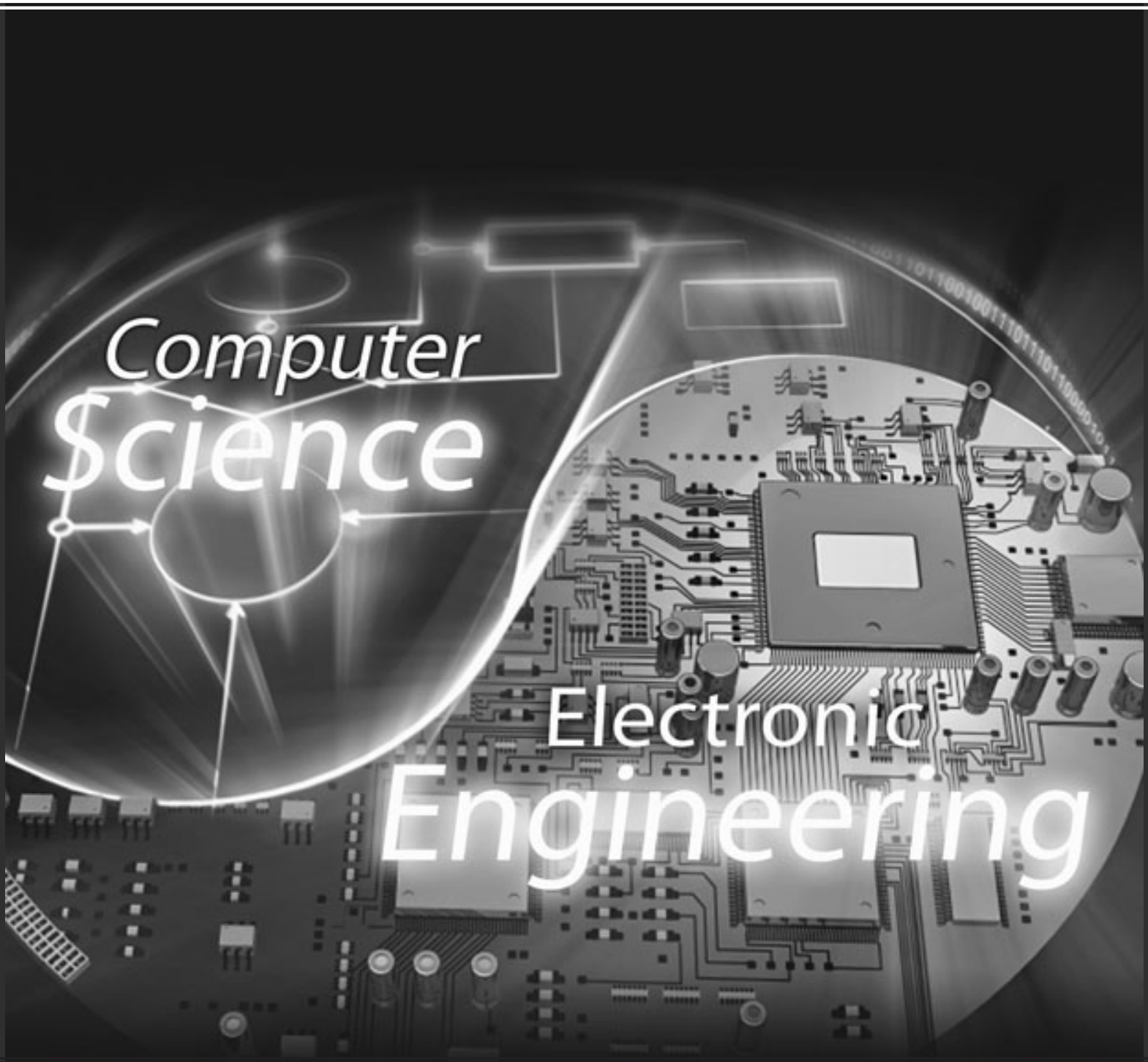
ISSN 2043-0167

Learning incoherent dictionaries for sparse approximation using iterative projections and rotations

Daniele Barchiesi*, Student Member, IEEE and Mark D. Plumbley, Member, IEEE

EECSRR-12-02 April 2012

Queen Mary University of London
School of Electronic Engineering and Computer Science



Learning incoherent dictionaries for sparse approximation using iterative projections and rotations

Daniele Barchiesi*, *Student Member, IEEE* and Mark D. Plumbley, *Member, IEEE.*
School of Electronic Engineering and Computer Science
Queen Mary University of London
Mile End Road, London E1 4NS, UK
Email: <firstname.secondname>@eecs.qmul.ac.uk
Tel: +44 2078827518

Abstract

This article deals with learning dictionaries for sparse approximation whose atoms are both adapted to a training set of signals and mutually incoherent. To meet this objective, we employ a dictionary learning scheme consisting of sparse approximation followed by dictionary update and we add to the latter a decorrelation step in order to reach a target mutual coherence level. This step is accomplished by an iterative projection method followed by a rotation of the dictionary, which is obtained using a Lie group optimisation strategy. Experiments on musical audio data and a comparison with the method of optimal coherence-constrained directions (MOCOD) and the incoherent K-SVD (INK-SVD) illustrate that the proposed algorithm can learn highly incoherent dictionaries while providing a sparse approximation with better signal-to-noise ratio than the benchmark techniques.

Index Terms

Sparse approximation, dictionary learning, iterative projections, Lie group methods.

EDICS Category: AUD-ANSY

This work was supported by the Queen Mary University of London School Studentship, the EU FET-Open project FP7-ICT-225913-SMALL. Sparse Models, Algorithms and Learning for Large-scale data and a Leadership Fellowship from the UK Engineering and Physical Sciences Research Council (EPSRC).

Learning incoherent dictionaries for sparse approximation using iterative projections and rotations

I. Introduction: Learning Incoherent Dictionaries

A. Sparse approximation and dictionary learning

In this paper we consider a sparse synthesis model where a signal $\mathbf{y} \in \mathbb{R}^N$ is approximated by a sparse linear combination of elementary functions $\{\boldsymbol{\phi}_k\}_{k=1}^K, \boldsymbol{\phi}_k \in \mathbb{R}^N$ called *atoms*. Arranging the atoms along the columns of the dictionary matrix $\boldsymbol{\Phi}$, we can express the model as:

$$\mathbf{y} \approx \boldsymbol{\Phi}\mathbf{x} \tag{1}$$

where $\mathbf{x}$ is a sparse vector of representation coefficients, with $||\mathbf{x}||_0 \leq S$. Here the ℓ_0 pseudo-norm $||\cdot||_0$ counts the number of non-zero coefficients of its argument and S is the number of active atoms.

A dictionary learning problem for sparse approximation consists of optimising the set of $K \geq N$ atoms given a set of $M \geq K$ observed data $\{\mathbf{y}_m\}_{m=1}^M$, such that every signal in the training set can be effectively represented by the sparse model (1)[15]. This can be concisely written by arranging the observed signals along the columns of the matrix $\mathbf{Y} \in \mathbb{R}^{N \times M}$:

$$\mathbf{Y} \approx \boldsymbol{\Phi}\mathbf{X} \tag{2}$$

where $\mathbf{X}$ is a sparse matrix whose columns contain the vectors of representation coefficients.

Optimising the dictionary $\boldsymbol{\Phi}$ is a challenging problem for which no analytic solution can be found. The numerical strategy commonly employed consists in iterative algorithms that start from an initial dictionary and alternate between the following steps:

- *Sparse coding*: given a fixed dictionary $\boldsymbol{\Phi}$, the matrix $\mathbf{X}$ of sparse approximation coefficients is calculated using any suitable algorithm for sparse approximation.
- *Dictionary update*: given a fixed approximation matrix $\mathbf{X}$, the dictionary $\boldsymbol{\Phi}$ is updated in order to minimise the residual cost function $||\mathbf{Y} - \boldsymbol{\Phi}\mathbf{X}||_{\mathrm{F}}$.

In addition, the dictionary is usually constrained to belong to a set $\mathcal{D} \stackrel{\mathrm{def}}{=} \{\boldsymbol{\Phi} \in \mathbb{R}^{N \times K} : ||\boldsymbol{\phi}_k||_2 = 1 \quad \forall\, k\}$

of admissible dictionaries whose atoms have unit ℓ_2 norm. Many dictionary learning algorithms [1], [10], [6], [19], [8] that follow this approach have been proposed in the literature.

The sparse approximation (1) that is at the core of the sparse coding step of dictionary learning has been proved to be a NP hard problem [4], and a great number of sub-optimal algorithms [23], [13], [2], [11], [12] have been developed in order to tackle it. An important research effort has been devoted to understand whether these algorithms can approximate the true solution, that is, under which conditions they are able to retrieve a sparse linear combination of atoms (for example in a compressed sensing application), or they are able to optimally represent an arbitrary signal using a given number S of active atoms [21], [**?**].

The theorems that have been developed link the success of sparse approximation with the degree of similarity between different atoms, as expressed by the coherence of the dictionary.

B. Dictionary coherence and sparse recovery

The coherence of a dictionary indicates the degree of similarity between different atoms or different collections of atoms. The simpler measure proposed in the literature is the mutual coherence $\mu(\mathbf{\Phi})$, which is defined as the maximum absolute inner product between any two different atoms of the dictionary:

$$\mu(\mathbf{\Phi}) \stackrel{\text{def}}{=} \max_{i \neq j} \left| \left\langle \boldsymbol{\phi}_i, \boldsymbol{\phi}_j \right\rangle \right|$$

where we use the ordinary euclidean inner product for real vectors $\langle \mathbf{v}, \mathbf{w} \rangle \stackrel{\text{def}}{=} \sum_{n=1}^{N} v_n w_n$.

A generalisation of the mutual coherence is the p-comulative coherence $\mu_p(S, \mathbf{\Phi})$, which involves the sum of correlations between an atom $\boldsymbol{\phi}_i$ and an S-dimensional sub-dictionary that does not include it:

$$\mu_p(S, \mathbf{\Phi}) \stackrel{\text{def}}{=} \max_{i \notin J} \left\{ \max_{|J|=S} \left(\sum_{j \in J} \left| \left\langle \boldsymbol{\phi}_i, \boldsymbol{\phi}_j \right\rangle \right|^p \right)^{1/p} \right\} \tag{3}$$

where $p = \{1, 2\}$, $|J| = S$ indicates the cardinality of the set J and $\mu(\mathbf{\Phi}) = \mu_p(1, \mathbf{\Phi})$. For clarity of notation, we will from now on write coherence measures without explicitly indicating the dependence on the dictionary when unambiguous from the context.

Tropp [21] showed that, given a sparse signal generated according to the model (1), the orthogonal matching pursuit algorithm (OMP) [13] is guaranteed to retrieve the correct support of the representation coefficients only if $\mu < 1/(2S-1)$ or, more generally, if:

$$\mu_1(S) + \mu_1(S-1) < 1 \tag{4}$$

(see [21] for a proof of the above recovery bounds). Considering that the p-cumulative coherence is a positive, strictly increasing function of the number of active atoms S, $\mu_1(S)$ should be as small as possible in order to ensure the success of this sub-optimal algorithm.

Similar results have been reported for other methods: for example, Schnass and Vandergheynst [18] proved that the same cumulative coherence bound holds for the thresholding algorithm [17]. Unfortunately, they also proved that the 1-cumulative coherence is lower-bounded by:

$$\mu_1(S) \geq S\sqrt{\frac{K-N}{N(K-1)}}$$

which implies that only signals that are synthesised from a small number of atoms are guaranteed to be correctly recovered. As an illustrative example, a 100-dimensional signal generated as a sparse linear combination of 200 atoms, has a cumulative coherence $\mu_1(S) \geq 0.07S$, and the maximum number of active atoms that guarantees the success of the above sub-optimal algorithms is $S_{\max} \leq 7$.

The bound (4) is referred as a worst-case bound, and it is linked to the condition number of an arbitrary sub-dictionary of the matrix $\mathbf{\Phi}$ [22]. Less pessimistic results can be obtained by considering random sub-dictionaries, an insight that leads to average-case bounds expressed as the probability of success or failure of a sparse approximation algorithm [17]. These are linked with the function $\mu_2(S)$ and allow for less strict conditions on the maximum number of active atoms.

C. Learning incoherent dictionaries

The brief review of bounds for sparse recovery described so far highlights the importance of intrinsic properties of the dictionary, such as mutual coherence and p-cumulative coherence. While the matrix factorisation of a set of training data would not necessarily benefit from learning incoherent dictionaries, we can train a dictionary in a supervised fashion, optimising it on data of a given class in order to use it for the analysis of new data of the same sort. In this scenario, $\mathbf{\Phi}$ should be both well adapted to describe the signals and incoherent, so that existing sub-optimal algorithms are guaranteed to succeed in the sparse recovery problem.

Previous works attempting to join the approximation and incoherence objectives include a penalised optimisation [16] in which the dictionary update step is modified in order to minimise the coherence of the dictionary and a greedy algorithm [9], in which a decorrelation step is added after the dictionary update. We will review these methods in Section II.

In this paper we propose a novel technique which employes a decorrelation step inspired by a method used to construct Grassmannian frames [24]. Our main contributions are that we employ this technique

within the context of incoherent dictionary learning, as explained in Section III-A, and adapt it to the approximation objective through a rotation step, as described in Section III-B. Section IV presents numerical experiments on musical audio data (a class of signals for which we found traditional dictionary learning led to coherent dictionaries), and a comparison with the methods previously proposed in [16], [9]. Section V contains our conclusions and plans for further investigation.

II. Previous work

A. Method of optimal coherence-constrained directions (MOCOD)

Ramìrez et al. [16] proposed a dictionary learning algorithm inspired by the method of optimal directions (MOD) [6] in which the sparse approximation is performed using a novel penalty term derived from a probabilistic formulation of the sparse model (1), and the dictionary update step is modified in order to promote incoherent atoms.

In particular, the incoherence objective is pursued by introducing into the dictionary learning optimisation the term

$$||\mathbf{G} - \mathbf{I}||_{\mathrm{F}}$$

where each element g_{ij} of the Gram matrix $\mathbf{G} \stackrel{\mathrm{def}}{=} \mathbf{\Phi}^T\mathbf{\Phi}$ contains the inner product between the i-th and the j-th atom of the dictionary. The above expression measures the Frobenious distance between the Gram matrix of the dictionary and the identity matrix, which corresponds to the Gram matrix of an orthonormal dictionary whose mutual coherence is zero.

Overall, the optimisation presented in [16] reads as:

$$\begin{aligned}(\hat{\mathbf{\Phi}}, \hat{\mathbf{X}}) = \underset{\mathbf{\Phi},\mathbf{X}}{\arg\min} \quad ||\mathbf{Y} - \mathbf{\Phi}\mathbf{X}||_{\mathrm{F}}^2 + \tau \sum_{k=1}^{K}\sum_{m=1}^{M} \log(|x_{km}| + \beta) + \\ + \zeta\, ||\mathbf{G} - \mathbf{I}||_{\mathrm{F}}^2 + \eta \sum_{k=1}^{K} \left(||\boldsymbol{\phi}_k||_2^2 - 1 \right)^2\end{aligned} \tag{5}$$

In this unconstrained minimisation, the first term represents the modelling error, while the desired properties of dictionary and representation coefficients are enforced through penalty terms. In particular, the factor multiplied by τ promotes sparsity of the representation coefficients, while the factors multiplied by ζ and η promote incoherence and unit norm of the dictionary atoms respectively.

In order to solve this optimisation, the sparse approximation is followed by a MOCOD dictionary update step, derived by setting to zero the derivative of the above cost function with respect to the dictionary

$\mathbf{\Phi}$. The resulting update can be written as:

$$\mathbf{\Phi}' = \left(\mathbf{Y}\mathbf{X}^T + 2(\zeta+\eta)\mathbf{\Phi}\right)\left[\mathbf{X}\mathbf{X}^T + 2\zeta\mathbf{G} + 2\eta\,\mathrm{diag}(\mathbf{G})\right]^{-1}.$$

Note that setting to zero the penalty factors ζ and η results in the MOD update derived in [6].

B. Dictionary decorrelation and INK-SVD

An alternative strategy for learning incoherent dictionaries can be pursued by including a decorrelation step to the iterative scheme illustrated in Section I. At each iteration of the dictionary learning algorithm consisting of sparse approximation followed by dictionary update, we add the following optimisation problem:

$$\begin{aligned} \hat{\mathbf{\Phi}} = \underset{\mathbf{\Phi}\in\mathcal{D}}{\arg\min} \quad & \mathcal{C}(\mathbf{\Phi}) \\ \text{such that} \quad & \mu(\mathbf{\Phi}) \leq \mu_0 \end{aligned} \tag{6}$$

where the objective $\mathcal{C}(\mathbf{\Phi})$ is a cost function that describes the approximation quality of the dictionary. Mailhé et al. [9] proposed a matrix nearness problem where

$$\mathcal{C}(\mathbf{\Phi}) = \left|\left|\bar{\mathbf{\Phi}} - \mathbf{\Phi}\right|\right|_{\mathrm{F}} \tag{7}$$

and $\bar{\mathbf{\Phi}}$ is the matrix returned by the dictionary step, which translates as finding the closest dictionary (in a Frobenious norm sense) to a given dictionary subject to a mutual coherence constraint. In order to tackle this optimisation, the authors propose an iterative algorithm which consists of identifying a sub-dictionary of highly correlated atoms and decorrelating pairs of atoms in a greedy fashion, until the desired mutual coherence is achieved. This technique was used in conjunction with the K-SVD algorithm [1] and is called incoherent K-SVD (INK-SVD) dictionary learning.

The choice of the cost function (7) is not optimal in that it does not explicitly measure the approximation accuracy, but it rather implicitly assumes that dictionaries that are close to each other are well suited to represent the same set of data. In contrast, we use in the present work the cost function $\mathcal{C}(\mathbf{\Phi}) = ||\mathbf{Y} - \mathbf{\Phi}\mathbf{X}||_{\mathrm{F}}$ that measures the Frobenious norm of the residual.

C. Dictionary preconditioning

Apart from incoherent dictionary learning algorithms, Schnass and Vandergheynst presented in [18] a method for dictionary preconditioning that aims at tackling the problem of coherent dictionaries. In

this work, a *sensing* matrix is multiplied to a coherent dictionary in order to obtain an equivalent sparse approximation problem with low cross-cumulative coherence, and improve the performance of greedy sparse approximation algorithms. Although related to the present work, we choose not to further detail this algorithm for the sake of brevity.

III. Iterative projections and Rotations algorithm

In the present work, we seek the solution to the following optimisation problem:

$$\begin{aligned} \hat{\mathbf{\Phi}} = \underset{\mathbf{\Phi}\in\mathcal{D}}{\arg\min} \quad & ||\mathbf{Y} - \mathbf{\Phi}\mathbf{X}||_{\mathrm{F}} \\ \text{such that} \quad & \mu(\mathbf{\Phi}) \leq \mu_0 \\ & ||\mathbf{x}_m||_0 \leq S \quad \forall m \end{aligned} \tag{8}$$

For this purpose, after performing a sparse approximation that satisfies the sparsity constraint and a dictionary update, we employ a dictionary decorrelation consisting of two iterative steps:

- *Dictionary decorrelation*: obtained through the iterative projection algorithm described in Section III-A, this step ensures that the incoherence constraint is satisfied.
- *Dictionary rotation*: obtained through a Lie algebra optimisation strategy described in Section III-B, this step optimises the dictionary with respect to the objective function (8) without affecting its mutual coherence.

A. Constructing Grassmannian Frames with Iterative Projections

A Grassmannian frame is a collection of atoms that have unit norm and minimal mutual coherence. It can be proved that, for an $N \times K$ dictionary,

$$\mu \geq \sqrt{\frac{K-N}{N(K-1)}}$$

and the lower bound is reached when the dictionary is an equiangular tight frame, that is, a Grassmannian frame where any pair of different atoms have the same absolute inner product [20].
Constructing Grassmannian frames is an open research problem for which there is generally no analytic solution. One possible approach is to use an iterative projection method, as presented in [24].

1) Iterative projections algorithm: To illustrate this algorithm, we define two constraint sets, namely the spectral constraint set $\mathcal{F}$ as the set of symmetric positive semidefinite square matrices with rank

smaller than or equal to N:

$$\mathcal{F} \stackrel{\text{def}}{=} \left\{ \mathbf{F} \in \mathbb{R}^{K \times K} : \mathbf{F} = \mathbf{F}^T, \text{eig}(\mathbf{F}) \geq \mathbf{0}, \text{rank}(\mathbf{F}) \leq N \right\}$$

and the structural constraint set $\mathcal{K}_{\mu_0}$ as the set of symmetric square matrices with unit diagonal and off-diagonal values with magnitude smaller or equal than μ_0:

$$\mathcal{K}_{\mu_0} \stackrel{\text{def}}{=} \left\{ \mathbf{K} \in \mathbb{R}^{K \times K} : \mathbf{K} = \mathbf{K}^T, \text{diag}(\mathbf{K}) = \mathbf{1}, \max_{i \neq j} |k_{i,j}| \leq \mu_0 \right\}.$$

In the above expressions, the operators $\text{eig}(\cdot)$ and $\text{diag}(\cdot)$ return the vector of eigenvalues and the vector of diagonal elements of their arguments respectively.

The iterative projection algorithm starts from an initial dictionary $\mathbf{\Phi}$, calculates its Gram matrix $\mathbf{G}$, and iteratively projects it onto the sets $\mathcal{F}$ and $\mathcal{K}_{\mu_0}$ for a given number of iteration, or until a stopping criterion is met.

- *Projection onto the spectral constraint set*. Given an arbitrary dictionary $\mathbf{\Phi}$, its Gram matrix is by definition a symmetric, positive semidefinite matrix. Its projection $\mathbf{G}'$ onto the spectral constraint set $\mathcal{F}$ can be obtained through the following steps:
  1) Calculate the eigenvalue decomposition (EVD) $\mathbf{G} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^T$
  2) Threshold the eigenvalues, so that the smallest $K - N$ ones are set to zero, and therefore $\text{rank}(\mathbf{G}') \leq N$.
  3) Update the Gram matrix as $\mathbf{G}' = \mathbf{Q}\,\text{Thresh}(\mathbf{\Lambda}, N)\mathbf{Q}^T$.
- *Projection onto the structural constraint set*. Given an arbitrary Gram matrix $\mathbf{G}$, its projection $\mathbf{G}'$ onto the structural constraint set can be obtained by setting its diagonal values to one and by thresholding the magnitude of its off-diagonal values:
  1) Set $\text{diag}(\mathbf{G}') = \mathbf{1}$
  2) Limit the off-diagonal elements so that, for $i \neq j$,

$$g'_{i,j} = \text{Limit}(g_{i,j}, \mu_0) = \begin{cases} g_{i,j} & \text{if} \quad |g_{i,j}| \leq \mu_0 \\ \mu_0 & \text{if} \quad g_{i,j} > \mu_0 \\ -\mu_0 & \text{if} \quad g_{i,j} < -\mu_0 \end{cases}$$

- *Factorization of the Gram matrix*. The updated Gram matrix is factorized as the product

$$\mathbf{G} = \mathbf{\Phi}^T \mathbf{\Phi} \tag{9}$$

through the following steps:

1) Calculate the EVD decomposition $\mathbf{G} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^T$
2) Set $\mathbf{\Phi} = \mathrm{Thresh}(\mathbf{\Lambda}, N)^{\frac{1}{2}}\mathbf{Q}^T$.

so that $\mathbf{\Phi}^T\mathbf{\Phi} = \mathbf{Q}\,\mathrm{Thresh}(\mathbf{\Lambda}, N)\mathbf{Q}^T$. Note that this last step is not guaranteed to produce a dictionary with bounded coherence, hence the iterative nature of the method. Algorithm 1 summarises the operations described so far.

Algorithm 1 Iterative Projections: $\mathbf{\Phi} = \textsc{ip}(\mathbf{\Phi}, \mu_0, \mathrm{nIter})$

Require: $\mathbf{\Phi}, \mu_0, \mathrm{nIter}$
  iIter $\leftarrow 1$
  while iIter $\leq$ nIter **and** $\mu(\mathbf{\Phi}) > \mu_0$ **do**
    {Calculate Gram matrix}
    $\mathbf{G} \leftarrow \mathbf{\Phi}^T\mathbf{\Phi}$
    {Project onto spectral constraint set}
    $[\mathbf{Q}, \mathbf{\Lambda}] \leftarrow \textsc{evd}(\mathbf{G})$
    $\mathbf{\Lambda} \leftarrow \mathrm{Thresh}(\mathbf{\Lambda}, N)$
    $\mathbf{G} \leftarrow \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^T$
    {Project onto structural constraint set}
    $\mathrm{diag}(\mathbf{G}) \leftarrow \mathbf{1}$
    $\mathbf{G} \leftarrow \mathrm{Limit}(\mathbf{G}, \mu_0)$
    {Factorise}
    $[\mathbf{Q}, \mathbf{\Lambda}] \leftarrow \textsc{evd}(\mathbf{G})$
    $\mathbf{\Lambda} \leftarrow \mathrm{Thresh}(\mathbf{\Lambda}, N)$
    $\mathbf{\Phi} \leftarrow \mathbf{\Lambda}^{1/2}\mathbf{Q}^T$
    iIter $\leftarrow$ iIter $+1$
  end while

2) Employing iterative projections as a dictionary decorrelation step: We can use the iterative projection algorithm illustrated so far to decorrealte a dictionary starting from the matrix returned by the dictionary update step. However, optimising the Gram matrix with the only objective being reducing the mutual coherence means that the decomposition (9) is likely to lead to an updated dictionary that exhibits a poor approximation performance. To resolve this issue, we employ a dictionary rotation which does not modify the mutual coherence and that is optimised for the dictionary learning objective (8).

B. Dictionary Rotation using Lie Group Optimisation

The decomposition (9) is not unique, since for any orthogonal matrix $\mathbf{W}$ we obtain:

$$(\mathbf{W}\mathbf{\Phi}^T)(\mathbf{W}\mathbf{\Phi}) = \mathbf{\Phi}^T\mathbf{W}^T\mathbf{W}\mathbf{\Phi} = \mathbf{\Phi}^T\mathbf{\Phi} = \mathbf{G}.$$

Therefore, it is possible to apply an orthogonal matrix to the dictionary obtained from each iteration of the iterative projection algorithm in order to minimise the residual norm expressed in (8). The resulting optimisation problem can be expressed as follows:

$$\hat{\mathbf{W}} = \underset{\mathbf{W}\in\mathcal{O}(N)}{\arg\min} \quad ||\mathbf{Y} - \mathbf{W}\mathbf{\Phi}\mathbf{X}||_{\mathrm{F}} \tag{10}$$

where $\mathcal{O}(N)$ is the set of $N \times N$ orthogonal matrices. This optimisation is similar to a problem encountered for non-negative independent component analysis (NN-ICA), making it possible to borrow methods employed in that field for our purpose. We refer the interested reader to [14] for an exhaustive explanation of NN-ICA and the relative optimisation techniques. Here, we limit our discussion to the one employed in our algorithm, namely a conjugate gradient optimisation constrained to the $\mathcal{SO}(N)$ manifold of special orthogonal matrices with positive determinant.

1) Constrained optimisation in the $\mathcal{SO}(N)$ manifold: The set $\mathcal{O}(N)$ is a manifold embedded in the space of general $N \times N$ matrices. If we associate to this set the matrix multiplication operation, we obtain a *group*, which is defined as an algebraic structure consisting of a set together with an operation which satisfies the following properties:

1) Closure under the operation: the multiplication of any two orthogonal matrices returns an orthogonal matrix.
2) Associativity: matrix multiplication is associative. Given the matrices $\mathbf{A}, \mathbf{B}$ and $\mathbf{C}$, the equality $\mathbf{ABC} = (\mathbf{AB})\mathbf{C} = \mathbf{A}(\mathbf{BC})$ holds.
3) Existence of an identity element: the orthogonal identity matrix $\mathbf{I}$ maps any matrix $\mathbf{A}$ to itself $\mathbf{IA} = \mathbf{A}$.
4) Existence of an inverse element: the set includes, for every element $\mathbf{W} \in \mathcal{O}(N)$, an inverse element $\mathbf{W}^{-1} \in \mathcal{O}(N)$, such that $\mathbf{W}^{-1}\mathbf{W} = \mathbf{I}$. For orthogonal matrices, $\mathbf{W}^{-1} = \mathbf{W}^{T}$.

It has been proved that the group described so far is a disconnected Lie group, which loosely means that we can associate a system of coordinates, as in a vector space $\mathbb{R}^{N\times N}$, to a local region of the manifold (much like two-dimensional cartographic maps are associated with local regions of the earth), but that we can only move smoothly from one point to another in the manifold if these do not belong to *disconnected* regions [14]. We would rather consider *connected* Lie groups, where this complication does not occur and we can move around the manifold in every direction. The subset $\mathcal{SO}(N) \subset \mathcal{O}(N)$ of orthogonal matrices with determinant equal to one, with the matrix multiplication operation, is a connected Lie group. Therefore, we choose to modify the problem (10) by imposing the constraint $\mathbf{W} \in \mathcal{SO}(N)$. This

results in a *rotation* of the dictionary expressed by the following:

$$\hat{\mathbf{W}} = \underset{\mathbf{W}\in\mathcal{SO}(N)}{\arg\min} \quad ||\mathbf{Y} - \mathbf{W}\mathbf{\Phi}\mathbf{X}||_{\mathrm{F}}\,. \tag{11}$$

In order to solve (11), one option is to choose an update that is locally tangent to the manifold (by exploiting the local isomorphism between the manifold and the relative vector space, as in the cartographic analogy) and then to project back the updated matrix onto the manifold $\mathcal{SO}(N)$ [5]. However, we found that this method exhibited slow convergence in our experiments. Instead, we perform the optimisation in a Lie algebra associated to the constraint manifold.

2) Conjugate gradient descent in the Lie algebra $\mathfrak{so}(N)$: A Lie algebra is a vector space with an associated binary operation called Lie bracket (see [14] for a more detailed exposition). It can be shown that the space of skew-symmetric matrices, that is, any matrix $\mathbf{B}$ that satisfies $\mathbf{B} = -\mathbf{B}^T$, with the matrix commutator operation $[\mathbf{A}, \mathbf{B}] = \mathbf{AB} - \mathbf{BA}$ is a Lie algebra associated with the constraint manifold $\mathcal{SO}(N)$, and we denote it by $\mathfrak{so}(N)$.

Moreover, any element belonging to this Lie algebra can be mapped into an element belonging to the Lie group $\mathcal{SO}(N)$ by a matrix exponential (and vice versa using the matrix logarithm). That is, for every $\mathbf{B} \in \mathfrak{so}(N)$, $\exp(\mathbf{B}) \in \mathcal{SO}(N)$.

A Lie group method [7] can be used to optimise a cost function working in the Lie algebra while satisfying the manifold constraint. Its steps can be summarised as follows:

1) Start from a matrix $\mathbf{B} = \log(\mathbf{W}) \in \mathfrak{so}(N)$, for example from the zero matrix, that corresponds to the matrix logarithm of the identity $\mathbf{0} = \log(\mathbf{I}) \in \mathfrak{so}(N)$.
2) Find an update $\Delta\mathbf{B}$ that improves the cost function and move in the Lie algebra to an updated $\mathbf{B}' = \mathbf{B} + \Delta\mathbf{B}$.
3) Map the updated matrix onto the constraint manifold $\mathbf{V} = \exp(\mathbf{B}') \in \mathcal{SO}(N)$.
4) Calculate $\mathbf{W}' = \mathbf{VW} \in \mathcal{SO}(N)$.

It is possible to perform steps 2 to 4 iteratively by using the method of *parallel transport* (again, the interested reader can find more detailed information in [14] and references therein), which allows us to work in the Lie algebra $\mathfrak{so}(N)$ and use any of the tools developed for numerical optimisation in vector spaces. In our proposed algorithm, we employ a conjugate gradient optimisation that consists of the following steps at each iteration i:

1) Calculate the unconstrained gradient of the cost function $\mathcal{C}(\mathbf{W}) = \frac{1}{2}||\mathbf{Y} - \mathbf{W}\mathbf{\Phi}\mathbf{X}||_{\mathrm{F}}^2$.

$$(\bigtriangledown_{\mathbf{W}}\mathcal{C})^{(i)} = (\mathbf{W}^{(i)}\mathbf{\Phi}\mathbf{X} - \mathbf{Y})(\mathbf{\Phi}\mathbf{X})^T$$

2) Map the gradient to the Lie algebra, obtaining

$$\mathbf{R}^{(i)} = 2\,\mathrm{skew}\left[\left(\bigtriangledown_{\mathbf{W}}\mathcal{C}\right)^{(i)}\left(\mathbf{W}^{(i)}\right)^{T}\right]$$

where $\mathrm{skew}(\mathbf{A}) = \frac{1}{2}(\mathbf{A} - \mathbf{A}^T)$ is the skew-symmetric component of the matrix $\mathbf{A}$.

3) Find a conjugate search direction in the Lie algebra as:

$$\mathbf{H}^{(i)} = -\mathbf{R}^{(i)} + \gamma\mathbf{H}^{(i-1)}$$

where $\gamma = \frac{\left\langle \mathbf{R}^{(i)}, \mathbf{R}^{(i)} - \mathbf{R}^{(i-1)} \right\rangle}{\left\langle \mathbf{R}^{(i-1)}, \mathbf{R}^{(i-1)} \right\rangle}$ is the Polak-Ribière formula and $\langle \mathbf{A}, \mathbf{B} \rangle = \mathrm{Tr}[\mathbf{A}^T\mathbf{B}]$ indicates the matrix inner product.

4) Perform a line search in the direction $\mathbf{H}^{(i)}$ as:

$$\hat{t}^{(i)} = \underset{t\in\mathbb{R}}{\mathrm{argmin}}\ \mathcal{C}\left(\exp\left(t\mathbf{H}^{(i)}\right)\right)$$

5) Update the orthogonal matrix as:

$$\mathbf{W}^{(i+1)} = \exp\left(\hat{t}^{(i)}\mathbf{H}^{(i)}\right)\mathbf{W}^{(i)}$$

After the algorithm has converged to a solution of the problem (11), we obtain a decorrelated dictionary that is rotated to minimise the residual norm of the sparse approximation. We refer to this algorithm as iterative projections and rotations (IPR), and we summarise it in Algorithm 2.

IV. Numerical Experiments

We tested the proposed decorrelation method with the K-SVD dictionary learning algorithm in order to assess if it converges to a dictionary for sparse approximation that exhibits minimal coherence and good approximation quality. The test signal we used is a 16 kHz guitar recording that is part of the data included in SMALLBOX [3][1], a Matlab toolbox for testing and benchmarking dictionary learning algorithms that was used in our evaluation and that contains the code used to generate the results presented here. A musical audio signal was chosen because previous informal experiments resulted in K-SVD learning a highly coherent dictionary for this type of data.

We divided the recording into 50% overlapping blocks of 256 samples (corresponding to 16ms) with rectangular windows and arranged the resulting vectors as columns of the training data matrix $\mathbf{Y}$. Then, we initialised a twice over-complete dictionary for sparse representation using either a randomly chosen

[1]http://small-project.eu/software-data/smallbox

Algorithm 2 Iterative Projections and Rotations: $\mathbf{\Phi} = \text{IPR}(\mathbf{\Phi}, \mathbf{Y}, \mathbf{X}, \mu_0, \text{nIter}_{\text{IP}}, \text{nIter}_{\text{R}})$

Require: $\mathbf{\Phi}, \mathbf{Y}, \mathbf{X}, \mu_0, \text{nIter}_{\text{IP}}, \text{nIter}_{\text{R}}$
  $\text{iIter}_{\text{IP}} \leftarrow 1$
  while $\text{iIter}_{\text{IP}} \leq \text{nIter}_{\text{IP}}$ **and** $\mu(\mathbf{\Phi}) > \mu_0$ **do**
    {Perform one iteration of the iterative projections algorithm 1}
    $\mathbf{\Phi} \leftarrow \text{IP}(\mathbf{\Phi}, \mu_0, 1)$
    {Rotate dictionary}
    $\mathbf{W} \leftarrow \mathbf{I}$ {Initialise rotation matrix}
    $\mathbf{H} \leftarrow \mathbf{0}$ {Initialise search direction}
    for $\text{iIter}_{\text{R}} = 1 : \text{nIter}_{\text{R}}$ **do**
      {Find an update direction and step in the Lie algebra}
      $\bigtriangledown_{\mathbf{W}} \mathcal{C} \leftarrow (\mathbf{W}\mathbf{\Phi}\mathbf{X} - \mathbf{Y})(\mathbf{\Phi}\mathbf{X})^T$
      $\mathbf{R} \leftarrow 2\,\text{skew}\left[(\bigtriangledown_{\mathbf{W}} \mathcal{C})\,\mathbf{W}^T\right]$
      $\mathbf{H} \leftarrow -\mathbf{R} + \gamma \mathbf{H}$
      $t \leftarrow \underset{t \in \mathbb{R}}{\text{argmin}}\ \mathcal{C}\left(\exp\left(t\mathbf{H}\right)\right)$
      {Map the update to the constraint manifold}
      $\mathbf{W} \leftarrow \exp\left(t\mathbf{H}\right)\mathbf{W}$
    end for
    $\mathbf{\Phi} \leftarrow \mathbf{W}\mathbf{\Phi}$
    $\text{iIter} \leftarrow \text{iIter} + 1$
  end while

subset of the training data or an over-complete Gabor dictionary. We run the dictionary learning algorithms for 20 iterations, allowing for 12 non-zero coefficients in each representation (which corresponds to about 5% of active elements if compared with the ambient dimension N). When testing the algorithm proposed in [16], we used OMP as a sparse approximation step and MOCOD for the dictionary update. INK-SVD and IPR were implemented using OMP for the sparse approximation step and K-SVD for the dictionary update. Table I summarises the tested algorithms.

Algorithm (Reference)	Sparse Approximation	Dictionary Update	Dictionary Decorrelation
Ramìrez et al. [16]	OMP	MOCOD	-
Mailhé et al. [9]	OMP	K-SVD	INK-SVD
Proposed method	OMP	K-SVD	IPR

Table I
ALGORITHMS FOR LEARNING INCOHERENT DICTIONARIES

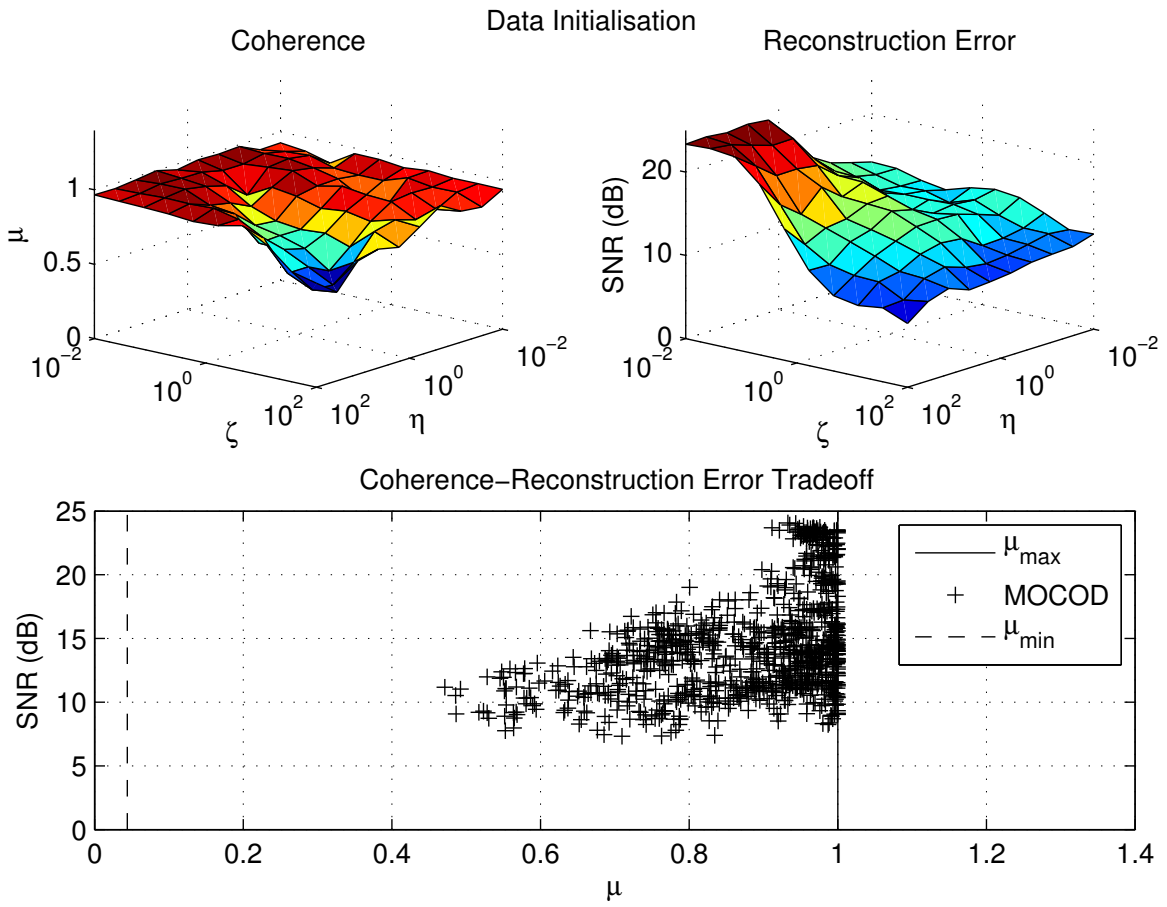


Figure 1. Mutual coherence and reconstruction error achieved using the MOCOD dictionary update and randomly chosen samples from the training set as the initial dictionary. In the lower plot, the levels $\mu_{max} = 1$ and $\mu_{min} = \sqrt{(K-N)/N(K-1)}$ indicate the maximum and minimum coherence attainable by a $N \times K$ dictionary.

A. MOCOD *updates*

The unconstrained optimisation illustrated in (5) relies on the penalty factors ζ and η in order to promote incoherence of the dictionary and unit norm of the atoms respectively. In order to evaluate the MOCOD dictionary update for the purpose of incoherent dictionary learning, we tested different values of these factors on a logarithmic scale between 10^{-2} and 10^{2}, assessing the resulting coherence and signal-to-noise ratio achieved by the optimised dictionary. Figures 1 and 2 depict the results of our experiment using respectively randomly chosen data from the training set and a twice over-complete Gabor dictionary for the initialisation. We run the experiment 10 times to increase the significance of our results whenever the initialisation involved a random element.

When $\zeta \to 0$ and $\eta \to \infty$, the optimisation (5) converges to a traditional dictionary learning where

the atoms are not forced to be incoherent, but are constrained to be unit norm. This case corresponds to the left corner of the surf plots in Figures 1 and 2.

We can note that a data initialisation produces a highly coherent dictionary with the best representation quality, while a Gabor initialisation results in a lower coherence at the expense of a worse SNR. Continuing our analysis in the case of data initialisation, keeping $\eta \to \infty$ and increasing the coherence penalty factor ζ results in a dictionary with lower mutual coherence, but also in a worse approximation quality. This behaviour is further illustrated by the coherence-reconstruction scatter plot, which depicts μ against SNR of the sparse approximation for every learned dictionary and exhibits a clear (although highly variable) trend. In the case of Gabor initialisation, on the other hand, it seems that the parameter ζ does not affect coherence and reconstruction error for high values of η, while decreasing the penalty factor η has a negative effect on both μ and SNR of the learned dictionaries.

B. IPR *and* INK-SVD

Unlike MOCOD, INK-SVD and the proposed IPR algorithm allows us to set a target coherence μ_0 and to run the dictionary decorrelation iteratively until it is achieved. We therefore set the target in equally spaced intervals from 0.1 to 1 and compared the two algorithms by evaluating the achieved SNR. Again, when applying the methods to an initial dictionary formed by randomly selected vectors from the training set, we run the experiment for 10 independent trials to obtain more significant results.

Figures 3 and 4 depict the results of our experiment. As can be noted, both IPR and INK-SVD achieve the target coherence level for both initialisations, except when μ_0 is bigger than the coherence level achieved without dictionary decorrelation (in which case, the two algorithms simply act as a K-SVD dictionary learning without any coherence constraint). In the case of data initialisation, we can observe that INK-SVD obtains a good SNR for mutual coherence values greater than $\mu = 0.6$, after that its performance degrades substantially. On the contrary, the proposed IPR does not perform as well for high coherence values, but does not significantly degrade from $\mu = 0.5$ to $\mu = 0.1$, making it the best choice whenever we need a very incoherent dictionary.

The results for Gabor initialisation, on the other hand, favour the proposed algorithm showing a better SNR and no significant approximation degradation for all the target coherence values.

V. CONCLUSIONS AND PLANS FOR FUTURE INVESTIGATION

We presented the iterative projections and rotations (IPR) decorrelation algorithm, a method for dictionary decorrelation to be used within the context of dictionary learning. Our technique is based on an

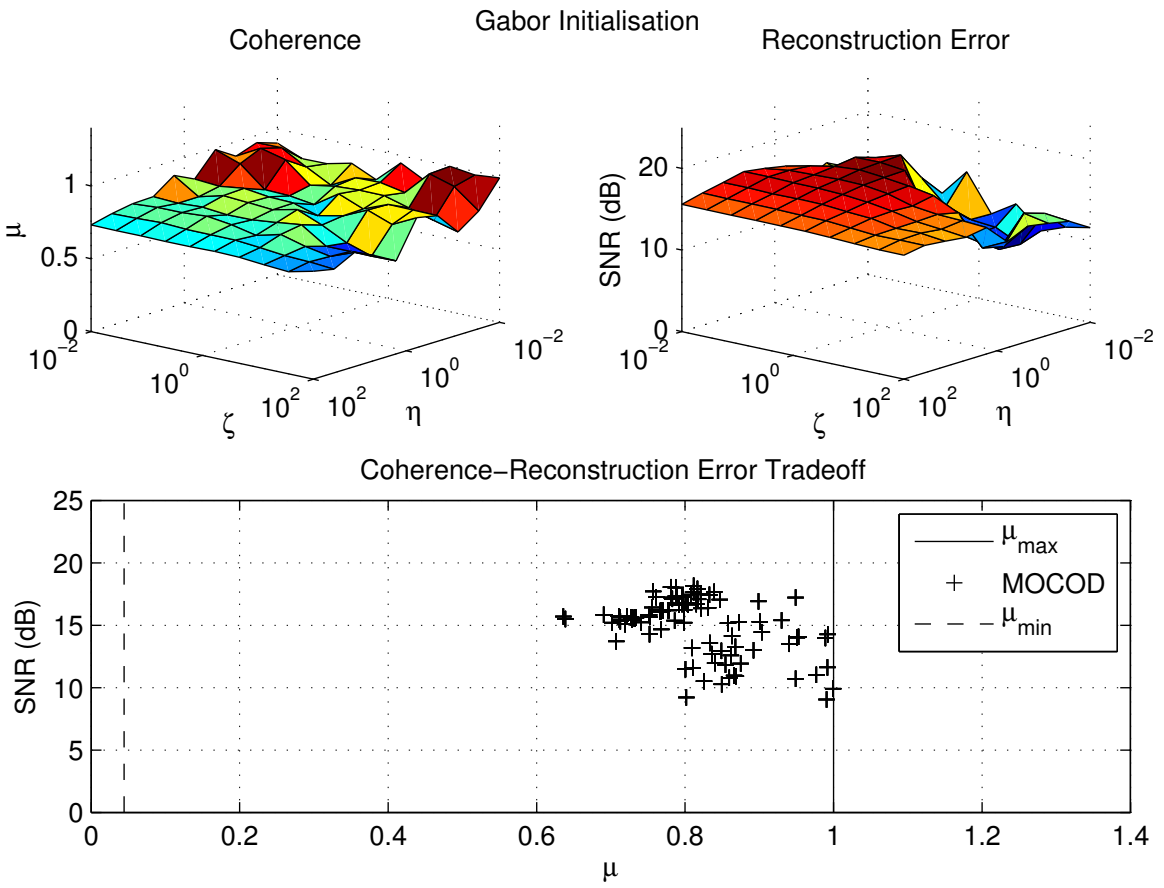


Figure 2. Mutual coherence and reconstruction error achieved using the MOCOD dictionary update and a twice over-complete Gabor initial dictionary. In the lower plot, the levels $\mu_{max} = 1$ and $\mu_{min} = \sqrt{(K-N)/N(K-1)}$ indicate the maximum and minimum coherence attainable by a $N \times K$ dictionary.

iterative projection optimisation used to construct Grassmannian frames and includes a dictionary rotation step that makes it suitable for the approximation objective (2) of dictionary learning, achieved using a Lie group optimisation. Experiments on musical audio data demonstrate the performance of IPR and suggest that it can outperform state-of-the-art algorithms when a very low mutual coherence is required and when applied to decorrelate an over-complete Gabor dictionary.

The main drawback of IPR is its high computational cost. Although this issue is mitigated by the fact that, in a supervised learning setting, the algorithm can be applied off-line in order to provide an optimal dictionary for the analysis of new data, additional investigation should be carried out to speed its computation, especially regarding the rotation step.

In addition, future research efforts can be devoted to the design of a decorrelation algorithm aimed at minimising the cumulative coherence of the dictionary, rather than the mutual coherence, in order to

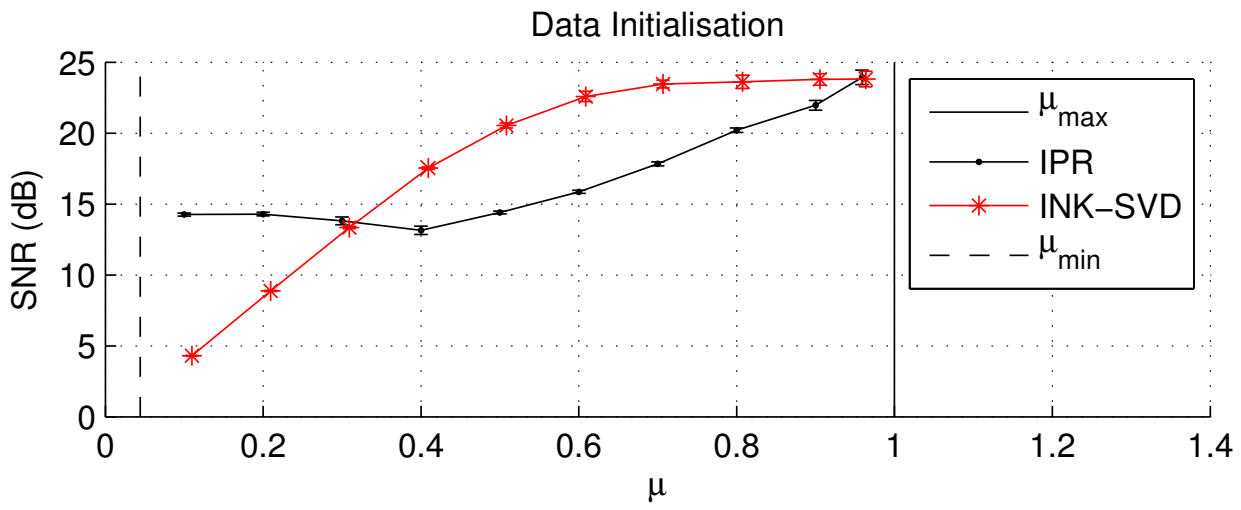


Figure 3. Mutual coherence and reconstruction error achieved using the proposed iterative projections and rotations (IPR) algorithm and INK-SVD dictionary decorrelation, initialised with randomly chosen samples from the training set as the initial dictionary. The levels $\mu_{max} = 1$ and $\mu_{min} = \sqrt{(K-N)/N(K-1)}$ indicate the maximum and minimum coherence attainable by a $N \times K$ dictionary. The error bars indicate the standard deviation resulting from 10 independent trials of the experiment and indicate that the results are consistent, regardless the random element introduced in the initialisation.

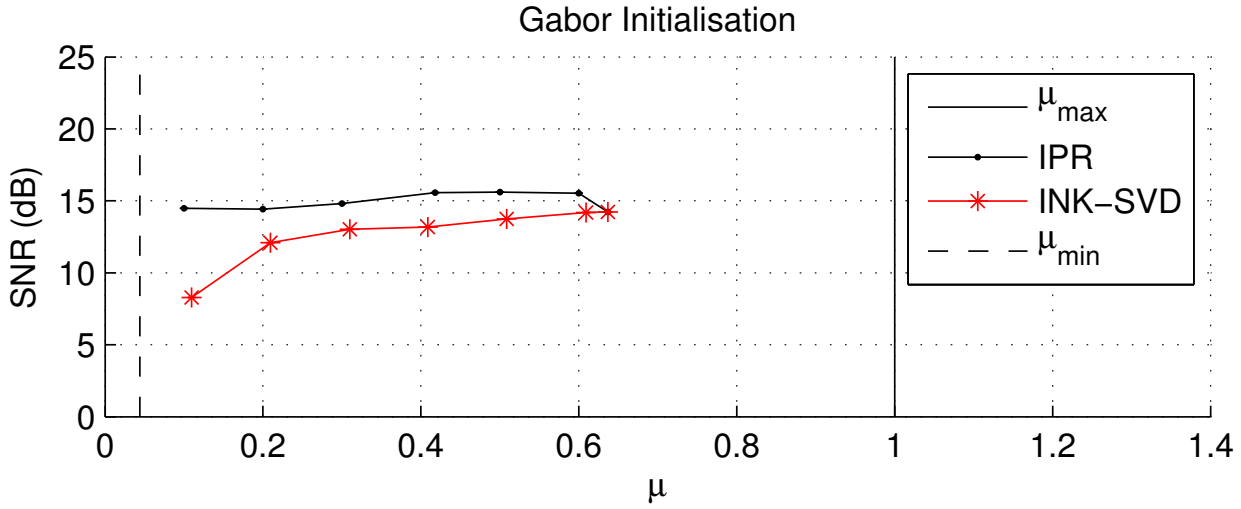


Figure 4. Mutual coherence and reconstruction error achieved using the proposed iterative projections and rotations (IPR) algorithm and INK-SVD dictionary decorrelation, initialised with a twice over-complete Gabor dictionary. The levels $\mu_{max} = 1$ and $\mu_{min} = \sqrt{(K-N)/N(K-1)}$ indicate the maximum and minimum coherence attainable by a $N \times K$ dictionary.

keep the pace with advances in the sparse recovery theory.

REFERENCES

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

[2] S. S. Chen, D. L. Donoho, and M. A. Sauders. Atomic decomposition by basis pursuit. *SIAM Review*, 43(1):129–159, Mar. 2001.

[3] I. Damnjanovic, M. E. P. Davies, and M. D. Plumbley. SMALLbox - An evaluation framework for sparse representations and dictionary learning algorithms. In *Proceedings of the International Conference on Latent Variable Analysis and Signal Separation (LVA/ICA)*, pages 418–425, 2010.

[4] G. Davis, S. Mallat, and M. Avellaneda. Adaptive greedy approximations. *Constructive Approximation*, 13(1):57–98, 1997.

[5] S. C. Douglas. Self-stabilized gradient algorithms for blind source separation with orthogonality constraints. *IEEE Trans on Neural Networks*, 11(6):1490–1497, Nov. 2000.

[6] K. Engan, S. O. Aase, and J. H. Husøy. Method of optimal directions for frame design. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 5, pages 2443–2446, 1999.

[7] A. Iseries, H. Munthe-Kaas, S. P. Nørsett, and A. Zanna. Lie-group methods. *Acta Numerica*, 9:215–365, 2000.

[8] M. Lewicki and T. Sejnowski. Learning overcomplete representations. *Neural Computation*, 12(2):337–365, Feb. 2000.

[9] B. Mailhé, D. Barchiesi, and M. D. Plumbley. INK-SVD: Learning incoherent dictionaries for sparse representations. To appear in Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2012).

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

[11] D. Needell and J. A. Tropp. CoSaMP: Iterative signal recovery from incomplete and inaccurate samples. *Applied and Computational Harmonic Analysis*, 26:301–321, 2008.

[12] D. Needell and R. Vershynin. Uniform uncertanty principle and signal recovery via regularized orthogonal matching pursuit. *Foundations of Computational Mathematics*, 9(3):317–334, 2009.

[13] Y. Pati, R. Rezaiifar, and P. Krishnaprasad. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In *Proceedings of the 27th Asilomar Confrence on Signals, Systems and Computers*, volume 1, pages 40–44, Nov. 1993.

[14] M. D. Plumbley. Geometrical methods for non-negative ICA: Manifolds, Lie groups and toral subalgebras. *Neurocomputing*, 67:161–197, 2005.

[15] R. Rubinstein, A. Bruckstein, and M. Elad. Dictionaries for sparse representation modeling. *Proceedings of the IEEE*, 98(6):1045–1057, Jun. 2010.

[16] G. Sapiro, I. Ramírez, and F. Lecumberry. Sparse modeling with universal priors and learned incoherent dictionaries. Technical Report 2279, Institute for Mathematics and its Applications, University of Minnesota, Sep. 2009.

[17] K. Schnass and P. Vandergheynst. Average performance analysis for thresholding. *IEEE Signal Processing Letters*, 14(11):828–831, Nov. 2007.

[18] K. Schnass and P. Vandergheynst. Dictionary preconditioning for greedy algorithms. *IEEE Trans. on Signal Processing*, 56(5):1994–2002, May 2008.

[19] K. Skretting and K. Engan. Recursive least squares dictionary learning algorithm. *IEEE Trans. on Signal Processing*, 58(4):2121–2130, Apr. 2010.

[20] T. Strohmer and R. W. J. Heath. Grassmannian frames with applications to coding and communication. *Applied and Computational Harmonic Analysis*, 14(3):257–275, 2003.

[21] J. A. Tropp. Greed is good: Algorithmic results for sparse approximation. *IEEE Trans. on Information Theory*, 50(10):2231–2242, Oct. 2004.

[22] J. A. Tropp. On the conditioning of random subdictionaries. *Applied and Computational Harmonic Analysis*, 25(1):1–24, Jul. 2008.

[23] J. A. Tropp. Computational methods for sparse solution of linear inverse problems. *Proceedings of the IEEE*, 98(6):948–958, Jun. 2010.

[24] J. A. Tropp, I. S. Dhillon, R. W. J. Heath, and T. Strohmer. Designing structured tight frames via an alternating projection method. *IEEE Trans. on Information Theory*, 51(1):188–209, Jan. 2005.