Eurographics Workshop on 3D Object Retrieval (2012)
M. Spagnuolo, M. Bronstein, A. Bronstein, and A. Ferreira (Editors)

SHREC'12 Track: Stability on Abstract Shapes

S. Biasotti[1], X. Bai[3], B. Bustos[4], A. Cerri[2], D. Giorgi[1], L. Li[3], M. Mortara[1], I. Sipiran[4], S. Zhang[3], M. Spagnuolo[1]

[1]Istituto di Matematica Applicata e Tecnologie Informatiche 'E. Magenes', CNR, Italy
[2]ARCES, University of Bologna, Italy
[3]Northwestern Polytechnical University, Xi'an, China
[4]Department of Computer Science, University of Chile, Chile

Abstract

This contribution reports the results of the SHREC 2012 track: Stability on Abstract Shapes. This track saw six registrations of which only three participants effectively sent the results of their runs.

Categories and Subject Descriptors (according to ACM CCS): H.3.1 [Information Storage and Retrieval]: Content Analysis and Indexing—Abstracting methods;

1. Intruduction

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. In this report the results of the SHREC 2012 track: *Stability on Abstract Shapes* are presented. The aim of this track is to evaluate the stability of shape matching algorithms with respect to input perturbations that modify the representation of the object in terms of mesh connectivity and simple affine deformations. The novelty of this track is both in the choice of the models that are chosen in a set of mathematical primitives and in the choice of the shape deformations that can be non-isometric. The shape perturbations include geometric noise, varying sampling patterns and non-isometric shape deformations such as non-rigid stretches.

2. Data Collection and Queries

The dataset is made of 504 watertight mesh models. Given a *base* set of 18 shapes (e.g. cubes, spheres, torii, multiple-torii, see Figure 1), each model is perturbed with 9 transformations applied with three different intensities, for a total of 27 modifications each (see Figure 3).

The set of transformations is: i) an additive Gaussian noise applied with 1%, 2% and 5% intensity; ii) three uneven sampling distributions obtained with the command-line version of the ReMESH software [AF06] with parameters 50, 70 and 80; iii) three uniform model samplings (with 1000, 2000 and 5000 vertices); iv) and v) three stretches with respect to one or two shape axes (2, 3 anf 4 times the original lenght); vi) and vii) three non-uniform dilatations orthogonal to one or two shape axes; viii) and ix) three non-uniform erosions with respect to one or two shape axes (the perturbations of the cube model are shown in Figure 2). At the end each class of

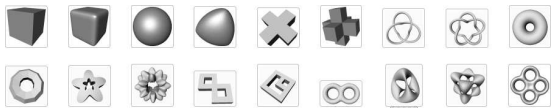

Figure 1: *The 18 models from which the dataset is built.*

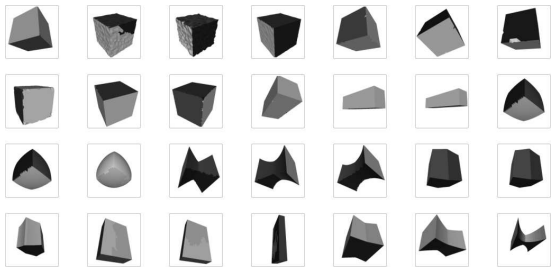

Figure 2: *From left to right and from top to bottom: a model (top-left) and its variations (9 rigid and the 18 non-isometric transformations).*

DOI: 10.2312/3DOR/3DOR12/101-107

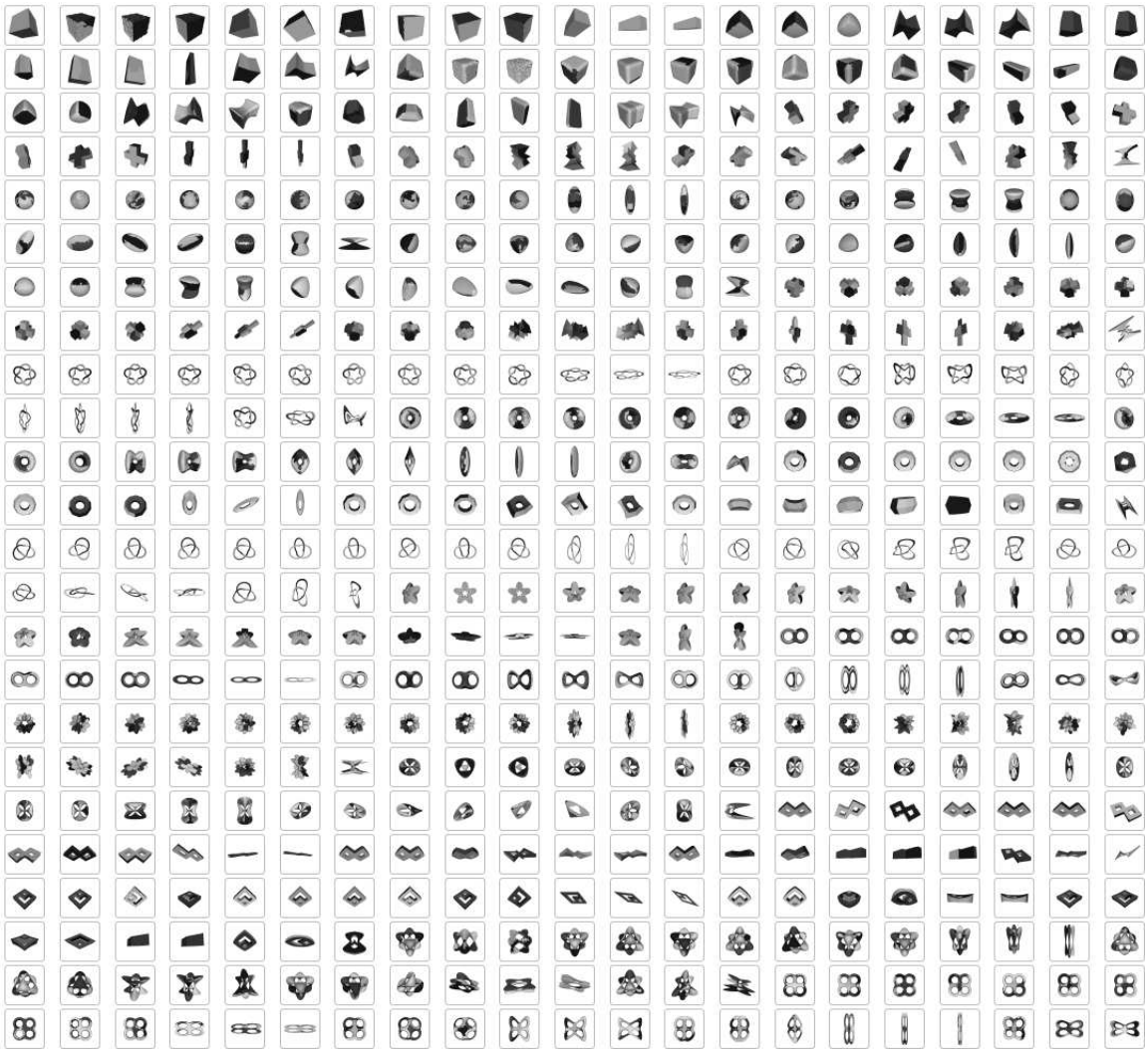

Figure 3: *The dataset of the Stability on Abstract Shapes track.*

the dataset was made of 28 models (the original model and 27 transformations) as depicted in Figure 3.

3. Participants

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

Three groups participated to the SHREC'12 track: Stability on Abstract Shapes and seven dissimilarity matrices were submitted:

1. X. Bai, L. Li and S. Zhang from the Northwestern Polytechnical University, Xi'an, China, participated with 3 matrices (LSU-r02.txt LSU-r03.txt and LSU-sum.txt), the method is detailed in Section 4.1);
2. I. Sipiran and B. Bustos from the Dept. of Computer Science, University of Chile, Chile participated with 2 matrices (run1.matrix and run2.matrix), the method is detailed in Section 4.2;
3. A. Cerri, D. Giorgi and M. Mortara from ARCES, University of Bologna and IMATI-CNR in Genova, Italy, participated with 2 matrices (size1.txt and size2.txt), the method is detailed in Section 4.3.

In addition to the three groups of participants listed above, three further registrations to the track were received but participants withdrew the track.

4. Description of the methods

In this section the descriptions of the three methods that effectively sent their outcomes over the track dataset are listed.

4.1. Abstract Shape Retrieval Using Local Shape Distributions

The proposed method addresses the problem of retrieving abstract shapes using the local shape features. The method is inspired by the BOGH algorithm [LGB*11]. However, a different strategy is used , in which the new Local Shape Distribution (LSD) descriptor is suggested as the shape representation for 3D shapes.

4.1.1. Local Distribution Descriptor

Let P denotes a surface point of a 3D object. Its r-neighborhood is defined as the spherical region centered at P with the radius r. The LDS descriptor associated to this region is the histogram-vector of the Euclidean-distances between P and the other surface points within the region. Since all points in the r-neighborhood of p have their own contributions to the local shape of the 3D object in this region, and such contributions are decreased with the increase of distances between the points and the center of the region, i.e. point P, each bin of the LDS histogram is Gaussian weighted ($\sigma = 0.3$), with an attempt at accurately indicating the shape distribution in the region.

4.1.2. Feature Extraction

The proposed method starts feature extraction by randomly sampling n points on the surface of a 3D object. Note that, it's assumed that a scale normalization on the object has been conducted. For each sample point, the LSD descriptor of its r-neighborhood is computed, which is composed of d bins (d = 32). After that, the k-means algorithm is employed to carry out clustering on the resulting n LSD descriptors. The aim of this step is to select those characteristic descriptors, i.e. the centers of k clusters, so as to improve speed in similarity matching. In this way, the 3D object is represented by a set of k LSD descriptors. In this track, n and k are set to 3000 and 250 respectively. The procedure of feature extraction is shown in Figure 4.

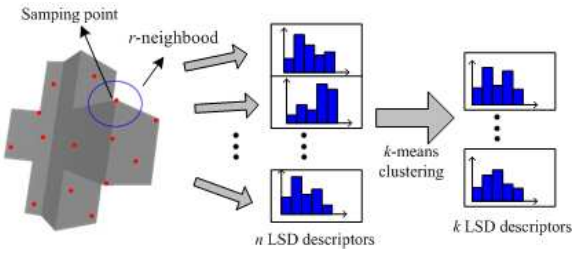

Figure 4: *Procedure of feature extraction.*

4.1.3. Similarity Matching

Similarity matching in the proposed method is analogous to that in BOGH. Let L_Q and L_C denote the LSD descriptor sets of a query object and a candidate in shape database respectively. The Hungarian algorithm is employed to build the correspondences between L_Q and L_C. The likeness of two descriptors is determined by using χ^2.

4.1.4. Settings of the experiments

For this track, two configurations of the proposed method (r = 0.2 and 0.3) were chosen to calculate the dissimilarity matrices. A fused version, which combined the above resulting matrices with the result from the configuration in which r is set to 0.7, was also presented.

Software to compute LSD descriptor is freely available at [BLZ12].

4.2. Shape Retrieval with the Signature Quadratic Form Distance

Our method relies on the use of a flexible distance which is suitable to compare two entities represented by feature sets. The Signature Quadratic Form Distance is a context-free distance that has proven to be effective in the image domain [BUS09]. In addition, it is also a good alternative to Bag of Feature approaches.

The aim is to partition the whole feature space of a 3D object to define the feature signatures. Let P a 3D model with m vertices. Each vertex is represented by a heat kernel signature. The feature space of P is represented as the normalized heat kernel signatures [SOG09] defined as:

$$FS(P) = \{\frac{hks(v_i)}{\|hks(v_i)\|} | v_i \to P, i = 1,\ldots,m\}.$$

Thus, the feature space FS will be used to create the signature features for a object. First of all, a clustering algorithm derived from Leow and Li [LL04] is applied. Briefly, the clustering uses two thresholds to define the inter-cluster and intra-cluster properties, so it does not depend on the number of clusters. Therefore, it is an adaptive method which depends on the distribution of points in the space.

Given a partiotioning after the clustering, the feature signature S^P of an object P is defined as a set of tuples $FSxR^+$ as follows

$$S^P = \{(c_i^P, w_i^P), i = 1,\ldots,n\} \quad (1)$$

where c_i^P is the average heat kernel signature in the i-th cluster and w_i^P is the fraction of elements belonging to the i-th cluster. Note that the representation of an object depends of the clustering and it is not neccesary that two objects have

the same number of clusters. Finally, given two objects represented by their feature signatures, the Signature Quadratic Form Distance [BUS09] is directly applied.

For the implementation of the results presented in this paper, the surface area of the objects is firstly normalized to 1.0. Then, the heat kernel signatures as proposed originally by Sun et al [SOG09] is computed. Next, the clustering using two configurations for the thresholds is applied. For the first experiment, the values 0.1 and 0.2 are set as intra-cluster and inter-cluster thresholds, admitting clusters with at least 15 elements. For the second experiment, the values 0.05 and 0.1 are used for the intra-cluster and inter-cluster thresholds, and the clusters contain 5 elements at least. Finally, a Gaussian function with $\alpha = 0.9$ for the similarity in the SQFD is adopted.

4.3. Size functions as an hybrid 2D/3D approach

In this Track a hybrid retrieval method which combines a description of the 3D mesh with a description of its *best* 2D view is proposed. A modular transform, called *size function*, forms the basis for both descriptions. The peculiarity of this transform is that it provides a topological and geometrical analysis of shapes, and can be applied to different types of input [CFG06]: in this case, both images and 3D meshes.

Size functions. The main idea behind size functions is to model the shape of an object as a pair (S, φ), where S is a topological space and $\varphi : S \to \mathbb{R}$ is a continuous function called *measuring function*. The role of S is to represent the object under study, while φ can be seen as a descriptor of some properties of interest according to which the object is analyzed [FL99]. Roughly speaking, size functions code the topological evolution of S counting the number of connected components which remain disconnected passing from a lower level set S_u of S to another level set S_v with $u < v$, where a lower level set is defined as $S_u = \{P \in S : \varphi(P) \leq u\}$ with $u \in \mathbb{R}$. As shown by Fig. 5, size functions can be seen as collections of points lying in the half-plane $\{(u,v) \in \mathbb{R}^2 : u < v\}$, called *cornerpoints* (red dots in Fig. 5$(a-b)$), which describe the lifespan of connected components. For each point, the u-coordinate denotes the *birth* of a connected component, in terms of the values of the measuring function that generates it; similarly, the v-coordinate denotes its *death*. The distance from the diagonal $u = v$ represents the component lifespan, which in turn signals the importance of the feature that component represents: points far from the diagonal describe important, *long-lived* features, whereas points close to the diagonal describe local information such as smaller details and noise. The red vertical line in Fig. 5(a) – as well as the one in Fig. 5(b) – can be seen as a point at infinity, representing a connected component that *will never die*, i.e. its u-component corresponds to the smallest value of the function and its v-component is equal to $+\infty$. The representation using cornerpoints allows for the comparison of size functions using distances between sets of points and lines, such as the Hausdorff distance or the *matching distance*, which measure the cost of moving one set of points into the other. Size functions are stable with respect to the matching distance [dFL10], thus implying resistance to noise.

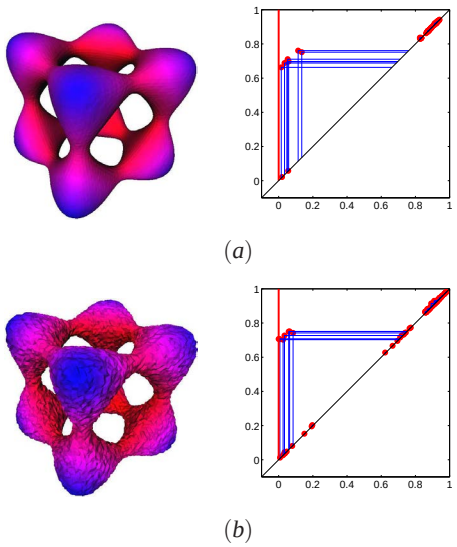

Figure 5: (a) *A model from the dataset with the integral geodesic distance color-coded (left), and the corresponding size function (right).* (b) *A noisy version of the model and its shape descriptor: Noise induces only small changes in the position of points far from the diagonal, possibly producing variations close to the diagonal.*

An interesting aspect of size functions is that it is possible to get different descriptors just by changing the space S and the real function φ. This is exactly what happens here: For each model, S is in turn the triangle mesh, or the shape contour of its 2D silhouette; φ is chosen accordingly, as described in what follows. The result is a collection of descriptors which inherit the invariance properties of the underlying measuring functions, and able to provide information about a shape according to different viewpoints.

3D descriptors. On 3D meshes the measuring functions considered are the distance from the barycenter, and the integral geodesic distance, that is, for each point the average geodesic distance to all other points is taken. Both measuring functions are rotation and translation invariant; moreover, the integral geodesic distance is also isometry invariant. The choice of these functions ensures that our descriptors are robust to noise, rigid and non-rigid movements. Scale invariance was obtained by normalizing the models a priori. For each measuring function, two models are compared by computing the matching distance between the associated size functions; the final 3D similarity score is defined as the average of the two matching distances.

2D descriptors. For each 3D model, its *best view* is selected. Starting from a set of viewpoints uniformly sampled

over a viewing sphere surrounding the object, for each viewpoint it is evaluated a scoring function which takes into account the visibility of salient features (percentage of visible surface), their relevance (surface of the feature with respect to the whole object), and the semantic type of each salient feature. The viewpoint with highest score determines the selected view. On this 2D view, four measuring functions are computed on the silhouette contour, namely four estimates on the local curvature at each contour point, computed at four different resolutions [GMS10]. The final 2D similarity score is defined as the average of the four corresponding matching distances, augmented with the L_2 distance between the first 35 Zernike moments of the silhouette.

The final 2D/3D similarity score is the sum of the 3D and 2D scores.

5. Performance Measures

Each model was used in turn as a query against the remaining part of the database. The performance of the methods on the dataset has been evaluated on the basis of the ground truth that was established a priori by three classification schemes (focusing either on the isometric transformations (i-iii) or the non-isometric ones (iv-ix) and the overall dataset). The first scheme considers in the same class the models in the original class and their rigid perturbations, that is, each class is made of the original model plus 9 transformations, so that each class is made of 10 elements. The second one (finer) considers in the same class just a single model and its 18 non-rigid perturbations, that is, each class is made of 19 models. Finally, the third scheme considers in the same class the original models and all of its 27 perturbations, i.e. each class is made of 28 elements.

As performance measures of the method the nearest neighbour, the first and the second tier, already used in past tracks of SHREC [GBP07, BA08] are adopted. In addition the **precision** and **recall** are considered, 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 [SM83].

Recall and precision are represented in a diagram, where precision has been computed as average if the precision scores after each relevant item in the scope. Then, the precision-recall measures computed for each query are averaged over the entire database. Finally, the area under the precision-recall diagrams is considered which is relevant to evaluate the overall performance of a method.

6. 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 see Table 1 and Figure 6; hence, it makes sense to consider the best run for each method and compare the methods according to these best runs. However, for completeness, precision-recall diagrams are depicted all together in a single graphical panel.

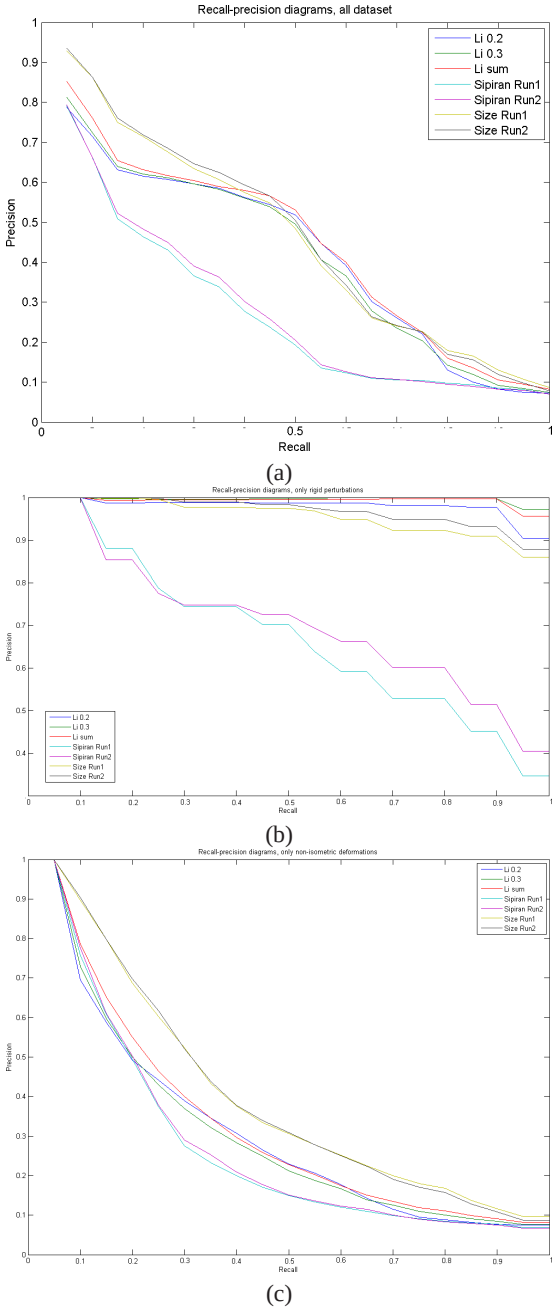

Figure 6: *Recall precision graphs of each participant over the (a) whole dataset, (b) rigid transformations and (c) non-rigid deformations.*

Method	NN	FT	ST	PRarea
LSD-r02	68.66	62.54	39.03	0.41207
LSD-r03	69.65	60.80	39.30	0.40905
LSD-sum	73.22	63.11	40.44	0.43072
Sipiran run1	67.66	40.97	26.19	0.26358
Sipiran run2	69.65	42.17	27.02	0.27157
Size run1	87.90	66.08	40.79	0.44500
Size run2	88.50	66.97	41.03	0.44994

(a)

Method	NN	FT	ST	PRarea
LSD-r02	98.34	53.26	27.64	0.97840
LSD-r03	99.45	54.70	27.78	0.99425
LSD-sum	98.89	54.63	27.78	0.99198
Sipiran run1	80.56	33.96	19.22	0.65921
Sipiran run2	77.78	35.68	20.17	0.69160
Size run1	99.45	51.30	27.24	0.95243
Size run2	100	52.23	27.38	0.96545

(b)

Method	NN	FT	ST	PRarea
LSD-r02	53.81	30.74	19.84	0.29458
LSD-r03	55.56	29.81	20.55	0.29241
LSD-sum	61.12	31.44	21.65	0.31141
Sipiran run1	62.87	25.90	16.59	0.25964
Sipiran run2	66.96	26.17	17.13	0.26432
Size run1	82.17	38.26	25.15	0.38546
Size run2	82.46	25.15	24.98	0.38415

(c)

Table 1: *Retrieval performance on (a) the whole dataset, (b) rigid perturbartions and (c) non-rigid deformations using four standard measures.*

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. Therefore, for each method, the best run was selected as the one with the maximum area under the precision-recall diagram.

6.1. Performance on the dataset

Figure 6 shows the recall precision diagram obtained using the three classifications of the dataset.

Interestingly, while all methods are quite stable when small noise or re-sampling operations affect the models, most of methods strongly degradates when the non-rigid deformations are considered, see for instance the statistics in Table 1(a). In Table 1 the symbols NN denote the nearest-neigbour, FT the first tier, ST the second tier and PRarea the area under the recall precision diagram. In particular, the highest the values of the FT and ST the smallest the number of false positives in the first 28 (and respectively) 56 retrieved items.

Both recall precision diagrams in Figure 6 and the statistics in Table 1 indicate that the dataset is really challenging. In fact there are many difficulties in terms both of deformations and in the choise of the models (for instance the first and the second class differ from the fact the two cubes have sharp edges or not). In one hand methods that are able to detect small differences would fail when a significant deformation is applied; on the other hand, method that roughly describe the overall shape are not able to distinguish between sharp and smooth corners and edges.

More details of the performance of the methods over the different classes are shown in Figure 7 that represents the histograms of the NN classifier. In this Figure only the best run of the methods are shown. The results highlight how the performance degradate when non-isometric deformations are considered and how the methods that are able to distinguish the rough structure of the shape have difficulties when two shapes differ for small details (see, for instance, in Figure 7(c) the performance of the LSD-sum method with respect to the classes 1 and 2 of the dataset that correspond to the cubes with sharp and smooth edges).

In general the methods based on the hybrid 2D/3D approach given by the size function framework performs quite well on the whole dataset (and they generally scored best in all parameters, that is NN-Neighbour, First Tier, Second Tier, and Precision-Recall). More in detail, size functions scored best on the sub-set of non-rigid transformations, and had comparable performance on rigid transformations.

7. Conclusions

In this paper, the new track of SHREC'12 on *Stability on abstract shapes* is introduced describing how the dataset was built and the kind of shape perturbations made on a set of simple abstract primitives. In fact, to better estimate the pros and cons of the methods that participated to the track simple and well defined mathematical surfaces are chosen, these shapes are characterized by different genus, sharp edges and corners or smooth appareance, etc.. This is the first time time that a track of SHREC has the focus of specifically estimate the robustness of the methods not only on small shape perturbation but also on affine transformations like non-linear stretching and bumping.

The experimental results show that the dataset is really challenging: most of methods have some difficulties to perform well both on fine transformations (noise or re-sampling) and non-isometric transformations (non-linear stretching). An additional challenge is to keep the intra-class variability (for instance considering the sphere similar to the ellipse) discarding the sometime small extra-class variability (for instance the cube with sharp from the one smooth edges). Finally, it is to be hoped that this new benchmark will promote further investigation on affine-independent shape comparison and retrieval methods.

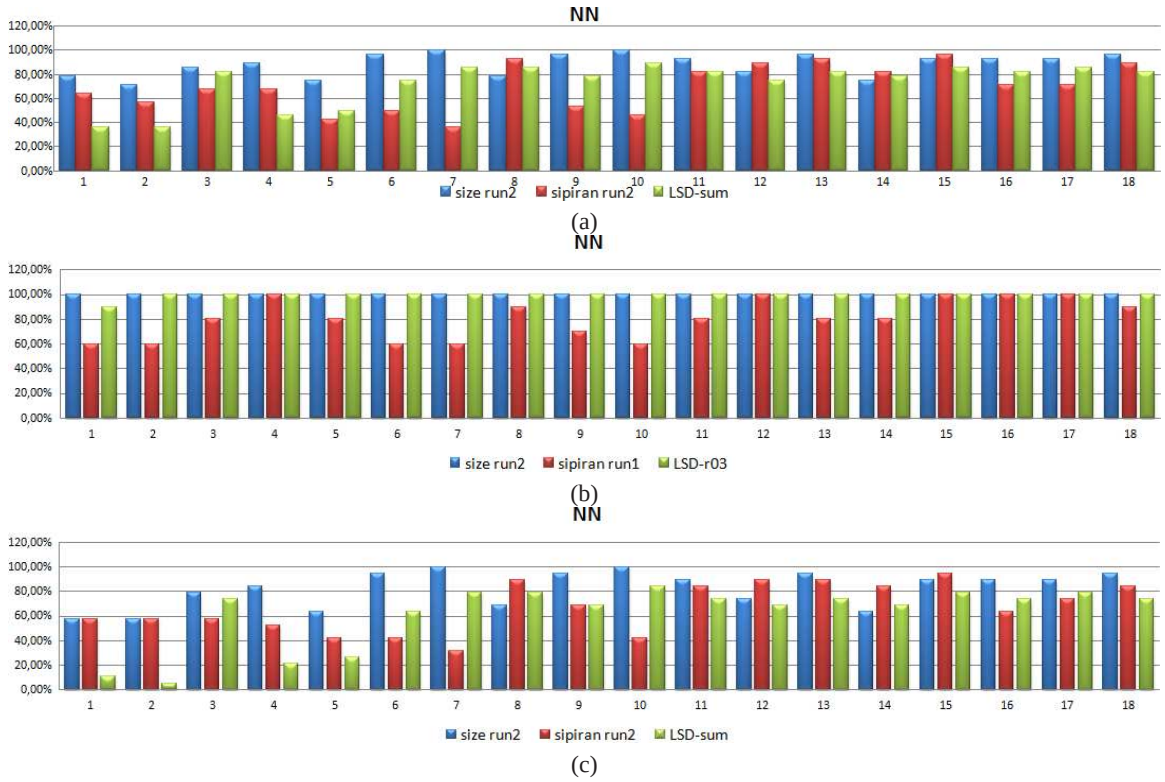

Figure 7: *Nearest-neighbor performance (best runs) over the (a) whole dataset, (b) rigid transformations and (c) non-rigid deformations. The number of the class correspond to the order of the models in Figure 1.*

Acknowledgments

The authors would like to thank M. Attene for providing software for automatic mesh processing. This work has been developed in the CNR research activity (ICT-P04) and partially supported by MIUR-PRIN Project N. 2009B3SAFK_002.

References

[AF06] ATTENE M., FALCIDIENO B.: 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* (Washington, DC, USA, 2006), IEEE Computer Society, pp. 271–276. 1

[BA08] BIASOTTI S., ATTENE M.: Shape retrieval contest 2008: Stability of watertight models. In *Shape Modeling and Applications, 2008. SMI 2008. IEEE International Conference on* (2008), IEEE, pp. 217–218. 5

[BLZ12] BAI X., LI L., ZHANG S.: Software for 3d model retrieval using local shape distributions. http://code.google.com/p/shape-retrieval, Feb. 2012. 3

[BUS09] BEECKS C., UYSAL M. S., SEIDL T.: Signature quadratic form distances for content-based similarity. In *Proc. of the ACM Int. Conf. on Multimedia* (New York, NY, USA, 2009), MM '09, ACM, pp. 697–700. 3, 4

[CFG06] CERRI A., FERRI M., GIORGI D.: Retrieval of trademark images by means of size functions. *Graph. Models 68*, 5 (2006), 451–471. 4

[dFL10] D'AMICO M., FROSINI P., LANDI C.: Natural pseudo-distance and optimal matching between reduced size functions. *Acta. Appl. Math. 109* (2010), 527–554. 4

[FL99] FROSINI P., LANDI C.: Size theory as a topological tool for computer vision. *Pattern Recogn. and Image Anal. 9* (1999), 596–603. 4

[GBP07] GIORGI D., BIASOTTI S., PARABOSCHI L.: *Watertight Models Track.* Tech. Rep. 09, IMATI, Genova, Italy, 2007. 5

[GMS10] GIORGI D., MORTARA M., SPAGNUOLO M.: 3d shape retrieval based on best view selection. In *Proc. ACM Workshop on 3D Object Retrieval* (2010), pp. 9–14. 5

[LGB*11] LIAN Z., GODIL A., BUSTOS B., DAOUDI M., HERMANS J., KAWAMURA S., KURITA Y., LAVOU E G., NGUYEN H. V., OHBUCHI R., OTHERS: Shrec'11 track: Shape retrieval on non-rigid 3d watertight meshes. *Eurographics Workshop on 3D Object Retrieval 10* (2011), 079–088. 3

[LL04] LEOW W. K., LI R.: The analysis and applications of adaptive-binning color histograms. *Comput. Vis. Image Underst. 94* (April 2004), 67–91. 3

[SM83] SALTON G., MCGILL M.: *Introduction to modern information retrieval.* McGraw Hill, 1983. 5

[SOG09] SUN J., OVSJANIKOV M., GUIBAS L. J.: A concise and provably informative multi-scale signature based on heat diffusion. *Comput. Graph. Forum 28*, 5 (2009). 3, 4