Robust Dictionary Learning on the Hilbert Sphere in Kernel Feature Space

Suyash P. Awate(✉) and Nishanth N. Koushik

Computer Science and Engineering Department,
Indian Institute of Technology (IIT) Bombay, Mumbai, India
suyash@cse.iitb.ac.in

Abstract. This paper presents a novel dictionary learning method in kernel feature space that is part of a reproducing kernel Hilbert space (RKHS). Our method focuses on several popular kernels, e.g., radial basis function kernels like the Gaussian, that implicitly map data to a Hilbert sphere, a Riemannian manifold, in RKHS. Our method exploits this manifold structure of the mapped data in RKHS, unlike typical methods for kernel dictionary learning that use linear methods in RKHS. We show that dictionary learning on a Hilbert sphere in RKHS is possible without the need of the explicit lifting map underlying the kernel, but using solely the Gram matrix. Unlike the typical L^1 norm sparsity prior, we incorporate the non-convex L^p quasi-norm based penalty, with $p < 1$, on coefficients to enforce a stronger sparsity prior and achieve more robust dictionary learning in the presence of corrupted training data. We evaluate our method for image classification on two large publicly available datasets and demonstrate the improved performance of our method over the state of the art dictionary learning methods.

1 Introduction

Methods for modeling data as sparse linear combinations of a set of basis elements, often referred to as a dictionary, have found a wide spectrum of applications in machine learning, signal and image processing, and statistical analyses [20,30–32,48,54]. Each element of the dictionary is often referred to as an atom. Dictionary-based modeling leads to optimization problems that can be thought of as posterior mode estimation problems [45], where the dictionary fit relates to the likelihood term and the sparsity-based regularization relates to a prior term on the coefficients in the linear combination. The problem of fitting a given dictionary to the data is a sparse-regression problem for which different sparsity penalties lead to different forms of regression such as the Lasso [54] and subset selection [30], where the efficacy of the Lasso formulation exploits the convexity of the underlying optimization problem [18,31]. The principles of sparse

We thank funding through IIT Bombay Seed Grant 14IRCCSG010.

P. Frasconi et al. (Eds.): ECML PKDD 2016, Part I, LNAI 9851, pp. 731–748, 2016.
DOI: 10.1007/978-3-319-46128-1_46

modeling also play a key role in compressed sensing [20] during reconstruction of signals from corrupted and missing data.

When the data exhibits a nonlinear manifold structure within Euclidean space, the representation of the data using a linear combination of atoms can become inefficient because the linearity in the dictionary representation can fail to adapt to the nonlinearity of the data distribution. Better fits may be obtained by increasing the number of atoms in the dictionary, but doing that increases the complexity of the model and makes the problems of dictionary learning and fitting more difficult, often leading to higher variance [30]. A standard way of adapting to the nonlinearity in the data in input space is to use a nonlinear *kernel* [49,50] to (implicitly) map the data to a high-dimensional kernel feature space, where the nonlinearity in the distribution of the mapped data in kernel feature space is significantly reduced. This mapping to the kernel feature space is typically denoted by $\Phi(\cdot)$. Subsequently, dictionary learning methods can employ standard Euclidean/linear learning in kernel feature space, utilizing the kernel trick to avoid the need to explicitly map the data to the high-dimensional kernel feature space. Our method also exploits kernels and the kernel trick for learning a dictionary model, but, furthermore, exploits the additional spherical structure of the mapped data in the kernel feature space associated with several popular kernels.

In some special cases, the data, in input space, resides on a *Riemannian manifold* that is analytically known, e.g., the space of symmetric positive definite matrices [6,7,14,21,47,51], the Grassmann manifold [15,55], the hypersphere [40,53], or shape space [27,33,36]. In such cases, the dictionary model and the learning [14,29,52,61,65] can exploit the known structure of the manifold to provide atoms that also, desirably, reside on the manifold and efficiently capture the variability in the data. Our method shares the spirit of these approaches in exploiting any known geometrical structure of the data for better modeling and improving performance in practice.

Our method relies on standard kernels that (implicitly) map the data in input space to a known manifold in kernel feature space, specifically, the unit Hilbert sphere in a reproducing kernel Hilbert space (RKHS). Such a mapping occurs for (i) several common kernels [23] including the radial basis function (RBF) kernels like the Gaussian, (ii) kernel normalization, which is common, e.g., in pyramid match kernel [26], and (iii) polynomial and sigmoid kernels when the input points have constant L^2 norm, which is common in certain image analyses [49]. This special structure arises because for these kernels $\kappa(\cdot,\cdot)$, the self similarity of any data point x equals unity, i.e., $\kappa(x,x)=1$. The kernel defines the inner product in the RKHS $\mathcal{H}$, and thus, $\langle \Phi(x), \Phi(x) \rangle_{\mathcal{H}} = 1$, which, in turn, equals the distance of the mapped point $\Phi(x)$ from the origin in $\mathcal{H}$. Thus, all of the mapped points $\Phi(x)$ lie on a Hilbert sphere in RKHS. Figure 1(a) illustrates this behavior. Subsequently, we exploit the Riemannian structure of the Hilbert sphere in RKHS, on which the mapped data resides, to perform dictionary learning in RKHS.

An ideal notion of sparsity relates to the subset-selection problem [30] or, equivalently, constraints or penalties related to the L^0 pseudo-norm [17] of the coefficients for atoms in the dictionary fit. However, because this ideal notion of sparsity leads to combinatorial optimization problems that are NP hard, typical approaches constrain or penalize the L^1 norm of the coefficients. We know that the L^0 pseudo-norm penalty is actually the limiting case of penalizing the p-th power of the L^p quasi-norm as p approaches zero. In fact, there are close theoretical relationships [22,64] between L^0 regularization/combinatorial optimization and L^p-based regularization for sufficiently small $p > 0$. In practice, compared to the squared-L^2-norm penalty, the use of the L^1 norm leads to increased robustness to corruption and outliers in the data [56,58] via a stronger sparsity-promoting penalty. Extending the argument, for our dictionary learning framework, we incorporate the non-convex L^p-to-the-power-p penalties with $p < 1$ on coefficients to enforce a stronger sparsity-promoting penalty and achieve more robust dictionary learning, compared to the L^1-norm penalty.

This paper makes the following contributions. First, we propose a novel dictionary learning framework adapted to the Riemannian manifold of the Hilbert sphere in RKHS. In this way, we combine the advantages of kernel-based frameworks together with the benefits of adapting the analysis to the Riemannian manifold of the (mapped) data. We show that such a learning algorithm does *not* require the knowledge of the explicit mapping function $\Phi(\cdot)$ implied by the kernel $\kappa(\cdot, \cdot)$, but only requires the Gram matrix. Second, our method relies on a stronger notion of sparsity, i.e., the L^p quasi-norm for $p < 1$, which is similar to the desired L^0 sparsity for sufficiently small p, thereby increasing the robustness of the learning algorithm to the noise and outliers in the data. Third, it uses empirical evaluations on two publicly available large real-world datasets to demonstrate the advantages of (i) Riemannian modeling in RKHS as well as (ii) sparsity priors stronger than the typical L^1 norm.

2 Related Work

This section describes the relationships of our method to several related works in the literature, including the state of the art.

Some recent works have presented dictionary learning on the manifolds of symmetric positive definite matrices [14,52,61] and the Grassmann manifold [29,65], which rely on adapting the analyses to the Riemannian metric underlying the manifold. However, the data can have significantly nonlinear distributions even within such manifolds, where typical dictionary models, relying on linear combinations of atoms, can be challenged. Unlike these methods, we rely on (i) kernel-based mapping to reduce the nonlinearity in the data (even when it may lie on a manifold) and (ii) a stronger L^p quasi-norm based sparsity model for robustness to large amounts of corruptions in the training data.

The recent interesting work in [41] exploits kernels for learning dictionaries in RKHS, but does *not* exploit the manifold structure of the mapped data in the RKHS. The method in [41] relies on a greedy approximation algorithm

(orthogonal matching pursuit) to handle the sparsity constraint. Another very recent work [28] adapts dictionary learning for non-Euclidean data by designing the kernel based on the Riemannian metric of the manifold (in input space) on which the data lies. Thus, while [28] exploits the manifold structure in the input space, it ignores the manifold structure in the RKHS. Our framework allows exploitation of the manifold structure of the data in both the input space (via kernel design) and the RKHS (via spherical statistics). We also demonstrate robustness to large levels of noise and outliers.

Several previous works [1,9,16,19,25,49] perform statistical analyses on the Hilbert sphere in RKHS, but are outside the realm of dictionary learning and sparse modeling. Nevertheless, the spirit of these works, in adapting the analyses to the Hilbert sphere in RKHS to improve performance, is akin to the spirit of our approach. This prior knowledge of the mapped data (and atoms) lying on a sphere leads to added regularization that can prevent overfitting in high-dimension low-sample-size scenarios where Euclidean analysis is known to be unstable and error prone as it interacts with the curvature of the sphere on which the data resides [2]. For directional data, although distributions modeling the covariance structure exist in the literature, the underlying parameter estimation is intractable in high-dimensional spaces [10,40,46]; furthermore, these distributions don't exploit sparsity-based regularization. Indeed Gaussian and Factor models have exploited sparsity as a regularizer [37,59].

Several applications require learning that is robust to large levels of corruption in the (training) data. Some approaches [39,44] achieve robustness to non-Gaussianity of the data distributions, resulting from non-Gaussianity of the noise/likelihood model, by replacing the squared-L^2-norm penalty on the residual arising from the data fitting term by the L^1-norm penalty. Other approaches [34] further propose limiting such a L^1-norm penalty by capping the penalty at a fixed level. Alternate strategies [13] continue to penalize the squared L^2 norm of the residual, but modify the residual to explicitly include outlier variables such that residual distribution remains close to a Gaussian. Instead of modifying the data-fit (likelihood) term, robustness can result from using a stronger sparsity prior that prevents the overfitting of the dictionary to the (highly) corrupted data. In this spirit, some approaches achieve robustness by changing the squared-L^2-norm penalty on the coefficients to the L^1-norm penalty on the coefficients [56,58]; this is akin to replacing ridge regression by Lasso. Other approaches [42] use the L^1-norm penalty for the data fit along with a mixed $l_{2,1}$-norm penalty on the sparse matrix of coefficients to obtain robustness. In contrast to these approaches, our method employs L^p quasi-norm penalties for $p \in (0, 1)$ for the sparsity prior to demonstrate improved robustness to corruption in data in the form of noise and outliers.

Alternate approaches to sparsity include the local coordinate coding or, rather, locally constrained sparse coding [62,63] that relies on the empirical observation that the sparse code for a datum is likely to exhibit non-zero coefficients for atoms that are in the locality of the datum. Subsequently, [62,63] redesign the standard L^1 penalty on coefficients as a weighted L^1 penalty, where,

for each datum, the weight increases the penalty for atoms far from that datum. A fast approximate extension of local coordinate coding for image classification appears in [57] that replaces the dictionary with a local dictionary (nearest few atoms) for coding each datum; such schemes have also been used in RKHS [29]. Complementary to these approaches, our approach also adds to the conventional notion of L^1 sparsity, but it does so by using the L^p quasi-norm and tuning the parameter p to get solutions closes to the ones found with the L^0 pseudo-norm.

3 Riemannian Geometry of the Hilbert Sphere in RKHS

Many popular kernels are associated with an infinite-dimensional RKHS. So, the analysis in this paper focuses on such spaces. Nevertheless, analogous theory holds for other important kernels (e.g., normalized polynomial) where the RKHS is finite dimensional.

Let X be a random variable taking values x in *input space* $\mathcal{X}$. Let $\{x_n\}_{n=1}^N$ be a set of observations in input space. Let $\kappa(\cdot,\cdot)$ be a real-valued Mercer kernel with an associated map $\Phi(\cdot)$ that maps x to $\Phi(x) := \kappa(\cdot, x)$ in a RKHS $\mathcal{H}$ [5,49]. Consider two points in RKHS, within the span of the mapped data, represented as $f := \sum_{i=1}^I \alpha_i \Phi(x_i)$ and $f' := \sum_{j=1}^J \beta_j \Phi(x_j)$ where the weights $\alpha_i \in \mathbb{R}$ and $\beta_j \in \mathbb{R}$. The inner product $\langle f, f' \rangle_{\mathcal{H}} := \sum_{i=1}^I \sum_{j=1}^J \alpha_i \beta_j \kappa(x_i, x_j)$. The norm $\|f\|_{\mathcal{H}} := \sqrt{\langle f, f \rangle_{\mathcal{H}}}$.

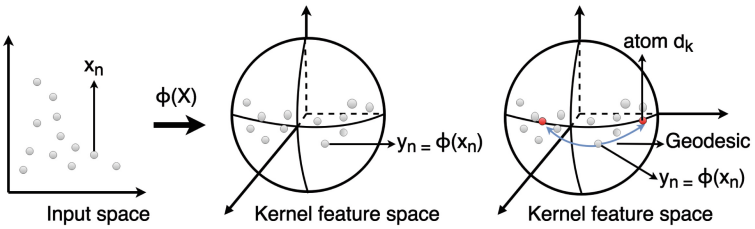
Let $Y := \Phi(X)$ be the random variable taking values y in RKHS. Previous methods for kernel-based dictionary learning [28,41] model each y as a sparse linear combination of atoms in RKHS. The analysis in this paper applies to kernels that map points in input space to a Hilbert sphere in RKHS, i.e., $\forall x : \kappa(x, x) = \theta$, a constant (without loss of generality, we assume $\theta = 1$). Methods for non-Euclidean dictionary learning in input space [29,52,61,65] exploit the Riemannian structure of the manifold on which the data lie to propose dictionary learning on the manifold. The proposed dictionary-learning method exploits the property of y lying on the Riemannian manifold of the unit Hilbert sphere [4,11] in RKHS. Thus, we now introduce differential geometric constructs specific to this Hilbert sphere in RKHS [9].

Consider a and b on the unit Hilbert sphere in RKHS represented, in general, as $a := \sum_n \gamma_n \Phi(x_n)$ and $b := \sum_n \delta_n \Phi(x_n)$. The logarithmic map, or Log map, of a with respect to b is the vector

$$\mathrm{Log}_b(a) = \frac{a - \langle a, b \rangle_{\mathcal{H}} b}{\|a - \langle a, b \rangle_{\mathcal{H}} b\|_{\mathcal{H}}} \arccos(\langle a, b \rangle_{\mathcal{H}}) \tag{1}$$

$$= \sum_n \zeta_n \Phi(x_n), \text{ where } \forall n : \zeta_n \in \mathbb{R}. \tag{2}$$

Clearly, $\mathrm{Log}_b(a)$ can always be written as a linear combination of points $\{\Phi(x_n)\}_{n=1}^N$. The tangent vector $\mathrm{Log}_b(a)$ lies in the tangent space, at b, of the unit Hilbert sphere. The tangent space to the Hilbert sphere in RKHS inherits the



(a) Points implicitly mapped to Hilbert sphere (b) Dictionary modeling

Fig. 1. Dictionary Modeling on a Hilbert Sphere in RKHS. (a) Points x_n in input space get mapped *implicitly*, via several popular Mercer kernels, to $y_n := \Phi(x_n)$ on a Hilbert sphere in RKHS. **(b)** Dictionary atoms d_k, on the Hilbert sphere in RKHS, being used to fit to a point y_n.

same structure (inner product) as the ambient space and, thus, is also a RKHS. The geodesic distance between a and b is $d_g(a,b) = \|\text{Log}_b(a)\|_{\mathcal{H}} = \|\text{Log}_a(b)\|_{\mathcal{H}}$.

Now, consider a tangent vector $t := \sum_n \beta_n \Phi(x_n)$ lying in the tangent space at b. The exponential map, or Exp map, of t with respect to b is

$$\text{Exp}_b(t) = \cos(\|t\|_{\mathcal{H}})b + \sin(\|t\|_{\mathcal{H}})\frac{t}{\|t\|_{\mathcal{H}}} \tag{3}$$

$$= \sum_n \omega_n \Phi(x_n), \text{ where } \forall n : \omega_n \in \mathbb{R}. \tag{4}$$

Clearly, $\text{Exp}_b(t)$ can always be written as a linear combination of points $\{\Phi(x_n)\}_{n=1}^N$. $\text{Exp}_b(t)$ maps a tangent vector t to the unit Hilbert sphere, i.e., $\|\text{Exp}_b(t)\|_{\mathcal{H}} = 1$.

4 Robust Dictionary Learning on a Hilbert Sphere in RKHS

Motivated by principles underlying sparse modeling, we model each $y = \Phi(x)$ in RKHS, where $\|y\|_{\mathcal{H}} = 1$, to have been generated as a sparse, but nonlinear, combination of atoms that also lie on the unit Hilbert sphere in RKHS. We consider the dictionary to have K atoms $\{d_k \in \mathcal{H}\}_{k=1}^K$, each with $\|d_k\|_{\mathcal{H}} = 1$. Given data $\{x_n\}_{n=1}^N$ and the number of atoms K, we learn the atoms $\{d_k\}_{k=1}^K$. Figure 1(b) gives an illustration.

In Euclidean spaces, dictionary learning typically penalizes the squared norm of the residual between the datum and its representation as a sparse linear combination of atoms. The natural generalization of this residual to a Riemannian manifold is the logarithmic map. The notion of sparsity in Euclidean spaces is generalized to the notion of affine sparsity in non-Euclidean spaces [61–63], which makes the dictionary representation independent of a coordinate frame shift or a notion of origin in the Riemannian space. Affine sparsity constraints the sum of weights, in the dictionary fit, to 1 and is important in the nonlinear

sparse-coding setting because Riemannian analyses treats y as a point, unlike linear sparse coding in Euclidean space that treats y as a vector. Another motivation is as follows. Consider K atoms on the Hilbert sphere in RKHS in general position, i.e., these points are *not* contained in any great $(K-2)$-sphere [43]. Given K points in general position on a Hilbert sphere, there is a unique great $(K-1)$-sphere containing them [43]. Given K atoms in general position and a sparsity level $M \leq K$, principles underlying sparse modeling motivate us to model each y as lying within the unique great $(M-1)$-sphere containing some M atoms. Thus, ideally we would desire to fit y by, say, $\widehat{y}$, where $\widehat{y}$ lies within such a great $(M-1)$-sphere. When $\widehat{y}$ lies within this great $(M-1)$-sphere, there exists a set of weights $w_m \in \mathbb{R}$ where

$$\sum_{m=1}^{M} w_m = 1 \text{ and } \sum_{m=1}^{M} w_m \text{Log}_{\widehat{y}}(d_m) = 0, \tag{5}$$

i.e., in the tangent space at $\widehat{y}$, there exist weights w_m that satisfy the affine constraint and make the weighted combination of vectors $\text{Log}_{\widehat{y}}(d_m)$ coincide with the origin. Thus, our fitting problem for y, given the dictionary, reduces to finding weights w_m that, under the sparsity prior and the affine constraints, minimize

$$\| \sum_{m=1}^{M} w_m \text{Log}_y(d_m) \|_{\mathcal{H}}. \tag{6}$$

This fitting term is in the same spirit as the fitting terms in [24,61].

However, because the ideal sparsity prior on weights (L^0 pseudo-norm [17] equivalent to subset selection [30]) leads to a combinatorial optimization problem that is NP hard, the sparsity prior is typically relaxed by (i) using all K atoms, instead of some subset of size $M < K$, for the fitting and (ii) penalizing the p-th power of the L^p norm of the weight vector w associated with the fit, where $p \in \mathbb{R}_{>0}$ is a free parameter. In the Euclidean case, the typical relaxed penalty is the L^1 norm of the weight vector, i.e., $p = 1$, that is motivated by the desire to retain the convexity of the optimization problem when the dictionary-fitting term is the squared residual between the datum and the linear combination of atoms. As with other manifolds, in the case of the Hilbert sphere, the dictionary-fitting term is itself nonconvex. Furthermore, choosing $p \in (0, 1)$ leads to the penalty $\|w\|_p^p = \sum_m |w_m|^p$ that tends to the L^0 pseudo-norm as $p \to 0$ and improves the robustness of the dictionary learning to outliers.

We now formulate the robust dictionary learning problem. Let $\{x_n\}_{n=1}^N$ be the data points in input space. Let $y_n := \Phi(x_n)$ be the (implicitly) mapped points in RKHS where the mapping $\Phi(\cdot)$ ensures that $\|y_n\|_{\mathcal{H}} = 1$. We do *not* need the mapping $\Phi(\cdot)$ explicitly, but use the kernel trick for all the analysis in RKHS. Let the dictionary D comprise K atoms d_k, as described before. Let W be the weight matrix comprising columns w_n that comprise the weights w_{nk} for the contribution of the k-th atom in the fit for the n-th point y_n. Then, we propose to formulate robust dictionary learning as

$$\arg\min_{D}\left[\min_{W}\sum_{n=1}^{N}\left(\|\sum_{k=1}^{K}w_{nk}\mathrm{Log}_{y_n}(d_k)\|_{\mathcal{H}}^2+\lambda\|w_n\|_{p,\epsilon}^p\right)\right]$$

$$\text{under the constraints: } \forall n,\sum_{k=1}^{K}w_{nk}=1 \text{ and } \forall k,\|d_k\|_{\mathcal{H}}=1, \tag{7}$$

where $\lambda > 0$ is the regularization parameter balancing the data-fitting/fidelity term and the sparsity prior term, $p \in (0,1)$ is a free parameter, and $\|w_n\|_{p,\epsilon}^p$ is the p-th power of the ϵ-regularized L^p quasi-norm

$$\|w_n\|_{p,\epsilon}^p := \sum_{k=1}^{K}(|w_{nk}|^2+\epsilon)^{p/2} \tag{8}$$

that is smooth and amenable to gradient-based optimization. The affine constraint is useful here, without which the formulation leads to the trivial solution: $w_{nk} = 0, \forall n, k$.

Because the dictionary fit attempts to minimize the norm of a linear combination of tangent vectors $\mathrm{Log}_{y_n}(d_k)$, it is natural to have atoms d_k within a subsphere $\mathcal{S}$ that corresponds to the intersection of (i) the linear subspace in RKHS representing the span of the set of points y_n with (ii) the unit Hilbert sphere in RKHS. For instance, if all but one atom d_l lies outside the span of the data, then either (i) that atom remains unused, i.e., all weights w_{nl} for atom d_l are zero, or (ii) the use of atom d_l leads to an increase in the norm of the linear combination, as compared to using an atom that is the projection of d_l on the subsphere $\mathcal{S}$, because it leads to an additional vector component in the linear combination along a direction orthogonal to the other log-mapped atoms $\mathrm{Log}_{y_n}(d_l)$ in the tangent space at y_n. Thus, if d_l is to be used, replacing d_l with its projection on the subsphere $\mathcal{S}$ will reduce the objective function and be a better solution. Take another instance where multiple atoms d_k lie outside the span of the data, or equivalently outside the lowest-dimensional subsphere $\mathcal{S}$ that contains all the mapped points y_n. In this case, if there is a sparse combination of atoms d_k that exactly fits the data, i.e., $\|\sum_{k=1}^{K} w_{nk}\mathrm{Log}_{y_n}(d_k)\|_{\mathcal{H}} = 0$, then we can use the same weights w_{nk} and use the projections of atoms d_k on the subsphere $\mathcal{S}$ to again get the exact fit. Thus, it is unnecessary to use atoms d_k outside the subsphere $\mathcal{S}$. So, we represent each atom as

$$d_k := \sum_{n=1}^{N}\gamma_{kn}\Phi(x_n), \text{ where } \forall n : \gamma_{kn} \in \mathbb{R} \text{ and } \|d_k\|_{\mathcal{H}} = 1. \tag{9}$$

This representation for atoms ensures that each logarithmic map $\mathrm{Log}_{y_n}(d_k)$ can be represented as a linear combination of the $\Phi(y_n)$.

Within the first term in the objective function, $\forall y_n$, the norm $\|\sum_{k=1}^{K} w_{nk}\mathrm{Log}_{y_n}(d_k)\|_{\mathcal{H}}^2$ can be represented purely in terms of the Gram matrix and the coefficients γ_{kn} underlying the atoms' representation, as follows. Each tangent vector $\mathrm{Log}_{y_n}(d_k)$ can be represented as a weighted combination of $\Phi(x_n)$.

Thus, the linear combination $\sum_{k=1}^{K} w_{nk}\text{Log}_{y_n}(d_k)$ is also weighted combination of $\Phi(x_n)$. Then, the norm

$$\|\sum_{k=1}^{K} w_{nk}\text{Log}_{y_n}(d_k)\|_{\mathcal{H}}^2 = \|\sum_{p=1}^{N} \xi_p \Phi(x_p)\|_{\mathcal{H}}^2, \text{ for some weights } \xi_p \in \mathbb{R} \tag{10}$$

$$= \sum_{n'=1}^{N}\sum_{n''=1}^{N} \xi_{n'}\xi_{n''}\langle\Phi(x_{n'}), \Phi(x_{n''})\rangle_{\mathcal{H}} = \sum_{n'=1}^{N}\sum_{n''=1}^{N} \xi_{n'}\xi_{n''}G_{n'n''}, \tag{11}$$

where G is the Gram matrix such that $G_{n'n''} := \langle\Phi(x_{n'}), \Phi(x_{n''})\rangle_{\mathcal{H}}$ with all diagonal elements $G_{nn} := \langle\Phi(x_n), \Phi(x_n)\rangle_{\mathcal{H}} = 1$. The scalar weights ξ_p are

$$\xi_p = \sum_{k=1}^{K} w_{nk} \frac{\arccos(\langle d_k, y_n\rangle_{\mathcal{H}})}{\|d_k - \langle d_k, y_n\rangle_{\mathcal{H}} y_n\|_{\mathcal{H}}} \left(\gamma_{kp} - \mathbf{I}_{n,p}\langle d_k, y_n\rangle_{\mathcal{H}}\right), \tag{12}$$

where the indicator function $\mathbf{I}_{n,p} = 1$ if $n = p$ and $\mathbf{I}_{n,p} = 0$ otherwise.

The constraint on the atoms, $\|d_k\|_{\mathcal{H}} := \|\sum_{n=1}^{N} \gamma_{kn}\Phi(x_n)\|_{\mathcal{H}} = 1$, can also be specified in terms of the Gram matrix G as

$$\sum_{n'=1}^{N}\sum_{n''=1}^{N} \gamma_{kn'}\gamma_{kn''}G_{n'n''} = 1. \tag{13}$$

This confirms that the optimization problem, both the objective function and the constraints, can be specified purely in terms of the variables $\{\{w_{nk} \in \mathbb{R}\}_{k=1}^{K}\}_{n=1}^{N}$ and $\{\{\gamma_{kn} \in \mathbb{R}\}_{n=1}^{N}\}_{k=1}^{K}$, given the Gram matrix G. Hence, we do *not* need the explicit mapping $\Phi(\cdot)$ for our dictionary learning method.

We optimize the atoms d_k, represented through the parameters $\{\gamma_{kn} \in \mathbb{R}\}$, and weights $w_n \in \mathbb{R}^K$ alternatingly using projected gradient descent that guarantees convergence to a stationary point. For each atom d_k, the projection is on the unit Hilbert sphere, which involves a simple rescaling of the parameters γ_{kn}. For each weight vector w_n, the projection is on the hyperplane through the origin $\sum_{k=1}^{K} w_{nk} = 1$, which is also straightforward. We adjust the step sizes, using a line search, to ensure that the objective function decreases. To alleviate the difficulty of the non-convexity resulting from the ϵ-regularized L^p quasi-norm, we use continuation-based optimization [3,60] (a general strategy for optimizing non-convex functions; similar to annealing) that starts with a relatively large value of $\epsilon = 1$ and gradually reduces ϵ to 0.001. In practice, such continuation strategies help find better local minima for non-convex objective functions.

Given the number of atoms K, we initialize atoms d_k using kernel k-means using geodesic distances on the Hilbert sphere in RKHS. We initialize the k-means using the k-means++ algorithm [8]. After initializing the atoms d_k, we initialize the weight w_{nk} in inverse proportion to the distance of y_n from d_k $\text{Log}_{y_n}(d_k)$, normalized such that the weights of any input sum to 1 as required by the affine constraint, i.e.,

$$w_{nk}^{\text{init}} := \frac{\|\text{Log}_{y_n}(d_k)\|_{\mathcal{H}}^{-1}}{\sum_{k'=1}^{K}\|\text{Log}_{y_n}(d_{k'})\|_{\mathcal{H}}^{-1}}. \tag{14}$$

For each y_n, this gives larger weights w_{nk} for those atoms d_k that are closer to y_n.

We summarize the proposed *algorithm for dictionary learning* below:

1. **Inputs:** A set of points $\{x_n\}_{n=1}^N$ in input space. Gram matrix G underlying a kernel such that all diagonal elements $G_{nn} = 1$. Number of atoms K. Parameters $\lambda > 0$ and $p \in (0, 1)$. Set iteration number $i = 0$.
2. Initialize the dictionary D^i comprising atoms $\{d_k^i\}_{k=1}^N$ using kernel k-means adapted to the Hilbert sphere in RKHS, as described previously. Each atom d_k is represented using parameters $\{\gamma_{kn}^i\}$, as described in Eq. 9.
3. Initialize the weights matrix W^i, as described before.
4. Fix $\epsilon = 1$.
5. Fixing the weights W^i, use projected gradient descent to optimize for $\{\gamma_{kn}\}$ based on the formulation in (7) to get the updated parameters $\{\gamma_{kn}^{i+1}\}$ used to represent atoms $\{d_k^{i+1}\}_{k=1}^N$ in the dictionary D^{i+1}.
6. Fixing the parameters $\{\gamma_{kn}^{i+1}\}$, use projected gradient descent to optimize for W based on the formulation in 7 to get the updated weight matrix W^{i+1}.
7. If the relative change in the values of the objective function, in 7, evaluated at (W^i, D^i) and (W^{i+1}, D^{i+1}), is less than a small threshold, then terminate, otherwise increment i by 1 and repeat the last 2 steps.
8. Reduce $\epsilon \leftarrow \epsilon/10$. If $\epsilon < 0.001$, then terminate, otherwise repeat the last 3 steps (projected gradient descent optimization) with the initial solution as that obtained for the previous value of ϵ.
9. **Outputs:** A set of parameters $\{\gamma_{kn}^*\}$ representing the optimal atoms d_k^* in the optimal dictionary D^* and an optimal weight matrix W^* representing the coefficients w_n, for all atoms, in the fit to each y_n.

5 Application: Image Classification

We apply the proposed dictionary learning framework for image classification. The framework is general and considers image feature vectors x_n extracted from each image as the training data. We assume a kernel, e.g., Gaussian, such that the self similarity $\kappa(x, x) = 1$ for all feature vectors x. We first learn a dictionary from training data and then use it to code each training datum, where each datum's code is the vector of coefficients obtained from the dictionary fit. We then train a linear support vector machine (SVM) to learn a classifier on these codes.

We summarize the proposed *algorithm for learning a dictionary-based classifier* for the purpose of image classification.

1. **Inputs:** For the Q classes (denoted by $q = 1, 2, \cdots, Q$), N_q feature vectors $\{x_{qn}\}_{n=1}^{N_q}$ for class q. Gram matrix G, for the pooled dataset, underlying a kernel such that all diagonal elements equal 1. The number of atoms K_q in the dictionary D_q for each class q. Parameters $\lambda > 0$ and $p \in (0, 1)$.
2. For all Q classes, use the dictionary learning algorithm in Sect. 4 to learn dictionary D_q of K_q atoms each for class q.

3. Pool all the dictionaries $\{D_q\}_{q=1}^{Q}$ to create a dictionary D having $K := \sum_{q=1}^{Q} K_q$ atoms.
4. Use the pooled dictionary D to fit to the mapped data $\{y_{qn} := \Phi(x_{qn})\}_{n=1}^{N_q}$ for all classes q, using the algorithm in Sect. 4, but keeping the dictionary fixed to D. This gives weight matrices W_q for all feature vectors in each class q, where each column w_{qn} has length K.
5. Learn a classifier $\mathcal{C}$ based on feature vectors $\{w_{qn} \in \mathbb{R}^K\}_{n=1}^{N_q}$ for each class q, using a linear SVM to classify any w into one of the Q classes. We do so by training Q one-versus-all SVM classifiers [12].
6. **Outputs:** Pooled dictionary D and the classifier $\mathcal{C}$.

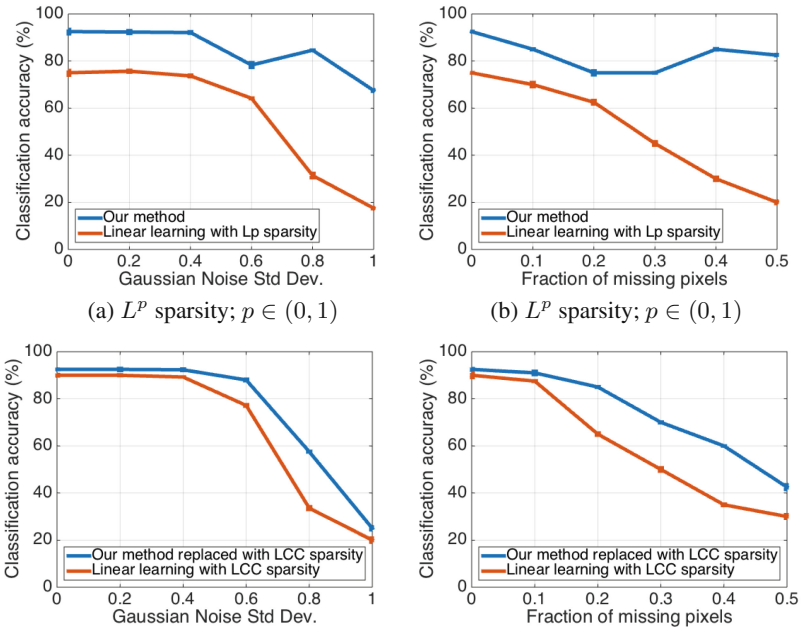
We summarize the proposed *algorithm for dictionary-based image classification.*

1. **Inputs:** Pooled dictionary D and the Gram matrix G for which it is learned. The classifier $\mathcal{C}$. Test image x to be classified along with the extension of the Gram matrix (one row/column) for this test image's feature vector x, giving kernel similarity of the test image's feature vector x with all training image feature vectors.
2. Use the pooled dictionary D to fit to the mapped datum $y := \Phi(x)$ using the algorithm in Sect. 4, but keeping the dictionary fixed to D. This gives weight vector w of length K.
3. Use classifier $\mathcal{C}$ to classify the weight vector w into one of the Q classes, say q'.
4. **Output:** Class q'.

6 Results and Discussion

This section shows the results of empirical evaluation of our dictionary-learning based method, compared with the state of the art, on two large publicly available real-world image datasets of handwritten digits: (i) the MNIST dataset [38] available at http://yann.lecun.com/exdb/mnist/ and (ii) the USPS dataset [30] available at www-stat.stanford.edu/~tibs/ElemStatLearn/data.html. For evaluation on each dataset, we consider each raw image, after vectorization, as an input feature vector x_n. We use the popular Gaussian kernel $\kappa(x_i, x_j) := \exp(-0.5\|x_i - x_j\|_2^2/\sigma^2)$ and set σ^2, as per convention, to the average squared Euclidean distance between all pairs (x_i, x_j). We measure performance by the classification accuracy, i.e., the percentage of correctly classified images out of the total number of image classified.

We compare our method against state-of-the-art approaches involving kernel-based dictionary learning (all methods use the same dictionary size) and varying sparsity priors. First, we compare our Riemannian dictionary learning on the unit Hilbert sphere in RKHS with the alternate strategy that assumes the mapped data to lie in a linear space and performs standard linear dictionary learning in RKHS, e.g., in [28,41]. This compares two different data-fitting strategies in kernel feature space, i.e., our method using Hilbert-sphere modeling in RKHS and



(a) L^p sparsity; $p \in (0, 1)$ (b) L^p sparsity; $p \in (0, 1)$

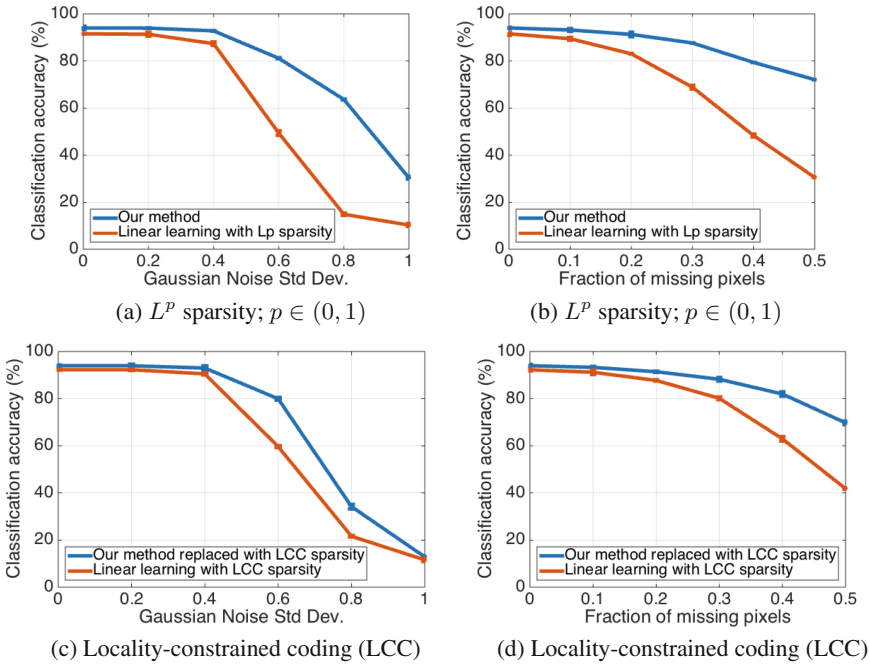
(c) Locality-constrained coding (LCC) (d) Locality-constrained coding (LCC)

Fig. 2. USPS Handwritten Digit Image Recognition. Recognition rates (averages and standard deviations obtained by bootstrap sampling of the training dataset) for images with varying levels of corruptions (i.i.d. additive zero-mean Gaussian noise or missing pixels), when the sparsity prior is the **(a)-(b)** L^p quasi norm, with optimally tuned value of $p \leq 1$ for each noise level, and **(c)-(d)** locality-constrained coding, with optimally tuned value of $\rho > 0$ for each noise level.

the state of the art involving Euclidean modeling in RKHS. Second, we evaluate both aforementioned methods for two different kinds of sparsity priors, i.e., our L^p quasi-norm for $p < 1$ and the alternate strategy based on locality constrained coding, e.g., in [29,57,62,63]. However, unlike the faster, but approximate, versions [29,57] that use the few nearest atoms for coding each datum, we use the version with the weighted L^1 penalty over all atoms, i.e.,

$$\lambda \sum_{n=1}^{N} \sum_{k=1}^{K} \Big(w_{nk} \exp(\rho \|y_n - d_k\|) \Big)^2 \tag{15}$$

where $\rho \in \mathbb{R}_{>0}$ is a free parameter. Unlike the heuristic of choosing the nearest atoms for coding, this penalty lends itself to optimization via gradient descent that guarantees the reduction in the objective function at each step. When we use Hilbert-sphere modeling in RKHS, we choose the distance $\|y_n - d_k\|$ as the geodesic distance $\|\mathrm{Log}_{y_n}(d_k)\|_{\mathcal{H}}$; otherwise, we choose the norm $\|y_n - d_k\|_{\mathcal{H}}$ in RKHS.



(a) L^p sparsity; $p \in (0, 1)$ (b) L^p sparsity; $p \in (0, 1)$

(c) Locality-constrained coding (LCC) (d) Locality-constrained coding (LCC)

Fig. 3. MNIST Handwritten Digit Image Recognition. Recognition rates (averages and standard deviations obtained by bootstrap sampling of the training dataset) for images with varying levels of corruptions (i.i.d. additive zero-mean Gaussian noise or missing pixels), when the sparsity prior is the **(a)-(b)** L^p quasi norm, with optimally tuned value of $p \leq 1$ for each noise level, and **(c)-(d)** locality-constrained coding, with optimally tuned value of $\rho > 0$ for each noise level.

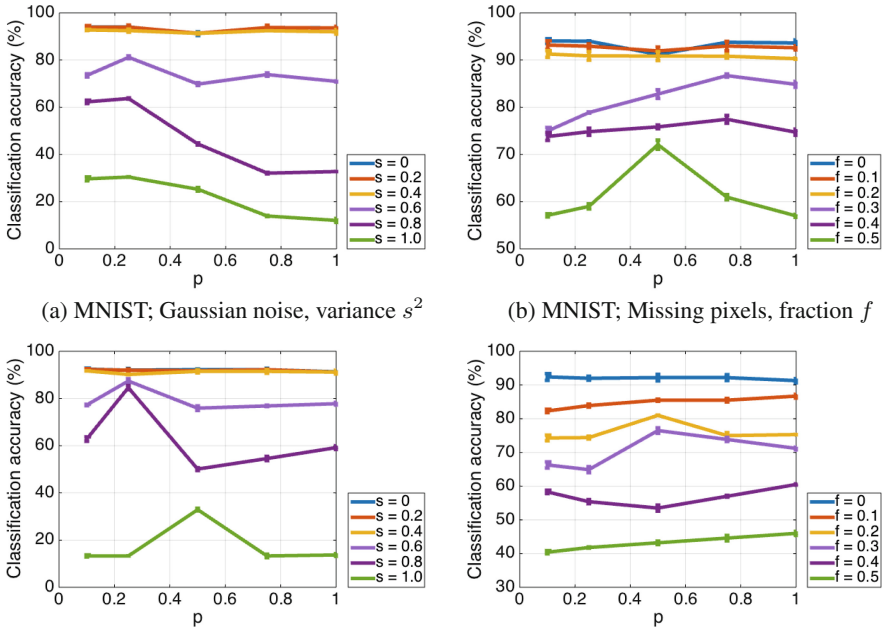
We evaluate the robustness of the dictionary learning methods under two kinds of corruptions in the *training* data: (i) independent and identically distributed (i.i.d.) additive zero-mean Gaussian noise and (ii) missing data at randomly chosen pixels in the image. In case of images with missing intensities at specific pixels, we replace the missing intensity values with the average intensity over the pixels where the intensities are observed. We found this strategy of filling in missing information to be easy, parameter free, and leading to stable results. In the real-world, missing-pixel scenarios arise naturally in solving recognition problems for partially visible/occluded objects [35]. For both these kinds of corruptions, we evaluate performance over a range of corruption levels. At each corruption level, we tune the parameters underlying each method, i.e., the regularization parameter λ and the parameter underlying the sparsity prior (i.e., either p or ρ) using 5-fold cross validation [12] on the chosen training data.

To measure the variability of the classification performance with respect to the variability in the training data sample, we use bootstrap sampling to randomly select 90 % of the available training set to learn the classifier and, then,

use the learned classifier to classify the test set. Repeated bootstrap sampling, learning, and classification gives us the reliability of the performance of the methods. For each corruption level, we perform bootstrap 25 times to get 25 different training data samples.

The results on the USPS dataset (in Fig. 2) and the MNIST dataset (in Fig. 3) show that the proposed Riemannian modeling on the Hilbert-sphere in RKHS clearly achieves superior classification accuracy over linear modeling in RKHS for (i) both sparsity priors and (ii) both kinds of corruptions. Compared to linear dictionary modeling in RKHS, the gains from our Hilbert-spherical modeling in RKHS are often more than 10 % and are almost always statistically significantly better, as determined by a two-sample Student's t-test (p-value < 0.01).

The results on the MNIST and USPS datasets in Fig. 4 show that for higher corruption levels, of both noise and missing-pixel types, the best performance is typically obtained when the values of the parameter p in our L^p quasi-norm spar-



(a) MNIST; Gaussian noise, variance s^2

(b) MNIST; Missing pixels, fraction f

(c) USPS; Gaussian noise, variance s^2

(d) USPS; Missing pixels, fraction f

Fig. 4. Robustness of Dictionary Learning to Corrupted Training Data. Recognition rates (averages and standard deviations obtained by bootstrap sampling of the training dataset) for images with varying levels of corruptions (i.i.d. additive zero-mean Gaussian noise or missing pixels), with various values of the parameter $p < 1$ in the L^p quasi-norm sparsity penalty for **(a)** MNIST dataset with i.i.d. additive zero-mean Gaussian noise, **(b)** MNIST dataset with a fraction of pixels with missing intensity values, **(c)** USPS dataset with i.i.d. additive zero-mean Gaussian noise, and **(d)** USPS dataset with a fraction of pixels with missing intensity values.

sity penalty is less than 1. For very small values of p, the performance expectedly degrades possibly because of the increasing non-convexity of the penalty and the greater tendency to get stuck in a local minimum. This can also happen in some rare cases, where better continuation strategies can help. Nevertheless, for large corruption levels, the optimal performance almost always occurs for values of p significantly less than 1; for these datasets, the optimal p is typically in the range 0.2 and 0.8. This confirms the benefits of our proposed L^p quasi-norm sparsity penalty for tuned values of $p < 1$.

7 Conclusion

This paper presents a new method for kernel-based dictionary learning that addresses the hyperspherical geometry of the (implicitly) mapped points in RKHS, which naturally arises from many popular kernels and kernel normalization. In the same spirit that motivates manifold-based dictionary learning in input space, we perform manifold based dictionary learning in RKHS. We also propose stronger sparsity priors in the form of the non-convex L^p quasi-norm penalties that we deal with practically using a continuation-based optimization algorithm. We utilize the new dictionary learning algorithm for recognizing handwritten digits on large standard datasets and clearly demonstrate improved performances resulting from the modeling the Hilbert-sphere geometry in RKHS. We also demonstrate the gain in the robustness of the dictionary learning, to corruptions in the training data, arising from the stronger sparsity constraint.

References

1. Ah-Pine, J.: Normalized kernels as similarity indices. In: Proceeding of Pacific-Asia Conference Advances in Knowledge Discovery and Data Mining, vol. 2, pp. 362–373 (2010)
2. Ahn, J., Marron, J.S., Muller, K., Chi, Y.Y.: The high-dimension, low-sample-size geometric representation holds under mild conditions. Biometrika **94**(3), 760–766 (2007)
3. Allgower, E., Georg, K.: Introduction to Numerical Continuation Methods. SIAM (2003)
4. Amari, S., Nagaoka, H.: Methods of Information Geometry. Oxford Univ. Press, New York (2000)
5. Aronszajn, N.: Theory of reproducing kernels. Trans. Amer. Math. Soc. **68**(3), 337–404 (1950)
6. Arsigny, V., Fillard, P., Pennec, X., Ayache, N.: Log-Euclidean metrics for fast and simple calculus on diffusion tensors. Mgn. Reson. Med. **56**(2), 411–421 (2006)
7. Arsigny, V., Fillard, P., Pennec, X., Ayache, N.: Geometric means in a novel vector space structure on symmetric positive-definite matrices. SIAM J. Mat. Anal. Appl. **29**(1), 328–347 (2007)
8. Arthur, D., Vassilvitskii, S.: k-means++: the advantages of careful seeding. In: Proceeding Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 1027–1035 (2007)

9. Awate, S.P., Yu, Y.-Y., Whitaker, R.T.: Kernel principal geodesic analysis. In: Calders, T., Esposito, F., Hüllermeier, E., Meo, R. (eds.) ECML PKDD 2014, Part I. LNCS, vol. 8724, pp. 82–98. Springer, Heidelberg (2014)
10. Banerjee, A., Dhillon, I., Ghosh, J., Sra, S.: Clustering on the unit hypersphere using von Mises-Fisher distributions. J. Mach. Learn. Res. **6**, 1345–1382 (2005)
11. Berger, M.: A Panoramic View of Riemannian Geometry. Springer, Heidelberg (2007)
12. Bishop, C.: Pattern Recognition and Machine Learning. Springer, New York (2006)
13. Chen, Z., Wu, Y.: Robust dictionary learning by error source decomposition. In: Proceeding International Conference on Computer Vision, pp. 2216–2223 (2013)
14. Cherian, A., Sra, S.: Riemannian sparse coding for positive definite matrices. In: Proceeding European Conference on Computer Vision, pp. 299–314 (2014)
15. Common, P., Golub, G.: Tracking a few extreme singular values and vectors in signal processing. Proc. IEEE **78**(8), 1327–1343 (1990)
16. Courty, N., Burger, T., Marteau, P.: Geodesic analysis on the Gaussian RKHS hypersphere. In: European conference Machine Learning Practice of Knowledge Discovery Data, vol. 1, 299–313 (2012)
17. Donoho, D., Elad, M.: Optimally sparse representation in general (nonorthogonal) dictionaries via $l1$ minimization. Proc. Nat. Acad. Sci. **100**(5), 2197–2202 (2003)
18. Efron, B., Hastie, T., Johnstone, I., Tibshirani, R.: Least angle regression. Ann. Stat. **32**(2), 407–451 (2004)
19. Eigensatz, M.: Insights into the geometry of the Gaussian kernel and an application in geometric modeling. Master thesis. Swiss Federal Institute of Technology (2006)
20. Elad, M.: Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing. Springer, New York (2010)
21. Fletcher, P.T., Joshi, S.: Riemannian geometry for the statistical analysis of diffusion tensor data. Signal Process. **87**(2), 250–262 (2007)
22. Fung, G., Mangasarian, O.: Equivalence of minimal l_0- and l_p-norm solutions of linear equalities, inequalities and linear programs for sufficiently small p. J. Optim. Theory Appl. **151**(1), 1–10 (2011)
23. Genton, M.: Classes of kernels for machine learning: a statistics perspective. J. Mach. Learn. Res. **2**, 299–312 (2001)
24. Goh, A., Vidal, R.: Clustering and dimensionality reduction on Riemannian manifolds. In: Proceeding of Computer Vision and Pattern Recognition, pp. 1–7 (2008)
25. Graf, A., Smola, A., Borer, S.: Classification in a normalized feature space using support vector machines. IEEE Trans. Neural Netw. **14**(3), 597–605 (2003)
26. Grauman, K., Darrell, T.: The pyramid match kernel: efficient learning with sets of features. J. Mach. Learn. Res. **8**, 725–760 (2007)
27. Hamsici, O., Martinez, A.: Rotation invariant kernels and their application to shape analysis. IEEE Trans. Pattern Anal. Mach. Intell. **31**(11), 1985–1999 (2009)
28. Harandi, M., Salzmann, M.: Riemannian coding and dictionary learning: Kernels to the rescue. In: Proceeding of Computer Vision and Pattern Recognition, pp. 3926–3935 (2015)
29. Harandi, M., Sanderson, C., Shen, C., Lovell, B.: Dictionary learning and sparse coding on Grassmann manifolds: An extrinsic solution. In: International Conference on Computer Vision, pp. 3120–3127 (2013)
30. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer, New York (2009)
31. Hoyer, P.: Non-negative sparse coding. In: Neural Networks for Signal Processing, pp. 557–565 (2002)

32. Hoyer, P.: Non-negative matrix factorization with sparseness constraints. J. Mach. Learn. Res. **5**, 1457–1469 (2004)
33. Jayasumana, S., Salzmann, M., Li, H., Harandi, M.: A framework for shape analysis via Hilbert space embedding. In: International Conference on Computer Vision, pp. 1249–1256 (2013)
34. Jiang, W., Nie, F., Huang, H.: Robust dictionary learning with capped l_1-norm. In: Proceeding of International Conference on Artificial Intelligence, pp. 3590–3596 (2015)
35. Johnson, J., Olshausen, B.: The recognition of partially visible natural objects in the presence and absence of their occluders. Vision Res. **45**, 3262–3276 (2005)
36. Kendall, D.: A survey of the statistical theory of shape. Statist. Sci. **4**(2), 87–99 (1989)
37. Lan, A., Waters, A., Studer, C., Baraniuk, R.: Sparse factor analysis for learning and content analytics. J. Mach. Learn. Res. **15**(1), 1959–2008 (2014)
38. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proc. IEEE **86**(11), 2278–2324 (1998)
39. Lu, C., Shi, J., Jia, J.: Online robust dictionary learning. In: Proceeding Computer Vision and Pattern Recognition, pp. 415–422 (2013)
40. Mardia, K., Jupp, P.: Directional Statistics. Wiley, Chichester (2000)
41. Nguyen, H., Patel, V., Nasrabadi, N., Chellappa, R.: Design of non-linear kernel dictionaries for object recognition. IEEE Trans. Imag. Proc. **22**(12), 5123–5135 (2013)
42. Nie, F., Huang, H., Cai, X., Ding, C.: Efficient and robust feature selection via joint l2,1-norms minimization. In: Advances in Neural Information Processing Systems, pp. 1813–1821 (2010)
43. Onishchik, A., Sulanke, R.: Projective and Cayley-Klein Geometries. Springer, Heidelberg (2006)
44. Pan, Q., Kong, D., Ding, C., Luo, B.: Robust non-negative dictionary learning. In: Proceedings AAAI Conference on Artificial Intelligence, pp. 2027–2033 (2014)
45. Park, T., Casella, G.: The Bayesian lasso. Am. Stats. **103**(482), 681–686 (2008)
46. Peel, D., Whiten, W., McLachlan, G.: Fitting mixtures of Kent distributions to aid in joint set identification. J. Amer. Stat. Assoc. **96**, 56–63 (2001)
47. Pennec, X., Fillard, P., Ayache, N.: A Riemannian framework for tensor computing. Int. J. Comp. Vis. **66**(1), 41–66 (2006)
48. Rubinstein, R., Bruckstein, A., Elad, M.: Dictionaries for sparse representation modeling. Proc. IEEE **98**(6), 1045–1057 (2010)
49. Scholkopf, B., Smola, A.: Learning with Kernels. MIT Press, Cambridge (2002)
50. Scholkopf, B., Smola, A., Muller, K.R.: Nonlinear component analysis as a kernel eigenvalue problem. Neural Comput. **10**, 1299–1319 (1998)
51. Sra, S.: A new metric on the manifold of kernel matrices with application to matrix geometric means. In: Advances in Neural Information Processing Systems, pp. 144–152 (2012)
52. Sra, S., Cherian, A.: Generalized dictionary learning for symmetric positive definite matrices with application to nearest neighbor retrieval. In: Proceeding of European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases, pp. 318–332 (2012)
53. Srivastava, A., Jermyn, I., Joshi, S.: Riemannian analysis of probability density functions with applications in vision. In: Proceeding of International Conference Computer Vision and Pattern Recognition, pp. 1–8 (2007)
54. Tibshirani, R.: Regression shrinkage and selection via the lasso. J. Royal Stat. Soc. Ser. B **58**(1), 267–288 (1996)

55. Turaga, P., Veeraraghavan, A., Srivastava, A., Chellappa, R.: Statistical computations on Grassmann and Stiefel manifolds for image and video-based recognition. IEEE Trans. Pattern Anal. Mach. Intell. **33**(11), 2273–2286 (2011)
56. Wagner, A., Wright, J., Ganesh, A., Zhou, Z., Mobahi, H., Ma, Y.: Towards a practical face recognition system: Robust alignment and illumination by sparse representation. IEEE Trans. Pattern Anal. Mach. Intell. **34**(2), 372–386 (2012)
57. Wang, J., Yang, J., Yu, K., Lv, F., Huang, T., Gong, Y.: Locality-constrained linear coding for image classification. In: Proceeding Computer Vision Pattern Recognition, pp. 3360–3367 (2010)
58. Wang, N., Wang, J., Yeung, D.Y.: Online robust non-negative dictionary learning for visual tracking. In: Proceeding International Conference on Computer Vision, pp. 657–664 (2013)
59. Wong, E., Awate, S.P., Fletcher, P.T.: Adaptive sparsity in Gaussian graphical models. In: International Conference Machine Learning, vol. 1, pp. 311–319 (2013)
60. Wu, Z.: The effective energy transformation scheme as a special continuation approach to global optimization with application to molecular conformation. SIAM J. Opt. **6**, 748–768 (2006)
61. Xie, Y., Ho, J., Vemuri, B.: On a nonlinear generalization of sparse coding and dictionary learning. J. Mach. Learn. Res. **28**, 1480–1488 (2013)
62. Yu, K., Zhang, T.: Improved local coordinate coding using local tangents. In: Proceeding International Conference Machine learning, pp. 1215–1222 (2010)
63. Yu, K., Zhang, T., Gong, Y.: Nonlinear learning using local coordinate coding. In: Advances in neural information processing systems, pp. 2223–2231 (2009)
64. Yukawa, M., Amari, S.I.: l_p-regularized least squares $(0 < p < 1)$ and critical path. IEEE Trans. Info. Th. **62**(1), 488–502 (2016)
65. Zeng, X., Bian, W., Liu, W., Shen, J., Tao, D.: Dictionary pair learning on Grassmann manifolds for image denoising. IEEE Trans. Imag. Proc. **24**(11), 4556–4569 (2015)