# 3D-shape retrieval using curves and HMM

Hedi Tabia, Olivier Colot, Mohamed Daoudi, Jean-Philippe Vandeborre

HAL Id: hal-00667989

https://hal.science/hal-00667989

Submitted on 8 Feb 2012

# 3D-Shape Retrieval using Curves and HMM

Hedi Tabia, Olivier Colot
*LAGIS FRE CNRS 3303*
*University of Lille 1*
*Villeneuve d'Ascq, France*
*Email: hedi.tabia@telecom-lille1.eu*
*Email: olivier.colot@univ-lille1.fr*

Mohamed Daoudi, Jean-Philippe Vandeborre
*LIFL UMR CNRS 8022*
*Institut TELECOM; TELECOM Lille 1*
*Villeneuve d'Ascq, France*
*Email: mohamed.daoudi@telecom-lille1.eu*
*Email: jean-philippe.vandeborre@telecom-lille1.eu*

*Abstract*—In this paper, we propose a new approach for 3D-shape matching. This approach encloses an off-line step and an on-line step. In the off-line one, an alphabet, of which any shape can be composed, is constructed. First, 3D-objects are subdivided into a set of 3D-parts. The subdivision consists to extract from each object a set of feature points with associated curves. Then the whole set of 3D-parts is clustered into different classes from a semantic point of view. After that, each class is modeled by a *Hidden Markov Model* (HMM). The HMM, which represents a character in the alphabet, is trained using the set of curves corresponding to the class parts. Hence, any 3D-object can be represented by a set of characters. The on-line step consists to compare the set of characters representing the 3D-object query and that of each object in the given dataset. The experimental results obtained on the TOSCA dataset show that the system efficiently performs in retrieving similar 3D-models.

*Keywords*-3D-shape retrieval; HMM; Curve analysis;

## I. Introduction

Three dimensional shape retrieval systems based on visual similarity aim at helping human users browsing large collections of 3D shapes in an interactive and intuitive way. In this framework, the user specifies a 3D model as an example query and the system is expected to sort the entries of the collection by decreasing visual similarity, providing as top results the most similar entries. An important literature has been provided for shape retrieval based on global similarity, presenting methods enabling retrieval of similar objects despite rigid transformations [1], [2] or even non-rigid transformations (such as shape bending or character articulation) [3]. We defer the reader to survey articles [4] for a broad overview of retrieval methods based on global similarity. However, other applicative contexts, such as modeling by example [5] (where new shapes are created by cutting and pasting existing shape sub-parts) or classification [6], might require to retrieve objects with regard to partial similarity. The partial matching is usually approached using the *recognition by part* idea [7], [8]: segmentation of the shape in significant parts, and matching pairs of parts as whole shapes.

In this paper, we propose a new approach for 3D-object matching in presence of non-rigid deformations and partially similar models. Our approach consists of two main steps namely the indexing step and the retrieval step.

The indexing step, which is completely off-line, aims to construct an alphabet from a dataset of 3D-models. The alphabet is used to represent any 3D-object. For this end, the dataset objects are subdivided into a set of parts which are grouped into different classes. The parts, represented by a sequence of curves, of a given class share the same geometric shape. An HMM is then used to learn the shape of each class. Each trained HMM represents a character of the alphabet, and each 3D-object in the dataset is then represented by a set of characters which corresponds to its parts.

The retrieval step, which is on-line, computes the similarity between the query and the 3D-objects in the dataset. Indeed, the query parts are aligned with HMMs (trained in the off-line step). As a result of considering each alignment, we obtain a set of characters representing the 3D-object query. A similarity is then computed between this set of characters and that of each object in the dataset.

The paper is organized as follows. In section 2, the off-line stage of the approach is detailed step by step. The on-line stage is presented in section 3. Section 4 describes experimental results, before concluding.

## II. Off-line step: 3D-object indexing

### A. Partitioning and grouping

For 3D-object partitioning, we use feature point based technique. The feature points are characteristic of the shape and correspond to points of geometrical and perceptual interests in 3D-objects. They are extracted using Tierny et al.'s [7] method: extremities of prominent components are identified by intersecting the sets of extrema of two scalar functions based on the geodesic distances to the two furthest vertices on the surface mesh representing the 3D-object. Figure 1 shows the localization of feature points extracted from a human 3D-object. Around each feature point an indexed collection of simple, closed curves in $\mathbb{R}^3$ is extracted. The part is then considered as a set of curves associated to one feature point. Curves are defined
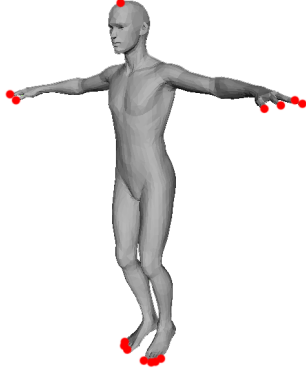
Figure 1. A man 3D-object feature points.



Figure 2. Geodesic path between two curves (last level) of cow-head part and horse-head part.

as level curves of an intrinsic distance function on the 3D-part surface. Let $F_i$ be a feature point on a 3D-triangulated surface. A geodesic distance function is defined on that 3D-surface such that $F_i$ be its origin. The geodesic function is split into a set of levels. Vertices which are in the same level are extracted with respect to an arbitrary order. Let $\lambda$ be a level set corresponding to the geodesic distance function $f$. The set of ordered vertices $V$ such that $f(V) = \lambda$ builds
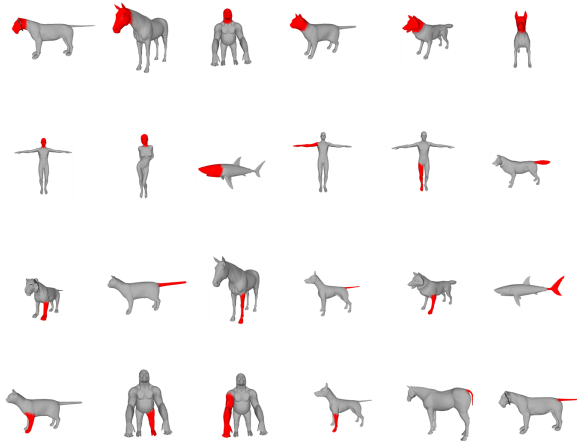


Figure 3. The set of the alphabet (composed of 24 part classes).

one curve. The first row of Figure 2 shows two 3D-parts and their corresponding sets of curves are shown in the second row.

At last, the goal of the grouping step is to fix the character set and to provide in turn a training data to the HMMs. For this end, the 3D-parts, earlier extracted, are grouped manually into $C$ classes (characters). $C$ represents the cardinality of the alphabet. Figure 3 shows 24 representatives of classes deducted from the Tosca dataset.

### B. Class shape learning using HMM

*1) Curve analysis:* Before learning HMM, we need to define a distance between two curves. In our approach, we treat curves as closed, parameterized in $\mathbb{R}^3$ with fixed origins for parameterizations and we rescale them to have the same length, say $2\pi$. This allows us to use one of many methods already available for elastic analysis of closed curves. The key idea in elastic analysis is that the points which are matched together are at unequal distances from their origins. Such matching can be considered as an elastic matching, as one curve has to (locally) stretch, compress and bend to match the other. Here, we adopt the Joshi et al.'s approach [9] because it simplifies the elastic shape analysis. The main steps are: (i) define a space of closed curves of interest, (ii) impose a Riemannian structure on this space using the elastic metric, and (iii) compute geodesic paths under this metric. These geodesic paths can then be interpreted as optimal elastic deformations of curves. We will use the notation $d(\beta_1, \beta_2)$ to denote the geodesic distance, or the length of the geodesic between two curves $\beta_1$ and $\beta_2$. Figure 2 shows the illustration of our idea. The top row shows two 3D-part surfaces and their related feature points. The sets of associated curves are shown in the bottom row. Between them, the geodesic path between the last level curves of both 3D-parts is displayed.

*2) HMM modeling:*

**HMM formalism:** Let us consider $O = \{\beta_1, \beta_2, ..., \beta_\lambda\}$ as the sequence of level curves extracted from a 3D-part. We assume that this sequence is stochastic. Hence, it can be described by an HMM as being an indirect observation of a hidden random sequence $Q = \{q_1, q_2, ..., q_\lambda\}$. The HMM is defined by the following entities:

- $S = \{S_1, S_2, ..., S_N\}$ the finite set of the states.
- the transition matrix $A = \{a_{ij}, 1 \leq i, j \leq N\}$ representing the probability to go from state $S_i$ to state $S_j$, $a_{ij} = P[q_{t+1} = S_j/q_t = S_i], 1 \leq i, j \leq N$ with $a_{ij} \geq 0$ and $\sum_{j=1}^{N} a_{ij} = 1$.
- the emission matrix $B = \{b(\beta_i/S_j)\}$, indicating the probability of the emission of the symbol $\beta_i$ when system state is $S_j$.
- $\pi = \{\pi_i\}$, the initial state probability distribution, representing probabilities of initial states, $\pi_i = P[q_1 = S_i], 1 \leq i \leq N \ with \ \pi_i \geq 0$ and $\sum_{i=1}^{N} \pi_i = 1$.

For convenience, we denote a HMM as a triplet $k = (A, B, \pi)$.

**HMM training:** The standard way to use HMM is to train one model for each class (character). Let's consider the set of parts in one class. Each 3D-part, in this set, is represented by a sequence of curves $\{\beta_i\}$. The training of the model is performed using the standard Baum–Welch re-estimation procedure [10], able to determine the parameters $(A, B, \pi)$ that maximize the probability $P(\{\beta_i\}/k)$. This method is based on the well-known Expectation Maximization (EM) algorithm. No assumptions about the topology have been made in this paper, letting it free to be determined by the transition matrix derived from the training strategy. The observation are determined using a clustering algorithm.

### C. 3D-Object shape learning

After the training of the HMMs, each 3D-object in the dataset can be described by a set of alphabet characters (a set of parts from those shown in Figure 3).

For this end, the parts of a 3D-object are aligned with HMMs and a likelihood is computed for each part. More formally, given a part $P_i$, ($1 \leq i \leq n_P$ with $n_P$ the number of parts of the 3D-object) a vector $L_i = (l_1, ..., l_{24})$ containing the likelihoods is built (see the next section). As a result of considering all the 3D-object parts, we obtain $n_P$ vectors that can be combined to form a resulting vector quantifying the shape of the whole object. $L = \sum_{i=1}^{n_P} (L_i)$.

### III. ON-LINE STEP: 3D-OBJECT RETRIEVAL

### A. Part shape recognition

Upon the receipt of a 3D-object query, we follow the same procedure earlier used to extract feature points and curves. Then, each part sequence of curves is aligned with HMMs and a likelihood is computed. In the HMM theory, there are two possibilities to define the likelihood of an observation sequence for a given HMM. Either the highest likelihood of all possible state sequences is used (Viterbi recognition)[10], or the sum of the likelihoods of all possible state sequences is considered as the likelihood of the observation sequence (Forward recognition)[10]. In the system described in this paper the first possibility is used for recognizing 3D-part shapes.

### B. 3D-object shape retrieval

As a result of considering all parts of the 3D-object query, we obtain a set of likelihood vectors which can be combined to form one resulting vector quantifying the 3D-object query shape. A similarity is then computed between the query and the 3D-objects in the dataset based on the the L2 distance.

### IV. EXPERIMENTAL RESULTS

The proposed approach has been tested on the TOSCA dataset. This dataset has been proposed by Bronstein et al. [11] for non-rigid shape correspondence measures. The
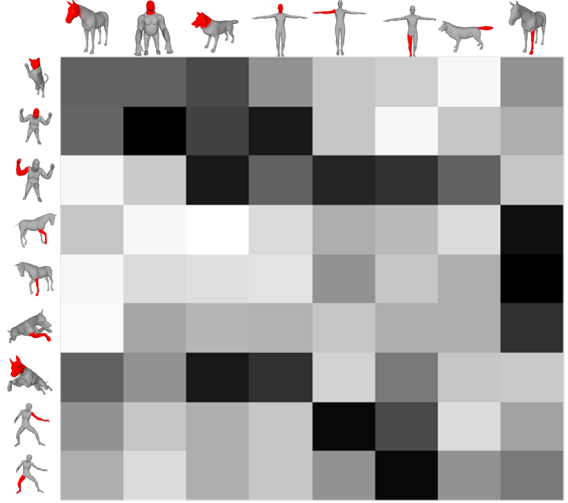


Figure 4. The likelihood of 3D-part queries (rows) and 3D-part models (columns).

database consists of 148 models, enclosing 12 classes. Each class contains one 3D-shape under a variety of poses. For our experiments, the database was divided into independent training set (30 3D-objects) and test sets (118 3D-objects).

From a qualitative point of view, Figure 4 shows the effectiveness of the method in recognizing similar parts. In this Figure rows show 3D-part queries and columns present some parts from the alphabet. In this visualization of the matrix, the lightness of each element $(i; j)$ is proportional to the magnitude of the distances between 3D-objects $i$ and $j$. That is, each square, in this matrix, represents the distances between two 3D-objects. Darker elements represent better matches, while lighter elements indicate worse matches.

For the global similarity metric, Figure 5 shows some 3D-queries (rows) and their similarities with some 3D-objects in the dataset. One can notice the effectiveness of our method in the presence of non-rigid transformation and partially similar models. Human in Figure 5 are similar together. They are more similar to gorillas and centaurs than animals. Centaurs are similar together and partially similar to human and animals. From a more quantitative point of view, in Figure 6, we present the recall and precision plot of the approach compared with the geodesic D2. The Geodesic D2 is an extension of the Euclidean D2 [12]. It is computed as a global distribution of geodesic distances in 3D-shapes. Figure 6 shows that our approach provides better retrieval precision in this experiment.

### V. CONCLUSION

In this paper, a new method for 3D-shape retrieval has been proposed based on curve analysis and Hidden Markov Models. A set of an indexed closed curves in $\mathbb{R}^3$ is extracted from 3D-part surfaces. Curves represent a sequence of ob-
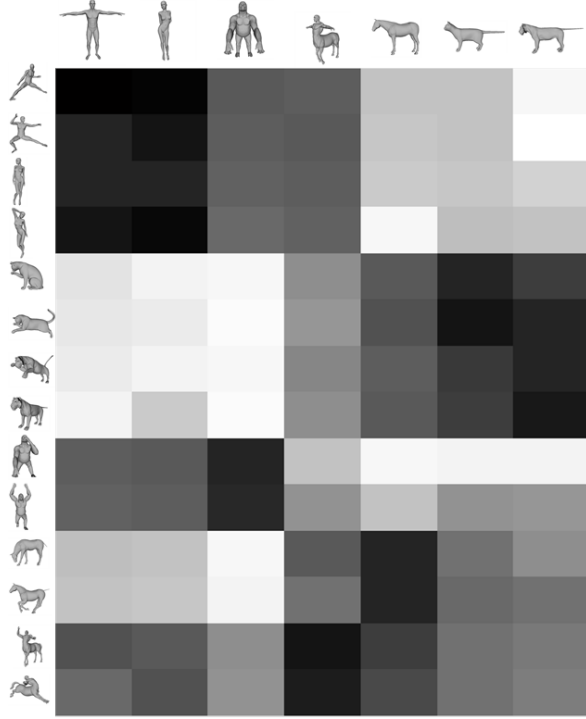
Figure 5. The similarity of 3D-queries (rows) and 3D-object models (columns).
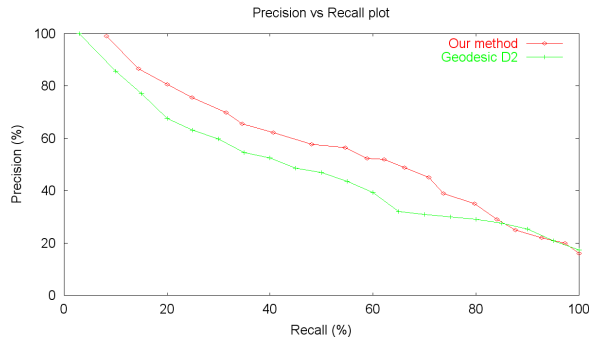


Figure 6. The precision vs recall plot of our approach.

servation in the HMM formalism. Part recognition is carried out using a Viterbi algorithm. A global similarity metric is computed based on the 3D-part likelihoods. A thorough experimental evaluation has shown that the proposed approach is very promising for recognition 3D-parts and 3D-objects.

## ACKNOWLEDGMENT

## REFERENCES

[1] D.-Y. Chen, X.-P. Tian, and M. Shen, Y-T.and Ouhyoung, "On visual similarity based 3d model retrieval," *Computer Graphics Forum*, vol. 22, pp. 223–232, 2003.

[2] G. Antini, S. Berretti, A. Del Bimbo, and P. Pala, "Retrieval of 3d objects using curvature correlograms," in *IEEE International Conference on Multimedia & Expo*, July 2005.

[3] R. Gal, A. Shamir, and D. Cohen-or, "Pose oblivious shape signature," *IEEE Transactions on Visualization and Computer Graphics*, vol. 13, pp. 261–271, 2007.

[4] B. Bustos, D.-A. Keim, D. Saupe, T. Schreck, and D.-V. Vranic, "Feature-based similarity search in 3d object databases," *ACM Computing Surveys*, vol. 37, pp. 345–387, 2005.

[5] T. Funkhouser, M. Kazhdan, P. Shilane, P. Min, W. Kiefer, A. Tal, S. Rusinkiewicz, and D. Dobkin, "Modeling by example," *ACM Transactions on Graphics*, vol. 23, pp. 652–663, 2004.

[6] D. Huber, A. Kapuria, R. Donamukkala, and M. Hebert, "Parts-based 3d object classification." in *IEEE Computer Vision and Pattern Recognition*, 2004, pp. 82–89.

[7] J. Tierny, J.-P. Vandeborre, and M. Daoudi, "Partial 3D shape retrieval by reeb pattern unfolding," *Computer Graphics Forum - Eurographics Association*, vol. 28, pp. 41–55, March 2009.

[8] Y. Liu, H. Zha, and H. Qin, "Shape topics: A compact representation and new algorithms for 3d partial shape retrieval," in *Computer Society Conference on Computer Vision and Pattern Recognition*, 2006.

[9] S. Joshi, E. Klassen, A. Srivastava, and I. Jermyn, "Removing shape-preserving transformations in square-root elastic (sre) framework for shape analysis of curves," in *EMMCVPR*, 2007, pp. 387–398.

[10] L. Rabiner and B. Jnuang, "Fundamentals of speech recognition," *Prentice-Hall, Englewood Cliffs*, 1993.

[11] A. Bronstein, M. Bronstein, and R. Kimmel, "Efficient computation of isometry-invariant distances between surfaces," *IEEE Trans. Visualization and Computer Graphics*, vol. 13/5, pp. 902–913, 2007.

[12] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin, "Shape distributions." *ACM Transactions on Graphics (TOG)*, vol. 21(4), pp. 807–832, 2002.