This is the Pre-Published Version.

The following publication Q. Wang, P. Li, W. Zuo and L. Zhang, "RAID-G: Robust Estimation of Approximate Infinite Dimensional Gaussian with Application to Material Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 4433-4441 is available at https://doi.org/10.1109/CVPR.2016.480.

RAID-G: Robust Estimation of Approximate Infinite Dimensional Gaussian with Application to Material Recognition

Qilong Wang[1], Peihua Li[1,*], Wangmeng Zuo[2], Lei Zhang[3]
[1]Dalian University of Technology, [2]Harbin Institute of Technology, [3]Hong Kong Polytechnic University
qlwang@mail.dlut.edu.cn, peihuali@dlut.edu.cn, wmzuo@hit.edu.cn, cslzhang@comp.polyu.edu.hk

Abstract

Infinite dimensional covariance descriptors can provide richer and more discriminative information than their low dimensional counterparts. In this paper, we propose a novel image descriptor, namely, robust approximate infinite dimensional Gaussian (RAID-G). The challenges of RAID-G mainly lie on two aspects: (1) description of infinite dimensional Gaussian is difficult due to its non-linear Riemannian geometric structure and the infinite dimensional setting, hence effective approximation is necessary; (2) traditional maximum likelihood estimation (MLE) is not robust to high (even infinite) dimensional covariance matrix in Gaussian setting. To address these challenges, explicit feature mapping (EFM) is first introduced for effective approximation of infinite dimensional Gaussian induced by additive kernel function, and then a new regularized MLE method based on von Neumann divergence is proposed for robust estimation of covariance matrix. The EFM and proposed regularized MLE allow a closed-form of RAID-G, which is very efficient and effective for high dimensional features. We extend RAID-G by using the outputs of deep convolutional neural networks as original features, and apply it to material recognition. Our approach is evaluated on five material benchmarks and one fine-grained benchmark. It achieves 84.9% accuracy on FMD and 86.3% accuracy on UIUC material database, which are much higher than state-of-the-arts.

1. Introduction

Recently, the covariance matrices as region image representations have attracted increasingly attentions in a number of computer vision tasks, such as pedestrian detection [45], visual tracking [38], image set classification [49], action recognition [24], semantic segmentation [7], Diffusion Tensor Imaging (DTI) segmentation [25], and texture classification [44, 22, 29, 31]. However, covariance descriptors in the original low-dimensional feature space usually have limited capability in encoding richer and more discriminative information [23, 20]. Meanwhile, dramatic increase of feature dimension brings challenges on the robust estimation of covariance representations.

To address limitations of covariance descriptors in the original low-dimensional feature space, one of recent extensions to covariance representations is infinite dimensional covariance descriptors, which usually are significantly superior to the ones constructed in the original feature space [53, 23, 20]. The underlying idea of infinite dimensional covariance descriptors is to map, through some kernel functions, the original features into some Reproducing Kernel Hilbert Space (RKHS) $\mathcal{H}$ in which one constructs covariance descriptors. However, explicit forms of covariances in RKHS often cannot be obtained because of unknown mapping functions, and therefore kernel tricks are exploited. Zhou *et al*. derived a family of probabilistic distance measures (e.g. Bhattacharyya distance and KL divergence) in RKHS for matching infinite dimensional Gaussian models (covariances with additional means) [53]. Harandi *et al*. compared covariance descriptors in RKHS with the distances induced by Bregman divergences. Minh *et al*. generalized the Log-Euclidean metric [2] for matching the infinite dimensional covariance descriptors [20].

The aforementioned methods all depend on computation of Gram matrices and kernel-based classifiers (e.g. kernel SVM), which involve high computational cost and memory usage, unscalable for large scale problems. To tackle these issues, Faraki *et al*. [17] proposed to approximate infinite dimensional covariance descriptors in RKHS by employing two classical explicit feature mappings, namely, random Fourier transform (rFt) [39] and Nyström method [50]. This method is very efficient since it gives approximately finite and explicit forms of infinite dimensional covariances. However, as explained in [47], rFt requires a large number of random projections (several times the dimensionality of the original features) to obtain better approximation,

*Peihua Li is the corresponding author.
The work was supported by the National Natural Science Foundation of China (61471082, 61271093) and the Hong Kong RGC GRF grant (PolyU 5313/13E). We thank NVIDIA corporation for donating GPU.

Methods	Descriptor	Kernels or mappings	Estimator	Metric	Linear SVM ?
Zhou *et al*. [53]	Gaussian	RBF kernel (no explicit mapping)	Ledoit-Wolf estimator	Probabilistic distances in $\mathcal{H}$	No
Harandi *et al*. [23]	Covariance	RBF kernel (no explicit mapping)	Ledoit-Wolf estimator	Bregman Divergences in $\mathcal{H}$	No
Log-HS [20]	Covariance	RBF kernel (no explicit mapping)	Ledoit-Wolf estimator	Log-Hilbert-Schmidt metric	No
Faraki *et al*. [17]	Covariance	$\{$Random Fourier transform, Nyström method$\}$ for RBF kernel	Ledoit-Wolf estimator	Log-Euclidean metric	Yes
RAID-G (Ours)	Gaussian	Explicit feature maps of $\{$Hellinger's kernel, $\mathcal{X}^2$ kernel$\}$	Regularized MLE with von Neumann divergence	Gaussian Embedding and vectorization	Yes

Table 1. Comparison of different infinite dimensional image descriptors.

which may be intractable if the dimensions of the original features themselves are very high. The Nyström method is data-dependent, and needs to compute Gram matrix for a set of training samples. However, it is not easy to select a small number of representative training samples. Based on the Nyström method, Perronnin *et al*. [37] proposed additive kernel principal component analysis to approximate additive kernels.

In this paper, we propose a novel approximate infinite dimensional Gaussian descriptor, which can tackle the issues in [53, 23, 20]. Different from [17], we estimate approximate infinite dimensional Gaussian by exploiting two approximate homogenous additive kernel functions, namely, explicit feature maps of Hellinger's kernel and $\mathcal{X}^2$ kernel, which are previously used to speed up large scale non-linear SVM [47]. Different from Nyström method, our method is data-independent. Compared with rFt of RBF kernel, our method allows for component-wise approximation of homogenous additive kernel, thus leads to more compact mapping vectors and is fit for high dimensional original features.

Alternative extension to covariance representations is to enhance the original features. The results in [7] demonstrated enhancement of original features can improve performance of covariance representations. Recently, many works have shown deep Convolutional Neural Networks (CNN) features perform much better than traditional, hand-crafted features [19, 15, 12]. Based on deep CNN features [9, 43], Cimpoi *et al*. [12] proposed a state-of-the-art texture descriptor called FV-CNN, which modeled outputs of convolutional layer from pre-trained deep CNN with Fisher vector (FV) coding [40]. Similar to [12], we construct infinite dimensional Gaussian descriptor by using the convolutional features as well, which are of high dimension (512 or 1536 in our case). As far as we know, previous works on covariance or Gaussian descriptors have never made such an attempt. The reason may be that, for an input image, usually only a very small number of deep CNN features are available, the dimensions (512 or higher) of which are inherently much higher than those of the traditional ones, making robust estimation of covariances difficult.

It is well known that conventional Maximum Likelihood Estimation (MLE) is not robust to high dimensional problems with a small number of samples [5, 51]. Specifically, covariance matrices in (approximate) RKHS suffer from rank-deficient problem as dimensionality of mapping features is larger than the number of samples. The aforementioned methods [53, 23, 20, 17] simply exploit the Ledoit-Wolf (LW) estimator [28] to tackle this problem, where at the core a small positive number is added to all diagonal entries of one sample covariance matrix. The LW estimator is simple and efficient but has limited capability. In this paper, we propose a regularized MLE to robustly estimate high dimensional covariance in the Gaussian setting. The key idea is to impose structural constraint in the original MLE through the von Neumann matrix divergence [14], which is intimately connected with exponential distributions. Specifically, by encouraging the identity matrix structure in the covariance matrix, we obtain a robust and efficient estimator. This estimator, which we call vN-MLE, has a closed-form expression and can significantly improve performance of approximate infinite dimensional Gaussians.

Comparison of our method with related work [53, 23, 20, 17] is presented in Table 1. The main contribution of this paper is a robust approximate infinite dimensional Gaussian (RAID-G) descriptor, and we apply it to material recognition. We introduce explicit mapping functions and propose a novel robust covariance estimator, obtaining closed-form approximate infinite-dimensional Gaussian descriptors. With Gaussian embedding [34] and vectorization, RAID-G can be fed into a linear classifier (e.g. SVM), which is efficient and scalable to large scale classification problem. Experiments are conducted on five material and one fine-grained benchmarks, and the results demonstrate that RAID-G is a very competitive image descriptor.

1.1. Related work

In this paper, we model images with Gaussian descriptors. Compared with covariance descriptor, Gaussian has additional mean information which has proven useful in [35, 41] and also in our experiments. The Gaussian descriptors with hand-crafted features in the original low-dimensional space have been proposed for image classification [35, 41]. Different from them, we extend Gaussian descriptors to RKHS and employ the deep CNN descriptors as original features. Note that the Gaussian descriptors with deep CNN features in the original space is our baseline.

Robust estimation of covariance matrix has been an important topic in machine learning and statistics. One of the

most commonly used methods obtain a robust estimation by shrinking the eigenvalues of sample covariance matrix [13, 28, 10]. Alternative methods to handle this problem are regularized MLE [5, 51, 52]. Many regularizers (e.g. sparsity, group-sparsity, or low rank) are proposed to estimate high dimensional parameters in recent years. As suggested in [52], these constraints for the covariance estimation in regularized MLE lead to non-convex problems. Won *et al*. [51] proposed a regularized MLE with condition number constraint, which can be solved by a Steinian-type shrinkage of eigenvalues. It has two parameters to be determined and lead to additional non-trivial computational load. Our method is a regularized MLE method, where we introduce a novel regularizer, i.e., the von Neumann divergence between covariance matrix and the identity matrix, which is clearly different from the previous methods.

2. Preliminary

This section briefly reviews the traditional MLE-based method to estimate Gaussian descriptors. Given a set of N features $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_N | \mathbf{x}_k \in \mathbb{R}^d\}$, we model their distribution with a Gaussian probability density

$$p(\mathbf{x}) = |2\pi\boldsymbol{\Sigma}|^{-\frac{1}{2}} \exp\big(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\big),$$

with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$, where $|\cdot|$ and T indicate matrix determinant and transpose, respectively.

The likelihood function of the sample set is $L(\mathbf{X}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \prod_{k=1}^{N} p(\mathbf{x}_k; \boldsymbol{\mu}, \boldsymbol{\Sigma})$, where $p(\mathbf{x}_k; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes the probability of $\mathbf{x}_k$ given the parameters of the mean and covariance. By MLE, the mean of samples can be estimated as

$$\boldsymbol{\mu} = \frac{1}{N}\sum\nolimits_{k=1}^{N} \mathbf{x}_k. \tag{1}$$

Then estimation of covariance matrix reduces to the following optimization problem:

$$\min_{\boldsymbol{\Sigma}} \frac{N}{2}\log|\boldsymbol{\Sigma}| + \frac{1}{2}\mathrm{tr}(\boldsymbol{\Sigma}^{-1}\mathbf{S}), \tag{2}$$

where $\mathbf{S} = \frac{1}{N}\sum_{k=1}^{N}(\mathbf{x}_k - \boldsymbol{\mu})(\mathbf{x}_k - \boldsymbol{\mu})^T$ is the sample covariance matrix, and $\mathrm{tr}(\cdot)$ indicates trace of matrix. By minimizing the objective (2) one obtains

$$\boldsymbol{\Sigma} = \frac{1}{N}\sum\nolimits_{k=1}^{N}(\mathbf{x}_k - \boldsymbol{\mu})(\mathbf{x}_k - \boldsymbol{\mu})^T, \tag{3}$$

which is equal to the sample covariance matrix $\mathbf{S}$. Hence, we can compute Gaussian descriptors in the original feature space with Eq. (1) and Eq. (3).

3. Proposed method

We start with approximate the infinite dimensional Gaussians by explicit feature mappings. Next, we propose the regularized MLE for robust estimation of high dimensional covariances. Then we describe application of the proposed Gaussian descriptor to material recognition. Finally we provide computational complexity analysis.

3.1. Approximate infinite dimensional Gaussians

In order to obtain infinite dimensional Gaussian descriptor, we need to compute mean vector and covariance matrix in RKHS. Given a set $\mathbf{X}$ of features (deep CNN features in our case) in the original feature space $\mathbb{R}^d$, we map $\mathbf{x}_k$ into a RKHS, denoted by $\mathcal{H}$, by some mapping function $\phi : \mathbb{R}^d \to \mathcal{H}, \mathbf{x}_k \mapsto \phi(\mathbf{x}_k)$, where $\phi(\cdot)$ is a function of much higher dimension or even infinite dimension. Then in $\mathcal{H}$ the mean $\widehat{\boldsymbol{\mu}}$ and sample covariance $\widehat{\mathbf{S}}$ of mapping features $\phi(\mathbf{x}_k)$ can be computed as

$$\widehat{\boldsymbol{\mu}} = \frac{1}{N}\sum\nolimits_{k=1}^{N}\phi(\mathbf{x}_k), \quad \widehat{\mathbf{S}} = \frac{1}{N}\Phi(\mathbf{X})\mathbf{J}\Phi(\mathbf{X})^T. \tag{4}$$

Here $\Phi(\mathbf{X}) = [\phi(\mathbf{x}_1)\ \cdots\ \phi(\mathbf{x}_N)]$ and $\mathbf{J} = \mathbf{I}_N - \frac{1}{N}\mathbf{1}_N\mathbf{1}_N^T$ is the centering matrix, where $\mathbf{I}_N$ is the identity matrix of order N and $\mathbf{1}_N$ is a N-dimensional vector with all elements being ones.

Though the mapping function ϕ usually is unknown, by the commonly used kernel trick $k(\mathbf{x}_k, \mathbf{x}_j) = \langle\phi(\mathbf{x}_k), \phi(\mathbf{x}_j)\rangle$ (e.g. the RBF kernel $k(\mathbf{x}_k, \mathbf{x}_j) = \exp(-\|\mathbf{x}_k - \mathbf{x}_j\|_2^2/2\sigma^2)$), one can obtain the inner product of any pair of mapping functions. However, such methods often can not obtain the explicit forms of $\widehat{\boldsymbol{\mu}}$ and $\widehat{\mathbf{S}}$, and suffer from high computational cost [53, 23, 20]. Faraki *et al*. [17] proposed rFt and Nyström method to obtain approximate mapping function ϕ. In contrast, we introduce approximate mapping functions corresponding to the Hellinger's kernel and $\mathcal{X}^2$ kernel.

The Hellinger's and $\mathcal{X}^2$ kernels have been successfully used in one of popular image classification methods, namely the Bag-of-visual words model. The Hellinger's kernel, also known as Bhattacharyya's coefficient, is of the form $k(\mathbf{x}_k, \mathbf{x}_j) = \sum_{m=1}^{d}\sqrt{x_k^m x_j^m}$, where x_k^m is the $m-$th element of $\mathbf{x}_k$. Clearly the mapping function ϕ_{Hel} of the Hellinger's kernel has closed-form expression

$$\phi_{Hel}(\mathbf{x}_k) = \sqrt{\mathbf{x}_k}. \tag{5}$$

Here $\sqrt{\cdot}$ should be interpreted as element-wise square root operation. Hence, its computation complexity is linear in dimension of the original features, rendering it very efficient for handling high dimensional features.

The $\mathcal{X}^2$ kernel is given by $k(\mathbf{x}_k, \mathbf{x}_j) = \sum_{m=1}^{d} 2(x_k^m x_j^m)/(x_k^m + x_j^m)$, which is a γ-homogeneous kernel widely used in histogram matching. The explicit mapping function of $\mathcal{X}^2$ kernel can be written as

$$\phi_{Chi}(\mathbf{x}_k) = e^{-i\omega\log(\mathbf{x}_k)}\sqrt{\mathbf{x}_k \mathrm{sech}(\pi\omega)},$$

where i is the imaginary unit and ω is frequency. For efficiency, ϕ_{Chi} is approximated by three sampled points of the following form[47]:

$$\widehat{\phi}_{Chi}(\mathbf{x}_k) = \sqrt{\mathbf{x}_k}\left[\sqrt{L}, \sqrt{2L\mathrm{sech}(L\pi)}\cos(L\log(\mathbf{x}_k)), \sqrt{2L\mathrm{sech}(L\pi)}\sin(L\log(\mathbf{x}_k))\right]^T, \quad (6)$$

where L is the sampling period. The map $\widehat{\phi}_{Chi}(\mathbf{x}_k)$ is three times of the dimension of $\mathbf{x}_k$. More accurate approximations are available which, however, will incur much higher dimensions of the mapping function. Eq. (6) involves only element-wise square root, $\log$, $\sin$ and $\cos$, so its computational complexity also is linear in the dimension of the original features.

The two mappings (5) and (6) are both data-independent, requiring no training stage and allowing for explicit forms of approximate infinite dimensional Gaussian in RKHS. Note that the Hellinger's and $\mathcal{X}^2$ kernels are designed for histogram matching. We in this paper use them to transform the outputs of the convolutional layer of CNNs, since they are histogram-like features in the sense that they are non-negative and sparse. The mapping (5) shares similar philosophy with rootSIFT [1], which is applied to histogram-like SIFT features and achieves non-trivial improvement in both image retrieval and classification. To our best knowledge, we are among the first who apply the Hellinger's and $\mathcal{X}^2$ kernel mappings to CNN features.

3.2. Robust estimation of approximate infinite dimensional Gaussian

So far, we can compute approximate infinite dimensional Gaussian with Eq. (4), Eq. (5) or Eq. (6) based on the classical MLE. However, it can not work well when dimension of the original features is very high (512 or 1536 in our case) and the number of samples is small. Imposing structural constraints is a commonly used method in robust covariance estimation. The well-known LW estimator [28] and its variants explore the solution among the convex combination of the sample covariance matrices and a prior SPD matrix (typically the identity matrix). Our work is motivated by these methods but the key difference is that we impose the constraint through the von Neumann matrix divergence [14] to measure the similarity between matrices.

Let ψ be a real-valued differential convex function over matrices. The Bregman matrix divergence is defined as

$$D_\psi(\mathbf{A};\mathbf{B}) = \psi(\mathbf{A}) - \psi(\mathbf{B}) - \langle \nabla\psi(\mathbf{A}), \mathbf{A}-\mathbf{B}\rangle \quad (7)$$

where $\langle\cdot,\cdot\rangle$ denotes the inner product and ∇ is the gradient with respect to matrix. In this paper, we only consider D_ψ defined on SPD matrices. When ψ is adopted to be the von Neumann entropy $\psi(\mathbf{A}) = \mathrm{tr}(\mathbf{A}\log(\mathbf{A}) - \mathbf{A})$ [36], where tr denotes the matrix trace and the resulting divergence D_ψ is called von Neumann matrix divergence:

$$D_{\text{vN}}(\mathbf{A},\mathbf{B}) = \mathrm{tr}(\mathbf{A}(\log(\mathbf{A}) - \log(\mathbf{B})) - \mathbf{A} + \mathbf{B}), \quad (8)$$

which is also known as the quantum relative entropy in the quantum mechanics. Note that D_{vN} has been successfully used in the low rank kernel learning problem [27]. The divergence D_{vN} is bounded, invariant to orthonormal transformation. There is a unique Bregman divergence associated with every member of the exponential family of probability distributions. This recommends the matrix divergences for solving statistical related problems.

The regularizer we introduced in the MLE is the von Neumann divergence $D_{\text{vN}}(\mathbf{I}, \widehat{\mathbf{\Sigma}})$, which encourages the structure of the identity matrix $\mathbf{I}$ in the covariance matrix $\widehat{\mathbf{\Sigma}}$. Our robust estimator of covariance matrix is regularized maximum likelihood estimator:

$$\min_{\widehat{\mathbf{\Sigma}}} \log|\widehat{\mathbf{\Sigma}}| + \mathrm{tr}(\widehat{\mathbf{\Sigma}}^{-1}\widehat{\mathbf{S}}) + \alpha D_{\text{vN}}(\mathbf{I}, \widehat{\mathbf{\Sigma}}), \quad (9)$$

where $0 < \alpha < 1$ is a regularizing parameter. As shown in [52], the type of regularized MLE problems is non-convex, but our estimator can obtain a unique optimal solution which has an analytic form. The following theorem gives our conclusion.

Theorem 1 *Let the singular value decomposition (SVD) of* $\widehat{\mathbf{S}}$ *be* $\widehat{\mathbf{S}} = \widehat{\mathbf{U}}\mathrm{diag}(\delta_k)\widehat{\mathbf{U}}^T$, *where* $\mathrm{diag}(\delta_k)$ *is the diagonal matrix of the singular values in decreasing order and* $\widehat{\mathbf{U}}$ *is the orthogonal matrix consisting of the eigenvectors. Then the optimal solution to the problem (9) can be computed as*

$$\widehat{\mathbf{\Sigma}} = \widehat{\mathbf{U}}\mathrm{diag}(\lambda_k)\widehat{\mathbf{U}}^T,$$
$$\lambda_k = \sqrt{\left(\frac{1-\alpha}{2\alpha}\right)^2 + \frac{\delta_k}{\alpha}} - \frac{1-\alpha}{2\alpha}. \quad (10)$$

The proof of Theorem 1 is given in the supplementary material. The proposed estimator essentially consists of a non-linear transformation of the eigenvalues of the sample covariance. It has analytic form and thus can be computed very efficiently, well suitable for large-scale problem.

We mention that other regularizers $D_{\text{vN}}(\mathbf{P}, \widehat{\mathbf{\Sigma}})$, where $\mathbf{P}$ is some known SPD matrix standing for a prior knowledge about the covariance matrix, can be used in our robust estimator (9). Such regularizers impose stronger structural constraint on the covariance than the naive identity matrix used in the current paper. We may obtain additional, potential benefits by using these regulaizers, but cannot derive analytic solutions.

3.3. Application to material recognition

This section presents application to material recognition of our RAID-G descriptor. Material recognition has attracted growing attentions in recent years [6, 32, 42, 11, 4],

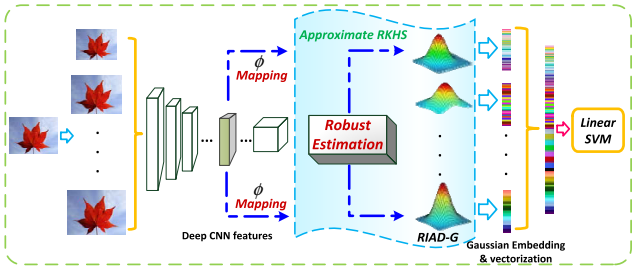

Figure 1. Overview of recognition paradigm with the proposed robust approximate infinite dimensional Gaussian.

possibly due to its broad applications in automatical types of household wastes discrimination, robotic navigation and assisted driving, etc. We adopt deep CNN features as the original features. Note that we have not seen previous attempts which use very high-dimensional CNN features in the context of covariance or Gaussian descriptors. The overview of our recognition paradigm is illustrated in Fig. 1. Given an input image, we extract CNN features in multi-scale setting by rescaling the image by some factors (e.g. $2^{\{-1,0,1,2\}}$). We employ a 19-layer convolutional network (VGG-VD19) [43] pre-trained on ILSVRC database to obtain 512-dimensional outputs of the last convolutional layer via MatConvNet implementation [46].

For the original CNN features, we first map them into RKHS through Eq. (5) or Eq. (6), then we compute mean $\widehat{\boldsymbol{\mu}}$ and sample covariance $\widehat{\mathbf{S}}$ through Eq. (4), and finally estimate covariance matrix $\widehat{\boldsymbol{\Sigma}}$ with the proposed vN-MLE method. As the space of Gaussian models forms a Riemannian manifold, we embed them into the space of symmetric positive definite (SPD) matrices by using the method in [30] with minor modification:

$$\mathcal{N}(\widehat{\boldsymbol{\mu}}, \widehat{\boldsymbol{\Sigma}}) \sim \mathbf{G}(\beta) = \begin{bmatrix} \widehat{\boldsymbol{\Sigma}} + \beta^2 \widehat{\boldsymbol{\mu}}\widehat{\boldsymbol{\mu}}^T & \beta\widehat{\boldsymbol{\mu}} \\ \beta\widehat{\boldsymbol{\mu}}^T & 1 \end{bmatrix}, \qquad (11)$$

where $\beta > 0$ is a parameter to balance the dimension and orders of magnitude between mean vector and covariance matrix. It is easy to see that $\mathbf{G}(\beta)$ is a SPD matrix.

Since we embed Gaussian models in the space of SPD matrices, the metrics based on the geometry of this manifold is ready for use. However, most of them entangle in measuring distance such that non-linear kernel machines have to be resorted. The Log-Euclidean metric is decoupled and so can be combined with a linear SVM. However, surprisingly, its performance is not satisfactory compared with the Frobenius norm (F-norm) between the SPD matrices (see comparison in Section 4.6). In addition, F-norm is computationally more efficient, particularly for much higher dimensional SPD matrices. In terms of these considerations, we simply view SPD matrices as being in the Euclidean space, which are directly fed to a linear SVM for training and testing after a vectorizing procedure. We implement a one-vs-all classifier by using LIBSVM package [8].

3.4. Computational complexity of RAID-G

Given CNN features extracted by CPU or GPU, RAID-G mainly includes three steps, namely, feature mappings (Eq. (5) or Eq. (6)), robust estimation (Eq. (10)), and Gaussian embedding (Eq. (11)). Let dimension of local descriptors be d, complexity of feature mapping via Eq. (5) and Eq. (6) are $O(d)$ and $O(3d)$, respectively. The robust estimation requires eigenvalue decomposition with $O(d^3)$ and $O(27d^3)$ costs for Hellinger's kernel mapping and $\mathcal{X}^2$ kernel mapping, respectively. This cost is dominant in the computation of our RAID-G. The Gaussian embedding takes $O(d^2)$ cost. We use a single GeForce GTX Titan Black GPU to extract deep CNN features which can process about 20 images per second. For each image, RAID-G with ϕ_{Hel} (resp. ϕ_{Chi}) takes about 0.3s (resp. 8s) on a workstation with Core i7 CPU at 3.8GHz by using Matlab programming.

4. Experiments

In this section, we evaluate the proposed methods on five material databases, which are briefly described as follows.
Flickr Material Database (FMD) [21] consists of 1,000 images of 10 material categories. It is a real-world dataset which contains appearance changes and high intra-class variations. We randomly pick 50 training and 50 testing images per class.
UIUC material [32] is a challenging benchmark for recognizing material in the wild. It contains 18 categories, 12 samples per category. We randomly choose half of the samples from each category for training, and the rest for testing.
KTH-TIPS 2b [6] is a widely used benchmark with scale, viewpoint and illumination changes. It contains 4,752 images in 11 classes where each class includes 4 subsets with 108 samples. For each class we randomly select one sample per subset for training and the remaining for testing.
Describable Textures Dataset (DTD) [11] consists of 5,640 material images collected from Internet (in the wild) which are jointly labeled with 47 classes. We utilize ten predefined splits in [11] to evaluate the proposed methods.
Open Surfaces [3] is recently proposed in the computer graphics community, and can be used for material recognition in clutter. In this paper, we exploit its subset of 10,422 images [12], which consists of 53,915 labeled material segments of 22 classes.

The following methods are evaluated in our experiments: COV-CNN, Gau-CNN and RoG-CNN which respectively indicates covariance descriptor, Gaussian descriptor with LW estimator, Gaussian descriptor with the proposed vN-MLE method, all using CNN features in the original space $\mathbb{R}^d$; RAID-G-CNN-Hel and RAID-G-CNN-Chi which indicate Gaussian descriptor with our vN-MLE method, both

Methods	FMD	UIUC Material	KTH-TIPS 2b	DTD	Open Surfaces
COV-CNN	80.2 ± 1.1	80.5 ± 3.6	76.7 ± 2.8	70.1 ± 1.2	55.0
Gau-CNN	81.3 ± 1.4	81.7 ± 2.9	77.5 ± 2.4	70.5 ± 1.5	55.7
RoG-CNN	83.6 ± 1.6	84.5 ± 1.8	79.5 ± 1.5	73.9 ± 1.1	58.9
RAID-G-CNN-Hel	84.4 ± 1.3	85.7 ± 2.1	80.4 ± 1.2	75.8 ± 1.4	60.3
RAID-G-CNN-Chi	$\mathbf{84.9 \pm 1.4}$	$\mathbf{86.3 \pm 2.9}$	81.3 ± 1.6	$\mathbf{76.4 \pm 1.1}$	**61.1**
FC [12]	77.4 ± 1.8	75.9 ± 2.3	75.4 ± 1.5	62.9 ± 0.8	43.4
FV-CNN [12]	79.8 ± 1.8	80.5 ± 2.7	$\mathbf{81.8 \pm 2.5}$	72.3 ± 1.0	59.5
FC + FV-CNN* [12]	82.4 ± 1.5	82.6 ± 2.1	81.1 ± 2.4	74.7 ± 1.0	60.9
State-of-the-art I	60.6 [42]	60.1 [18]	70.7 ± 1.6 [16]	61.2 ± 1.0 [40]	39.8 [40]
State-of-the-art II	66.5 ± 1.5 [4]	66.6 ± 3.1 [22]	77.3 ± 2.3 [11]	66.7 ± 0.9 [11]	-

Table 2. The accuracy (%) of various methods on five material benchmarks. $*$: The score level fusion is used to combine FC and FV-CNN.

using CNN features mapped to RKHS via the mapping functions ϕ_{Hel} and ϕ_{Chi}, respectively.

4.1. Experimental evaluation

The results of the proposed methods and state-of-the-art methods on five material databases are presented in Table 2.

Covariance vs. Gaussian The Gaussian descriptors always outperform covariance descriptors on all databases, achieving non-trivial improvements with little additional cost. We attribute this to combination of mean information, which is also reported in [35, 41]. The parameter β in Eq. (11) is set to 0.3 on FMD, UIUC, KTH-TIPS 2b and 0.1 on DTD, Open Surfaces, respectively, by cross validation.

Robust estimation RoG-CNN outperforms Gau-CNN on all databases and obtains more than 2.7% gains on average. This big performance improvements demonstrate that the proposed vN-MLE estimator has better capability to deal with very high-dimensional data. The covariance estimation will be further evaluated in Section 4.2.

Kernel mappings RAID-G-CNN-Hell and RAID-G-CNN-Chi can achieve about 1.1% and 1.8% gains over RoG-CNN on average, respectively. The improvement over RoG-CNN demonstrates benefits of Gaussian descriptors constructed in RKHS over those constructed in the original space. The $\mathcal{X}^2$ kernel mapping outperforms Hellinger's while the latter is more efficient. Notably, RAID-G with ϕ_{Hel} and ϕ_{Chi} can improve Gaussian descriptors in the original space (Gau-CNN) with LW estimator by about 3.8% and 4.5% on average, respectively. It indicates effectiveness of our EFM and vN-MLE method in estimation of approximate infinite dimensional Gaussian descriptors.

Comparison with state-of-the-arts From Table 2, we see that RAID-G performs much better than deep CNN features based methods [16, 11, 4], and has clear advantages over FC, in which fully-connected (FC) layer outputs of deep networks are fed to SVM for classification. Moreover, RAID-G outperforms FV-CNN and FC+FV-CNN, and achieves the best results on all benchmarks except KTH-TIPS 2b. RAID-G is slightly inferior to FV-CNN on KTH-TIPS 2b database, but their results are comparable. Finally, we mention that compared with [42, 40, 18, 22] employing classical hand-crafted features, RAID-G with deep CNN features achieves 10%$\sim$20% improvements.

Methods	FMD	UIUC Material
Gau-CNN (LW)	81.3 ± 1.4	81.7 ± 2.9
Gau-CNN (Stein)	81.9 ± 0.7	82.2 ± 1.8
Gau-CNN (MMSE)	81.2 ± 1.2	80.9 ± 1.9
Gau-CNN (EL-SP)	81.5 ± 1.6	82.0 ± 2.3
RoG-CNN (vN-MLE)	$\mathbf{83.6 \pm 1.6}$	$\mathbf{84.5 \pm 1.8}$
Gau-CNN-Chi (LW)	83.1 ± 0.9	81.6 ± 4.1
Gau-CNN-Chi (Stein)	83.2 ± 0.8	83.6 ± 3.0
Gau-CNN-Chi (MMSE)	83.1 ± 0.8	82.0 ± 4.3
Gau-CNN-Chi (EL-SP)	83.2 ± 1.1	82.1 ± 3.1
RAID-G-CNN-Chi (vN-MLE)	$\mathbf{84.9 \pm 1.4}$	$\mathbf{86.3 \pm 2.9}$

Table 3. Comparison with various robust estimators on FMD and UIUC material databases.

4.2. Effect of robust covariance estimation

Here, we compare the proposed vN-MLE with four robust covariance estimation methods (i.e. LW estimator [28], Stein estimator [13], MMSE estimator [10] and elementary sparse (EL-SP) estimator [52]) on FMD and UIUC material database. The LW estimator is implemented by adding a small scalar ($1e-3$ in our case) to diagonal elements of covariance. The Stein and MMSE estimators are implemented as described in their respective papers. For EL-SP, we employ element-wise hard thresholding, and decide the value of threshold by cross validation.

As Table 3 presents, in the original feature space, vN-MLE outperforms LW, Stein, MMSE and EL-SP on FMD and UIUC by over 1.7% and 2.3%, respectively. In RKHS, the advantage of regularized MLE is still obvious when approximate $\mathcal{X}^2$ kernel mapping is used, particularly on UIUC. These comparisons demonstrate that vN-MLE is superior to the competing methods in the very high-dimensional setting.

4.3. Comparison with explicit feature mappings

The explicit feature mappings result in analytic forms of RAID-G. The mapping function of Hellinger's kernel does

Methods	FMD	UIUC Material
RAID-G-CNN-rFt (1x)	79.7 ± 1.6	80.6 ± 2.2
RAID-G-CNN-rFt (3x)	80.6 ± 2.3	81.8 ± 2.7
RAID-G-CNN-Nyström (1x)	82.2 ± 2.2	83.3 ± 3.1
RAID-G-CNN-Nyström (3x)	82.8 ± 1.9	84.0 ± 2.7
RAID-G-CNN-Hel	$\mathbf{84.4 \pm 1.3}$	$\mathbf{85.7 \pm 2.1}$
RAID-G-CNN-Chi	$\mathbf{84.9 \pm 1.4}$	$\mathbf{86.3 \pm 2.9}$
CDL_{rFt} [17]	-	47.4 ± 3.1
$\text{CDL}_{Nyström}$ [17]	-	46.3 ± 2.6

Table 4. Effects of various feature mappings on FMD and UIUC material database.

not change the dimension of the original features (1x), and that of $\mathcal{X}^2$ kernel increases the dimension to 3 times that of the original ones (3x). We compare with two common mappings, i.e., rFt and Nyström method on FMD and UIUC material database. For fair comparison, we set the dimensions of the mapping features in rFt and Nyström method to 1x and 3x, respectively, and use the Gaussian descriptors (identified as SPD matrices using Eq. (11)). Note that larger number of basis in rFt and Nyström method is unaffordable due to rapidly growing dimensions of the mapping features. We implement both rFt and the Nyström method with the RBF kernels, whose bandwidths σ are set as $3e2$ and $1e2$ on FMD and UIUC, respectively, by cross-validation.

The comparison results are shown in Table 4, it can be seen that our methods always outperform rFt and the Nyström method. The rFt produces unsatisfactory results as a large number of random projections are required for better approximation, as mentioned in [47]. The Nyström method achieves favorable results, but it is data-dependent requiring a training stage. Finally, we mention that the proposed RAID-G-CNN achieves about 40% gains over the approximate infinite dimensional covariance descriptors with hand-crafted features [17].

4.4. Comparison with infinite dimensional covariance

Using the KTH-TIPS 2b database, we compare with two infinite dimensional covariance descriptors [23, 20] which are closely related to RAID-G. Following the settings in [23, 20], we randomly choose three samples per subset in each class for training and the remaining for testing. Meanwhile, we also compute RAID-G with 23D hand-crafted features, which consists of R, G, B color intensities and 20 Gabor filters with 4 orientations and 5 scales. We adopt a linear SVM to RAID-G with both hand-crafted features and deep CNN features. Note that, although hand-crafted features are not histogram-like ones, their elements always are nonnegative so we can compute RAID-G with the Hellinger's and $\mathcal{X}^2$ kernel mappings.

The results of the baseline, namely, covariance descriptors in the original space with Log-Euclidean metric and RBF kernel SVM, and two infinite dimensional covariance descriptors [23, 20] in Table 5 are duplicated from [20]. When hand-crafted features are used, RAID-G and the methods in [23, 20] all significantly outperform the baseline. The methods in [23, 20] are slightly better than RAID-G. However, if employing high dimensional deep CNN features, RAID-G achieves more than 7% improvements over infinite dimensional covariance descriptors [23, 20], where CNN features cannot be used due to unaffordable cost (see complexity analysis therein). The gains over [17, 23, 20] in Table 4 and Table 5 demonstrate that RAID-G can flexibly handle high dimensional deep CNN features while bringing significant performance improvement.

Methods	Accuracy (in %)
RAID-G-Hel (23D Handcrafted features)	78.8 ± 4.8
RAID-G-Chi (23D Handcrafted features)	78.2 ± 4.7
RAID-G-CNN-Hel	$\mathbf{89.0 \pm 5.4}$
RAID-G-CNN-Chi	$\mathbf{89.3 \pm 4.5}$
Log-E RBF (baseline) (23D Handcrafted features)	74.1 ± 7.4
Harandi *et al*. [23] (23D Handcrafted features)	80.1 ± 4.6
Log-HS [20] (23D Handcrafted features)	81.9 ± 3.3

Table 5. Comparison with infinite dimensional covariance descriptors on KTH-TIPS 2b database. We randomly select three training samples for per subset in each class and the remaining for testing. Combination of the CNN features with the methods in [23, 20] will be computationally prohibitive (see complexity analysis therein).

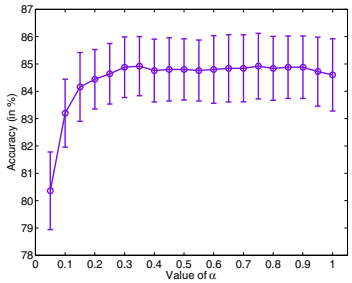

Figure 2. Accuracy against α in vN-MLE method with RAID-G-CNN-Chi on FMD.

4.5. Effect of α in vN-MLE

Our vN-MLE method has only one parameter α, which makes a tradeoff for the degree of regularization based on the von Neumann divergence. Here, we assess the effect of α in Eq. (10) on classification performance. We use RAID-G-CNN-Chi as image representation and FMD for evaluation. The results are illustrated in Fig. 2. We can see that the best result (84.9%) is achieved at $\alpha = 0.75$, and the accuracy does not change much for $\alpha = [0.3 \sim 0.9]$, which indicates that vN-MLE is insensitive in a wide range of α. We set α to 0.75 throughout all our experiments.

Databases	Log-Euclidean metric	F-norm	$\triangle_+/\triangle_-$
FMD	82.3 ± 1.3	84.4 ± 1.3	$+2.1$
UIUC Material	80.6 ± 4.3	85.7 ± 2.1	$+5.1$
KTH-TIPS 2b	79.0 ± 2.0	80.4 ± 1.2	$+1.4$
DTD	69.1 ± 0.9	75.8 ± 1.4	$+6.7$
Open Surfaces	57.9	60.3	$+2.4$
CUB200-2011	75.6	81.4	$+5.8$

Table 6. Comparison of Log-Euclidean metric and F-norm distance with RAID-G-CNN-Hel on various benchmarks.

4.6. Comparison of metrics between covariances

The Log-Euclidean metric and F-norm are two kinds of decoupled metrics for matching covariance matrices. We compare them by using RAID-G-CNN-Hel as image representation on various benchmarks. The comparison results are shown in Table 6. It can be seen that F-norm is superior to Log-Euclidean metric on all benchmarks, and achieves about 4% gains on average. The reason for the inferior performance of Log-Euclidean metric may be that logarithm of eigenvalues in the Log-Euclidean metric harm the effect of shrinkage in the proposed vN-MLE. Note that the focus of this paper is on robust estimation for approximate infinite dimensional Gaussians, so a full evaluation of different metrics between covariance matrices is beyond our scope.

5. Application to fine-grained recognition

Finally, we apply RAID-G-CNN to fine-grained recognition problem in order to assess the generality of our methods. Note that we do not use any part detection methods, bounding box and fine-tuning of CNN models. We employ Bird CUB200-2011 database including 11,788 images from 200 species [48]. It is a challenging benchmark with large intra-class variation and small inter-class variation. We report accuracy on the provided training/testing split. The results are illustrated in Fig. 3, and we can see that our RAID-G-CNN significantly outperforms FC and FV-CNN. Moreover, without bounding box and fine-tuning technique, RAID-G-CNN is superior to state-of-the-art methods specifically designed for fine-grained recognition, e.g. [26] where segmentation, alignment and part models based on eight-layer CNNs are exploited, and [33] where an eight-layer CNN and sixteen-layer CNN are combined. The results on the fine-grained problem indicate that RAID-G has potential in diverse vision tasks.

6. Conclusion

In this paper, we study the problem of robust estimation of approximate infinite dimensional Gaussians. To our best knowledge, our work is among the first which constructs Gaussian or approximate infinite dimensional Gaussian using very high-dimensional features (e.g. CNN features of 512 dimension), and which reveals, through the proposed vN-MLE method, the crucial influence of robust estimation on high dimensional Gaussian descriptors. Our RAID-G achieves very competitive results on most of the material benchmarks. The feature mappings we introduced are general and can be used in other types of approximate infinite dimensional descriptors. The vN-MLE method is suitable for high dimensional covariance estimation, which can be extended to high dimensional mixture model estimation, such as Gaussian mixture model. In future, we will apply RAID-G to a diversity of computer vision tasks.

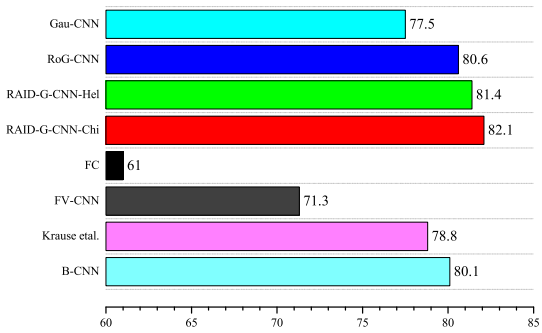

Figure 3. Results of different methods on Bird CUB200-2011 database without part detection, bounding box and fine-tuning. Note that by using bounding box and fine-tuning, Krause *et al*. [26] and B-CNN [33] can achieve 82.8% and 85.1%, respectively.

References

[1] R. Arandjelovic and A. Zisserman. Three things everyone should know to improve object retrieval. In *CVPR*, 2012.

[2] V. Arsigny, P. Fillard, X. Pennec, and N. Ayache. Fast and simple calculus on tensors in the Log-Euclidean framework. In *MICCAI*, 2005.

[3] S. Bell, P. Upchurch, N. Snavely, and K. Bala. Opensurfaces: a richly annotated catalog of surface appearance. *ACM TOG.*, 32(4):1–17, 2013.

[4] S. Bell, P. Upchurch, N. Snavely, and K. Bala. Material recognition in the wild with the materials in context database. In *CVPR*, 2015.

[5] P. J. Bickel and E. Levina. Regularized estimation of large covariance matrices. *The Annals of Statistics*, 2006.

[6] B. Caputo, E. Hayman, and P. Mallikarjuna. Class-specific material categorisation. In *ICCV*, 2005.

[7] J. Carreira, R. Caseiro, J. Batista, and C. Sminchisescu. Free-form region description with second-order pooling. *IEEE TPAMI*, 37(6):1177–1189, 2015.

[8] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM TIST*, 2(3):27, 2011.

[9] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the devil in the details: Delving deep into convolutional nets. In *BMVC*, 2014.

[10] Y. Chen, A. Wiesel, Y. C. Eldar, and A. O. Hero. Shrinkage algorithms for mmse covariance estimation. *IEEE TSP*, 58(10):5016–5029, 2010.

[11] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, and A. Vedaldi. Describing textures in the wild. In *CVPR*, 2014.

[12] M. Cimpoi, S. Maji, and A. Vedaldi. Deep filter banks for texture recognition and segmentation. In *CVPR*, 2015.

[13] M. J. Daniels and R. E. Kass. Shrinkage estimators for covariance matrices. *Biometrics*, 57(4):1173–1184, 2001.

[14] I. S. Dhillon and J. A. Tropp. Matrix nearness problems with bregman divergences. *SIAM J. MAP*, 29(4):1120–1146, 2008.

[15] M. Dixit, S. Chen, D. Gao, N. Rasiwasia, and N. Vasconcelos. Scene classification with semantic Fisher vectors. In *CVPR*, 2015.

[16] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *ICML*, 2014.

[17] M. Faraki, M. Harandi, and F. Porikli. Approximate infinite-dimensional region covariance descriptors for image classification. In *ICASSP*, 2015.

[18] M. Faraki, M. Harandi, and F. Porikli. Material classification on symmetric positive definite manifolds. In *WACV*, 2015.

[19] Y. Gong, L. Wang, R. Guo, and S. Lazebnik. Multi-scale orderless pooling of deep convolutional activation features. In *ECCV*, 2014.

[20] M. Ha Quang, M. San Biagio, and V. Murino. Log-Hilbert-Schmidt metric between positive definite operators on Hilbert spaces. In *NIPS*, 2014.

[21] L. Haran, R. Rosenholtz, and E. H. Adelson. Material perception: What can you see in a brief glance? *Journal of Vision*, 9(8):784, 2009.

[22] M. T. Harandi, M. Salzmann, and R. Hartley. From manifold to manifold: Geometry-aware dimensionality reduction for SPD matrices. In *ECCV*, 2014.

[23] M. T. Harandi, M. Salzmann, and F. M. Porikli. Bregman divergences for infinite dimensional covariance matrices. In *CVPR*, 2014.

[24] M. E. Hussein, M. Torki, M. A. Gowayyed, and M. El-Saban. Human action recognition using a temporal hierarchy of covariance descriptors on 3D joint locations. In *IJCAI*, 2013.

[25] S. Jayasumana, R. I. Hartley, M. Salzmann, H. Li, and M. T. Harandi. Kernel methods on Riemannian manifolds with Gaussian RBF kernels. *IEEE TPAMI*, 37:2464–2477, 2015.

[26] J. Krause, H. Jin, J. Yang, and L. Fei-Fei. Fine-grained recognition without part annotations. In *CVPR*, 2015.

[27] B. Kulis, M. A. Sustik, and I. S. Dhillon. Low-rank kernel learning with Bregman matrix divergences. *JMLR*, 10:341–376, 2009.

[28] O. Ledoit and M. Wolf. A well-conditioned estimator for large-dimensional covariance matrices. *JMA*, 88(2):365–411, 2004.

[29] P. Li and Q. Wang. Local Log-Euclidean covariance matrix (L^2ECM) for image representation and its applications. In *ECCV*, 2012.

[30] P. Li, Q. Wang, and L. Zhang. A novel Earth Mover's Distance methodology for image matching with Gaussian mixture models. In *ICCV*, 2013.

[31] P. Li, Q. Wang, W. Zuo, and L. Zhang. Log-Euclidean kernels for sparse representation and dictionary learning. In *ICCV*, 2013.

[32] Z. Liao, J. Rock, Y. Wang, and D. Forsyth. Non-parametric filtering for geometric detail extraction and material representation. In *CVPR*, 2013.

[33] T. Lin, A. RoyChowdhury, and S. Maji. Bilinear CNN models for fine-grained visual recognition. In *ICCV*, 2015.

[34] M. Lovric, M. Min-Oo, and E. A. Ruh. Multivariate normal distributions parametrized as a Riemannian symmetric space. *JMVA*, 74(1):36–48, 2000.

[35] H. Nakayama, T. Harada, and Y. Kuniyoshi. Global Gaussian approach for scene categorization using information geometry. In *CVPR*, 2010.

[36] M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2011.

[37] F. Perronnin, J. Sánchez, and Y. Liu. Large-scale image categorization with explicit data embedding. In *CVPR*, 2010.

[38] F. Porikli, O. Tuzel, and P. Meer. Covariance tracking using model update based on Lie algebra. In *CVPR*, 2006.

[39] A. Rahimi and B. Recht. Random features for large-scale kernel machines. In *NIPS*, 2007.

[40] J. Sanchez, F. Perronnin, T. Mensink, and J. Verbeek. Image classification with the Fisher vector: Theory and practice. *IJCV*, 105(3):222–245, 2013.

[41] G. Serra, C. Grana, M. Manfredi, and R. Cucchiara. GOLD: Gaussians of local descriptors for image representation. *CVIU*, 134:22–32, 2015.

[42] L. Sharan, C. Liu, R. Rosenholtz, and E. H. Adelson. Recognizing materials using perceptually inspired features. *IJCV*, 103(3):348–371, 2013.

[43] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

[44] O. Tuzel, F. Porikli, and P. Meer. Region covariance: A fast descriptor for detection and classification. In *ECCV*, 2006.

[45] O. Tuzel, F. Porikli, and P. Meer. Pedestrian detection via classification on Riemannian manifolds. *IEEE TPAMI*, 30(10):1713–1727, 2008.

[46] A. Vedaldi and K. Lenc. MatConvNet – convolutional neural networks for MATLAB. In *ACMM*, 2015.

[47] A. Vedaldi and A. Zisserman. Efficient additive kernels via explicit feature maps. *IEEE TPAMI*, 34(3):480–492, 2012.

[48] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical report, 2011.

[49] R. Wang, H. Guo, L. S. Davis, and Q. Dai. Covariance discriminative learning: A natural and efficient approach to image set classification. In *CVPR*, 2012.

[50] C. K. I. Williams and M. Seeger. Using the Nyström method to speed up kernel machines. In *NIPS*, 2001.

[51] J.-H. Won, J. Lim, S.-J. Kim, and B. Rajaratnam. Condition-number-regularized covariance estimation. *J. R. Statist. Soc. B*, 75(3):427–450, 2013.

[52] E. Yang, A. Lozano, and P. Ravikumar. Elementary estimators for sparse covariance matrices and other structured moments. In *ICML*, 2014.

[53] S. K. Zhou and R. Chellappa. From sample similarity to ensemble similarity: Probabilistic distance measures in reproducing kernel Hilbert space. *IEEE TPAMI*, 28(6):917–929, 2006.