Eurographics Workshop on 3D Object Retrieval (2015)
I. Pratikakis, M. Spagnuolo, T. Theoharis, L. Van Gool, and R. Veltkamp (Editors)

SHREC'15 Track: Scalability of Non-Rigid 3D Shape Retrieval

I. Sipiran[†1], B. Bustos[†2], T. Schreck[†1], A. Bronstein[3], M. Bronstein[4], U. Castellani[5], S. Choi[6],

L. Lai[7], H. Li[7], R. Litman[3], L. Sun[7]

[1]Department of Computer and Information Sciences, University of Konstanz
[2]Department of Computer Science, University of Chile, [3]School of Electrical Engineering, Tel-Aviv University
[4]Faculty of Informatics, University of Lugano, [5]Department of Computer Science, University of Verona
[6]Department of Biomedical Engineering, Seoul National University, [7]Beijing Technology and Business University

Abstract

Due to recent advances in 3D acquisition and modeling, increasingly large amounts of 3D shape data become available in many application domains. This rises not only the need for effective methods for 3D shape retrieval, but also efficient retrieval and robust implementations. Previous 3D retrieval challenges have mainly considered data sets in the range of a few thousands of queries. In the 2015 SHREC track on Scalability of 3D Shape Retrieval we provide a benchmark with more than 96 *thousand shapes. The data set is based on a non-rigid retrieval benchmark enhanced by other existing shape benchmarks. From the baseline models, a large set of partial objects were automatically created by simulating a range-image acquisition process. Four teams have participated in the track, with most methods providing very good to near-perfect retrieval results, and one less complex baseline method providing fair performance. Timing results indicate that three of the methods including the latter baseline one provide near- interactive time query execution. Generally, the cost of data pre-processing varies depending on the method.*

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Picture/Image Generation—Line and curve generation

1. Introduction

The experimental comparison of shape retrieval methods is important for the improvement of existing and the design of novel methods in this area. Regularly, experimental comparisons are carried out as part of the evaluation in technical publications, as well as the SHREC shape retrieval evaluation efforts. So far, shape retrieval evaluation has typically considered data sets of moderate size up to thousands of objects. For example, the dataset proposed in [LLL*14a] consisted of thousands of query and target objects, including 3D models and user queries. There, a large number of user query sketches was obtained previously by a crowd-sourced approach.

Considering scalable 3D retrieval is a relevant endeavor, as some 3D repositories like Sketchup 3D Warehouse [Ske] or TurboSquid [Tur] today comprise tens of thousands of shapes. Also, it can be expected that with increased availability of 3D acquisition facilities including crowd-based photogrammetric methods [GAF*10], or consumer-type sensors like Microsoft Kinect, large-scale shape retrieval will become important. Scalable approaches should provide *efficient* similarity computation and ranking, to answer user queries interactively. Also, and as a pragmatic aspect, scalable methods should work also *robustly* in a fault-tolerant way regarding outlying and degenerate models, as may be encountered when studying large-scale 3D repositories. The provision of large-scale retrieval benchmarks has recently been limited by availability of real data, which often is expensive to obtain.

In this track, we increase the number of query objects by an order of magnitude. Our benchmark is based on a set of

† Track organizers.

Dataset	Size	Year
National Taiwan University [CTSO03]	10,911	2003
Princeton Shape Benchmark [SMKF04]	1,814	2004
SHREC'09 Generic Retrieval [GDA*09]	600	2009
Toyohashi [TKA12]	10,000	2012
SHREC'13 Range Images [SMB*14]	7,200 queries, 600 targets	2013
SHREC'14 Large-scale [LLL*14b]	8,987	2014

Table 1: *Shape retrieval benchmarks and their sizes. Note that, in general, the size of shape retrieval datasets has not been increased in ten years.*

229 non-rigid objects [SP04] which have been extended by several other existing 3D retrieval benchmarks. For each object, several partial objects were generated. The total benchmark amounts to over 96 thousand objects. The goal was to test the scalability of current retrieval methods for coping with a large number of partial queries, and evaluating the performance in terms of retrieval effectiveness and efficiency.

Four teams have participated in the track with methods based on heat kernel descriptors in sparse coding, features combination with learned feature weighs, diffusion descriptors with quadratic form distance, and histograms of geodesic distances. Three more complex methods provide very good and up to near-perfect retrieval precision; one less complex method provides fair retrieval precision. Timing results taken on current hardware configurations indicate that the latter method provides for fast object pre-processing and query execution. Generally, three of the methods provide near-interactive time query execution with less than 2 seconds per query. The time cost of data pre-processing varies depending on the method.

2. Related Work

As we are living the Big Data Era, big efforts are focused on making computational systems scalable. In the context of multimedia information, the topic of image retrieval has recently been addressed from the perspective of scalability. Examples of scalable image retrieval systems have been proposed by Wang et al. [WKC10] and Wu et al. [WKSS11]. Similarly in video retrieval, a good exemplary system is Video Google [SZ03], a successful method to retrieve objects by similarity in large amounts of videos. Nevertheless, in 3D object retrieval the scalability has not been thoroughly studied yet, mainly due to the lack of large scale benchmarks.

A brief review of some datasets and their sizes can give us a better idea of the evolution of 3D retrieval datasets. Table 1 shows common datasets in the 3D retrieval community. An interesting fact is that in ten years, the size of the datasets for evaluation has not increased. In our opinion, this is a clear evidence that scalability has not been a high-priority evaluation criterion.

A few methods with scalability as central concept have been proposed so far. Bronstein et al. [BBGO11] proposed the Shape Google approach. The method applies the well-known Bag-of-features approach using diffusion descriptors. More interestingly, the adoption of a spatially sensitive Bag of Features and the learning of binary descriptors (invariant to transformations) allow to enhance the performance of shape retrieval. It is worth noting that Shape Google was evaluated with a dataset of 1,184 shapes. In the same direction, Litman et al [LBBC14] recently proposed a supervised method to learn the dictionary of Bag of Features through a sparse coding approach. The dataset used in experimentation was also the Shape Google dataset. Finally, Dadi and Daoudi [DD13] proposed a method based on multi-core architectures to deal with large datasets. However, in the evaluation the method was tested using the Princeton Shape Benchmark (1,814 shapes).

As can be seen, although many methods could perfectly work with real-world large-scale datasets, until now it was not possible to test them in a large-scale scenario. Our main motivation to create a large scale dataset is to fill the gap in the evaluation of current approaches and their capabilities.

3. The Benchmark

This task is built upon a non-rigid shape retrieval task which has been shown to be successful in recent years in previous SHREC editions [LGB*, LGB*13]. Indeed, our decision of building a large-scale dataset upon a non-rigid retrieval task relies on the observation that there is a good number of methods and the achieved effectiveness is high.

Our base dataset is the publicly available collection from Sumner [SP04]. It contains 229 non-rigid shapes categorized in 9 classes. Subsequently, we populate our dataset including unclassified objects from several publicly available shape collections such as Princeton Shape Benchmark [SMKF04], Konstanz database [Vra04], SHREC' 09 Generic dataset [GDA*09], and SHREC' 14 Large Scale Comprehensive Retrieval [LLL*14b]. After including models from the previous datasets, we generate a large set of shapes using the method proposed in [SMB*14] to generate simulated range images using the SHREC' 14 Large Scale dataset as target models. After obtaining a first large-scale set of shapes, we applied a careful post-processing step in order to repair non-manifold objects and merge objects with more than 1 connected component.

Additionally, as many state-of-the-art approaches use geodesic distances and heat diffusion descriptors, we did a final check in the large dataset. We only kept the models in which it was possible to compute the eigendecomposition of their Laplace-Beltrami operator and their pair-wise geodesic distances. Our final dataset only contains manifold objects with one connected component. This pre-processing step will guarantee that most of the current approaches work with our dataset. In total, our collection contains 96,487 models. The dataset, ground-truth and evaluation code can be downloaded in [Dat].

4. Methods

In the competition, we received six runs from four different groups. We present the list of contributors, the name of the technique and the abbreviation used during the evaluation:

- Roee Litman and Alex Bronstein: *Supervised learning of bag-of-features shape descriptors using sparse coding* (Sparse-BoF). The team provided only one run.
- Sungbin Choi: *Geodesic distance distribution* (GDD). The team provided two runs.
- Long Lai, Li Sun, and Haisheng Li :*Combined features with a genetic algorithm* (GA). The team provided only one run.
- Ivan Sipiran and Benjamin Bustos: *Signature quadratic form distance on diffusion descriptors* (SQFD). The team provided two runs.

Next, we present the methods submitted and the configurations used in the experiments.

4.1. Supervised learning of bag-of-features shape descriptors using sparse coding

The settings described here closely follow the ones presented in [LBBC14, P*14]. All shapes were down-sampled to have 4,500 triangles, and scaled to have a similar area. For each shape $\mathcal{S}$ in the data-set, a 16-dimensional SI-HKS [BK10] descriptor $\mathbf{x}_i$ was calculated for every vertex $i \in \mathcal{S}$. The *bag-of-features* representation of each $\mathbf{x}_i$ in a dictionary $\mathbf{D}$ is obtained by means of solving the pursuit problem

$$\min_{\mathbf{z}_i} \frac{1}{2}\|\mathbf{x}_i - \mathbf{D}\mathbf{z}_i\|_2^2 + \lambda\|\mathbf{z}_i\|_1. \quad (1)$$

The resulting sparse codes $\mathbf{z}_i$ were subsequently pooled into a single histogram using mean pooling $\mathbf{h} = \sum_i \mathbf{z}_i w_i$, with w_i being area element of point i. The resulting global shape descriptor $\mathbf{h}$ was use for retrieval using ℓ_1 distance.

The learning process of the dictionary $\mathbf{D}$ was done over a disjoint set of shapes taken from the ones used in Shape-Google [BBGO11]. An initial $\mathbf{D}$ was obtained by unsupervised dictionary learning over randomly selected descriptors from the training set, using the SPAMS toolbox [MBPS09]. Lastly, $\mathbf{D}$ has undergone supervised training using stochastic gradient descent of the loss-function defined in [WS09]. The derivation of (1) was partially inspired by [MBP12]. The size of $\mathbf{D}$ was 64 atoms, and additionally the values of λ and μ were set to 0.5 and 0.3, respectively. Please refer to [LBBC14] for further details.

4.2. Geodesic Distance Distribution

We utilized geodesic distance distributions of mesh model. Per each mesh model in the data set, 500 points are down-sampled. Then, geodesic distances between sampled points and all other vertices in mesh models are calculated, which becomes 500 x N size geodesic distance matrix (GDM). GDM is converted to a histogram having 200 bins. We calculated dissimilarity between different mesh models by applying similarity measure function (GDD-Lp: Euclidean distance; GDD-Corr: Correlation measure) [SFH*09]. This method is rather simple compared to other more advanced techniques, but it was efficient to compute.

4.3. Combined feature with Genetic Algorithm

Our main idea is to carefully select different kind of approaches and integrate them optimally to develop more discriminative and efficient 3D shape retrieval systems. So we extracted three feature descriptors, Heat Kernel Signature (HKS) [SOG09], Wave Kernel Signature (WKS) [ASC11] and Surface Area (SA) [P*14] of the models in the dataset. For each feature, we find a polynomial that better fits all the descriptors in a shape. The evaluation of this polynomial in equi-spaced values gives a descriptor for the shape. Then a genetic algorithm was used to calculate the weight of each feature in the computation of the final distance. Finally, according to the weights, we formed a combination of the three single feature descriptor by linear combination.

For the training step, we used the real dataset provided by Pickup et al. [P*14]. We computed three dissimilarity matrices using the features HKS, WKS and SA. Let S_{HKS}, S_{WKS} and S_{SA} be the dissimilarity matrices for each type of descriptors. We generated a set of chromosomes randomly, and each chromosome contained three real numbers, which represents the weights. We initially chose a random set of weights $[x_{HKS}, x_{WKS}, x_{SA}]$ with $x_{HKS} + x_{WKS} + x_{SA} = 1$ and computed the combined dissimilarity matrix as

$$S_{new} = x_{HKS} \times S_{HKS} + x_{WKS} \times S_{WKS} + x_{SA} \times S_{SA} \quad (2)$$

The fitness of S_{new} used in the genetic algorithm is defined as $f(S_{new}) = -\sum(NN + FT + ST + E\_Measure + DCG)$.

For the fitness computation, the effectiveness measures were computed using S_{new} as input dissimilarity matrix. We found the best chromosome by adjusting the probability of inheritance, selection, crossover, and mutation operator applied over the set of weights. Finally, we kept the best set of weights, which were used to compute the dissimilarity matrix for the current track. It is worth noting that we submit only one run where the corresponding weights were $x_{HKS} = 0.1013$, $x_{WKS} = 0.0745$, $x_{SA} = 0.8242$.

4.4. Signature quadratic form distance on diffusion descriptors

The Signature Quadratic Form Distance [BUS10] provides a flexible way to compare sets of local features between two multimedia objects. The first step is to generate a signature for each object in the dataset. As the track proposes a non-rigid shape retrieval task, we chose the diffusion descriptors as local descriptors, which have proven to be informative and representative in the non-rigid domain. Given a shape O, we compute a set of descriptors F_O, one per each vertex in O. Subsequently, we group the feature set F_O using an adaptive clustering algorithm. The signature of O is the set of tuples

$$S_O = \{(c_i, w_i), i = 1, \ldots, m\} \tag{3}$$

where m is the number of clusters, c_i is the average descriptor in the $i-$th cluster and w_i is a weight proportional to the number of element in the $i-$th cluster with respect to the total number of descriptors in O.

The SQFD computes the distance between two signatures using a similarity function applied to the centroid's clusters. A common choice is the exponential function of a scaled version of the L_p distance. The scale factor α and the ground distance used need to be experimentally tuned for a particular dataset. For more details, we refer to [BUS10].

For this track, we provide two configurations summarized next:

- SQFD-HKS: 100-dimensional HKS [SOG09], ground distance L_1, $\alpha = 0.01$.
- SQFD-SIHKS: 6-dimensional SI-HKS [BK10], ground distance L_1, $\alpha = 0.01$.

For the computation of the diffusion descriptors, we first normalized the area of the objects to one. We computed the Laplace-Beltrami operator using the discretization proposed by Sun [BSW08].

5. Experiments

This section is dedicated to present the experiments and results of our track. First we describe the setup of the competition. Next, we present the effectiveness measures used in our evaluation. Finally we show the results and discussion.

5.1. Setup

Our dataset contains 96,487 objects. The idea is to evaluate the capability of algorithms to retrieve non-rigid objects using this large-scale dataset. Hence, from the total number of objects, only those from the Sumner dataset are considered as queries. The remaining objects are considered as unclassified and are not used as queries. We asked participants to provide a distance matrix of dimension $229 \times 98,487$ for the effectiveness evaluation. We also asked participant to provide detailed information about hardware setup and processing times.

5.2. Evaluation methodology

To measure the effectiveness of submitted methods, we use common information retrieval measures such as

- **Nearest Neighbor (NN):** Given a query, it is the precision at the first object of the retrieved list.
- **First Tier (FT):** Given a query, it is the precision when C objects have been retrieved, where C is the number of relevant objects to the query.
- **Second Tier (ST):** Given a query, it is the precision when 2*C objects have been retrieved, where C is the number of relevant objects to the query.
- **Mean Average Precision (MAP):** Given a query, its average precision is the average of all precision values computed in each relevant object in the retrieved list. Given several queries, the mean average precision is the mean of average precision of each query.

5.3. Results

Effectiveness

Table 2 shows the results obtained for each submitted method. A visual comparison of the same measures is presented in Fig. 1. It is clear that Sparse-BoF consistently performs better in all evaluation measures. Also, the NN measure is perfect, and therefore the method always retrieve a relevant object in the first position of the rank. The second and third best method is SQFD-SIHKS and GA, respectively. This behavior can also be appreciated in the Precision-Recall plot in Fig. 2.

Interestingly, methods that use diffusion descriptors obtain high performance. This seems to indicate the representative power of these kind of descriptors in non-rigid shape retrieval compared to the use of geodesic distances. Clearly, the diffusion-based features are more distinctive, and therefore they provide good scalability in terms of retrieval effectiveness. In contrast, the histogram of geodesic distances obtained a comparatively low performance, which can be attributed to the lower distinctiveness of such representation.

Another important corner for the analysis is the need for a learning step in current approaches. Two approaches

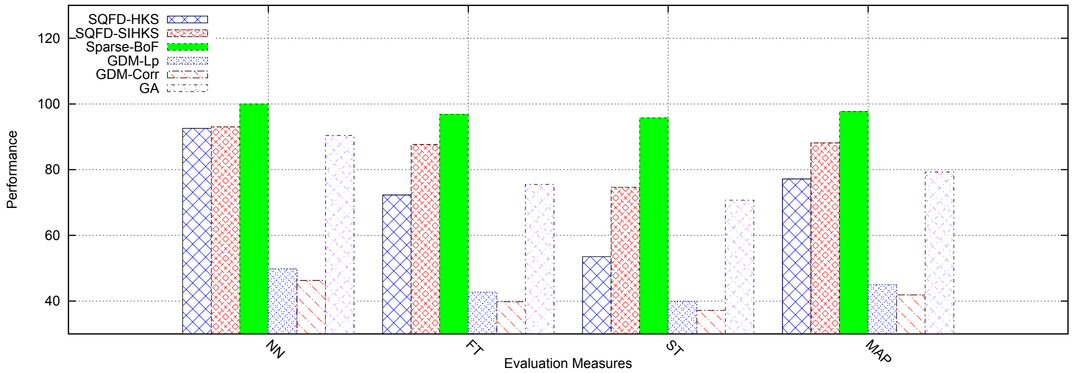

Figure 1: *Effectiveness measures for the submitted methods.*

Method	Variant	NN	FT	ST	MAP
Sparse-BoF	—	**100%**	**96.84%**	**95.77%**	**97.70%**
GDD	GDD-Lp	49.78%	42.72%	39.89%	44.99%
	GDD-Corr	46.28%	39.79%	37.18%	41.89%
GA	—	90.39%	75.52%	70.70%	79.28%
SQFD	SQFD-HKS	92.57%	72.27%	53.53%	77.17%
	SQFD-SIHKS	93.01%	87.66%	74.66%	88.12%

Table 2: *Effectiveness results of submitted methods. We show in bold the best performance for each evaluation measure.*

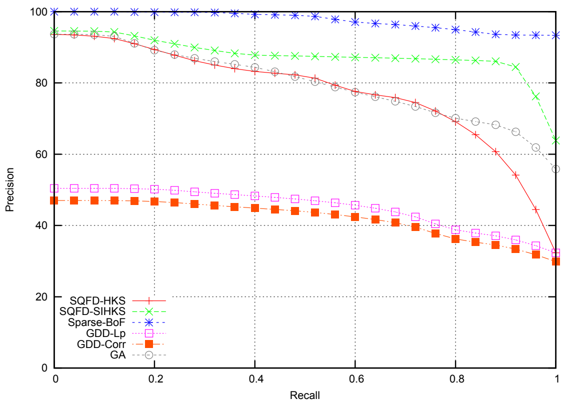

Figure 2: *Precision-recall plot for all runs submitted for our track.*

(Sparse-BoF and GA) requires a learning stage where the algorithm finds optimized parameters for further descriptions. It is obvious that the learning step in Sparse-BoF is an essential part of the scalability of the method in terms of retrieval capabilities. With respect to GA, the optimization of the combination weights presumably ensures the exploitation of several description in conjunction. On the other hand, it is interesting to note that SQFD approaches (which do not require learning) reached a good performance as well. This fact is not new if we consider previous evidence of high performances in the application of SQFD in the image domain [LGS13].

Efficiency

A central point in our evaluation is also the efficiency of retrieval systems. To cope this goal, we asked participants to report the time needed to process the large collection and make the retrieval queries. Obviously, participants used different hardware equipments for the competition, hence the reported times are dependent on the platform. Although these times are not directly comparable, we still believe that this information is valuable to be presented and evaluated. A more convenient way of evaluation is the theoretical complexity of the evaluated algorithms, which is a future direction in this research to enhance the scalability evaluation.

A summary of the system configurations and times are shown in Table 3. We divided the reported times in off-line and on-line processing. Off-line processing contains the time required by the methods to pre-process the entire collection. These pre-processing steps are commonly dedicated to transform the input shape collection into a intermediate representation, which is finally used in the retrieval system. Instead, on-line processing consist of all the required steps to compute a similarity ranking given a query shape. Typically, on-line processing consist of the transformation of the

Method	Equipment	Off-line Proc. Time	On-line Proc. Time
Sparse-BoF	Two 2GHz Xeon Processors - 64GB RAM	**Pre-processing/shape:** 10 sec. **Pre-processing includes:** LBO and its eigendecomposition, descriptor computation and sparse coding. **Dictionary learning:** 10 min. (unsupervised), 3h. (supervised).	**Pre-processing query:** 10 sec. **Query time:** 0.5 sec.
GDD	Intel Core i7-4770 @ 3.40GHz - 16GB RAM	**Pre-processing/shape:** 1.55 sec. **Pre-processing includes:** Geodesic distances computation and the histogram.	**Pre-processing query:** 1.55 sec. **Query time:** 0.62 sec. (GDD-Lp), 0.96 sec. (GDD-Corr)
GA	Two Intel Xeon CPU E5620 Processors - 12 GB RAM	**Preprocessing/shape:** 41 sec. **Pre-processing includes:** computation of HKS, WKS and SA. **Genetic algorithm:** 290 sec.	**Pre-processing query:** 41 sec. **Query time:** 1.27 sec.
SQFD	Intel Core i7-4770 3.40GHz - 32GB RAM	**Pre-processing/shape:** 15 sec. **Pre-processing includes:** LBO and its decomposition, description and signatures computation.	**Preprocessing query:** 15 sec **Query time:** 2 sec.

Table 3: *Summary of processing times reported by track participants.*

input query into the intermediate representation and the subsequent similarity search.

Methods using diffusion descriptors (Sparse-BoF, GA, and SQFD) spent much of the time in the computation of the eigendecomposition of the Laplace-Beltrami operator, which indeed dominates the time measurements in general. In contrast, the histogram of geodesic distances (GDD) is almost one order of magnitude faster than the other methods. This fact evidences a clear trade-off between effectiveness and efficiency. The diffusion-based methods can obtain high effectiveness measures at expenses of a high computation time. It is important to remark that these methods require at least ten seconds to process an input query and deliver a retrieval result. Therefore, the adoption of these methods in real-world application will largely depend on whether users are interested in real-time responses or not.

Methods that use Euclidean distances (Sparse-BoF and GDD-Lp) to perform the final similarity search are faster than the quadratic form counterpart. The SQFD distance is a quadratic form that requires the computation of pair-wise distances between elements of the signature, and therefore it is more time-consuming than performing L_p.

Discussion

If we focus our attention on the effectiveness, we can conclude that current approaches are scalable. At least it holds for the case of non-rigid shape retrieval. From our experiments, one of the most important learned lesson is that the use of diffusion descriptors guarantees a superior performance, even in large-scale datasets. The near-to-perfect performance of Sparse-BoF is also a strong evidence of the maturity of tools for the analysis of non-rigid shapes, which can be exploited for the implementation of high-performance retrieval systems.

Nevertheless, the main concern is regarding scalability in terms of efficiency. In this direction, we noted that if some of the evaluated approaches would be implemented in a retrieval system, users would have to wait at least ten seconds to obtain a result given a query. The computation of diffusion descriptors depends on a time-consuming pre-processing step (eigendecomposition of the Laplace-Beltrami operator) which slows down the retrieval system response. In this sense, we could say that the current approaches are still far from being scalable. On the other hand, the query response time can be considered close to interactive for the studied methods and on our 96.000 object benchmark. This situation may however, change as even larger data sets may become available in the future, e.g., by crowd-sourced and mobile 3D acquisition approaches. Leveraging indexing techniques from the database research domain could further improve the response times of 3D retrieval systems.

6. Conclusions and Future Work

We presented a 3D retrieval benchmark one order of magnitude larger than typical previous benchmarks, to test for scalability of current non-rigid 3D shape retrieval method. We evaluated the scalability in terms of effectiveness and efficiency. For the competition, four teams submitted results which were evaluated using commonly used measures for retrieval systems, as well as the time required to process the dataset and provide a retrieval result. In general, for the case of non-rigid shape retrieval, we can conclude that the use of appropriate descriptors makes systems scalable in terms of effectiveness. This can be appreciated in the near-to-perfect effectiveness achieved by one of the submitted technique. This result encourages us to think that the progress has been fruitful in this area. Nevertheless, it would be important to go one step further and start thinking of more challenging datasets, not only for non-rigid shape retrieval tasks but also for generic retrieval. On the other hand, it is clear that we are far from providing real-time rates to answer a query, mainly due to the time-consuming processing required to pre-process the shapes. Although it may change in the future when more powerful computers will be available, we also should notice that we require to pay attention on more efficient shape analysis tools.

In the future, we plan to enhance the current dataset, including more classes and more non-rigid objects. The idea is to use this benchmark as basis for even more larger datasets. Additionally, we would like to extend the experimentation by analyzing the dinamicity of the dataset. It means, we would like to simulate the behavior of classical database systems which grow with respect to time. The idea is to evaluate the effectiveness and efficiency of current methods and study the performance degradation in this situation.

Acknowledgments

The work of Benjamin Bustos has been partially funded by FONDECYT(Chile) Project 1140783. Alex M. Bronstein is supported by the ERC starting grant 335491 (RAPID). Roee Litman is supported by the Google Europe PhD Fellowship in Machine Learning.

References

[ASC11] AUBRY M., SCHLICKEWEI U., CREMERS D.: The wave kernel signature: A quantum mechanical approach to shape analysis. In *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on* (Nov 2011), pp. 1626–1633. 3

[BBGO11] BRONSTEIN A. M., BRONSTEIN M. M., GUIBAS L. J., OVSJANIKOV M.: Shape google: Geometric words and expressions for invariant shape retrieval. *TOG 30*, 1 (2011), 1–20. 2, 3

[BK10] BRONSTEIN M. M., KOKKINOS I.: Scale-invariant heat kernel signatures for non-rigid shape recognition. In *Proc. CVPR* (2010). 3, 4

[BSW08] BELKIN M., SUN J., WANG Y.: Discrete laplace operator on meshed surfaces. In *Proceedings of the Twenty-fourth Annual Symposium on Computational Geometry* (New York, NY, USA, 2008), SCG '08, ACM, pp. 278–287. 4

[BUS10] BEECKS C., UYSAL M. S., SEIDL T.: Signature quadratic form distance. In *Proceedings of the ACM International Conference on Image and Video Retrieval* (New York, NY, USA, 2010), CIVR '10, ACM, pp. 438–445. 4

[CTSO03] CHEN D.-Y., TIAN X.-P., SHEN Y.-T., OUHYOUNG M.: On visual similarity based 3d model retrieval. *Computer Graphics Forum 22*, 3 (2003), 223–232. 2

[Dat] http://users.dcc.uchile.cl/ isipiran/track.html. 3

[DD13] DADI E., DAOUDI E.: Large scale 3d shape retrieval based on multi-core architectures. In *Networked Systems*, Gramoli V., Guerraoui R., (Eds.), vol. 7853 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2013, pp. 295–299. 2

[GAF*10] GOESELE M., ACKERMANN J., FUHRMANN S., KLOWSKY R., LANGGUTH F., MÜCKE P., RITZ M.: Scene reconstruction from community photo collections. *IEEE Computer 43*, 6 (2010), 48–53. 1

[GDA*09] GODIL A., DUTAGACI H., AKGÜL C. B., AXENOPOULOS A., BUSTOS B., CHAOUCH M., DARAS P., FURUYA T., KREFT S., LIAN Z., NAPOLEON T., MADEMLIS A., OHBUCHI R., ROSIN P. L., SANKUR B., SCHRECK T., SUN X., TEZUKA M., VERROUST-BLONDET A., WALTER M., YEMEZ Y.: SHREC'09 Track: Generic Shape Retrieval. In *Proc. Workshop on 3D Object Retr. (3DOR)* (2009), Spagnuolo M., Pratikakis I., Veltkamp R. C., Theoharis T., (Eds.), Eurographics Association, pp. 61–68. 2

[LBBC14] LITMAN R., BRONSTEIN A., BRONSTEIN M., CASTELLANI U.: Supervised learning of bag-of-features shape descriptors using sparse coding. In *Computer Graphics Forum* (2014), vol. 33, Wiley Online Library, pp. 127–136. 2, 3

[LGB*] LIAN Z., GODIL A., BUSTOS B., DAOUDI M., HERMANS J., KAWAMURA S., KURITA Y., LAVOUÉ G., NGUYEN H. V., OHBUCHI R., OHKITA Y., OHISHI Y., PORIKLI F., REUTER M., SIPIRAN I., SMEETS D., SUETENS P., TABIA H., VANDERMEULEN D.: SHREC '11 Track: Shape Retrieval on Non-rigid 3D Watertight Meshes. pp. 79–88. 2

[LGB*13] LIAN Z., GODIL A., BUSTOS B., DAOUDI M., HERMANS J., KAWAMURA S., KURITA Y., LAVOUÃĽ G., NGUYEN H. V., OHBUCHI R., OHKITA Y., OHISHI Y., PORIKLI F., REUTER M., SIPIRAN I., SMEETS D., SUETENS P., TABIA H., VANDERMEULEN D.: A comparison of methods for non-rigid 3d shape retrieval. *Pattern Recognition 46*, 1 (2013), 449 – 461. 2

[LGS13] LOKOČ J., GROŠUP T., SKOPAL T.: On scalable approximate search with the signature quadratic form distance. In *Similarity Search and Applications*, Brisaboa N., Pedreira O., Zezula P., (Eds.), vol. 8199 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2013, pp. 312–318. 5

[LLL*14a] LI B., LU Y., LI C., GODIL A., SCHRECK T., AONO M., BURTSCHER M., CHEN Q., CHOWDHURY N., FANG B., FU H., FURUYA T., LI H., LIU J., JOHAN H., KOSAKA R., KOYANAGI H., OHBUCHI R., TATSUMA A., WAN Y., ZHANG C., ZOU C.: A comparison of 3D shape retrieval methods based on a large-scale benchmark supporting multimodal queries. *Computer Vision and Image Understanding* (2014). 1

[LLL*14b] LI B., LU Y., LI C., GODIL A., SCHRECK T., AONO M., CHEN Q., CHOWDHURY N. K., FANG B., FURUYA T., JOHAN H., KOSAKA R., KOYANAGI H., OHBUCHI R., TATSUMA

A.: Large Scale Comprehensive 3D Shape Retrieval. In *Eurographics Workshop on 3D Object Retrieval* (2014), Bustos B., Tabia H., Vandeborre J.-P., Veltkamp R., (Eds.), The Eurographics Association. 2

[MBP12] MAIRAL J., BACH F., PONCE J.: Task-driven dictionary learning. *Trans. PAMI 34*, 4 (2012), 791–804. 3

[MBPS09] MAIRAL J., BACH F., PONCE J., SAPIRO G.: Online dictionary learning for sparse coding. In *Proceedings of the 26th Annual International Conference on Machine Learning* (2009), ACM, pp. 689–696. 3

[P*14] PICKUP D., ET AL.: SHREC'14 track: Shape retrieval of non-rigid 3D human models. In *Proc. 3DOR* (2014). 3

[SFH*09] SMEETS D., FABRY T., HERMANS J., VANDERMEULEN D., SUETENS P.: Isometric deformation modelling for object recognition. In *Computer Analysis of Images and Patterns*, Jiang X., Petkov N., (Eds.), vol. 5702 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2009, pp. 757–765. 3

[Ske] https://3dwarehouse.sketchup.com/. 1

[SMB*14] SIPIRAN I., MERUANE R., BUSTOS B., SCHRECK T., LI B., LU Y., JOHAN H.: A benchmark of simulated range images for partial shape retrieval. *The Visual Computer 30* (2014), 1293 – 1308. 2

[SMKF04] SHILANE P., MIN P., KAZHDAN M. M., FUNKHOUSER T. A.: The Princeton Shape Benchmark. In *Proc. Shape Modeling Int.* (2004), IEEE Computer Society, pp. 167–178. 2

[SOG09] SUN J., OVSJANIKOV M., GUIBAS L.: A concise and provably informative multi-scale signature based on heat diffusion. In *Proceedings of the Symposium on Geometry Processing* (Aire-la-Ville, Switzerland, Switzerland, 2009), SGP '09, Eurographics Association, pp. 1383–1392. 3, 4

[SP04] SUMNER R. W., POPOVIĆ J.: Deformation transfer for triangle meshes. *ACM Trans. Graph. 23* (August 2004), 399–405. 2

[SZ03] SIVIC J., ZISSERMAN A.: Video google: a text retrieval approach to object matching in videos. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on* (Oct 2003), pp. 1470–1477 vol.2. 2

[TKA12] TATSUMA A., KOYANAGI H., AONO M.: A large-scale shape benchmark for 3d object retrieval: Toyohashi shape benchmark. In *Signal Information Processing Association Annual Summit and Conference (APSIPA ASC), 2012 Asia-Pacific* (Dec 2012), pp. 1–10. 2

[Tur] http://www.turbosquid.com/. 1

[Vra04] VRANIC D.: *3D Model Retrieval*. PhD thesis, University of Leipzig, 2004. 2

[WKC10] WANG J., KUMAR S., CHANG S.-F.: Semi-supervised hashing for scalable image retrieval. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on* (June 2010), pp. 3424–3431. 2

[WKSS11] WU Z., KE Q., SUN J., SHUM H.-Y.: Scalable face image retrieval with identity-based quantization and multireference reranking. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 33*, 10 (Oct 2011), 1991–2001. 2

[WS09] WEINBERGER K. Q., SAUL L. K.: Distance metric learning for large margin nearest neighbor classification. *JMLR 10* (2009), 207–244. 3