Learning Orthogonal Sparse Representations by Using Geodesic Flow Optimization

Henry Schütze, Erhardt Barth, and Thomas Martinetz
{schuetze | barth | martinetz}@inb.uni-luebeck.de
Institute for Neuro- and Bioinformatics
University of Lübeck
Ratzeburger Allee 160
23562 Lübeck, Germany

***Abstract*—In this paper we propose the novel algorithm GF-OSC, which learns an orthogonal basis that provides an optimal K-sparse data representation for a given set of training samples. The underlying optimization problem is composed of two nested subproblems: (*i*) given a basis, to determine an optimal K-sparse coefficient vector for each data sample, and (*ii*) given a K-sparse coefficient vector for each data sample, to determine an optimal basis. Both subproblems have closed form solutions, which can be computed alternately in an iterative manner. Due to the nesting of the subproblems, however, this approach can only find an optimal solution if the underlying sparsity level is sufficiently high. To overcome this shortcoming, our GF-OSC algorithm solves subproblem (*ii*) via gradient descent on the corresponding cost function within the underlying lower dimensional space of free dictionary parameters. This algorithmic substep is based on the geodesic flow optimization framework proposed by Plumbley. On synthetic data, we show in a comparison with four alternative learning algorithms the superior recovery performance of GF-OSC and show that it needs significantly fewer learning epochs to converge. Furthermore, we demonstrate the potential of GF-OSC for image compression. For five standard test images, we derived sparse image approximations based on a GF-OSC basis that was trained on natural image patches. In terms of PSNR, the approximation performance of the GF-OSC basis is between 0.09 to 0.32 dB higher compared to using the 2D DCT basis, and between 1.66 to 3.4 dB higher compared to using the 2D Haar wavelet basis.**

I. Introduction

Early work on sparse coding is based on the efficient-coding hypothesis which proposes that the goal of visual coding is to accurately represent the visual input with minimal neural activity, an idea that goes back to Barlow [1] and is based on earlier work of Ernst Mach and Donald MacKay. From image statistics it is known that natural images do not occupy the entire signal space. As a consequence, they can be encoded sparsely, meaning that they can be represented by a linear combination of rather few elementary signals out of a given collection. Sparsity is a generic principle that is not restricted to visual data only, but applies also to other classes of natural signals, for instance acoustic signals [2].

The fact that natural images can be sparsely encoded has already been utilized in technical applications such as image compression and compressive sampling. By choosing an adequate analytic transform, e.g. the Discrete Cosine Transform (DCT) or suitable wavelets, many transform coefficients of natural images are small and thus need not be encoded [3], [4]. An important progress has been made by going from such predefined transforms to dictionaries that are learned and thereby adapted to particular signal classes [5]. However, to optimally encode natural image patches or even full size images by such learned dictionaries is computationally demanding, due to their non-orthogonal and redundant nature.

Furthermore, sparse representations are important for object recognition and do indeed often emerge in the first layers of deep convolutional neural networks when trained with labelled or unlabelled data.

A. Overcomplete versus Orthogonol Dictionaries

Learning a dictionary for sparse coding is equivalent to identifying, given a set of training samples, an appropriate collection (dictionary) of directions (atoms) in the input space, such that any K-subset of it spans a K-dimensional subspace. The objective is to accurately represent each sample by its projection onto one specific K-simensional subspace, which is optimal for that particular sample.

Learning overcomplete dictionaries allows to arbitrarily increase the collection of atoms to a size larger than the dimensionality of the input space, which in turn increases the number of possible subspaces that can be used for sparse encodings. Subspaces composed from an overcomplete dictionary are, in general, mutually non-orthogonal, which enables a better adaptation to the training data set and can "represent a wider range of signal phenomena" [6]. However, not to require further conditions on the dictionary is problematic when it comes to calculating optimal sparse encodings. In general, this problem is NP-hard for overcomplete dictionaries [7]. Approximative greedy algorithms like Basis Pursuit or Orthogonal Matching Pursuit can only find optimal encodings if the dictionary obeys particular incoherence conditions, which require that the dictionary atoms are not too similar. Incoherence conditions can be interpreted as a relaxation of orthogonality. Note, however, that when learning a dictionary such incoherence conditions are much more difficult to enforce than orthogonality conditions.

Orthogonal dictionaries, on the other hand, are mathematically simple and, moreover, maximally incoherent. All possible K-dimensional subspaces are mutually orthogonal with the implication that optimally sparse encodings can be calculated simply by inner products. Moreover, an orthogonal dictionary can be easily inverted and serves simultaneosuly

as analysis and as synthesis operator. Nevertheless, orthogonal bases learned for sparse coding are able to provide efficient encodings as will be shown by our numerical experiments.

B. Related Work

1) Analytic Transform Design: The problem to design suitable signal transforms to efficiently encode image patches, and to compress images can be traced back to the Fourier Transform and local versions thereof [8] that finally converged to the first JPEG standard [3] based on the DCT. Pioneering work in the field of wavelet analysis [9] led to a signal decomposition scheme [10], [11] that provides orthogonal multiscale transforms simply by translating and dilating an elementary function (see, e.g., [12], [13]).

2) Learning Overcomplete Dictionaries: Olshausen and Field introduced SparseNet, the first batch learning algorithm to learn an overcomplete dictionary that minimizes a regularized joint cost function composed of a representation error term and a term that promotes the sparsity of the data representation [14]. Meanwhile, many alternative algorithms have been proposed to learn such overcomplete dictionaries.

Lewicki and Sejnowski proposed a probabilistic approach by gradient ascent on a log posterior with respect to the dictionary [15]. The authors also deduced that learning an overcomplete sparse coding dictionary is a generalization of Independent Component Analysis (ICA) [16].

Aharon et al. proposed K-SVD [17], an algorithm that generalizes K-means clustering and iterates two alternating stages. In the first stage, a pursuit algorithm approximates the optimal K-sparse representations of the training set. In the second stage, each dictionary atom, as well as associated non-zero coefficients, are sequentially updated via Singular Value Decomposition (SVD) of a particular error matrix.

Alternative approaches to learn overcomplete dictionaries for sparse coding can be found in [18], [19], [20], [21], [22], or [23] to name a few. However, all the above learning algorithms do not attempt to enforce orthogonality and thus learn, in general, non-orthogonal overcomplete dictionaries that can, e.g., capture invariances [6].

3) Learning Orthogonal Dictionaries: A few authors proposed to learn orthogonal dictionaries for sparse coding.

Coifman et al. proposed the Wavelet Packet Transform [24], which is an early attempt to enhance orthogonal transforms with a certain degree of adaptivity to the represented signal by allowing to select a basis from a large collection of dyadic time frequency atoms.

Mishali et al. proposed a two-stage method to learn an orthogonal sparse coding basis [25]. The first stage estimates the entire support pattern of the sparse coefficient matrix, the second stage iteratively adapts (*i*) the non-zero coefficients and (*ii*) the orthogonal basis via SVD based on the estimated support pattern from the first stage. Their approach suffers from a considerable dependence on very high sparsity levels and on the existence of a strictly K-sparse representation of the training data.

Dobigeon et al. proposed the Bayesian framework BOCA to learn undercomplete orthogonal dictionaries for sparse coding [26]. BOCA, however, relies on knowing specific prior distributions of unknown model parameters. Their approach models the sparse coefficients by a Bernoulli-Gaussian process and uses a uniform prior distribution on the Stiefel manifold to find the orthogonal dictionary. A comparison between GF-OSC and BOCA is out of the scope of this paper, because we here only address the task of learning complete orthogonal dictionaries.

Gribonval et al. considered the problem of learning an orthogonal sparse coding basis by minimizing the ℓ_1-norm of the coefficient matrix, such that the product of both matrices synthesizes the training data set [27]. Their main results are identifiability conditions that guarantee local convergence to the generating dictionary by the ℓ_1-norm minimization approach. They showed that the sparse Bernoulli-Gaussian model satisfies these conditions with high probability provided that enough samples are given. However, an explicit algorithm is not proposed. Furthermore, the convergence to the right solution relies on a sufficently good initialization.

Schütze et al. proposed the online learning algorithm OSC to learn an orthogonal basis for a sparse data representation [28]. OSC performs Hebbian-like updates of the dictionary atoms in decreasing order of their encoding relevance. Orthogonality of the dictionary is repeatedly reimposed by a Gram-Schmidt process. On natural image patches, the learned OSC basis attains superior K-term approximation performance compared to analytic orthogonal transforms and PCA. In [29], the same authors present results and argue that with OSC the true sparsity level can be very low and does not even need to be known.

In [29] a "canonical" approach (CA) is introduced to find orthogonal sparse coding bases via batch learning. CA iteratively alternates between (*i*) a sparse coding stage, and (*ii*) a dictionary update stage. For each stage, the closed form solution of the corresponding subproblem is computed.

Bao et al. proposed a batch algorithm to learn an orthogonal sparse coding basis [30]. Their method is related to CA as it computes closed form solutions of the two underlying subproblems. However, they address an unconstrained sparse model with a regularized joint cost function different from the one defined by Eq. (5) - see below. The sparse coding stage is realized by a hard thresholding operator that is applied to the coefficient matrix with a threshold heuristically derived from the regularization parameter λ that implicitly controls the trade-off between reconstruction error and sparsity. Without modifications, their approach does not bound the sparsity level of each sample and is therefore not suitable for a comparison in our experiments. Note that their dictionary update stage is also used by CA and variants of it have also previously been used, e.g., in [19] and [25].

C. Structure of the Paper

In Section II, we formally introduce the orthogonal K-sparse coding problem and summarize Plumbley's geodesic flow framework, which is the tool kit used for our GF-OSC algorithm. We derive an online learning rule for GF-OSC that is most relevant for an efficient update of the dictionary.

In Section III, we investigate the performance of GF-OSC when recovering a generating orthogonal basis from synthetic

K-sparse data and compare it to four alternative methods. We then apply GF-OSC to natural image patches and visualize the learned atoms. Moreover, we compute sparse approximations of test images by using the learned GF-OSC basis and compare its compression performance to that of the 2D DCT and 2D Haar wavelet bases.

II. Methods

A. Learning an Orthogonal Basis for Sparse Coding

This paper addresses the task of learning an orthogonal basis that provides an optimal K-sparse representation of a given training data set. We say that matrix $\boldsymbol{U} \in \mathbb{R}^{N\times N}$ is an orthogonal basis if it satisfies $\boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I}_N$. Given such a basis, any signal $\underline{x} \in \mathbb{R}^N$ has a unique representation by its coefficient vector $\underline{a} = \boldsymbol{U}^T\underline{x}$. We say $\underline{x}$ is K-sparse in $\boldsymbol{U}$, if its corresponding $\underline{a}$ has exactly K non-zero entries, which will be denoted by $\|\underline{a}\|_0 = K$. In the following, let $K \in \{1, ..., N-1\}$ be a specific sparsity level.

Suppose $\boldsymbol{U}$ is given, then an optimal coefficient vector (having a sparsity level of at most K) of a single sample $\underline{x}$ is found as a solution to

$$\underline{a}^*_{\boldsymbol{U},K}(\underline{x}) = \underset{\underline{a},\|\underline{a}\|_0\leq K}{\arg\min} \|\underline{x} - \boldsymbol{U}\underline{a}\|_2^2 \,. \tag{1}$$

Due to orthogonality and completeness of $\boldsymbol{U}$, a minimizer of the sparse approximation problem (1) can be easily determined by keeping the K largest entries $|a_n|$ of $\underline{a} = \boldsymbol{U}^T\underline{x}$ and setting the remaining entries to zero, which can be written as

$$\underline{a}^*_{\boldsymbol{U},K}(\underline{x}) = \boldsymbol{D}_K(\underline{x},\boldsymbol{U})\boldsymbol{U}^T\underline{x} \,, \tag{2}$$

where $\boldsymbol{D}_K(\underline{x},\boldsymbol{U})$ is a diagonal matrix having K entries equal to 1 (indicating the K largest entries $|a_n|$) and otherwise the entries equal to 0.

Let $\boldsymbol{X} = (\underline{x}_1, ..., \underline{x}_L)$ be a given training data set and $\boldsymbol{A}^*_{\boldsymbol{U},K}(\boldsymbol{X})$ a matrix of equal size as $\boldsymbol{X}$ which contains, for each sample $\underline{x}_i$, solution (2) to the sparse approximation problem (1) in its columns. The proper cost function which has to be minimized is given by

$$\begin{aligned} E_{\boldsymbol{X},K}(\boldsymbol{U}) &= \|\boldsymbol{X} - \boldsymbol{U}\boldsymbol{A}^*_{\boldsymbol{U},K}(\boldsymbol{X})\|_{\mathrm{F}}^2 &(3)\\ &= \sum_{i=1}^{L}\|\underline{x}_i - \boldsymbol{U}\underline{a}^*_{\boldsymbol{U},K}(\underline{x}_i)\|_2^2 &(4)\\ &= \|\boldsymbol{X}\|_F^2 - \sum_{i=1}^{L}\underline{x}_i^T\boldsymbol{U}\boldsymbol{D}_K(\underline{x}_i,\boldsymbol{U})\boldsymbol{U}^T\underline{x}_i \,. &(5)\end{aligned}$$

Note that the first term in (5) does not depend on $\boldsymbol{U}$ and can therefore be disregarded. We denote the minimizer of (5) by $\boldsymbol{U}^*_{\boldsymbol{X},K}$. When there is no risk of confusion, we will simply write $E(\boldsymbol{U})$ for (5), and $\boldsymbol{U}^*$ for its minimizer, respectively.

Our experiments have shown that minimizing (5) via batch learning is not as effective as via pattern-by-pattern learning. Therefore, we additionally write the cost function in terms of a single training sample, which is equivalent to (5) apart from the sum that is taken over a single summand only.

$$\begin{aligned} E_{\underline{x},K}(\boldsymbol{U}) &= \|\underline{x} - \boldsymbol{U}\underline{a}^*_{\boldsymbol{U},K}(\underline{x})\|_2^2 &(6)\\ &= \|\underline{x}\|_2^2 - \underline{x}^T\boldsymbol{U}\boldsymbol{D}_K(\underline{x},\boldsymbol{U})\boldsymbol{U}^T\underline{x} \,. &(7)\end{aligned}$$

B. The Geodesic Flow Optimization Framework

In general, minimizing a scalar-valued cost function with respect to a square matrix is an optimization problem with an N^2-dimensional search space. If, in addition, an orthogonality constraint is incorporated, the search space can be considerably reduced because any orthogonal $N \times N$ matrix has merely $\frac{N(N-1)}{2}$ degrees of freedom, rather than N^2.

For this kind of optimization problems, Plumbley proposed the geodesic flow framework [31] which exploits the reduced search space. Suppose the corresponding cost function is differentiable, then the geodesic flow approach allows to derive its gradient within the reduced space of free parameters, and therefore gradient based optimization techniques can be applied to minimize the cost function.

The set of orthogonal matrices, $O(N) = \{\boldsymbol{U} \in \mathbb{R}^{N\times N} \mid \boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I}_N\}$, is called (general) orthogonal group and consists of two disjoint subgroups[1]: $SO(N)$, the set of orthogonal matrices with determinant $+1$, and $O(N) \setminus SO(N)$, the set of orthogonal matrices with determinant -1. The geodesic flow approach is restricted to the subgroup $SO(N)$, because it is not possible to go smoothly from one subgroup to the other. $SO(N)$ forms a Lie group with an associated Lie algebra given by the set of skew-symmetric matrices, $\mathfrak{so}(N) = \{\boldsymbol{B} \in \mathbb{R}^{N\times N} \mid \boldsymbol{B}^T = -\boldsymbol{B}\}$ and the Lie bracket given by the matrix cummutator $[\boldsymbol{Q},\boldsymbol{R}] = \boldsymbol{Q}\boldsymbol{R} - \boldsymbol{R}\boldsymbol{Q}$. Since $SO(N)$ is a matrix Lie group, the matrix exponential $\exp(\boldsymbol{B}) = \sum_{n=0}^{\infty}\frac{\boldsymbol{B}^n}{n!}$ provides a surjective mapping from $\mathfrak{so}(N)$ to $SO(n)$.

Let $E : \mathbb{R}^{N\times N} \to \mathbb{R}$ be the differentiable cost function that is to be optimized under the orthogonality constraint. By using gradient $\nabla_{\boldsymbol{U}}E$, the gradient of E with respect to the Lie algebra $\mathfrak{so}(N)$ is derived as follows:

$$\nabla_{\boldsymbol{B}}E = (\nabla_{\boldsymbol{U}}E)\,\boldsymbol{U}^T - \boldsymbol{U}\,(\nabla_{\boldsymbol{U}}E)^T \,. \tag{8}$$

The geodesic flow approach starts with some initial $\boldsymbol{U}_0$ and optimizes $\boldsymbol{U}_t$ sequentially according to the iteration variable $t = 1, ..., t_{\max}$. For the most recent $\boldsymbol{U}_{t-1}$ an adaptation within $\mathfrak{so}(N)$ into the steepest descent direction $\Delta\boldsymbol{B} = -\eta\nabla_{\boldsymbol{B}}E$ is determined by (8), where η is a sufficiently small step length. This adaptation within $\mathfrak{so}(N)$ is mapped to $SO(N)$ by the matrix exponential, i.e., $\Delta\boldsymbol{U} = \exp(\Delta\boldsymbol{B})$. Subsequently, the adaptation within $SO(N)$ is applied rotationally to $\boldsymbol{U}_{t-1}$, thus providing the new orthogonal matrix $\boldsymbol{U}_t = (\Delta\boldsymbol{U})\,\boldsymbol{U}_{t-1}$. This iterative scheme enables the minimization of a scalar-valued cost function subject to the $SO(N)$ and based on a gradient descent in $\mathfrak{so}(N)$, which is the space of the underlying degrees of freedom. Each gradient descent step yields naturally a new orthogonal basis $\boldsymbol{U}_t$. As a consequence, reimposing the orthogonality constraint separately is unnecessary.

C. Geodesic Flow Orthogonal Sparse Coding (GF-OSC)

We now present the online learning algorithm GF-OSC for minimizing (5). In order to solve the basis update subproblem by the geodesic flow framework, we first derive $\nabla_{\boldsymbol{U}}E$ of the (single sample) cost function (7) and insert it subsequently into (8) to obtain $\nabla_{\boldsymbol{B}}E$.

[1] However, each of the two subgroups is connected.

Suppose $\underline{x}$ is the current training sample randomly selected from $\boldsymbol{X}$. We solve (1) by using (2) and fixate the locations of the K largest coefficients subject to the current $\boldsymbol{U}$. The gradient with respect to $\boldsymbol{U}$ is given by

$$\nabla_{\boldsymbol{U}} E = -2\underline{x}\underline{x}^T \boldsymbol{U} \boldsymbol{D}_K(\underline{x}, \boldsymbol{U}) \; . \tag{9}$$

Inserting (9) into (8) yields the desired gradient $\nabla_{\boldsymbol{B}} E$ of the cost function (7) with respect to the Lie algebra $\mathfrak{so}(N)$. Note that the derived $\nabla_{\boldsymbol{B}} E$ is the key ingredient of our GF-OSC algorithm and that it can be simplified as follows:

$$\nabla_{\boldsymbol{B}} E = \underline{\hat{x}}_{\boldsymbol{U},K}(\underline{x})\underline{x}^T - \underline{x}\underline{\hat{x}}_{\boldsymbol{U},K}(\underline{x})^T, \tag{10}$$

where $\underline{\hat{x}}_{\boldsymbol{U},K}(\underline{x})$ is an optimal K-term approximation of the sample $\underline{x}$ with respect to $\boldsymbol{U}$.

Algorithm 1 GF-OSC

Input: training data set $\boldsymbol{X} = (\underline{x}_1, ..., \underline{x}_L) \in \mathbb{R}^{N \times L}$
number of learning steps $t_{\max}$
expected sparsity level K
initial orthogonal basis $\boldsymbol{U}_0$
Output: orthogonal basis $\boldsymbol{U} \in SO(N)$

for all $t = 1, ..., t_{\max}$ **do**
  select a sample $\underline{x}$ from $\boldsymbol{X}$ randomly
  compute its optimal K-term approximation $\underline{\hat{x}}_{\boldsymbol{U}_{t-1},K}(\underline{x})$
  compute $\nabla_{\boldsymbol{B}} E$ according to Eq. (10)
  select a suitable step length η_t
  $\Delta\boldsymbol{B} \leftarrow -\eta_t \nabla_{\boldsymbol{B}} E$
  $\Delta\boldsymbol{U} \leftarrow \exp(\Delta\boldsymbol{B})$
  $\boldsymbol{U}_t \leftarrow (\Delta\boldsymbol{U})\boldsymbol{U}_{t-1}$
end for
$\boldsymbol{U} \leftarrow \boldsymbol{U}_{t_{\max}}$

The pseudo code of GF-OSC is listed in Algorithm 1. To update the basis by GF-OSC, different strategies can be chosen to select the step length η_t. It seems natural to apply a dynamic step length that decreases from a large intial value to a small final value over the number of conducted learning steps. We also tested an adaptive step length η calculated via backtracking line search based on the Armijo-Goldstein condition [32] and observed that the convergence of GF-OSC is increased for synthetic data wheras it is worsen in a learning scenario with natural image patches, at least for the constant value of $\alpha = 5$ that we tested.

III. Experiments

A. Experiments on Synthetic Data

We investigated how reliable GF-OSC and four alternative methods recover a generating orthogonal basis from K-sparse synthetic data. To this end, we fixed the signal dimensionality to $N = 256$ and sample size to $L = 1000$, and generated training data sets for sparsity levels $K \in \{2, 6, ..., 58, 62\}$. Each data sample was generated as a 16×16 patch being K-sparse in the non-standard 2D Haar wavelet basis, see Fig. 2a. We modeled the sparse coefficients by a Bernoulli-Gaussian process. The support pattern of each sample, i.e., the K locations of non-zero coefficients (in the Haar wavelet domain) were uniformly selected at random. Subsequently, the K non-zero coefficients were drawn from a standard Gaussian distribution. To investigate deviations of recovery rates over multiple runs, we created 10 different training data sets for each sparsity level. We also randomly generated one initial $\boldsymbol{U}_0$ for each training data set, such that each method starts its iteration at the same initial position.

To measure the basis recovery performance, we followed the procedure proposed in [17], i.e., for each generating basis vector its most similiar estimated basis vector is identified in terms of mutual overlap[2]. We considered a generating basis vector as recovered, if its overlap to its most similar estimated version is at least 0.8. The recovery rate of a full basis is then given by the relative number of recovered basis vectors.

Considering the generated data sets, we compared the basis recovery performances between K-SVD [17], the algorithm of Mishali et al. [25], CA [29], OSC [28], and GF-OSC. All methods were provided with the known sparsity level as user parameter K. Each method was allowed to conduct at most 1000 learning epochs. In the case that all reference basis vectors were already recovered after fewer learning epochs, the learning phase was stopped. Note that K-SVD is an algorithm for learning arbitrary, non-orthogonal sparse coding dictionaries and does therefore not benefit from the orthogonality of an underlying dictionary. Nevertheless, orthogonality is a good-natured scenario for K-SVD, because the mutual coherence is minimal.

With GF-OSC we applied a backtracking line search based on the Armijo-Goldstein condition [32] for which we set $\alpha = 5$. With OSC we let the learning rate decrease exponentially from the initial value $\eta_1 = 10^{-1}$ to the final value $\eta_{t_{\max}} = 10^{-4}$. Note that the choice of this combination affects the ability to converge as well as the speed of convergence.

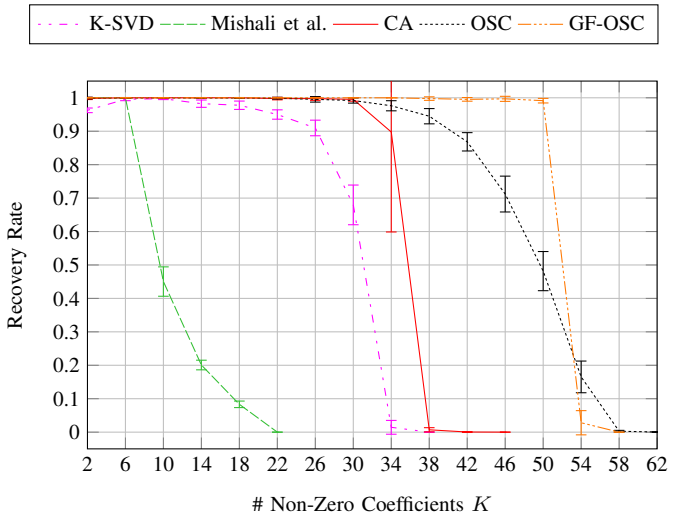
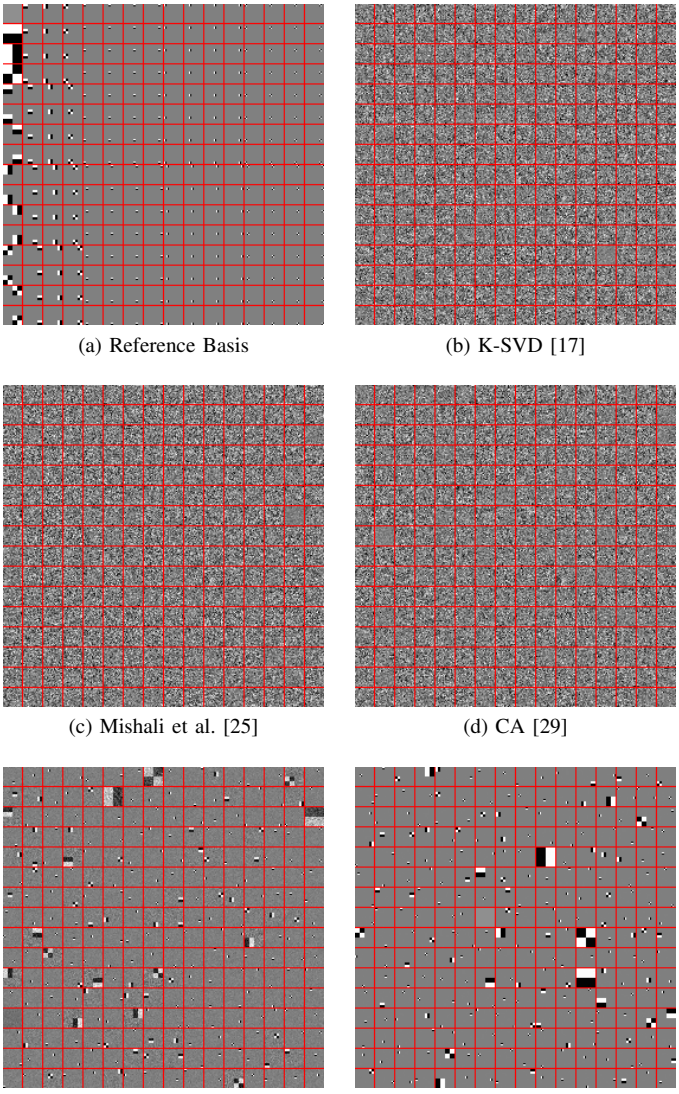


Fig. 1: Mean basis recovery rate (and standard deviations) against sparsity level K on synthetic data sets (1000 patches of size 16×16 being K-sparse in the 2D Haar wavelet basis) over 10 runs. For each method, the total number of learning epochs was limited to 1000.

From Fig. 1 can be seen that the algorithm proposed by Mishali et al. achieves perfect recovery up to $K \leq 6$. Its

[2]The overlap between two unit length vectors $\underline{v}$ and $\underline{w}$ is defined as $|\underline{v}^T \underline{w}|$.

recovery performance decreases for $6 < K \leq 22$, and is zero for $K \geq 22$. The recovery rate of K-SVD is not perfect in all runs, but on average above 0.95 for $K \leq 22$. It decreases for $22 \leq K \leq 34$, and is zero for $K \geq 38$. CA recovers the generating basis nearly perfectly for $K \leq 30$. The recovery performance decreases fast and is nearly zero for $K \geq 38$. OSC recovers the generating basis in a similar range as CA but its recovery performance deacreses very slowly, and is zero for $K \geq 58$. The proposed GF-OSC algorithm performs best with a nearly perfect recovery up to $K \leq 50$.



(a) Reference Basis (b) K-SVD [17] (c) Mishali et al. [25] (d) CA [29] (e) OSC [28] (f) GF-OSC

Fig. 2: Sparse coding bases learned from a synthetic data set (1000 patches of size 16×16 being 42-sparse ($\approx 16.4\%$ non-zero coefficients) in the 2D Haar wavelet basis). For this rather low sparsity level, OSC [28] and GF-OSC are able to extract the underlying reference basis whereas K-SVD [17], the approach of Mishali et al. [25], and CA [29] fail. For display purposes, the entries of each basis patch (except the estimated DC component) are shifted to have zero mean and are subsequently scaled to unit supremum norm.

Fig. 2 illustrates the dictionaries that were learned on a synthetic data set with the rather low sparsity level of $K = 42$ ($\approx 16.4\%$ non-zero coefficients). For this quite challenging scenario, K-SVD, the algorithm of Mishali et al., as well as CA fail to recover the generating basis from the synthetic data set, see Fig. 2b - 2d. The bases learned by OSC and GF-OSC distinctly reveal the underlying Haar wavelet basis, see Fig. 2e - 2f. Note that an optimal solution is merely unique up to the order and signs of the basis vectors.

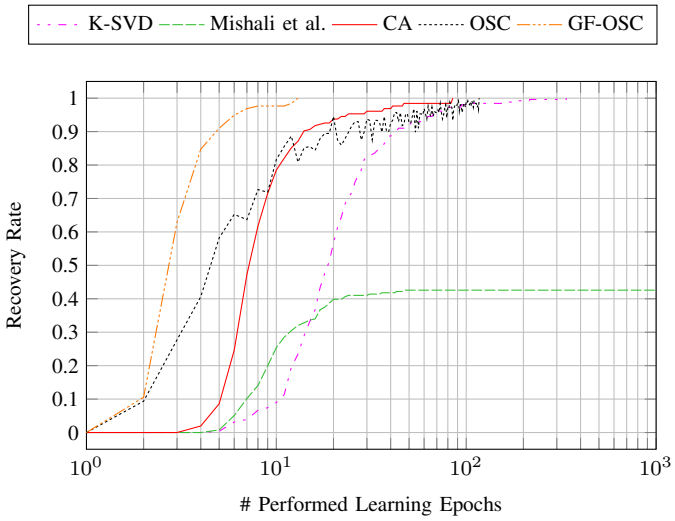


Fig. 3: Recovery rate against the number of performed learning epochs for a single run on a synthetic data set (1000 patches of size 16×16 being 10-sparse in the 2D Haar wavelet basis)

Fig. 3 shows, exemplarily for a single run, a plot of the recovery rate of the sparse coding basis against the number of performed learning epochs. In order to adequately compare the investigated sparse coding methods, a data set with the rather high sparsity level of $K = 10$ was selected. The algortihm of Mishali et al. is merely able to recover $\approx 40\%$ of the reference basis and is saturated at this rate after 50 epochs. K-SVD converges after 337, OSC after 117, CA after 85, and GF-OSC already after 13 learning epochs.

B. Experiments on Natural Image Patches

We let GF-OSC learn orthogonal bases to sparsely encode natural image patches. We extracted image patches from set one of the Nature Scene Collection [33], i.e., from images of nature scenes containing no man made objects or people. The uncompressed RGB images have a resolution of 2844×4284 pixels. The color channels are linearly scaled, each with a depth of 16 bits per pixel (bpp). Each color channel was transformed by $\log_2(\,\cdot\, + 1)$ and subsequently scaled by 2^{-4} into the double precision floating point range $[0, 1]$. Subsequently, the color images were converted to grayscale images. From the entire set of 308 images, we randomly selected 250 images. From each image, we extracted 400 patches of size 16×16 pixels at random positions. These 10^5 image patches were used for the learning with GF-OSC. We set user parameter $K = 64$ and let the learning rate η_t decrease exponentially from $\eta_1 = 10^0$ to $\eta_{t_{\max}} = 10^{-1}$, where the total number of

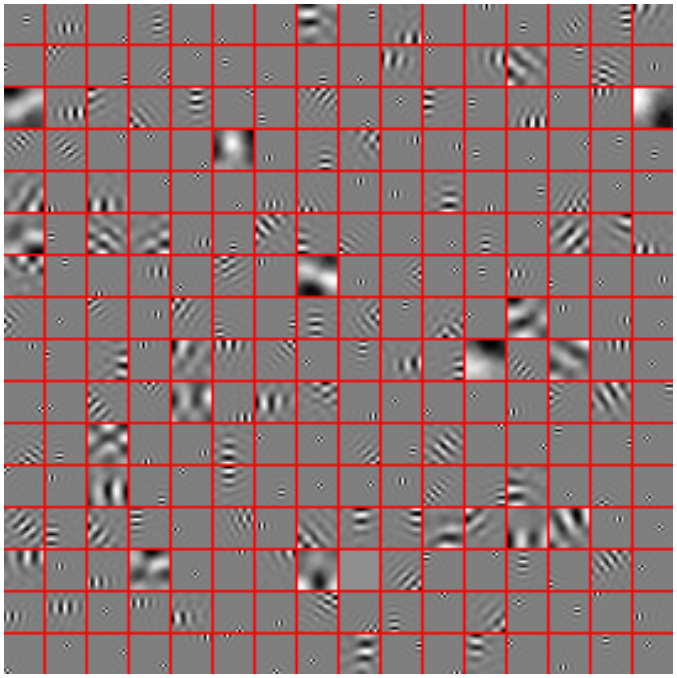


Fig. 4: Orthogonal sparse coding basis learned by GF-OSC on natural image patches. On different scales, the learned basis patches reveal selectivity for inputs with particular frequencies, orientations, and spatial localizations. For display purposes, the entries of each basis patch (except the estimated DC component) are shifted to have zero mean and are subsequently scaled to unit supremum norm.

learning steps was $t_{\max} = 10^7$. The initial basis $\boldsymbol{U}_0$ was an orthogonalized $N \times N$ random matrix.

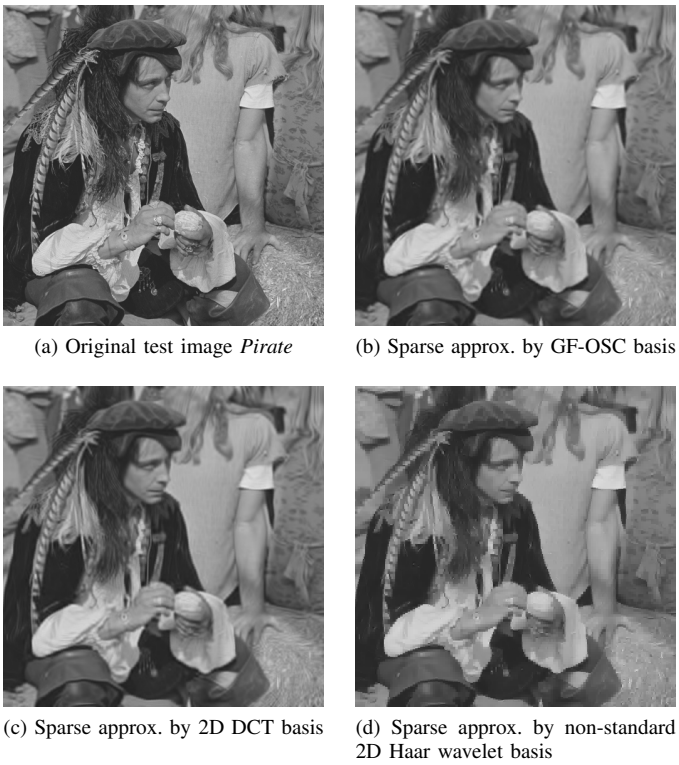
Fig. 4 shows the orthogonal sparse coding basis learned by GF-OSC on the data set of natural image patches. On different scales, the learned basis patches reveal selectivity for inputs with particular frequencies, orientations, and spatial localizations.

C. Sparse Image Approximation using GF-OSC

We conducted experiments to investigate how well natural test images, which were not included in the training, can be sparsely approximated by using the GF-OSC basis depicted in Fig. 4. From each test image (512×512 pixels), we extracted patches of size 16×16 pixels and computed their optimal 8-sparse approximation with respect to the GF-OSC basis. The sparsely approximated patches were then fused back to reconstruct the image. To avoid blocking artifacts we extracted overlapping patches with a stride of 4 pixels, such that all pixels (except for pixels at the margins) of the image are averaged from 16 approximated patches. For a comparison, we interchanged the orthogonal GF-OSC basis with (*i*) the orthogonal 2D DCT basis, and (*ii*) the orthogonal non-standrad 2D Haar wavelet basis. At the chosen parameter set, the test images were approximated more accurately by the GF-OSC basis than by the DCT and Haar wavelet bases. Table I lists the sparse approximation performance as measured by the peak signal-to-noise ratio (PSNR) for five standard test images. Fig. 5 shows results of the sparse image approximation approach for the test image *Pirate* and the three different orthogonal bases.

TABLE I: Sparse approximation performance for test images (512×512 pixels) as measured by the peak signal-to-noise ratio (PSNR).

	GF-OSC	2D DCT	2D Haar
Cameraman	**31.02 dB**	30.93 dB	27.62 dB
Lena	**31.26 dB**	31.12 dB	28.51 dB
Mandril	**26.39 dB**	26.30 dB	24.11 dB
Peppers	**31.08 dB**	30.88 dB	28.82 dB
Pirate	**28.85 dB**	28.57 dB	27.19 dB



(a) Original test image *Pirate*

(b) Sparse approx. by GF-OSC basis

(c) Sparse approx. by 2D DCT basis

(d) Sparse approx. by non-standard 2D Haar wavelet basis

Fig. 5: Sparse approximations of test image *Pirate* (512×512 pixels) based on optimal 8-sparse image patch representations (patch size: 16×16 pixels) with respect to different orthogonal bases.

IV. Conclusion

In this paper we have adressed the problem of learning a complete dictionary with orthogonal atoms to sparsely encode a given set of training data samples. The corresponding optimization problem consist of two nested subproblems: (*i*) given a basis, to determine optimal K-sparse coefficient vectors for each data sample, and (*ii*) given a set of K-sparse coefficient vectors for each data sample, to determine an optimal basis. Both subproblem have per se closed form solutions. Solving these subproblems by alternation in an iterative scheme, as with CA [29], yields acceptable results. However, the GF-OSC algorithm that is proposed in this paper significantly

outperforms CA as well as three other alternative methods (K-SVD [17], the algorithm of Mishali et al. [25], and OSC [28]) at the task of recovering a generating orthogonal basis from synthetic K-sparse data. The superiority is twofold. First, GF-OSC needs fewer learning epochs to converge to the right solution. Second, GF-OSC accurately recovers the correct basis even if the sparsity level is very low, i.e., K is very large.

GF-OSC is an online learning algorithm that solves subproblem (*ii*) via stochastic gradient descent within the space of free dictionary parameters. The corresponding gradient is derived according to the geodesic flow optimization framework proposed by Plumbley.

We used GF-OSC to learn an orthogonal basis from natural image patches and derived basis functions with distinct sensitivity to particular frequencies, orientations and spatial localizations of the inputs. Furthermore, the learned GF-OSC basis seems to be organized over several scales.

We have demonstrated the applicability of GF-OSC by using a basis learned on natural image patches to sparsely approximate images. Due to performance improvements by the learned sparse coding basis over fixed ones, the possibility to integrate GF-OSC into an image compression codec should be further investigated. Since the basis is learned from training examples, it can be adapted to a particular image class and should facilitate further improvements of approximation accuracy and compression rate.

A further future research direction for GF-OSC could be pursued in the field of compressive imaging. Gan proposed a compressed sensing framework for images based on a block decomposition [34]. Since compressed sensing relies on an accurate, sparse representation of the signal in an orthogonal basis, it would be interesting to investigate if a GF-OSC basis (learned on a particular image class) can improve the reconstruction performance of such a compressed sensing approach.

ACKNOWLEDGMENT

The research has been funded by the DFG Priority Programme SPP 1527, grant number MA 2401/2-1.

REFERENCES

[1] H. B. Barlow, "Possible principles underlying the transformation of sensory messages," *Sensory Communication*, pp. 217–234, 1961.

[2] M. Lewicki *et al.*, "Efficient coding of natural sounds," *Nature Neuroscience*, vol. 5, no. 4, pp. 356–363, 2002.

[3] W. B. Pennebaker and J. L. Mitchell, *JPEG Still Image Data Compression Standard*. New York: Van Nostrand Reinhold, 1992.

[4] D. Taubman and M. Marcellin, Eds., *JPEG2000: Image Compression Fundamentals, Standards and Practice*. Springer, 2001.

[5] B. A. Olshausen and D. J. Field, "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," *Nature*, no. 381, pp. 607–609, 1996.

[6] R. Rubinstein, A. M. Bruckstein, and M. Elad, "Dictionaries for Sparse Representation Modeling," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1045–1057, Jun. 2010.

[7] G. Davis, S. Mallat, and M. Avellaneda, "Adaptive greedy approximations," *Constructive Approximation*, vol. 13, no. 1, pp. 57–98, Mar. 1997. [Online]. Available: http://dx.doi.org/10.1007/bf02678430

[8] J. B. Allen and L. R. Rabiner, "A Unified Approach to Short-Time Fourier Analysis and Synthesis," *Proceedings of IEEE*, vol. 65, no. 11, pp. 1558–1564, Nov. 1977.

[9] A. Grossmann and J. Morlet, "Decomposition of Hardy functions into square integrable wavelets of constant shape," *SIAM Journal of Mathematical Analysis*, vol. 15, no. 4, pp. 723–736, 1984. [Online]. Available: http://link.aip.org/link/?SJM/15/723/1

[10] S. G. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 11, no. 7, pp. 674–693, Jul. 1989. [Online]. Available: http://dx.doi.org/10.1109/34.192463

[11] I. Daubechies, "Orthonormal bases of compactly supported wavelets ii: Variations on a theme," *SIAM J. Math. Anal.*, vol. 24, no. 2, pp. 499–519, Mar. 1993. [Online]. Available: http://dx.doi.org/10.1137/0524031

[12] ——, *Ten Lectures on Wavelets*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 1992.

[13] S. Mallat, *A Wavelet Tour of Signal Processing, Third Edition: The Sparse Way*, 3rd ed. Academic Press, Dec. 2008. [Online]. Available: http://www.worldcat.org/isbn/0123743702

[14] B. Olshausen and D. Field, "Sparse coding with an overcomplete basis set: A strategy employed by V1?" 1997.

[15] M. S. Lewicki and T. J. Sejnowski, "Learning overcomplete representations," *Neural Comput.*, vol. 12, no. 2, pp. 337–365, Feb. 2000.

[16] P. Comon, "Independent component analysis, a new concept?" *Signal Process.*, vol. 36, no. 3, pp. 287–314, Apr. 1994. [Online]. Available: http://dx.doi.org/10.1016/0165-1684(94)90029-9

[17] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.

[18] K. Engan, S. O. Aase, and J. Hakon Husoy, "Method of optimal directions for frame design," vol. 5, pp. 2443–2446 vol.5, 1999.

[19] S. Lesage, R. Gribonval, F. Bimbot, and L. Benaroya, "Learning Unions of Orthonormal Bases with Thresholded Singular Value Decomposition," in *Acoustics, Speech and Signal Processing, 2005. ICASSP 2005. IEEE International Conference on*, vol. V. Philadelphia, PA, United States: IEEE, 2005, pp. V/293–V/296.

[20] M. Yaghoobi, T. Blumensath, and M. E. Davies, "Dictionary learning for sparse approximations with the majorization method." *IEEE Transactions on Signal Processing*, vol. 57, no. 6, pp. 2178–2191, 2009.

[21] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online learning for matrix factorization and sparse coding," *J. Mach. Learn. Res.*, vol. 11, pp. 19–60, Mar. 2010.

[22] K. Skretting and K. Engan, "Recursive least squares dictionary learning algorithm." *IEEE Transactions on Signal Processing*, vol. 58, no. 4, pp. 2121–2130, 2010.

[23] K. Labusch, E. Barth, and T. Martinetz, "Sparse coding neural gas: Learning of overcomplete data representations," *Neurocomputing*, vol. 72, no. 7-9, pp. 1547–1555, 2009.

[24] R. R. Coifman, Y. Meyer, and V. Wickerhauser, "Wavelet analysis and signal processing," in *Signal Processing, Part I: Signal Processing Theory*, L. Auslander, T. Kailath, and S. K. Mitter, Eds. New York, NY: Springer-Verlag, 1990, pp. 59–68.

[25] M. Mishali and Y. Eldar, "Sparse source separation from orthogonal mixtures," in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, 2009, pp. 3145–3148.

[26] N. Dobigeon and J.-Y. Tourneret, "Bayesian orthogonal component analysis for sparse representation," *IEEE Trans. Signal Process.*, vol. 58, no. 5, pp. 2675–2685, 2010.

[27] R. Gribonval and K. Schnass, "Dictionary Identifiability from Few Training Samples," in *European Signal Processing Conference (EUSIPCO'08)*, Lausanne, Switzerland, Aug. 2008.

[28] H. Schütze, E. Barth, and T. Martinetz, "Learning orthogonal bases for k-sparse representations," in *Workshop New Challenges in Neural Computation 2013*, ser. Machine Learning Reports, B. Hammer, T. Martinetz, and T. Villmann, Eds., vol. 02/2013, 2013, pp. 119–120.

[29] H. Schütze, E. Barth, and T. Martinetz, "Learning efficient data representations with orthogonal sparse coding," 2014, submitted.

[30] C. Bao, J.-F. Cai, and H. Ji, "Fast sparsity-based orthogonal dictionary learning for image restoration," in *The IEEE International Conference on Computer Vision (ICCV)*, December 2013.

[31] M. D. Plumbley, "Lie Group Methods for Optimization with Orthogo-

nality Constraints," *Independent Component Analysis and Blind Signal Separation*, pp. 1245–1252, 2004.

[32] L. Armijo, "Minimization of functions having lipschitz continuous first partial derivatives." *Pacific J. Math.*, vol. 16, no. 1, pp. 1–3, 1966.

[33] W. S. Geisler and J. S. Perry, "Statistics for optimal point prediction in natural images," *Journal of Vision*, vol. 11, no. 12, Oct. 2011.

[34] L. Gan, "Block compressed sensing of natural images," in *Digital Signal Processing, 2007 15th International Conference on*. IEEE, 2007, pp. 403–406.