation of the salient points.

This function for a 3D mesh is approximated using a tessellation of its surface into compact regions, such that (1) is transformed to:

$$pf(v) = \sum_i g(v,b_i)\,area(V_i) \tag{2}$$

where b_i denotes the center of the region V_i.

Also, another approximation of the protrusion function might alternatively be used as in [15]:

$$pf(v) = \sum_{v_i\in\mathrm{S}} g(v,v_i) \tag{3}$$

where v_i denotes the vertices of the mesh.

For every $v \in \mathrm{S}$ a neighborhood of points N_v is defined which can be either:

- *a k-ring neighborhood* defined as the set of vertices within k edges away from vertex v;
- *a geodesic neighborhood* defined as the set of vertices for which the geodesic distance from vertex v is less than a threshold. This threshold is called the *radius* of the geodesic neighborhood.

The salient point of a mesh is formally defined as:

$$v \text{ is a salient point } \Longleftrightarrow \begin{cases} pf(v) > pf(v_i)\ \forall\, v_i \in \mathrm{N}_v \\ pf(v) > 0.45 \quad pf(v) \text{ normalized in } [0,\ 1] \end{cases} \quad (4)$$

Definition (4) ensures that the salient point will reside at the tip of a protrusion. In our implementation, N_v is set as a geodesic neighborhood with radius $\sqrt{5 \cdot 10^{-3} \cdot area(\mathrm{S})}$ as also proposed in [19].

It often happens that the extracted salient points belong to sub-components of the objects. For example, in Fig. 3(a) there exist salient points that correspond to the fingers of the 'human' model. Since the salient points are used in the proposed segmentation algorithm to represent a single protrusion it is necessary to cluster them, each one of the clusters representing a single protrusion of the object. Thus the fingers of the 'human' model in Fig. 3(a) need to be grouped in one cluster in order to represent the arms of the object.

The salient points that are required to be clustered are those which are close to each other in terms of geodesic distance. Once the salient points are grouped the salient point with the largest protrusion value is chosen as the representative of each cluster and is called the *representative salient point*.

In Fig. 3(b), the result of the clustering of the salient points in the 'human' object is shown. As it can be observed each cluster represents a unique protrusible component of the object.

3.1.2 Core Approximation

As already mentioned, the proposed segmentation algorithm assumes that the mesh approximating the 3D object consists of a main body (its core) and its protrusible parts. An effective algorithm which approximates the core of the mesh should acquire all the elements (vertices or faces) of the mesh except those that belong to its protrusions. Towards this concept an algorithm is proposed that uses the minimum cost paths between the representative salient points found in Section 3.1.1.

Specifically, let assume $\hat{S} = \{\hat{s}_i,\, i = 1, \ldots,\ N_C\}$ be the set of representative salient points, where N_C denote the number of clusters found in section 3.1.1 and $\hat{s}_i$ the representative of the i^{th} cluster.

Also, let $P = \{P_{ij}, i, j \in \{1, \ldots, N_C\}\}$ be the set of all minimum cost paths of the points of $\hat{S}$, where P_{ij} denote the minimum cost path between $\hat{s}_i$, $\hat{s}_j$. The idea of the core approximation algorithm is to expand a set of vertices in ascending order of protrusion function value until the set contains a certain percentage of all elements of P. The pseudo-code of the proposed core approximation algorithm is shown in Fig. 4. Initially, the vertices of the mesh M are inserted in a priority queue *PFHeap* in which the vertex with the minimum protrusion function is extracted first. The algorithm proceeds by extracting points from the priority queue which incrementally expands the list *CoreList* where the approximation of the core is stored. A path P_{ij} in P remains *active* if the ratio of the number of vertices in the path P_{ij} which have been visited during expansion over the total number of vertices that the path contains is less than $\mathrm{t_C}$ which is equal to 0.15. A salient point $\hat{s}_i \in \hat{S}$ remains *active* if $\exists P_{ij}$ for some j $\in$ {1,...,N_C} $\neq$ i : P_{ij} active. A vertex v of the Mesh *CanBeAdded* in *CoreList* if its geodesic nearest salient point in $\hat{S}$ is active. *StopGrowing* becomes 'TRUE' when all salient points become non-active.

```
1: for all vertices v ∈ M do
2:    insert v in PFHeap with priority pf(v)
3: end for
4: StopGrowing = false
5: while !StopGrowing do
6:    pop a vertex v from PFHeap
7:    if v CanBeAdded then
8:       CoreList.add(v)
9:    end if
10:   for all P_ij ∈ P do
11:      if P_ij.active then
12:         if v ∈ P_ij then
13:            increment P_ij.counter
14:            if P_ij.counter / P_ij.SizeOfPath ≥ t_c then
15:               P_ij.active = false
16:            end if
17:         end if
18:      end if
19:   end for
20:   for all ŝ_i ∈ Ŝ do
21:      if ŝ_i.active then
22:         ŝ_i.active = false
23:         for all ŝ_j ∈ Ŝ − ŝ_i do
24:            if P_ij.active then
25:               ŝ_i.active = true
26:            end if
27:         end for
28:      end if
29:   end for
30:   //StopGrowing becomes true if all ŝ_i become non active
31: end while
```

Fig. 4 The pseudo-code of the proposed core approximation algorithm

In Fig. 5 the core approximation of the 'human' 3D object is presented. It can be observed that the proposed algorithm approximates consistently the core of the object and that its boundaries are near the partitioning boundaries of the object.

3.1.3 Partitioning Boundary Detection

In this section, the stage of the segmentation algorithm that finds the partitioning boundary is presented. This boundary separates a protrusion from the main body of the 3D object. At the area which divides the main body from the protrusion, it is considered that a sud-

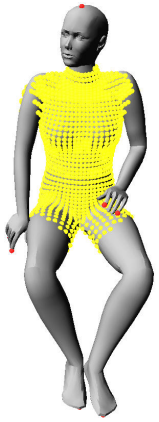


Fig. 5 Example of core approximation for the 'human' 3D object. The vertices representing the core are coloured in yellow

den change of object volume should occur, delimiting the partitioning boundary. The proposed algorithm aims to detect this abrupt change by examining the perimeter of closed boundaries placed at an area which contains the partitioning boundary.

These closed peripheries are constructed using a distance function D which is associated to the salient point of the cluster representing the protrusion (section 3.1.1). Formally, for a salient point $\hat{s}$, which is the representative of a cluster representing the protrusion, the distance function D is defined for every point v of the mesh as the shortest distance between v and $\hat{s}$. The shortest distance is computed using the Dijkstra algorithm with source $\hat{s}$ while each of the edges (u,v) is assigned the following cost term:

$$\text{cost}(\text{u}, v) = \delta \frac{\text{length}(\text{u}, v)}{\text{avg\_length}} + (1 - \delta) \frac{\text{prot}(\text{u}, v)}{\text{avg\_prot}} \quad (5)$$

where $\text{prot}(u, v) = |pf(\text{u}) - pf(v)|$ and avg_length, avg_prot denote the average values of the length and protrusion difference of the edges of the mesh, respectively. This distance function was introduced in [19]. In our implementation, we set δ equal to 0.4.

Using the distance function D the closed boundaries are constructed by interpolating on the mesh isocontours generated by setting constant values on the function D. Taking also advantage of the proximity between the core approximation boundaries and the mesh partitioning boundaries, the area that should contain the partitioning boundary is the part of the mesh whose values of D lie in the interval $[(1 - \text{d}_1)\,\text{D}_{\text{coremin}}, (1+\text{d}_2)\,\text{D}_{\text{coremin}}]$. $\text{D}_{\text{coremin}}$ denotes the value of the distance function between the nearest point of the core approximation and the representative $\hat{s}$, while d_1, d_2 denote the extent of the interval ($0 < \text{d}_1 < 1$, $\text{d}_2 > 0$). In this work, we set $\text{d}_1 = 0.1$, $\text{d}_2 = 0.4$.

In order to approximate the partitioning boundary, this area is swept by the closed boundaries in fixed steps equal to $\frac{(\text{d}_1+\text{d}_2)\text{D}_{\text{coremin}}}{l_{per}}$, where $l_{per} = 12$ and the sweeping is terminated when the ratio of the perimeters between successive closed boundaries exceeds a certain threshold equal to 1.3. When the ratio between successive perimeters becomes greater than the threshold then the abrupt change in the volume of the object is signified and the closed boundary where this occurs is considered to be the approximation of the protrusion boundary.

Choosing the representative of the cluster representing the protrusion as a source of the distance function D may lead to the creation of skewed closed boundaries. This choice is refined by properly selecting as source the point that has the minimum protrusion value on an area enclosing the salient points of the cluster. This source point leads to the creation of closed boundaries that are positioned near to the true partitioning boundary.

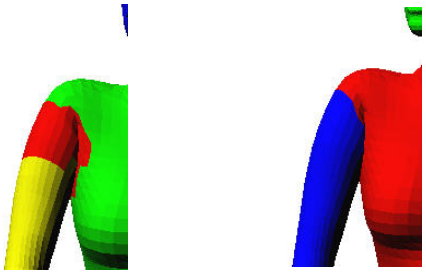
3.1.4 Partitioning boundary refinement

The partitioning boundary detected in section 3.1.3 is an iso-contour of the distance function D approximating the true partitioning boundary. In most of the cases, this approximation is rough, i.e. it deviates from the true partitioning boundary. As mentioned in section 3.1.3, the partitioning boundary is delimited at the area where there is a sudden change in the volume between the main body and the protrusion while taking into account Hoffman and Richards [13] it should reside at the concavities of the object. The partitioning boundary approximation of section 3.1.3 is not constrained to the concavities wherein the true partitioning boundary pass through, thus, there is a need to refine the partitioning boundary approximation so that it passes through the concavities.

To this end, a Region **C** is constructed that contains the true partitioning boundary, as in the following. First, the calculation of the average geodesic distance (AvgGeodDist)is addressed, between the partitioning boundary approximation and the refined representative that have both been detailed in section 3.1.3. Then, region **C** is defined as the set of mesh triangles whose vertices geodesic distance from the refined representative lies in the interval [0.9·AvgGeodDist, 1.1·AvgGeodDist]. Fig. 6(a) illustrates this region in the 'human' model. As it can be observed, this region contains the true partitioning boundary. For segmenting the object at the exact partitioning boundary the minimum-cut methodology of Katz and Tal [16] is used. Specifically, a flow network graph is constructed using the dual graph of the mesh [2]. In order to construct the network of [16], two additional regions are defined; Region **A** containing the triangles of the protrusion of the mesh (yellow triangles of fig. 6(a)) and region **B** containing the faces of the remainder of the mesh (green triangles of fig. 6(a)). Region **C** plays the role of the fuzzy region explained in [16].

Taking into account all three aforementioned regions, a flow network as in [16] is constructed in order that the application of the minimum cut algorithm on this

network will lead to the mesh segmentation on the true partitioning boundary (Fig. 6(b)).



(a) (b)

Fig. 6 Example of the partitioning boundary refinement stage: (a) region **A** is shown with yellow, region **B** is shown with green and region **C** is shown with red; (b) The final segmentation of the protrusion from its main body after the application of the minimum cut algorithm

3.2 EMD-based Matching

As has already been mentioned from the very beginning of the description of the proposed retrieval methodology in order to match the query object with those objects contained in a database, a graph matching algorithm is required to match the query's ARG with each of the corresponding object ARG in the database. In this section, the creation of the object's ARG will be described along with the proposed graph matching algorithm between two ARGs.

The proposed segmentation algorithm is capable to segment an object into its core (main body) and its protrusible parts. Taking advantage of this capability a simple ARG can be constructed, its nodes are the segmented components and each of the nodes representing a protrusible part is connected with the node representing the core of the object forming by this way the edges of the ARG. A segmented 'Human' object and its corresponding graph structure is shown in Fig. 7. Unary and binary attributes will be assigned to the nodes and edges of the ARG respectively. In this manner, the two ARGs that need to be matched are constructed. The matching algorithm will find the correspondences of the nodes between the two ARGs and will provide a distance measure which quantifies the degree of similarity of the two graphs.

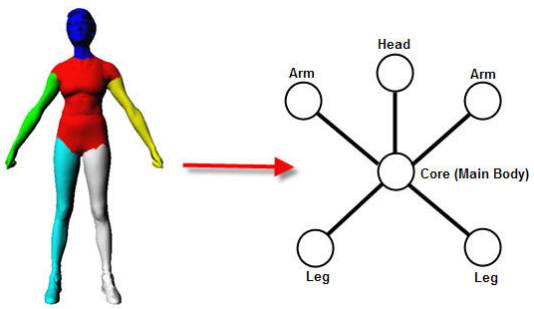


Fig. 7 The graph structure of the segmented 'Human' model

Formally, let $G = (V, E, \mathbf{U}, \mathbf{B})$, $\hat{G} = (\hat{V}, \hat{E}, \hat{\mathbf{U}}, \hat{\mathbf{B}})$ be the attributed relational graphs that need to be matched, where $V = \{v_i\}_{i=1}^{n}$, $\hat{V} = \{\hat{v}_j\}_{j=1}^{m}$ are the nodes (v_1, $\hat{v}_1$ represent the core component of the two objects respectively), $E = \{r_{1i}\}_{i=2}^{n}$, $\hat{E} = \{\hat{r}_{1j}\}_{j=2}^{m}$ are the edges, $\mathbf{U} = \{\mathbf{u}_i\}_{i=1}^{n}$, $\hat{\mathbf{U}} = \{\hat{\mathbf{u}}_j\}_{j=1}^{m}$ are the unary attributes of the nodes and $\mathbf{B} = \{\mathbf{b}_i\}_{i=2}^{n}$, $\hat{\mathbf{B}} = \left\{\hat{\mathbf{b}}_j\right\}_{j=2}^{m}$ are the binary attributes of the edges of the two graphs, respectively. Let assume that $n \geq m$. As already mentioned, it is assumed that the nodes v_1, $\hat{v}_1$ represent the core component of the two models, respectively. These nodes are considered as *fixed* and are always matched in the matching algorithm. Also additional $n - m$ nodes, $\{\hat{v}_j\}_{j=m+1}^{n}$, are inserted in $\hat{G}$ which are called in this work *delete* nodes. The reason for doing this is to penalize the $n - m$ nodes of G that are not mapped to any of the nodes of $\hat{G}$. All other nodes are considered as *normal*. Unary attributes $\hat{\mathbf{U}}_d = \left\{\hat{\mathbf{u}}_{d_j}\right\}_{j=m+1}^{n}$ are assigned to the delete nodes that correspond to components with no information.

In this paper, the similarity of the two ARGs is measured by the Earth Mover's Distance (EMD) [25]. In general, the EMD computes the distance between two distributions, which are represented by two signatures. The signatures are sets of weighted features that capture the distributions. The EMD expresses the least amount of work needed to transform one signature to another.

In our case, the two ARGs are considered as the distributions and the two signatures are the set of nodes $V = \{v_i\}_{i=1}^{n}$, $\hat{V} = \{\hat{v}_j\}_{j=1}^{n}$ of each of the graphs G, $\hat{G}$, respectively. A uniform distribution of weights $\{w_i\}_{i=1}^{n}$, $\{\hat{w}_j\}_{j=1}^{n}$ are assigned to the nodes, respectively, and each of them is equal to $\frac{1}{n}$. In this manner, the signatures $S = \{v_i, w_i\}_{i=1}^{n}$, $\hat{S} = \{\hat{v}_j, \hat{w}_j\}_{j=1}^{n}$ are constructed.

Intuitively, the set of weights $\{w_i\}_{i=1}^{n}$ can be considered as piles of earth that needs to be transferred to the holes that the other set of weights create in the feature space. Each unit of earth is transferred from pile i to hole j with cost $d(v_i, v_j)$ (called *ground distance*). This transfer symbolizes the matching of node v_i to node $\hat{v}_j$ under a certain cost (distance measure). The total amount of earth (weight) that is transferred from pile i (node v_i) to hole j (node $\hat{v}_j$) is denoted as $f^{(i,j)}$ and is called the *flow* of weight. The transportation problem is solved with a linear programming optimization approach that finds the optimal flow of weight between the two distributions [25].

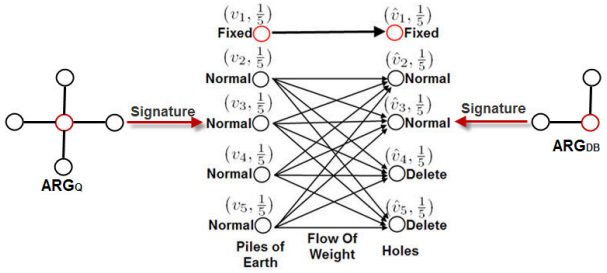


Fig. 8 The proposed matching scheme between two ARGs

The optimal cost of the optimization process is the EMD that is defined as follows:

$$\text{EMD} = \sum_{i=1}^{n}\sum_{j=1}^{n} f^{(i,j)} d(v_i, \hat{v}_j) \tag{6}$$

As can be seen in Equation (6), the EMD is a distance measure between the two signatures since it is a weighted sum of the ground distances and expresses the similarity of the two signatures, thus the similarity of the two ARGs. It can also be observed that the ground distances are the definitive terms of the EMD thus the whole matching process is based on their proper definition because they indicate how the nodes are matched. In our case the ground distances depend upon the unary and binary attributes of the ARGs since these attributes should define how the matching between the nodes of the graphs should be addressed.

In the matching process, the fixed nodes of the two graphs that indicate the core elements (v_1, $\hat{v}_1$) should always be matched, thus, there is a need to constrain the optimization process for the calculation of the EMD in order to always match the fixed nodes. All other nodes can be matched without any constraint. In Fig. 8, the proposed matching between two ARGs in the form of signatures is shown, wherein the first ARG consists of five and the other of three nodes.

In order to achieve the aforementioned matching the following ground distance is defined:

$$d(v_i, \hat{v}_j) = \begin{cases} 3\frac{D_{normal}(v_i,\hat{v}_j)}{1+D_{normal}(v_i,\hat{v}_j)} & \text{if } v_i, \hat{v}_j \text{ normal} \\ 3\frac{D_{fixed}(v_i,\hat{v}_j)}{1+D_{fixed}(v_i,\hat{v}_j)} & \text{if } v_i,\ \hat{v}_j \text{ fixed} \\ 5\frac{D_{delete}(v_i,\hat{v}_j)}{0.1+D_{delete}(v_i,\hat{v}_j)} & \text{if } \begin{matrix} v_i \text{ normal,} \\ \hat{v}_j \text{ delete} \end{matrix} \\ \infty & \text{otherwise} \end{cases} \tag{7}$$

where,

$$\begin{aligned} D_{normal}(v_i,\hat{v}_j) &= \sqrt{\|\mathbf{u}_i - \hat{\mathbf{u}}_j\|^2 + \|\mathbf{b}_i - \hat{\mathbf{b}}_j\|^2} \\ D_{fixed}(v_i,\hat{v}_j) &= \sqrt{\|\mathbf{u}_i - \hat{\mathbf{u}}_j\|^2} \\ D_{delete}(v_i,\hat{v}_j) &= \sqrt{\|\mathbf{u}_i - \hat{\mathbf{u}}_{d_j}\|^2} \end{aligned} \tag{8}$$

As can be seen in equation (7), in the case of matching a normal node with a delete node there exists a ground distance for which its derivative is much steeper than the derivative of the ground distances in the cases when the fixed nodes and the normal nodes are matched (see Fig. 9). This occurs in order to avoid the matching of normal nodes that hold significant information with the delete nodes that hold no information.

It can also be observed in equation (8) that the binary attributes are considered only in the normal nodes since we want to exploit the relation that they have with the fixed node (core). When the fixed nodes are matched only the unary attributes are considered since the core relation with the other nodes is already considered when the normal nodes are matched.

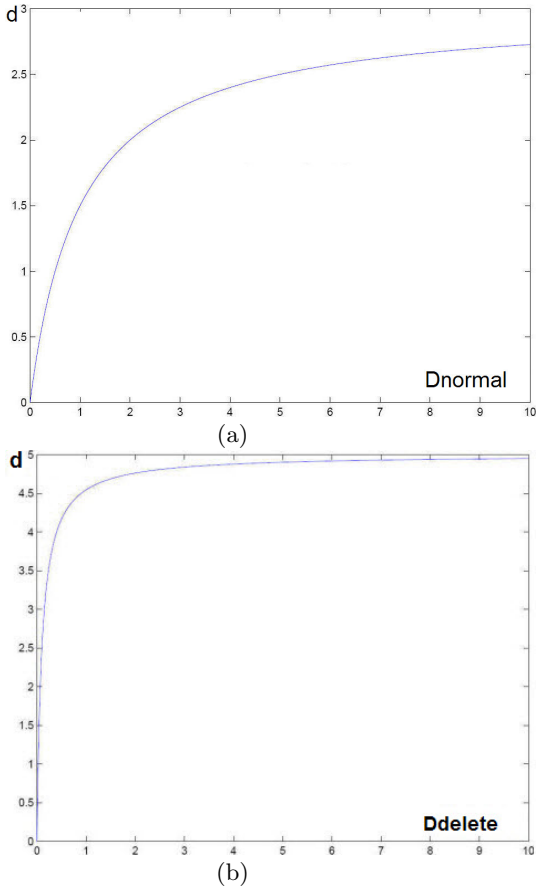


Fig. 9 Indicative ground distance plots in the case of (a) the normal node matching and, (b) the delete node matching

Note also that with the selected ground distance the fixed nodes are always going to be matched.

The Unary attributes that need to be defined for the nodes of the ARG should carry the geometric properties of the component they represent. These properties may, for example, be the relative size, the convexity of the components or they can be described in the frequency domain using spherical harmonics. The binary

attributes should express the relationship that the neighboring components have, e.g., the distance of the centroids of the neighboring components. In this paper the following unary and binary attributes are used :

i. The unary and binary attributes of Kim *et al.* [18]. The purpose of this assignment is to compare the proposed matching methodology with that used in [18] in order to show the efficiency of the proposed segmentation and matching algorithms.
ii. Unary attributes defined by Papadakis *et al.* [22] descriptor. The descriptor consists of spherical harmonic coefficients derived from the object's component after pose normalization. The spherical harmonics provide a description of the component's geometry in the frequency domain. For further details see [22].

Considering Kim *et al.* [18] attribute assignment, the unary attributes that are assigned to the nodes of the ARG representing the object components are the relative size (rs) of the component, the convexity (c) of the component and the eccentricities (e_1, e_2) of the ellipsoid approximating the component. The relative size of the component is approximated by its area, the convexity is approximated by first voxelizing the component and then dividing the number of voxels of the component by the number of voxels of its convex hull while the eccentricities are approximated by the variances of the component mesh points along the axes created by principal component analysis. The binary attributes that are assigned to the edges of the ARG are the distance (l) of the centroids of the components connected by an edge of the graph and the angles (a_1, a_2) that the two most significant principal axes of the connected components create with each other. All of the attributes are normalized in the interval [0, 1]. By this way, the vector $[rs, c, e_1, e_2]$ is assigned to the normal and fixed nodes and the vector $[l, a_1, a_2]$ is assigned to the edges of the graphs. All delete nodes are assigned the vector $[0, 1, 1, 1]$. In equation (8), the norm $\|\cdot\|$ denotes the $\mathbf{L}_2$ norm of the attribute vectors.

Considering Papadakis *et al.* [22] attribute assignment, we set to the normal and fixed nodes their spherical harmonic descriptor vector. The descriptor consists of two sets of coefficients corresponding to two aligned versions of the model using two methodologies based on principal component analysis, namely CPCA and NPCA. CPCA aligns the component according to the surface area distribution and NPCA aligns the component according to the surface orientation distribution, see [22]. To the delete nodes the vector with zero entries is assigned whose dimension is the same as their descriptor. Please note that in this case we do not assign any binary attributes to the graphs, thus in equation (8) there exist no binary term and the norm $\|\cdot\|$ denotes the $\mathbf{L}_1$ norm of the spherical harmonic vectors which is defined as in [22].

Considering both the aforementioned ground distance assignment and ARG definition, the EMD measure is computed between the two ARGs which denotes the degree of similarity between the two objects that need to be matched. In order to compute the EMD, the implementation of Rubner *et al.* [25] is used.

4 Experimental Results

The evaluation of the proposed retrieval methodology for 3D articulated objects was run on the standard McGill 3D object database [31] and the ISDB database [11] which encounters objects with articulations. In particular the McGill Database contains ten classes that comprise a total of 255 articulated objects, namely, 'Ants', 'Crabs', 'Spectacles', 'Hands', 'Humans', 'Octopuses', 'Pliers', 'Snakes', 'Spiders' and 'Teddy-bears' each one of them containing approximately twenty to thirty models. The ISDB Database contains nine classes that comprise a total of 106 articulated objects, namely 'Cats', 'Dinos', 'Dogs', 'Frogs', 'Hands', 'Horses', 'Humans', 'Lions' and 'Wolfs'.

Since the proposed mesh segmentation algorithm requires that the objects should be manifolds, a transformation for each object to manifolds has been applied.

The experiments addressed in this paper aim to reach a threefold goal. First, the superior performance of the proposed retrieval methodology will be shown against two other state of the art 3D object retrieval methodologies, namely Kim *et al.* [18] and Papadakis *et al.* [23]. The former is based on a graph-based representation using a descriptor and similarity measure that have been adopted by MPEG-7 standardization while the latter uses a global hybrid shape descriptor.

Second, the improved performance of the proposed segmentation algorithm will be shown in terms of retrieval accuracy against the segmentation algorithm used in Kim *et al.* [18] retrieval methodology. This is achieved by accommodating the ARG created by the proposed segmentation algorithm using Kim *et al.* [18] attributes enabling a fair comparison with the original retrieval methodology presented by Kim *et al.*

Finally, the impact of the proposed retrieval methodology for improving the retrieval accuracy in the case of intra-class variability will be shown. In particular, a refinement of the results achieved by Papadakis *et al.* [23] method will be addressed. It is shown that if we encounter the first n retrieved objects achieved by a retrieval method that takes into consideration global shape descriptors like Papadakis *et al.* [23], this portion of the ranked results can be used to apply the proposed retrieval methodology resulting in an updated re-ranking with improved retrieval accuracy.

In the sequel, we will use the following abbreviations:

– The graph-based retrieval methodology that encounters the proposed mesh segmentation and the EMD-based matching using Papadakis *et al.* [22] attributes is denoted as **EMD-PPPT**.

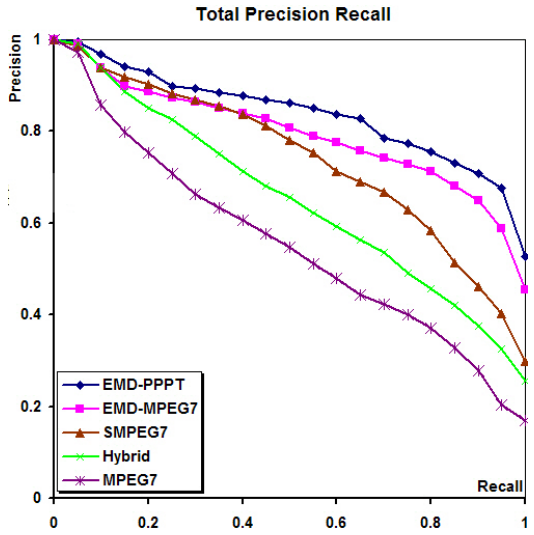


Fig. 10 Precision-Recall curves of the examined retrieval methodologies for the McGill Database

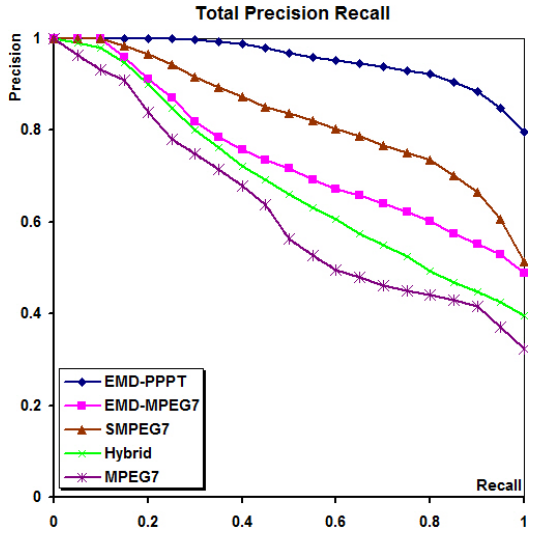


Fig. 11 Precision-Recall curves of the examined retrieval methodologies for the ISDB Database

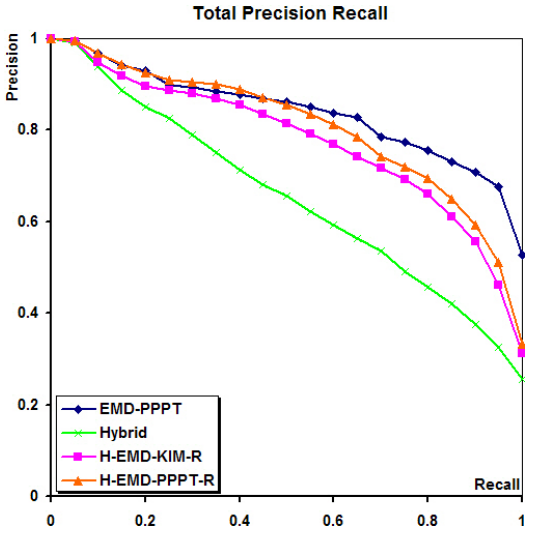


Fig. 12 Precision-Recall curves when a refinement of the ranked results is used in the McGill Database

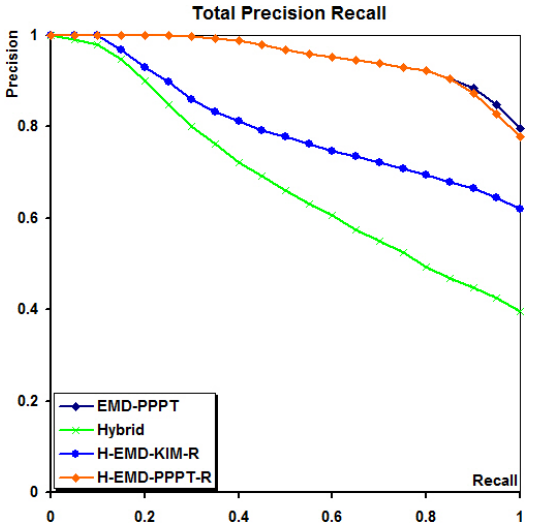


Fig. 13 Precision-Recall curves when a refinement of the ranked results is used in the ISDB Database

– The graph-based retrieval methodology that encounters the proposed mesh segmentation and EMD-based matching using Kim *et al.* [18] attributes is denoted as **EMD-MPEG7**.
– The graph-based retrieval methodology that encounters the proposed mesh segmentation and the graph matching of Kim *et al.* [18] is denoted as **SMPEG7**.
– The graph-based retrieval methodology that encounters the segmentation and matching of Kim *et al.* [18] is denoted as **MPEG7**.
– The retrieval methodology of Papadakis *et al.* [23] that encounters a global shape representation is denoted as **Hybrid**.
– The retrieval methodology of Papadakis *et al.* [23] refined by the proposed retrieval methodology using Kim *et al.* [18] attributes is denoted as **H-EMD-KIM-R**.
– The retrieval methodology of Papadakis *et al.* [23] refined by the proposed retrieval methodology using Papadakis *et al.* [22] attributes is denoted as **H-EMD-PPPT-R**.

Evaluation of the retrieval results achieved by the aforementioned methodologies is based upon Precision-Recall (P-R) diagrams wherein the evaluation was performed by using each model in the dataset as a query on the remaining set of models and computing the average precision-recall performance over all models. Furthermore, the quantitative evaluation was augmented by taking into account the performance measures in the following :

– Nearest Neighbor (NN): The percentage of queries where the closest match belongs to the query's class.

– First Tier (FT): The recall for the (k-1) closest matches, where k is the cardinality of the query's class.
– Second Tier (ST): The recall for the 2(k-1) closest matches, where k is the cardinality of the query's class.
– Discounted Cumulative Gain (DCG): A statistic that weights correct results near the front of the list more than correct results later in the ranked list under the assumption that a user is less likely to consider elements near the end of the list.

These measures range from 0% to 100% and higher values indicate better performance.

In Fig. 10 and Fig. 11, Precision-Recall curves show the performance of all methodologies for 3D object retrieval used against the proposed methodology (EMD-PPPT) for the McGill and ISDB 3D database of articulated objects, respectively. It is shown that EMD-PPPT methodology achieves the best performance. This implies that the spherical-harmonics attributes set on the components of the object can provide a meaningful description that directly leads in high quality retrieval results. Although the chosen attributes for the segmented parts of object are being only unary without any complementary binary attributes, it is shown that EMD-PPPT outperforms EMD-MPEG7 that uses both binary and unary attributes as described in Kim *et al.* [18].

Examining the contribution of the proposed mesh segmentation in the improvement of the performance at the retrieval pipeline process in terms of retrieval accuracy, we made a comparison between SMPEG7 and MPEG7 methodology. Fig. 10 and Fig. 11 clearly indicates the superiority in performance of SMPEG7 which differs from MPEG7 only at the mesh segmentation stage.

Since the importance of the proposed retrieval methodology acquires higher impact in the case of intra-class variability we made an experiment as in the following. We first applied a retrieval methodology with high performance that relies upon a hybrid global shape descriptor and then we applied to part of the m top ranked results the proposed graph-based retrieval methodology using either the Kim *et al.* attributes [18], namely 'H-EMD-KIM-R' or Papadakis *et al.* attributes [22], namely 'H-EMD-PPPT-R'. Fig. 12 and Fig. 13 shows that the refinement of the ranked results by a methodology which can become less error prone to intra-class variability provides improvement to retrieval accuracy. Again, refinement with the proposed graph-based representation along with using the Papadakis *et al.* attributes achieves the highest performance.

In Table 1 and Table 2 the corresponding scores for each of the retrieval methodologies for each class of the database as well as the average scores for the complete McGill and ISDB databases are shown. As can be observed the EMD-PPPT and H-EMD-PPPT-R methodologies perform better in total and in most of the classes of the databases.

Table 1 Quantitative measure scores of the examined retrieval methodologies for the McGill Database

Class	Method	NN(%)	FT(%)	ST(%)	DCG(%)
Complete McGill db	EMD-PPPT	**97.6**	**74.1**	**91.1**	**93.3**
	EMD-MPEG7	93.3	69.2	88.9	90.8
	SMPEG7	91.8	65.2	78.3	89.1
	Hybrid	92.5	55.7	69.8	85.0
	H-EMD-KIM-R	94.1	70.7	82.9	90.2
	H-EMD-PPPT-R	97.3	69.9	75.8	90.5
	MPEG7	97.3	73.1	84.0	91.9
Ants	EMD-PPPT	96.7	54.9	79.7	88.4
	EMD-MPEG7	96.7	58.5	79.9	87.5
	SMPEG7	80.0	57.1	75.6	86.7
	Hybrid	**100**	**73.6**	**89.2**	**94.8**
	H-EMD-KIM-R	96.7	63.4	83.2	88.9
	H-EMD-PPPT-R	96.7	58.3	81.5	89.2
	MPEG7	90.0	62.1	75.5	87.1
Crabs	EMD-PPPT	**100**	**98.2**	**99.8**	**99.9**
	EMD-MPEG7	100	89.8	98.2	99.2
	SMPEG7	100	72.9	90.3	95.9
	Hybrid	100	55.2	71.8	88.7
	H-EMD-KIM-R	100	87.5	92.9	98.0
	H-EMD-PPPT-R	100	92.6	94.3	98.6
	MPEG7	90.0	45.9	65.5	82.2
Spectacles	EMD-PPPT	**100**	**70.3**	**99.8**	**94.0**
	EMD-MPEG7	96.0	63.7	94.3	89.2
	SMPEG7	96.0	55.8	63.7	82.7
	Hybrid	96.0	53.5	63.3	85.9
	H-EMD-KIM-R	96.0	74.0	80.0	90.5
	H-EMD-PPPT-R	96.0	73.8	80.0	91.5
	MPEG7	84.0	37.8	50.8	73.6
Hands	EMD-PPPT	**95.0**	**83.9**	**88.9**	**95.2**
	EMD-MPEG7	95.0	79.7	88.2	93.4
	SMPEG7	95.0	78.7	87.9	93.0
	Hybrid	90.0	43.4	57.6	77.8
	H-EMD-KIM-R	95.0	77.4	83.7	92.3
	H-EMD-PPPT-R	95.0	79.7	83.9	94.0
	MPEG7	60.0	30.0	41.3	63.1
Humans	EMD-PPPT	96.6	**93.5**	96.4	**98.1**
	EMD-MPEG7	96.6	86.8	**99.3**	97.4
	SMPEG7	96.6	84.5	98.0	97.3
	Hybrid	**100**	47.0	63.8	83.1
	H-EMD-KIM-R	96.6	79.6	85.2	94.3
	H-EMD-PPPT-R	96.6	82.0	84.7	94.6
	MPEG7	79.3	40.5	59.1	77.9
Octopuses	EMD-PPPT	**88.0**	**58.8**	**81.8**	**88.1**
	EMD-MPEG7	80.0	45.2	73.2	79.1
	SMPEG7	84.0	42.0	63.0	80.5
	Hybrid	56.0	29.5	45.0	68.9
	H-EMD-KIM-R	76.0	45.7	71.2	78.1
	H-EMD-PPPT-R	88.0	57.8	80.3	87.0
	MPEG7	72.0	46.8	76.2	77.8
Pliers	EMD-PPPT	**100**	**100**	**100**	**100**
	EMD-MPEG7	100	85.5	100	98.6
	SMPEG7	100	86.1	95.5	97.8
	Hybrid	100	71.6	87.9	94.6
	H-EMD-KIM-R	100	92.4	99.7	99.0
	H-EMD-PPPT-R	100	99.7	99.7	99.9
	MPEG7	95.0	65.5	77.9	89.5
Snakes	EMD-PPPT	**100**	43.2	**95.2**	**84.7**
	EMD-MPEG7	80.0	**46.2**	85.8	83.4
	SMPEG7	80.0	44.2	48.0	76.6
	Hybrid	80.0	23.7	28.7	62.4
	H-EMD-KIM-R	88.0	42.3	47.3	75.7
	H-EMD-PPPT-R	96.0	43.7	47.3	75.4
	MPEG7	76.0	36.8	40.7	69.3
Spiders	EMD-PPPT	**100**	87.2	**100**	**98.4**
	EMD-MPEG7	100	85.7	97.3	97.5
	SMPEG7	96.8	74.8	86.6	93.9
	Hybrid	100	71.5	91.0	93.7
	H-EMD-KIM-R	100	85.7	96.9	97.6
	H-EMD-PPPT-R	100	**87.3**	**99.0**	98.3
	MPEG7	90.3	37.3	61.8	77.8
Teddy-bears	EMD-PPPT	100	45.3	63.2	83.9
	EMD-MPEG7	85.0	42.6	66.3	78.8
	SMPEG7	90.0	55.8	70.8	84.6
	Hybrid	**100**	**90.3**	**98.4**	**99.1**
	H-EMD-KIM-R	90.0	54.7	87.4	85.5
	H-EMD-PPPT-R	100	52.6	87.4	89.1
	MPEG7	100	79.2	84.5	93.4

Also in the PR-curves of Figure 15 it can be observed that in some classes like in 'Dogs' and 'Cats' the PR-Curve is low this means that the Retrieval system confuses the models in these classes. This is attributed to the global alignment problem in Papadakis *et al.* [22] work, the parts fail to be consistently aligned.

Although the primary goal of our experimental work is to show the improvement in retrieval accuracy that is achieved by the proposed approach against other schemes that use a part-based representation, we have extended our experimental framework to include approaches that

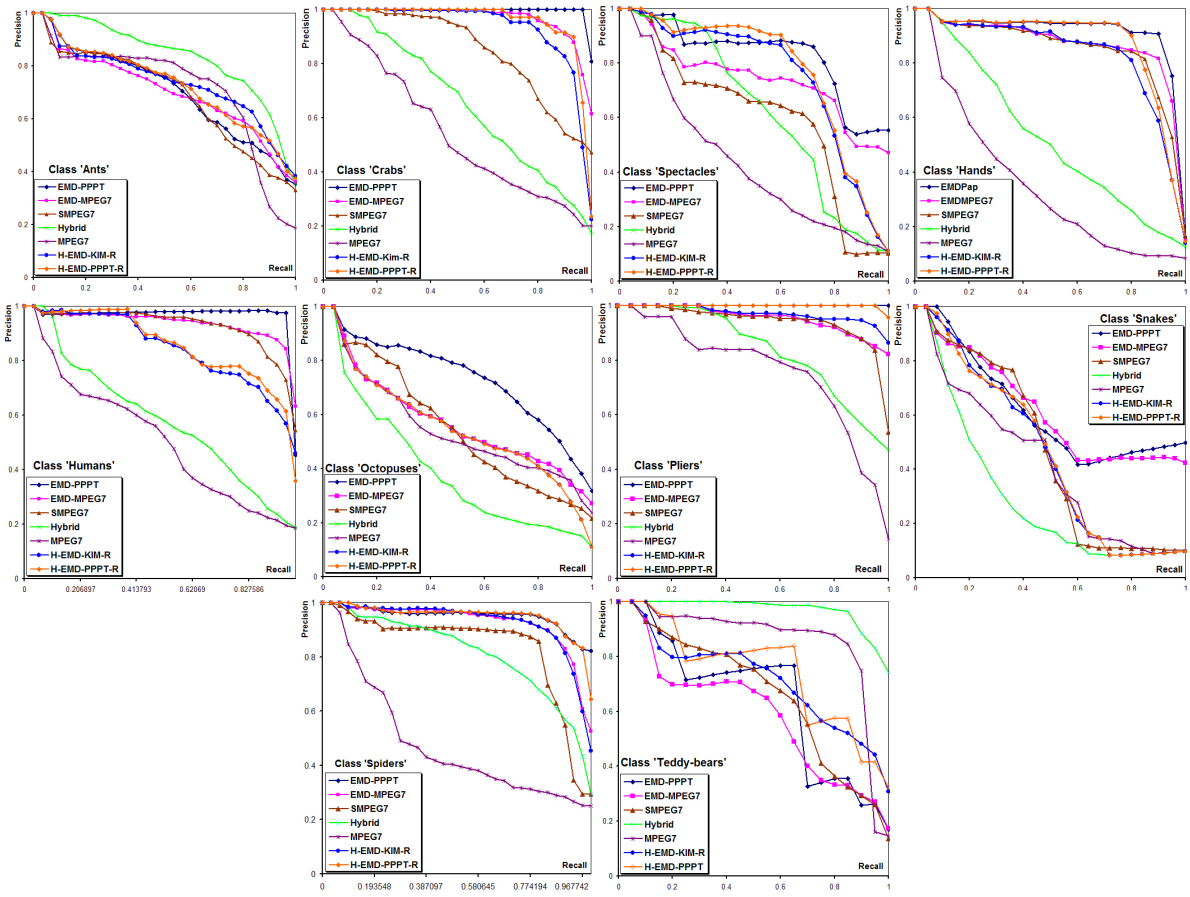


Fig. 14 Precision-Recall curves of each distinct class in the McGill database

deal with 3D articulated objects without taking into account 3D object partitioning. For this purpose, we encountered the ISDB database for which the state-of-the-art method of Gal *et al.* [11] was tested against. In Table 3, the scores of Gal *et al.* [11] retrieval methodology is presented. It can be observed that although the proposed approach has already achieved a very good performance, the scores achieved by [11] show a better performance in the complete ISDB database. To be fair in the final conclusion, it is imperative that we should also have the performance of Gal *et al.* retrieval methodology for the standard McGill database, for which, unfortunately, has not been tested yet.

To provide a further qualitative measure for the performance of the proposed methodology 'EMD-PPPT' against 'MPEG7' the produced ranking is shown in Fig. 16 for particular queries of classes like 'humans', 'octopuses' and 'hands'. It can be observed that the proposed retrieval methodology clearly outperforms the 'MPEG7' methodology.

In the case of perturbation the retrieval methodology is quite robust and the retrieval results for the query models of of Fig. 16 undergone strong Gaussian noise are shown in Fig. 17.

It should be noted that the segmentation methodology of Katz *et al.* [15] could also be used in the experimental results. Our Segmentation methodology though has many advantages over this methodology in the core extraction methodology:

- There is no need to do multidimensional scaling, which is a time consuming process, in order to extract the core. Instead only the minimum cost paths are used in order to check whether the core has expanded sufficiently. This implies far less complexity;
- We have introduced a percentage of minimum cost path traces that should be covered for the termination of core expansion. Those traces span the protrusible parts at most. Thus, the selection of a percentage of the traces provides a high confidence that the core points will cover areas of the protrusible parts or being very close to the neighboring areas in which the real boundary is situated.

5 Conclusions

In this paper a graph-based retrieval methodology is proposed. The method builds the structural description of

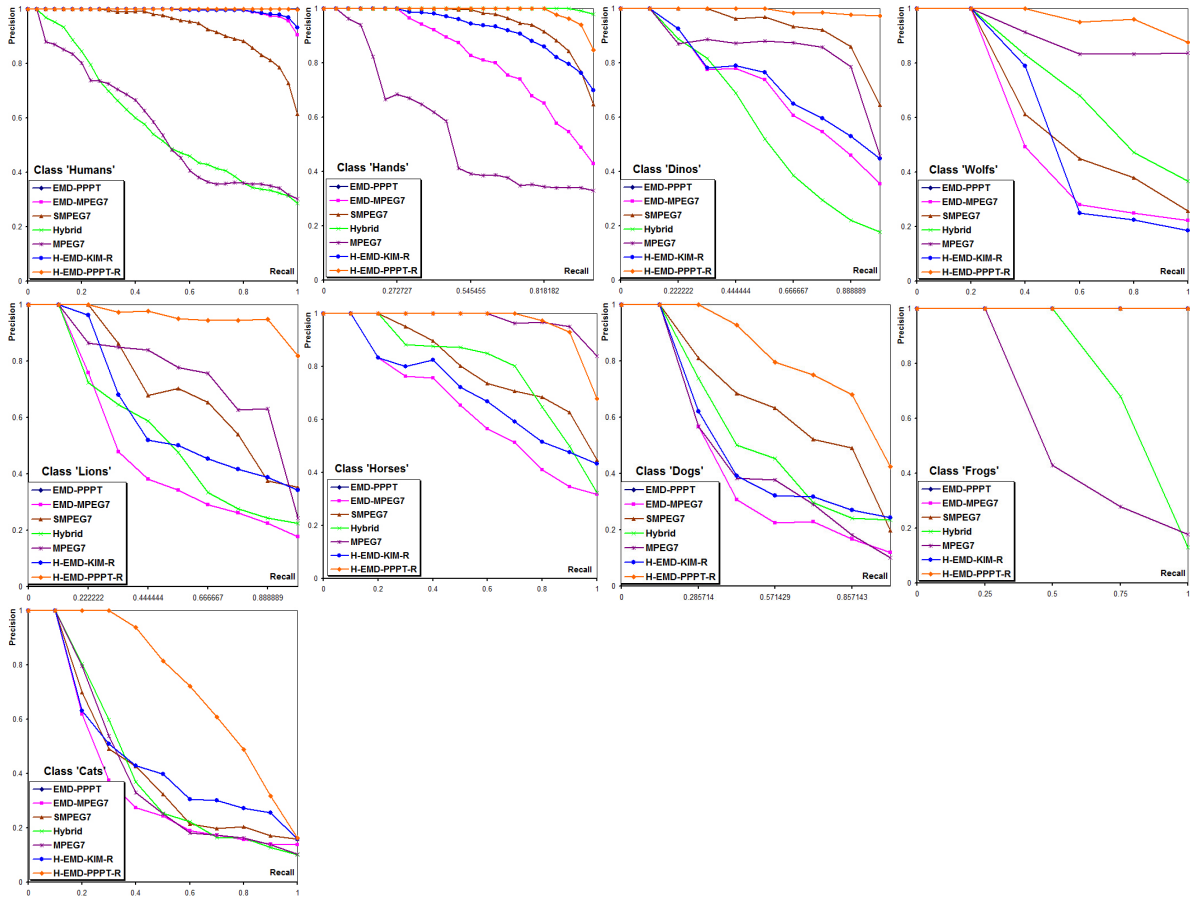


Fig. 15 Precision-Recall curves of each distinct class in the ISDB database

the object using a mesh segmentation algorithm that produces meaningful results. The produced structural description is represented by an attributed relational graph (ARG). A query retrieval is performed by matching the query's ARG with the ARGs of the database objects using an EMD-based matching approach which comprises new ground distance assignments.

The proposed methodology is very efficient in retrieving articulated objects and exhibits a significantly better performance against the compared state of the art retrieval algorithms that take into account a part-based representation in the McGill and ISDB Database of articulated objects after an extensive evaluation in both qualitative and quantitative terms.

References

1. A. Agathos, I. Pratikakis, S. Perantonis, and N. Sapidis. A protrusion-oriented 3D mesh segmentation. *Visual Computer*. DOI http://dx.doi.org/10.1007/s00371-009-0383-8.
2. A. Agathos, I. Pratikakis, S. Perantonis, N. Sapidis, and P. Azariadis. 3D mesh segmentation methodologies for CAD applications. *Computer-Aided Design and Applications*, 4(6):827–841, 2007.
3. M. Ben-Chen and Craig Gotsman. Characterizing shape using conformal factors. In *Eurographics Workshop on 3D Object Retrieval*, pages 1–8, 2008.
4. S. Biasotti, S. Marini, M. Spagnuolo, and B. Falcidieno. Sub-part correspondence by structural descriptors of 3D shapes. *Computer-Aided Design*, 38(9):1002–1019, 2006.
5. I. Biederman. Recognition-by-components: A theory of human image understanding. *Psychological Review*, 94(2):115–147, 1987.
6. Alexander M. Bronstein, Michael M. Bronstein, and Ron Kimmel. Topology-invariant similarity of nonrigid shapes. *Int. J. Comput. Vision*, 81:281301, 2009.
7. B. Bustos, D. Keim, D. Saupe, T. Schreck, and D. Vranic. Automatic selection and combination of descriptors for effective 3D similarity search. In *IEEE Sixth Int. Symp. on Multimedia Software Engineering*, pages 514–521, 2004.
8. D. Y. Chen, X. P. Tian, Y. T. Shen, and M. Ouhyoung. On visual similarity based 3D model retrieval. *Eurographics, Computer Graphics Forum*, pages 223–232, 2003.

Table 2 Quantitative measure scores of the examined retrieval methodologies for the ISDB Database

Class	Method	NN(%)	FT(%)	ST(%)	DCG(%)
Complete ISDB db	EMD-PPPT	**100**	**89.5**	**95.8**	**97.2**
	EMD-MPEG7	74.5	60.7	74.4	80.5
	SMPEG7	87.7	69.9	84.8	88.1
	Hybrid	84.9	54.1	68.5	79.9
	H-EMD-KIM-R	78.3	67.4	83.2	84.2
	H-EMD-PPPT-R	100	89.3	95.6	97.1
	MPEG7	76.4	46.7	61.0	76.6
Humans	EMD-PPPT	**100**	**100**	**100**	**100**
	EMD-MPEG7	100	97.0	99.9	99.8
	SMPEG7	100	85.1	97.4	98.3
	Hybrid	96.7	46.3	68.9	84.5
	H-EMD-KIM-R	100	99.9	100	100
	H-EMD-PPPT-R	100	100	100	100
	MPEG7	83.3	45.1	65.2	82.2
Dinos	EMD-PPPT	**100**	**97.2**	**100**	**99.6**
	EMD-MPEG7	77.8	56.9	76.4	79.2
	SMPEG7	100	81.9	94.4	96.5
	Hybrid	77.8	38.9	55.6	72.8
	H-EMD-KIM-R	77.8	58.3	81.9	80.8
	H-EMD-PPPT-R	100	97.2	100	99.6
	MPEG7	77.8	72.2	84.7	88.7
Frogs	EMD-PPPT	**100**	**100**	**100**	**100**
	EMD-MPEG7	100	100	100	100
	SMPEG7	100	100	100	100
	Hybrid	100	50.0	58.3	71.4
	H-EMD-KIM-R	100	100	100	100
	H-EMD-PPPT-R	100	100	100	100
	MPEG7	0	16.7	25.0	45.2
Lions	EMD-PPPT	**100**	**86.1**	**98.6**	**96.9**
	EMD-MPEG7	44.4	26.4	50.0	59.0
	SMPEG7	100	51.4	76.4	80.0
	Hybrid	44.4	37.5	52.8	64.5
	H-EMD-KIM-R	88.9	41.7	62.5	71.1
	H-EMD-PPPT-R	100	86.1	98.6	96.9
	MPEG7	77.8	59.7	77.8	83.3
Wolfs	EMD-PPPT	**100**	**85.0**	**100**	**95.2**
	EMD-MPEG7	20.0	10.0	35.0	44.8
	SMPEG7	20.0	25.0	55.0	55.4
	Hybrid	80.0	40.0	60.0	68.0
	H-EMD-KIM-R	20.0	10.0	50.0	48.4
	H-EMD-PPPT-R	100	85.0	100	95.2
	MPEG7	60.0	20.0	30.0	51.9
Cats	EMD-PPPT	**100**	**57.8**	**73.3**	**87.4**
	EMD-MPEG7	40.0	16.7	32.2	50.6
	SMPEG7	40.0	24.4	37.8	56.7
	Hybrid	70.0	23.3	35.6	58.0
	H-EMD-KIM-R	40.0	28.9	51.1	58.2
	H-EMD-PPPT-R	100	55.6	71.1	86.8
	MPEG7	60.0	23.3	32.2	56.8
Dogs	EMD-PPPT	**100**	**64.3**	**83.3**	**86.0**
	EMD-MPEG7	28.6	16.7	28.6	46.7
	SMPEG7	57.1	45.2	59.5	71.8
	Hybrid	42.9	28.6	47.6	59.7
	H-EMD-KIM-R	28.6	21.4	50.0	53.5
	H-EMD-PPPT-R	100	64.3	83.3	86.0
	MPEG7	42.9	19.0	35.7	49.8
Horses	EMD-PPPT	**100**	**88.9**	**96.7**	**98.6**
	EMD-MPEG7	50.0	51.1	68.9	76.3
	SMPEG7	100	66.7	90.0	88.2
	Hybrid	100	65.6	80.0	87.5
	H-EMD-KIM-R	50.0	56.7	82.2	80.4
	H-EMD-PPPT-R	100	88.9	96.7	98.6
	MPEG7	100	90.0	100	99.0
Hands	EMD-PPPT	**100**	**95.2**	**99.4**	**99.7**
	EMD-MPEG7	100	69.5	89.4	94.5
	SMPEG7	100	86.6	98.1	98.7
	Hybrid	100	98.7	100	100
	H-EMD-KIM-R	100	81.4	99.4	98.0
	H-EMD-PPPT-R	100	95.2	99.4	99.7
	MPEG7	90.9	44.6	55.6	80.0

Table 3 Quantitative measure scores of Gal *et al.* retrieval methodology for the ISDB Database

Class	Method	NN(%)	FT(%)	ST(%)	DCG(%)
Complete ISDB db	Gal	100	98.34	99.67	99.81

9. N. Cornea, M. F. Demirci, D. Silver, A. Shokoufandeh, S. Dickinson, and P. Kantor. 3D object retrieval using many-to-many matching of curve skeletons. In *Proceedings of shape modeling and applications*, pages 366–371, 2005.
10. T. Funkhouser and P. Shilane. Partial matching of 3D shapes with priority-driven search. In *Fourth Eurographics symposium on Geometry processing*, pages 131–142, 2006.
11. R. Gal, A. Shamir, and D. Cohen-Or. Pose oblivious shape signature. *IEEE Transactions of Visualization and Computer Graphics*, 13(2):261–271, 2007.
12. M. Hilaga, Y. Shinagawa, T. Komura, and T. L. Kunii. Topology matching for full automatic similarity estimation of 3d. In *ACM SIGGRAPH*, pages 203–212, 2001.
13. D. Hoffman and W. Richards. Parts of recognition. *Cognition*, 18:65–96, 1984.
14. V. Jain and H. Zhang. A spectral approach to shape-based retrieval of articulated 3D models. *Comp. Aided Des.*, 39(5):398–407, 2007.
15. S. Katz, G. Leifman, and A. Tal. Mesh segmentation using feature point and core extraction. *The Visual Computer*, 21(8-10):649–658, 2005.
16. S. Katz and A. Tal. Hierarchical mesh decomposition using fuzzy clustering and cuts. *ACM TOG*, 22(3):954–961, 2003.
17. M. Kazhdan, T. Funkhouser, and S. Rusinkiewicz. Rotation invariant spherical harmonic representation of 3D shape descriptors. In *Eurographics/ACM SIGGRAPH symposium on Geometry processing*, pages 156–164, 2003.
18. D. H. Kim, I. K. Park, I. D. Yun, and S. U. Lee. A new mpeg-7 standard: Perceptual 3D shape descriptor. In *5th Pacific Rim Conference on Multimedia*, pages 238–245, 2004.
19. H. S. Lin, H. M. Liao, and J. Lin. Visual salience-guided mesh decomposition. *IEEE Transactions On Multimedia*, 9(1):46–57, 2007.
20. S. Marini, M. Spagnuolo, and B. Falcidieno. Structural shape prototypes for the automatic classification of 3D objects. *Computer Graphics and Applications*, 27(4):28–37, 2007.
21. R. Ohbuchi, K. Osada, T. Furuya, and T. Banno. Salient local visual features for shape-based 3d model retrieval. In *IEEE Int. Conf. on Shape Modeling and Applications*, pages 93–102, 2008.
22. P. Papadakis, I. Pratikakis, S. Perantonis, and T. Theoharis. Efficient 3D shape matching and retrieval using a concrete radialized spherical projection representation. *Pattern Recognition*, 40(9):2437–2452, 2007.
23. P. Papadakis, I. Pratikakis, T. Theoharis, G. Passalis, and S. Perantonis. 3D object retrieval using an efficient and compact hybrid shape descriptor. In *Eurographics Workshop on 3D Object Retrieval*, pages 9–16, 2008.
24. G. Passalis, T. Theoharis, and I. A. Kakadiaris. Ptk: A novel depth buffer-based shape descriptor for three-dimensional object retrieval. *Visual Computer*, 23(1):5–14, 2007.
25. Y. Rubner, C. Tomasi, and L. J. Guibas. The earth mover's distance as a metric for image retrieval. *International Journal of Computer Vision*, 40(2):99–121, 2000. http://www.cs.duke.edu/ tomasi/software/emd.htm.
26. H. Sundar, D. Silver, N. Gagvani, and S. Dickinson. Skeleton based shape matching and retrieval. In *Shape modeling International*, pages 130–139, 2003.
27. A. Tal and E. Zuckerberger. Mesh retrieval by components. In *International Conference on Computer Graphics Theory and Applications*, pages 142–149, 2006.
28. T. Tung and F. Schmitt. The augmented multiresolution reeb graph approach for content-based retrieval of 3D shapes. *International Journal of Shape Modeling*, 11(1):91–120, 2005.
29. D. Vranic. Desire: a composite 3D-shape descriptor. In *IEEE International conference on Multimedia and Expo*, pages 145–156, 2005.
30. D. V. Vranic. *3D Model Retrieval*. PhD thesis, University of Leipzig, 2004.
31. McGill 3D Shape Benchmark Objects with articulating parts. http://www.cim.mcgill.ca/∼shape/benchMark/.

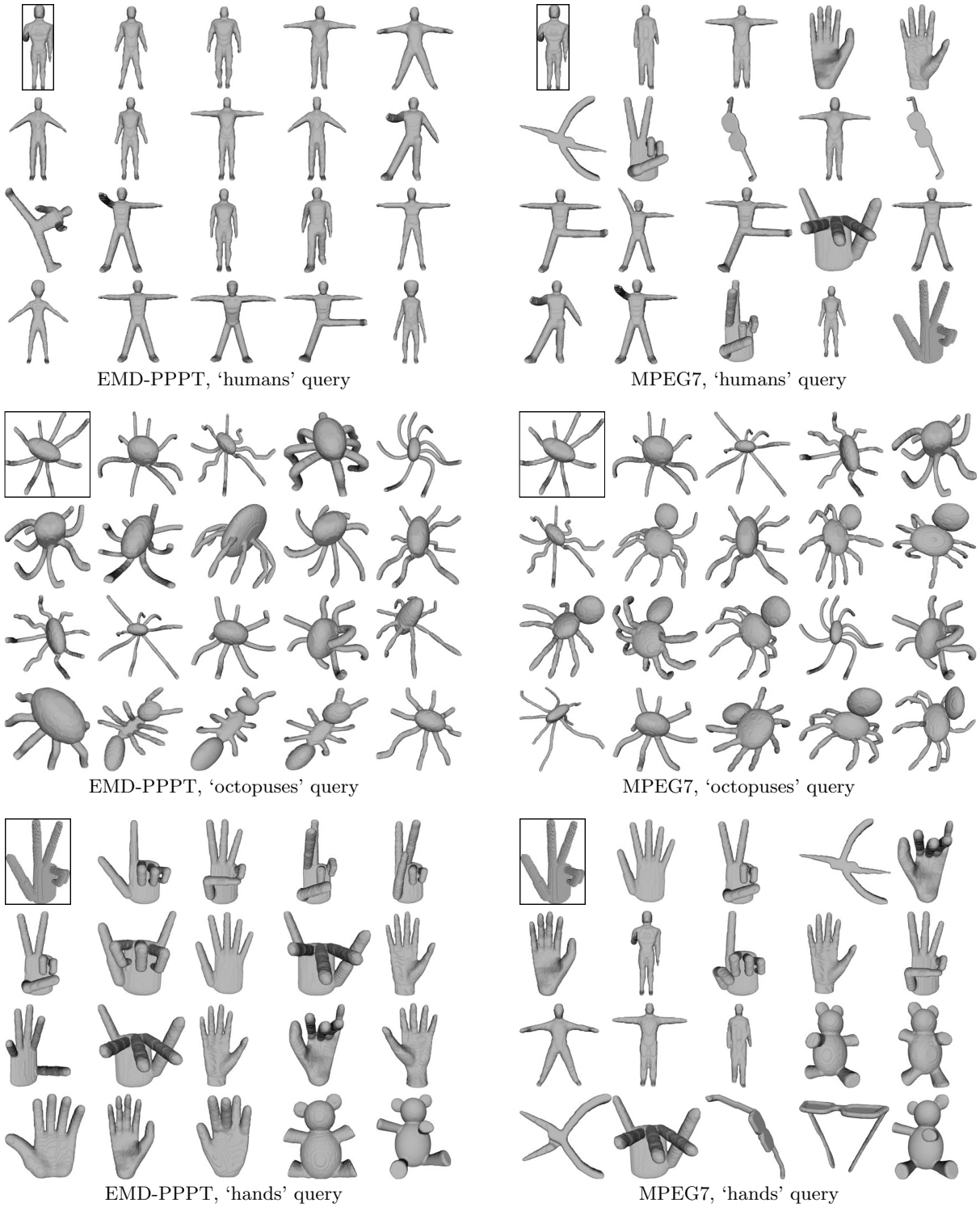
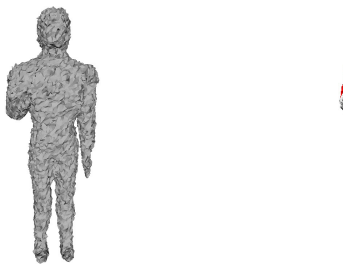
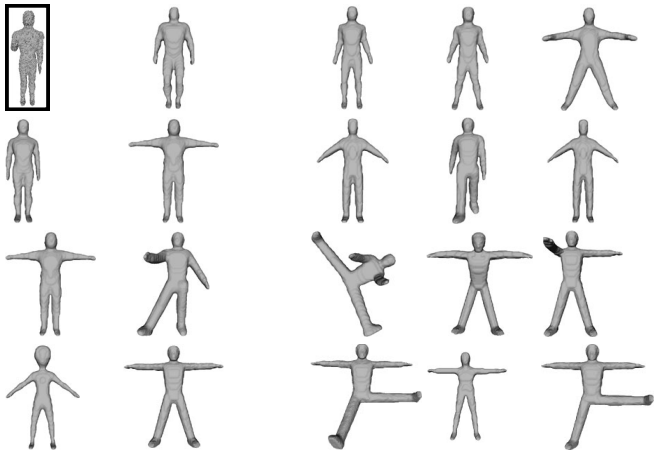


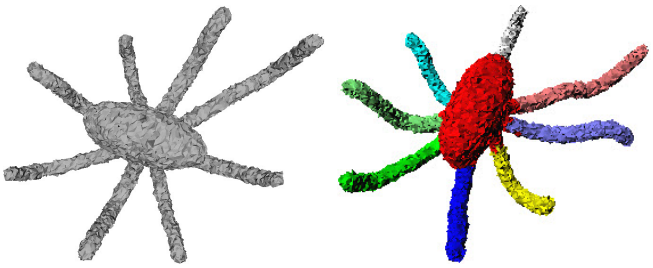
Fig. 16 Retrieval results for queries that correspond to ‘humans’, ‘octopuses’ and ‘hands’ classes, respectively, using either the ‘EMD-PPPT’ or ‘MPEG7’ 3D object retrieval methodology. The query object is shown on the top left side and the ranking order follows a top-to-bottom and left-to-right sequential arrangement.



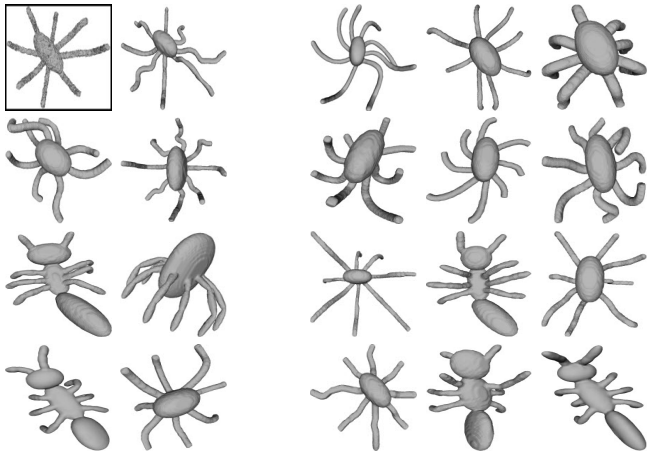
‘Human’ model under Gaussian noise and its segmentation



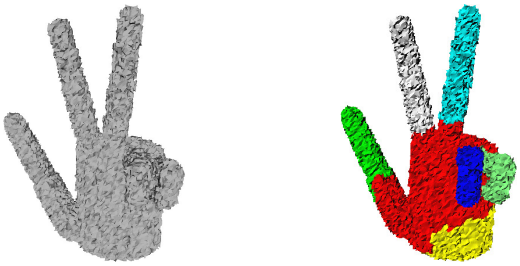
EMD-PPPT, ‘humans’ query under Gaussian noise



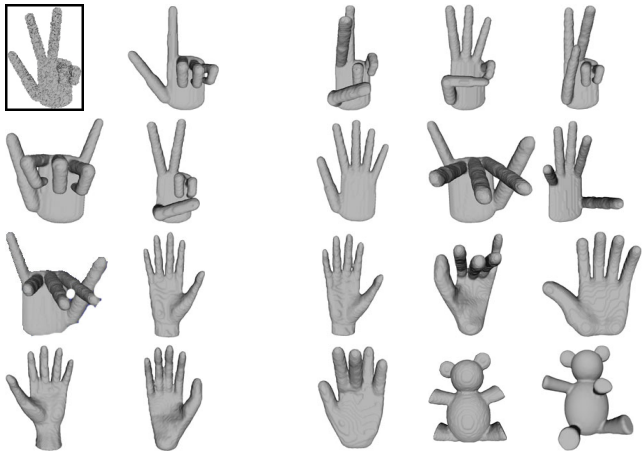
‘Octopuses’ model under Gaussian noise and its segmentation



EMD-PPPT, ‘octopuses’ query under Gaussian noise



‘Hand’ model under Gaussian noise and its segmentation



EMD-PPPT, ‘hands’ query under Gaussian noise

Fig. 17 Retrieval results for queries that correspond to ‘humans’, ‘octopuses’ and ‘hands’ classes, respectively using ‘EMD-PPPT’. On the left column the mesh under Gaussian noise and its segmentation are shown. On the right column the query object is shown on the top left side and the ranking order follows a top-to-bottom and left-to-right sequential arrangement.