Eurographics Workshop on 3D Object Retrieval (2010)
I. Pratikakis, M. Spagnuolo, T. Theoharis, and R. Veltkamp (Editors)

The Fast Reject Schema for Part-in-Whole 3D Shape Matching

Marco Attene, Simone Marini, Michela Spagnuolo and Bianca Falcidieno[1]

[1]IMATI-CNR, Italy

Abstract

This paper proposes a new framework for an efficient detection of template shapes within a target 3D model, or scene. The proposed approach distinguishes from the previous literature because the part-in-whole matching between the template and the target is obtained by extracting off-line only the shape descriptor of the template, while the description of the target is dynamically and adaptively extracted during the matching process. This novel framework, called the Fast Reject schema, exploits the incremental nature of a class of local shape descriptors to significantly reduce the part-in-whole matching time, without any expensive processing of the models for the extraction of the shape descriptors. The schema have been tested on three different descriptors and results are discussed in details. Experiments show that the gain in computational performances does not compromise the accuracy of the matching results.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Additional keywords: *3D shape, Fast partial matching, Semantic annotation.*

1. Introduction

Today, 3D acquisition technologies as well as advanced 3D design tools, make it possible to digitize our environment at several scales, starting from small objects up to large scale spaces. This enables 3D digital models to become active actors in several contexts such as industrial manufacturing, ambient surveillance and security, biology, serious gaming and simulation and many other applications. Most of the aforementioned applications need to detect relevant shape parts to perform several tasks, including the semantic annotation of large model collections: once the matching has been established between a template model and the target, the corresponding sub-parts can be annotated with the information associated to the template, hence enabling powerful semantics-based retrieval paradigms [ARSF09, GKF09].

In the literature, the location of relevant parts in a 3D target model (e.g. a 3D complex scene) is often approached through *part-in-whole* shape matching, where various regions of the *target* model are compared with a given *template* model: the parts which are sufficiently similar to the template are tagged as *relevant*. Typically, local descriptors of various regions in the target shape are computed offline, so that at query time it is sufficient to compare them with the template's descriptor. Although this approach is efficient for retrieval purposes, it might become inappropriate in some other contexts such as, for example, when the target model changes dynamically.

In this paper, we tackle the problem of detecting relevant objects in the target model or scene using an innovative framework that combines the advantages of part-in-whole matching with very promising time and quality performances. As a reference scenario, we consider a set of 3D models stored in a library of objects (*template* models) considered relevant for a specific application context. The goal is to detect the occurrences of the template models in the target model (e.g. a 3D scene). The proposed part-in-whole method combines the use of a particular class of local shape descriptors with an original matching schema that we call the *Fast Reject*. Differently from existing part-in-whole approaches, the Fast Reject schema requires only the shape descriptor of the template model to be extracted off-line, while the descriptor of the target is computed through an adaptive procedure during the matching process. The approach adopted is inspired by search strategies used in computer vi-

© The Eurographics Association 2010.

DOI: 10.2312/3DOR/3DOR10/023-030

delivered by EUROGRAPHICS DIGITAL LIBRARY
www.eg.org diglib.eg.org

sion and pattern recognition, such as the cascade detector [VJ01] or the coarse-to-fine strategy [FG01]. To our knowledge extent, similar strategies were not applied in a systematic manner to the context of 3D object matching, while the gain in performance they achieve prompts for a wider use in computer graphics.

The gain in computational performance obtained with the proposed approach is due to its capability to exploit the *layered* structure of a class of local shape descriptors. Roughly speaking, the Fast Reject schema is based an iterative algorithm that initially analyzes a *large number* of *small surface* regions. At each iteration the number of regions decreases, by discarding portions of the target model that do not resemble the template model. At the same time, the remaining regions grow in order to add new information to the matching process.

The paper is organized as follows. In section 3, the Fast Reject schema is described; in section 4, the implementation of the Fast Reject schema is described with three different local descriptors, the spherical harmonics, a coarse volumetric descriptor and a third descriptors based on curvature analysis; section 5 describes our experiments and results; conclusions and suggestions for future work end the paper.

2. Related Work

In the last decade several specific approaches to partial matching have been proposed. Methods for part-in-whole matching frequently exploit structural decompositions of the object shape. Graph-matching techniques for instance build first a graph representation of relevant object's parts and their adjacency relationships, then, by matching the graphs, the correspondence between nodes, that is, object parts, is derived. The information associated to the structural descriptor makes it easier to estimate also the global similarity based of the objects. Relevant examples are [BSRS03, CDS*05, BMSF06, TVD09]. In [SYYS05, SYS05] the 3D model is initially decomposed into sub-components and then shape descriptors for these shapes are computed using a rotation invariant shape descriptors encoded as histograms. Part-in-whole detection is performed by comparing histograms among sub-components. In [FMA*09], the authors avoid to use an arbitrary shape decomposition by using a collection-aware decomposition approach combined with a shape thesaurus, where inverted indexes are used to describe and retrieve 3D models. Similarly, in [BBCK09], the authors define the significance of a part of a shape by its discriminative power with respect do a given shape database and use the term frequency-inverse document frequency for partial matching.

Another category of approaches uses local shape descriptors. One of the most important methods, which inspired many others, uses the spin-images to provide a set of local shape descriptors [JH97]. This method samples the object surface into a set of oriented points (3D points with surface normals) and associates to each sampled point a 2D description of the surface around it: the spin-image. A similarity measure between 2D images is used to evaluate the similarity between two spin-images and thus between two oriented points of the compared objects. In this way a point-to-point correspondence between the two objects is provided. In [RCSM03] the similarity measure defined among spin-images is used to group oriented points into patches. This latter approach allows the correspondence between patches instead of points.

More recently, partial matching is solved by comparing the so-called *salient features* of two objects [GCO06]. First, a set of local descriptors on the object surface is defined, each associated to a surface point and defined using a quadric patch approximating the surface around the point. Salient features are identified by grouping the local descriptors according to curvature variance and intensity. Finally, each salient features is associated to an indexing vector and inserted into a geometric hash table, as an access point to object parts. Also in [SF07] important regions of the object surface are used to perform partial matching. A region of an object is considered important, or distinctive, if it is useful to discriminate the object with respect to other objects of a given data-set. Distinctive regions are identified by randomly sampling the points on the object surface and by associating to each sampled point the spherical harmonics descriptor [Kaz04] computed at four different scales. Each descriptor is compared with all the others descriptors at the same scale in a data-set of objects grouped into object classes, and the distinctiveness of each point is obtained from the discounted cumulative gain of the ranked list obtained from the comparison by measuring how often objects of the same class appear near the front of the list. These approaches are particularly appropriate for reatrieval purposes because, after an offline preprocessing, the search space at query time is significantly reduced.

To summarize, the extraction of structural descriptors is generally demanding in computational terms and as such is considered to be run off-line with respect to the matching phase. For this reason, they are not suited to applications where the generation of the shape description and the matching process have to be performed rapidly, such as for example in realtime contexts. In [GCO06, SYYS05, SYS05], object sub-parts relevant for the matching are selected depending on geometric properties of the given object. On the contrary, the method proposed in [SF07, FMA*09, BBCK09, BBBK09] can be used for partial matching purposes by exploiting the semantics-oriented selection of distinctive features. Unfortunately the comparison schema proposed by the authors requires a significant amount of temporal resources. Finally, the spin-image methodology is not based on the multi-scale local description of the shape, thus the computational cost for the extraction of the local descriptors and the expensive matching schema do not satisfy possible high-

© The Eurographics Association 2010.

performance requirements. Compared to the discussed previous work, the Fast Reject schema presented in this paper fully exploits the performance of a class of local descriptors, formalized in the paper as layered or *onion* descriptors, and achieves for this class of methods a significant gain in computational performance without compromising the accuracy of the matching results.

3. The Fast Reject Schema

The requirements we want to satisfy within the proposed matching schema, are both the efficient processing of the input 3D scene and the effective part-in-whole association. The capability of the Fast Reject schema to fulfil these requirements relies on the original use of a specific category of local shape descriptors that we call *onion descriptors*.

3.1. Overview

With the aim to quickly match a template shape with one or more parts of a scene, the Fast Reject schema works as follows. At the beginning of the process, only a small region of the template is searched within the scene by looking for matching portions of its descriptor; due to the reduced size of the query, such a search is extremely fast. Afterwards, a slightly bigger piece of the template is searched. This time, however, the search space is not the whole scene, but only a subset for which the first run produced a good match. Then, an even bigger region of the template is searched among the good matches of the second run, and so on, until the whole template is considered and possibly matched. The clear assumption behind this approach is the following: if a region of the template cannot be found in a part of the scene, then we can exclude the possibility to find the whole template in the same part. Notice that though this assumption is obvious for exact matches, it is worth discussing the case of approximate matches involving threshold distances. To do this, it is necessary to introduce some terminology.

3.2. Terminology

Let $\mathcal{K}$ be the space of all the instances of a given shape descriptor. We refer to a **multi-fielded** local shape descriptor $S_M(\mathbf{c},r) = (s_M^1,\dots,s_M^N)$ as to a vector of $\mathcal{K}^N$ encoding the surface M of an object around a specific point $\mathbf{c}$ up to a distance r expressed using a specific metric (not necessarily Euclidean, e.g. geodesic). Each descriptor s_M^i is called a *field*, and can be a real number, a vector, or even a more structured descriptor such as a graph, depending on the nature of $\mathcal{K}$.

A **onion descriptor** is a vector of multi-fielded descriptors of non-decreasing dimension defined as in Equation 1.

$$\begin{gathered} O_M(\mathbf{c},r_1,\dots,r_k) = (S_M(\mathbf{c},r_1),\dots,S_M(\mathbf{c},r_k)), \\ i<j \implies dim(S_M(\mathbf{c},r_i)) \le dim(S_M(\mathbf{c},r_j)), \\ i<j \implies r_i<r_j \end{gathered} \tag{1}$$

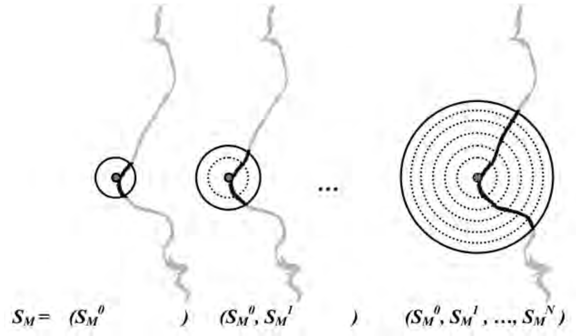

Figure 1: *An example of incrementally-defined onion descriptor of a nose. From left to right: the first circle encloses a small region around the nosetip; this region defines the first layer S_M^0. The second circle encloses a slightly larger region defining both S_M^0 and S_M^1. The whole nose is enclosed by the largest circle on the right, which defines the complete onion descriptor of the shape.*

Each multi-fielded descriptor constituting a onion is called a *layer*. Note that the various layers of the sequence are all referred to the same center point $\mathbf{c}$, but consider a varying size of the neighborhood. Specifically, while to compute $S_M(\mathbf{c},r_k)$, with $k>1$, it is necessary to analyze M up to a distance r_k, to compute $S_M(\mathbf{c},r_1)$ it is sufficient to deal with a smaller neighborhood of $\mathbf{c}$. If $S_M(\mathbf{c},r_j) \mapsto S_M(\mathbf{c},r_i)$ for each $j>i$ the onion descriptor is said to be *incrementally defined* (see Figure 1); herewith we use the notation $v_2 \mapsto v_1$ to indicate that a vector $v_1 \in \mathcal{K}^n$ is made of the first n coordinates of another vector $v_2 \in \mathcal{K}^m$, with $n \le m$.

Let $O_M = (S_M^1,\dots,S_M^N)$ and $O_L = (S_L^1,\dots,S_L^N)$ be two onion descriptors of the same dimension, and let their layers be pairwise comparable using a proper distance function $d : \mathcal{K}^i \times \mathcal{K}^i \to \mathbb{R}^+ \cup \{0\}$. If $d(S_M^i,S_L^i)$ grows monotonically as i grows, then we say that the onion descriptor is *monotonic* with respect to d.

3.3. Description of the framework

The Fast Reject schema is based on the computation of monotonic onion descriptors for the part-in-whole matching. To start with, a onion descriptor $O_T = \{S_T(\mathbf{c}_T,r_i), i = 1,\dots,k\}$ is computed for the template model, and the input 3D scene is discretized into a set D_I of *scene points*.

Initially, all the points of the scene are candidate to be good matches. In the first iteration, for each scene point only the first layer of its onion descriptor is computed, and its distance from the first layer of O_T is calculated. If such a distance exceeds a given threshold, the scene point is excluded from the search space or, in other words, it is *rejected* from the set of potential good matches. In the second iteration, for each non-rejected point the second layer is com-

© The Eurographics Association 2010.

Listing 1: *The Fast Reject schema for the part-in-whole matching*

```
input:
  O_T = (S_T(c_T, r_1), ..., S_T(c_T, r_k))
  ε = user threshold
  I = input scene

initialization:
      D_I ← {v_1, ..., v_M}

iteration:
  for i from 0 to k {
      for each v ∈ D_I {
          S_I(v, r_i) ← computeLayer(I, v, i)
          if d(S_T(c_T, r_i), S_I(v, r_i)) ≥ ε
              D_I ← D_I \ {v}
      }
  }

result:
  Matched ← D_I
```

puted and compared with the second layer of O_T. Also in this case, an excessive distance causes the rejection of the point from the search space. The process stops when the last layer is computed for all the remaining non-rejected points. Among these remaining points, the good matches are those whose last layer has a distance from the last layer of O_T smaller than the threshold. This process is summarized by the pseudo-code shown in the Listing 1.

Having assumed that the distance between successive pairs of layers grows monotonically, we can avoid to perform further computations for the rejected points because we are guaranteed that the eventual distance of their descriptors from O_T exceeds the dissimilarity threshold. Note that, in order to be efficient, the Fast Reject schema does not require onion descriptors to be incrementally defined. Nevertheless, in case of incrementally-defined descriptors the schema has a further advantage because the computation of each layer can exploit the previous layers as they are, without the need to recompute them.

The parameters to be investigated in order to implement the framework are a few, and include the specific onion descriptor along with its order, the number of sample points to discretize the scene (if not already part of the input, eg. a point cloud), and the threshold distance to be used. To define the center $\mathbf{c}_T$ and the radii r_i of the template's onion descriptor, the only constraint is that the largest radius guarantees the enclosure of the whole template. Provided this, $\mathbf{c}_T$ can be any point of the template's surface.

Several examples exist of multi-fielded descriptors that can constitute onion descriptors, and further can be implemented to take advantage of the Fast Reject schema. Among existing descriptors, we cite the Shape Contexts [BMP02], the Spherical Harmonic local representation [SF07], the *Tailor* characterization introduced in [MPS*03], the multi-scale surface characteristics proposed in [HFG*06] and the vectors of geometric moments used in [ETA02]. In the following sections, we provide two examples of implementation of the Fast Reject schema: one is based on the sequential use of two different local descriptors, detailed in Sections 4.1 and 4.2; the other one is based on a novel curvature multi-scale estimation (Section 4.3).

4. Implementation of the Fast Reject schema

In order to validate the performances and the accuracy of the framework, we have implemented the Fast Reject schema based on three different shape descriptors fulfilling the requirements discussed in section 3: the Spherical Harmonics (SHs), a coarse volumetric descriptor, and a novel surface descriptor based on curvature analysis. While the Spherical Harmonics provide a consolidated tool for the description of rigid shapes, the volumetric and curvature analysis descriptors allow for a suitable representation of non-rigid shapes. It is worth noticing that, despite their simplicity with respect to the SHs, the volumetric and curvature analysis descriptors provide good performances and accurate matching results.

4.1. Implementation based on the SH representation

The Spherical Harmonics (SH, for short) provide a geometrical description of a 3D object by characterising its surface distribution around a given point [KFR03, Kaz04]. If the barycentre of the object is considered for the generation of the SH description and the whole surface of the object is analysed, the SH representation behaves as a global descriptor and provides information about the overall shape of the object, as proposed in [FMK*03]. The SH representation can also be defined on an arbitrary point of the object surface and only a small region surrounding this point can be analysed. In this case the SH representation behaves as a local descriptor as shown in [SF07].

The idea behind the SH representation is to decompose a 3D model into a collection of functions defined on concentric spheres, where for each function a spherical harmonics decomposition is used to produce a 1D descriptor. Combining the 1D descriptors, by analyzing spheres at different radii, a 2D descriptor is obtained. Since the r-th 1D descriptor can be computed independently of the previous $r-1$ 1D descriptors, this approach benefits of all the advantages provided by the Fast Reject schema as it can constitute an incrementally-defined onion descriptor. In particular, for each point $\mathbf{c}$ of the surface, a multi-fielded descriptor can be defined by instantiating each field with the corresponding 1D

© The Eurographics Association 2010.

descriptor $sh^i_M(\mathbf{c}), 1 \leq i \leq N$ associated to the ith radius:

$$S_M(\mathbf{c},r) = (sh^1_M(\mathbf{c}),\ldots,sh^N_M(\mathbf{c})), \tag{2}$$

Thus, when implemented using the SH representation, the first layer of the onion descriptor is made of one vector $(sh^1_M(\mathbf{c}))$, the second layer is made of two vectors $(sh^1_M(\mathbf{c}), sh^2_M(\mathbf{c}))$, and so on, up to the N^{th} layer made of N vectors $(sh^1_M(\mathbf{c}),\ldots,sh^N_M(\mathbf{c}))$. Clearly, such an onion descriptor is monotonic with respect to the usual L^2 metric.

As in [KFR03], our implementation is based on a raster representation of the object. If the input is not in raster form, it must be converted to a binary 3D image through proper voxelization algorithms. The efficiency and accuracy of the local descriptor shown in Equation 2, depends on the number, N, of considered raddii. Moreover efficiency and accuracy of each component $sh^i_M(\mathbf{c})$ depends on two parameters: the function sampling frequency, represented by a bandwidth B, and the degree L selected for the computation of the 1D descriptor (see [KFR03] for details). The matching experiments shown in section 5 have been obtained by setting $N = 6$, $B = 32$ and $L = 16$.

4.2. Implementation based on volumetric similarity

The voxel-based raster representation of the object can be exploited to define a coarse descriptor that improves both efficiency and accuracy of the matching process done using more sophisticated descriptors, such as the SH, by benefiting again of the Fast Reject schema. In particular, the volume $\mathcal{W}$ of the ball $\mathcal{B}$ centred in v with radius r, and the volume $\mathcal{V}$ of the intersection of $\mathcal{B}$ with the solid bounded by M are computed to provide a coarse description of the region surrounding v. The ratio between $\mathcal{V}$ and $\mathcal{W}$ represents the percentage of object mass surrounding the voxel v and is used to define a multi-fielded descriptor for the coarse filter. Thus, each field of such a descriptor is a scalar defined as in Equation 3:

$$CF^r_M(v) = \frac{\mathcal{V}}{\mathcal{W}}, \tag{3}$$

In this case, by considering a sequence of non decreasing radii $< r_1,...,r_N >$, we derive a onion descriptor whose first layer is $(CF^{r_1}_M(v))$, the second layer is $(CF^{r_1}_M(v), CF^{r_2}_M(v))$, and so on. After having filled the interior voxels starting from the rasterized version of M in G, the volumes $\mathcal{W}$ and $\mathcal{V}$ can be easily computed by counting voxels. The Fast reject schema rejects the voxel v if and only if:

$$\sum_{i=1}^{N} |CF^{r_i}_T(\mathbf{c}_T) - CF^{r_i}_G(v)| > \varepsilon', \tag{4}$$

where T is the template to be searched. Clearly, if a partial sum of the first $k < N$ positive terms already exceeds the threshold, then there is no need to further compute the remaining terms.

Such a pre-filtering task rejects a significant amount of non-relevant voxels while requiring significantly less time with respect to the SH-based filter described in Section 4.1. Thus, it can be used to sensibly reduce the search space prior to initiating the Fast Reject using more complex descriptors.

4.3. Implementation based on curvature analysis

The SH representation provides a good tool to compare shapes when they are assumed to be rigid. However, there are cases where a certain degree of flexibility is allowed (eg. articulated shapes or human faces with variable expression), and thus a proper descriptor must be used. Herewith we propose a onion descriptor based on a stratified calculation of the Gaussian curvature around a point. Namely, we compute the Gaussian curvature using a neighborhood of the point under analysis as described in [ACSD*03], where the curvature tensor $\mathcal{T}$ is estimated as follows:

$$\mathcal{T}(\mathbf{v}) = \frac{1}{|B|} \sum_{e:e\cap B \neq \emptyset} \beta(e)|e\cap B|\overline{e}\overline{e}^T \tag{5}$$

where $\mathbf{v}$ is an arbitrary vertex on the mesh, $|B|$ is the surface area around $\mathbf{v}$ over which the tensor is estimated, $\beta(e)$ is the signed angle between the normals to the two oriented triangles incident to edge e (positive if convex, negative if concave), $|e \cap B|$ is the length of $e \cap B$ (always between 0 and $|e|$), and $\overline{e}$ is a unit vector in the same direction as e.

In our framework, the curvature tensor is estimeted by using information collected at different *distance ranges*. With reference to equation 5, for each range i, B is the set of points of the surface whose distance from $\mathbf{v}$ is less than a given r_i. Note that this definition may lead to disconnected surface areas; in this case we consider only the connected component containing v. Within each range, the Gaussian curvature is estimated as the product of the two eigenvalues of $\mathcal{T}$ having maximum magnitude, and the sequence of so-computed Gaussian curvatures corresponding to the ranges $r_1, r_2, ..., r_n$, with $r_i < r_{i+1}$ for each i, forms an incrementally defined onion descriptor. In essence, this descriptor represents the shape around a point as a sequence of Gaussian curvature estimates taking into account different neighborhood sizes. Being incrementally defined, the whole onion descriptor can be compactly encoded through its highest-order multi-fielded descriptor, which is a real-valued vector $S_M(v, r_n) =< g_1, ..., g_n >$. The distance between two such descriptors is defined as the sum of squared distances between pairwise corresponding values. Namely, $d(< g_1,...,g_n >,< h_1,...,h_n >) = \sum(g_i - h_i)^2$.

Such a descriptor is suitable for our Fast Reject framework, it works directly on polygon meshes without conversion, and experiments (see Section5) show its effectiveness for nearly-isometry-invariant part-in-whole matching.

5. Experiments and results

We have experimented our framework using (1) the Spherical Harmonics descriptor (SHD) preceeded by a pre-filtering

© The Eurographics Association 2010.

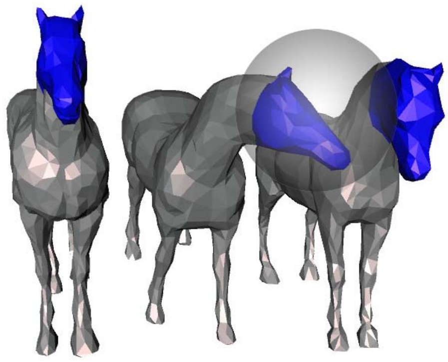

Figure 2: *An example showing an interactive application of the Fast Reject. After having selected one of the features (the head of the middle horse), the system automatically extends the selection to all the features which are sufficiently similar to the initial selected region.*

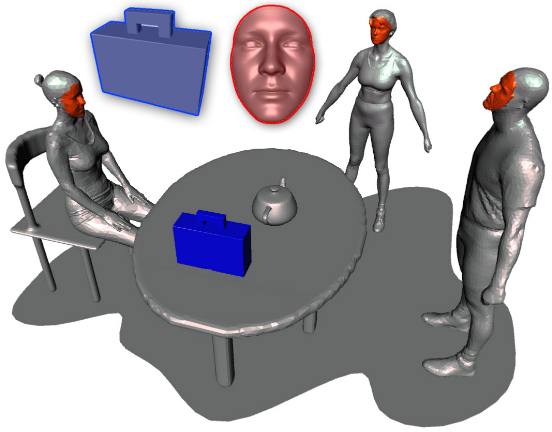

Figure 3: *Two different templates have been matched with corresponding portions of the scene using the pre-filtered SHD. The template face model is courtesy of Volker Blanz.*

based on the volumetric similarity discussed in Section 4.2, and (2) the Gaussian-curvature descriptor (GCD) introduced in Section 4.3. For the sake of experimentation, we have synthesized some scenes by composing several polygon meshes (see Figure 3), and for the SHD have converted them to voxel-based representations using a variant of [DCB*04]. After having computed the descriptor for each (voxelized) template model to be searched within the scene, we have run the Fast Reject schema and measured both the running time and the success rate; the latter has been assessed in terms of number of good matches without false positives. In a subset of the experiments, the template shape has been provided as a separate model, whereas in another subset it has been defined by selecting a piece of the scene through a semi-trasparent sphere (see Figure 2) using the ReMESH software [AF06]. In all of our experiments of the SHD the following parameters have been used to tune the algorithms. The polygon mesh scene is converted into a volumetric grid made of 512^3 voxels; the difference between the volumetric portion of the template and the one of the point considered is bounded by $\varepsilon' = 0.12$ (see Equation 4); the overall distance between the spherical harmonics descriptors is bounded by 0.5. The threshold distance ε used for experiments of the GCD was set as a function of the template size r, specifically $\varepsilon = 0.5/\pi r^2$. When the template T was provided as a separate model, both for the SHD and for the GCD, the center $\mathbf{c}_T$ was chosen as the point of the surface of T which is closest to the center of its bounding sphere, and the maxiumum radius was set as the distance of the point of T which is farthest from $\mathbf{c}_T$. When the template was provided through interactive selection, the parameters of the semi-transparent sphere defining the selection were used. The number of radii used to sample the descriptor was 6. The SHD and the GCD were tested on the same set of models and produced analogous results, with the exception of the Buddah example shown in Figure 4 for which the GCD was not discriminative enough. In contrast, the GCD was able to correctly match all the five faces of Figure 5 without false positives, while in this case the SHD was not discriminative enough.

The experiments have been run on a consumer PC equipped with an Intel Core 2 processor, 2Gb RAM and running Microsoft Windows Vista. The results confirmed that the Fast Reject framework can be used to find template shapes with an accuracy comparable with state-of-the-art approaches, but substantially faster. Such an assessment is based on both quantitative and qualitative measurements, summarized respectively in Table 1 and Figures 5 and 3. In Table 1 we did not include the time taken by [DCB*04] to perform the scan conversion because it is negligible when compared to the other processes. For the sake of comparison with one of the most significant approaches recently published, we provide an example of self-similarity search on the Stanford Buddah model (see Figure 4); in [GCO06] the localization of the four flowers needed 36 seconds, plus 19 minutes to pre-process the model; in contrast, our Fast Reject approach using the pre-filtered SHD needed less than 3 seconds to perform all the operations, including the voxelization, the construction of the template's descriptor and the actual matching.

6. Conclusions and future work

In this paper we have introduced a novel framework which exploits the layered nature of some shape descriptors to quickly locate template objects within a scene. Though the framework is generic, we have implemented it using the Spherical Harmonics descriptors on a voxelized version of the scene, and using a novel descriptor based on Gaussian curvature multi-scale estimates. Our experiments have

© The Eurographics Association 2010.

Table 1: *For each model, this table reports the number of mesh vertices, the initial candidate voxels constituting the* skin *of the solid, the time consumed by the prefilter and by the subsequent SH filter, and the time taken by the GCD computed around the mesh vertices (the latter including both the tensor estimation and the eigenvalue computation). The SH filter could correctly match all the instances of the template in all the scenes but faces(1). In faces(1) the SH filter was run with the default parameters, and the face of the middle* doll *was not matched. Conversely, in faces(2) the threshold for the SH filter was increased up to* 0.8 *in order to capture the doll's face as well, but in this case its breasts were captured too. While capturing all the instances with its default parameters, the GCD detected only one false positive on the Buddah model.*

Model Name	**Fig.**	**Num. Vertices**	**Skin Voxels**	**Prefilter Secs.**	**SH filter Secs.**	**GCD Secs.**
suitcase	3	93618	266501	0.73	2.02	7.92
face	3	93618	266501	0.69	1.99	6.01
hands	5	42682	104576	0.23	1.42	2.83
horses	2	2999	640910	0.98	2.34	0.32
buddah	4	50789	318631	0.94	1.82	3.11
faces(1)	5	42682	104576	0.31	1.47	2.87
faces(2)	5	42682	104576	0.31	1.98	2.87

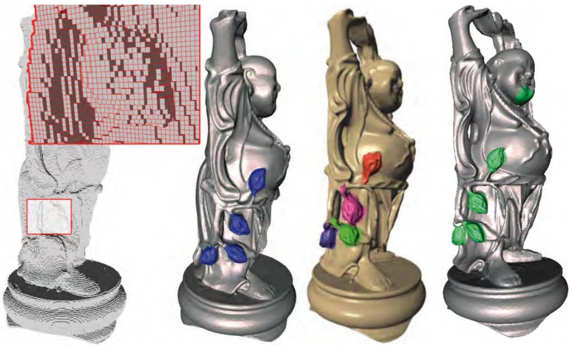

Figure 4: *An example showing our approach running on a complex model. A flower was selected on the voxelized Stanford Buddah (left), and our algorithm was able to automatically select all the other flowers (middle-left) using the pre-filtered SHD. The self-similarity result of Gal and Cohen-Or's algorithm [GCO06] is shown for comparison (middle-right). By replacing the SHD with the GCD filter, our algorithm captures a false positive (right).*

shown that the algorithm proposed is significantly faster than state-of-the-art approaches, while maintaining a comparable quality of the results. Differently from existing algorithms, the Fast Reject schema does not require any costly preprocessing of the scene, which makes it suitable for use in applications where the scene changes dynamically.

Although we have not run experiments in this sense, we argue that the Fast Reject schema on voxel-based input can be easily parallelized in order to obtain better performances. In our future research, we plan to investigate these aspects and will evaluate the impact of a parallel implementation on a prototypal realtime system. A further challenging direction for future investigation is the extension of the approach to the case of scale-invariant template matching.

Acknowledgements This work has been partially supported by the EU FP7 project FOCUS K3D. Special thanks are given to the members of the *Shape Modelling Group* at IMATI-CNR for helpful discussions.

References

[ACSD*03] ALLIEZ P., COHEN-STEINER D., DEVILLERS O., LEVY B., DESBRUN M.: Anisotropic polygonal remeshing. *ACM Trans. on Graphics* (2003), 485–493.

[AF06] ATTENE M., FALCIDIENO B.: Remesh: An interactive environment to edit and repair triangle meshes. In *Shape Modeling and Applications* (2006), pp. 271–276.

[ARSF09] ATTENE M., ROBBIANO F., SPAGNUOLO M., FALCIDIENO B.: Characterization of 3d shape parts for semantic annotation. *Computer-Aided Design 41*, 10 (2009), 756–763.

[BBBK09] BRONSTEIN A. M., BRONSTEIN M. M., BRUCKSTEIN A. M., KIMMEL R.: Partial similarity of objects, or how to compare a centaur to a horse. *Int. J. Comput. Vision 84*, 2 (2009), 163–183.

[BBCK09] BRONSTEIN A. M., BRONSTEIN M. M., CARMON Y., KIMMEL R.: Partial similarity of shapes using a statistical significance measure. *IPSJ Trans. Computer Vision and Application 1* (2009), 105–114.

[BMP02] BELONGIE S., MALIK J., PUZICHA J.: Shape matching and object recognition using shape contexts. *IEEE Trans. on Pattern Analysis and Machine Intelligence* (2002), 509–522.

[BMSF06] BIASOTTI S., MARINI S., SPAGNUOLO M., FALCIDIENO B.: Sub-part correspondence by structural descriptors of 3d shapes. *Computer-Aided Design 38*, 9 (2006), 1002–1019.

[BSRS03] BESPALOV D., SHOKOUFANDEH A., REGLI W. C., SUN W.: Scale-space representation of 3d models and topological matching. In *Proceedings of the* 8^{th} *ACM symposium on Solid Modeling and Applications* (New York, NY, USA, June 2003), ACM Press, pp. 208–215.

[CDS*05] CORNEA N. D., DEMIRCI M. F., SILVER D., SHOKOUFANDEH A., DICKINSON S. J., KANTOR P. B.: 3d object

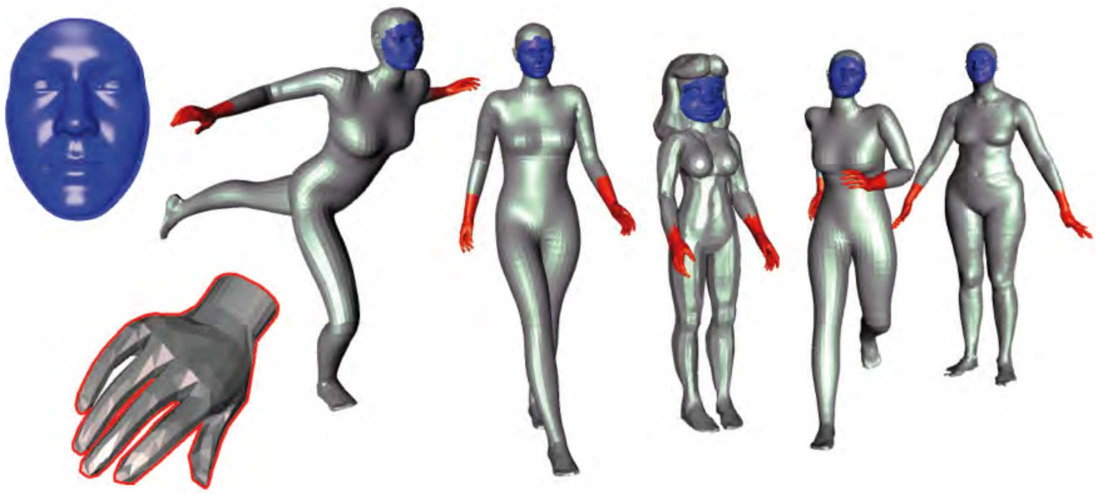

Figure 5: *The hand's template shown in the bottom-left of this picture has been successfully matched with all the ten hands present in the scene without false positives using the pre-filtered SHD. The head template was matched without false positives using the GCD.*

retrieval using many-to-many matching of curve skeletons. In *SMI'05 Proceedings* (2005), IEEE, pp. 368–373.

[DCB*04] DONG Z., CHEN W., BAO H., ZHANG H., PENG Q.: Real-time voxelization for complex polygonal models. In *PG '04: Proceedings of the Computer Graphics and Applications, 12th Pacific Conference* (Washington, DC, USA, 2004), IEEE Computer Society, pp. 43–50.

[ETA02] ELAD M., TAL A., AR S.: Content based retrieval of vrml objects: an iterative and interactive approach. In *Proceedings of the sixth Eurographics workshop on Multimedia 2001* (New York, NY, USA, 2002), Springer-Verlag New York, Inc., pp. 107–118.

[FG01] FLEURET F., GEMAN D.: Coarse-to-fine face detection. *IJCV* (2001).

[FMA*09] FERREIRA A., MARINI S., ATTENE M., FONSECA M., SPAGNUOLO M., JORGE J., FALCIDIENO B.: Thesaurus-based 3d object retrieval with part-in-whole matching. *International Journal of Computer Vision* (2009).

[FMK*03] FUNKHOUSER T., MIN P., KAZHDAN M., CHEN J., HALDERMAN A., DOBKIN D., JACOBS D.: A search engine for 3d models. *ACM Trans. on Graphics 22*, 1 (2003), 83–101.

[GCO06] GAL R., COHEN-OR D.: Salient geometric features for partial shape matching and similarity. *ACM Transactions on Graphics 25*, 1 (2006), 130–150.

[GKF09] GOLOVINSKIY A., KIM V., FUNKHOUSER T.: Shape-based recognition of 3d point clouds in urban environments. In *International Conference on Computer Vision (ICCV)* (September 2009).

[HFG*06] HUANG Q.-X., FLÖRY S., GELFAND N., HOFER M., POTTMANN H.: Reassembling fractured objects by geometric matching. *ACM Trans. Graph. 25*, 3 (2006), 569–578.

[JH97] JOHNSON A. E., HEBERT M.: Recognizing objects by matching oriented points. In *CVPR '97: Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)* (1997), IEEE Computer Society, p. 684.

[Kaz04] KAZHDAN M.: *Shape Representation and Algorithms for 3D Model Retrieval*. PhD thesis, Princeton University, 2004.

[KFR03] KAZHDAN M., FUNKHOUSER T., RUSINKIEWICZ S.: Rotation invariant spherical harmonic representation of 3D shape descriptors. In *Proceedings of Symposium in Geometry Processing* (Aachen, Germany, June 2003), Kobbelt L., Schröder P., Hoppe H., (Eds.), pp. 156–165.

[MPS*03] MORTARA M., PATANE G., SPAGNUOLO M., FALCIDIENO B., ROSSIGNAC J.: Blowing bubbles for multi-scale analysis and decomposition of triangle meshes. *Algorithmica 38*, 1 (2003), 227–248.

[RCSM03] RUIZ-CORREA S., SHAPIRO L. G., MEILA M.: A new paradigm for recognizing 3-d object shapes from range data. In *ICCV '03: Proceedings of the Ninth IEEE International Conference on Computer Vision* (Washington, DC, USA, 2003), IEEE Computer Society, p. 1126.

[SF07] SHILANE P., FUNKHOUSER T.: Distinctive regions of 3d surfaces. *ACM Trans. Graph. 26*, 2 (e 07), 7.

[SYS05] SUZUKI M. T., YAGINUMA Y., SHIMIZU Y.: A partial shape matching technique for 3d model retrieval systems. In *ACM SIGGRAPH 2005 Posters* (New York, NY, USA, 2005), ACM Press, p. 128.

[SYYS05] SUZUKI M. T., YAGINUMA Y., YAMADA T., SHIMIZU Y.: A partial shape matching method for 3d model databases. In *Proceedings of the Ninth IASTED International Conference on Software Engineering and Applications (SEA2005)* (Phoenix, USA, November 2005), ACTA Press, pp. 389–394.

[TVD09] TIERNY J., VANDEBORRE J.-P., DAOUDI M.: Partial 3d shape retrieval by reeb pattern unfolding. *Computer Graphics Forum 28* (2009), 41–55.

[VJ01] VIOLA P., JONES M.: Rapid object detection using a boosted cascade of simple features. In *CVPR* (2001).

© The Eurographics Association 2010.