EUROGRAPHICS 2014 / Thomas Funkhouser and Shi-Min Hu
(Guest Editors)

Volume 33 (2014), Number 5

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

Roee Litman[1], Alex Bronstein[1], Michael Bronstein[2] and Umberto Castellani[3]

[1]School of Electrical Engineering at Tel-Aviv University
[2]Faculty of Informatics at University of Lugano
[3]Department of Computer Science at University of Verona

Abstract

We present a method for supervised learning of shape descriptors for shape retrieval applications. Many content-based shape retrieval approaches follow the bag-of-features (BoF) paradigm commonly used in text and image retrieval by first computing local shape descriptors, and then representing them in a 'geometric dictionary' using vector quantization. A major drawback of such approaches is that the dictionary is constructed in an unsupervised manner using clustering, unaware of the last stage of the process (pooling of the local descriptors into a BoF, and comparison of the latter using some metric). In this paper, we replace the clustering with dictionary learning, where every atom acts as a feature, followed by sparse coding and pooling to get the final BoF descriptor. Both the dictionary and the sparse codes can be learned in the supervised regime via bi-level optimization using a task-specific objective that promotes invariance desired in the specific application. We show significant performance improvement on several standard shape retrieval benchmarks.

1. Introduction

The recent advance of 3D acquisition and printing technology has been an important driver to the growth of large databases of 3D models, bringing with it an increased interest in efficient methods for shape retrieval [OLGM11, MWZ*13]. Shape retrieval if probably a field where the proverbial 'one picture is worth a thousand words' is very true: while it is very hard to provide a textual description of a geometric shape, it is easy to provide an example of a similar shape. Therefore, significant research of shape retrieval has been focusing on *content-based* approaches, where the query is a shape, and the search algorithm tries to find similar shapes to the query.

Shape retrieval methods rely on some shape representation (*signature* or *descriptor*) able to capture the most distinctive shape properties for retrieval purposes, while being insensitive to 'noise' (in the broad sense, which can include e.g. inelastic deformation, acquisition artifacts, etc). Then, the similarity between two shapes is determined by the similarity between their respective descriptors. Methods like [ASYS10, LN08, LWW*10] that further optimize this similarity measure for retrieval are beyond the scope of this paper.

Related works Shape descriptors are a popular and important topic of research in the geometry community, with numerous efficient methods (see [Let al.13] for a recent survey). In general, descriptors can be categorized as *global* or *local*. Global descriptors characterize the whole 3D model [FKMS05, GLWT13], while local ones refer to object parts like points or regions [SOG09, BK10]. Some methods build the signature directly over local features, keeping some relative geometric data. One recent examples of such an approach is [HSG13], which selects discriminative volumetric features over pre-aligned shapes.

Typically, a global shape descriptor can be constructed in a bottom-up manner, by aggregating local descriptors. A standard way of doing it is the *bag-of-features* (BoF) paradigm, inspired by bag-of-words methods in text retrieval where text documents are represented by the frequency of appearance of single words from a fixed dictionary. This method was successfully applied to images and video [SZ03, CDF*04] and, more recently, to 3D shape description [BBGO11, TCF10, DK12, Lav12, LGSX13]. The geometric equivalent of 'words' are local feature descriptors, which are quantized in a representative collection of descriptors ('geometric dictionary') to obtain the 'bag-of-geometric words'. Several approaches use point descriptors [DK12, BBGO11, Lav12], regions [TCF10], or partial views [LGSX13]. Moreover, different variants of the original BoF paradigm have appeared to exploit hierarchical structures of the shape like pyramid matching [GD05], spatial relationships [DK12, BBGO11, Lav12], or a combination of the two, usually known as a spatial pyramid [LH13a, LSGFRC*13, LSP06]. Other methods exploit further text-inspired approaches by defining

relations between 'geometric words'. For instance, in [BNJ03, Hof01, BBGO11] the concept of visual *topic* as 'words' co-occurrence is introduced. In [JP12], topological relations between 'words' are used by imposing a grid structure for the involved topics, the so called counting grid. Such approaches demonstrated successfully results in both 2D and 3D object matching [Hof01, LZQ06, JP12].

In all aforementioned methods, feature quantization is performed by an unsupervised clustering procedure using standard k-means algorithms [DHS01], after which the cluster centroids are retained as the 'words' of the dictionary. This procedure is completely agnostic to the pooling into a histogram that is subsequently applied to the quantized descriptors (Figure 1, left). The clustering, and therefore the dictionary construction, is performed without using information about the shape class labels. Typically, for classification purposes the discriminative process is introduced at a later stage by a discriminative classifier, such as support vector machine (SVM) [Vap98] or similarity-sensitive hashing [BBGO11].

In this paper, we propose a new supervised BoF framework with the discriminative training introduced already at the dictionary construction step. To this end, a sparse coding approach is exploited as an alternative to the standard vector quantization strategy. Sparse coding is a *generative* approach representing a "signal" as a linear combination of predefined atoms of a dictionary. Sparse coding methods have been used in discriminative tasks [MBP*08, BC13, HFL12] using a *reconstuctive* approach, where one first trains a dictionary per class, and then the representation of an object is attempted in each of the dictionaries, and the class label is assigned according to the dictionary in which the smallest representation error is obtained [BC13]. Alternatively, residual errors from all the dictionaries can be collected into a global descriptor that can be subsequently used for discriminative classification [SH06]. In both cases, the dictionaries are constructed in an unsupervised data-driven fashion by minimizing the representation error on the training data. Therefore, dictionary learning can simply be viewed as an extension of the k-means clustering [AEB06].

In order to introduce discriminative learning into sparse coding, a *supervised* dictionary learning procedure was proposed in [MBP*08, WYNH13]. The authors trained class-specific dictionaries with the objective to minimize the representation error for a given class, while maximizing it for the rest of the classes. Similar ideas were used by [LR09] to construct class-specific dictionaries for vector quantization-based representation. Class labels are assigned, as before, based on the smallest representation error. While shape retrieval (and any content-based retrieval in general) can in principle be viewed as binary classification of pairs of shapes into positives (similar) and negatives (dissimilar), in contrast to standard classification problems, the descriptors have to be computed ahead of time for each of the shapes individually. This makes impossible to use of the residual error as a means to produce class labels.

Main contributions. The main contribution of the present paper is a task-specific dictionary learning approach tailored for retrieval problems. We follow the spirit of [MBP12] learning the dictionary that explicitly enforces a margin separating the distances between the bag-of-feature descriptors computed on knowingly positive and negative training pairs of shapes. In contrast to standard unsupervised dictionary learning aiming at minimizing the reconstruction error, we optimize a task-specific objective that takes into account the encoding of the local geometry descriptors, their pooling into a global shape descriptor, and the comparison of the latter descriptor using some standard metric. The proposed approach can also be interpreted as supervised metric learning, with two key advantages. First, unlike the majority of metric learning approaches that use a linear transformation of the data, ours is non-linear, allowing to learn more complicated metrics. Second, unlike most existing nonlinear metric learning techniques, our approach does not require any out-of-sample extension procedures.

We show experimentally that the supervised construction of shape descriptors can (sometimes, significantly) increase the performance of popular shape retrieval approaches.

2. Background

We model the shape as a two-dimensional manifold S sampled at n points $s_1,\ldots,s_n$ and represented as a triangular mesh. We denote the Laplace-Beltrami operator of S by Δ_S, and discretize it using the cotangent formula [PP93]. The eigenfunctions and the eigenvalues of the Laplace-Beltrami operator $\Delta_S\phi_l = \lambda_l\phi_l$ are denoted by $\{\phi_l,\lambda_l\}_{l\geq 1}$. The heat kernel associated with Δ_S is given by

$$h_t(s_i,s_j) = \sum_{l\geq 1} e^{-\lambda_l t}\phi_l(s_i)\phi_l(s_j). \qquad (1)$$

2.1. Local descriptors

Local descriptors try to represent the geometric structure of the shape in a small neighborhood of a point. In some cases, feature description is preceded by feature detection, which subsamples the surface at a repeatably detectable subset of points; in the following, we assume w.l.o.g. that the descriptor is dense, and each point s_i is associated with a q-dimensional local descriptor $\mathbf{x}(s_i) = (x_1(s_i),\ldots,x_q(s_i))^\top$. There exists a plethora of methods for local shape description; we outline below two popular spectral descriptors that are later employed in our experiments.

HKS. Ovsjanikov et al. [SOG09] used the diagonal of the heat kernel taken at q log-sampled time values $t = \alpha^\tau$ as a local intrinsic feature descriptor referred to as the *heat kernel*

signature (HKS)

$$\mathbf{x}(s_i) = (h_{\alpha^{\tau_1}}(s_i,s_i),\ldots,h_{\alpha^{\tau_q}}(s_i,s_i))^\top. \quad (2)$$

Note that HKS is not invariant to shape scaling transformations.

SI-HKS. Bronstein and Kokkinos [BK10] developed a scale-invariant version of the HKS by first constructing a *scale-covariant heat kernel*

$$\bar{h}_\tau(s_i,s_i) = \frac{-\sum_{l\geq 1}\lambda_l\alpha^\tau\log\alpha e^{-\lambda_l\alpha^\tau}\phi_l^2(s_i)}{\sum_{l\geq 1}e^{-\lambda_l\alpha^\tau}\phi_l^2(s_i)} \quad (3)$$

that undergoes shift in τ by $2\log_\alpha c$ as a result of shape scaling by a factor of c. In the Fourier domain, this shift results in a complex phase $\bar{H}(\omega)e^{-i\omega 2\log_\alpha c}$, where $\bar{H}(\omega)$ denotes the Fourier transform of $\bar{h}_\tau$ w.r.t. τ. Finally, the scale-invariant HKS (SI-HKS) descriptor is constructed by taking the absolute value of $H(\omega)$ (thus undoing the phase) and then sampling $|H(\omega)|$ at q frequencies,

$$\mathbf{x}(s_i) = (|H(\omega_1)|,\ldots,|H(\omega_q)|)^\top. \quad (4)$$

2.2. Bag-of-features

Given a set of local q-dimensional descriptors computed w.l.o.g. at all the n points of the shape, we represent them as a $q\times n$ matrix

$$\mathbf{X} = (\mathbf{x}_1,\ldots,\mathbf{x}_n) = (\mathbf{x}(s_1),\ldots,\mathbf{x}(s_n)).$$

A bag-of-features is a global shape descriptor constructed by replacing the local descriptors with closest entries in a geometric dictionary and then computing the frequency of appearance of these geometric words, as shown in Figure 1 (top).

Geometric dictionary is a $q\times v$ matrix $\mathbf{D} = (\mathbf{d}_1,\ldots,\mathbf{d}_v)$ whose columns are 'representative' descriptors referred to as geometric words or atoms. The geometric dictionary is constructed offline using a large collection of shapes, by clustering the respective descriptors (points in q-dimensional descriptor space) into v Voronoi regions using, e.g., the k-means algorithm.

Quantization. Given a dictionary $\mathbf{D}$, each local descriptor $\mathbf{x}$ is replaced by the closest entry

$$i^* = \arg\min_{i=1,\ldots,v}\|\mathbf{x}-\mathbf{d}_i\|_2$$

in the geometric dictionary, which can be represented as the v-dimensional *code* vector $\mathbf{z}^*$ containing one at the i^*-th position and zeros elsewhere. This process is known as *vector quantization* (VQ) and can be posed as the problem of constrained sparse coding

$$\mathbf{Z}^*(\mathbf{X},\mathbf{D}) = \arg\min_{\mathbf{Z}\in\{0,1\}^{v\times n}}\|\mathbf{X}-\mathbf{D}\mathbf{Z}\|_{\mathrm{F}} \text{ s.t. } \mathbf{Z}^\top\mathbf{1}=\mathbf{1}, \quad (5)$$

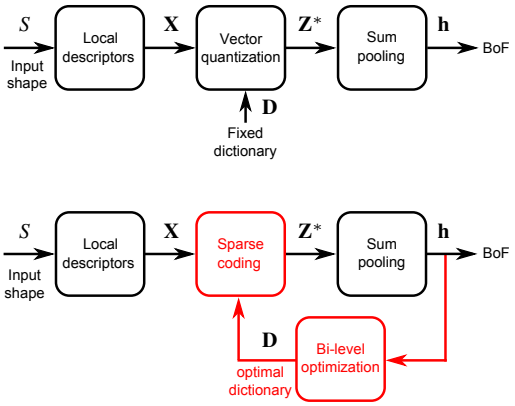

Figure 1: *Top: a flow diagram of a traditional BoF framework using VQ in a fixed dictionary. Bottom: flow diagram of the proposed framework. VQ is replaced by sparse coding, and the dictionary is learned by a bi-level optimization scheme that tries to maximize the discriminativity of the resulting BoFs on a training set.*

in which the codes are binary and are allowed to have only one non-zero element. The output of VQ is a $v\times n$ matrix $\mathbf{Z}^*$ containing the v-dimensional code for each shape point.

Pooling. Finally, the codes are pooled into a single v-dimensional bag-of-features vector $\mathbf{h}(\mathbf{X},\mathbf{D}) = \mathbf{Z}^*(\mathbf{X},\mathbf{D})\mathbf{p}$, where, in the simplest case, $\mathbf{p} = \frac{1}{n}\mathbf{1}$ (mean pooling). In this case, $\mathbf{h}$ can be regarded as the frequency of appearance of different geometric words on the shape. More accurately, the pooling should account for possible non-uniform sampling, weighting each point by its area element a_i, $\mathbf{p} = (a_1,\ldots,a_n)^\top/\sum_{i=1}^n a_i$. Finally, more elaborate weighting can also account for the overall frequency of the words, downweighing common words (a strategy referred to as term frequency-inverse document frequency, or *tf-idf* [SZ03]).

The main drawback of the standard BoF construction outlined above is that all the stages are performed independently. In particular, the dictionary construction is unaware of the following quantization and pooling stages. As a result, even though the local descriptors may show good invariance under the desired class of transformations, the final BoFs may differ significantly (consider a pathological case where the descriptors are close the boundaries of the Voronoi cells in the descriptor space and, due to noise and numerical inaccuracies, are quantized to very different code vectors).

3. Learning BoFs

The key idea of this paper is to revisit the aforementioned BoF construction procedure, performing it in a supervised manner. First, we replace the VQ stage with sparse coding. Second, the unsupervised dictionary learning is replaced with supervised learning maximizing the end-to-end retrieval performance. The flow of the proposed method is depicted in Figure 1 (bottom).

Figure 2: *Visualization of $\mathbf{Z}^*$ based on unsupervised dictionary learning. Each column represents a different dimension (atom) of $\mathbf{Z}^*$, and each row includes a different shape from the synthetic part of SHREC'14 data-set [P*14]. The top two rows are approximate isometric deformations of the same shape, while the bottom row is a different shape. The values of $\mathbf{Z}^*$ are color-mapped from zero (white) to high values (red). Note that the two leftmost atoms capture the specific pose of the shape rather than begin isometry agnostic. This effect is remedied when supervision is introduced (see Figure 3).*

Figure 3: *The same three shape presented in Figure 2, this time using supervised learning. Note that each atom has some discriminative power.*

Sparse coding. Given an overcomplete $q \times v$ dictionary $\mathbf{D}$ ($v > q$), the VQ procedure (5) can be replaced by solving the standard synthesis pursuit problem[†]

$$\mathbf{Z}^*(\mathbf{X},\mathbf{D}) = \arg\min_{\mathbf{Z}} \frac{1}{2}\|\mathbf{X}-\mathbf{D}\mathbf{Z}\|_{\mathrm{F}}^2 + \lambda\|\mathbf{Z}\|_1 + \frac{\lambda_2}{2}\|\mathbf{Z}\|_{\mathrm{F}}^2, \quad (6)$$

producing for each q-dimensional descriptor vector $\mathbf{x}_i$ a v-dimensional sparse code $\mathbf{z}_i^*$. Note that, unlike VQ, now each column of $\mathbf{Z}^*(\mathbf{X}) = (\mathbf{z}^*(\mathbf{x}_1),\ldots,\mathbf{z}^*(\mathbf{x}_n))$ contains a few non-zero coefficients with arbitrary magnitudes.

Unsupervised dictionary learning. Since the sparse codes $\mathbf{Z}^*$ depend on the dictionary $\mathbf{D}$, one may add the dictionary as an optimization variable to (6), resulting in the non-convex problem [AEB06, EAHH99]

$$\mathbf{Z}^*(\mathbf{X}) \;=\; \arg\min_{\mathbf{Z},\mathbf{D}} \frac{1}{2}\|\mathbf{X}-\mathbf{D}\mathbf{Z}\|_{\mathrm{F}}^2 + \lambda\|\mathbf{Z}\|_1 + \frac{\lambda_2}{2}\|\mathbf{Z}\|_{\mathrm{F}}^2, \quad (7)$$

which can be interpreted as a matrix factorization problem, in which $\mathbf{X}$ is approximated by the product of the left factor $\mathbf{D}$ and the sparse right factor $\mathbf{Z}$. Note that such dictionary learning (DL) is unsupervised (the optimal dictionary tries to minimize the reconstruction error), and, thus, is again agnostic to the subsequent pooling of the code vectors and the use of the resulting bags of features in classification or retrieval tasks. An example of sparse coding based on unsupervised DL can be seen in Figure 2.

Bi-level supervised dictionary leaning. Let S and S_+ be two shapes from the same class, possibly affected by some transformation, and S_- be a shape from a different class (for example, S is a human, S_+ a non-rigid deformation thereof, and S_- is a dog; see examples shown in Figure 5). We refer to the pair S, S_+ as *positives* and to S, S_- as *negatives*, and denote the corresponding descriptor matrices of sizes $q \times n$, $q \times n_\pm$ by $\mathbf{X}$ and $\mathbf{X}_\pm$, respectively.

Ideally, we would like the BoFs of the positives to be similar and those of the negatives to be dissimilar, i.e., make $\|\mathbf{h}(\mathbf{X}) - \mathbf{h}(\mathbf{X}_+)\|$ as small as possible, while keeping $\|\mathbf{h}(\mathbf{X}) - \mathbf{h}(\mathbf{X}_-)\|$ as large as possible. This can be achieved by minimizing the loss

$$\mathcal{L} = \sum_{\mathbf{X},\mathbf{X}_+,\mathbf{X}_- \in \mathcal{T}} \ell(\mathbf{X},\mathbf{X}_+,\mathbf{X}_-) \quad (8)$$

over all the triplets $\mathbf{X},\mathbf{X}_+,\mathbf{X}_-$ in a given training set $\mathcal{T}$, where $\ell = \alpha\ell_+ + (1-\alpha)\ell_-$ with

$$\begin{aligned} \ell_+(\mathbf{X},\mathbf{X}_+) &= \|\mathbf{h}(\mathbf{X}) - \mathbf{h}(\mathbf{X}_+)\|_1, \\ \ell_-(\mathbf{X},\mathbf{X}_+,\mathbf{X}_-) &= \max\{\mu, \|\mathbf{h}(\mathbf{X}) - \mathbf{h}(\mathbf{X}_+)\|_1 \\ &\qquad - \|\mathbf{h}(\mathbf{X}) - \mathbf{h}(\mathbf{X}_-)\|_1\}. \end{aligned}$$

The term ℓ_-, known as the *hinge loss*, tries to achieve a separation by at least μ between the dissimilarity of the positive and the negative pair [WS09]. The parameter $\alpha \geq 0$ sets the tradeoff between the two losses, allowing to control the tradeoff between the false positive and the false negative rates.

[†] Note that we use a slightly modified version of the classical pursuit problem, for the following reason: the addition of the λ_2-term makes the problem strictly convex, meaning that it has a unique minimizer. Therefore, $\mathbf{Z}^*(\mathbf{X})$ defines a bijection from the space of descriptors to the space of their sparse codes. In practice, λ_2 can assume vanishingly small values.

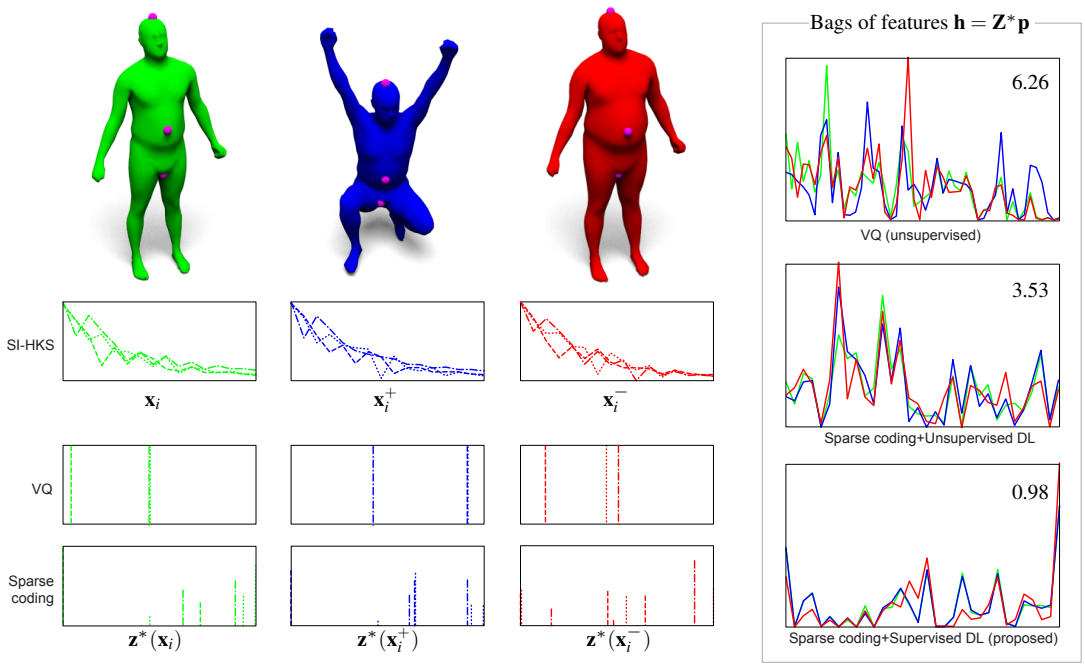

Figure 4: *Example of BoF construction. Green and blue are positives (two near-isometric deformations of the same person), while red is a negative (a different person; note that the difference between the persons is hard to notice even for a human observer). Left, from top to bottom: local SI-HKS descriptors of three representative points on the belly (dotted), groin (dashed) and head (dash-dotted; colors represent different shapes); vector quantization of the local descriptors in a fixed dictionary; sparse coding of the local descriptors in an optimal task-specific dictionary computed by the proposed procedure. Right, top to bottom: BoF using standard VQ, sparse coding with unsupervised DL, and the proposed sparse coding with supervised DL. Ideally, the green and blue BoFs should coincide, while the red one should be distinct. Numbers represent the ratio* $\|\mathbf{h}(\mathbf{X}) - \mathbf{h}(\mathbf{X}_+)\|_1 / \|\mathbf{h}(\mathbf{X}) - \mathbf{h}(\mathbf{X}_-)\|_1$ *(the smaller the better).*

Note that in the above expressions, the BoFs $\mathbf{h} = \mathbf{Z}^*\mathbf{p}$ depend on the codes $\mathbf{Z}^*$, which in turn depend on the dictionary $\mathbf{D}$. Therefore, supervised dictionary learning results in a *bi-level minimization* problem [CMS07]

$$\min_{\mathbf{D}} \sum_{\mathbf{X},\mathbf{X}_+,\mathbf{X}_- \in \mathcal{T}} \ell(\mathbf{Z}^*(\mathbf{X},\mathbf{D}),\mathbf{Z}^*(\mathbf{X}_+,\mathbf{D}),\mathbf{Z}^*(\mathbf{X}_-,\mathbf{D})), \quad (9)$$

which depends on the minimizer of (6). The solution of problem (9) produces a *task-specific* dictionary that optimally (in the sense of the loss ℓ) separates between the BoFs of positive and negative pairs. An example of sparse-coding based on supervised DL can be seen in Figure 3.

An example of all the stages of the BoF construction is shown in Figure 4.

4. Numerical optimization

In order to solve problem (9) we need to compute the gradients of the loss $\mathcal{L}$ with respect to the dictionary $\mathbf{D}$. Since $\mathcal{L}$ consists of a sum of losses $\ell(\mathbf{X},\mathbf{X}_+,\mathbf{X}_-)$ given for a triplet $\mathbf{X},\mathbf{X}_+,\mathbf{X}_-$, we henceforth consider the gradient of an individual loss ℓ. It is well-established in [MBP12] that the map $\mathbf{Z}^*(\mathbf{X},\mathbf{D})$ is almost everywhere differentiable with respect to $\mathbf{D}$. Denoting by Λ the active set of $\mathbf{Z}^* = \mathbf{Z}^*(\mathbf{X},\mathbf{D})$ (i.e., the set of indices at which it attains non-zero values), we define

$$\beta_\Lambda = (\mathbf{D}_\Lambda^\top \mathbf{D}_\Lambda + \lambda_2 \mathbf{I}_\Lambda)^{-1} (\nabla_{\mathbf{Z}} \ell)_\Lambda, \quad (10)$$

where

$$\nabla_{\mathbf{Z}} \ell = \alpha \nabla_{\mathbf{Z}} \|\mathbf{Z} - \mathbf{Z}_+\|_1 + (1-\alpha) \nabla_{\mathbf{Z}} \max\{\mu, \|\mathbf{Z} - \mathbf{Z}_+\|_1 - \|\mathbf{Z} - \mathbf{Z}_-\|_1\} \quad (11)$$

is the gradient of the loss function with respect to $\mathbf{Z}$. The elements of β outside Λ are set to zero. Similarly, one can construct $\beta_\pm$ for $\mathbf{Z}_\pm^* = \mathbf{Z}^*(\mathbf{X}_\pm,\mathbf{D})$, with

$$\begin{aligned} \nabla_{\mathbf{Z}_-} \ell &= (1-\alpha) \nabla_{\mathbf{Z}_-} \max\{\mu, \|\mathbf{Z} - \mathbf{Z}_+\|_1 - \|\mathbf{Z} - \mathbf{Z}_-\|_1\} \\ \nabla_{\mathbf{Z}_+} \ell &= \alpha \nabla_{\mathbf{Z}_+} \|\mathbf{Z} - \mathbf{Z}_+\|_1 + \\ &\quad (1-\alpha) \nabla_{\mathbf{Z}_+} \max\{\mu, \|\mathbf{Z} - \mathbf{Z}_+\|_1 - \|\mathbf{Z} - \mathbf{Z}_-\|_1\} \end{aligned} \quad (12)$$

replacing $\nabla_{\mathbf{Z}} \ell$ in (10), and the active set Λ replaced by the corresponding actives sets of $\mathbf{Z}_\pm^*$.

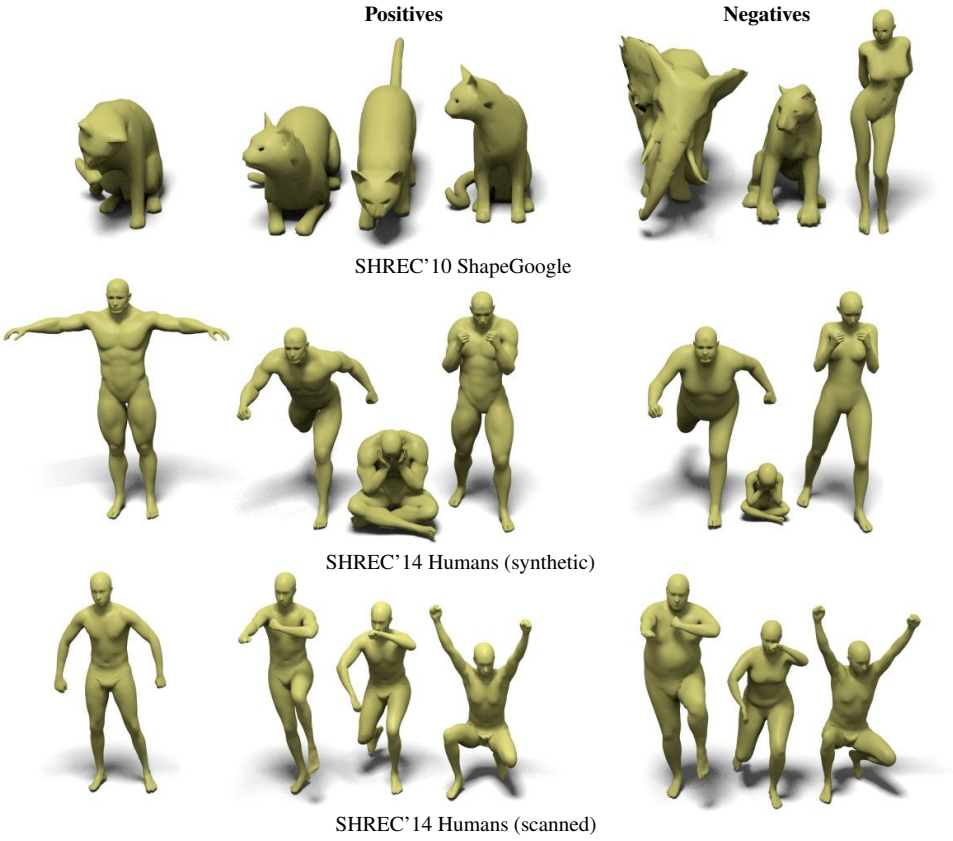

Figure 5: *Examples of shapes from three datasets used in our experiments, from the easiest to the hardest (top to bottom): SHREC'11 ShapeGoogle dataset contains shapes of different bi- and quadrupeds, SHREC'14 Humans (synthetic) dataset contains CAD shapes of humans differing in size and body shape, and SHREC'14 Humans (scanned) contains scanned human shapes. In the latter, the differences between different humans are very subtle (note, for example, that the leftmost and the rightmost shapes in the last row belong to different persons, which is very hard to tell). Shown are a representative shape (leftmost), three positives (center) and three negatives (right) used to construct the training set.*

The gradient of ℓ with respect to $\mathbf{D}$ can be expressed as

$$\begin{aligned}\nabla_{\mathbf{D}}\ell \;=\; & (\mathbf{X}-\mathbf{D}\mathbf{Z}^*)\beta^\top + (\mathbf{X}-\mathbf{D}\mathbf{Z}_+^*)\beta_+^\top + (\mathbf{X}-\mathbf{D}\mathbf{Z}_-^*)\beta_-^\top \\ & -\mathbf{D}\left(\beta\mathbf{Z}^{*\top} + \beta_+\mathbf{Z}_+^{*\top} + \beta_-\mathbf{Z}_-^{*\top}\right) \end{aligned} \tag{13}$$

(for derivation details, the reader is referred to [MBP12]).

We perform the minimization of the bi-level problem (9) using stochastic gradient descent as done in [MBP12], which at every iteration approximates the gradient of the loss

$$\nabla_{\mathbf{D}}\mathcal{L} = \sum_{\mathbf{X},\mathbf{X}_+,\mathbf{X}_-\in\mathcal{T}} \nabla_{\mathbf{D}}\ell(\mathbf{X},\mathbf{X}_+,\mathbf{X}_-) \tag{14}$$

by randomly drawing a batch of a few triplets (in the extreme case, only a single one) from the training set $\mathcal{T}$. As the initial $\mathbf{D}$, we used the solution of the unsupervised dictionary learning problem (7).

5. Results

In this section, we evaluate the proposed sparse coding with supervised dictionary learning method on several standard shape retrieval benchmarks. Our code was implemented in MATLAB and is available from our SVN server [‡]. Sparse coding and unsupervised dictionary learning was done using the SPAMS toolbox [MBPS09]. The dictionary size v and the value of λ were found empirically using standard cross-validation techniques.

Retrieval performance was evaluated using *precision* (the fraction of retrieved shapes that match the query class) and *recall* (the fraction of shapes from the query class that is retrieved). In addition, we used the *mean average precision* (mAP) as a performance criterion. Evaluation was performed on datasets from the Shape Retrieval Contest (SHREC).

‡ https://vista.eng.tau.ac.il:8443/svn/main/pub/SupervisedBoF, username "guest", blank password

SHREC'10 ShapeGoogle [BBGO11] dataset consisted of 1184 synthetic shapes, out of which 715 shapes were obtained from 13 shape classes with simulated transformation (55 per shape) used as queries, and 456 unrelated distractor shapes, treated as negatives (see examples in Figure 5, top). We used HKS [SOG09] as the local descriptor of dimension $q = 31$, with the same parameters as in [BBGO11]. In order to make the dataset more challenging, we re-scaled all the shapes to have the same size, and removed the 'don't-care' ground truth labels used in the original benchmark (e.g., male and female shapes were considered the same class). For training, we took two shapes from each of the 13 shape classes (total 26 shapes), using pairs of shapes from the same class as positives and pairs from different classes as negatives (total of one positive and 25 negatives for each query). The values of $\mu = 0.5$, $\lambda = 0.5$, and $v = 48$ were used. Typical training time using stochastic gradient descent was approximately 30 sec for a batch of 25 triplets, and took less than 500 iterations to converge, resulting in nearly 4 hours in total on a machine with a 3.2GHz CPU.

Testing was performed on the rest of the shapes, disjoint with the training set (total of 53 positives and 1105 negatives for each query). We compared the performance of different methods for creating BoF descriptors highlighted in Sections 2 and 3: the original ShapeGoogle method [BBGO11] based on VQ, sparse coding with unsupervised DL (7) and the proposed supervised DL (9).

Evaluation results are summarized in Figure 7 (left). One can observe slight performance improvement resulting from replacing VQ with sparse coding with unsupervised dictionary (compare black and gray curves), and a significant improvement from learning the dictionary in a supervised manner by the proposed bi-level optimization (red curve). Figure 6 and Table 1 show the breakdown of the retrieval results by transformation classes present in the ShapeGoogle dataset. The proposed method is able to learn invariance to all the transformations from one example and outperforms the simple-minded VQ and unsupervised DL.

Transformation	VQ	Unsup. DL	**Sup. DL**
Isometry	98.8	97.7	**99.4**
Topology	100	100	100
Isometry+topology	93.3	93.4	**95.6**
Partiality	94.7	94.8	**95.1**
Triangulation	95.4	95.0	**95.5**
All	89.1	89.1	**91.2**

Table 1: *Comparison of different retrieval methods in terms of mean average precision (mAP, in %) on the SHREC'10 ShapeGoogle dataset, broken down according to transformation classes.*

SHREC'14 Humans [P*14] consisted of two different subsets. The first part (synthetic) contained 15 different human models created using DAZ Studio, each in 20 different poses

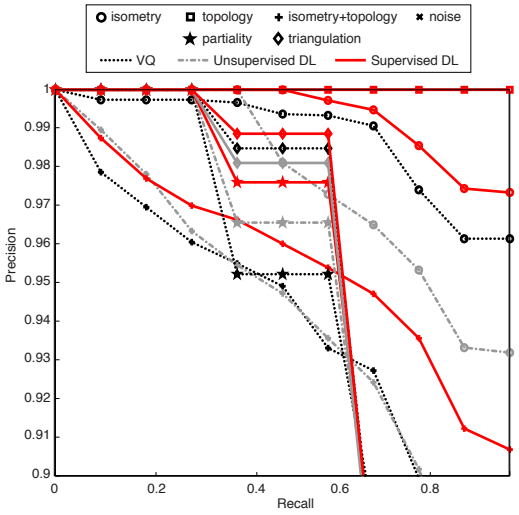

Figure 6: *Performance (in terms of Precision/Recall) of different retrieval methods on the SHREC'10 ShapeGoogle dataset, broken down by transformation class.*

(total of 300 models, Figure 5, middle). The second one (scanned) contained scans of 40 human subjects, each in 10 different poses (400 shapes in total, Figure 5, bottom). All shapes were down-sampled to have 4.5×10^3 triangles.

Note that both datasets (in particular, the scanned shapes) are extremely challenging, as they contain geometrically similar human shapes (very difficult to distinguish even for a human observer). Due to the big variability in the shape sizes, we used the 16-dimensional SI-HKS as the local descriptor, with settings according to [BK10]. For training, we used four examples per class for both datasets. For the synthetic dataset, the number of positives was 3 and the number of negatives was 56 per query. For the scanned dataset, we used 3 positives and 36 negatives per query. We used $\mu = 0.2, v = 32$ for both datasets, $\lambda = 0.5$ for the synthetic dataset, and $\lambda = 0.25$ for the scanned one.

Testing was performed on the rest of the shapes, disjoint with the training set. For each query in the synthetic dataset, the number of positives was 15 and the number of negatives was 224. For each query in the scanned dataset, the number of positives was 5 and the number of negatives was 234. In addition to ShapeGoogle (VQ), unsupervised DL and the proposed supervised DL, we compared to the recent state-of-the-art shape retrieval methods that achieved top performance on the SHREC benchmark, based on Histograms of Area Projection Transform (HAPT) [GL12], Deep Belief Network (DBN) [P*14], Intrinsic Spatial Pyramid Matching (ISPM) [LH13b, LH13a], and Reduced Biharmonic Distance Matrix (R-BiHDM) [YYY13]. Evaluation results are summarized in Figure 7 (center and right) and in Table 2. The proposed approach consistently outperforms all the compared methods.

Figure 8 contains an example of top five matches returned

by different methods in response to a female shape query. The difficulty of the 'fine-grained' human shape retrieval task is evident from this example (all the mismatched shapes appear 'reasonable'), and the fact that our method produces all correct matches is remarkable.

Method	Synthetic	Scanned
ISPM [LH13b, LH13a]	90.2	25.8
DBN [P*14]	84.2	30.4
R-BiHDM [YYY13]	64.2	64.0
HAPT [GL12]	81.7	63.7
ShapeGoogle (VQ) [BBGO11]	81.3	51.4
Unsupervised DL	84.2	52.3
Supervised DL	**95.4**	**79.1**

Table 2: *Comparison of different retrieval methods in terms of mean average precision (mAP, in %) on the SHREC'14 Humans datasets.*

6. Conclusion

In this work, we presented a method for learning bag-of-features shape descriptors in a fully supervised manner. Unlike previous approaches that tried to introduce supervision in some parts of the BoF pipeline (e.g. making the VQ process supervised), our training is done 'end-to-end', optimizing a task-specific penalty dependent on the final BoF. Working in such a supervised regime allows to learn invariance to practically any kind of transformations or degree of variability, provided that representative examples of positive and negative shapes are available. Ideologically, our approach follows [BBGO11], which advocated in favor of learning invariance from examples rather than trying to construct invariant descriptors axiomatically.

Experimental results on the recent challenging SHREC benchmarks show that the proposed method achieves state-of-the-art performance, and especially excels in cases where there are subtle differences between the shape classes. Such 'fine-grain' recognition problems are currently considered the most difficult in the pattern recognition community [GFS*13, HSG13].

Our method beats some state-of-the-art algorithms, doing so 'out of the box' based on older descriptors that are no longer considered such. Some of the newer shape descriptors can trivially be used within our framework.

Future directions. We see several follow-up directions for our work. First, in our approach we used a very naïve mean pooling operator. It is possible to use a different pooling strategy such as tf-idf, or more generally, to learn the pooling operator. Second, if using spectral local descriptors, we can incorporate the approach of [LB14] for learning the optimal transfer function into our pipeline. Third, the L_1-norm used in our loss function to compare between BoFs can be replaced by any differentiable dissimilarity between histograms, that does not necessarily have to be a metric. Finally, we can binarize the BoFs produced by our approach,

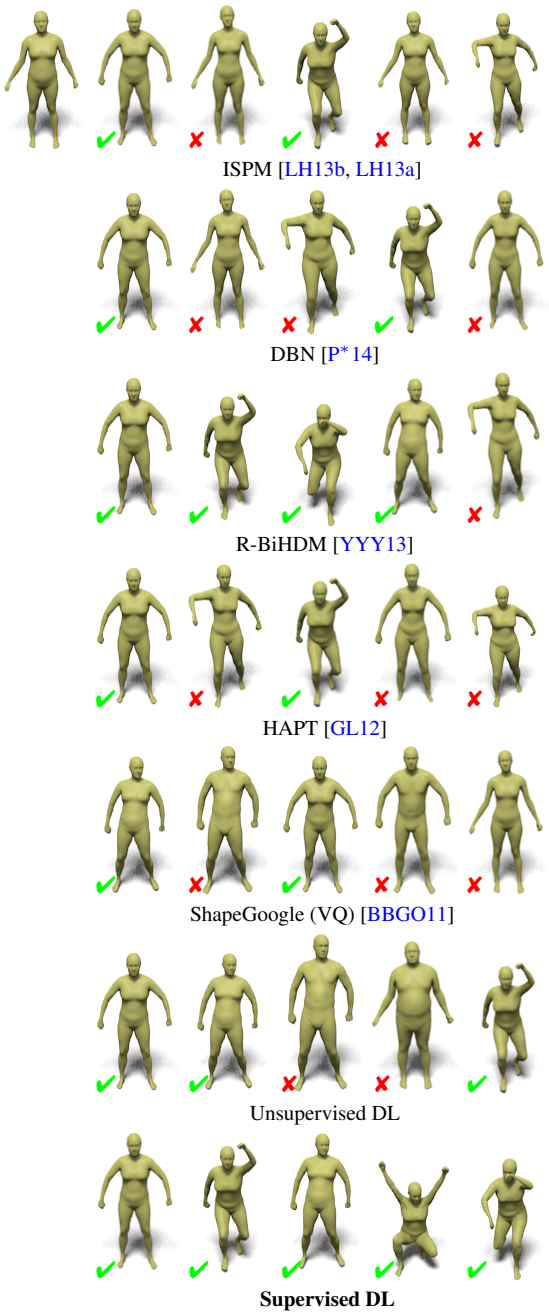

Figure 8: *Top five matches to a female query (left) produced by different retrieval methods on the SHREC'14 Humans (scanned) dataset. Correct/wrong matches are marked in green/red.*

thus hashing the descriptors similarly to Bronstein et al. [BBGO11]. The big advantage of binary descriptors is their compactness (which is of importance in large-scale applications) and the efficient computation of the Hamming metric used for their retrieval. As opposed to applying standard similarity-sensitive hashing techniques to BoF descrip-

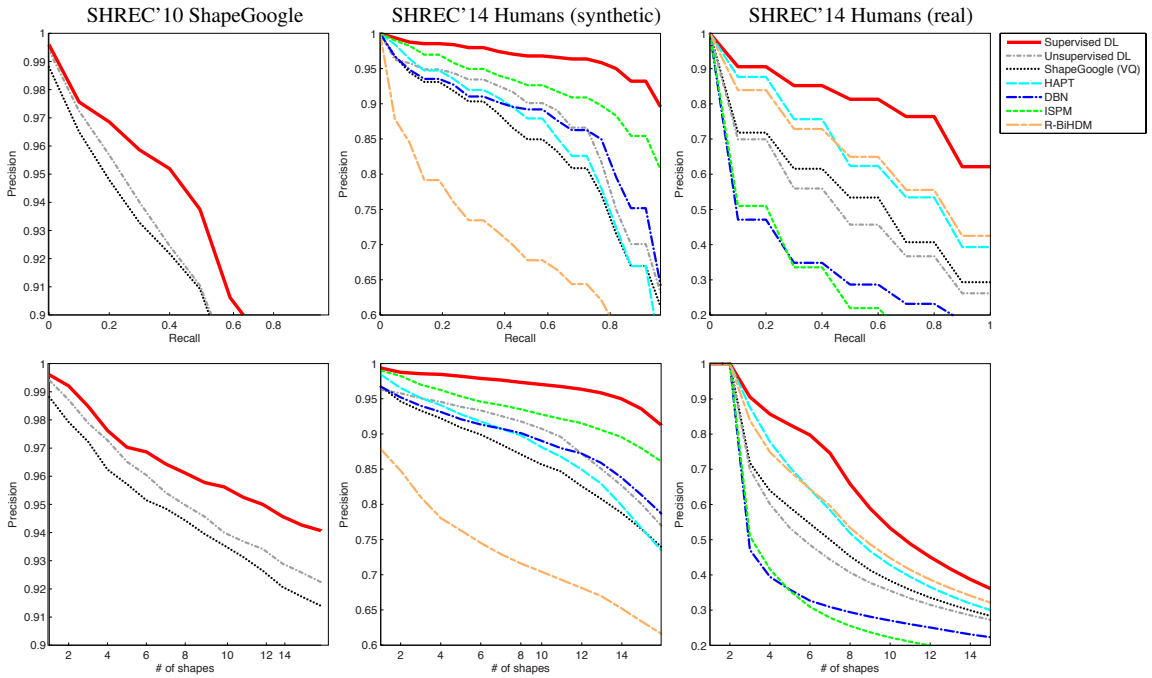

Figure 7: *Performance of different retrieval methods on the ShapeGoogle (left) and SHREC'14 Humans (synthetic, center and real, right) datasets. Show are Precision-Recall (top) and Precision@N (bottom) curves. The proposed Supervised DL method achieves the best performance in all the experiments.*

tors, the use of sparse codes allows achieving efficient retrieval without compromising the recall, as recently shown in [MBB*13].

Limitations. With our current implementation, training times can be prohibitively long in some situations. Fortunately, there have been several recent approaches to this problem, one of which is approximating the sparse coding optimization problem by a special neural network [GL10, SBS12]. This way, an iterative optimization procedure producing the sparse code is replaced by a few layers of a neural network, each of which corresponds to an iteration of the iterative shrinkage (ISTA) algorithm [DDDM04]. The resulting speedup can be in the range of several orders of magnitude [SBS12].

References

[AEB06] AHARON M., ELAD M., BRUCKSTEIN A.: *k*-SVD: an algorithm for designing overcomplete dictionaries for sparse representation. *Trans. Signal Processing 54*, 11 (2006), 4311–4322. 2, 4

[ASYS10] AKGÜL C. B., SANKUR B., YEMEZ Y., SCHMITT F.: Similarity learning for 3d object retrieval using relevance feedback and risk minimization. *International Journal of Computer Vision 89*, 2-3 (2010), 392–407. 1

[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. 1, 2, 7, 8

[BC13] BOSCAINI D., CASTELLANI U.: Local signature quantization by sparse coding. In *Proc. 3DOR* (2013). 2

[BK10] BRONSTEIN M. M., KOKKINOS I.: Scale-invariant heat kernel signatures for non-rigid shape recognition. In *Proc. CVPR* (2010), IEEE, pp. 1704–1711. 1, 3, 7

[BNJ03] BLEI D. M., NG A. Y., JORDAN M. I.: Latent dirichlet allocation. *J. Mach. Learn. Res. 3* (Mar. 2003), 993–1022. 2

[CDF*04] CSURKA G., DANCE C., FAN L., WILLAMOWSKI J., BRAY C.: Visual categorization with bags of keypoints. In *ECCV Workshops* (2004). 1

[CMS07] COLSON B., MARCOTTE P., SAVARD G.: An overview of bilevel optimization. *Ann. Oper. Res. 153*, 1 (2007), 235–256. 5

[DDDM04] DAUBECHIES I., DEFRISE M., DE MOL C.: An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Comm. Pure and Applied Mathematics 57*, 11 (2004), 1413–1457. 9

[DHS01] DUDA R., HART P., STORK D.: *Pattern Classification*, second ed. John Wiley & Sons, 2001. 2

[DK12] DAROM T., KELLER Y.: Scale-Invariant Features for 3-D Mesh Models. *Trans. Image Processing 21*, 5 (2012), 2758–2769. 1

[EAHH99] ENGAN K., AASE S. O., HAKON HUSOY J.: Method of optimal directions for frame design. In *Proc. ICASSP* (1999), vol. 5, pp. 2443–2446. 4

[FKMS05] FUNKHOUSER T., KAZHDAN M., MIN P., SHILANE P.: Shape-based retrieval and analysis of 3D models. *Comm. ACM 48* (2005), 58–64. 1

[GD05] GRAUMAN K., DARRELL T.: The pyramid match kernel:

Discriminative classification with sets of image features. In *Proc. ICCV* (2005). 1

[GFS*13] GAVVES E., FERNANDO B., SNOEK C., SMEULDERS A., TUYTELAARS T.: Fine-grained categorization by alignments. In *Proc. ICCV* (2013). 8

[GL10] GREGOR K., LECUN Y.: Learning fast approximations of sparse coding. In *Proc. ICML* (2010). 9

[GL12] GIACHETTI A., LOVATO C.: Radial symmetry detection and shape characterization with the multiscale area projection transform. *Computer Graphics Forum 31*, 5 (2012), 1669–1678. 7, 8

[GLWT13] GONG B., LIU J., WANG X., TANG X.: Learning semantic signatures for 3d object retrieval. *Multimedia, IEEE Transactions on 15*, 2 (2013), 369–377. 1

[HFL12] HU R., FAN L., LIU L.: Co-segmentation of 3d shapes via subspace clustering. *Computer Graphics Forum 31*, 5 (2012), 1703–1713. 2

[Hof01] HOFMANN T.: Unsupervised learning by probabilistic latent semantic analysis. *Mach. Learn. 42*, 1-2 (2001), 177–196. 2

[HSG13] HUANG Q.-X., SU H., GUIBAS L.: Fine-grained semi-supervised labeling of large shape collections. *ACM Transactions on Graphics (TOG) 32*, 6 (2013), 190. 1, 8

[JP12] JOJIC N., PERINA A.: Multidimensional counting grids: Inferring word order from disordered bags of words. *arXiv preprint arXiv:1202.3752* (2012). 2

[Lav12] LAVOUE G.: Combination of bag-of-words descriptors for robust partial shape retrieval. *The Visual Computer 26* (2012), 1257–1268. 1

[LB14] LITMAN R., BRONSTEIN A.: Learning spectral descriptors for deformable shape correspondence. *Trans. PAMI* (2014). 8

[L*et al.*13] LIAN Z., *et al.*: A Comparison of Methods for Non-rigid 3D Shape Retrieval. *Pattern Recognition 46*, 1 (2013), 449–461. 1

[LGSX13] LIAN Z., GODIL A., SUN X., XIAO J.: CM-BOF: visual similarity-based 3D shape retrieval using clock matching and bag-of-features. *Machine Vision and Applications 24*, 8 (2013), 1685–1704. 1

[LH13a] LI C., HAMZA A.: Intrinsic spatial pyramid matching for deformable 3d shape retrieval. *J. Multimedia Information Retrieval 2*, 4 (2013), 261–271. 1, 7, 8

[LH13b] LI C., HAMZA A. B.: A multiresolution descriptor for deformable 3d shape retrieval. *The Visual Computer 29*, 6-8 (2013), 513–524. 7, 8

[LN08] LAGA H., NAKAJIMA M.: Supervised learning of similarity measures for content-based 3d model retrieval. In *Proc. Large-Scale Knowledge Resources*. 2008, pp. 210–225. 1

[LR09] LAZEBNIK S., RAGINSKY M.: Supervised learning of quantizer codebooks by information loss minimization. *Trans. PAMI 31*, 7 (July 2009), 1294–1309. 2

[LSGFRC*13] LÓPEZ-SASTRE R. J., GARCÍA-FUERTES A., REDONDO-CABRERA C., ACEVEDO-RODRÍGUEZ F. J., MALDONADO-BASCÓN S.: Evaluating 3d spatial pyramids for classifying 3d shapes. *Computers and Graphics 37*, 5 (2013), 473–483. 1

[LSP06] LAZEBNIK S., SCHMID C., PONCE J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Proc. CVPR* (2006), pp. 2169–2178. 1

[LWW*10] LIU Y., WANG X.-L., WANG H.-Y., ZHA H., QIN H.: Learning robust similarity measures for 3d partial shape retrieval. *International Journal of Computer Vision 89*, 2-3 (2010), 408–431. 1

[LZQ06] LIU Y., ZHA H., QIN H.: Shape topics: A compact representation and new algorithms for 3d partial shape retrieval. In *Proc. CVPR* (2006). 2

[MBB*13] MASCI J., BRONSTEIN A. M., BRONSTEIN M. M., SPRECHMANN P., SAPIRO G.: Sparse similarity-preserving hashing. *arXiv:1312.5479* (2013). 9

[MBP*08] MAIRAL J., BACH F., PONCE J., SAPIRO G., ZISSERMAN A.: Discriminative learned dictionaries for local image analysis. In *Proc. CVPR* (June 2008), pp. 1–8. 2

[MBP12] MAIRAL J., BACH F., PONCE J.: Task-driven dictionary learning. *Trans. PAMI 34*, 4 (2012), 791–804. 2, 5, 6

[MBPS09] MAIRAL J., BACH F., PONCE J., SAPIRO G.: Online dictionary learning for sparse coding. In *Proc. ICML* (2009). 6

[MWZ*13] MITRA N., WAND M., ZHANG H. R., COHEN-OR D., KIM V., HUANG Q.-X.: Structure-aware shape processing. In *Proc. SIGGRAPH Asia* (2013), ACM, p. 1. 1

[OLGM11] OVSJANIKOV M., LI W., GUIBAS L., MITRA N. J.: Exploration of continuous variability in collections of 3D shapes. In *TOG* (2011), vol. 30, p. 33. 1

[P*14] PICKUP D., ET AL.: SHREC'14 track: Shape retrieval of non-rigid 3D human models. In *Proc. 3DOR* (2014). 4, 7, 8

[PP93] PINKALL U., POLTHIER K.: Computing discrete minimal surfaces and their conjugates. *Experimental mathematics 2*, 1 (1993), 15–36. 2

[SBS12] SPRECHMANN P., BRONSTEIN A. M., SAPIRO G.: Learning efficient sparse and low rank models. *arXiv:1212.3631* (2012). 9

[SH06] SKRETTING K., HUSØY J. H.: Texture classification using sparse frame based representation. *J. Applied Signal Processsing* (2006), 102–102. 2

[SOG09] SUN J., OVSJANIKOV M., GUIBAS L.: A Concise and Provably Informative Multi-Scale Signature Based on Heat Diffusion. vol. 28, pp. 1383–1392. 1, 2, 7

[SZ03] SIVIC J., ZISSERMAN A.: Video google: A text retrieval approach to object matching in videos. In *Proc. ICCV* (2003), pp. 1470–1477. 1, 3

[TCF10] TOLDO R., CASTELLANI U., FUSIELLO A.: The bag of words approach for retrieval and categorization of 3D objects. *The Visual Computer 26* (2010), 1257–1268. 1

[Vap98] VAPNIK V.: *Statistical Learning Theory*. Wiley, New York, 1998. 2

[WS09] WEINBERGER K. Q., SAUL L. K.: Distance metric learning for large margin nearest neighbor classification. *JMLR 10* (2009), 207–244. 4

[WYNH13] WANG Z., YANG J., NASRABADI N., HUANG T.: A max-margin perspective on sparse representation-based classification. In *Proc. ICCV* (2013), pp. 1217–1224. 2

[YYY13] YE J., YAN Z., YU Y.: Fast nonrigid 3d retrieval using modal space transform. In *Proc. Conf. Multimedia Retrieval* (2013), pp. 121–126. 7, 8