Pacific Graphics (2013)
B. Levy, X. Tong, and K. Yin (Editors)

Non-rigid 3D shape retrieval via sparse representation

L. L. Wan[†1,2] , S. Li[3] , Z. J. Miao[1,2] and Y. G. Cen[1,2]

[1] Institute of Information Science, Beijing Jiaotong University, Beijing, China
[2] Beijing Key Laboratory of Advanced Information Science and Network Technology, Beijing, China
[3] State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing, China

Abstract

Shape descriptor design is an important but challenging problem for non-rigid 3D shape retrieval. Recently, bag-of-words based methods are widely used to integrate a model's local shape descriptors into a global histogram. In this paper, we present a new method to pool the local shape descriptors into a global shape descriptor by means of sparse representation. Firstly, we employ heat kernel signature (HKS) to depict the multi-scale local shape. Then, for each model in the training dataset, we take the HKSs corresponding to its mesh vertices to serve as training signals, and thus an over-complete dictionary can be learned from them. Finally, the HKSs of each 3D model are sparsely coded based on the learned dictionary, and such sparse representations can be further integrated to form an object-level shape descriptor. Moreover, we conduct extensive experiments on the state-of-the-art benchmarks, wherein comprehensive evaluations state our method can achieve better performance than other bag-of-words based approaches.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Curve, surface, solid, and object representations

1. Introduction

3D shape retrieval of plentiful models shared on the Internet has been gaining momentum. The shape descriptors are especially important to discriminate shapes, and have significant influence on the retrieval accuracy.

Recently, more and more researchers have focused their investigations on non-rigid 3D shape analysis. Non-rigid objects are very common in the real world. The most challenging problem in non-rigid 3D shape matching is to maintain invariance to various geometrical changes. Unfortunately, rigid shape analyzing techniques [SMKF04] usually tend to recognize such shape changes as different kinds of objects. With the recent progress of isometry-invariant local shape descriptors, the bag-of-words (BoW) framework has been exploited to integrate a model's local shape descriptors into a global shape descriptor [LGB*11]. The BoW framework is state-of-the-art to the retrieval of images and meshes [DK12]. It starts with the computation of a set of representative vectors named as geometric words. And then, hard or soft assignment [GGVS08] is utilized to compute the distribution of geometric words, resulting in a word histogram as the global shape descriptor.

Heat diffusion is an elegant mathematical tool that is well suitable for the analysis of non-rigid 3D shapes. Heat kernel signature (HKS) [SOG09] has achieved great success with increasing popularity in geometry processing, primarily because of its built-in advantages such as being robust, multi-scale, informative, and invariant to isometric transformations. HKS has been widely studied in non-rigid 3D shape retrieval. For example, Ovsjanikov et al. [OBBG09] defined a compact shape descriptor based on HKS and utilized the BoW framework to generate a global shape descriptor. Bronstein et al. [BK10] proposed a scale-invariant heat kernel signature (SI-HKS) comprising the magnitudes of the Fourier transform, and they also used the BoW framework to construct a global shape descriptor.

In this paper, we propose a novel sparse representation based framework to pool a 3D model's local descriptors into a global shape descriptor. Taking the improved HKS as local shape descriptors, we can learn a redundant descriptor dictionary from the training dataset. Each local descriptor

† Corresponding author: llwan@bjtu.edu.cn

DOI: 10.2312/PE.PG.PG2013short.011-016

can be further approximated by sparse coding, and then, the sparse representations of all local shape descriptors are integrated over the entire shape to form a global shape descriptor. To the best of our knowledge, although sparse representation has many successful applications in image processing [Ela10], it has been seldom used to analyze 3D shapes. Specially, Abdelrahman et al. [AEMF12] proposed a sparse representation based global descriptor called SRCP-TD for non-rigid shape retrieval. For each model, they firstly detected some critical points and concatenated their shape signatures to serve as the global shape descriptor, wherein sparse representation is used to reduce the dimensionality of the global shape descriptor. In sharp contrast, our central idea is to sparsely code all the local shape descriptors of a 3D model, because naively detecting critical points does not work for some common cases (e.g., stained or mutilated 3D models). So our approach is more robust than theirs.

2. Related work

With a point heat source at x on the surface X, the HKS [SOG09] can be defined as

$$h(x,t)=\sum_{i=0}^{\infty} e^{-\lambda_i t}\phi_i^2(x),$$

where λ_i and ϕ_i are the i-th eigenvalue and eigenfunction of the Laplace-Beltrami operator Δ_X.

Ovsjanikov et al. [OBBG09] defined a compact shape descriptor by sampling the HKS in time $t=\alpha^{i-1}t_0$. It is an n-dimensional descriptor vector $\mathbf{p}(x)=(p_1(x),\cdots,p_n(x))^T$, whose elements are

$$p_i(x)=c(x)h(x,\alpha^{i-1}t_0),\ i=1,\cdots,n.$$

They determined the constant $c(x)$ by the constraint $\|\mathbf{p}(x)\|_2=1$, and set $t_0=1024$, $\alpha=1.32$, $n=6$ in the experiments. In this paper, we denote it as C-HKS to distinguish it from other HKS extensions.

Bronstein et al. [BK10] presented a scale-invariant heat kernel signature (SI-HKS). They defined a function to turn the shape scaling into a time-scale shift, and then took the discrete-time Fourier transform (DFT) magnitude to eliminate this time-shift. In [KBY12], by sampling the HKS in time $t=\beta^\tau$ and denoting the HKS as $h(\tau)=h(x,\beta^\tau)$, they proposed a new definition as $\tilde{h}(\tau)=\frac{d}{d\tau}\ln h(\tau)$, and took its DFT modulus $|H(\omega)|$ as a local shape descriptor. Then, the first six frequencies of $|H(\omega)|$ were selected as a compact descriptor in their experiments.

3. Global shape descriptor based on sparse representation

The goal of sparse representation is to represent a given signal by the linear combination of a small number of atom signals in an over-complete dictionary. Based on this theory, we propose a framework to extract a global shape descriptor, which is called Sparse Representation of HKS (SC_{HKS}). As shown in Figure 1, the framework has two main steps: one is to learn the over-complete dictionary from the training dataset with the K-SVD algorithm [AEB06], the other is to integrate the local shape descriptors of a 3D model into a global shape descriptor.

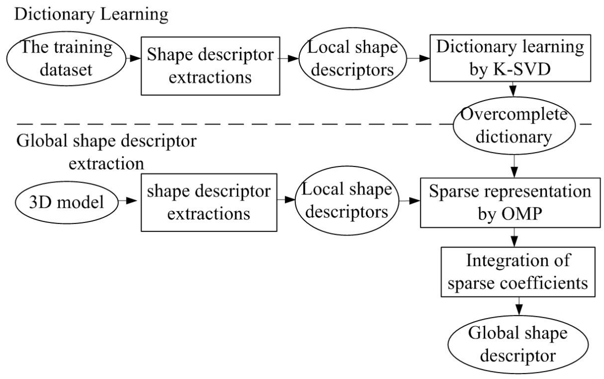

Figure 1: *Overview of our framework*

3.1. Local shape descriptors based on the HKS

Considering the good properties of the HKS, we modify it to fit our framework. Compared with the dimension of a signal, the sparsity should be small enough to ensure the convergence of the K-SVD algorithm. However, since the C-HKS and SI-HKS are both computed at only 6 scales, it is hard to determine a suitable sparsity. We set the sampling time for two reasons: 1) to get a longer shape descriptor; 2) to be adaptive to the global scaling. Therefore, for each 3D model, we compute the first N eigenvalues and eigenvectors, and then sample n points over the time interval $[t_{min},t_{max}]$ with $t_{min}=|4\ln 10/\lambda_N|$, $t_{max}=|4\ln 10/\lambda_2|$. In [SOG09], they explain that the HKS with $t>t_{max}$ remains almost unchanged and the HKS with $t<t_{min}$ needs to compute more eigenvalues and eigenvectors. The sampling time can be formulized as

$$t_i=e^{\mu_i},\quad i=0,\cdots,n-1,$$

where $\mu_i=\ln t_{min}+(\ln t_{max}-\ln t_{min})i/(n-1)$. And then, for a point x, we get a vector $\mathbf{h}(x)=(h(x,t_0),\cdots,h(x,t_{n-1}))$. Finally, after normalized to unit L2 length using $\mathbf{f}(x)=\mathbf{h}(x)/\|\mathbf{h}(x)\|_2$, the local shape descriptor $\mathbf{f}(x)$ is ready for the subsequent steps. The adaptive sampling time and the normalization can collectively guarantee to be invariant to global scaling.

3.2. Dictionary learning

The HKS descriptors are computed at all the vertices on each model from the training dataset, and are taken as the training signals. The K-SVD algorithm is used to train a dictionary from given example data, and has shown excellent performance in many image processing tasks [TF11]. The algorithm accepts a matrix $\mathbf{F}$ whose columns are training signals, the atom signal number K and a sparsity threshold T, and aims to iteratively improve the dictionary to achieve the sparser representations of the training signals by solving the optimization problem

$$\min_{\mathbf{D},\Gamma} \|\mathbf{F}-\mathbf{D}\Gamma\|_2^2 \quad \text{Subject To} \quad \forall i \ \|\gamma_i\|_0 \le T,$$

where Γ is the matrix containing all the sparse representations of training signals, γ_i is the i-th column of the matrix Γ, and $\mathbf{D}$ is a learned dictionary whose columns are atom signals.

We utilize an efficient implementation of the K-SVD algorithm proposed by Rubinstein [RZE08] for good time efficiency. The implementation replaces the exact SVD computation with a much quicker approximation and uses the Batch-OMP method for performing the sparse-coding operations. Orthogonal Matching Pursuit (OMP) [PRK93] is a greedy algorithm to achieve the sparse representation of a signal. The Batch-OMP algorithm is specifically optimized for sparse-coding large sets of signals over the same dictionary.

3.3. Global shape descriptor

Considering that many 3D models consist of as many as tens of thousands of vertices, we still adopt the Batch-OMP algorithm. For all the vertices of each model, we change the sparsity threshold to $T+1$ and compute the sparse coefficients of their HKS descriptors.

Next, to avoid being sensitive to the variation of mesh tessellation, we take the area weight $a(x)$ of the vertex x into account while integrating the sparse representations of local shape signatures over the entire shape X, and construct a global shape descriptor $\mathbf{g}(X)$ which is a $K\times 1$ vector

$$\mathbf{g}(X) = \int_X \gamma(x)da(x).$$

At last, we perform a normalization step as follow

$$\mathbf{g}'(X) = \mathbf{g}(X)/\|\mathbf{g}(X)\|_2.$$

Each element of $\mathbf{g}'(X)$ represents the contribution of the corresponding atom signal to the global descriptor. Using this descriptor, the dissimilarity between two shapes X and Y can be defined as a distance between $\mathbf{g}'(X)$ and $\mathbf{g}'(Y)$ in $\mathbf{R}^K$, e.g., the L1 distance

$$d(X,Y) = \|\mathbf{g}'(X)-\mathbf{g}'(Y)\|_1.$$

4. Results

In order to assess our global shape descriptor (SC_{HKS}), we compared it with two BoW approaches ($BoW_{C\text{-}HKS}$ and $BoW_{SI\text{-}HKS}$) in which C-HKS and SI-HKS are used as local shape descriptors, respectively. In these approaches, the cotangent weight approximation was used to compute the Laplace-Beltrami operator. For C-HKS and SI-HKS, the number of geometric words was set to be 48 as in [BK10]. For C-HKS, we used t=1024,1351,1783,2353,3104,4096 as in [OBBG09]. For SI-HKS, we used a logarithmic scale-space with base $\beta = 2$ and τ ranging from 1 to 25 with increments of 1/16, and took the first six discrete frequencies (these are setting identical to [BK10]). We used the code provided by Bronstein [Bro] to compute C-HKS and SI-HKS. The code provided by Sun [SOG09] is modified for fitting SC_{HKS}.

We evaluated the retrieval performance based on precision-recall curves as well as the following five quantitative measures (see [SMKF04] for details): Nearest Neighbor (NN), First Tier (FT), Second Tier (ST), E-Measure (E), and Discounted Cumulative Gain (DCG).

4.1. Shape retrieval on non-rigid 3D watertight meshes

Our method was firstly tested on 3D watertight meshes. The experiments were on two shape benchmarks: SHREC 2010 non-rigid 3D shape benchmark [LGF*10] and SHREC 2011 non-rigid 3D shape benchmark [LGB*11]. They only contain watertight triangle meshes that are equally classified. The former with 200 meshes is partitioned into 10 classes, and the other with 600 meshes is classified into 30 categories. To create the training dataset, we selected the first two models from each class due to the classification file of each benchmark. We carried out evaluations not only on the average performance of the whole benchmark, but also on the result corresponding to each specific class.

For SHREC 2010 non-rigid 3D shape benchmark, we adopted four groups of parameters for SC_{HKS}, which can all guarantee the convergence of the K-SVD algorithm. In Table 1, the superscripts of SC_{HKS} denote different selections of parameters, which are also in use in the following tables and figures. As shown in Table 1, the four groups of parameters for SC_{HKS} get very close retrieval accuracies, and SC^3_{HKS} performs best among them. Moreover, we can see that it is better to use $N = 300$ eigenvalues and eigenvectors to approximate the heat kernel than $N = 100$, because the number of vertices ranges from 2000 to 30000 in this benchmark. Therefore, we used $N = 300$ for C-HKS and SI-HKS too. From Table 1 and Figure 2, we can find that our approach clearly outperforms $BoW_{C\text{-}HKS}$ and $BoW_{SI\text{-}HKS}$. Figure 3 displays the precision-recall curves measured for all classes. Our approach obtains the best results when searching for most of classes but not *hand*, *spider* and *teddy*.

For SHREC 2011 non-rigid 3D shape benchmark, we have unexpectedly found that the scales of the 3D models are so small that they have bad effects on C-HKS and SI-HKS. According to the descriptions in Section 3.1, too large sam-

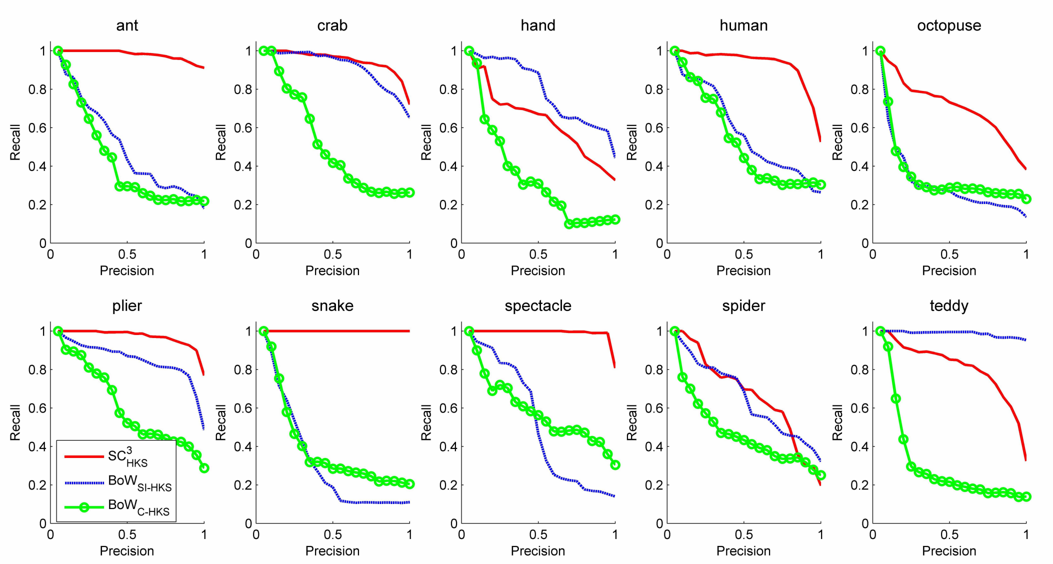

Figure 3: *Precision-recall curves for all the classes in SHREC 2010 non-rigid 3D shape benchmark*

Table 1: *Evaluations on SHREC 2010 non-rigid 3D shape benchmark*

Method	NN	FT	ST	E	DCG
SC^1_{HKS}	0.965	0.736	0.873	0.630	0.924
SC^2_{HKS}	0.945	0.754	0.865	0.626	0.920
SC^3_{HKS}	0.975	0.786	0.899	0.653	0.942
SC^4_{HKS}	0.980	0.756	0.856	0.621	0.929
$BoW_{SI\text{-}HKS}$	0.840	0.548	0.711	0.501	0.827
$BoW_{C\text{-}HKS}$	0.785	0.343	0.549	0.366	0.716

SC^1_{HKS}: $N = 300$, $n = 15$, $K = 50$, $T = 3$
SC^2_{HKS}: $N = 100$, $n = 15$, $K = 50$, $T = 3$
SC^3_{HKS}: $N = 300$, $n = 50$, $K = 100$, $T = 3$
SC^4_{HKS}: $N = 300$, $n = 15$, $K = 50$, $T = 4$

pling time will result in unchanged HKS. Thus, all the sampling time for C-HKS and a considerable part for SI-HKS are inappropriate, so that C-HKS can no longer act as local shape descriptors and SI-HKS can only lead to poor retrieval performance. We still used $N = 300$ for SI-HKS. In Table 2, our approach is compared with $BoW_{SI\text{-}HKS}$ and SRCP-TD. Figure 4 shows the precision-recall curves. From them, we can conclude that our approach achieves better performance than $BoW_{SI\text{-}HKS}$ and SRCP-TD in this benchmark.

Several approaches presented in [LGF*10] and [LGB*11] have better performance than ours. But they can only deal with watertight meshes. Our approach is more robust because it can also deal with non-watertight meshes. We will show the results in the next section.

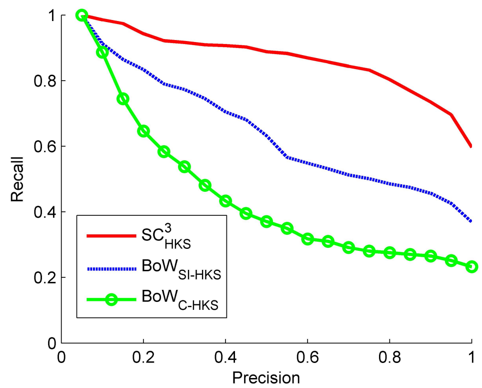

Figure 2: *Precision-recall curves for SHREC 2010 non-rigid 3D shape benchmark*

Table 2: *Evaluations on SHREC 2011 non-rigid 3D shape benchmark (We cited the best retrieval performance of SRCP-TD in this benchmark from [AEMF12])*

Method	NN	FT	ST	E	DCG
SC^1_{HKS}	0.993	0.906	0.951	0.702	0.977
$BoW_{SI\text{-}HKS}$	0.362	0.170	0.259	0.175	0.504
SRCP-TD	0.978	0.811	0.900	0.660	0.947

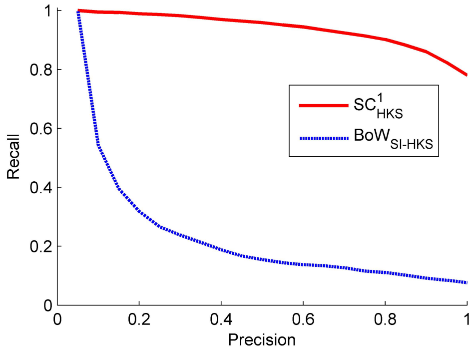

Figure 4: *Precision-recall curves for SHREC 2011 non-rigid 3D shape benchmark*

4.2. Retrieval Robustness

Most meshes are not watertight in the real world. SHREC 2011 robust shape benchmark [BBGO11] is provided for evaluating the retrieval performance on a large-scale dataset. In this benchmark, the meshes without watertight constraints have a wider variety of transformations. The transformations of a shape are split into 12 classes shown in Figure 5.

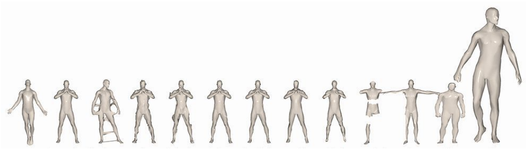

Figure 5: *Transformations of a human shape: null, isometry, topology, noise, shot noise, holes, microholes, sampling, rasterizing, missing parts, view, affine and scale. (cited from [BBGO11])*

Because of no classification file of the whole benchmark, we only used its training dataset containing 684 shapes. The 12 original shapes were taken for training. We put a shape and its 56 transformations into the same class, and got 12 classes. Table 3 reports five quantitative statistics on the average performance, comparing our approach with two BoW approaches. We notice that SC^2_{HKS} performs better than SC^1_{HKS} this time. It is because the minimum number of vertices is 300 in this benchmark, so $N = 300$ is too large for computing the heat kernel. Therefore, we used $N = 100$ for C-HKS and SI-HKS as in [BBGO11]. Our approach still get better performance than $BoW_{SI\text{-}HKS}$ and $BoW_{C\text{-}HKS}$. Similar observations can be made from Figure 6.

4.3. Running time

All the experiments in this section were carried out using MATLAB R2010b on a machine with 2.6GHz dual-core CPU and 3GB RAM.

Table 3: *Evaluations on SHREC 2011 robust shape benchmark*

Method	NN	FT	ST	E	DCG
SC^1_{HKS}	0.845	0.513	0.615	0.501	0.827
SC^2_{HKS}	0.857	0.513	0.623	0.498	0.828
$BoW_{SI\text{-}HKS}$	0.709	0.366	0.504	0.349	0.750
$BoW_{C\text{-}HKS}$	0.636	0.341	0.509	0.315	0.732

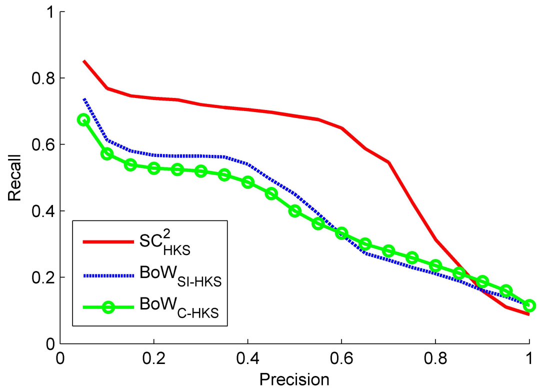

Figure 6: *Precision-recall curves for the SHREC 2011 robust shape benchmark*

We firstly measured the time of the HKS computing for all the vertices on a 3D model. Table 4 shows the results for three models (T186, T78 and T130) from SHREC 2010 non-rigid 3D shape benchmark. For each model, we computed 100 or 300 eigenvalues and eigenvectors, and sampled 15 or 50 points over the time. From the results, we find that the time cost is much lower when using $N = 100$, and is only slightly increased by sampling more points.

Table 4: *Time for computing the HKS*

#vertices	$N = 100$		$N = 300$	
	$n = 15$	$n = 50$	$n = 15$	$n = 50$
5160	3.8s	3.9s	19.4s	19.6s
10348	11.7s	12.2s	41.6s	42.2s
23547	34.1s	34.4s	101.5s	102.0s

Table 5 shows the time of dictionary learning after 20 iterations, corresponding to the different size of training signal matrix **F**.

We selected the same models as in the HKS computing experiment. Table 6 shows the time for constructing the SC_{HKS}.

Table 5: *Time for training the dictionary*

Benchmark	#F	K-SVD
SHREC 2011 non-rigid	15×263045	69.7s
SHREC 2010 non-rigid	15×561984	148.3s
SHREC 2011 robust	15×18049	4.5s

Table 6: *Time for constructing the global descriptor*

Model	#F	Time
T186	15×5160	0.04s
T78	15×10348	0.07s
T130	15×23547	0.19s

5. Conclusions

In this paper, we have presented a sparse representation based framework to integrate local shape descriptors into a global shape descriptor for non-rigid 3D shape retrieval. The key idea is to sparsely code all the HKSs of a 3D model over a learned dictionary. Under a certain sparsity constraint, the representation of HKS only use a few representative atom signals, and it is good for forming a more distinguished global shape descriptor. Our framework is possible to apply other local shape descriptors for non-rigid 3D shape retrieval. However, we note that the parameters should not be selected blindly, and they should guarantee the convergence of the K-SVD algorithm. In addition, using a few samples from each class for dictionary leaning may give rise to the incomplete dictionary for some classes, which will be studied in the future.

Acknowledgment

This work is supported by Beijing Natural Science Foundation (4123104), Fundamental Research Funds for the Central Universities (2011JBM224), National 973 Key Research Program of China (2011CB302203), NSFC (61273274, 61272028) and Tsinghua-Tencent Joint Lab for IIT.

References

[AEB06] AHARON M., ELAD M., BRUCKSTEIN A.: K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on Signal Processing 54*, 11 (2006), 4311–4322. doi:10.1109/TSP.2006.881199. 2

[AEMF12] ABDELRAHMAN M., EL-MELEGY M., FARAG A.: Heat kernels for non-rigid shape retrieval: Sparse representation and efficient classification. In *Computer and Robot Vision (CRV), 2012 Ninth Conference on* (2012), pp. 153–160. doi:10.1109/CRV.2012.28. 2, 4

[BBGO11] BRONSTEIN A. M., BRONSTEIN M. M., GUIBAS L. J., OVSJANIKOV M.: Shape google: Geometric words and expressions for invariant shape retrieval. *ACM Trans. Graph. 30*, 1 (2011), 1:1–1:20. doi:10.1145/1899404.1899405. 5

[BK10] BRONSTEIN M., KOKKINOS I.: Scale-invariant heat kernel signatures for non-rigid shape recognition. In *2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2010), pp. 1704–1711. doi:10.1109/CVPR.2010.5539838. 1, 2, 3

[Bro] BRONSTEIN M.:. URL: http://vision.mas.ecp.fr/Personnel/iasonas/code/sihks.zip. 3

[DK12] DAROM T., KELLER Y.: Scale-invariant features for 3-d mesh models. *IEEE Transactions on Image Processing 21*, 5 (2012), 2758–2769. doi:10.1109/TIP.2012.2183142. 1

[Ela10] ELAD M.: *Sparse and redundant representations: From Theory to Applications in Signal and Image Processing*. Springer, 2010. 2

[GGVS08] GEMERT J. C., GEUSEBROEK J.-M., VEENMAN C. J., SMEULDERS A. W.: Kernel codebooks for scene categorization. In *Proceedings of the 10th European Conference on Computer Vision: Part III* (Berlin, Heidelberg, 2008), ECCV '08, Springer-Verlag, pp. 696–709. doi:10.1007/978-3-540-88690-7_52. 1

[KBY12] KOKKINOS I., BRONSTEIN M. M., YUILLE A.: *Dense Scale-Invariant Descriptors for Images and Surfaces*. Tech. rep., 2012. 2

[LGB*11] LIAN Z., GODIL A., BUSTOS B., DAOUDI M., HERMANS J., KAWAMURA S., KURITA Y., LAVOUÉ G., NGUYEN H., 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. In *Proceedings of the 4th Eurographics conference on 3D Object Retrieval* (2011), pp. 79–88. doi:10.2312/3DOR/3DOR11/079-088. 1, 3, 4

[LGF*10] LIAN Z., GODIL A., FABRY T., FURUYA T., HERMANS J., OHBUCHI R., SHU C., SMEETS D., SUETENS P., VANDERMEULEN D., WUHRER S.: Shrec'10 track: non-rigid 3d shape retrieval. In *Proceedings of the 3rd Eurographics conference on 3D Object Retrieval* (Aire-la-Ville, Switzerland, Switzerland, 2010), EG 3DOR'10, Eurographics Association, pp. 101–108. doi:10.2312/3DOR/3DOR10/101-108. 3, 4

[OBBG09] OVSJANIKOV M., BRONSTEIN A., BRONSTEIN M., GUIBAS L.: Shape google: a computer vision approach to isometry invariant shape retrieval. In *2009 IEEE 12th International Conference on Computer Vision Workshops (ICCV Workshops)* (2009), pp. 320–327. doi:10.1109/ICCVW.2009.5457682. 1, 2, 3

[PRK93] PATI Y., REZAIIFAR R., KRISHNAPRASAD P. S.: Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition. In *The Twenty-Seventh Asilomar Conference on Signals, Systems and Computers* (1993), pp. 40–44 vol.1. doi:10.1109/ACSSC.1993.342465. 3

[RZE08] RUBINSTEIN R., ZIBULEVSKY M., ELAD M.: *Efficient Implementation of the K-SVD Algorithm using Batch Orthogonal Matching Pursuit*. Tech. rep., CS Technion, 2008. 3

[SMKF04] SHILANE P., MIN P., KAZHDAN M., FUNKHOUSER T.: The princeton shape benchmark. In *Proceedings of the IEEE conference on shape modeling applications* (2004), pp. 167–178. doi:10.1109/SMI.2004.1314504. 1, 3

[SOG09] SUN J., OVSJANIKOV M., GUIBAS L.: A concise and provably informative multi-scale signature based on heat diffusion. *Computer Graphics Forum 28*, 5 (2009), 1383–1392. doi:10.1111/j.1467-8659.2009.01515.x. 1, 2, 3

[TF11] TOSIC I., FROSSARD P.: Dictionary learning. *IEEE Signal Processing Magazine 28*, 2 (2011), 27–38. doi:10.1109/MSP.2010.939537. 3