SHape REtrieval Contest 2008: Stability Track on Watertight Models

Silvia Biasotti* Marco Attene†
IMATI CNR, Dept. of Genova,
Via De Marini 6, 16149, Genova, Italy

March 31, 2008

Abstract

This contribution reports the results of the SHREC 2008 track on *Stability on Watertight Models*. This track saw six registrations of which only three participants effectively sent the results of their runs.

1 Introduction

A major barrier to a widespread adoption of 3D retrieval techniques in both commercial and academic systems is the lack of a standardized evaluation of the methods. What is the best shape characterization or the best similarity measure for a given domain? The answer is not trivial at all and depends on several factors. The aim of SHREC is to evaluate the performance of existing 3D shape retrieval algorithms, by highlighting their strengths and weaknesses, using a common test collection that allows for a direct comparison of methods. After the first successful experience of SHREC 2006, from 2007 the contest has moved towards a multi-track organization, in which different datasets are used to target different retrieval contexts. In this report we present the results of the *Stability on Watertight Models Track*, whose aim is to evaluate the stability of algorithms with respect to input perturbations that modify the representation of the object without changing its overall shape significantly. Examples of such perturbations include geometric noise, varying sampling patterns, small shape deformations and topological noise.

2 Data Collection and Queries

Two data collections have been provided with this track. Both collections are made of watertight triangle models in which various kinds of perturbations were introduced. Two sets of models A and B were provided, the set B containing the models in A. More in detail, the set B is made of 15 classes of 100 models each, for a total of 1500 models; A contains 1229 models (all the models in B after having excluded the 271 models with self-intersections).

The set B has been generated as follows. Among the 20 classes used in the SHREC07 track *Watertight models* [2], we have selected 15 classes, namely *humans*, *cups*, *glasses*, *airplanes*, *chairs*, *octopuses*, *tables*, *hands*, *fishes*, *birds*, *springs*, *armadillos*, *busts*, *mechanical parts*, *four leg animals* (see Figure 1); then, we perturbed the 20 models in each class with additive Gaussian noise, uneven re-sampling, small protrusions, and topological noise (see an example in Figure 2). At the end, each class of the dataset B was made of of 100 models.

The dataset A was obtained removing from B the elements with self-intersections. A command-line version of the ReMESH software [1] was used to perturb the models and to detect self-intersections.

*e-mail: silvia.biasotti@ge.imati.cnr.it
†e-mail:marco.attene@ge.imati.cnr.it

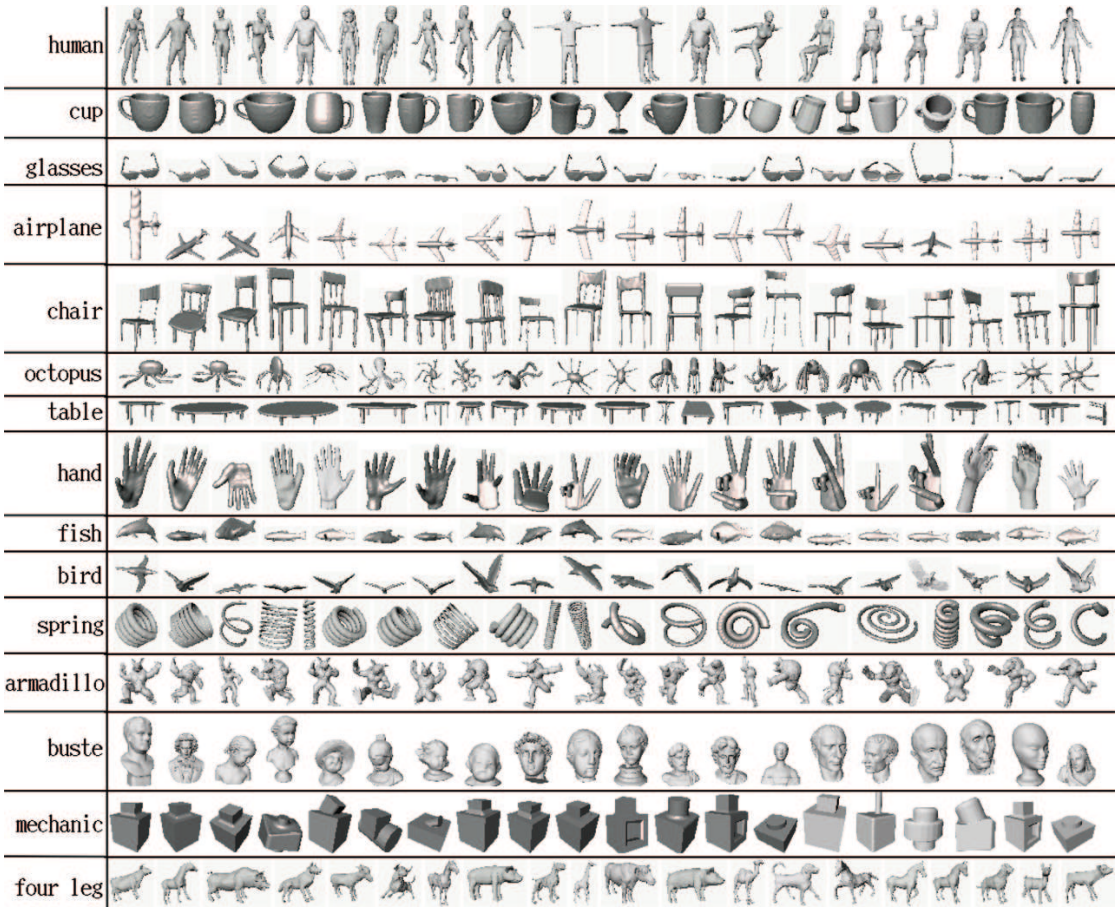


Figure 1: The set of original models used to create the datasets A and B.

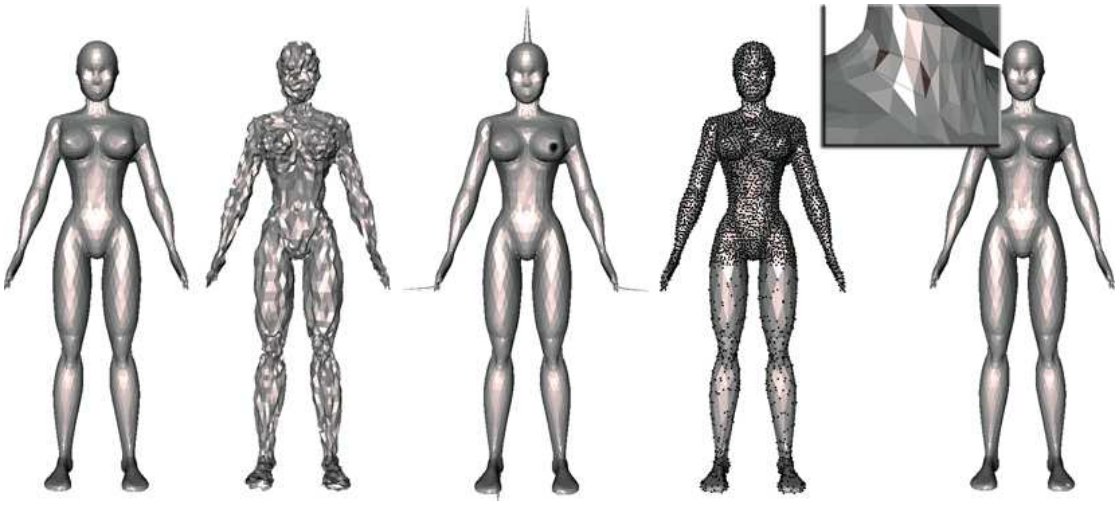


Figure 2: (a) A model of the database [2] and its perturbations: (b) Gaussian noise, (c) small protrusions, (d) uneven re-sampling and (e) adding topological noise.

Each model was used in turn as a query against the remaining part of the database. For a given query, the goal of the track is to retrieve the most similar objects. The relevance, marginal relevance or non-relevance of the models for a given query, i.e. the ground truth, was established a priori by two classification schemes. The performances of the algorithms have been evaluated using the measures and tools described in section 5.

3 Participants

Each participant was asked to submit up to 3 runs of his/her algorithm, in the form of dissimilarity matrices; each run could be for example the result of a different setting of parameters or the use of a different similarity metric. We remind that the entry (i, j) of a dissimilarity matrix represents the distance between models i and j.

This track included 3 groups of participants:

1. Tony Tung and Francis Schmitt with 3 matrices;
2. Thibault Napolon, Tomasz Adamek, Francis Schmitt and Noel E. OConnor with 2 matrices;
3. Dong Xu, Li Cui, Ping Hu, Weiguo Cao and Hua Li, with 3 matrices.

For details on the algorithms and the different runs proposed by the participants, the reader is referred to their papers, included at the end of this report.

In addition to the three groups of participants listed above, three further registrations to the track were received from Indriyati Atmosukarto (University of Washington, USA), Julien Tierny (Telecom Lille 1, France) and Ryutarou Ohbuchi (University of Yamanashi, Japan). These additional participants withdrew the track.

4 Performance Measures

The performance of the methods on the dataset B has been evaluated by considering two different levels of ground truth. The first classification (coarser) considers in the same class the models in the original class and their perturbations, that is, each class is made of the 20 original models plus their four perturbations so that a total of 100 elements per each class was reached. The second classification (finer) considers in the same class just a single model and its perturbations, that is, each class is made of 5 models: 1 original model plus its four perturbed versions. Then, this classification subdivides the dataset in 300 classes of five elements.

The two schemes correspond to two possible interpretations of the stability of the methods: in the first case we evaluate how much the models and their perturbations are still recognized to belong to the original class while in the second case the attention is on the model and its perturbations rather than to the other models in the same original class.

As performance measures of the method we have adopted the **precision** and **recall**, that are two fundamental measures often used in evaluating search strategies. Recall is the ratio of the number of relevant records retrieved to the total number of relevant records in the database, while precision is the ratio of the number of relevant records retrieved to the size of the return vector [3].

In our contest, for each query the total number of relevant records in the database is 100 for the coarser classification and 5 for the finer one, that is the size of each class. Starting from here, we evaluate the precision-recall measures for each query, and then average it over each class and over the entire database.

Recall and precision are represented in a diagram, where precision has been computed as average of the precision scores after each relevant item in the scope. Finally, we consider the area under the diagrams which is relevant to evaluate the overall performance of a method.

5 Results and Discussions

Each participant sent two or three matrices corresponding to different choices of the parameters. A general observation is that the performances of each method do not vary significantly across its parameter settings; hence, it makes sense to consider the best run for each method and compare the methods according to such best runs. For each method, the best run was selected as the one with the maximum area under the precision-recall diagram. For completeness, however, precision-recall diagrams are also depicted all together in a single graphical panel.

In all the cases, precision-recall curves shifted upwards and towards the right indicate a superior performance; in a number, the performance can be roughly expressed as the area under the graph.

5.1 Performance on the dataset B

Figure 3 shows the recall precision diagrams obtained using the coarse classification of the dataset, i.e., the original models of a class and their perturbations are considered in the same class. Figure 4 shows the recall precision diagrams obtained using the fine classification of the dataset, i.e., a single class of models is made of the original model and its four perturbations. Finally, Figure 5 details, for each participant, the results reported in the Figures 3 and 4.

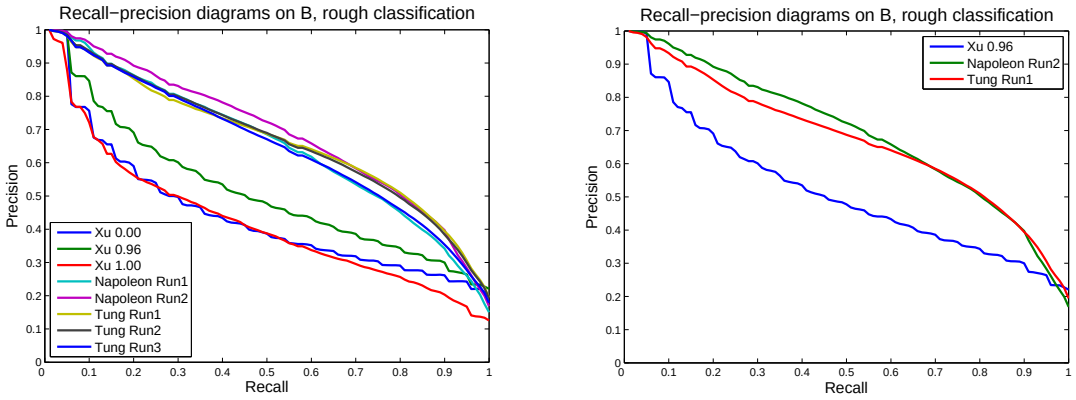


Figure 3: Comparison of the best final recall precision graphs of each participant over the coarser classification. Left: all the runs. Right: best runs only.

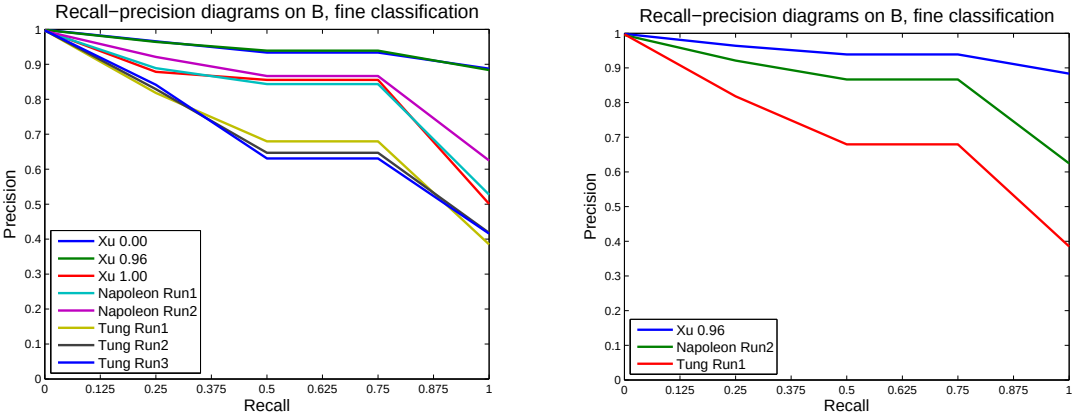
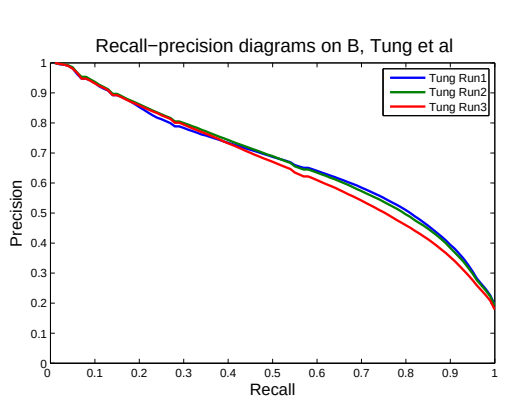


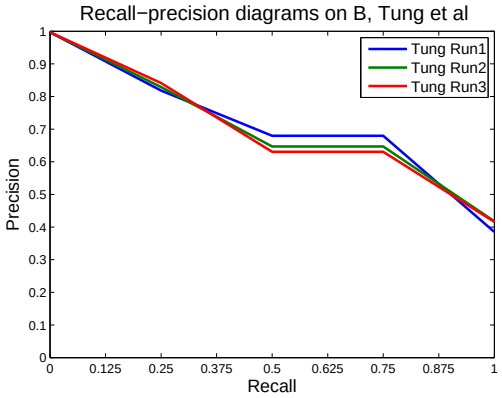
Figure 4: Comparison of the best final recall precision graphs of each participant over the finer classification. Left: all the runs. Right: best runs only.

Interestingly, there is no method that performs better than the others in all the conditions. Specifically, the method by Xu et al. seems to be the less performant within the coarse classification, while it jumps to the first position in the fine classification. On the contrary, a significant improvement of the performances can be observed for the methods by Tung et al. and Napoleon et al. when moving from the fine to the coarse classification.

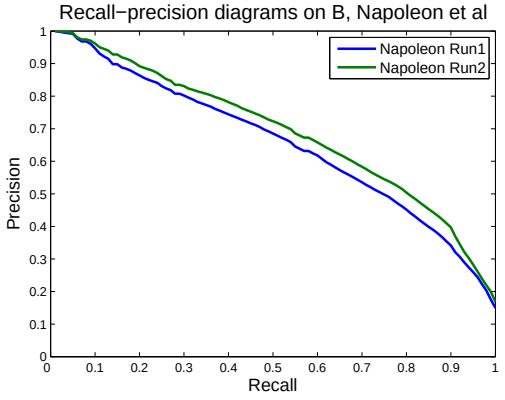
In order to assess the various methods thoroughly, we have also studied the impact of the various kinds of perturbation on the performances of each method. To do this, we have evaluated the retrieval performances of the methods when the original models are used as queries against one perturbation at a time and when the models obtained using a single perturbation are used as queries against theirselves.



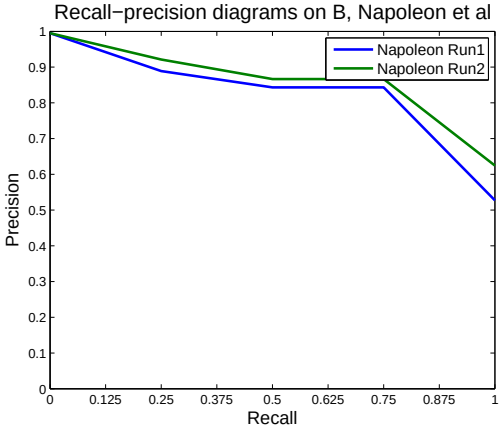
(a)



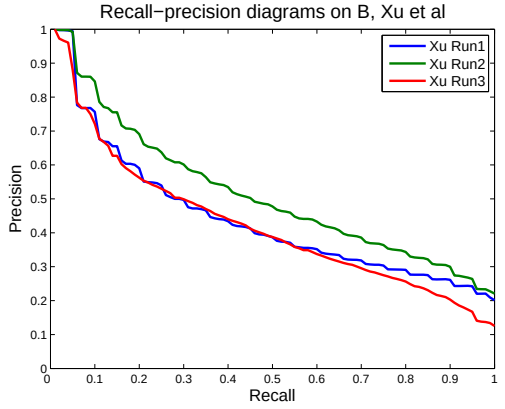
(b)



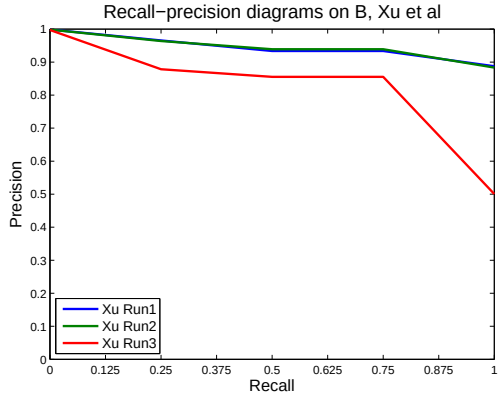
(c)



(d)



(e)



(f)

Figure 5: From top to bottom, performances on the dataset B of Tung et al. (a,b); Napoleon et al. (c,d) and Xu et al. (e,f) with respect to the coarse (left) and the fine (right) classification.

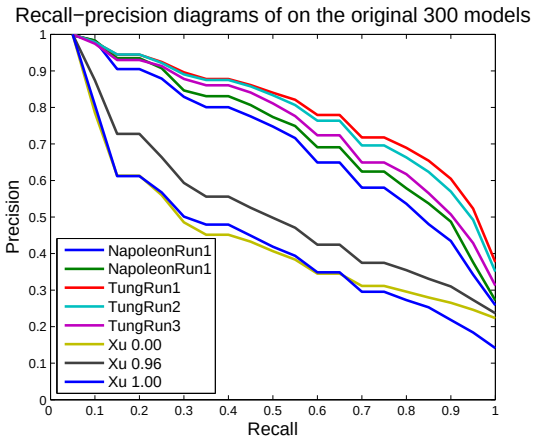


Figure 6: Performance of the various methods over the dataset without perturbations (20 original models per class).

Degradation of the retrieval performance				
Method	*Guassian Noise*	*Small Protrusions*	*Topological Noise*	*Uneven Re-sampling*
Tung et al., *Run1*	47.92%	34.14%	44.68%	39.44%
Tung et al., *Run2*	47.42%	35.90%	44.46%	39.46%
Tung et al., *Run3*	46.43%	36.75%	44.32%	38.90%
Napoleon et al., *Run1*	44.96%	37.50%	47.83%	36.54%
Napoleon et al., *Run2*	48.73%	41.35%	55.75%	37.34%
Xu et al, *0.00*	30.48%	33.86%	41.85%	24.67%
Xu et al, *0.96*	36.11%	35.08%	43.27%	29.35%
Xu et al, *1.00*	44.75%	41.28%	40.29%	31.99%

Table 1: The same type of pertubed models are used both as queries and dataset, each class is made of 20 elements.

For each method, a precision-recall graph was tracked starting from the results on the original models only (classes of 20 elements, each element used in turn as a query). Then, a second graph was tracked starting from the results on the models deriving from a single perturbation (again, classes of 20 elements, all with the same kind of perturbation). When comparing the second graph with the first one for the same method, the loss of area (as a percentage) represents the degradation of the method (see Table 1) when both the queries and the dataset are perturbed.

Figures 7 and 8 show the recall precision diagrams of the runs of the three algorithms that participated to the track. In this case, the method by Tung et al. seems to be the most stable on average.

Also, a third graph was tracked by comparing original models with perturbed models (here the queries are not perturbed, while the dataset is made of classes of 20 elements with the same kind of perturbation). Once again, when comparing the third graph with the first one for the same method, the loss of area represents the degradation of the method (see Table 2) when only the dataset is perturbed. This analysis reveals that the method by Tung et al. degrades more when the dataset contains Gaussian noise, while it is less sensitive to the presence of small protrusions. Differently, the method by Napoleon et al. degrades when the models have topological noise, while it is less sensitive to unbalanced samplings of the surface. Finally, the method by Xu appears to be rather stable to unbalanced sampling patterns, while it degrades a little bit more when topological noise occurs.

5.2 Performance on the dataset A

Similar tests to those presented in Section 5.1 for the complete dataset B, have been performed for the smaller dataset A, where models with self-intersections were removed. Figure 9 depicts the recall precision diagrams

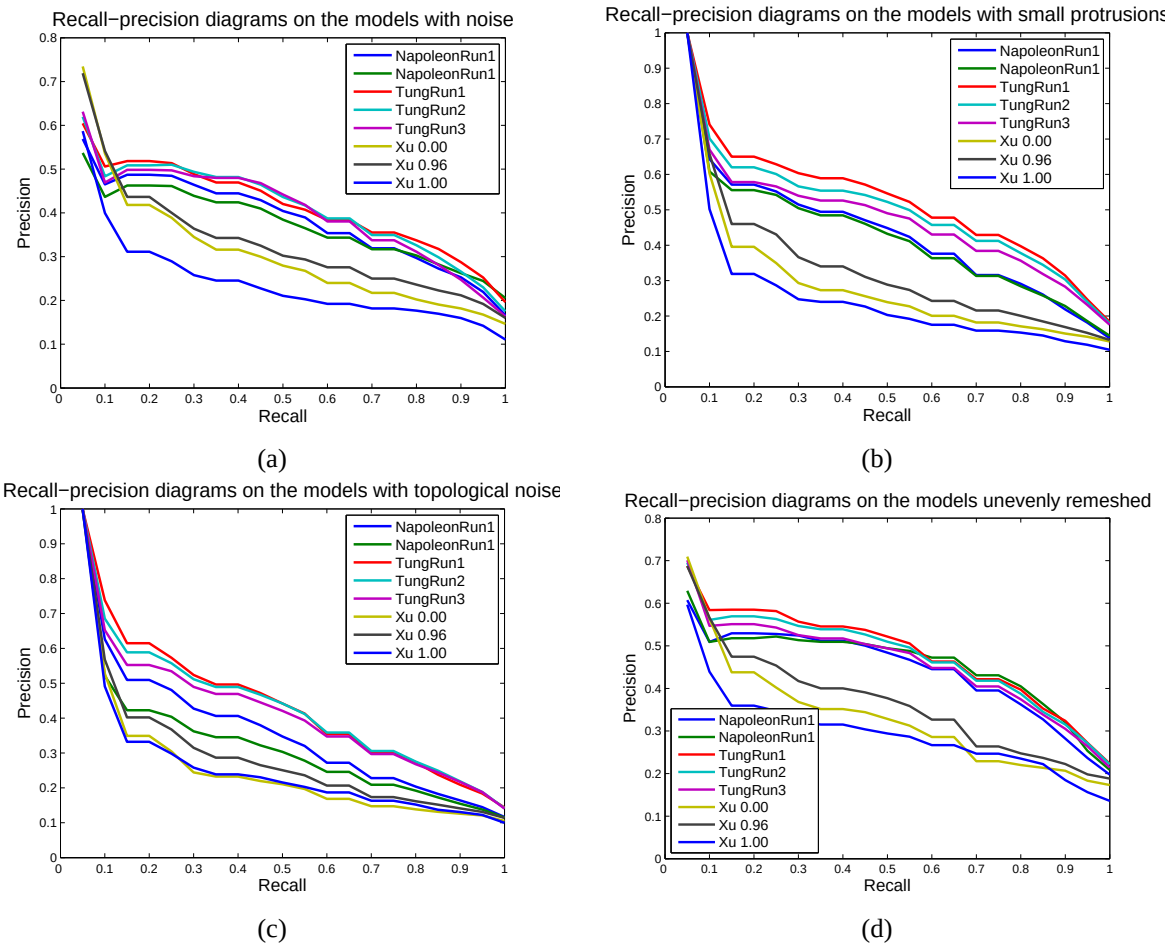


Figure 7: Comparison of the different methods over different classes of perturbations.

Degradation of the retrieval performance				
Method	*Guassian Noise*	*Small Protrusions*	*Topological Noise*	*Uneven Re-sampling*
Tung et al., *Run1*	55.26%	51.49%	48.11%	39.05%
Tung et al., *Run2*	54.31%	50.85%	49.12%	40.01%
Tung et al., *Run3*	53.68%	50.03%	49.34%	40.30%
Napoleon et al., *Run1*	45.20%	54.89%	48.56%	38.04%
Napoleon et al., *Run2*	47.63%	53.96%	49.87%	39.30%
Xu et al, *0.00*	33.48%	51.78%	44.28%	26.63%
Xu et al, *0.96*	36.65%	54.03%	45.51%	30.22%
Xu et al, *1.00*	39.95%	55.24%	49.13%	32.27%

Table 2: The original models are used as queries against the corresponding perturbed models, the classes of both queries and dataset are made of 20 elements.

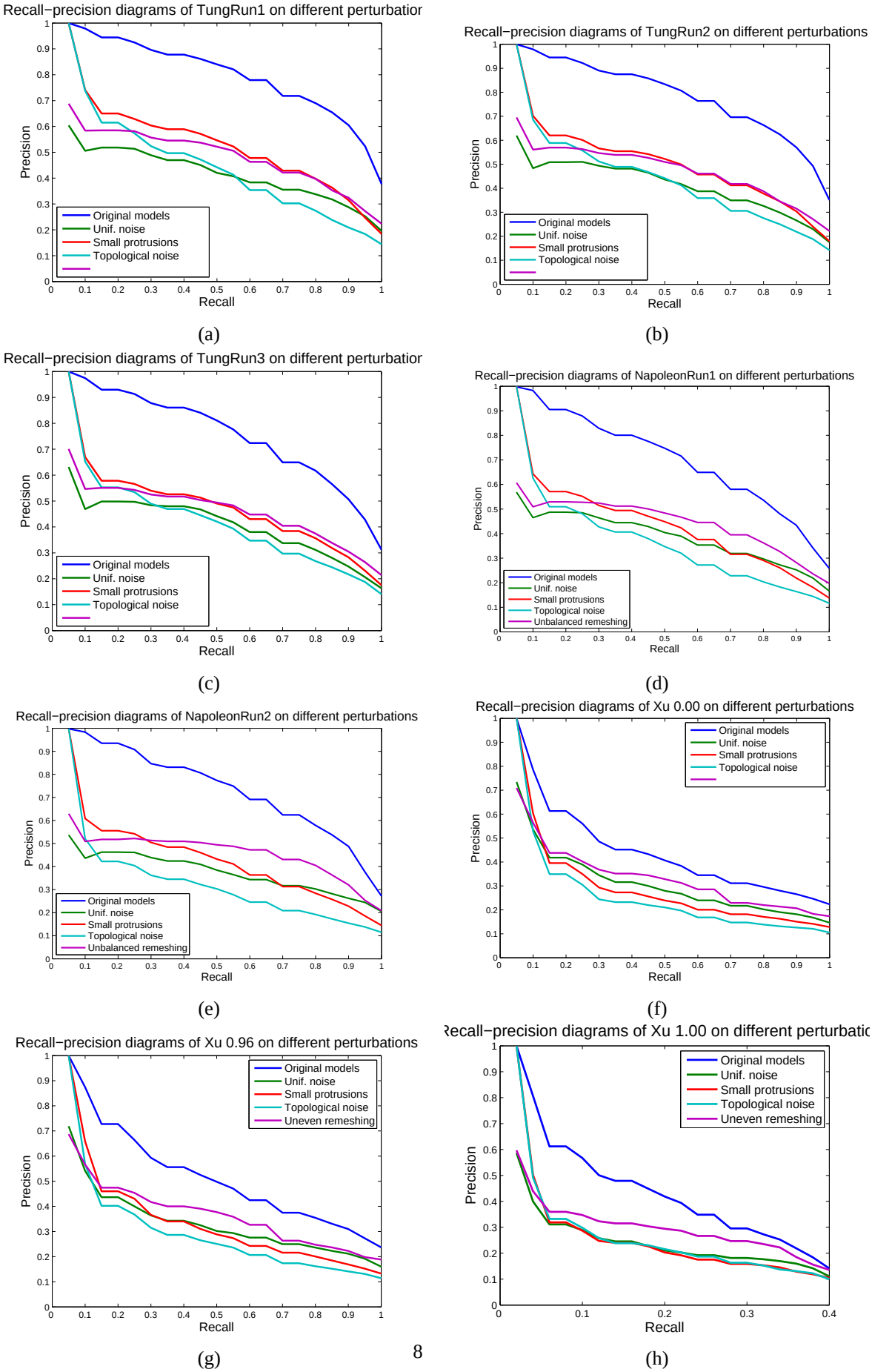


Figure 8: Degradation of the performance with respect to the different types of perturbations.

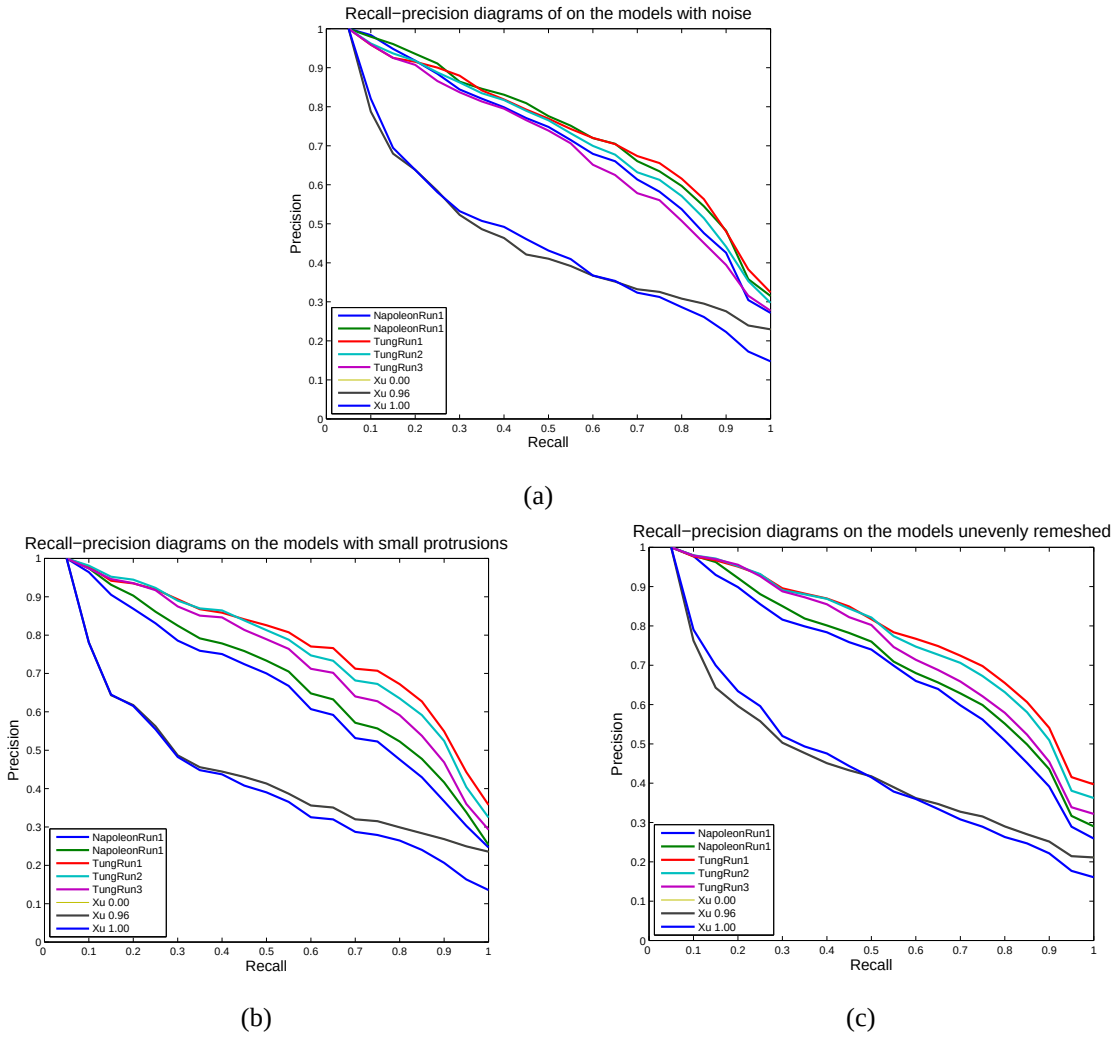


Figure 9: Comparison of the different methods on dataset A when different perturbations are considered.

of the different runs over the different types of perturbation. Since the number of models with topological noise in the dataset A is considerably smaller than the other kind of perturbed models and the recall/precision perfomance measures depend on the size of the dataset, in Figure 9 we do not report the degradation of the methods for models perturbed with topological noise.

Furthermore, differently from what we did for dataset B, we did not compare the performance of the original dataset over the different kinds of perturbation; using precision-recall diagrams for such a comparison, in fact, would have been not fair because the cardinality of the dataset is not constant across the various kinds of perturbation. For the same reason, we did not report any table with the level of degradation of the methods across the different perturbations.

Acknowledgments

The authors would like to thank Daniela Giorgi and Simone Marini for their support during the preparation of the datasets and the evaluation of the results. This work has been developed in the CNR research activity (ICT-P04) and partially supported by the European Network of Excellence “AIM@SHAPE” (contract number 506766).

References

[1] M. Attene and B. Falcidieno. ReMESH: An interactive environment to edit and repair triangle meshes. In *SMI '06: Proceedings of the IEEE International Conference on Shape Modeling and Applications 2006*, pages 271–276, Washington, DC, USA, 2006. IEEE Computer Society.

[2] D. Giorgi, S. Biasotti, and L. Paraboschi. Watertight models track. Technical Report 09, IMATI, Genova, Italy, 2007.

[3] G. Salton and M. McGill. *Introduction to modern information retrieval*. McGraw Hill, 1983.