A Comparison Framework for 3D Object Classification Methods

S. Biasotti, D. Giorgi, S. Marini, M. Spagnuolo, and B. Falcidieno

CNR-IMATI, Genova, Italy,
{silvia,daniela,simone,michi,bianca}@ge.imati.cnr.it,
WWW home page: http://www.ge.imati.cnr.it

Abstract. 3D shape classification plays an important role in the process of organizing and retrieving models in large databases. Classifying shapes means to assign a query model to the most appropriate class of objects: knowledge about the membership of models to classes can be very useful to speed up and improve the shape retrieval process, by allowing the reduction of the candidate models to compare with the query.
The main contribution of this paper is the setting of a framework to compare the effectiveness of different query-to-class membership measures, defined independently of specific shape descriptors. The classification performances are evaluated against a set of popular 3D shape descriptors, using a dataset consisting of 14 classes made up of 20 objects each.

1 Introduction

The development of 3D data collections and 3D retrieval methods is rapidly increasing, due to the advances in 3D data generation and processing [1, 2]. Given a query model, retrieval is the process that ranks the objects in a database according to their similarity to the query, while classification is the process that identifies the class in the database which the query belongs to. Retrieval and classification are obviously related. Classification makes explicit some kind of similarity within large sets of objects. Therefore it is a natural way to organize a 3D database. Shape classification can also be used as a pre-processing in a retrieval system: once the query-model class has been identified, the search space for the retrieval can be significantly reduced, thus speeding up the process. Moreover, the reduction of the set of objects to be compared with the query permits to achieve higher quality results by diminishing the number of false positives in the system response.

Until now most research on 3D classification of large databases has been performed in Computer Vision [3, 4]. Usually all techniques distinguish the training phase, in which the classes of the database are constructed (even manually), from the actual classification, that associates the query model to one class, or proposes a ranking of possible memberships. The use of groups, or classes, and class prototypes has been also proposed to handle large databases [5, 6].

Methods that automatically classify 3D models consider either the geometry of the model or its structure or both. For instance, the method in [7] uses reasoning on geometric properties, like curvature, orientation and planarity, while

in [3] objects are described in terms of graph descriptors and a method for the automatic generation of a class representative is proposed. Finally, the method proposed in [8] divides the models into sub-parts that are grouped into part classes using a bottom-up hierarchical clustering method, named agglomerative clustering.

In this paper we propose a formal workbench consisting of five different shape classifiers, each one defined as a distance function between a query object and a previously defined database class. Each of these classifiers has its own properties, which are discussed and compared. Experiments are performed on a database of 280 models grouped into 14 disjoined classes, in order to investigate the behavior of the five classifiers. Moreover, results are provided to discuss the suitability of four different popular 3D shape descriptors with respect to shape classification. To our knowledge, this is the first time that such a comparative evaluation study is proposed in the context of 3D shape classification.

The remainder of the paper is organized as follows. In Section 2 the five classifiers are proposed, highlighting their main properties. Then, in Section 3 these classifiers are tested on the database, with respect to four shape similarity methods. Moreover, we show the potential of shape classification to improve the retrieval process. Conclusions and future developments end the paper.

2 Classification Framework

Suppose we are given a database D containing n models, which are grouped into m classes of similar objects. By *similar objects* we mean that objects share a common form and/or structure and/or function. When a query model Q is provided – Q not necessarily belonging to D – the aim of a classification process is to determine the class the object most reasonably belongs to. The classification process involves the description of each model by a shape signature (the *shape descriptor*), and the introduction of a suitable *(dis-)similarity measure* between descriptors. Obviously, many different choices are possible both for the shape signatures and the (dis-)similarity measures, each characterized by different properties. The aim of this section is to introduce a formal framework for shape classification, whose definition is independent of the choice of particular descriptors and measures.

*O*bject classification by shape signatures is formally expressed as follow:

Given a database $D = \{M_i\}$, $i = 1, \ldots, n$ whose n models are grouped into m classes C_k, $k = 1, \ldots, m$, so that $C_k \neq \phi \ \forall k$, $\bigcup_k C_k = D$ and $C_k \bigcap C_l = \phi$ for $1 \leq k, l \leq m, k \neq l$, and given a query model Q:

1. represent the models in the database by a set of shape signatures $\{s_i\}_i$, $i = 1, \ldots, n$ and the query by a signature q;
2. use the signature representation to classify the query, i.e. select the class $C_{\overline{k}}$ which minimizes the distance $\tilde{d}$ between the query and the class. In symbols:

$$q \mapsto C_{\overline{k}} \Longleftrightarrow C_{\overline{k}} = \arg \min_{C_k \in D} \ \tilde{d}(q, C_k). \tag{1}$$

Evaluating the distance $\tilde{d}$ involves the computation of a (dis-)similarity measure d between descriptors. We propose five different definitions for $\tilde{d}$, independently of d: the *Minimum Distance Classifier*, the *Maximum Distance Classifier*, the *Average Distance Classifier*, the *Centroid Distance Classifier*, and the *Atipicity Distance Classifier*. The only assumption on d is that of being a *dissimilarity* measure rather than a similarity measure, i.e. a measure which assumes lower values as the resemblance between models increases. Obviously, this assumption can be easily changed by modifying the proposed definitions to include both measures. This is not discussed here for the sake of space limitation.

Minimum Distance Classifier (MinDC). The distance $\tilde{d}(q, C_k)$ is defined as the minimum distance between the query descriptor q and the descriptors belonging to C_k, as shown in equation (2):

$$\tilde{d}(q, C_k) = \min_{s \in C_k} d(q, s). \tag{2}$$

This classifier coincides with the popular Nearest Neighbor classifier.

Maximum Distance Classifier (MaxDC). In this case the distance between the query q and the class C_k is obtained by selecting the maximum distance among the query and the members of the class, as in equation (3):

$$\tilde{d}(q, C_k) = \max_{s \in C_k} d(q, s). \tag{3}$$

Hence the query is classified by taking into account the most dissimilar descriptor belonging to the class.

Average Distance Classifier (AvgDC). $\tilde{d}$ is defined as the average distances between the query and the members of the class, as shown in equation (4):

$$\tilde{d}(q, C_k) = \frac{\sum_{s \in C_k} d(q, s)}{|C_k|}. \tag{4}$$

It is worth noticing that the $MinDC$ does not take into account the heterogeneity of the class, that is, an overall estimate of how much the members of the class resemble each other. On the contrary, the $MaxDC$ is based on a strong assumption of class homogeneity, since the performance of this classifier naturally decays if there exists in a class a shape descriptor which differs much from the other descriptors in the same class. Also, if the query is classified into a particular class $C_{\overline{k}}$ according to the $MaxDC$, we can say that the distance between the query and the class is smaller than the distances between the query and all the other members of $C_{\overline{k}}$. The dependence of the classification performance on the heterogeneity of the classes is reduced when considering $AvgDC$.

The main drawback of the considered classifiers is the need of computing all the comparisons between the query and the models in the database, which motivates the introduction of the following two classification schemes, namely the

Centroid and the *Atipicity Distance Classifier*. For these measures, the number of comparisons to be performed at run time is reduced by selecting a representative model $\overline{s}$ for each class, and evaluating the distance between the query and a class as the distance between the query and the representative of that class:

$$\tilde{d}(q, C_k) = d(q, \overline{s}_k). \tag{5}$$

Since the class representatives can be computed and stored in an off-line step, at run time matching a query against the representatives is sufficient for classification. Two proposals for the representatives are detailed in what follows.

Centroid Distance Classifier (CDC). The distance d is defined as in equation 5, where the representative $\overline{s}_k$ of a class C_k is the member of C_k which satisfies

$$\overline{s}_k = \arg\min_{s \in C_k} avg_{C_k}(s) \tag{6}$$

where the function $avg_{C_k}(\cdot)$ is given by

$$avg_{C_k}(s) = \frac{\sum_{r \in C_k} d(s, r)}{|C_k|} \tag{7}$$

with $|\cdot|$ the cardinality of the class.

Atypicity Distance Classifier (ADC). This classifier evokes the notion of typicity introduced in [9], to represent how much a descriptor is typical of the class it belongs to, with respect to the elements in the other classes. Such concept is modified to cope with dissimilarity rather than similarity measures. As before, the distance $\tilde{d}$ is defined as in equation 5, while the representative $\overline{s}_k$ of a class C_k is chosen as the member of C_k verifying

$$\overline{s}_k = \arg\min_{s \in C_k} atyp_{C_k}(s) \tag{8}$$

where the function $atyp_{C_k}(\cdot)$ is given by

$$atyp_{C_k}(s) = \frac{avg_{C_k}(s)}{avg_{S'}(s)} \tag{9}$$

with S' the set of all descriptors not belonging to the class C_k. Notice that the lower the value of $atyp(\cdot)$, the higher the level of tipicity of the descriptor for the given class. Values close to 1 represent members that strongly resemble members of other classes.

3 Experiments

In this section we provide a set of experimental results to assess both the potential of the five classifiers proposed in Section 2 and the effectiveness of four popular shape retrieval methods for the classification problem. With respect to

the classification of similarity and retrieval methods in [2], we have chosen a volume-based descriptor, the spherical harmonics (SH) in [10], an image-based descriptor, the lightfield descriptor (LF) in [11], and two topological matching methods, the Multiresolution Reeb graph in (MRG) [12] and the Extended Reeb graph (ERG) in [13]. We are not aware of any prior experimentation that evaluates the capability of these techniques to cope with the classification task.

To this purpose, we are using a pre-classified test database of 280 triangle meshes grouped in 14 classes, each one consisting of 20 elements, see Figure 1. The original models of our database were collected from several web repositories: the AIMSHAPE repository[1], the National Design Repository at Drexel University[2], the CAESAR Data Samples[3] and the McGill 3D Shape Benchmark[4].

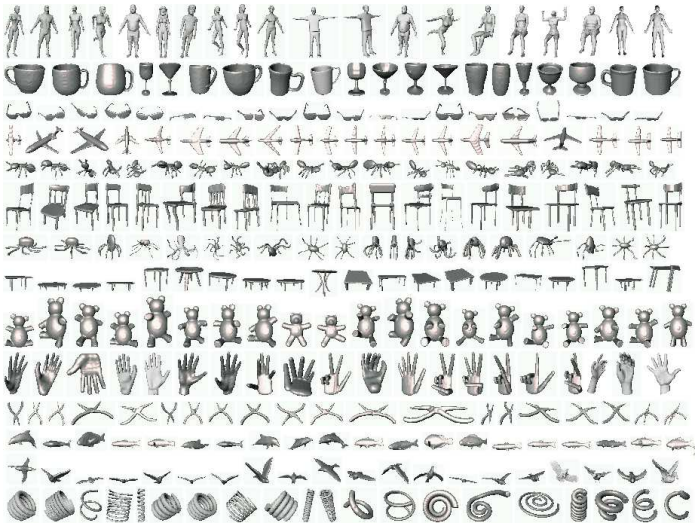

Fig. 1. Our database. Each row represents a class of objects.

As a first test, a *classification rate* has been computed for each distance classifier. The classification performance of every classifier and descriptor is evaluated through a "leave-one-out" strategy. Each class of the database is split into four subsets of five elements and the classification is performed by using the five models as queries against the remaining fifteen descriptors of the class. Once this procedure has been repeated for the four subsets of models, the classification rate is obtained by computing the percentage of queries correctly classified. Since a distance classifier also provides the distance between a query object and

[1] http://shapes.aim-at-shape.net
[2] http://www.designrepository.org
[3] http://www.hec.afrl.af.mil/HECP/Card1b.shtml#caesarsamples
[4] http://www.cim.mcgill.ca/∼shape/benchMark/

Methods	Classifiers										Average	
	MinDC		MaxDC		AvgDC		CDC		ADC			
	Rate	Pos	Rate	Pos	Rate	Pos	Rate	Pos	Rate	Pos	Rate	Pos
SH	89%	2	33%	6	66%	3	64%	3	63%	3	63%	4
LF	88%	2	38%	5	73%	3	68%	3	68%	3	67%	4
MRG	88%	2	41%	5	74%	2	76%	2	73%	2	70%	3
ERG	83%	2	38%	5	58%	3	60%	3	58%	3	59%	4
	87%	2	37%	6	67%	3	67%	3	65%	3		

Table 1. Overall performance of our classifiers with respect to the four descriptors. Rate is the percentage of query models which are correctly classified, Pos is the position of the correct class with respect to the rank identified by the clssifier.

the database classes, the smaller the score of the class the object belongs to, the better the performance of the classifier. Therefore, the rank in which the right class is recognized is a good indicator of the classification effectiveness.

The results of the experiments both on classifiers and descriptors are shown in Table 1. Each entry is related to the performance of a given shape descriptor (enumerated in the first column) for a given classifier (reported in the second row of the table). The performance is evaluated in terms of classification rate (i.e. the percentage of query models which are correctly classified) and the position of the correct class in the ranked list of classes. The last row highlights the performance of the 5 classification schemes, since the values in this row are averaged over the 4 shape descriptors. Analogously, the last column refers to the performance of the shape descriptors for the classification task.

The highest values for the classification rate are reached by the descriptors when they are coupled with the $MinDC$ classifier. This classification scheme allows to obtain a perfect score by considering only the first 2 classes in the system response. In the same way, in our experimental setting the worst classifier results to be the $MaxDC$, independently of the chosen shape descriptor, as it can be deduced from the low values of the classification rates and the high number of classes required to guarantee that an object is correctly classified. The two classification schemes based on the selection of a representative model show similar performances when averaged over the whole database. As shown, the classification rate goes from a minimum of 58% to a maximum of 76%, depending on the chosen descriptor. The number of classes to examine for ensuring a correct classification is between 2 and 3. We remind here that the added value of these classification approaches is that they allow a strong reduction of the number of comparisons to be performed at run time.

Figures 2 and 3 show the different behavior of the classifiers and the descriptors on the database. The classification rates are averaged over the descriptors and the classifiers, respectively. Finally, Figure 4 shows how prior classification of a query improves the results of the retrieval process in terms of precision, by discarding a number of false positives. The top 10 retrieval results for a human model are shown with respect to the whole database, and after restricting

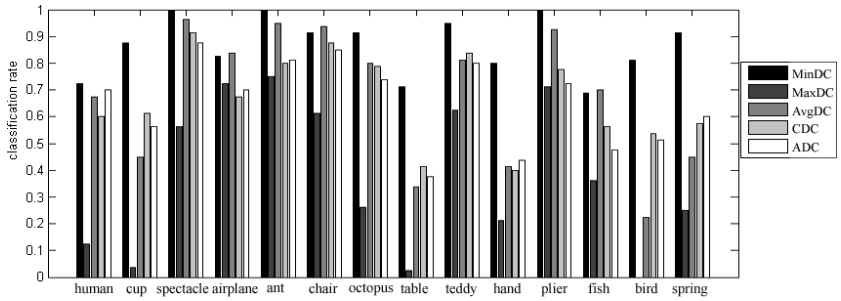

Fig. 2. Classification performance of the distance classifiers w.r.t. to the single classes.

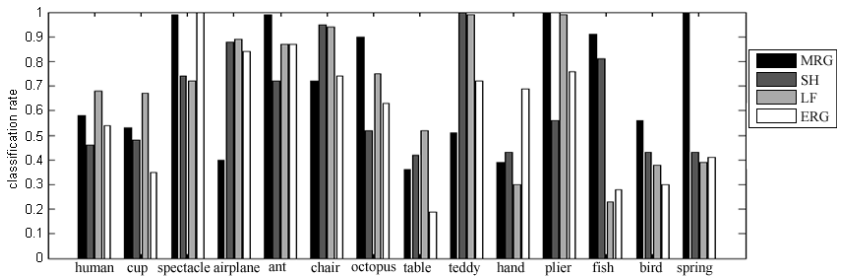

Fig. 3. Classification performance of the shape descriptors w.r.t. the single classes.

the search to the top classes in the classification (using the LF descriptor, the MinDC classifier and the "humans" and "teddy" classes).

4 Conclusion

We have described a general framework for comparing different classification methods proposing experiments on five classifiers and four shape descriptors. Currently, we are investigating how to extend our comparison framework to improve the classification of large databases. In particular, we are focusing on the definition of classifiers based on model prototypes. Further research will focus on the combination of different descriptors concurring to the classification of the single model classes. Finally, we are planning to introduce a statistical training on the database classification in order to reduce the search space for object retrieval.

Acknowledgements

This work has been developed in the CNR research activity (ICT-P03) and it is partially supported by the EU NoE "AIM@SHAPE" (http://www.aimatshape.net).

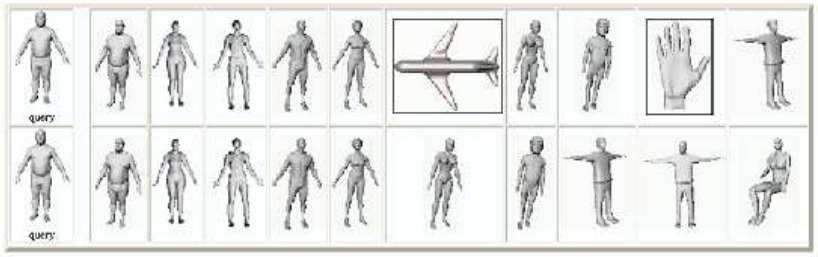

Fig. 4. Top 10 retrieval results with respect to the LF descriptor when the query is performed on the whole database (first row) and on the 2 top classes obtained with respect to the MinDC.

References

1. Tangelder, J., Veltkamp, R.: A survey of content based 3d shape retrieval methods. In: Proc. Shape Modeling Applications 2004. (2004) 145–156
2. Bustos, B., Keim, D.A., Saupe, D., Schreck, T., Vranić, D.V.: Feature-based similarity search in 3D object databases. ACM Computing Surveys **37**(4) (2005) 345–387
3. Sengupta, K., Boyer, K.L.: Organizing large structural mordelbases. IEEE Trans. on Pattern Analysis and Machine Intelligence **17**(4) (1995) 321–332
4. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification. John Wiley and Sons Inc. (2001)
5. Lam, W., Keung, C.K., Liu, D.: Discovering useful concept prototypes for classification based on filtering and abstraction. IEEE Trans. on Pattern Analysis and Machine Intelligence **24**(8) (2002) 1075–1090
6. Donamukkala, R., Huber, D., Kapuria, A., Hebert, M.: Automatic class selection and prototyping of 3-D object classification. In: Proc. 5^{th} Int. Conf. on 3-D Digital Imaging and Modeling /3DIM'05), IEEE (2005) 64–71
7. Csákáky, P., Wallace, A.M.: Representation and classification of 3-D objects. IEEE Trans. on Systems, Man and Cybernetics - Part B: Cybern. **33**(4) (2003) 638–647
8. Huber, D., Kapuria, A., Donamukkala, R., Hebert, M.: Part-based 3D object classification. In: Proc. IEEE Conf. on Computer Vision and pattern Recognition (CVPR'04). Volume 2. (2004) 82–89
9. Zhang, J.: Selecting typical instances in instance-based learning. In: Proc. Int. conf. Machine Learning. (1992) 470–479
10. Kazhdan, M., Funkhouser, T., Rusinkiewicz, S.: Rotation invariant spherical harmonic representation of 3D shape descriptors. In Kobbelt, L., Schröder, P., Hoppe, H., eds.: Proc. Symposium in Geometry Processing. (2003) 156–165
11. Chen, D., Ouhyoung, M., Tian, X., Shen, Y.: On visual similarity based 3D model retrieval. Computer Graphics Forum **22** (2003) 223–232
12. Hilaga, M., Shinagawa, Y., Kohmura, T., Kunii, T.L.: Topology matching for fully automatic similarity estimation of 3D shapes. In: Computer graphics proceedings, annual conference series: SIGGRAPH conference. (2001) 203–212
13. Biasotti, S., Marini, S.: 3D object comparison based on shape descriptors. International Journal of Computer Applications in Technology **23**(2/3/4) (2005) 57–69