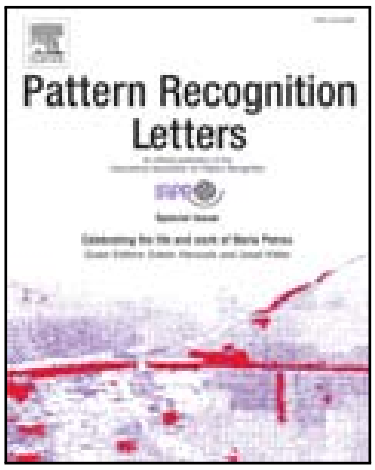


Multiple Vocabulary Coding for 3D Shape Retrieval Using Bag of Covariances

Hedi Tabia, Hamid Laga

PII: S0167-8655(17)30213-1
DOI: 10.1016/j.patrec.2017.06.007
Reference: PATREC 6846

To appear in: *Pattern Recognition Letters*

Received date: 24 November 2016
Revised date: 11 April 2017
Accepted date: 10 June 2017

Please cite this article as: Hedi Tabia, Hamid Laga, Multiple Vocabulary Coding for 3D Shape Retrieval Using Bag of Covariances, *Pattern Recognition Letters* (2017), doi: 10.1016/j.patrec.2017.06.007

Highlights

- A new compact shape signature that is built from multiple vocabulary.
- A mechanism for reducing both the impact of vocabulary correlation as well as the size of the signature.
- Experiments on standard shape retrieval benchmarks show that the proposed approach outperforms the state-of-the-art.

Pattern Recognition Letters
journal homepage: www.elsevier.com

Multiple Vocabulary Coding for 3D Shape Retrieval Using Bag of Covariances

Hedi Tabia[a,**], Hamid Laga[b]

[a] *ETIS/ENSEA, University of Cergy-Pontoise, CNRS, UMR 8051, France*
[b] *School of Engineering and IT, Murdoch University, Australia*

ABSTRACT

Bag of Covariance matrices (BoC) have been recently introduced as an extension of the standard Bag of Words (BoW) to the space of positive semi-definite matrices, which has a Riemannian structure. BoC descriptors can be constructed with various Riemannian metrics and using various quantization approaches. Each construction results in some quantization errors, which are often reduced by increasing the vocabulary size. This, however, results in a signature that is not compact, increasing both the storage and computation complexity. This article demonstrates that a compact signature, with minimum distortion, can be constructed by using multiple vocabulary based coding. Each vocabulary is constructed from a different quantization method of the covariance feature space. The proposed method also extracts non-linear dependencies between the different BoC signatures to compose the final compact signature. Our experiments show that the proposed approach can boost the performance of the BoC descriptors in various 3D shape classification and retrieval tasks.

1. Introduction

Recent advances in 3D acquisition technology, the widespread of online 3D repositories, such as Trimble 3D Warehouse, and the impact of Web3D on various domains, have been important drivers to the growing need for efficient 3D shape classification and retrieval techniques. Among existing techniques, Bag of Words (BoW), motivated by their success in image retrieval and classification, are widely used in 3D shape classification and retrieval tasks. Their advantage is that they enable the aggregation of local descriptors, of the same type, computed at different locations on a given shape, into a single global descriptor, which then inherits the invariance and robustness properties of the local descriptors.

Standard BoW approaches assume that features are elements of Euclidean spaces and thus one can use the $\mathbb{L}^2$ metric for building the dictionary and for computing the BoW signatures. In the past years, however, several authors showed that the use of features that are elements of non-linear Riemannian manifolds can lead to substantial improvement in classification and retrieval performances. Examples of such features include covariance-based descriptors, originally introduced for image analysis (Porikli et al., 2006), and recently extended to 3D shapes (Tabia et al., 2014; Tabia and Laga, 2015). Tabia et al. (2014); Tabia and Laga (2015) extended the standard BoW paradigm that operates on Euclidean spaces to the space of positive semi-definite matrices, which has a Riemannian structure. They studied four different quantization methods and concluded that k-means using the geodesic distance and linear update of centers produces less distortion and thus is a suitable way for vocabulary construction. However, this distortion generally depends on the codebook size. The larger the codebook is, the smaller the distortion will be. Nevertheless, when using a very large codebook, signatures become non compact and thus the retrieval system loses the benefits of the BoC representation.

In this paper, we demonstrate that the distortion caused by the quantization, when using BoC representation, can be reduced by fusing multiple BoC signatures each one is computed with a different quantization method and using a different metric. Indeed, codebooks are computed while taking into account the different geometries of the space of covariance matrices (e.g. Riemannian and Euclidean). By computing multiple BoC signatures with different codebooks, more candidate features are

**Corresponding author: Tel.: +0-000-000-0000; fax: +0-000-000-0000;
e-mail: hedi.tabia@ensea.fr (Hedi Tabia)

recalled, which enables to gain rich and effective 3D shape representation while reducing quantization distortions and errors.

Contributions. The contributions of this article are; (1) A new compact shape signature that is built from multiple BoC vocabularies, each one constructed with a different metric and a different quantization method. By using multiple BoC vocabularies, more features are recalled, which corrects quantization artifacts. (2) A mechanism for reducing both the impact of vocabulary correlation as well as the size of the signature. (3) We show that using multiple vocabularies one improves the 3D retrieval and classification performance over existing individual BoC signatures. (4) Finally, our experiments on standard 3D shape retrieval benchmarks show that the proposed approach outperforms the state-of-the-art.

Figure 1 overviews the proposed approach. Below, we detail each step of the pipeline.

2. Bag of Covariance matrices (BoC)

We use covariance-based descriptors (Tabia et al., 2014) as local features that encode the local geometry of a 3D shape. Covariance descriptors are computed on local patches $\{P_i, i = 1 \dots m\}$, which may be overlapping. Each patch P_i is extracted around a representative point $p_i = (x_i, y_i, z_i)^t$. We denote by X_i the Symmetric Positive Definite (SPD) covariance matrix which corresponds to our descriptor. In our implementation, X_i encodes the covariance of a $(dim = 5)-$dimensional feature vector f_j of all points $p_j = (x_j, y_j, z_j)^t, j = 1 \dots, n_i$, belonging to the patch P_i. The dimensions correspond to (1) the location of the point p_j with respect to the patch center $p_c = \frac{1}{n_i}\sum_{k=1}^{n_i} p_k$. It is given by $p_j - p_c$, (2) the distance of the point p_j to p_i, and (3) the volume of the parallelepiped formed by the coordinates of the point p_j. We then compute the covariance matrix X_i of these features. The diagonal elements of X_i represent the variance of each feature while its off-diagonal elements represent their respective co-variations. It has a fixed dimension (5×5) independently of the size of the patch P_i.

The space of covariance matrices $\mathcal{M} = Sym_d^+$ is a special type of homogeneous space which carries a natural Riemannian structure. The geodesic distance, $d_g^2(X, Y)$, between two points X and Y on $\mathcal{M}$ is given by:

$$\left\langle \log_X(Y), \log_X(Y) \right\rangle_X = \text{trace}\left(\log^2\left(X^{-\frac{1}{2}} Y X^{-\frac{1}{2}}\right)\right) \quad (1)$$

Since the space of covariance matrices $\mathcal{M}$ carries a Riemannian structure, Tabia and Laga (2015) proposed the extension of the vocabulary construction paradigm, originally designed for Euclidean spaces, to non-linear Riemannian manifolds of covariance matrices. They proposed to compute the vocabulary while taking into account the geometry of the manifold. Four construction methods have been proposed:

K-medoid with geodesic distance (*KM*). Instead of the K-means, the KM constrains the center of clusters to be one of the data points. Using KM requires the computation of the pairwise geodesic distances between all training data, which can be time consuming. The final set of medoids is used as vocabulary.

K-means on the tangent space (*KTS*). The KTS first maps all the training points to the tangent space of the manifold at one point (e.g. the mean point), resulting in an Euclidean representation of the manifold-valued data. K-means algorithm is then applied to construct a set of clusters with minimum average distortion in the tangent space. Finally the cluster centers are projected back to the SDP space to form the vocabulary.

K-means with geodesic distance and linear update of centers (*KGL*). The KGL uses the geodesic distance with the Frobenius distance when computing the centers of k-means. The idea is that the Euclidean average of covariance matrices lies in the Riemannian manifold. Indeed, any non-negative linear combination of SPD matrices is an SPD matrix. This implies that the linear average $\hat{X}$ of $X_1, ..., X_N$ given by: $\hat{X} = \frac{1}{N}(X_1 + ... + X_N)$ is an SPD matrix. The Frobenius distance from which the linear average came is given by:

$$d_F^2(X_1, X_2) = \sum_{1 \le i,j \le dim} \left|(X_1 - X_2)_{ij}\right|^2. \quad (2)$$

When assigning a data point X_i to its closest center $\hat{X}$, KGL uses the geodesic distance.

K-means with Frobenius distance (*KF*). The KF algorithm considers the Euclidean ambient space of SDP matrices and ignores its Riemannian structure. It uses Frobenius distance for both K-means steps; assignment, and center computing.

It has been shown in Tabia and Laga (2015) that different effect on the clustering can be observed based on the choice of distance. This demonstrates that different quantizers have different effects on the data. We aim in this article to exploit these differences to build an efficient shape descriptor.

3. Multiple vocabulary fusion

In the training step of the BoC approach, we use the four algorithms presented in Section 2 to learn four codebooks from all the features. Depending on the distance measure used, Each algorithm attempts to minimize the following distortion:

$$\min_{\hat{X}_k} \frac{1}{|\mathcal{T}|} \sum_{k=1}^{K} \sum_{X \in C_k} d^2(X, \hat{X}_k), \quad (3)$$

where K is the number of codewords in each codebook, $|\mathcal{T}|$ represents the cardinality of the input training data set, and $\hat{X}_k$ is the center of the cluster to which X is assigned to (i.e. $C_k = \left\{X | d(X, \hat{X}_k) < d(X, \hat{X}_{k'}),\ \forall k' \neq k\right\}$). Here, $d(\cdot, \cdot)$ is the distance measure (or metric) used for clustering. A good set of visual words is the result of a clustering that has minimum distortion with respect to the training features. For L different codebooks, with K words per each codebook, then the minimization of the total distortion from all codebooks can be written as :

$$\min_{\hat{X}_k^l} \frac{1}{L|\mathcal{T}|} \sum_{l=1}^{L} \left(\sum_{k=1}^{K} \sum_{X \in C_k^l} d_l^2(X, \hat{X}_k) \right). \quad (4)$$

Here $C_k^l = \left\{X | d(X, \hat{X}_k^l) < d(X, \hat{X}_{k'}^l),\ \forall k' \neq k\right\}$.

It is well known in Nearset Neighbour (NN) search problems (Jegou et al., 2010; Xia et al., 2013) that minimizing the distortion from multiple quantizers such as in Eqn. (4) is more

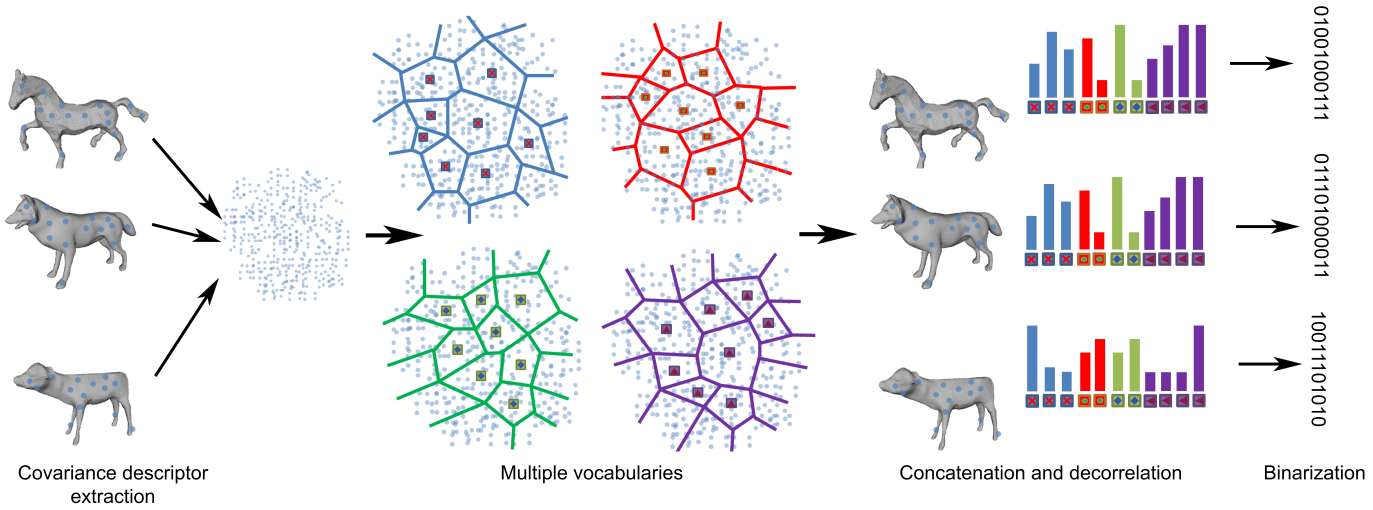


Fig. 1. Overview of the framework; Instead of using a single vocabulary for covariance descriptor coding, we proposed the use of four vocabularies constructed from different ways. We then jointly concatenate 3D signatures from each vocabulary, decorrelate them using KPCA like algorithm. Finally, we binarize the signature to speedup the process.

effective an leads to better recall and precision (Wu et al., 2009; Babenko and Lempitsky, 2012) than minimizing the distortion from a single quantizer (Eqn. (3)). Typically, multi-vocabulary merging can be performed either at score level, e.g., by concatenating the BoW histograms, or at rank level, e.g., by rank aggregation (Jegou et al., 2010). By analogy with NN search problems, we propose to use a multiple vocabulary-based coding approach for the BoC-based 3D shape retrieval. First, it is necessary to generate L different codebooks using the same set of covariance descriptors. Diversity can be obtained by using the same algorithm and varying initializations. It is very unlikely that these different initializations give the same solution for a big data, as the algorithm converges to a local minimum.

Various codebooks can be also obtained by using the different algorithms presented in section 2. This latter, which we develop in what follows, is particularly more interesting since it allows the codebook construction from different quantization methods, each one using a different metric (e.g. Riemannian or Frobenius). Each metric captures different aspects of the data space. To illustrate this idea, we consider the simple 2D data exampled shown in Figure 2. Suppose we have two quantizers each with two codewords. The first quantizer is constructed using the Euclidean distance (Figure 2 (a)) while the second one is constructed using a non-linear Riemannian distance (Figure 2 (b)). The space partitioning of the two quantizers are quite different ((a) and (b)), and thus the gain of having the second quantizer is effective. If objects, such as $\mathbb{Q} = \{q_1, q_2\}$, $\mathbb{S} = \{s_1, s_2\}$ and $\mathbb{T} = \{t_1, t_2\}$, have local descriptors located very close to the partitioning boundary of one quantizer, they will be correctly managed by the second quantizer. In Figure 2-(a), by considering hard assignment, the objects $\mathbb{Q}, \mathbb{S}$, and $\mathbb{T}$ are represented by $(2, 0)$, $(2, 0)$ and $(0, 2)$, respectively. From this representation, $\mathbb{Q}$ and $\mathbb{S}$ are identical, yet $\mathbb{S}$ and $\mathbb{T}$ are orthogonal. On the other hand, in Figure 2-(b) when using the Riemannian quantizer $\mathbb{Q} = (2, 0)$ and $\mathbb{S} = (0, 2)$ will be considered orthogonal yet $\mathbb{T}(0, 2)$ and $\mathbb{S}(0, 2)$ are considered identical. This is in contradiction with the results from the Euclidean quantizer. Now when using multiple quantizers (in Figure 2-(c)) one can notice that $\mathbb{S} = (2, 0, 0, 2)$ and $\mathbb{Q} = (2, 0, 2, 0)$ are not identical but share some local descriptors. Same for $\mathbb{S} = (2, 0, 0, 2)$ and $\mathbb{T} = (0, 2, 0, 2)$, which are not orthogonal.

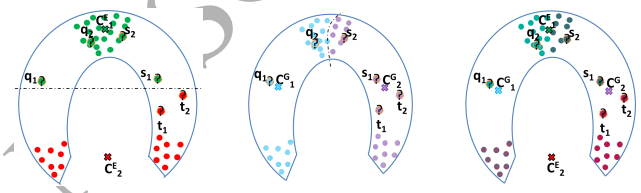


Fig. 2. We illustrate two quantizers with two codewords in each quantizer. (a) Clustering using Euclidean-based quantizer, which contains two codewords $\{C_1^E, C_2^E\}$. (b) Clustering using a non-linear Riemannian-based quantizer, which has also two codewords $\{C_1^G, C_2^G\}$. (c) Multiple quantizers from the joint vocabulary merging $\{C_1^E, C_2^E, C_1^G, C_2^G\}$. Here, $\mathbb{Q} = \{q_1, q_2\}$, $\mathbb{S} = \{s_1, s_2\}$ and $\mathbb{T} = \{t_1, t_2\}$ are three shapes.

With this motivation, we propose a new multiple vocabulary-based coding for BoC-based 3D shape retrieval: once the vocabularies have been constructed, 3D shapes are described using vectors of visual word frequencies. In our case, each visual word is a representative covariance matrix. For a given vocabulary, each point P_i on a given 3D model, represented by its covariance descriptor, is assigned to its closest center using the geodesic distance for KM, KTS and KGL algorithms and using the Frobenius distance for KF algorithm. Mathematically, we associate each point P_i with a vector w^i of size K such that:

$$w_j^i = 1 \text{ if } j = \underset{k \in [1..K]}{\arg\min}\, d_{(g,F)}(X_i - \hat{X}_k), \text{ otherwise, } w_j^i = 0. \quad (5)$$

where $d_{(g,F)}$ is either the geodesic or the Frobenius distance depending on the clustering algorithm used for quantization. From each vocabulary, a signature vector W of a 3D model is computed as the sum over i of the vectors w^i. It encodes the

number of times a given visual word appears in that model.

We then propose to serially aggregate the signature vectors produced from the different codebooks and reduce its dimensionality using KPCA while retaining its discriminative power. Formally, we starts by integrating the different BoC representations into one single vector. Let $W_l, l = 1, \ldots, L$, be the BoC vectors produced from L different codebooks. The proposed new signature is defined by the vector $V = (W_1, W_2, \ldots, W_L)^T$, the the concatenation of the L signatures. If the BoC representation W_l is of dimension d_l, then the produced signature V is of dimension $d = \sum_{l=1}^{L} d_l$.

Beside the fact that the straightforward concatenation of the BoC vectors linearly increases the storage needs, it improves only slightly the retrieval accuracy. This is due to the large correlation between the BoC representations constructed using the four quantizers. To overcome this issue, we proceed as follows. We compute the Gram matrix G where $G_{i,j} = \exp(-\gamma \left\|V_i - V_j\right\|^2), \quad i, j \in S$. Here $\{V_i\}_{i=1..N}$ is the set of concatenated signatures representing a collection of shapes $S = \{S_i\}_{i=1..N}$, and γ is empirically set to 0.1. Then, we compute a low rank approximation of G. We denote by $\mathbf{D}_t$ the diagonal matrix whose diagonal elements are the t-largest eigenvalues: $\mathbf{D}_t = \mathrm{diag}\,(\lambda_1 ... \lambda_t)$. We also denote by Λ_t the matrix of the first t eigenvectors: $\mathbf{U}_t = [\Lambda_1 ... \Lambda_t]$. We can then define $\mathbf{G}_t$ as an approximation of G: $\mathbf{G}_t = \mathbf{U}_t \mathbf{D}_t \mathbf{U}_t^\top$. Then, we compute the projectors of the signatures in approximated subspace $\mathbf{P}_t = \mathbf{V}\mathbf{U}_t\mathbf{D}_t^{-1/2}$, with $\mathbf{V} = [\mathbf{V}_1 ... \mathbf{V}_N]$ is the matrix of the signatures. For each 3D shape, we compute the projection of the signature in the sub-space as: $\mathbf{V'}_i = \mathbf{P}_t^\top \mathbf{V}_i$.. Here, $\mathbf{V'}_i$ contains an approximate and low dimensional version of $\mathbf{V}_i$. The subspace defined by the projectors preserves most of the similarity even for very small dimension. It also improves the discrimination power of the concatenated signatures since there is a large correlation between the BoC representations constructed from the earlier presented quantizers.

4. Binary Quantization

Finally, for efficient similarity search, we propose to learn similarity-preserving binary codes using the iterative quantization (ITQ) approach of Gong et al. (Gong et al., 2013). The goal is to compute hash codes such that the difference between the hash codes and the original data items, by considering each bit as the quantization value along the corresponding dimension, is minimized. The smaller the quantization loss $\left\|\mathrm{sign}(V') - V'\right\|^2$ is, the better the resulting code will preserve the original locality structure of the data.

ITQ estimates a rotation matrix R in such a way that the distortion due to binarization, i.e. the mapping from the original real-valued space to the Hamming space, decreases. We perform 40 iterations to estimate the relevant rotation. Then, the signature V' is hashed into a binary code B according to; $B = \mathrm{sign}(V'R)$, where $\mathrm{sign}(V')$ is a vector whose element i is equal to one if the element i of V' is strictly positive. It is zero otherwise. The similarity between the 3D shapes in database and a given query is efficiently computed using the binarized signature. Given the signature vector V' obtained after vocabulary fusion and dimensionality reduction of a shape query Q, V' is hashed into a binary code B. The distance between the query Q and the 3D shape S in the database is computed as the Hamming distance between their respective binary codes.

5. Experimental results

We evaluate the performance of the proposed multiple vocabulary-based 3D shape retrieval using different standard benchmarks, study and discuss its benefits, and perform a comprehensive comparison to the state-of-the-art. We conducted experiments on five different datasets; (1) The Mcgill dataset (Siddiqi et al., 2008) composed of 255 articulated 3D shapes divided into ten classes. Each class contains one 3D shape but in different poses. (2) The WM-SHREC07 dataset[2] containing 400 shapes grouped into 20 classes. (3) The PM-SHREC07 dataset[5] composed of 400 models and a query set of 30 composite models. It will be used to test the performance of the proposed method in partial 3D shape retrieval. (4) The SHREC14LSCTB dataset (Li et al., 2015), which contains 8987 polygon soup models categorized into 171 classes with an average of 53 models per class. (5) SHREC'15 (Godil et al., 2015) containing 1200 models and 180 range scans of 60 physical objects.

Given a ground-truth classification, we use different metrics such as Nearest Neighbor (NN), First Tier(FT), Second Tier (ST), and Discounted Cumulative Gain (DCG), in addition to precision-recall graphs, to evaluate and compare the classification performance of our method and other state-of-the-art shape retrieval algorithms:

5.1. Evaluation and parameter study

We have evaluated several scenarios of possible vocabulary merging, with at least two vocabularies. All the experiments have been done with a 128-dimensional signatures after KPCA based merging, unless another setup is specified. The distance between two shapes in this setting is given by the L2 distance between their signatures. We first randomly sample $m = 1000$ points on each 3D model and extract one patch P_i around each sample point p_i. Each patch has a radius $r = 15\%$ of the radius of the shape's bounding sphere. We then compute a 5×5 covariance matrix from the features defined in Section 2. We build different vocabularies using three construction methods: (1) k-means on the tangent space of the mean data point (KTS), (2) k-means using the geodesic distance and linear update of centers (KGL), and (3) k-means using Frobenius distance (KF). We do not report results about the KM since the minimization in the k-medoid problem is computationally very expensive as it uses exhaustive search.

Single versus multiple vocabularies. Table 1, which reports results on the WM-SHREC07 dataset, shows how the retrieval performance of the proposed method varies with respect to multiple vocabularies. It shows that fusing multiple vocabularies imporoves the performance of the retrieval system in all cases. Note that the individual vocabularies in this experiment have

[2] http://watertight.ge.imati.cnr.it/
[5] http://partial.ge.imati.cnr.it/

Table 1. Same size vocabulary fusion size (256). Results are on WM-SHREC07 dataset.

Vocabularies	NN	1-Tier	2-Tier	DCG
KTS	0.9000	0.5826	0.6951	0.8380
KF	0.8750	0.4799	0.6086	0.7712
KGL	0.9300	0.6237	0.7374	0.8639
KTS and KGL	0.9325	0.6351	0.7432	0.8648
KTS and KF	0.9125	0.6484	0.7566	0.8741
KGL and KF	0.9550	0.6788	0.7989	0.8945
KTS, KGL and KF	0.9575	0.6437	0.8140	0.8995
KGL and KGL	0.9425	0.67132	0.8093	0.8750
KF and KF	0.8925	0.6484	0.7523	0.8641
KTS and KTS	0.9250	0.6675	0.7512	0.8877

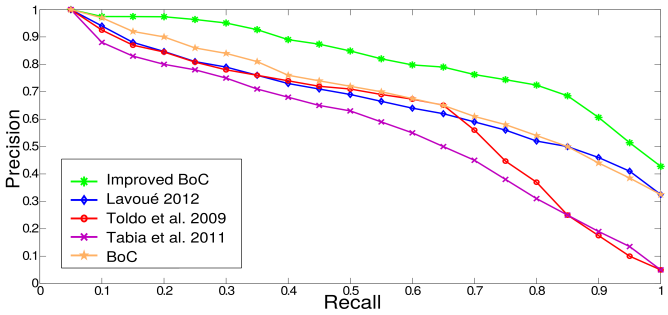


Fig. 3. Precision/recall curves of our method for the WM-SHREC07 dataset with comparison with state-of-the-art methods.

equal sizes of 256 words. From Table 1, we note that the KF-based vocabulary, which is based on the Frobenius distance, gives lower performance compared to both KTS and KGL taken separately. When using both KTS and KGL jointly, the performance is lower than using KF and KGL or KF and KTS. Using KF and KGL gives higher performance than using KGL and KGL or KF and KF. This behavior is mainly due to the complementarity between the different types of quantizers, which use different distances (e.g. Riemannian and Frobenius) in the assignment step. By using three vocabularies, the retrieval system achieves the best performance.

The influence of the vocabulary sizes. Table 2 presents the performance of the retrieval system using multiple vocabularies having different sizes. The results are reported on WM-SHREC07 dataset. From this table, one can notice that the performance slightly improves when increasing the size of both vocabularies. This behavior was predictable since distortion is reduced while increasing the vocabulary sizes. Table 2 also shows that increasing the size of the KF vocabulary has more effect on the performance after fusion. This may be due to the Euclidean propriety of KF, which does not take into account the Riemannian structure of the space of covariance matrices. Thus, the KF quantizer needs more centers than KGL to be sufficiently representative of the whole feature set.

Weighting vocabulary signatures. We also study the performance of the proposed method using various fusion weights when concatenating BoC signatures. Table 3, which reports

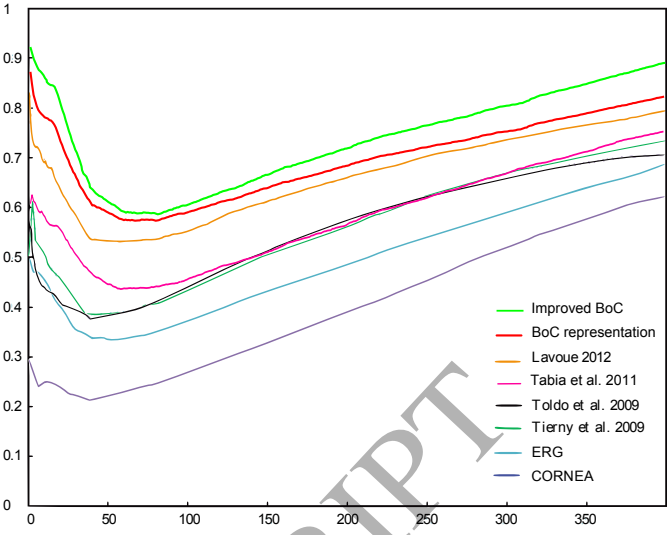


Fig. 4. NDCG curves of our method compared to recent state of the art methods, for the PM-SHREC07 dataset.

Table 2. Different size vocabulary fusion. Results are on WM-SHREC07 dataset.

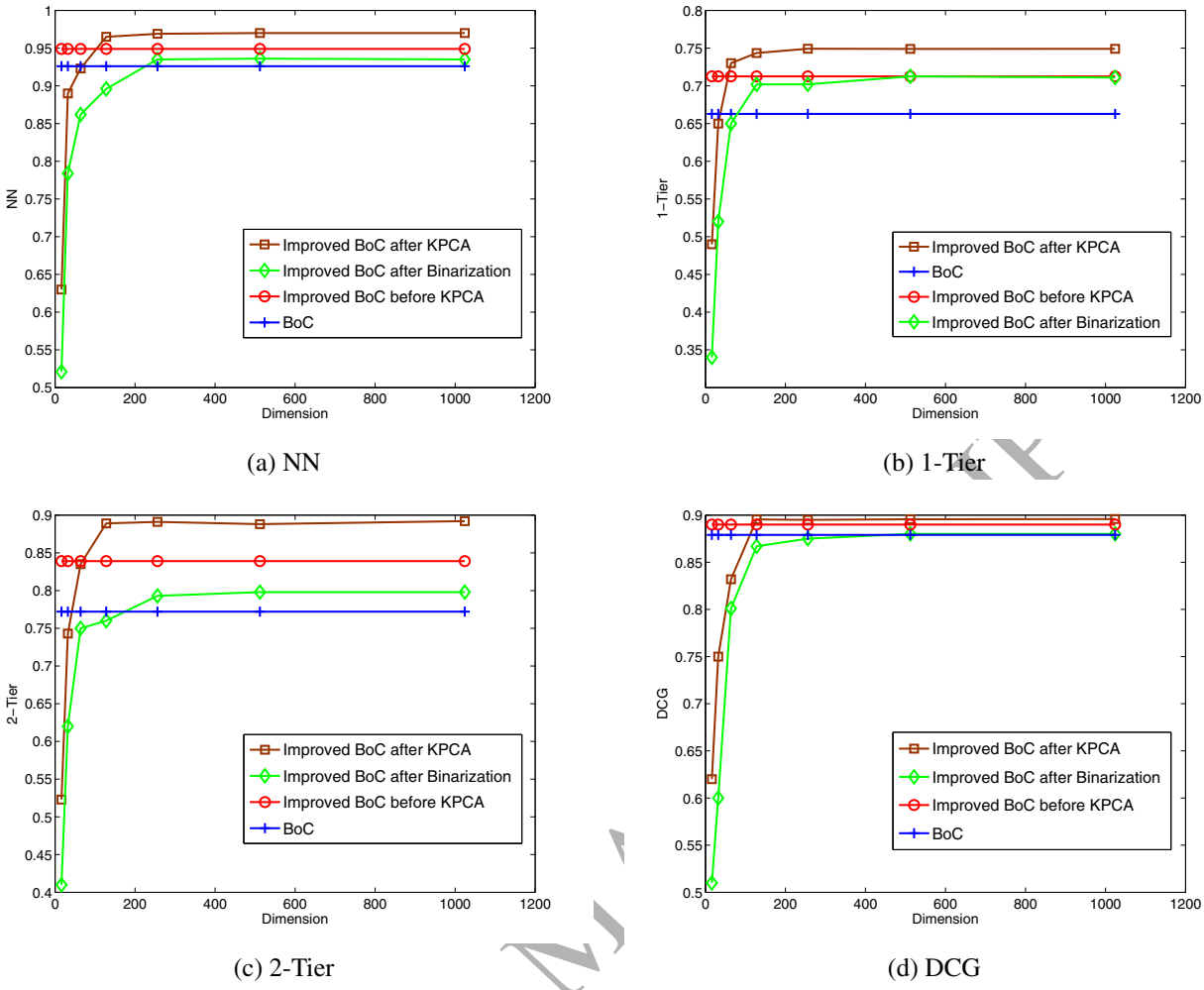
Vocabularies (size)	NN	1-Tier	2-Tier	DCG
KGL (256) and KF (256)	0.9550	0.6788	0.7989	0.8945
KGL (128) and KF (256)	0.9425	0.6410	0.7654	0.8466
KGL (256) and KF (128)	0.9375	0.6454	0.7532	0.8253
KGL (512) and KF (256)	0.9550	0.6632	0.7943	0.8934
KGL (256) and KF (512)	0.9575	0.6342	0.7384	0.8645
KGL (256) and KF (1024)	0.9625	0.6934	0.8346	0.8694
KGL (512) and KF (1024)	0.9650	0.6965	0.8354	0.8989
KGL (1024) and KF (1024)	0.9650	0.6878	0.7890	0.8541

results on WM-SHREC07 dataset, shows that adding different weights to the fusion process does not drastically increase the performance. Using hard assignment with same weights the method gives slightly better results than using inverse document frequency (IDF) weighted signatures.

The influence of signature reduction and binarization. Fig. 5 presents the influence of the KPCA and the binarization of the final shape signatures on the retrieval performance. We use four different metrics (NN, 1-Tier, 2-Tier and DCG) to evaluate the proposed method, denoted by *Improved BoC*, before KPCA, after KPCA, and after binarization. Figs. 5-(a), (b), (c) and (d) show that the improved BoC after KPCA achieves better results than the improved BoC before KPCA starting from 128-dimensional signatures. This can be explained by the fact that applying KPCA de-correlates the vocabularies and thus improves the performance. One can notice in Figure 5 that the binarization is effective for generating compact and accurate binary codes. Fig. 5-(b) shows that with only 512 bits one can reach the same performance of the non-reduced signatures in terms of 1-Tier. The improved binarized BoC (with 256-bit signature size) achieves better results than the BoC method.

5.2. Comparison with the state-of-the-art

We compare our method with the results from state-of-the-art method on the five datasets described earlier. Our approach



(a) NN (b) 1-Tier

(c) 2-Tier (d) DCG

Fig. 5. The influence of the KPCA and the binarization on the retrieval performance using different metrics. Reported results are on WM-SHREC07 dataset. Note that the performance of both "Improved BoC before KPCA" and the "BoC" method are presented without dimension reduction (i.e they are constant with respect to the dimension).

Table 3. Weighted vocabulary fusion (KGL (512) and KF (1024)). Reported results are on WM-SHREC07 dataset.

Vocabularies (Weight)	NN	1-Tier	2-Tier	DCG
KGL (1) and KF (1)	0.9650	0.6965	0.8354	0.8694
KGL (idf) and KF (idf)	0.9550	0.6543	0.7343	0.8562
KGL (1) and kF (0.5)	0.9600	0.6342	0.8330	0.8732
KGL (1) and kF (0.75)	0.9650	0.7435	0.8890	0.8954

gives competitive results with respect to the state-of-the-art methods.

On the WM-SHREC07 as shown in Table 4, the proposed approach outperforms the state-of-the-art in terms of 1-Tier, 2-Tier and DCG measures. It gives the second best performance in term of NN measure behind the Kernel on Extended Reeb Graphs (KERG) method by (Barra and Biasotti, 2013). This later achieved 100% in NN but the performance in terms of 1-Tier and 2-Tier drops to 62.44% and 82.92%. Our methods achieves 96.5% in NN, which 3.5% lower than the approach of (Barra and Biasotti, 2013). However it outperforms significantly the work of (Barra and Biasotti, 2013) in terms of 1-Tier, 2-Tier with 12% and 6% respectively. Compared to seven state-of-the-art methods, ours achieves the best performance on the McGill dataset, see Table 5, even though our method does not consider structural information of shapes. Our approach improved the BoC results on all the metrics (more than 11% in 2-Tier measure). Using the NDCG curves (Marini et al., (2007), we compared the performance of our approach in partial 3D shape retrieval against other state-of-the-art methods. The NDCG curve of Fig. 4-(c) shows that our improved BoC boosts the performance of the BoC method and clearly outperforms the state-of-the-art. This demonstrates that BoC signature fusion leads to high descriptive power. Table 6 compares our method with the methods benchmarked in (Li et al., 2015) for large-scale 3D shape retrieval. It demonstrates that the improved BoC outperforms the original BoC approach and the MFF-EW and KVLAD methods on most of the different evaluation metrics. The improved BoC is ranked three in terms of average precision measure. Note that except MFF-EW method, which combines

Table 4. Global shape retrieval performance on WM-SHREC07 dataset.

Methods	NN	1-Tier	2-Tier	DCG
Tabia et al. (Tabia et al., 2011)	0.853	0.527	0.639	0.719
Hybrid BoW (Lavoué, 2012)	0.918	0.600	0.740	0.847
KERG (Barra and Biasotti, 2013)	**1.000**	0.6244	0.8292	-
Bai et al. (Bai et al., 2014)	0.955	0.690	0.807	**0.901**
BoC(Tabia and Laga, 2015)	0.9325	0.6628	0.7728	0.8795
Improved BoC	0.9650	**0.7435**	**0.8890**	0.8954

Table 5. Performance of our method on the McGill dataset.

Method	NN	1-Tier	2-Tier	DCG
2D/3D Hybrid (Papadakis et al., 2008)	0.925	0.557	0.698	0.850
Hybrid BoW (Lavoué, 2012)	0.957	0.635	0.790	0.886
PCA-based VLAT (Tabia et al., 2013)	0.969	0.658	0.781	0.894
TLC + J-Pair (Bai et al., 2015)	**0.988**	0.795	0.921	0.956
TLC + I-Pair (Bai et al., 2015)	0.980	0.807	0.933	0.956
BoC (Tabia and Laga, 2015)	0.977	0.732	0.818	0.937
Improved BoC	**0.988**	**0.809**	**0.935**	**0.962**

geometric features, all others use features from 2D views captured around the shapes.

Since our covariance-based method is generic and allows including features from other modalities, we have specifically designed a 2D view-based BoC method and compared its performance with same state-of-the-art methods. After normalizing the input shapes for translation and scale, we extract 80 views and capture depth images of size 256×256. From the depth images, we extract a set of covariance descriptors on a dense regular grid at three different scales (16×16, 24×24, 32×32). We then obtain a large unordered set of local descriptors. Covariance descriptors are computed from the feature vector $\left[x, y, |I_x|, |I_y|, |I_x x|, |I_y y|, \sqrt{I_x^2 + I_y^2}, arctan(\frac{|I_x|}{|I_y|})\right]$, where x, y are pixel locations and I_x, I_y, $|I_{xx}|$ and $|I_{yy}|$ are intensity derivatives. The covariance matrix for an image patch of arbitrary size is an 8×8 SPD matrix. We then use our improved BoC to estimate the dissimilarity between 3D shapes.

The last two rows in Table 6 shows that the performance of the BoC and the improved BoC methods when using 2D features has been significantly improved. The improved BoC still outperforms the BoC method. These results also demonstrate that using 2D features improves competitiveness of both methods when compared to state-of-the-art. The improved BoC ranked first in terms of 1-Tier and 2-Tier measures and ranked second in terms of average precision measure. Note that the LCDR-DBSVC (Li et al., 2015), which is a boosted version of DBSVC (Li et al., 2015), uses a learning scheme such as the conventional manifold ranking methods and requires more computation time because of calculating a matrix product repeatedly. The LCDR-DBSVC signatures have 270k dimensions, which is a very high compared to the improved BoC in which signatures are of size 128. Note also that similar boosting scheme can be applied to our approach. The results, after applying a similar LCDR to our method are: NN = 0.892, 1-tier = 0.545, 2-tier = 0.719, E= 0.310, DCG = 0.853, and AV = 0.570. This clearly shows the higher performance of our method compared to the LCDR-DBSVC approach.

Table 6. Performance of our method on the SHREC14LSCTB dataset.

Method	NN	1-Tier	2-Tier	E	DCG	AP
KVLAD (Li et al., 2015)	0.605	0.413	0.546	0.214	0.746	0.396
MFF-EW (Li et al., 2015)	0.566	0.138	0.204	0.076	0.570	0.114
DBNAA_DERE (Li et al., 2015)	0.817	0.355	0.464	0.188	0.731	0.344
MR-D1SIFT (Li et al., 2015)	0.856	0.465	0.578	0.234	0.792	0.464
ZFDR (Li et al., 2015)	0.838	0.386	0.501	0.209	0.757	0.387
LCDR-DBSVC (Li et al., 2015)	**0.864**	0.528	0.661	0.255	**0.823**	**0.541**
BoC (Tabia and Laga, 2015)	0.707	0.421	0.461	0.330	0.764	0.412
Improved BoC	0.754	0.425	0.480	0.364	0.772	0.437
BoC_2DF	0.801	0.498	0.582	0.254	0.781	0.476
Improved BoC_2DF	0.852	**0.531**	**0.687**	**0.312**	0.815	0.539

Table 7. Performance of our method on the SHREC'15 dataset.

Method	NN	1-Tier	2-Tier	DCG
P-SV-DSIFT (Godil et al., 2015)	0.639	0.413	0.558	0.712
Silouettes-66views-100DCT (Godil et al., 2015)	0.478	0.190	0.275	0.541
BoF-FPFH_harris-MSD (Godil et al., 2015)	0.072	0.036	0.064	0.336
Fpfh1_r0.01 (Godil et al., 2015)	0.033	0.018	0.037	0.316
SNU_1 (Godil et al., 2015)	0.167	0.076	0.119	0.388
Continuous-hough_bw-0.5 (Godil et al., 2015)	0.139	0.075	0.124	0.389
BoC (Tabia and Laga, 2015)	0.721	0.634	0.742	0.787
Improved BoC	**0.761**	**0.653**	**0.767**	**0.816**

Additional experimental results on SHREC'15 are reported in Table 7. The objective of this experiment on SHREC'15 is to retrieve 3D models that are relevant to a query range scan. This task corresponds to a real life scenario where the query is a 3D range scan of an object acquired from an arbitrary view direction. We use the 2D view-based BoC method and compared its performance with state-of-the-art methods. Table 7 shows that the BoC and the improved BoC methods give the best performance. The DSIFT-based and the DCT-based methods achieve moderate results compared to our method. The performance gap between our method and the other four approaches is very important.

6. Conclusion

We proposed a mechanism for early as well as late fusion of Bag of Covariance signatures. While covariance-based description of 3D shapes provides an elegant mechanism for early fusion of heterogeneous features, different metrics and quantization methods can be used for building BoC vocabularies. We showed that the retrieval performance can be significantly improved by fusing multiple BoC signatures. These signatures are produced from multiple vocabularies constructed using different metrics that take into account the different geometries of the space of covariance matrices. We demonstrated the performance of the proposed improved BoC on different shape retrieval tasks including global, partial, non-rigid as well as large scale ones. The improved BoC clearly boosts the performance of the original BoC approach and outperforms the state-of-the-art. The results reveal that different types of quantizers that use different metrics are indeed complementarity.

References

Babenko, A., Lempitsky, V., 2012. The inverted multi-index, in: Computer Vision and Pattern Recognition, IEEE Conference on, IEEE. pp. 3069–3076.

Bai, X., Bai, S., Zhu, Z., Latecki, L.J., 2015. 3d shape matching via two layer coding. Pattern Analysis and Machine Intelligence, IEEE Transactions on 37, 2361–2373.

Bai, X., Rao, C., Wang, X., 2014. Shape vocabulary: A robust and efficient shape representation for shape matching. Image Processing, IEEE Transactions on 23, 3935–3949.

Barra, V., Biasotti, S., 2013. 3d shape retrieval using kernels on extended reeb graphs. Pattern Recognition 46, 2985–2999.

Godil, A., Dutagaci, H., Bustos, B., Choi, S., Dong, S., Furuya, T., Li, H., Link, N., Moriyama, A., Meruane, R., et al., 2015. Range scans based 3d shape retrieval, in: Proceedings of the 2015 Eurographics Workshop on 3D Object Retrieval, Eurographics Association. pp. 153–160.

Gong, Y., Lazebnik, S., Gordo, A., Perronnin, F., 2013. Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval. IEEE Transactions on PAMI 35, 2916–2929.

Jegou, H., Schmid, C., Harzallah, H., Verbeek, J., 2010. Accurate image search using the contextual dissimilarity measure. Pattern Analysis and Machine Intelligence, IEEE Transactions on 32, 2–11.

Lavoué, G., 2012. Combination of bag-of-words descriptors for robust partial shape retrieval. The Visual Computer 28, 931–942.

Li, B., Lu, Y., Li, C., Godil, A., Schreck, T., Aono, M., Burtscher, M., Chen, Q., Chowdhury, N.K., 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., 2015. A comparison of 3d shape retrieval methods based on a large-scale benchmark supporting multimodal queries. Comput. Vis. Image Underst. 131, 1–27.

Marini, S., Paraboschi, L., Biasotti, S., (2007). Shape retrieval contest 2007: Partial matching track. SHREC (in conjunction with SMI) .

Papadakis, P., Pratikakis, I., Theoharis, T., Passalis, G., Perantonis, S., 2008. 3d object retrieval using an efficient and compact hybrid shape descriptor, in: Eurographics Workshop on 3D object retrieval.

Porikli, F., Tuzel, O., Meer, P., 2006. Covariance tracking using model update based on lie algebra, in: CVPR.

Siddiqi, K., Zhang, J., Macrini, D., Shokoufandeh, A., Bouix, S., Dickinson, S., 2008. Retrieving articulated 3-d models using medial surfaces. Mach. Vision Appl. 19.

Tabia, H., Daoudi, M., Vandeborre, J.P., Colot, O., 2011. A new 3d-matching method of nonrigid and partially similar models using curve analysis. IEEE Trans. on PAMI 33.

Tabia, H., Laga, H., 2015. Covariance-based descriptors for efficient 3d shape matching, retrieval, and classification. Multimedia, IEEE Transactions on 17, 1591–1603.

Tabia, H., Laga, H., Picard, D., Gosselin, P.H., 2014. Covariance descriptors for 3d shape matching and retrieval, in: Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, IEEE. pp. 4185–4192.

Tabia, H., Picard, D., Laga, H., Gosselin, P.H., 2013. Compact vectors of locally aggregated tensors for 3d shape retrieval, in: 3DOR.

Wu, Z., Ke, Q., Sun, J., Shum, H.Y., 2009. A multi-sample, multi-tree approach to bag-of-words image representation for image retrieval, in: Computer Vision, 2009 IEEE 12th International Conference on, IEEE. pp. 1992–1999.

Xia, Y., He, K., Wen, F., Sun, J., 2013. Joint inverted indexing, in: Proceedings of the IEEE International Conference on Computer Vision, pp. 3416–3423.