INVARIANT CATEGORISATION OF POLYGONAL OBJECTS USING MULTI-RESOLUTION SIGNATURES

Roberto Lam

Instituto Superior de Engenharia, Universidade do Algarve, Campus da Penha, Faro, Portugal

J. M. Hans du Buf

ISR-Vision Laboratory, FCT, Universidade do Algarve, Campus de Gambelas, Faro, Portugal

Keywords: 3D Shape matching, Volumetric models, Manifold meshes.

Abstract: With the increasing use of 3D objects and models, mining of 3D databases is becoming an important issue. However, 3D object recognition is very time consuming because of variations due to position, rotation, size and mesh resolution. A fast categorisation can be used to discard non-similar objects, such that only few objects need to be compared in full detail. We present a simple method for characterising 3D objects with the goal of performing a fast similarity search in a set of polygonal mesh models. The method constructs, for each object, two sets of multi-scale signatures: (a) the progression of deformation due to iterative mesh smoothing and, similarly, (b) the influence of mesh dilation and erosion using a sphere with increasing radius. The signatures are invariant to 3D translation, rotation and scaling, also to mesh resolution because of proper normalisation. The method was validated on a set of 31 complex objects, each object being represented with three mesh resolutions. The results were measured in terms of Euclidian distance for ranking all objects, with an overall average ranking rate of 1.29.

1 INTRODUCTION AND RELATED WORK

One might say that technological developments will lead us towards using increasingly complex illustrations, i.e., moving from 2D to 3D space. There are digital scanners which produce 3D models of real objects. CAD software can also produce 3D models, from complex pieces of machinery with lots of corners and edges to smooth sculptures. Very complex protein structures play an important role in pharmacology and related medical areas. Many actors in the World Wide Web have started to incorporate 3D models in sites and home pages. As a consequence of this trend, there is a strong interest in methods for 3D similarity analysis (Bustos et al., 2005; Tangelder and Veltkamp, 2007). Similarity analysis is a fast way to discard many irrelevant objects from a database, i.e., before precise object recognition (matching) which may be very time consuming because of all variations that may occur: different position (object origin), rotation, size and also mesh resolution.

Similarity analysis does not require precise shape comparisons, global nor local. Instead, this approach is based on computing sets of features (FV or feature vector) of a query object and comparing its FV with all FVs of known objects in a database. The FVs can be obtained by a variety of methods, from very simple ones (bounding box, area-volume ratio, eccentricity) to very complex ones (curvature distribution of sliced volume, spherical harmonics, 3D Fourier coefficients) (Saupe and Vranic, 2001; Pang et al., 2006; Sijbers and Dyck, 2002). We mention two approaches which are related to our own approach. Assfalg et al. (2006) projected a 3D object onto 2D curvature maps. This is preceded by smoothing and simplification of the polygonal mesh, and final retrieval is based on comparing the 2D curvature maps. Chuang et al. (1991) and Suzuki (2007) used the fractal dimension for characterising 3D objects.

The intrinsic nature of the objects may pose some constraints, and some methods may be more

suitable—and faster—for the extraction of FVs than others. For example, methods based on spherical harmonics and 3D Fourier coefficients are not suitable for concave objects (non-star-shaped), whereas other methods have problems with open (non-closed) objects. Some limitations can be solved by combining two or more methods. However, since many objects can yield very similar FVs by applying only one method, i.e., mathematically possibly an infinite number of objects, normally several methods are combined to achieve the best results.

2 OVERVIEW OF OUR APPROACH

We use a set of 31 models, each one represented with four different mesh resolutions. The models were selected from the (AIM@SHAPE, 2008) database. This database has high-definition objects which can be converted to other mesh resolutions by means of one parameter between 9.9 (max mesh size) and 5.5 (min mesh size). The models were downloaded in PLY format and only "watertight" ones—closed, without gaps and with regular meshes—were selected. Figure 1 shows a few examples, and Table 1 a list of all objects with their mesh resolutions: the first three resolutions are used for creating the training set FVs, the fourth one as test object for similarity search.

In order to obtain invariance to translation and scale (size), the models were normalised to the unitary sphere (radius 1.0) after the origin of the models was translated to the center of the sphere. Rotation invariance is achieved by the fact that our FV is global to the model as proven in (Vranić, 2004). Invariance to mesh resolution is obtained by proper feature normalisations, which will be explained below. We apply two different but complementary methods in order to generate two kinds of features for object retrieval. These are based on mesh smoothing (Section 2.1) and on dilation and erosion (Section 2.2).

2.1 Mesh Smoothing

Mesh smoothing serves to reduce noise, for example for decreasing the mesh size by re-triangulation of planar areas. (Glendinning and Herbert, 2003) used smoothing of principal components for shape classification in 2D. Here, the idea is related to iterative and adaptive (nonlinear) mesh smoothing in 3D, i.e., smoothing in quasi-planar regions but not at sharp edges (Lam et al., 2001). However, here we simply apply the *linear* version which will smooth the mesh

Table 1: All 31 models with their mesh resolutions, the last resolution was used in similarity search.

N	Model	Resolutions
1	Blade	6.5; 7.5; 9.9; 8.0
2	Bimba	6.0; 8.5; 9.5; 8.0
3	Block	5.0; 6.5; 8.0; 8.5
4	Bunny	6.5; 7.5; 9.9; 8.0
5	Cow	6.0; 6.4; 9.9; 7.1
6	Cow2	6.0; 7.5; 9.9; 8.9
7	DancingChildren	6.0; 7.5; 9.9; 6.8
8	Dragon	6.0; 8.0; 9.5; 7.7
9	Duck	6.0; 7.5; 9.9; 6.7
10	Eros	6.0; 7.5; 9.9; 6.5
11	Fish	6.0; 7.5; 8.0; 8.0
12	FishA	6.0; 7.5; 9.9; 7.0
13	GreekSculpture	6.5; 7.0; 7.7; 8.5
14	IsidoreHorse	6.0; 7.5; 9.9; 7.0
15	Mouse	6.0; 7.5; 9.9; 7.8
16	Pulley	6.0; 7.5; 9.9; 7.0
17	Torso	6.0; 7.5; 9.9; 7.7
18	CamelA	6.0; 7.5; 9.9; 7.8
19	Carter	6.0; 8.5; 9.5; 7.3
20	Chair	6.5; 7.5; 9.9; 6.9
21	Dancer	6.0; 7.5; 99; 7.7
22	Dente	6.0; 7.5; 9.9; 7.0
23	Elk	6.0; 7.5; 9.9; 7.9
24	Grayloc	6.0; 7.5; 9.9; 7.8
25	Horse	6.0; 7.5; 9.9; 8.0
26	Kitten	6.0; 7.5; 9.9; 7.3
27	Lion-dog	6.0; 7.5; 9.9; 8.0
28	Neptune	6.0; 8.0; 9.5; 7.6
29	Ramesses	6.0; 7.5; 9.9; 8.0
30	Rocker	6.0; 7.5; 9.9; 7.1
31	Squirrel	6.0; 7.5; 9.9; 7.2

at *all* vertices: it starts by eliminating very sharp object details like in- and protruding dents and bumps, and then, after more iterations, less sharp details. The sum of the displacements of all vertices, in combination with the contraction ratio of the surface area, generates a quadratic function which can characterise the model quite well.

If $V_i, i = 1, N$, is the object's vertex list with associated coordinates (x_i, y_i, z_i), the triangle list $T(V)$ can be used to determine the vertices at a distance of one, i.e., all direct neighbour vertices connected to V_i by only one triangle edge. If all neighbour vertices of V_i are $V_{i,j}, j = 1, n$, the centroid of the neighbourhood is obtained by $\bar{V}_i = 1/n \sum_{j=1}^{n} V_{i,j}$. Each vertex V_i is moved to $\bar{V}_i$, with displacement $\bar{D}_i = ||V_i - \bar{V}_i||$. The total displacement is $D = \sum_{i=1}^{N} \bar{D}_i$. The entire procedure is repeated 10 times, because we are mainly interested in the deformation of the object at
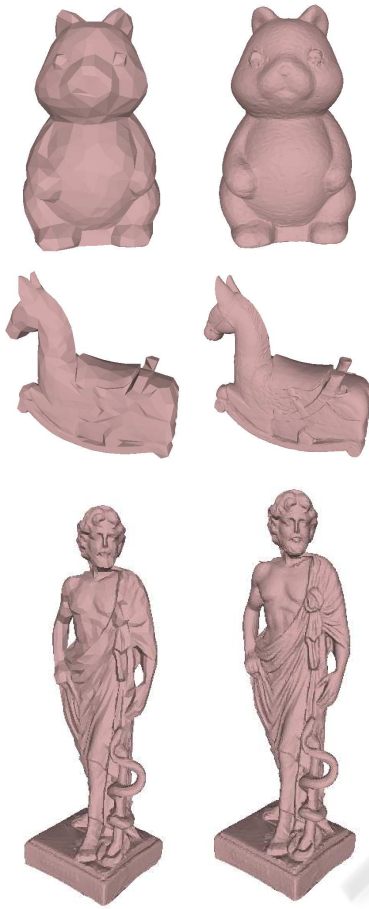
Figure 1: Examples of models: Squirrel (top), IsidoreHorse (middle) and GreekSculpture (bottom). Low resolutions at left and high ones at right.

the start, when there still are many object details, and more iterations do not add useful information anymore. Hence, displacements are accumulated by $A_l = \sum_{m=1}^{l} D_m$ with $m = 1...10$. In order to obtain invariance to mesh size, in each iteration m the displacement D_m is corrected using

$$D_m := D_m \cdot \frac{NP_m \cdot N}{A_{10} \cdot S_m}, \qquad (1)$$

with N the total number of vertices, NP_m the number of participating vertices (in non-planar regions which contributed to the displacement), S_m the surface of the object (sum of all triangles) after the smoothing step, and A_{10} the final, maximum accumulated displacement after all 10 iterations. Then the curve of each object and each mesh resolution is further normalised by the total contraction ratio defined by S_{10}/S_0 (final surface and original surface), and the three curves (10 data points) are averaged over the three mesh resolutions. In the last step, the averaged A_l is least-squares approximated by a quadratic polynomial in order to

reduce 10 parameters to 3. Figure 3 shows representative examples of curves A_l. It should be stressed that, in contrast to the second method as described below, no re-triangulation of the mesh of the object after each iteration is done, i.e., the number of vertices—and triangles—remains the same. Figure 2 shows a model and the influence of mesh smoothing.

2.2 Mesh Dilation and Erosion

The second method is based on the estimation of the 3D fractal dimension by applying a sphere as structural element with increasing radius. A sphere is applied to the model, its origin being placed at each vertex. This yields two surfaces: the dilated surface grows and the eroded one shrinks as a function of sphere radius, both showing less object detail. Instead of computing the fractal dimension, we compute the volume between the two surfaces as a function of sphere radius in order to obtain characteristic curves, like the ones described in the previous section, for characterising the objects. The growth (increasing radius) of the sphere is related to the mesh resolution of the model. Therefore, before computing the volumes, we eliminate vertices which are inside the neighbourhood defined by the sphere with the radius used in the dilation-erosion process. If ΔL is the difference between the maximum and minimum edge lengths of a model, i.e., $\Delta L = L_{\max} - L_{\min}$, then $\Delta R = 0.05\Delta L$. Hence, the radius at iteration m is $R_m = m\Delta R$, which results in volumes V_0 (the volume of the original model) and V_m (the volume between dilated and eroded models at iteration m). For obtaining invariance to mesh size, we apply

$$V_m := V_m \cdot \frac{V_0}{R_m} \quad ; \quad m = 1, 2, ... \qquad (2)$$

As can be seen in Fig. 4, the dilation-erosion curves are quite similar, for different mesh resolutions, at the start of the process, but then start to diverge when the radius becomes too big and noise is introduced to the model. Therefore we averaged the V_m of the three mesh resolutions and only included in the FV two parameters: V_0 and V_2. Figure 5 shows the Bimba model with erosion and dilation.

3 RESULTS AND DISCUSSION

The 31 models listed in Table 1 were used, each with four mesh resolutions. As explained before, the first three mesh resolutions were used for constructing the FV of the model, and the last one was used for testing. Each model was characterised by
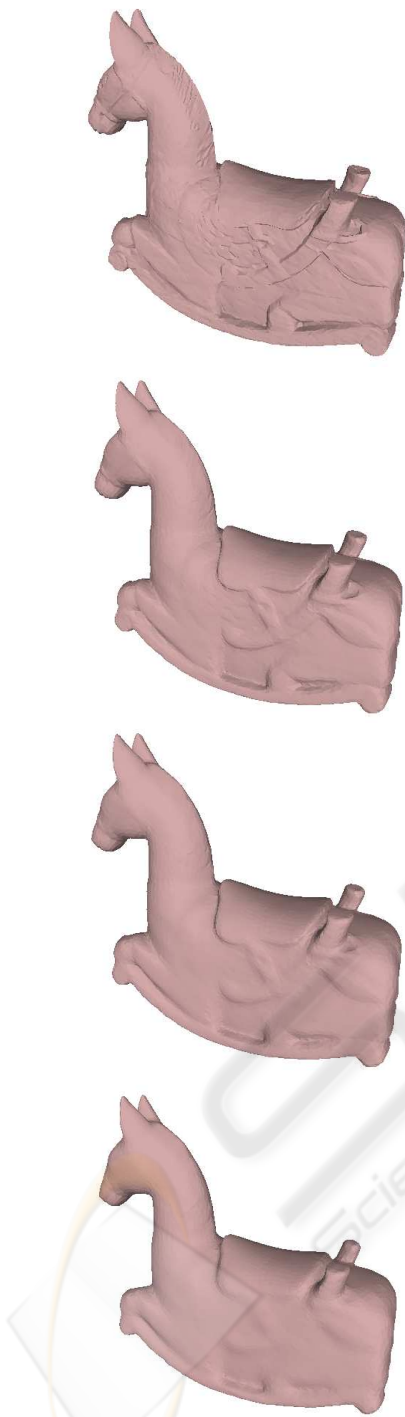
Figure 2: Mesh smoothing applied to IsidoreHorse. From top to bottom: original and smoothed meshes after 3, 6 and 10 iterations.

7 parameters, 5 from the method described in Section 2.1 (surface of original model after normalisation to unit sphere; contraction ratio after 10 iterations; 3 coefficients of the quadratic approximation of the
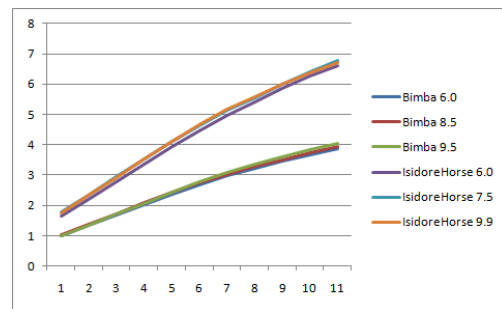

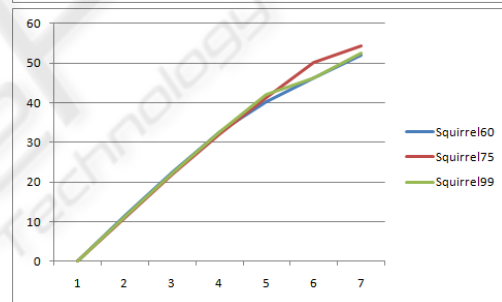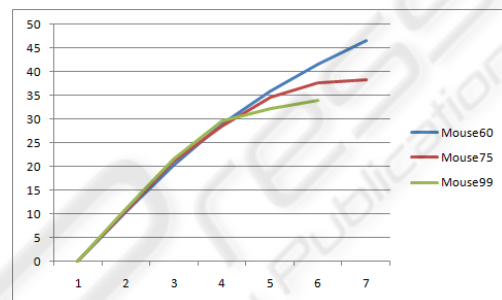Figure 3: Characteristic curves from mesh smoothing of the Bimba and IsidoreHorse models.

Figure 4: Characteristic curves from mesh dilation and erosion of the Mouse and Squirrel models.

smoothing curves), and 2 from Section 2.2 (volume of original model after normalisation to unit sphere; volume between dilated-eroded surfaces after 2 iterations). The FVs of the objects' test resolutions were compared with all FVs of the database, and the objects were sorted by using the Euclidean distance between the FVs. Table 2 lists the results, starting with the object with the smallest distance, then the object with the next smallest distance, and so forth, until the fifth object. Table 2 shows that in 26 of 31 cases the correct object was ranked first. Similar objects (IsidoreHorse, 14; CamelA, 18; Horse, 25) were always ranked at position 1 to 3. The average ranking rate $R = (1/31)\sum_{i=1}^{31} P_i$, where P_i is the ranked position of object i, is 1.29. This means that the majority of objects is ranked at position 1 or 2, at least within the first 3 or 5 positions for narrowing a full object comparison in a big and complex database. It should
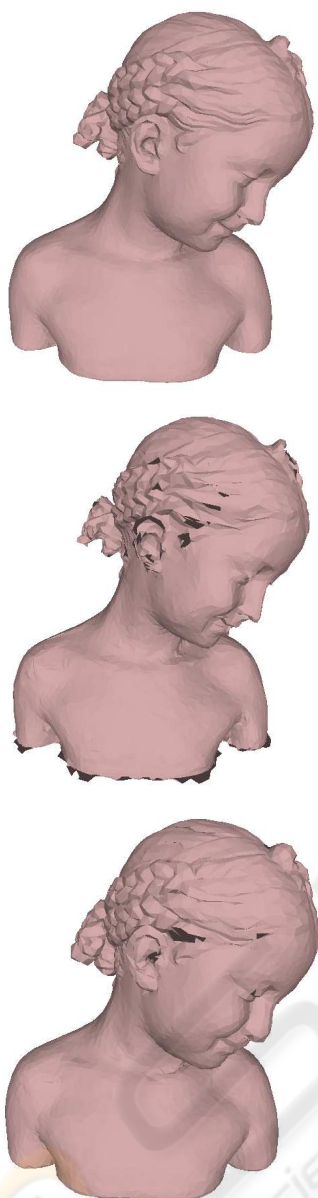
Figure 5: Mesh dilation and erosion applied to the Bimba model. From top to bottom: original plus eroded and dilated models after 4 iterations.

be stressed that, although 26 objects were ranked first, this does not mean that the correct object has been identified. The most similar object may have been detected, but in real conditions, i.e., with big and complex databases like protein structures, the search has been narrowed in order to save time for detailed object comparisons. Future work involves improving further the methods described in this paper, but in relation to detailed object comparisons which may provide additional parameter models for a similarity search. Most important is to improve the erosion-dilation method

such that identical curves are obtained for different object resolutions at larger sphere radii, after which the relative importance of the individual features and feature sets can be studied.

Table 2: Results.

N	Testing Model	Ordered output
1	Blade	**1**-6-30-22-12
2	Bimba	**2**-26-31-22-15
3	Block	**3**-27-9-15-31
4	Bunny	**4**-7-23-14-25
5	Cow	12-**5**-6-17-11
6	Cow2	**6**-12-30-22-1
7	DancingChildren	**7**-4-23-14-10
8	Dragon	**8**-31-23-9-2
9	Duck	**9**-31-15-8-2
10	Eros	18-25-**10**-14-26
11	Fish	**11**-6-12-1-30
12	FishA	**12**-6-1-5-11
13	GreekSculpture	**13**-28-20-17-5
14	IsidoreHorse	**14**-25-18-10-26
15	Mouse	**15**-31-9-22-2
16	Pulley	**16**-24-19-3-9
17	Torso	**17**-5-12-20-6
18	CamelA	25-**18**-14-10-26
19	Carter	**19**-24-16-8-9
20	Chair	**20**-17-5-12-1
21	Dancer	11-12-5-6-**21**
22	Dente	**22**-26-30-2-6
23	Elk	**23**-7-4-14-8
24	Grayloc	**24**-19-16-8-9
25	Horse	**25**-14-18-10-26
26	Kitten	**26**-2-22-10-30
27	Lion-dog	**27**-3-9-15-17
28	Neptune	**28**-21-5-20-17
29	Ramesses	**29**-10-14-4-7
30	Rocker	22-**30**-26-2-6
31	Squirrel	**31**-15-9-2-22

ACKNOWLEDGEMENTS

REFERENCES

AIM@SHAPE. (2008). http://www.aimatshape.net

Assfalg, J., Del Bimbo, A. and Pala, P. (2006). Content-based retrieval of 3D models through curvature maps: a CBR approach exploiting media conversion. *Multimedia Tools Appl.*, 31(1):29–50.

Bustos, B., Keim, D. A., Saupe, D., Schreck, T. and Vranić, D.V. (2005). Feature-based similarity search in 3D object databases. *ACM Comput. Surv.*, 37(4):345–387.

Chuang, K., Valentino, D. and Huang, H. (1991). Measurement of fractal dimension using 3-D technique. *Proc. SPIE*, Vol 1445,:341-34.

Glendinning, R.H. and Herbert, R.A. (2003). Shape classification using smooth principal components. *Pattern Recognition Letters*, 24(12):2021-2030.

Lam, R., Loke, R. and du Buf, H. (2001). Smoothing and reduction of triangle meshes. *Proc. 10th Portuguese Computer Graphics Meeting*, 97–105.

Pang, M., Wenjun, D., Gangshan, W. and Fuyan, Z. (2006). On volume distribution features based 3D model retrieval. *Proc. Advances in Artificial Reality and Tele-Existence ICAT*, LNCS Vol. 4282, 928–937.

Saupe, D. and Vranić, D.V. (2001). 3D model retrieval with spherical harmonics and moments. *Proc. DAGM Symposium*, 392–397.

Sijbers, J. and Dyck, D.V. (2002). Efficient algorithm for the computation of 3D Fourier descriptors. *Proc. Int. Symp. 3D Data Processing Visualization and Transmission*, 640–643.

Suzuki, M. (2007). A three dimensional box counting method for measuring fractal dimension of 3D models. *Proc. 11th IASTED Int. Conf. on Internet and Multimedia Systems and Applications*, PID:577-082.

Tangelder, J. and Veltkamp, R. (2007). A survey of content based 3D shape retrieval methods. *Multimedia Tools and Applications*, 441–471.

Vranić D.V. (2004). 3D Model Retrieval. *Ph.D. Thesis* University of Leipzig.