Extracting Understandable 3D Object Groups with Multiple Similarity Metrics

Antonio Adán and Miguel Adán

Departamento Ingeniería E. E. A. C. Universidad de Castilla La Mancha. Spain
{Antonio.Adan,Miguel.Adan}@uclm.es

Abstract. Some of the main difficulties involved in the clustering problem are the interpretation of the clusters and the choice of the number of clusters. The imposition of a complete clustering, in which all the objects must be classified might lead to incoherent and not convincing groups. In this paper we present an approach which alleviates this problem by proposing incomplete but reliable clustering strategies. The method is based on two pillars: using a set of different metrics which are evaluated through a *clustering confidence measure* and achieving a hard/soft clustering consensus. This method is particularly addressed to 3D shape grouping in which the objects are represented through geometric features defined over mesh models. Our approach has been tested using eight metrics defined on geometrical descriptors in a collection of free-shape objects. The results show that in all cases the algorithm yields coherent and meaningful groups for several numbers of clusters. The clustering strategy here proposed might be useful for future developments in the unsupervised grouping field.

Keywords: 3D shape representation, 3D Shape similarity, 3D Object Clustering.

1 Incomplete Grouping

Different clustering techniques, either hierarchical or partitional based strategies, aim to categorize similar objects together using any type of similarity metrics. Agglomerative clustering methods [1], place all objects into singleton clusters and iteratively merge them one at a time. Others, like spectral clustering [2] and graph partitioning [3] methods, partition the data into relatively dense subgraphs, minimizing the edges between the subgraphs. In addition, semi-supervised (or constrained) clustering methods have recently attracted considerable attention [4, 5].

In all those cases, the use of a unique metric and the imposition of a complete clustering (in which all the objects of the dataset must be classified in a particular cluster) might lead to incoherent solutions. The group coherence is broken when outliers (objects clearly different to the rest in the group) are included. In this paper we propose an approach which eliminates these two constraints. The paper's originality can be synthesized in two aspects: a clustering consensus between several similarity metrics, which will guarantee coherent groups, and an incomplete clustering proposal, which signifies that some objects might not belong to any cluster.

L. Alvarez et al. (Eds.): CIARP 2012, LNCS 7441, pp. 179–186, 2012.

Traditional clustering methods have been developed to analyze complete data sets. However there exist some examples which deal with clustering in incomplete datasets. Cheng et al. [6] produce fine clusters on incomplete high-dimensional data space. Hathaway et al. [7] introduce four strategies for doing fuzzy c-means (FCM) clustering of incomplete data sets. Incomplete data consists of vectors that are missing one or more of the feature values. All approaches are tested using real and artificially generated incomplete data sets. In [8] the so-called kernel fuzzy c-means algorithm adopts a new kernel-induced metric in the data space to replace the original Euclidean norm metric in FCM. It is used to cluster incomplete data. Himmelspach et al. [9] present an extension for existing fuzzy c-means clustering algorithms for incomplete data, which uses the information about the dispersion of clusters.

Although algorithms dealing with clustering in incomplete datasets are relatively frequents in literature ([6-9]), to the best of our knowledge there are no works focused on incomplete clustering, in which we understand the term "incomplete" as that explained above, that is "some objects might not belong to any cluster".

Another clear difference between our approach and the aforementioned works is that we compare the clustering results for several similarity measures and, finally, we make decisions by integrating the clustering results. Most of the works just evaluate different approaches and select the most convenient but they do not integrate metrics. Some examples can be found in [10] and [11]. Jayanti et al [10] evaluate the clustering results on a classified benchmark database of 3D models after using two different approaches. The authors solely compare the clustering effectiveness of these two approaches, and do not integrate the methods. In [11] three existing 3D similarity search methods were chosen but no integration of methods is therefore performed.

Our approach is explained in the following sections. Sections 2 and 3 tackle the evaluation and clustering procedure. We present the term *confidence measure* as the parameter which globally evaluates the clustering result from each particular similarity metric and explain how incomplete clusters are extracted after a simple voting strategy. Section 4 is devoted to showing the experimental work and results. Our conclusions are provided in Section 5.

2 Clustering Evaluation for Different Metrics

Since our method is used in the object clustering research context, from here on we will talk about object clustering or, in general, 3D shape clustering.

Let us assume an object database with n objects O_i , i=1,…n and a set of metrics $d_p, p = 1, \dots r$ which have previously been defined. We define *r,* nxn Similarity Matrices S_p as follows:

$$S_{p,ij} = d_p(O_i, O_j), \qquad i, j = 1, \dots n, p = 1, \dots r \tag{1}$$

Note that each Similarity Matrix stores the entire similarity information in a database, depending on the metric used. These Similarity Matrices S_p are used to carry out the clustering process. We specifically apply the Ward's hierarchical clustering method [12] based on the Similarity Matrices of equation (1). As is known, in hierarchical strategies, the data (in our case *S*) are not partitioned into a set of groups in a single step. Instead, a series of partitions takes place, which may run from a single cluster

containing all objects to *n* clusters, each containing a single object. Hierarchical clustering is subdivided into agglomerative methods, which proceed by a series of fusions of the *n* objects into groups, and divisive methods, which separate *n* objects into successively finer groupings. The result may be represented by a dendogram which illustrates the fusions or divisions that take place in each successive stage of analysis. In each step, the clusters are defined by minimizing an objective function which usually measures the separation between clusters.

Ward's linkage uses the incremental sum of squares; that is, the increase in the total within-cluster sum of squares as a result of joining two clusters. The within-cluster sum of squares is defined as the sum of the squares of the distances between all objects in the cluster and the centroid of the cluster. For example, for clusters C_i and C_j, the Ward's equivalent distance is given by:

$$D^2(C_i, C_j) = \langle C_i \rangle \langle C_j \rangle \frac{\left\| \bar{x}_{C_i} - \bar{x}_{C_j} \right\|^2}{\langle C_i \rangle + \langle C_j \rangle} \tag{2}$$

where $\| \ \|$ signifies Euclidean distance, $\bar{x}_{C_i}$ and $\bar{x}_{C_j}$ are the centroids, and $\langle C_i \rangle$ and $\langle C_j \rangle$ are the respective number of objects in cluster C_i and C_j.

Let us now assume that the set of objects has been grouped into *w* clusters C_j , $j=1,\ldots w$, $w<n$ following our hierarchical clustering approach. In order to evaluate the goodness of the grouping result we propose the object confidence value K which measures how well a particular object is assigned to a certain cluster. Equation (3) shows the object confidence value when object O_i is assigned to cluster C_k after using a metric d_p.

$$K_p(O_i | O_i \in C_k) = \frac{\min\{\hat{d}_p(O_i, C_j), \forall j \neq k\} - \hat{d}_p(O_i, C_k)}{max\left\{min\{\hat{d}_p(O_i, C_j), \forall j \neq k, \hat{d}_p(O_i, C_k)\}\right\}} \qquad p = 1, \ldots r \tag{3}$$

$$\hat{d}_p(O_i, C_j) = \frac{\sum_{\forall O_h \in C_j} d_p(O_i, O_h)}{\langle C_j \rangle} \tag{4}$$

Equation (4) calculates the mean inter-cluster distance of object O_i, and symbol $\langle . \rangle$ signifies the cardinal of a set.

Given a clustering proposal, the mean of the object confidence values for all the objects in the database is taken as the overall evaluation parameter. Thus, the *clustering confidence measure* for metric *p*, $\bar{K}_p$, is eventually defined as follows:

$$\bar{K}_p = \frac{\sum_{i=1}^{n} K_p(O_i)}{n} \qquad p = 1, \ldots r \tag{5}$$

3 Hard/Soft Incomplete Clustering

The definitive groups are built by using the *clustering confidence measures* for each metric and setting the number of clusters. Let us assume that p metrics are used and that up to w clusters can be taken in an object database. A *clustering confidence matrix* $M(pxw)$ containing parameters $\bar{K}_p$ for p rows, each corresponding to a particular metric, and w columns, each for a particular number of clusters, is then obtained. The objective is to extract the most suitable numbers of clusters and the best metrics for each case.

The system extracts a vector containing the number of clusters which reports the highest $\overline{K}_p$ value per row in M and obtains the median of the vector as the most reliable choice. The best metrics are also chosen for each column of M, penalising the cases which yield unitary clusters. Let $\hat{w}$, $\hat{w} < w$, and $\hat{p}$, $\hat{p} < p$, be the number of selected clusters and metrics.

The next stage consists of achieving a common cluster labeling for all the metrics considered. A common labeling is imposed on all the clusters from each metric by considering the maximum concordance between them. The main idea is that the same label is assigned to the clusters belonging to different metrics in which the intersection is maximum. This is synthesized in equation (6). A short explanation follows.

Let $\hat{p} \leq p$ and $\hat{w} \leq w$ be the number of metrics and clusters established in the earlier stage, C_k^j be the k-th cluster after applying the j-th metric, $\langle A \cap B \rangle$ be the number of common objects in generic clusters A and B, $\langle C_m^1 \cap \{C_k^j\}_{k=1\ldots\hat{w}} \rangle$ be the $\hat{w}$-vector which contains the number of common objects in the *m*-th cluster of the first metric with each one of the clusters obtained with the *j*-th metric. Assume also that the labeling of the first metric is denoted as $\Lambda(C_k^1) = k,\ k = 1, \ldots \hat{w}$.

In equation (6), $\Lambda(m)$ is a $\hat{p}$-vector containing the indices of the *m*-th group for each particular metric. Figure 1 a) presents an explicatory example of the labeling stage.

$$\Lambda(m) = \left\{ \arg max_k \langle C_m^1 \cap \{C_k^j\}_{k=1\ldots\hat{w}} \rangle \right\}_{j=2\ldots\hat{p}} ,\ m = 1, \ldots \hat{w} \tag{6}$$

In case of a certain metric *h* provides equal number of common objects for two or more clusters, the indexation of *j* in (6) is updated so that $j = 1 \ldots h-1, h+1, .. \hat{p}, h$. That is, the turn corresponding to the metric *h* is postponed at the end of the list.

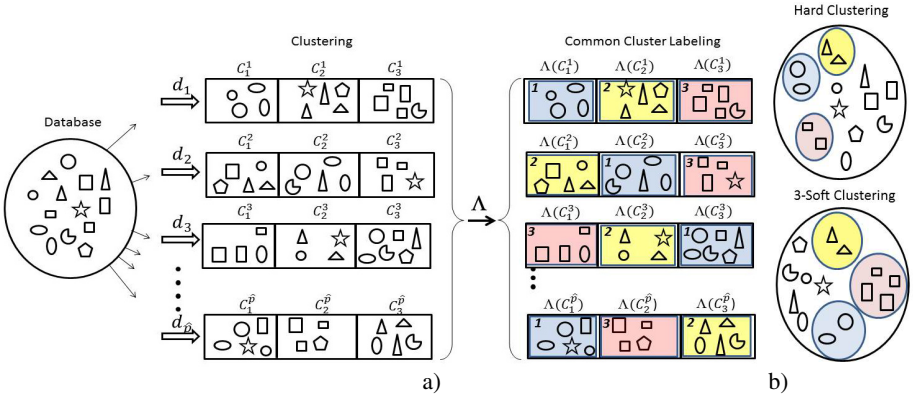

Fig. 1. Explanatory picture of the common cluster labeling process. a) The figure represents fourteen objects which are grouped in three cluster according $\hat{p}$ metrics. Labels 1, 2, and 3 are depicted in different colors. b) Incomplete clustering using *hard* and *3-soft coincidence* criteria taking $\hat{p} = 4$.

The third step concerns the integration and definition of the definitive incomplete clusters, taking a simple voting-consensus strategy. We defined two performance parameters to quantitatively evaluate the clustering concordance of an object from different metrics. We thus distinguish between *hard coincidence* and *soft coincidence*.

Hard coincidence occurs when all metrics classify the object in the same labeled cluster, whereas *j-soft coincidence* occurs if at least *j*, ($1<j<\hat{p}$), metrics do so. It is consequently possible to obtain the respective *hard coincidence groups* and *j-soft coincidence groups*. Figure 1 b) illustrates the clustering results after taking four metrics. Note that the clustering is incomplete for *hard* and *3-soft coincidence*.

4 Experimental Results

Our approach has been tested to cluster 3D shapes. To characterize a 3D shape we chose the representation model based on *RC-images* and took eight different metrics. Definition and metrics based on *RC*-images can be found in [13]. Basically a *RC-image* is an image in which the pixel (*r, c*) contains the relative voting frequency of features *r* and *c* in the nodes of the mesh model; *r* being the distance of the node to the mesh center and *c* being the absolute value of the curvature around the node. The *RC-image* is normalized, so that $r \in [0,1]$ and $c \in [0,\pi/2]$.

We built a model database with 45 real free-form shapes and studied the performance of the method through the parameter $\overline{K}_p$ for a different number of classes and similarity metrics. We then selected the three best metrics and analyzed the concordance between the respective clusters obtained. As will be shown, our method was able to provide coherent-incomplete clustering results after establishing a voting-consensus between metrics.

The first analysis stage consisted of calculating the *clustering confidence matrix* M. Table 1 shows M for $w = 6$ and $p = 8$. Each row corresponds to a particular metric d_p and each column for a particular number of clusters w. The last row and column corresponds to the average values of $\overline{K}_p$ per metric and per number of clusters respectively. Taking the highest $\overline{K}_p$ value per row, the optimum numbers of cluster are {3,5,2,2,6,6,3,2}, and we therefore chose the three most voted cases, that is $w_1 = 2, w_2 = 3, w_3 = 6$. With regard to the best metrics, we performed a selection for each *w,* bearing in mind several aspects such as: the average $\overline{K}_p$ values and the number of unitary clusters. The criterion is that if any unitary cluster appears for all the number of clusters selected, the metric is refused. The selected triplets were

Table 1. Clustering *confidence measure* calculated for metrics d_1 to d_8 and two to six clusters

#clusters / Metric	#2	#3	#4	#5	#6	Average
d_1	0,3164	0,3328	0,2631	0,2461	0,2637	0,2844
d_2	0,3124	0,3162	0,3333	0,3638	0,3006	0,3252
d_3	0,3563	0,3342	0,3049	0,3269	0,3398	0,3324
d_4	0,6053	0,4339	0,1859	0,2033	0,2151	0,3287
d_5	0,2226	0,2202	0,237	0,2397	0,2656	0,2370
d_6	0,1365	0,1687	0,2084	0,2365	0,2631	0,2026
d_7	0,2963	0,322	0,306	0,2926	0,3163	0,3067
d_8	0,4749	0,351	0,3009	0,3019	0,3157	0,3489
Average	0,3022	0,2922	0,2791	0,2868	0,295	0,2910

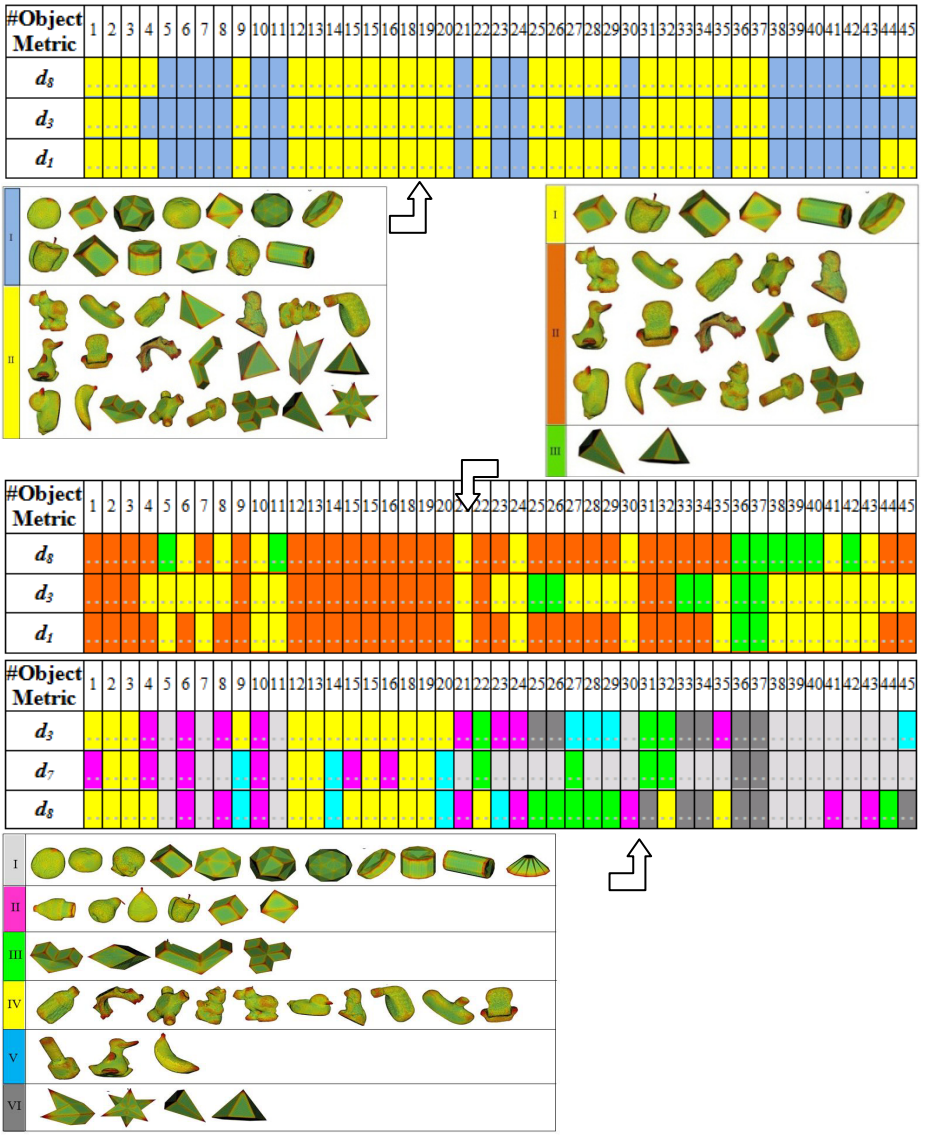

Fig. 2. Incomplete clustering results. *Hard coincidence* groups for two and three clusters and *2-Soft coincidence* groups for six clusters.

(d_1, d_3, d_8), (d_1, d_3, d_8) and (d_3, d_7, d_8). Note that the metric d_4 was not selected in cases $w_1 = 2$ and $w_2 = 3$ for not verifying the unitary cluster criterion. Thus, in the experimentation presented in this paper, *hard coincidence* cases occur when all three metrics classify the object in the same cluster. As regards *soft coincidence*, it is reduced to the case *j=2*.

Figure 2 shows the mesh models of the objects belonging to each group. The visual representation provides a qualitative evaluation of the curvature in the color scale which goes from yellow (low values) to red (high values). This color representation is used in order to better argue the discussion concerning the global geometry properties of each of the groups. The groups are also identified by colors in the tables introduced. Note that our method is able to provide coherent-incomplete clustering results after establishing a voting-consensus between metrics. Several comments follow.

In the case w=2 (first colored table)), note that the models in Group I have large flat zones and small high-curvature areas, signifying that parameter c in a large part of the model is low and is only high in a few nodes. Also note that there is a slight variation of the parameter r in the majority of the objects. Group II contains objects with high-curvature nodes which have disparate r values and, in general, the curvature distribution is more uniform than in Group I. Similar comments can be made for case w=3 (second table)) with regards Groups I and II. The objects in Group I have large flat and small high-curvature areas, whereas Group II contains objects which have disparate r values and, in general a more uniform curvature distribution than in Group I. Group III is composed solely of two pyramids. For six clusters (w=6) see the last table. In all groups it is possible to find a common property of the objects which makes these results compelling. For example, Group I is composed of more or less round objects, Group II contains round objects with some kind of tip or stalk, Group III contains parallelepide shapes, Group IV consists of typical smooth free form shapes which are extended to Group V but including pointed zones on the objects. Finally, all the objects in Group VI are polyhedral with extremely sharp areas.

5 Conclusions

In this paper we present a new approach to generate incomplete but reliable clusters. The method has been applied to cluster 3D objects through their mesh model representations. The clustering process is carried out under a consensus algorithm in which a set of metrics are considered for each particular database. In the classification process, the goodness of each metric is globally evaluated through the *clustering confidence measure.* The consensus algorithm finally establishes the so-called *hard* and *soft coincidence* parameters inside each of the groups, and decides which objects are definitively classified from different metrics.

This clustering approach has successfully been tested on a set of mesh models belonging to a wide variety of objects. The results proved in all those cases in which *hard* or *soft coincidences* between metrics were considered, very coherent clusters were obtained.

The idea of recovering coherent 3D shape groups using a consensus between different similarity metrics is interesting and may provide promising results in the future. Unfortunately, it is difficult to evaluate the quality of our results in comparison to other methods, principally owing to the lack of similar approaches and to the fact that it depends largely on the details of our test database. Our future lines of work are focused on labeling reliable groups of objects using extended databases with the aim of discovering the applicability of our method in a real environment. The goal in this

case is to extract, from a large set of objects, specific subgroups of objects with a clear particularity. This will certainly be our objective in the future.

Acknowledgments. This research has been supported by the Spanish DPI2009-14024-C02-01 project.

References

1. Tan, P., Steinbach, M., Kumar, V.: Introduction to Data Mining. Addison Wesley (2005)
2. Shi, J., Malik, J.: Normalized cuts and image segmentation. IEEE Trans. Pattern Analysis and Machine Intelligence 22(8), 888–905 (2000)
3. Ertoz, L., Steinbach, M., Kumar, V.: Finding clusters of different sizes, shapes, and densities in noisy, high dimensional data. In: Proceedings of SDM 2003, SIAM Int'l Conf on Data Mining, San Francisco, CA (2003)
4. Basu, S., Bilenko, M., Mooney, R.: A probabilistic framework for semi-supervised clustering. In: Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Seatle, WA (2004)
5. Davidson, I., Ravi, S.: Clustering with constraints: feasibility issues and the k-means algorithm. In: Proceedings of SDM 2006: SIAM International Conference on Data Mining, Newport Beach, CA (2005)
6. Cheng, Z., Zhou, D., Wang, C., Guo, J., Wang, W., Ding, B., Shi, B.-L.: CLINCH: Clustering Incomplete High-Dimensional Data for Data Mining Application. In: Zhang, Y., Tanaka, K., Yu, J.X., Wang, S., Li, M. (eds.) APWeb 2005. LNCS, vol. 3399, pp. 88–99. Springer, Heidelberg (2005)
7. Hathaway, R.J., Bezdek, J.C.: Fuzzy c-means clustering of incomplete data. IEEE Trans. System Man and Cybernetic Part B 31(5), 735–744 (2001)
8. Zhang, D.-Q., Chen, S.-C.: Clustering incomplete data using kernel-based fuzzy c-means Algorithm. Neural Processing Letters 18(3), 155–162 (2003)
9. Himmelspach, L., Conrad, S.: Fuzzy clustering of incomplete data based on cluster dispersion. In: 13th International Conference on Information Processing and Management of Uncertainty
10. Jayanti, S., Kalyanaraman, Y., Ramani, K.: Shape-based clustering for 3D CAD objects: A comparative study of effectiveness. In: Computer-Aided Design, vol. 41(12), pp. 999–1007 (2009)
11. Chakraborty, T.: Shape-based Clustering of Enterprise CAD Databases. Computer-Aided Design & Applications 2(1-4), 145–154 (2005)
12. Ward, J.H.: Hierarchical Grouping to Optimize an Objective Function. Journal of the American Statistical Association 58(301), 236–244 (1963)
13. Adán, A., Adán, M.: Incomplete-Clustering Consensus Strategy Using RC-images. Pattern Recognition. 3DVC&R, UCLM Technical Report (2012)