On A Nonlinear Generalization of Sparse Coding and Dictionary Learning

Yuchen Xie YXIE@CISE.UFL.EDU
Qualcomm Technologies, Inc., San Diego, CA 92121 USA

Jeffrey Ho JHO@CISE.UFL.EDU
Baba Vemuri VEMURI@CISE.UFL.EDU
University of Florida, Gainesville, FL 32611 USA

Abstract

Existing dictionary learning algorithms are based on the assumption that the data are vectors in an Euclidean vector space $\mathbb{R}^d$, and the dictionary is learned from the training data using the vector space structure of $\mathbb{R}^d$ and its Euclidean $\mathbf{L}^2$-metric. However, in many applications, features and data often originated from a Riemannian manifold that does not support a global linear (vector space) structure. Furthermore, the extrinsic viewpoint of existing dictionary learning algorithms becomes inappropriate for modeling and incorporating the intrinsic geometry of the manifold that is potentially important and critical to the application. This paper proposes a novel framework for sparse coding and dictionary learning for data on a Riemannian manifold, and it shows that the existing sparse coding and dictionary learning methods can be considered as special (Euclidean) cases of the more general framework proposed here. We show that both the dictionary and sparse coding can be effectively computed for several important classes of Riemannian manifolds, and we validate the proposed method using two well-known classification problems in computer vision and medical imaging analysis.

1. Introduction

Dictionary learning has been widely used in machine learning applications such as classification, recognition, image restoration and others (e.g., (Aharon et al., 2006)). Under this model, each data point is assumed to be generated *linearly* using only a small number of atoms, and this assumption of linear sparsity is responsible for much of its generalization power and success. The underlying linear process requires that the data points as well as the atoms are vectors in a vector space $\mathbb{R}^d$, and the dictionary is learned from the input data using the vector space structure of $\mathbb{R}^d$ and its metric (typically the $\mathbf{L}^2$-norm). However, for many applications such as those in directional statistics (e.g., (Mardia & Jupp, 1999)), machine learning (e.g., (Yu & Zhang, 2010)), computer vision (e.g., (Turaga et al., 2008)), and medical image analysis, data and features are often presented as points on known Riemannian manifolds, e.g., the space of symmetric positive-definite matrices (Fletcher & Joshi, 2007), hyperspheres for parameterizing square-root densities (Srivastava et al., 2007), Stiefel and Grassmann manifolds (Mardia & Jupp, 1999), etc.. While these manifolds are equipped with metrics, their lack of vector space structures is a significant hindrance for dictionary learning, and primarily because of this, it is unlikely that the existing dictionary learning methods can be extended to manifold-valued data without serious modifications and injections of new ideas. This paper takes a small step in this direction by proposing a principled extension of the linear sparse dictionary learning in Euclidean space $\mathbb{R}^d$ to general Riemannian manifolds.

By the Nash embedding theorem (Nash, 1956), any abstractly-defined Riemannian (data) manifold $\mathcal{M}$ can be isometrically embedded in some Euclidean space $\mathbb{R}^d$, i.e., $\mathcal{M}$ can be considered as a Riemannian submanifold of $\mathbb{R}^d$. It is tempting to circumvent the lack of linear structure by treating points in $\mathcal{M}$ as points in the embedding space $\mathbb{R}^d$ and learning the dictionary in

Proceedings of the 30th International Conference on Machine Learning, Atlanta, Georgia, USA, 2013. JMLR: W&CP volume 28.

$\mathbb{R}^d$. Unfortunately, this immediately raises two thorny issues regarding the suitability of this approach. First, for most manifolds, such as Grassmann and Stiefel manifolds, there simply does not exist known **canonical** embedding into $\mathbb{R}^d$ (or such embedding is difficult to compute), and although Nash's theorem guarantees the existence of such embedding, its non-uniqueness is a difficult issue to resolve as different embeddeings are expected to produce different results. Second, even in the case when the existing methods can be applied, due to their extrinsic nature (both the vector space structure and metric structure in $\mathbb{R}^d$ are extrinsic to $\mathcal{M}$), important intrinsic properties of the data manifold are still very difficult to capture using an extrinsic dictionary. For example, a linear combination of atoms in $\mathbb{R}^d$ does not in general define a proper feature (a point in $\mathcal{M}$), and the inadequacy can be further illustrated by another simple example: it is possible that two points $x, y \in \mathcal{M}$ have a large geodesic distance separating them but under an embedding $i : \mathcal{M} \to \mathbb{R}^d$, $i(x), i(y)$ are near each other in $\mathbb{R}^d$. Therefore, sparse coding using dictionary learned in $\mathbb{R}^d$ is likely to code $i(x), i(y)$ (and hence x, y) using the same set of atoms with similar coefficients. This is undesirable for classification and clustering applications that use sparse coding coefficients as discriminative features, and for dictionary learning to be useful for manifold-valued data, sparse coding must reflect some degree of similarity between the two samples $x, y \in \mathcal{M}$ as measured by the intrinsic metric of $\mathcal{M}$.

While the motivation for seeking an extension of the existing dictionary learning framework to the more general nonlinear (manifold) setting has been outlined above, it is by no means obvious how the extension should be correctly formulated. Let $x_1, \cdots, x_n$ denote a collection of data points on a Riemannian manifold $\mathcal{M}$. An important goal of dictionary learning on $\mathcal{M}$ is to compute a collection of atoms $\{a_1, \cdots, a_m\} \subset \mathcal{M}$, also points in $\mathcal{M}$, such that each data point x_i can be *generated* using only a small number of atoms (sparsity). In the Euclidean setting, this is usually formulated as

$$\min_{D, w_1, \cdots, w_n} \sum_{i=1}^{n} \|x_i - D w_i\|^2 + \mathbf{Sp}(w_i), \qquad (1)$$

where D is the matrix with columns composed of the atoms a_i, w_i the sparse coding coefficients for x_i and $\mathbf{Sp}(w)$ the sparsity-promoting regularizer. One immediate technical hurdle that any satisfactory generalization must overcome is the generalization of the corresponding sparse coding problem (with a fixed D above), and in particular, the crucial point is the proper generalization of linear combination $x_i = D w_i$ for data and atoms belonging to the manifold $\mathcal{M}$ that does not support a (global) vector space structure.

Once the nonlinear sparse coding has been properly generalized (Section 3), the dictionary learning algorithm can then be formulated by formally modifying each summand in Equation 1 so that the sparse coding of a data x_i with respect to the atoms $\{a_1, \cdots, a_m\} \subset \mathcal{M}$ is now obtained by minimizing (log denoting the Riemannian logarithm map (do Carmo, 1992))

$$\min_{w_i} \| \sum_{j=1}^{m} w_{ij} \log_{x_i} a_j \|_{x_i}^2 + \mathbf{Sp}(w_i), \qquad (2)$$

with the important affine constraint that $\sum_{j=1}^{m} w_{ij} = 1$, where $w_i = (w_{i1}, \ldots, w_{im})^T$. Mathematically, the lack of global vector space structure on $\mathcal{M}$ is partially compensated by the local tangent space $T_x\mathcal{M}$ at each point x, and global information can be extracted from these local tangent spaces using Riemannian geometry operations such as covariant derivatives, exponential and logarithm maps. We remark that this formulation is completely coordinate-independent since each $\log_{x_i} a_j$ is coordinate-independent, and Equation 2 is a direct generalization of its Euclidean counterpart this is the individual summand in Equation 1. Computationally, the resulting optimization problem given by Equation 2 can be effectively minimized. In particular, the gradient of the objective function in some cases admits a closed-form formula, and in general, it can be evaluated numerically to provide the input for gradient-based optimization algorithms on manifolds, e.g., (Edelman et al., 1998).

Before moving onto the next section, we remark that our context is very different from the context of manifold learning that is perhaps better known in the machine learning community. For the latter, the manifold $\mathcal{M}$ on which the data reside is not known and the focus is on estimating this unknown $\mathcal{M}$ and characterizing its geometry. For us, however, $\mathcal{M}$ and its geometry (Riemannian metric) are known, and the goal is not to estimate $\mathcal{M}$ from the data but to compute a dictionary as a finite subset of points in $\mathcal{M}$.

2. Preliminaries

This section presents a brief review of the necessary background material from Riemannian geometry that are needed in the later sections and we refer to (Spivak, 1979) for more details. A manifold $\mathcal{M}$ of dimension d is a topological space that is locally homeomorphic to open subsets of the Euclidean space $\mathbb{R}^d$. With a globally defined differential structure, manifold $\mathcal{M}$ becomes a differentiable manifold. The tan-

gent space at $x \in \mathcal{M}$, denoted by $T_x\mathcal{M}$, is a vector space that contains all the tangent vectors to $\mathcal{M}$ at x. A Riemannian metric on $\mathcal{M}$ associates to each point $x \in \mathcal{M}$ an inner product $\langle \cdot, \cdot \rangle_x$ in the tangent space $T_x\mathcal{M}$. Let x_i, x_j be two points on the manifold $\mathcal{M}$. A geodesic $\gamma : [0,1] \to \mathcal{M}$ is a smooth curve with vanishing covariant derivative of its tangent vector field, and in particular, the Riemannian distance between two points $x_i, x_j \in \mathcal{M}$, $\mathbf{dist}^2_{\mathcal{M}}(x_i, x_j)$, is the infimum of the lengths of all geodesics joining x_i, x_j. Let $v \in T_x\mathcal{M}$ be a tangent vector at x. There exists a unique geodesic γ_v satisfying $\gamma_v(0) = x$ with initial tangent vector v, and the Riemannian exponential map (based at x) is defined as $\exp_x(v) = \gamma_v(1)$. The inverse of the exponential map $\exp_x$ is the log map, denoted as $\log_x : \mathcal{M} \to T_x\mathcal{M}$. We remark that in general the domain of $\exp_x$ is not the entire tangent space $T_x\mathcal{M}$ and similarly, $\log_x$ is not defined on all of $\mathcal{M}$. However, for technical reasons, we will assume in the following sections that $\mathcal{M}$ is a complete Riemannian manifold (Spivak, 1979) such that $\log_x$ is defined everywhere in $\mathcal{M}$ for all $x \in \mathcal{M}$, and the subtle technical point when $\exp, \log$ are not defined everywhere will be addressed in a future work. Under this assumption, there are two important consequences: 1) the geodesic distance between x_i and x_j can be computed by the formula $\mathbf{dist}_{\mathcal{M}}(x_i, x_j) = \| \log_{x_i}(x_j) \|_{x_i}$, and 2) the squared distance function $\mathbf{dist}^2_{\mathcal{M}}(x, -)$ is a smooth function for all $x \in \mathcal{M}$.

3. Nonlinear Sparse Coding

In linear sparse coding, a collection of m atoms $a_1, \cdots, a_m$, are given that form the columns of the (overcomplete) dictionary matrix D. The sparse coding of a feature vector $x \in \mathbb{R}^d$ is determined by the following l_0-minimization problem:

$$\min_{w \in \mathbb{R}^m} \|w\|_0, \ \ \mathbf{s.t.}\ x = \mathcal{F}_D(w), \tag{3}$$

where the function $\mathcal{F}_D : \mathbb{R}^m \to \mathbb{R}^d$ is defined as $\mathcal{F}_D(w) = Dw$. In the proposed nonlinear generalization of sparse coding to a Riemannian manifold $\mathcal{M}$, the main technical difficulty is the proper interpretation of the function $\mathcal{F}_D(w) : \mathbb{R}^m \to \mathcal{M}$ in the manifold setting, where the atoms $a_1, \cdots, a_m$ are now points in $\mathcal{M}$ and D now denotes the set of atoms since it is no longer possible to stack together the atoms to form a matrix as in the linear case.

Moving to the more general manifold setting, we have forsaken the vector space structure in $\mathbb{R}^d$, and at the same time, we are required to work only with notions that are coordinate independent (do Carmo, 1992). This latter point is related to the fact that on $\mathcal{M}$ there does not have a special point such as the origin (zero vector) in $\mathbb{R}^d$, and the subtlety of this point seems to have been under-appreciated. In particular, the notion of sparsity that we are accustomed to is very much dependent on the choice (location) of the origin.

Sparsity, Coordinate Invariance and Affine Constraint To illustrate the above point, let $x \in \mathbb{R}^d$ be a sparse vector (according to D) such that $x = a_1 + a_2 + a_3$, i.e., x can be reconstructed using only three atoms in D. Changing the coordinates by translating the origin to a new vector t, each atom a_i becomes $a_i - t$, and similarly for x, we have $x - t$. Under this new coordinates with a different origin, $x - t$ can no longer be reconstructed using the three (translated) atoms $a_1 - t, a_2 - t, a_3 - t$, and most likely, in this new coordinates, the same point cannot be reconstructed by a small number of atoms, i.e., it is not a sparse vector with respect to the dictionary D. This is not surprising because linear sparse coding considers each point in $\mathbb{R}^d$ as a vector whose specification requires a reference point (the origin). However, in nonlinear setting, each point cannot be considered as a vector and therefore, must be considered as a point, and this particular viewpoint is the main source of differences between linear and nonlinear sparse coding.

Fortunately, we can modify the usual notion sparsity using affine constraint to yield a coordinate-independent notion of sparsity: a vector is a (affine) sparse vector if it can be written as an affine linear combination of a small number of vectors:

$$x = w_1 a_1 + w_2 a_1 + ... + w_s a_s, \quad w_1 + ... + w_s = 1.$$

It is immediately clear that that the notion of affine sparsity is a coordinate-independent notion as the location of the origin is immaterial (thanks to the affine constraint), and it is this notion of affine sparsity that will be generalized. We note that the affine constraint was also used in (Yu et al., 2009; Yu & Zhang, 2010) for a related but different reason. We also remark that the usual exact recovery results (e.g., (Elad, 2010)) that establish the equivalence between the l_0-problem above and its l_1-relaxation remain valid for affine sparsity since the extra constraint introduced here is convex.

Re-interpreting $\mathcal{F}_D$ It is natural to require that our proposed definition for $\mathcal{F}_D$ on a manifold $\mathcal{M}$ must reduce to the usual linear combination of atoms if $\mathcal{M} = \mathbb{R}^d$. Furthermore, it is also necessary to require that $\mathcal{F}_D(w) \in \mathcal{M}$ and it is well-defined and computable with certain intrinsic properties of $\mathcal{M}$ playing a crucial role. An immediate candidate would be

$$\mathcal{F}_D(w) = \mathbf{argmin}_{x \in \mathcal{M}} \Psi_{D,w}(x), \tag{4}$$

where $\Psi_{D,w}(x)$ is the function of weighted sum of squared distances to the atoms:

$$\Psi_{D,w}(x) = \sum_{i=1}^{m} w_i \, \mathbf{dist}^2_{\mathcal{M}}(x, a_i).$$

While this definition uses the Riemannian geodesic distances, the intrinsic quantities that the learning algorithm is supposed to incorporate, it suffers from two main shortcomings. First, since there is no guarantees that the global minimum of $\Psi_{D,w}$ must be unique, $\mathcal{F}_D(w)$ is no longer single-valued but a *multi-valued* function in general. Second, much more importantly, $\mathcal{F}_D(w)$ may not exist at all for some w (e.g., when it contains both positive and negative weights w_i). However, if we are willing to accept a multi-valued generalization $\mathcal{F}_D(w)$, then the second shortcoming can be significantly remedied by defining

$$\mathcal{F}_D(w) = \mathbf{CP}(\Psi_{D,w}(x)), \tag{5}$$

where $\mathbf{CP}(\Psi_{D,w}(x))$ denotes the set of critical points of $\Psi_{D,w}(x)$, points $y \in \mathcal{M}$ with vanishing gradient: $\nabla\Psi_{D,w}(y) = 0$.

We remark that under the affine constraint $w_1 + \cdots + w_m = 1$, not all w_i can be zero simultaneously; therefore, $\Psi_{D,w}(x)$ cannot be a constant (zero) function, i.e., $\Psi_{D,w}(x) \neq \mathcal{M}$. Since a global minimum of a smooth function must also be its critical point, this implies that the existence of critical points is less of a problem than the existence of a global minimum. This can be illustrated using the simplest nontrivial Riemannian manifold of positive reals $\mathcal{M} = \mathbb{R}_+$ considered as the space of 1×1 positive-definite matrices equipped with its Fisher Information metric (see next section): for $x, y \in \mathcal{M}$,

$$\mathbf{dist}^2_{\mathcal{M}}(x, y) = \log^2(\frac{x}{y}).$$

For any two atoms $a_1, a_2 \in \mathcal{M}$, the function

$$\Psi_{D,w}(x) = w_1 \, \mathbf{dist}^2_{\mathcal{M}}(x, a_1) + w_2 \, \mathbf{dist}^2_{\mathcal{M}}(x, a_2)$$

does not have a global minimum if $w_1 w_2 < 0$. However, it has one unique critical point $a_1^{w_1} a_2^{w_2} \in \mathcal{M}$. In particular, for any number of atoms $a_1, ..., a_m$ and $w \in \mathbb{R}^m$ satisfying the affine constraint, $\Psi_{D,w}(x)$ has one unique critical point

$$F_D(w) = a_1^{w_1} \cdots a_m^{w_m}.$$

Furthermore, we have

Proposition 1 *If $\mathcal{M} = \mathbb{R}^d$, then $\mathcal{F}_D(w)$ is single-valued for all w such that $w_1 + \cdots + w_m = 1$ and*

$$\mathcal{F}_D(w) = w_1 a_1 + \cdots + w_m a_m.$$

The proof is straightforward since the unique critical point y is defined by the condition $\nabla\Psi_{D,w}(y) = 0$ and $\Psi_{D,w}(y) = w_1\|y - a_1\|^2 + \cdots + w_m\|y - a_m\|^2$. Thanks to the affine constraint, we have

$$y = w_1 a_1 + \cdots + w_m a_m,$$

and $\mathcal{F}_D(w) = y$.

While the acceptance of multi-valued $\mathcal{F}_D(w)$ may be a source of discomfort or even annoyance at first, we list the following five important points that strongly suggest the correctness of our generalization:

1. $\mathcal{F}_D(w) = y$ incorporates the geometry of $\mathcal{M}$ in the form of geodesic distances to the atoms, and in particular, for two nearby points $x, y \in \mathcal{M}$, their sparse codings are expected in general to be similar.
2. The generalization reduces to the correct form for $\mathbb{R}^d$.
3. $\mathcal{F}_D(w)$ is effectively computable because any critical point of a smooth function (e.g., $\Psi_{D,w}$) can be reached via gradient decent or ascent with appropriate initial point.
4. The above point also shows the viability of using $\mathcal{F}_D(w)$ as an approximation to a data x. Essentially, the approximation looks for the critical point of $\Psi_{D,w}$ that is near x, and such critical point, if exists, can be obtained by gradient descent or ascent from x.
5. While $\mathcal{F}_D(w)$ is multi-valued, it is continuous in the following sense: let $t_1, t_2, ... \in \mathbf{R}^m$ be a sequence such that $t_n \to t\mathbf{R}^m$ and $y_1, y_2, ... \in \mathcal{M}$ such that $y_i \in \mathcal{F}_D(t_i)$ for each i and $y_n \to y \in \mathcal{M}$ as $n \to \infty$. Then, $y \in \mathcal{F}_D(t)$. This follows from interpreting the gradient $\nabla\Psi_{D,t_i}$ as a smooth family of vector fields on $\mathcal{M}$ and y_i are the zeros of these vector fields.

We believe that in generalizing sparse coding to Riemannina manifolds, it is not possible to preserve every desirable property enjoyed by the sparse coding in the Euclidean space. In particular, in exchange for the multi-valued $\mathcal{F}_D(w)$, we have retained enough useful features and properties that will permit us to pursuit dictionary learning on Riemannian manifolds.

4. Dictionary Learning on Riemannian Manifolds

In this section, we present a dictionary learning algorithm based on the nonlinear sparse coding framework described above. Given a collection of features

$x_1, \ldots, x_n \in \mathbb{R}^d$, existing dictionary learning methods such as (Olshausen & Field, 1997; Lewicki & Sejnowski, 2000; Aharon et al., 2006) compute a dictionary $D \in \mathbb{R}^{d\times m}$ with m atoms such that each feature x_i can be represented as a sparse linear combination of these atoms $x_i \approx Dw_i$, where $w_i \in \mathbb{R}^m$. Using l_1 regularization on w_i, the learning problem can be succinctly formulated as an optimization problem (Mairal et al., 2010; Yang et al., 2009):

$$\min_{D,w_i} \sum_{i=1}^{n} \left(\|x_i - Dw_i\|_2^2 + \lambda \|w_i\|_1 \right), \tag{6}$$

where λ is a regularization parameter. There are several notable recent extensions of this well-known formula, and they include online dictionary learning (Mairal et al., 2010) and dictionary learning using different regularization schemes such as group-structured sparsity (Szabo et al., 2011; Huang et al., 2011) and local-coordinate constraints (Wang et al., 2010). For our nonlinear generalization, the main point is to make sense of the data fidelity term $\|x_i - Dw_i\|_2^2$ in the manifold setting. Formally, this is not difficult because, using previous notations, the data fidelity term can be defined analogously using

$$\mathbf{dist}_{\mathcal{M}}(x_i, \mathcal{F}_D(w))^2.$$

We remark that $\mathcal{F}_D(w_i)$ is a multi-valued function and the above equation should be interpreted as finding an point $\hat{x}_i \in \mathcal{F}_D(w_i)$ such that $\mathbf{dist}_{\mathcal{M}}(x_i, \hat{x}_i)^2$ is small, i.e., $\hat{x}_i$ is a good approximation of x_i. However, the distance $\mathbf{dist}_{\mathcal{M}}(x_i, \hat{x}_i)$ is generally difficult to compute, and to circumvent this difficulty, we propose a heuristic argument for approximating this distance using a readily computable function. The main idea is to note that $\hat{x}_i$ a critical point of the function $\Psi_{D,w_i}(x)$ and the equation $\nabla \Psi_{D,w_i}(\hat{x}_i) = 0$ implies that

$$\sum_{j=1}^{m} w_{ij} \log_{\hat{x}_i}(a_j) = 0$$

since the LHS is precisely the gradient $\nabla \Psi_{D,w_i}(\hat{x}_i)$ (Spivak, 1979). Therefore, if x_i is a point near $\hat{x}_i$, the tangent vector (at x_i) $\sum_{j=1}^{m} w_{ij} \log_{x_i}(a_j)$ should be close to zero. In particular, this immediately suggests[1] using

$$\| \sum_{j=1}^{m} w_{ij} \log_{x_i}(a_j) \|_{x_i}^2$$

as a substitute for the distance $\mathbf{dist}_{\mathcal{M}}(x_i, \hat{x}_i)^2$.

[1]The validity of this approximation and the heuristic argument will be examined more closely in a future work.

This heuristic argument readily leads to the following optimization problem for dictionary learning on $\mathcal{M}$:

$$\begin{aligned} \min_{\mathbf{W},\mathcal{D}} \; & \sum_{i=1}^{n} \| \sum_{j=1}^{m} w_{ij} \log_{x_i}(a_j) \|_{x_i}^2 + \lambda \|\mathbf{W}\|_1 \\ \text{s.t.} \; & \sum_{j=1}^{m} w_{ij} = 1, \; i = 1, \ldots, n, \end{aligned} \tag{7}$$

where $\mathbf{W} \in \mathbb{R}^{n\times m}$ and w_{ij} denotes its (i,j) component. Similar to Euclidean dictionary learning, the optimization problem can be solved using an iterative algorithm that iteratively performs

1. **Sparse Coding:** fix the dictionary $\mathcal{D}$ and optimize the sparse coefficients $\mathbf{W}$.
2. **Codebook Optimization:** fix the sparse coefficients $\mathbf{W}$ and optimize the dictionary $\mathcal{D}$.

The first step is the regular sparse coding problem, and because the optimization domain is in $\mathbb{R}^{n\times m}$, many fast algorithms are available. The codebook optimization step is considerably more challenging for two reasons. First, the optimization domain is no longer Euclidean but the manifold $\mathcal{M}$. Second, for a typical Riemannian manifold $\mathcal{M}$, its Riemannian logarithm map is very difficult to compute. Nevertheless, there are also many Riemannian manifolds that have been extensively studied by differential geometers with known formulas for their exp/log maps, and these results substantially simplify the computational details. The following two subsection will present two such examples. For optimization on the manifold $\mathcal{M}$, we use a line search-based algorithm to update the dictionary $\mathcal{D}$, and the main idea is to determine a descent direction v (as a tangent vector at a point $x \in \mathcal{M}$) and perform the search on a geodesic in the direction v. The formal similarity between Euclidean and Riemannian line search is straightforward and tansparent, and the main complication in the Riemannian setting is the computation of geodesics. We refer to (Absil et al., 2008) for more algorithm details and convergence analysis.

4.1. Symmetric Positive-Definite Matrices

Let $P(d)$ denote the space of $d\times d$ symmetric positive-definite (SPD) matrices. The tangent space $T_X P(d)$ at every point $X \in P(d)$ can be naturally identified with $\mathrm{Sym}(d)$, the space of $d \times d$ symmetric matrices. The general linear group $GL(d)$ acts transitively on $P(d) : X \longrightarrow GXG^T$, where $X \in P(d)$ and $G \in GL(d)$ is a $d \times d$ invertible matrix. Let $Y, Z \in T_M P(d)$ be two tangent vectors at $M \in P(d)$, and define an inner-product in $T_M P(d)$ using the formula

$$\langle Y, Z \rangle_M = \mathrm{tr}(YM^{-1}ZM^{-1}), \tag{8}$$

where tr is the matrix trace. This above formula defines a Riemannian metric on $P(d)$ that is invariant under the $GL(d)$-action (Pennec et al., 2006; Fletcher & Joshi, 2007), and the corresponding geometry on $P(d)$ has been studied extensively by differential geometers (see (Helgason, 2001) and (Terras, 1985)), and in information geometry, this is the Fisher Information metric for $P(d)$ considered as the domain for parameterizing zero-mean normal distributions on $\mathbb{R}^d$ (Amari & Nagaoka, 2007). In particular, the formulas for computing geodesics, Riemannian exponential and logarithm maps are well-known: The geodesic passing through $M \in P(d)$ in the direction of $Y \in T_M P(d)$ is given by the formula

$$\gamma(t) = G \operatorname{Exp}\left(G^{-1} Y G^{-T} t\right) G^T, \ t \in \mathbb{R}, \tag{9}$$

where Exp denotes the matrix exponential and $G \in GL(d)$ is a square root of M such that $M = GG^T$. Consequently, the Riemannian exponential map at M which maps $Y \in T_M P(d)$ to a point in $P(d)$ is given by the formula

$$\exp_M(Y) = G \operatorname{Exp}\left(G^{-1} Y G^{-T}\right) G^T,$$

and given two positive-definite matrices $X, M \in P(d)$, the Riemannian logarithmic map $\log_M : P(d) \to T_M P(d)$ is given by

$$\operatorname{Log}_M(X) = G \operatorname{Log}(G^{-1} X G^{-T}) G^T,$$

where Log denotes the matrix logarithm. Finally, the geodesic distance between M and X is given by the formula

$$\operatorname{dist}(M, X) = \| \log_M(X) \| = \sqrt{\operatorname{tr}(\operatorname{Log}^2(G^{-1} X G^{-T}))}.$$

The above formulas are useful for specializing Equation 7 to $P(d)$: let $X_1, \ldots, X_n \in P(d)$ denote a collection of $d \times d$ SPD matrices, and $A_1, \ldots, A_m \in P(d)$ the m atoms in the dictionary $\mathcal{D}$. We have

$$\sum_{j=1}^{m} w_{ij} \log_{X_i}(A_j) = \sum_{j=1}^{m} w_{ij} \, G_i \operatorname{Log}(G_i^{-1} A_j G_i^{-T}) \, G_i^T,$$

where $G_i \in GL(d)$ such that $G_i G_i^T = X_i$. With l_1 regularization, dictionary learning using Equation 7 now takes the following precise form for $P(d)$:

$$\min_{\mathbf{W}, \mathcal{D}} \sum_{i=1}^{n} \sum_{j=1}^{m} \sum_{k=1}^{m} w_{ij} w_{ik} \operatorname{tr}(L_{ij} L_{ik}) + \lambda \|\mathbf{W}\|_1$$
$$\textbf{s.t.} \sum_{j=1}^{m} w_{ij} = 1, \ i = 1, \ldots, n,$$

where $L_{ij} = \log(G_i^{-1} A_j G_i^{-T})$. The resulting optimization problem can be solved using the method outlined previously.

4.2. Spheres and Square-Root Density Functions

We next study dictionary learning on spheres, the most well-known class of closed manifolds. In the context of machine learning and vision applications, spheres can be used to parameterize the square roots of density functions. More specifically, for a probability density function p and its (continuous) square root $\psi = \sqrt{p}$, we have

$$\int_S \psi^2 ds = 1.$$

By expanding ψ using orthonormal basis functions (e.g., spherical harmonics if S is a sphere), the above equation allows us to identify ψ as a point on the unit sphere in a Hilbert space (see e.g., (Srivastava et al., 2007)). In other words, for a collection of density functions, we can consider them as a collection of points in some high-dimensional sphere, a finite-dimensional sphere spanned by these points in the unit Hilbertian sphere. Under this identification, the classical Fisher-Rao metric (Rao, 1945) for the density functions p corresponds exactly to the canonical metric on the sphere, and the differential geometry of the sphere is straightforward: Given two points ψ_i, ψ_j on a d-dimensional unit sphere $\mathbf{S}^d$, the geodesic distance is just the angle between ψ_i and ψ_j considered as vectors in $\mathbb{R}^{d+1}$

$$\operatorname{dist}(\psi_i, \psi_j) = \cos^{-1}(\langle \psi_i, \psi_j \rangle). \tag{10}$$

The geodesic started at ψ_i in the direction $v \in T_{\psi_i}(\mathbf{S}^d)$ is given by the formula

$$\gamma(t) = \cos(t)\psi_i + \sin(t)\frac{v}{|v|}, \tag{11}$$

and the exponential and logarithm maps are given by the formulas:

$$\exp_{\psi_i}(v) = \cos(|v|)\psi_i + \sin(|v|)\frac{v}{|v|},$$
$$\log_{\psi_i}(\psi_j) = u \cos^{-1}(\langle \psi_i, \psi_j \rangle) / \sqrt{\langle u, u \rangle},$$

where $u = \psi_j - \langle \psi_i, \psi_j \rangle \psi_i$. We remark that in order to ensure that the exponential map is well-defined, we require that $|v| \in [0, \pi)$, and similarly, the log map $\log_{\psi_i}(x)$ is defined on the sphere minus the antipodal point of ψ_i ($x \neq -\psi_i$).

Using these formulas, we can proceed as before to specialize Equation 7 to the sphere $\mathbf{S}^d$: Let $x_1, \ldots, x_n$ denote a collection of square-root density functions considered as points in a high-dimensional sphere $\mathbf{S}^d$, and $a_i, \ldots, a_m \in \mathbf{S}^d$ the atoms in the dictionary $\mathcal{D}$. $\mathbf{W}$ is an $n \times m$ matrix. Using l_1 regularization, Equation 7

takes the following form for $\mathbf{S}^d$:

$$\min_{\mathbf{W},\mathcal{D}} \sum_{i=1}^{n} \| \sum_{j=1}^{m} w_{ij} \cos^{-1}(\langle x_i, a_j \rangle) \frac{u_{ij}}{|u_{ij}|} \|_{x_i}^2 + \lambda \|\mathbf{W}\|_1$$

$$\textbf{s.t.} \sum_{j=1}^{m} w_{ij} = 1, \; i = 1, \ldots, n,$$

where $u_{ij} = a_j - \langle x_i, a_j \rangle x_i$. The resulting optimization problem can be efficiently solved using the method outlined earlier.

5. Experiments

This section presents the details of the two classification experiments used in evaluating the proposed dictionary learning and sparse coding algorithms. The main idea of the experiments is to transform each (manifold) data point x_i into a feature vector $w_i \in \mathbb{R}^m$ using its sparse coefficients w_i encoded with respect to the trained dictionary. In practice, this approach offers two immediate advantages. First, the sparse feature w_i is encoded with respect to a dictionary that is computed using the geometry of $\mathcal{M}$, and therefore, it is expected to be more discriminative and hence useful than the data themselves (Yang et al., 2009). Second, using w_i as the discriminative feature allows us to train a classifier in the Euclidean space $\mathbb{R}^m$, avoiding the more difficult problem of training the classifier directly on $\mathcal{M}$. Specifically, let $x_1, \ldots, x_n \in \mathcal{M}$ denote the n training data. A dictionary (or codebook) $\mathcal{D} = \{a_1, \ldots, a_m\}$ with m atoms is learned from the training data using the proposed method, and for each x_i, we compute its sparse feature $\mathbf{w}_i \in \mathbb{R}^m$ using the learned dictionary $\mathcal{D}$. For classification, we train a linear Support Vector Machine (SVM) using $w_1, ..., w_n$ as the labelled features. During testing, a test feature $y \in \mathcal{M}$ is first sparse coded using $\mathcal{D}$ to obtain its sparse feature w, and the classification result is computed by applying the trained SVM classifier to the sparse feature w. The experiments are performed using two public available datasets: Brodatz texture dataset and OASIS brain MRI dataset, and training and testing data are allocated by a random binary partition of the available data giving the same number of training and testing data.

For comparison, we use the following three alternative methods: 1) Geodesic K-nearest neighbor (GKNN), 2) SVM on vectorized data and 3) SVM on features sparse coded using a dictionary trained by KSVD (Aharon et al., 2006). GKNN is a K-nearest neighbor classifier that uses Riemannian distance on $\mathcal{M}$ for determining neighbors, and it is a straightforward method for solving classification problems on manifolds. In the experiments, K is set to 5. For the second method (SVM), we directly vectorize the manifold data x_i to form their Euclidean features (without sparse coding) $\tilde{w}_i$ and an SVM is trained using these Euclidean features. The third method (SVM+KSVD) applies the popular KSVD method to train a dictionary using the Euclidean features $\tilde{w}_i$, and it uses a linear SVM on the sparse features encoded used this dictionary. All SVMs used in the experiments are trained using the LIBSVM package (Chang & Lin, 2011). We remark that the first method (GKNN) uses the distance metric intrinsic to the manifold $\mathcal{M}$ without sparse feature transforms. The second and third methods are extrinsic in nature and they completely ignore the geometry of $\mathcal{M}$.

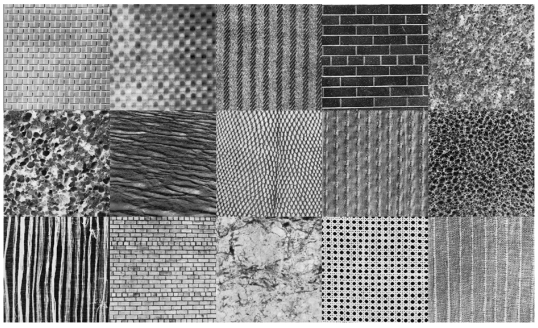

Figure 1. Samples of Brodatz textures used in the first experiment.

Table 1. The texture classification accuracy for different methods on the Brodatz dataset.

Class	SVM	KSVD+SVM	GKNN	Proposed
16	93.36	94.92	95.70	**99.02**
32	88.67	90.82	91.11	**95.70**

5.1. Brodatz Texture Dataset

In this experiment, we evaluate the dictionary learning algorithm for SPD matrices using Brodatz texture dataset (Brodatz, 1966). Using a similar experimental setup as in (Sivalingam et al., 2010), we construct 16-texture and 32-texture sets using the images from Brodatz dataset. Some sample textures are shown in Figure 1. Each 256×256 texture image is partitioned into 64 non-overlapping blocks of size 32×32. Inside each block, we compute a 5×5 covariance matrix $FF^\top$ (Tuzel et al., 2006) summing over the block, where $F = \left(I, \left|\frac{\partial I}{\partial x}\right|, \left|\frac{\partial I}{\partial y}\right|, \left|\frac{\partial^2 I}{\partial x^2}\right|, \left|\frac{\partial^2 I}{\partial y^2}\right|\right)^T$. To ensure that each covariance matrix is positive-definite, we use $FF^\top + \sigma E$, where σ is a small positive constant and E is the identity matrix. For k-class problem, where

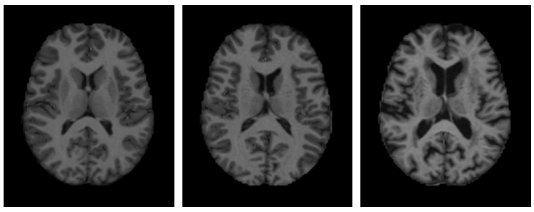

Figure 2. From left to right, three sample images from the OASIS dataset belonging to the Young, Middle-aged and Old groups, respectively.

Table 2. The classification accuracy for different methods on the OASIS dataset. There are three binary classifications (YM: Young vs Middle-aged, MO: Middled-aged vs Old and YO: Young vs Old), and YMO is three-class classification (Young, Middle-aged and Old).

	SVM	KSVD+SVM	GKNN	Proposed
YM	90.04	91.29	91.84	**98.34**
MO	97.32	98.08	100	**100**
YO	98.12	99.46	100	**100**
YMO	91.97	94.04	93.18	**98.62**

k=16 or 32, the size of the dictionary $\mathcal{D}$ is set to $5k$. The texture classification results are reported in Table 1. Our method outperforms the three comparative methods. Among the three comparative methods, sparse feature transform outperforms the one without (KSVD+SVM vs SVM) and the intrinsic method has advantage over extrinsic ones (GKNN vs. SVM and KSVD+SVM). Not surprisingly, our method utilizing both intrinsic geometry and sparse feature transform outperforms all three comparative methods.

5.2. OASIS Dataset

In this experiment, we evaluate the dictionary learning algorithm for square-root densities using the OASIS database (Marcus et al., 2007). OASIS dataset contains T1-weighted MR brain images from a cross-sectional population of 416 subjects. Each MRI scan has a resolution of $176 \times 208 \times 176$ voxels. The ages of the subjects range from 18 to 96. We divide the OASIS population into three (age) groups: young subjects (40 or younger), middle-aged subjects (between 40 and 60) and old subjects (60 or older), and the classification problem is to classify each MRI image according to its age group. Sample images from the three age groups are shown in Figure 2, and the subtle differences in anatomical structure across different age groups are apparent. The MR images in the OASIS dataset are first aligned (with respect to a template) using the nonrigid group-wise registration method described in (Joshi et al., 2004). For each image, we obtain a displacement field, and the histogram of the displacement vectors is computed for each image as the feature for classification (Chen et al., 2010). In our experiment, the number of bins in each direction is set to 4, and the resulting 64-dimensional histogram is used as the feature vector for the SVM+KSVD and SVM methods, while the square root of the histogram is used in GKNN and our method. The dictionaries (KSVD+SVM and our method) in this experiment have 100 atoms. We use five-fold cross validation and report the classification results in Table 2. The pattern among the three comparative methods in the previous experiment is also observed in this experiment, confirming the importance of intrinsic geometry and sparse feature transform. We note that all four methods produce good results for the two binary classifications involving Old group, which can be partially explained by the clinical observation that regional brain volume and cortical thickness of adults are relatively stable prior to reaching age 60 (Mortamet et al., 2005). For the more challenging problem of classifying young and middle-aged subjects, our method significantly outperforms the other three, demonstrating again the effectiveness of combining intrinsic geometry and sparse feature transform for classifying manifold data.

6. Summary and Conclusions

We have proposed a novel dictionary learning framework for manifold-valued data. The proposed dictionary learning is based on a novel approach to sparse coding that uses the critical points of functions constructed from the Riemannian distance function. Compared with the existing (Euclidean) sparse coding, the loss of the global linear structure is compensated by the local linear structures given by the tangent spaces of the manifold. In particular, we have shown that using this generalization, the nonlinear dictionary learning for manifold-valued data shares many formal similarities with its Euclidean counterpart, and we have also shown that the latter can be considered as a special case of the former. We have presented two experimental results that validate the proposed method, and the two classification experiments provide a strong support for the viewpoint advocated in this paper that for manifold-valued data, their sparse feature transforms should be formulated in the context of an intrinsic approach that incorporates the geometry of the manifold.

Acknowledgement This research was supported by the NIH grant NS066340 to BCV.

References

Absil, P.A., Mahony, R., and Sepulchre, R. *Optimization algorithms on matrix manifolds*. Princeton Univ Pr, 2008.

Aharon, M., Elad, M., and Bruckstein, A. K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Trans. on Signal Processing*, 54 (11):4311–4322, 2006.

Amari, S. and Nagaoka, H. *Methods of Information Geometry*. American Mathematical Society, 2007.

Brodatz, P. *Textures: a photographic album for artists and designers*, volume 66. Dover New York, 1966.

Chang, C.-C. and Lin, C.-J. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011.

Chen, T., Rangarajan, A., and Vemuri, B.C. Caviar: Classification via aggregated regression and its application in classifying oasis brain database. In *ISBI*, 2010.

do Carmo, M. *Riemannian Geometry*. Birkhauser, 1992.

Edelman, A., Arias, T.A., and Smith, S.T. The geometry of algorithms with orthogonality constraints. *Arxiv preprint physics/9806030*, 1998.

Elad, M. *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*. Springer, 2010.

Fletcher, P.T. and Joshi, S. Riemannian geometry for the statistical analysis of diffusion tensor data. *Signal Processing*, 87(2):250–262, 2007.

Helgason, S. *Differantial Geometry, Lie Groups, and Symmetric Spaces*. American Mathematical Society, 2001.

Huang, J., Zhang, T., and Metaxas, D. Learning with structured sparsity. *The Journal of Machine Learning Research*, 999888:3371–3412, 2011.

Joshi, S., Davis, B., Jomier, M., and Gerig, G. Unbiased diffeomorphic atlas construction for computational anatomy. *NeuroImage*, 23:S151–S160, 2004.

Lewicki, M.S. and Sejnowski, T.J. Learning overcomplete representations. *Neural computation*, 12(2):337–365, 2000.

Mairal, J., Bach, F., Ponce, J., and Sapiro, G. Online learning for matrix factorization and sparse coding. *The Journal of Machine Learning Research*, 11:19–60, 2010.

Marcus, D.S., Wang, T.H., Parker, J., Csernansky, J.G., Morris, J.C., and Buckner, R.L. Open access series of imaging studies (oasis): cross-sectional mri data in young, middle aged, nondemented, and demented older adults. *Journal of Cognitive Neuroscience*, 19(9):1498–1507, 2007.

Mardia, K. and Jupp, P. *Directional Statistics*. Wiley, 1999.

Mortamet, B., Zeng, D., Gerig, G., Prastawa, M., and Bullitt, E. Effects of healthy aging measured by intracranial compartment volumes using a designed MR brain database. *MICCAI*, pp. 383–391, 2005.

Nash, J. The imbedding problem for riemannian manifolds. *Annals of Mathematics*, 63(1):20–63, 1956.

Olshausen, B.A. and Field, D.J. Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision Research*, 37(23):3311–3325, 1997.

Pennec, X., Fillard, P., and Ayache, N. A riemannian framework for tensor computing. *International Journal of Computer Vision*, 66(1):41–66, 2006.

Rao, C.R. Information and accuracy attainable in the estimation of statistical parameters. *Bull. Calcutta Math. Soc*, 37(3):81–91, 1945.

Sivalingam, R., Boley, D., Morellas, V., and Papanikolopoulos, N. Tensor sparse coding for region covariances. In *ECCV*, pp. 722–735, 2010.

Spivak, M. *A comprehensive introduction to differential geometry*. Publish or perish Berkeley, 1979.

Srivastava, A., Jermyn, I., and Joshi, S. Riemannian analysis of probability density functions with applications in vision. In *CVPR*, 2007.

Szabo, Z., Poczos, B., and Lorincz, A. Online group-structured dictionary learning. In *CVPR*, pp. 2865–2872, 2011.

Terras, A. *Harmonic Analysis on Symmetric Spaces and Applications*. Springer-Verlag, 1985.

Turaga, P., Veeraraghavan, A., and Chellappa, R. Statistical analysis on stiefel and grassmann manifolds with applications in computer vision. In *CVPR*, 2008.

Tuzel, O., Porikli, F., and Meer, P. Region covariance: A fast descriptor for detection and classification. In *ECCV*, pp. 589–600, 2006.

Wang, J., Yang, J., Yu, K., Lv, F., Huang, T., and Gong, Y. Locality-constrained linear coding for image classification. In *CVPR*, pp. 3360–3367, 2010.

Yang, J., Yu, K., Gong, Y., and Huang, T. Linear spatial pyramid matching using sparse coding for image classification. In *CVPR*, pp. 1794–1801, 2009.

Yu, K. and Zhang, T. Improved local coordinate coding using local tangents. In *ICML*, pp. 1215–1222, 2010.

Yu, K., Zhang, T., and Gong, Y. Nonlinear learning using local coordinate coding. *NIPS*, 22:2223–2231, 2009.