Incoherent Dictionary Learning via Mixed-integer Programming and Hybrid Augmented Lagrangian

Yuan Liu, Stéphane Canu, Paul Honeine, Su Ruan

Incoherent Dictionary Learning via Mixed-integer Programming and Hybrid Augmented Lagrangian

Yuan Liu, Stéphane Canu, Paul Honeine, Su Ruan

the Normandie Univ, INSA Rouen, UNIROUEN, UNIHAVRE, LITIS, Avenue de l'Université, 76801 Saint-Étienne-du-Rouvray

Abstract

During the past decade, the dictionary learning has been a hot topic in sparse representation. With theoretical guarantees, a low-coherence dictionary is demonstrated to optimize the sparsity and improve the accuracy of the performance of signal reconstruction. Two strategies have been investigated to learn incoherent dictionaries: (*i*) by adding a decorrelation step after the dictionary updating (*e.g.* INK-SVD), or (*ii*) by introducing an additive penalty term of the mutual coherence to the general dictionary learning problem. In this paper, we propose a third method, which learns an incoherent dictionary by solving a constrained quadratic programming problem. Therefore, we can learn a dictionary with a prior fixed coherence value, which cannot be realized by the second strategy. Moreover, it updates the dictionary by considering simultaneously the reconstruction error and the incoherence, and thus does not suffer from the performance reduction of the first strategy.

The constrained quadratic programming problem is difficult problem due to its non-smoothness and non-convexity. To deal with the problem, a two-step alternating method is used: sparse coding by solving a problem of mixed-integer programming and dictionary updating by the hybrid method of augmented Lagrangian and alternating proximal linearized minimization. Finally, extensive experiments conducted in image denoising demonstrate the relevance of the proposed method, and illustrate the relation between coherence of dictionary and reconstruction quality.

Keywords: Sparse Representation, Dictionary Learning, Incoherent Dictionary, Augmented Lagrangian Method, Alternating Proximal Method, Mixed-Integer Quadratic Programming (MIQP)

Email addresses: yuan.liu@insa-rouen.fr (Yuan Liu), stephane.canu@insa-rouen.fr (Stéphane Canu), paul.honeine@univ-rouen.fr (Paul Honeine), su.ruan@univ-rouen.fr (Su Ruan)

1. Introduction

Sparse representations have been successfully applied in signal and image processing, as well as computer vision tasks, such as image denoising, image inpainting, object recognition, face recognition and many classification tasks (see for instance [1] and included references). In the sparse representation paradigm, a signal is approximated by a linear combination of a few atoms of a dictionary. Early developments in sparse representations considered pre-defined dictionaries, such as wavelets and many variants [2, 3]. More recently, data-driven dictionary learning has been proposed, which allows to have well-adapted and more natural representations for the signals at hand. Moreover, dictionaries can be learned for some specific tasks. For example, to achieve good performance in classification, the dictionary learning problem is optimized by minimizing the classification error measured by Fisher criterion [4] or logistic regression loss [5].

Dictionary learning seeks to solve an optimization problem with sparsity-prompting functional on the coefficients of the sparse representation. This is an NP-hard problem. It is often relaxed and solved by alternating between two minimization steps: In the so-called sparse coding step, the coefficients are estimated with the dictionary fixed; In the so-called dictionary updating step, the dictionary with sparse codes fixed. Several dictionary learning algorithms have been proposed in the literature, the most known being the K-SVD algorithm [6]. However, these conventional algorithms do not guarantee the "good" quality of the obtained dictionary, neither the resulting sparse representation.

The coherence measure is a fundamental measure to characterize a dictionary, corresponding to the largest correlation between the elements of the dictionary (*e.g.* null coherence for dictionaries with orthogonal elements). Beyond being elementary and very simple to compute, the coherence is intimately related to the sparsity level and the relevance of the resulting sparse representation. Indeed, several theoretical studies have demonstrated the prominence of having incoherent dictionaries, namely dictionaries having a low coherence measure [7, 8, 9]. Incoherent dictionary learning, as an extension of generic dictionary learning, aims at minimizing the reconstruction error by imposing sparsity on the coefficient and coherence of the dictionary, simultaneously. For this purpose, several incoherent dictionary learning algorithms have been proposed, within two major strategies: either adding a decorrelation step after dictionary updating at each iteration, such as INK-SVD and related algorithms [10, 11], or introduced an additional regularization of the coherence in the optimization problem [12, 13, 14]. While the latter strategy may provide better performance, the former is often recommended because it allows the fix the coherence level beforehand. See next section for a survey.

In this paper, we examine the exact resolution of the incoherent dictionary learning problem, by considering explicitly the constraints on the coherence and the unit-norm of the dictionary elements. This NP-hard problem is much more difficult than the typical dictionary learning problem. To address this constrained optimization problem, we provide a two-step alternating approach, in the same spirit as the generic dictionary learning algorithms.

We address the sparse coding step in its exact ℓ_0-norm formulation. To this end, the optimization problem is recast as a mixed-integer program (MIP), namely involving both integer and continuous variables. While the use of MIP for pattern recognition is not new [15], only very recently it has been investigated with success to generate fiducial marker [16], to perform multiple face tracking [17] and vehicle detection [18]. In sparse representation, preliminary studies conducted in [19] were restricted to tiny toy data (120-sample synthesized signals) due to high computational complexity. By taking advantage of breakthrough in optimization theory, we have more recently demonstrated the relevance of MIP in exact dictionary learning for sparse representation of well-known images [20].

The dictionary updating step faces a minimization problem with convex objective function but non-convex constraints, due to the constraints on the coherence and the unit norm of the dictionary elements. To deal with the constrained optimization problem, a first attempt is to use the augmented Lagrangian method, with a penalty to assure the second order sufficient condition (See Chapters 3 and 4 in [21] for details). However, with the inequality constraint associated to the coherence, it is hard to solve the problem directly by satisfying the Karush-Kuhn-Tucker (KKT) optimality condition. Thus, in this paper, we propose a resolution based on the hybrid method of augmented Lagrangian and the extended proximal alternating linearized minimization (EPALM) [22, 23]. The choice of the latter is motivated by its convergence for a large class of non-convex problems [22]. In the appendix of the present paper, the global convergence of the algorithm is demonstrated based on the theoretical convergence analysis of Kurdyka-Łojasiewicz function [24].

Finally, the relevance of the proposed method is examined with extensive experiments, on synthetic and real well-known images. A comparative analysis with the state-of-the-art methods is conducted. For this purpose, we examine two versions of the proposed method depending on the sparse coding algorithm, one based on the MIQP and one using the proximal method. By examining the properties and reconstruction results, we show the relation between the coherence of dictionary and the image reconstruction accuracy. Moreover, for several values of the coherence parameter, we show that the proposed MIQP+EPALM dictionary learning method outperforms the other methods.

The main contributions of this work are summarized as follows:

- The problem of the incoherent dictionary is formulated as a constrained quadratic programming problem, by explicitly adding the constraint of the dictionary mutual coherence.

- The incoherent dictionary is learned via the generic alternating strategy. Specifically, for the problem of dictionary updating, we propose to use the augmented Lagrangian method to transform the problem into an unconstrained quadratic programming problem. Then, the dictionary with the target mutual coherence can be learned by the algorithm of EPALM after a finite number of iterations. Moreover, we give the convergence analysis of our proposed algorithm.

- The incoherent dictionary learning algorithm is applied in image reconstruction, which shows better performance compared with state-of-the-art algorithms. The obtained results verify that increasing the incoherence may have positive effect on the performance. These results corroborate theoretical results that has not been proved by the compared algorithm.

The rest of the paper is organized as follows. In next section, related works are presented, with a focus on incoherent dictionary learning algorithms. The sparse representation problem with coherence constraints is introduced in Section 3. The proposed method is described in detail in Section 4, with convergence analysis given in the appendix. Section 5 provides extensive experiments for image reconstruction.

2. Related Works

Incoherent dictionary learning algorithms are designed via two principal strategies.

In the first strategy, a dictionary with low coherence is learned by adding a decorrelation step after the dictionary update step at each iteration. The leading method is INK-SVD [10], developed from the well-known K-SVD algorithm. INK-SVD seeks to minimize the approximation quality with a constraint of coherence level. To find the optimal solution, an iterative algorithm is proposed by identifying the sub-dictionary (in the same spirit as K-SVD) and decorrelating pairs of atoms with a greedy algorithm. Barchiesi et al [11] optimize the INK-SVD algorithm by considering simultaneously the minimization of the residual error of sparse approximation when learning the dictionary with a fixed target coherence level. While the employed algorithm introduces a decorrelation step after dictionary updating as in INK-SVD, the decorrelation step is accomplished by an iterative projection followed by a rotation of dictionary. In this paper, this method is denoted IPR (for iterative projections and rotations). These incoherent dictionary learning algorithms based on this strategy (*i.e.*, by adding a decorrelation step) allow to fix the coherence level beforehand, thus evaluate explicitly the relationship between the reconstruction performance and the coherence of dictionary.

Methods from the second strategy seek to learn an incoherent dictionary by minimizing a regularized objective function, where the regularization term constrains the coherence. The method of optimal coherence-constraint directions (MOCOD) [25], inspired from the method of optimal direction (MOD), introduces the regularization term of coherence and unit norm of dictionary elements. The MOCOD method outperforms the MOD method in image reconstruction. In [13], the incoherent dictionary learning problem is formulated by introducing only the coherence regularization, namely, the Frobenius norm of the difference of Gram matrix and identity. The incoherent dictionary is learned via a hybrid alternating proximal method, and the dictionary is normalized after dictionary updating at each iteration. Similarly, Abolghasemi et al [26] tackled the problem with the same coherence regularization as in [13]. However, they proposed an

incoherent dictionary learning algorithm with dictionary updating by a gradient descent method. In [27], another incoherence penalty was introduced to learn a discriminative dictionary, where it exploited the alternating algorithm with the dictionary updated by the method of alternating direction method of multipliers (ADMM). In addition, the coherence regularization was also measured by the sum of ℓ_1-norm of every two different atoms [28, 29]. Even, for some task (such as classification), the Fisher criterion [30] and the weighted auto-correlation between atoms [31] can be regarded as a coherence regularization, which makes the sub-dictionary of different class coherent. Incoherent dictionary learning algorithms of the second strategy achieve good performance in data reconstruction [26, 29], classification [13] and object recognition [30, 32]. However, they suffer from a major issue: it is not possible to constraint exactly the coherence level to a fixed value, because the relation between it and the regularization trade-off parameter is unknown.

In this paper, we consider the simplest way to formulate the problem, by adding the constraints of coherence and a unit norm of the dictionary elements into the generic dictionary learning problem. Hence, the resulting incoherent dictionary learning problem is the minimization of a quadratic objective function with a quadratic inequality constraint. It is noted that the problem is non-convex and non-smooth because of the sparsity-prompting ℓ_0-norm and the constraints. To the best of our knowledge, there is no work on incoherent dictionary learning by solving the problem with explicit constraints on dictionary coherence and its unit norm. To solve this constrained optimization problem, we take advantage of recent developments in optimization problem with orthogonality constraints, with the augmented Lagrangian method and the alternating proximal minimization method. A review of these methods is given in the following.

The optimization problem with orthogonality constraints has been recently addressed in physics [33], mathematics [34] and information science [35]. The Lagrangian multiplier method [21] is frequently used to deal with such a problem [34, 33]. However, it is not always easy to solve the Lagrangian function by satisfying the first order optimal condition. In [33], the Kohn-Sham problem was reformulated by the Lagrangian multiplier method, and the proximal gradient method was then proposed to solve the Lagrange function. Moreover, it was proven that the algorithm has good convergence property. Orthogonality constrained optimization problems were also solved via the augmented Lagrangian method [35, 22]. Compared with the Lagrangian method, the penalty method shows more stability [21]. However, the reformulated problem can be non-convex and non-smooth, which makes the problem hard to tackle. In [35], the alternating proximal method was combined with the augmented Lagrangian method and the existence of the sub-sequence to a KKT point was proven. The new proposed algorithm was then applied in compressed mode for variational problems in physics, illustrating the effectiveness and efficiency of the algorithm. In [22], an extended proximal alternating linearized minimization method was introduced to solve the Lagrangian function, and its convergence was proven based on the theory of the Kurdyka-Łojasiewicz inequality property [24].

3. Problem Statement

Given a matrix $Y = [\mathbf{y}_1, \ldots, \mathbf{y}_i, \ldots, \mathbf{y}_\ell] \in \mathbb{R}^{n \times \ell}$ of ℓ signals of dimension n, a sparse representation of Y consists in a decomposition of the form $Y = DX$, where the matrix $X = [\mathbf{x}_1, \ldots, \mathbf{x}_\ell] \in \mathbb{R}^{p \times \ell}$ containing the decomposition coefficients is sparse, and the matrix $D = [\mathbf{d}_1, \ldots, \mathbf{d}_p] \in \mathbb{R}^{n \times p}$ is the dictionary with each column called atom. We consider in this paper the overcomplete dictionary learning problem, namely $n < p$. The optimization problem is written as

$$\begin{aligned} \min_{D \in \mathcal{C}, \mathbf{x}_i \in \mathbb{R}^p} \quad & \frac{1}{\ell} \sum_{i=1}^{\ell} \tfrac{1}{2} \|\mathbf{y}_i - D\mathbf{x}_i\|_2^2 \\ \text{subject to} \quad & \|\mathbf{x}_i\|_0 \leq T, \quad i = 1, \ldots, \ell, \end{aligned} \tag{1}$$

where the cost function $\frac{1}{2}\|\mathbf{y}_i - D\mathbf{x}_i\|_2^2$ is the reconstruction error with $\| \, . \, \|_2$ being the Euclidean norm. The sparsity of each $\mathbf{x}_i$, measured with its quasi-norm $\|\mathbf{x}_i\|_0$ that refers to the number of non zero elements in $\mathbf{x}_i$, is constrained by the preset sparsity parameter T. In image denoising, T varies with the noise level σ, such as in [6] where T is determined with the constraint $\|\mathbf{y}_i - D\mathbf{x}_i\|_2^2 \leq \epsilon(\sigma)$ where $\epsilon(\sigma)$ is a function of σ. The dictionary D is restricted in the constraint

$$\mathcal{C} = \left\{ D \in \mathbb{R}^{n \times p} \;\middle|\; \mathbf{d}_j^T \mathbf{d}_j = 1, \; \forall j = 1, \ldots, p \right\},$$

in order to prevent the ℓ_2-norm of dictionary's atoms from being arbitrarily large, which leads to arbitrarily small decomposition coefficients in X.

The optimization problem (1) is addressed by using a relaxation procedure that alternates two phases. The first, called sparse coding, seeks to estimate X while the dictionary D is fixed. Because of the ℓ_0 norm, this optimization problem is non-convex and NP-hard. To overcome this difficulty, most of the work in this field operate a relaxation, by substituting the ℓ_0 norm with a convex one such as the ℓ_1-norm [36], or use a greedy approximate algorithm [37]. We have recently shown that it is possible to have an exact resolution, by the integer quadratic integer programming (MIQP) [20]. The second phase, called dictionary update, seeks to estimate the dictionary D while X is fixed. The most used algorithms are the least squares and the stochastic gradient descent [36]. Independently of the implemented algorithm, the resulting dictionary does not guarantee excellent performance, because its atoms can be arbitrarily correlated.

A fundamental measure to characterize the quantity of a dictionary is the coherence. It is defined as the greatest correlation, in absolute value, between two distinct atoms of the dictionary under scrutiny. When dealing with unit-norm atoms, the coherence is defined as

$$\mu = \max_{i \neq j} |\mathbf{d}_i^T \mathbf{d}_j|. \tag{2}$$

The importance of this measure to characterize dictionaries has been demonstrated in several works [7, 8, 9]. For example, it is proven in [9] that orthogonal matching pursuit and basic pursuit can correctly recover the signal under the condition $p < \frac{1}{2}(\mu^{-1} + 1)$. Though this condition may not be applicable for

all sparse coding methods, the importance of incoherent dictionary learning is undoubted. It motivates the research on learning incoherent dictionaries.

Two strategies have been proposed to learn incoherent dictionaries, as surveyed in Section 2. In one strategy (the second strategy in Section 2), an incoherent dictionary is learned by introducing the regularization term $\|D^T D - I_p\|_F^2$, where I_p is the identity matrix of size $p \times p$ [25, 13]. The normalization is realized by adding the regularization term $\sum_{i=1}^{p} (\|\mathbf{d}_i\|^2 - 1)^2$ [25] or by a normalization step following the dictionary updating [13]. However, with these methods, the influence of the coherence of dictionary on the sparse representation cannot be explicitly measured. In the first strategy given in Section 2 with INK-SVD and IPR [10, 11], the incoherent dictionary learning algorithms give the relation between coherence of dictionary and accuracy of sparse representation. The price to pay is reduced accuracy of the sparse representation.

In this paper, we consider the explicit constraints. The coherence of the dictionary is constrained with the inequality

$$|\mathbf{d}_k^T \mathbf{d}_h| \leq \mu_c, \quad \forall k, h \in \{1, 2, \ldots, p\}, \ k \neq h \tag{3}$$

where μ_c is the predefined coherence level. The unit norm of the dictionary's atoms is obtained by the equality

$$\mathbf{d}_k^T \mathbf{d}_k = 1, \quad \forall k = 1, 2, \ldots, p. \tag{4}$$

Thus, the problem of incoherent dictionary learning can be resumed as a constrained optimization problem with quadratic objective function and quadratic constraints

$$\begin{aligned} &\min_{D \in \mathbb{R}^{n \times p}, \mathbf{x}_i \in \mathbb{R}^p} \frac{1}{\ell} \sum_{i=1}^{\ell} \tfrac{1}{2} \|\mathbf{y}_i - D\mathbf{x}_i\|_2^2 \\ &\text{subject to } \begin{cases} |\mathbf{d}_k^T \mathbf{d}_h| \leq \mu_c, \ \forall k, h \in \{1, 2, \ldots, p\}, k \neq h \\ \mathbf{d}_k^T \mathbf{d}_k = 1, \quad k = 1, \ldots, p \\ \|\mathbf{x}_i\|_0 \leq T, \quad i = 1, \ldots, \ell. \end{cases} \end{aligned} \tag{5}$$

The problem of estimating simultaneously X and D is non-convex and belongs to NP-hard problems. In next section, we provide an algorithm to solve this problem by an alternating strategy between sparse coding and dictionary updating, under all the explicit constraints.

4. Proposed Incoherent Dictionary Learning

In this section, we propose an incoherent dictionary learning algorithm. The algorithm learns a dictionary through two alternating processes: sparse coding and dictionary updating. For sparse coding, the problem with respect to X is reformulated as an MIQP problem that can be solved by the advanced optimization techniques [38, 39]. We also examine another method for sparse coding based on the proximal operator [40]. With a fixed sparse code, the problem

with respect to the dictionary D becomes a non-convex constrained optimization problem. For solving this problem, the augmented Lagrange method and proximal alternating linearized minimization method are used. The convergence of the proposed algorithm is analyzed in the appendix.

4.1. Sparse Coding Algorithm

Sparse coding tackles the problem (5) with D fixed, which is written as:

$$\begin{array}{ll} \min\limits_{\mathbf{x}_i \in \mathbb{R}^p} & \dfrac{1}{\ell} \sum\limits_{i=1}^{\ell} \left(\frac{1}{2} \|\mathbf{y}_i - D\mathbf{x}_i\|_2^2 \right) \\ \text{subject to} & \|\mathbf{x}_i\|_0 \leq T, \quad i = 1, \ldots, \ell. \end{array} \tag{6}$$

By considering the independence of the signals, the above problem can be split into ℓ small problems,

$$\min_{\mathbf{x} \in \mathbb{R}^p} \ \tfrac{1}{2} \|\mathbf{y} - D\mathbf{x}\|_2^2, \qquad \text{subject to } \|\mathbf{x}\|_0 \leq T. \tag{7}$$

Here, for the sake of clarity, any signal $\mathbf{y}_i$ and its sparse representation $\mathbf{x}_i$ are denoted respectively by $\mathbf{y}$ and $\mathbf{x}$.

This problem can be easily solved by a greedy algorithm such as matching pursuit (MP) [41] and orthogonal matching pursuit (OMP) [42] algorithm, or it can be relaxed as a convex problem that is known as basis pursuit [43]. In this research, we explore two recently proposed methods, proximal method [44, 13] and MIQP method [44, 20], to compute the sparse code of a signal.

4.1.1. Proximal method

The proximal method is proven to be an efficient algorithm to deal with non-smooth constrained large-scale problems [40]. The proximal minimization algorithm solves a problem by applying iteratively the proximal operator

$$\text{prox}_f(\mathbf{v}) = \arg\min_{\mathbf{x} \in \mathbb{R}^p} f(\mathbf{x}) + \tfrac{1}{2} \|\mathbf{x} - \mathbf{v}\|^2. \tag{8}$$

Let $\mathcal{S}_T$ denotes the T-sparse space, namely

$$\mathcal{S}_T = \{v \in \mathbb{R}^p \mid \|v\|_0 \leq T\}.$$

The objective function in our problem can be written as

$$f(\mathbf{x}) = \delta_{\mathcal{S}_T}(\mathbf{x}) + \tfrac{1}{2} \|\mathbf{y} - D\mathbf{x}\|^2, \tag{9}$$

where $\delta_{\mathcal{S}_T}$ denotes the indicator function over $\mathcal{S}_T$, namely

$$\delta_{\mathcal{S}_T}(\mathbf{x}) = \begin{cases} 0 & \text{if } \|\mathbf{x}\|_0 \leq T; \\ +\infty & \text{otherwise.} \end{cases} \tag{10}$$

It is hard to have a closed-form solution to our problem. Thus, here, the proximal linearized minimization algorithm is considered. By denoting $q(\mathbf{x}) = \frac{1}{2}\|\mathbf{y} - D\mathbf{x}\|^2$, the sparse coding problem can be rewritten as

$$\arg\min_{\mathbf{x}\in\mathcal{S}_T} \langle \mathbf{x} - \mathbf{x}^k, \nabla_{\mathbf{x}} q(\mathbf{x}^k)\rangle + \tfrac{1}{2\lambda}\|\mathbf{x} - \mathbf{x}^k\|^2, \tag{11}$$

where λ is the decrease step size. The solution to this problem (11) can be expressed with the help of the proximal operator,

$$\mathbf{x}^{k+1} = \mathrm{prox}_{\mathcal{S}_T}(\mathbf{x}^k - \lambda\nabla_{\mathbf{x}} q(\mathbf{x}^k)), \tag{12}$$

where $\mathrm{prox}_{\mathcal{S}_T}(\cdot)$ boils down to the projection onto the T-sparse space. In practice, $\mathbf{x}^{k+1}$ is simple to be obtained by keeping the T largest absolute values of $\mathbf{x}^k - \lambda\nabla_{\mathbf{x}} q(\mathbf{x}^k)$. It will produce a series $(\mathbf{x}^k)$. The global optimal solution of the original problem will be achieved when a fixed point is reached. This process can be interpreted as a succession of minimization of an upper bound on the objective function value.

4.1.2. MIQP

In the sparse coding problem, the discontinuity of the constraint makes it hard to tackle. The classical optimization techniques cannot be applied directly. To address this issue, we have recently proposed in [20] a strategy based on introducing an auxiliary variable $\mathbf{z} \in \{0,1\}^p$ (*i.e.*, a vector of p binary variables) indicating if the corresponding element in $\mathbf{x}$ is zero, that is

$$(\mathbf{1}_p - \mathbf{z})^T\mathbf{x} = 0, \tag{13}$$

where $\mathbf{1}_p$ is a vector of ones of size p. The sparsity constraint on $\mathbf{x}$ can be now explained by a constraint on $\mathbf{z}$,

$$\mathbf{1}^T\mathbf{z} \leq T. \tag{14}$$

Thus, in this problem, the optimization variables $\mathbf{x}$ and $\mathbf{z}$ are respectively continuous and integer. It is called a mixed-integer programming problem, with a quadratic objective function and non-linear constraints. The discretization of the variable makes this problem hard to solve. Fortunately, $\mathbf{z}$ can be relaxed to $[0,1]^p$ and the final solution is still discrete. See in [45] for the proof.

Introduced in [20], a second strategy consists in recasting the logical relation by a 'big-M' reformulation. As a consequence, the ℓ_0-based sparse coding problem (7) becomes, for a given M large enough,

$$\begin{array}{ll} \min\limits_{\mathbf{x}\in\mathbb{R}^p, \mathbf{z}\in\{0,1\}^p} & \frac{1}{2}\|\mathbf{y} - D\mathbf{x}\|_2^2 \\ \text{subject to} & \left\{ \begin{array}{l} -\mathbf{z}M \leq \mathbf{x} \leq \mathbf{z}M \\ \mathbf{1}_p^T\mathbf{z} \leq T. \end{array} \right. \end{array} \tag{15}$$

In this formulation, all the constraints are linear. Hence, the sparse coding can be interpreted as an MIQP. To solve MIQP problems, various optimization software packages can be explored, for example CPLEX and Gurobi Optimizer.

In general, the indicator constraint formulation shows advantage in computing complexity. However, if a tight M is given, the second strategy will be the first choice. In practice, an initialization value is produced by applying the proximal method, which helps to define an appropriate and tight M. Therefore, the 'big-M' formulation is used in this paper to tackle the sparse coding problem.

4.2. Dictionary Update via Proximal Alternating Method and Augmented Lagrangian Method

The dictionary update aims at addressing the problem:

$$\begin{aligned} \min_{D\in\mathbb{R}^{n\times p}} \quad & \frac{1}{\ell}\sum_{i=1}^{\ell}\left(\tfrac{1}{2}\|\mathbf{y}_i - D\mathbf{x}_i\|_2^2\right) \\ \text{subject to} \quad & \begin{cases} |\mathbf{d}_k^T\mathbf{d}_h| \le \mu_c,\ \forall k,h\in\{1,2,\ldots,p\},\ k\neq h \\ \mathbf{d}_k^T\mathbf{d}_k = 1, \quad k=1,\ldots,p. \end{cases} \end{aligned} \tag{16}$$

By introducing a new variable $G\in\mathbb{R}^{p\times p}$ that satisfies the itentity

$$G = D^TD,$$

the problem can be written in the form

$$\begin{aligned} \min_{D\in\mathbb{R}^{n\times p}, G\in\mathbb{R}^{p\times p}} \quad & \tfrac{1}{2}\|Y - DX\|_F^2 \\ \text{subject to} \quad & \begin{cases} G = D^TD, \quad G\in\mathcal{S}_{\mathcal{G}} \\ \mathbf{d}_k^T\mathbf{d}_k = 1, \quad k=1,\ldots,p, \end{cases} \end{aligned} \tag{17}$$

where

$$\mathcal{S}_{\mathcal{G}} = \left\{ G\in\mathbb{R}^{p\times p} \;\middle|\; |G_{ij}|\le\mu_c, i,j=\{1,2,\ldots,p\}, i\neq j \right\}.$$

Let $\delta_{\mathcal{S}_{\mathcal{G}}}(G)$ be the indicator function on this set, namely

$$\delta_{\mathcal{S}_{\mathcal{G}}}(G) = \begin{cases} 0, & \text{if } G\in\mathcal{S}_{\mathcal{G}} \\ +\infty, & \text{otherwise.} \end{cases} \tag{18}$$

The constrained optimization problem can be solved by considering the augmented Lagrangian function:

$$\begin{aligned} L_{(c1,c2)}(D,G,\boldsymbol{\lambda},H) =& \frac{1}{2}\|Y - DX\|_F^2 + \sum_{k=1}^{p}\lambda_k(\mathbf{d}_k^T\mathbf{d}_k - 1) + \frac{c_1}{2}\sum_{k=1}^{p}(\mathbf{d}_k^T\mathbf{d}_k - 1)^2 \\ & + \operatorname{tr}(H(G - D^TD)) + \frac{c_2}{2}\|G - D^TD\|_F^2 + \delta_{\mathcal{S}_{\mathcal{G}}}(G), \end{aligned} \tag{19}$$

where $\boldsymbol{\lambda} = [\lambda_1,\ldots,\lambda_p]$ and H are respectively the vector and matrix associated to the equality constraints on the diagonal of D^TD and on G, c_1 and c_2 are the positive penalty parameters (the augmentation). When these parameters grow into infinity, the optimal solution of the original problem (17) can be reached.

Algorithm 1 The inexact ADMM framework for solving (20)

Input: The training data (Y and X), the initialization of the parameters ($\boldsymbol{\lambda}^0$, D^0, c_1^0, H^0, G^0, c_2^0, ρ_1, ρ_2), the stop criteria (ϵ, N_{iter}).
Output: The optimal solution D^*
function DICTIONARYUPDATING
for all $i = 0$ to $N_{iter} - 1$ **do**
1. Computing the optimal solution (D^i, G^i):

$$(D^i, G^i) = \arg\min L_{(c_1^i, c_2^i)}(D, G, \boldsymbol{\lambda}^i, H^i).$$

2. Updating the Lagrangian multiplier $(\boldsymbol{\lambda}^i, H^i)$:

$$\begin{cases} \boldsymbol{\lambda}^{i+1} = \boldsymbol{\lambda}^i + c_1^i(\mathrm{diag}((D^i)^T D^i) - \mathbf{1}); \\ H^{i+1} = H^i + c_2^i(G^i - (D^i)^T D^i). \end{cases}$$

3. Updating the penalty parameters (c_1^i, c_2^i):

$$\begin{cases} c_1^{i+1} = \rho_1 c_1^i; \\ c_2^{i+1} = \rho_2 c_2^i. \end{cases}$$

4. Output the solution if the stop criteria or the maximum iteration number is reached.
if $\max|(\mathbf{d}_k^i)^T \mathbf{d}_k^i - 1| \leq \epsilon$ and $\max|G^i - (D^i)^T D^i| \leq \epsilon$ **then**
return,
end if
end for
end function

Therefore, the optimization problem becomes:

$$\min_{D \in \mathbb{R}^{n \times p}, G \in \mathbb{R}^{p \times p}} L_{(c1,c2)}(D, G, \boldsymbol{\lambda}, H). \tag{20}$$

It is not the standard augmented Lagrangian method (where the objective function is convex and has only one term, in most case, the constraints are closed convex set). While our problem is non-convex and non-smooth, it is still reasonable to consider the inexact ADMM framework [21]. The resulting algorithm is illustrated in Algorithm 1.

As presented in Algorithm 1, the inexact augmented Lagrangian method operates in three alternating steps: In Step 1, the primal variables D, G are computed, namely

$$(D^i, G^i) = \arg\min_{D,G} L_{(c_1^i, c_2^i)}(D, G, \boldsymbol{\lambda}^i, H^i), \tag{21}$$

where D^i, G^i $\boldsymbol{\lambda}^i$ and H^i are the values in the i-th iteration; In Step 2, the Lagrangian multipliers $\boldsymbol{\lambda}$ and H are updated; And in Step 3, the penalty parameters c_1 and c_2 are increased. It is proven that the two parameters c_1 and c_2 can stay much smaller than $+\infty$ to solve the problem [21].

The problem (21) is a non-convex and non-smooth optimization problem. It is unsolvable by satisfying the KKT conditions. We propose to use the alternating strategy to address this problem. The optimal matrices of D and G are obtained by alternating the gradient descent method and the proximal method,

which can be regarded as the generalisation the extended proximal alternating linearized minimization (EPALM) introduced in [22]. However, the proposed modification makes it treat the more difficult problem.

To investigate the EPALM method, we rewrite the objective function in problem (21) in the form of three additive parts:

$$L_{(c_1^i,c_2^i)}(D,G,\boldsymbol{\lambda}^i,H^i) = f(D) + h(D,G) + g(G), \tag{22}$$

with the definition of:

$$\begin{cases} f(D) &= \frac{1}{2}\|Y - DX\|_F^2 + \sum_{k=1}^p \lambda_k^i(\mathbf{d}_k^T\mathbf{d}_k - 1) + \frac{c_1^i}{2}\sum_{k=1}^p(\mathbf{d}_k^T\mathbf{d}_k - 1)^2 \\ h(D,G) &= \mathrm{tr}\big(H^i(G - D^TD)\big) + \frac{c_2^i}{2}\|G - D^TD\|_F^2 \\ g(G) &= \delta_{\mathcal{S}_\mathcal{G}}(G), \end{cases}$$

This problem is proved to be well defined, the details of demonstration can be found in the appendix.

The problem (21) can now be solved by alternating the optimization problems with respect to D and G, respectively:

$$\begin{cases} D^{i,j} = \underset{D\in\mathbb{R}^{n\times p}}{\arg\min}\ f(D) + h(D^{i,j-1},G^{i,j-1}) \\ \qquad\qquad +\mathrm{tr}\big((D - D^{i,j-1})^T\nabla_D h(D^{i,j-1},G^{i,j-1})\big) + \frac{\tilde{t}_1}{2}\|D - D^{i,j-1}\|_F^2 \\ G^{i,j} = \underset{G\in\mathbb{R}^{p\times p}}{\arg\min}\ g(G) + h(D^{i,j},G^{i,j-1}) \\ \qquad\qquad +\langle G - G^{i,j-1},\nabla_G h(D^{i,j},G^{i,j-1})\rangle + \frac{t_2}{2}\|G - G^{i,j-1}\|_F^2, \end{cases} \tag{23}$$

where $\langle M_1, M_2\rangle = \mathrm{tr}(M_1^TM_2)$ is defined as the scalar product in the matrix space $\mathcal{M}_n(\mathbb{R})$, $\tilde{t}_1$ and t_2 are the coefficients associated respectively to the second order approximation term. By considering that the function f is differentiable, the problem of estimating D can be expressed as:

$$\begin{aligned} D^{i,j} = \underset{D\in\mathbb{R}^{n\times p}}{\arg\min}\ & f(D^{i,j-1}) + h(D^{i,j-1},G^{i,j-1}) \\ & +\mathrm{tr}\big((D - D^{i,j-1})^T\nabla_D(f(D^{i,j-1}) + h(D^{i,j-1},G^{i,j-1})\big) \\ & +\frac{t_1}{2}\|D - D^{i,j-1}\|_F^2. \end{aligned} \tag{24}$$

To solve this problem, it is easy to update D by the method of gradient descent. As for the problem of estimating G in (23), the proximal method is applicable. By combining both steps, the solution of D and G can be achieved by the following process:

$$\begin{cases} D^{i,j} = D^{i,j-1} - \frac{1}{t_1}\nabla_D\big(f(D^{i,j-1}) + h(D^{i,j-1},G^{i,j-1})\big) \\ G^{i,j} = \mathrm{prox}_{\frac{1}{t_2}g}\big(G^{i,j-1} - \frac{1}{t_2}\nabla_G(h(D^{i,j},G^{i,j-1}))\big), \end{cases} \tag{25}$$

where the notation prox_{uf} denotes the proximal operator defined in (8) of the scaled function uf (also called the proximal operator of g with parameter u), namely

$$\text{prox}_{\frac{1}{t_2}f}(\mathbf{v}) = \arg\min_{\mathbf{x}} f(\mathbf{x}) + \tfrac{1}{2t_2}\|\mathbf{x}-\mathbf{v}\|^2. \tag{26}$$

The partial derivatives of the three parts of (22) are:

$$\begin{cases} \nabla_D f(D^{i,j-1}) & = -(Y - D^{i,j-1}X)X^T \\ & \quad +2D^{i,j-1}\text{diag}(\boldsymbol{\lambda^i}) + 2c_1^i D^{i,j-1}\text{diag}(\mathbf{v^{j-1}}) \\ \nabla_D h(D^{i,j-1}, G^{i,j-1}) & = -D^{i,j-1}(H^i + (H^i)^T) \\ & \quad -c_2^i D^{i,j-1}\big(G^{i,j-1} + (G^{i,j-1})^T - 2(D^{i,j-1})^T D^{i,j-1}\big) \\ \nabla_G h(D^{i,j}, G^{i,j-1}) & = H^i + c_2^i(G^{i,j-1} - (D^{i,j})^T D^{i,j}), \end{cases} \tag{27}$$

where $\mathbf{v^{j-1}} = \text{diag}\big((D^{i,j-1})^T D^{i,j-1} - I_p\big)$ denotes the vector with the entries valued by the diagonal of the matrix $(D^{i,j-1})^T D^{i,j-1} - I_p$. The two expressions $\text{diag}(\boldsymbol{\lambda^i})$ and $\text{diag}(\mathbf{v^{j-1}})$ return the matrix with the diagonal filled by the elements in the vectors $\boldsymbol{\lambda^i}$ and $\mathbf{v^{j-1}}$, respectively. Moreover, $G^{i,j}$ can be computed as:

$$G^{i,j}(i_x, i_y) = \begin{cases} \widetilde{G}(i_x, i_y) & \text{if } |\widetilde{G}(i_x, i_y)| \le \mu_c \\ \text{sign}(\widetilde{G}(i_x, i_y))\mu_c & \text{otherwise,} \end{cases} \tag{28}$$

where $\widetilde{G} = G^{i,j-1} - \frac{1}{t_2}\nabla_G\big(h(D^{i,j}, G^{i,j-1})\big)$, $\text{sign}(\widetilde{G}(i_x, i_y))$ is the sign of the $\widetilde{G}(i_x, i_y)$, *i.e.*, $+1$ if $\widetilde{G}(i_x, i_y) \ge 0$ and -1 otherwise. In these expressions, i_x and i_y are respectively the row and column indices of the matrix $\widetilde{G}$, with $i_x, i_y = 1, 2, \ldots, p$.

The proposed EPALM method for estimating D and G is summarized in Algorithm 2.

5. Experiments and results

In this part, we describe the experiments conducted on synthetic and real images, and analyze on the results. We first evaluate the accuracy and the convergence property of the proposed algorithm using synthetic data. And then, the dictionary learning algorithm based on the EPALM method is used for image reconstruction on a real image. The results are compared with INK-SVD [10] and the incoherent dictionary learning algorithm by iterative projection and rotation (IPR) [11].

5.1. Experiment on synthetic data

While theoretical convergence is proven in this paper (see appendix), we provide in this section experiments on the accuracy and convergence using synthetic data. The analysis is conducted on the proposed dictionary updating algorithm, independently of the sparse coding algorithm.

Algorithm 2 EPALM algorithm for solving subproblem (21)

Input: The training data (Y and X), the parameters ($\boldsymbol{\lambda^i}$, H^i, c_1^i and c_2^i, the initialization of the variables $D^{i,0} = D^{i-1}$, the step size t_1 and t_2), the stop criteria (ϵ^i, N_{iter}^i, the subdifferential Θ^i of the $L_{(c_1^i,c_2^i)}(D^i, G^i, \boldsymbol{\lambda}^i, H^i)$)

Output: The solution D^i and G^i

function EPALM

Initialization $j = 0$,
$G^{i,0} = (D^{i,0})^T D^{i,0}$,
$G^{i,0}(i_x, i_y) = \text{sign}(G^{i,0}(i_x, i_y)) \min(|G^{i,0}(i_x, i_y)|, \mu_c)$,
$\Theta^i = \Theta^i(D^{i,0}, G^{i,0})$.

while $j < N_{iter}^i$ and $\Theta^i > \epsilon^i$ **do**

1. Updating $D^{i,j}$ by computing:

$$D^{i,j} = D^{i,j-1} - \tfrac{1}{t_1}\nabla_D\big(f(D^{i,j-1}) + h(D^{i,j-1}, G^{i,j-1})\big).$$

2. Computing:

$$\widetilde{G} = G^{i,j-1} - \tfrac{1}{t_2}\nabla_G\big(h(D^{i,j}, G^{i,j-1})\big).$$

3. Projecting the $\widetilde{G}$ in the space $\mathcal{S}_\mathcal{G}$:

$$G^{i,j}(i_x, i_y) = \begin{cases} \widetilde{G}(i_x, i_y) & \text{if } |\widetilde{G}(i_x, i_y)| \leq \mu_c; \\ \text{sign}(\widetilde{G}(i_x, i_y))\mu_c & \text{otherwise.} \end{cases}$$

4. Calculating the subdifferential $\Theta^i(D^{i,j}, G^{i,j})$.
5. $j = j + 1$.

end while

end function

In this first experiment, we consider $\ell = 100$ signals of dimension $n = 5$, and $p = 20$ atoms to be learned. A sparse matrix $X \in \mathbb{R}^{20\times 100}$ with the maximal column-wise sparsity level 3 is manually created. A learned dictionary $D \in \mathbb{R}^{5\times 20}$ is generated from the IPR incoherent dictionary learning algorithm [11] on an arbitrary image, with the coherence parameter set to 0.6; The obtained dictionary has a coherence computed by (2) of 0.608. Then, in each test, the set of signals Y can be generated in the way that:

$$Y = DX + \omega E, \tag{29}$$

where the second term in the right-hand-side corresponds to the unfitness noise, where E a white Gaussian zero-mean matrix with a noise level set to $\omega = 0.1$.

To provide an overall evaluation of the proposed algorithm, several different values of the coherence parameters are used, with $\mu_c = \{0.5, 0.55, 0.6, 0.7, 0.8, 0.9, 1\}$. It is worth noting that values below 0.4 cannot be reached due to geometric constraints [9], namely, the coherence of an overcomplete dictionary of size $n \times p$ is bounded by

$$\mu \geqslant \sqrt{\frac{p-n}{n(p-1)}}. \tag{30}$$

The algorithm is run in the Matlab® environment on a MacBook with 2 Intel Core i5 processors with a CPU clocked at 2.7 GHz. The parameters values are set as follows: For Algorithm 1, the maximal outer iteration number $N_{iter} = 50$,

Table 1: Accuracy results and computing time on synthetic data

Coherence parameter μ_c	0.5	0.55	0.6	0.7	0.8	0.9	1.0
Initial objective function value: $\frac{1}{2}\|Y - D_0X\|_F^2$	9523	9592	9318	9643	9340	9483	9446
Final objective function value: $\frac{1}{2}\|Y - \widetilde{D}X\|_F^2$	216.84	91.80	2.01	1.89	1.89	1.89	1.89
Accuracy: $\|\widetilde{D} - D^*\|_F$	1.480	1.060	0.058	0.100	0.100	0.100	0.100
Outer iteration number	50	22	9	4	4	4	4
Inner iteration number	972	877	692	312	319	319	318
$\max \|\mathbf{d}_k^T\mathbf{d}_k - 1\|$	0.015	0.001	0.003	0.007	0.007	0.007	0.007
$\max \|G - D^TD\|$	0.052	0.0057	0.0084	10^{-8}	10^{-8}	10^{-8}	10^{-8}
Computing time in seconds	155.78	47.31	16.33	3.38	3.42	3.41	3.39

the coefficient to update the penalty parameter $\rho_1 = \rho_2 = 1.5$, the stop criterion $\epsilon = 0.01$; For Algorithm 2: the maximal inner iteration number $N_{iter}^i = 1000$, the stop criterion $\epsilon^i = \epsilon^0 = 0.01$. For each coherence parameter value, five independent Monte Carlo simulations are conducted.

We analyze the algorithm through the accuracy $\|\widetilde{D} - D^*\|_F$ and the objective function value $\frac{1}{2}\|Y - \widetilde{D}X\|_F^2$, as well as the computing time, where $\widetilde{D}$ is output of the algorithm and D^* is the optimal known solution. Moreover, the iteration numbers with different coherence parameter settings is also studied. The results are listed in the Table 1. It shows that, with the decrease of the coherent parameter μ_c, more iterations are needed to converge, and thus more time. On the other hand, as the coherence parameter increases, the stopping criteria $\max(|\mathbf{d}_k^T\mathbf{d}_k - 1|) \leq 0.01$ and $\max|G - D^TD| \leq 0.01$ can be easily satisfied.

From Table 1, we observe that when $\mu_c = 0.6$, which is the closest value to the coherence of the target dictionary D^* (*i.e.*, $\mu^* = 0.607$), the results have the greatest accuracy of 0.058. For the other values of μ_c, the results remain consistent but with a deduced accuracy. This is easy to understand since, for $\mu_c > \mu^*$, the optimal solution D^* is in the feasible region, which should also be the output of the algorithm. But, influenced by the noise, the output of the algorithm cannot be exactly D^*. For this reason, the objective function for $\mu_c > \mu^*$ are always less than that when $\mu_c = 0.6$. However, when $\mu_c < \mu^*$, the situation is totally different, because μ^* is out of the feasible region. Therefore, a solution that satisfies the coherence constraint can be found, but the price to pay is an increase of the objective function, as well as the computational cost to converge. Consequently, by appropriately choosing the coherent parameter, an incoherent dictionary can be produced by this algorithm. Moreover, the smaller the target coherent parameter is, the greater the computational complexity will be.

5.2. Real image reconstruction

This experiment focuses on the performance of image reconstruction by using the proposed dictionary learning algorithm, namely by combining the EPALM dictionary update algorithm with either the proximal or the MIQP sparse coding algorithms. The property of the convergence and reconstruction accuracy

Figure 1: Barbara image

will be discussed in the following. Furthermore, the influence of the coherence parameter on the reconstruction results will be compared with other algorithms, such as INK-SVD [10] and the incoherent dictionary learning algorithm by iterative projection and rotation (IPR) [11].

The segment of image *Barbara* of size 121×121, as shown in Figure 1, is chosen as the experimental data. The overlapping patches of size 8×8 (namely, a signal is a vector of size 64) form the set of signals Y. With the signals, a dictionary D is learned by using the proposed method (EPALM for dictionary updating and proximal method or MIQP for sparse coding) and compared to the other two comparative incoherent dictionary learning algorithms. When both D and Y are known, the sparse code X can be easily obtained using a sparse coding method, namely proximal method and MIQP for our algorithm, OMP algorithm for the other two methods. Then, the reconstructed image is obtained by doing the matrix multiplication $\widetilde{Y} = DX$. Consequently, we compare their performance by calculating the peak signal-to-noise ratio (PSNR).

The MIQP problem is solved by the software Gurobi Optimizer 8.1.0. We run the programs in the Matlab® environment on a server with 4 Intel® Xeon® processors with a CPU clocked at 2.4 GHz. The parameter settings of Gurobi are fixed as the default values except that the time limit is set 0.5 seconds and iteration number 1000. The initialization of MIQP is given by running the proximal method with the maximal iteration number set to 200. When only the proximal method is used for sparse coding, the iteration number is set to 1000. The number of atoms is set to $p = 256$ and the sparsity level $T = 20$ (the active atoms is less than 8%). In the phase of dictionary update, the parameter setting is just set the same as in the test on synthetic data. The iteration number for learning a dictionary is determined 30, which is sufficient for the algorithms to converge, as shown in Figure 3. For the other two comparative methods, the parameter values are chosen as in the original papers [10, 11].

Figure 2 presents the convergence property of the algorithms, which is obtained by fixing the coherent parameter to $\mu_c = 0.6$, which corresponds to having angles between any two atoms greater than 53°. It is observable that the dictionary algorithm with MIQP for sparse coding and EPALM for dictionary updating has the fastest convergence and the value of limit is the smallest. It is worth pointing out that 30 iteration is sufficient for the algorithms to converge,

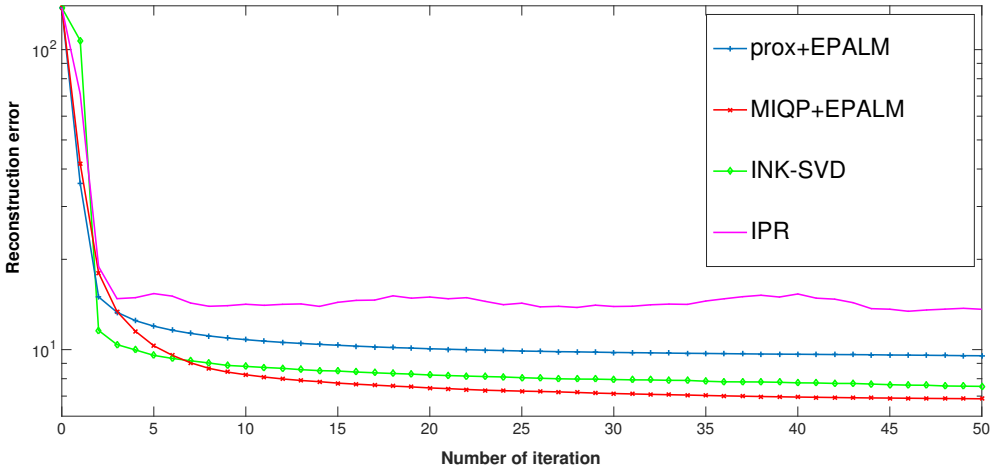

Figure 2: The convergence of the proposed algorithm and its comparison to the INK-SVD and IPR algorithms

Table 2: Statistics on the resulting dictionary

	μ	Average of $\{\lvert\mathbf{d}_i^T\mathbf{d}_j\rvert \mid i \neq j\}$	Variance of $\{\lvert\mathbf{d}_i^T\mathbf{d}_j\rvert \mid i \neq j\}$
INK-SVD	0.601	0.368	0.0177
IPR	0.711	0.557	0.0073
Proximal+EPALM	0.608	0.352	0.0176
MIQP+EPALM	0.609	0.382	0.0146

even though the IPR algorithm shows some convergence unstability.

Figure 3 shows the distribution of the absolute inner product between each two atoms in the learned dictionary. Combined by the statistics in Table 2, we notice that independently of the used sparse coding algorithm, the proposed method can achieve a dictionary with almost the target coherence parameter value, which is not the case of IPR. The proximal method combined with EPALM provides the smallest absolute average, which is an important property related to the so-called Babel function whose theory is well established [7, 8, 9]. However, this algorithm cannot beat the one with MIQP for sparse coding in terms of variance. INK-SVD outputs as well a dictionary with the almost the target coherence value, but with a higher variance. Nevertheless, INK-SVD updates the dictionary without considering the reconstruction error (see next paragraph). For the IPR algorithm, the target coherence parameter value cannot be obtained even though it shows the least variance. Considering the distribution of absolute inner products between each two atoms in learned dictionary, as illustrated in Figure 3, it is hard to tell if the proximal method or the MIQP is better to combine to EPALM. Next paragraph presents an analysis on the reconstruction error.

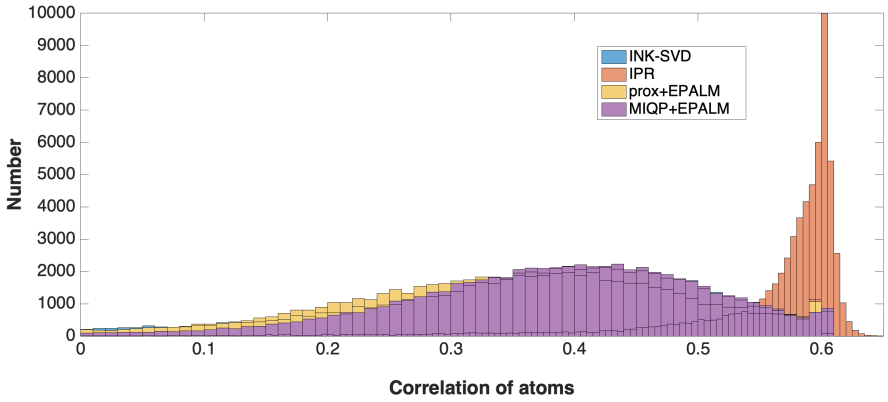

Figure 3: The distribution of the coherence between each two atoms of the proposed algorithms and its comparison to the INK-SVD and IPR algorithms

Table 3: The reconstruction errors in PSNR (in dB) by using the dictionary with different coherence parameter values $\mu_c = \cos(\theta_c)$

	Largest angle θ_c between two atoms						
	5°	15°	30°	45°	60°	75°	83°
INK-SVD	36.46	36.56	36.26	34,83	34.04	30.15	-
IPR	36.82	36.57	35.72	31.51	30.60	27.84	-
Proximal+EPALM	27.40	27.42	28.06	29.80	29.31	29.75	22.97
MIQP+EPALM	**37.60**	**37.26**	**38.89**	**38.55**	**36.52**	**35.31**	**33.97**

To analyze the reconstruction errors, we study seven different coherence values, that is, the angle between any two atoms is bigger than $\{5°, 15°, 30°, 45°, 60°, 75°, 83°\}$, the corresponding coherence values being $\mu_c = \{0.996, 0.966, 0.866, 0.707, 0.500, 0.259, 0.122\}$. For each value, a dictionary with such target coherence value is learned, and the relation between the reconstruction performance and the coherence of the dictionary is illustrated next.

Table 3 and Figure 4 illustrate the reconstruction results. The visualization results are showed in Figure 5. The combination of MIQP for sparse coding and EPALM for dictionary update outperforms the other methods for all the coherence parameter values. Moreover, it is interesting to find that for our proposed method, the reconstruction performance improve with the coherence of dictionary decreasing, with the best results when $\mu_c = \cos(45°)$ with proximal method for sparse coding and $\mu_c = \cos(30°)$ with MIQP; afterwards, the reconstruction performance begins to decrease. This is different from the results of INK-SVD and IPR algorithms whose performances monotonically decrease with the coherence (*i.e.*, the incoherence of the dictionary is increasing). Hence, our algorithm increases the dictionary incoherence without the risk of loss of reconstruction accuracy, which corroborates theoretical results proved in [9] but

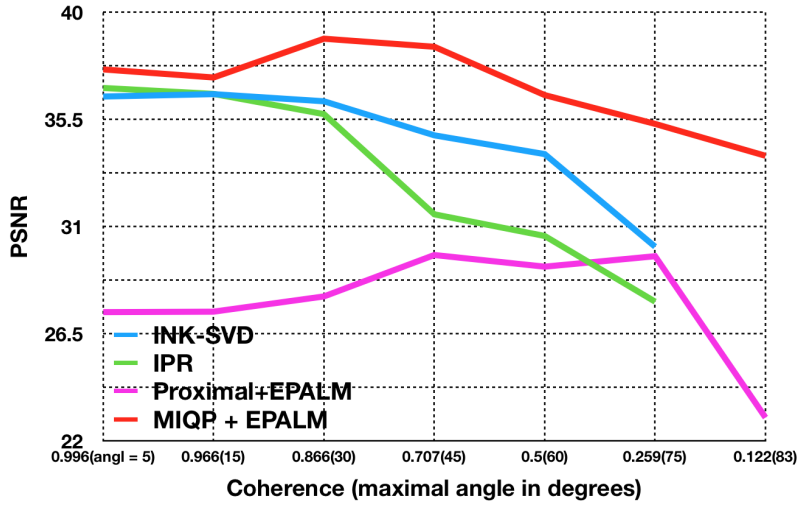

Figure 4: The figure of reconstruction results

not verified in real-world problem by algorithms such as INK-SVD or IPR. Furthermore, our method proves that an appropriate incoherent dictionary helps to improve the performance. However, one point should be noticed, incoherent dictionary learning algorithm with MIQP for sparse coding has the highest computing complexity comparing to the other methods.

6. Conclusion

This paper investigated the exact incoherent dictionary learning, where all the constraints were explicitly solved. To this end, we proposed a new dictionary update algorithm EPALM by combining the proximal alternating minimization method and augmented Lagrangian method. This algorithm was used for dictionary learning together with a sparse coding algorithm, such as the proximal method and MIQP. In this paper, we showed firstly the feasibility of the algorithm on synthetic data, examining the performance of the dictionary learning independently of the sparse coding algorithm. And then, the incoherent dictionary algorithm was used for real image reconstruction. We studied the statistics of the resulting dictionary, and the reconstruction performance for a large set of target coherence parameters. It was proven that the combination of EPALM for dictionary updating and MIQP for sparse coding always outperformed the other methods in terms of the reconstruction results. The relevance of having an incoherent dictionary was also demonstrated.

(a) INK-SVD (b) IPR

(c) Proximal+EPALM (d) MIQP+EPALM

Figure 5: Reconstructed images

Appendix A. Convergence Analysis

In this part, we focus on the convergence analysis of our algorithm. In terms of the subdifferential of the objective function in (21), as well as the choice of the parameters, a view of convergence analysis can be described as follows.

Definition 1. *([46]) Let $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ be a proper lower semicontinuous function. For each $x \in dom\ f$ (where $dom\ f = \{x \in \mathbb{R}^n \mid F(x) < +\infty\}$), the Fréchet subdifferential of f at x is*

$$\partial f = \bigcap_{z \in dom\ f} \{g \mid f(z) \geq f(x) + g^T(z - x)\}. \tag{A.1}$$

A necessary but not sufficient condition for $x \in \mathbb{R}^n$ to be a minimizer of f is $0 \in \partial f(x)$. Back to our optimization problem, the subdifferential of $L_{(c_1^i, c_2^i)}(D, G, \boldsymbol{\lambda}^i, H^i)$ at (D^i, G^i), denoted by $\Theta^i = (\Theta_D^i, \Theta_G^i)$ and expressed as

$$\Theta^i = \partial L_{(c_1^i, c_2^i)}(D^i, G^i, \boldsymbol{\lambda}^i, H^i),$$

can be computed directly and the result can be written in form of:

$$\begin{cases} \Theta_D^i = \nabla_D f(D^i) + \nabla_D h(D^i, G^i) \\ \Theta_G^i = t_2(G^{i-1} - G^i). \end{cases} \tag{A.2}$$

Thus a solution of the problem will be found when $\|\Theta^i\|_\infty \to 0$. According to the formulation of Θ^i in (A.2), D^i is exactly the local optimal solution of the subproblem with respect to D, and the sequence $G^{i,j}$ is convergent, since $\|G^i - G^{i-1}\|_F \to 0$.

Besides, it is noticed that to guarantee that every bounded sequence generated by the proposed method converges to a critical point of the $L_{(c_1^i, c_2^i)}(D^i, G^i, \boldsymbol{\lambda}^i, H^i)$, the parameters c_1, c_2 and the steps t_1, t_2 need to be appropriately chosen. The following can be noted:

- c_1^0 and c_2^0 should be carefully chosen to avoid the ill-condition, *i.e.*, the initial positive penalty parameters c_1^0 and c_2^0 satisfy the second order sufficient condition:
$$\nabla^2_{DD} L(D^i, G^i, \boldsymbol{\lambda}^{\boldsymbol{i}}, H^i) > 0;$$
Due to the complexity of the derivative of a matrix function with respect to a matrix (the derivative of the function with respect to each element of the matrix being a matrix), we do not give the detail here.

- The convergence of the algorithm requires that the descent steps, *i.e.*, $\frac{1}{t_1}$ and $\frac{1}{t_2}$ should not be too much big, which satisfy that $t_1 > L_D$ and $t_2 > L_G$, where L_D and L_G are the globally Lipschitz constant of the gradient of the functions $D \to h(D, G)$ and $G \to h(D, G)$.

Proposition 1. *To sum up, a sequence $((D^{i,j}, G^{i,j}))_{j\in\mathbb{N}}$ is generated by using the proposed method, then the following condition will be satisfied:*

- *When $j \to \infty$, $\|\Theta^i(D^{i,j}, G^{i,j})\|_\infty \to 0$*
- *The sequence $((D^{i,j}, G^{i,j}))_{j\in\mathbb{N}}$ has finite length, that is,*
$$\sum_{j=1}^{\infty} \|(D^{i,j+1}, G^{i,j+1}) - (D^{i,j}, G^{i,j})\|_F < \infty \tag{A.3}$$

In the following, we give the proof of the convergence of the proposed algorithm. As aforementioned, our algorithm aims at tackling the constrained optimization problem by transforming the problem into an unconstrained optimization problem via the augmented Lagrangian method. In each iteration of the augmented Lagrangian method, the minimization problem with respect to the primal variables is solved by the EPALM algorithm. Thus, for proving the convergence of the algorithm, we need to prove the convergence of the augmented Lagrangian method and that of the EPALM.

Appendix A.1. Convergence of the augmented Lagrangian method

Before proceeding and for completeness, we give here the convergence of the augmented Lagrangian method [21]. Consider the general expression of an equality-constrained problem:

$$\begin{array}{ll} \min & q(x) \\ \text{subject to} & p(x) = 0, \quad \forall x \in \mathcal{X}, \end{array} \tag{A.4}$$

where $\mathcal{X}$ is a closed set, and q and p are continuous functions in $\mathcal{X}$.

Proposition 2 (Proposition 4.2.1 in [21])**.** *Assume q and p are continuous functions, $\mathcal{X}$ is a closed set, and the constraint set $\{x \in \mathcal{X} \mid p(x) = 0\}$ is nonempty. For $k = 0, 1, \cdots$, let x^k be a global minimum of the optimization problem*

$$\min_{x \in \mathcal{X}} L_{c^k}(x, \lambda^k), \tag{A.5}$$

where λ^k is bounded, $0 < c^k < c^{k+1}$ for all k, and $c^k \to \infty$. Then every limit point of the sequence (x^k) is a global minimum of the original problem (A.4)*.*

Furthermore, according to the Proposition 4.2.2 in [21], the limit of the sequence $\{\lambda^k\}$ can be reached by iteratively updating λ^k through $\widetilde{\lambda}^k = \lambda^k + c^k p(x^k)$, and $\lim_{k\to\infty} \lambda^k + c^k p(x^k) = \lambda^*$. x^* is the solution of $\partial(q + \lambda^* p)(x^*) = 0$.

Appendix A.2. Sufficient condition for well-defining the problem (22)

The problem (22) is well defined with $f : \mathbb{R}^{n\times p} \to (-\infty, +\infty]$, $h : \mathbb{R}^{n\times p} \times \mathbb{R}^{p\times p} \to (-\infty, +\infty]$ being a $\mathcal{C}^1$ function (*i.e.*, continuously differentiable), and inf $f(D) > -\infty$, inf $h(D, G) > -\infty$, $g : \mathbb{R}^{p\times p} \to [0, \infty]$ a proper and lower semicontinuous function.

Appendix A.3. Convergence of the EPALM algorithm

For guaranteeing the convergence of the EPALM method, the following assumption should be satisfied:

(i) The functions $D \to h(D, G)$ and $G \to h(D, G)$ have their gradients globally Lipschitz continuous with module L_D and L_G, respectively. In other words, the partial gradients of h with respect to D and G show the property:

$$\begin{cases} \|\nabla_D h(D, G) - \nabla_D h(\bar{D}, G)\|_F \le L_D \|D - \bar{D}\|_F \\ \|\nabla_G h(D, G) - \nabla_G h(D, \bar{G})\|_F \le L_G \|G - \bar{G}\|_F, \end{cases} \tag{A.6}$$

for all $(D, \bar{D})$ and $(G, \bar{G})$. with $-\infty < \underline{L} < L_D, L_G < \overline{L} < +\infty$.

(ii) $L_{(c_1^i, c_2^i)}(D, G, \boldsymbol{\lambda}^i, H^i)$ satisfies the Kurdyka-Łojasiewicz inequality [22].

The definition of the Kurdyka-Łojasiewicz (KL) equality [47] is:

Definition 2. *(Kurdyka-Łojasiewicz function)*

(a) The function $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ is said to have the Kurdyka-Łojasiewicz property at $x^* \in \operatorname{dom} \partial f$ if there exist $\eta \in (0, +\infty]$, a neighborhood U of x^* and a continuous concave function $\phi : [0, \eta) \to \mathbb{R}_+$ such that:

(i) $\phi(0) = 0$

(ii) ϕ is $\mathcal{C}^1$ on $(0, \eta)$

(iii) for all $s \in (0, \eta)$, $\phi'(s) > 0$

(iv) for all x in $U \cap [f(x^*) < f < f(x^*) + \eta]$, the Kurdyka-Łojasiewicz inequality holds:

$$\phi^{'}(f(x) - f(x^*))dist(0, \partial f(x)) \geq 1 \tag{A.7}$$

(b) The proper lower semicontinuous functions that satisfy the Kurdyka-Łojasiewicz inequality at each point of dom ∂f *are called KL functions.*

We now study the convergence property of the algorithm, that is, the convergence of the sequence generated by the proposed algorithm in this paper. We will prove that the proposed algorithm generates a sequence $(x^k)_{k\in\mathbb{N}}$ that satisfies the following conditions:

H1. *(Sufficient decrease condition).* For each $k \in \mathbb{N}$,

$$f(x^{k+1}) + a\|x^{k+1} - x^k\|^2 \leq f(x^k);$$

H2. *(Relative error condition).* For each $k \in \mathbb{N}$, there exists $w^{k+1} \in \partial f(x^{k+1})$ such that

$$\|w^{k+1}\| \leq b\|x^{k+1} - x^k\|;$$

H3. *(Continuity condition).* There exists a subsequence $(x^{k_j})_{j\in\mathbb{N}}$ and $\tilde{x}$ such that

$$x^{k_j} \to \tilde{x} \text{ and } f(x^{k_j}) \to f(\tilde{x}), \qquad \text{when } j \to \infty. \tag{A.8}$$

Then the following theorem will be used to prove the convergence of the proposed algorithm [47, Theorem 2.9].

Theorem 1. *(Convergence to a critical point)*
*Let $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ be a proper lower semicontinuous function. Consider a sequence $(x^k)_{k\in\mathbb{N}}$ that satisfies the conditions **H1, H2, H3**.*

*If f has the Kurdyka-Łojasiewicz property at the cluster point $\tilde{x}$ specified in **H3**, then the sequence $(x^k)_{k\in\mathbb{N}}$ converges to $\bar{x} \to \tilde{x}$ as k goes to infinity, and $\bar{x}$ is a critical point of f.*

Moreover the sequence $(x^k)_{k\in\mathbb{N}}$ has a finite length, i.e.*,*

$$\sum_{k=0}^{+\infty} \|x^{k+1} - x^k\| < +\infty.$$

In the following, we begin with the proof of satisfaction of assumption on functions.

Proposition 3. *The objective function $L_{(c_1^i, c_2^i)}(D, G, \boldsymbol{\lambda}^i, H^i)$ is a KL function.*

Proof. The objective function $L_{(c_1^i, c_2^i)}(D, G, \boldsymbol{\lambda}^i, H^i)$ can be written in form of (22), namely $f(D) + h(D, G) + g(G)$. According to [24] and therein, it is easy to prove that f and h are KL functions. Moreover, g is also a KL function because it is the indicator function of a semi-algebraic set. Hence, the sum of the KL functions, *i.e.*, $L_{(c_1^i, c_2^i)}(D, G, \boldsymbol{\lambda}^i, H^i)$, is a KL function. □

Proposition 4. *In problem* (A.4)*, if $q(x)$ is a proper semicontinuous function in a closed set $\mathcal{X}$ and $p(x)$ is a proper lower continuous function in $\mathcal{X}$, then the augmented Lagrangian function $L_{c^k}(x, \lambda^k)$ is a proper lower semicontinuous function.*

Proof. Firstly, if $p(x)$ is a continuous function in $\mathcal{X}$, then $\{x \mid p(x) < \infty\} = \mathcal{X}$. Moreover, $q(x)$ is a proper function in $\mathcal{X}$. $L_{c^k}(x, \lambda^k) = q(x) + \lambda^k p(x) + \frac{c^k}{2} p^2(x)$ is consequently a proper function in $\mathcal{X}$.

Secondly, it is evident that if $p(x)$ is a continuous function in $\mathcal{X}$, then $\lambda^k p(x)$ and $\frac{c^k}{2} p^2(x)$ are continuous functions in $\mathcal{X}$. The sum of a semicontinuous function in $\mathcal{X}$, the function $q(x)$ and a continuous function $\lambda^k p(x) + \frac{c^k}{2} p^2(x)$, is still a semicontinuous function, *i.e.*, $L_{c^k}(x, \lambda^k)$ is a semicontinuous function.

Finally, p and q are both lower-bounded functions in $\mathcal{X}$, that is, $\forall x \in \mathcal{X}$, $p(x) > -\infty$ and $q(x) > -\infty$. $\lambda^k p(x) + \frac{c^k}{2} p^2(x)$ is a convex function because $c > 0$, then, $\lambda^k p(x) + \frac{c^k}{2} p^2(x) > -\infty$, $\forall x \in \mathcal{X}$. Hence, $L_{c^k}(x, \lambda^k)$ is the sum of two lower-bounded functions in $\mathcal{X}$.

Therefore, $L_{c^k}(x, \lambda^k)$ is a proper lower semicontinuous function. □

In our optimization problem, $q(x)$ is a proper lower semicontinuous function dedicating here to an indicator term and $p(x)$ is a set of linear functions and quadratic functions, which are all proper lower continuous function. By applying the above Proposition 4, we can deduce that the augmented Lagrangian function in our problem is a proper lower semicontinuous function.

Now, to prove the convergence of the algorithm, we still need to prove that the generated sequence $(D^{i,j}, G^{i,j})$ satisfies the conditions **H1**, **H2** and **H3**. The sequence is generated from the process

$$\begin{cases} D^{i,j+1} = \arg\min_{D \in \mathbb{R}^{n \times p}} P_1(D) \\ G^{i,j+1} = \arg\min_{G \in \mathbb{R}^{p \times p}} P_2(G), \end{cases} \tag{A.9}$$

where the functions P_1 and P_2 are defined as:

$$\begin{cases} P_1(D) = f(D^{i,j}) + h(D^{i,j}, G^{i,j}) \\ \qquad\qquad + \mathrm{tr}\big((D - D^{i,j})^T \nabla_D (f(D^{i,j}) + h(D^{i,j}, G^{i,j}))\big) + \dfrac{t_1}{2} \|D - D^{i,j}\|_F^2 \\ P_2(G) = g(G) + h(D^{i,j+1}, G^{i,j}) \\ \qquad\qquad + \mathrm{tr}\big((G - G^{i,j})^T \nabla_G h(D^{i,j}, G^{i,j})\big) + \dfrac{t_2}{2} \|G - G^{i,j}\|_F^2. \end{cases} \tag{A.10}$$

Proposition 5. *The process P_1 produces a sequence $(D^{i,j})$ that respects the conditions **H1, H2** and **H3**.*

Proof. The three functions $\nabla f : \mathbb{R}^{n \times p} \to \mathbb{R}^{n \times p}$, $\nabla_D h : \mathbb{R}^{n \times p} \to \mathbb{R}^{n \times p}$ and $\nabla_G h : \mathbb{R}^{p \times p} \to \mathbb{R}^{p \times p}$ are all Lipchitz continuous functions on their own domain. Then, there exists a Lipchitz constant $L_1 = L + L_D$, where L is the Lipchitz

constant for the function ∇f and L_D defined in (A.6), that is

$$\begin{aligned} & f(D^{i,j+1}) + h(D^{i,j+1}, G^{i,j}) \\ & \quad \leq f(D^{i,j}) + h(D^{i,j}, G^{i,j}) + \frac{L_1}{2}\|D^{i,j+1} - D^{i,j}\|_F^2 \\ & \qquad + \mathrm{tr}\big((D^{i,j+1} - D(i,j))^T \nabla_D (f(D^{i,j}) + h(D^{i,j}, G^{i,j})\big). \end{aligned} \tag{A.11}$$

The minimization of the problem (A.9) requires that

$$\mathrm{tr}\big((D - D^{i,j})^T \nabla_D (f(D^{i,j}) + h(D^{i,j}, G^{i,j})\big) + \tfrac{t_1}{2}\|D - D^{i,j}\|_F^2 \leq 0, \tag{A.12}$$

which makes sure the descent of the objective function. By, combining the inequality (A.12) with the inequality (A.11), we obtain the following result:

$$f(D^{i,j+1}) + h(D^{i,j+1}, G^{i,j}) + \tfrac{t_1 - L_1}{2}\|D^{i,j+1} - D^{i,j}\|_F^2 \leq f(D^{i,j}) + h(D^{i,j}, G^{i,j}) \tag{A.13}$$

The satisfaction of condition **H1** can be easily proven by choosing a t_1 greater than L_1.

We now begin to prove the condition **H2**. A big b can be found such that,

$$\|\nabla_D (f(D^{i,j}) + h(D^{i,j}, G^{i,j})\| \leq b\,\|D^{k+1} - D^{k+1}\|. \tag{A.14}$$

By considering the Lipchitz continuity of the function $D \to \nabla(f(D) + h(D, G))$ and applying the triangle inequality, the following is deduced:

$$\begin{aligned} \|\nabla_D f(D^{i,j+1}) + \nabla_D h(D^{i,j+1}, G^{i,j})\| & \leq \|\nabla_D \big(f(D^{i,j}) - h(D^{i,j}, G^{i,j})\big)\| \\ & \quad + \|\nabla_D \big(f(D^{i,j+1}) + h(D^{i,j+1}, G^{i,j})\big) \\ & \quad - \nabla_D \big(f(D^{i,j}) - h(D^{i,j}, G^{i,j})\big)\| \\ & \leq (b + L_1)\|D^{i,j+1} - D^{i,j}\|, \end{aligned} \tag{A.15}$$

which is the relative error condition **H2**.

The continuity condition **H3** is satisfied because of the continuity of the functions f and h with respect to D. □

Proposition 6. *The process P_2 produces a sequence $(G^{i,j})$ having the properties introduced in conditions **H1, H2** and **H3**.*

Proof. The minimization of the second subproblem in (A.9) assures that,

$$g(G^{i,j+1}) + \mathrm{tr}\big((G^{i,j+1} - G^{i,j})^T \nabla_G h(D^{i,j}, G^{i,j})\big) + \frac{t_2}{2}\|G^{i,j+1} - G^{i,j}\|_F^2 \leq g(G^{i,j}). \tag{A.16}$$

The function $G \to h(D, G)$ is a L_D-Lipchitz continuous function. Here, for simplification, let $L_2 = L_G$. Thus, the inequality (A.16) becomes

$$g(G^{i,j+1}) + \frac{-L_2 + t_2}{2}\|G^{i,j+1} - G^{i,j}\|_F^2 \leq g(G^{i,j}). \tag{A.17}$$

When $t_2 > L_2$, the condition **H1** is satisfied.

We prove the satisfaction of condition **H2** by using its first order necessary condition:

$$\partial g(G^{i,j+1}) + t_2(G^{i,j+1} - G^{i,j} + \tfrac{1}{t_2}\nabla_G h(D^{i,j+1}, G^{i,j})) = 0. \tag{A.18}$$

By moving the term about $G^{i,j+1} - G^{i,j}$ to the right-hand-side, we have

$$\partial g(G^{i,j+1}) + \nabla_G h(D^{i,j+1}, G^{i,j}) = -t_2(G^{i,j+1} - G^{i,j}). \tag{A.19}$$

Taking the norm of the left-hand-side and that of right-hand-side in equality (A.19), the following equality holds:

$$\|\partial g(G^{i,j+1}) + \nabla_G h(D^{i,j+1}, G^{i,j})\| = t_2\|G^{i,j+1} - G^{i,j}\|. \tag{A.20}$$

Then by applying the triangle inequality, the condition **H2** can be proven:

$$\begin{aligned}\|\partial g(G^{i,j+1}) + \nabla_G h(D^{i,j+1}, G^{i,j+1})\| &\leq \|\partial g(G^{i,j+1}) + \nabla_G h(D^{i,j+1}, G^{i,j})\| \\ &\quad + \|\nabla_G h(D^{i,j+1}, G^{i,j+1}) - \nabla_G h(D^{i,j+1}, G^{i,j})\| \\ &\leq (t_2 + L_2)\|G^{i,j+1} - G^{i,j}\| \end{aligned} \tag{A.21}$$

535 **H3** is satisfied for the continuous function h and the semicontinuous function g in $\mathcal{S}_G$. □

Proposition 7. *The iterative process P_1 and P_2 produces a sequence $((D^{i,j}, G^{i,j}))$ that satisfies the conditions **H1**, **H2** and **H3**.*

Proof. The Lipchitz continuity of the gradient of $G \to h(D, G)$ and the inequality (A.21) infer that there exists an $L' < 0$ that verifies

$$h(D^{i,j+1}, G^{i,j+1}) - h(D^{i,j+1}, G^{i,j}) \leq L'\|G^{i,j+1} - G^{i,j}\|_F^2. \tag{A.22}$$

By summing the inequalities (A.13) and (A.17), we get:

$$\begin{aligned} f(D^{i,j+1}) + h(D^{i,j+1}, G^{i,j}) + g(G^{i,j+1}) \\ + \tfrac{t_1 - L_1}{2}\|D^{i,j+1} - D^{i,j}\|_F^2 + \tfrac{t_2 - L_2}{2}\|G^{i,j+1} - G^{i,j}\|_F^2 \\ \leq f(D^{i,j}) + h(D^{i,j}, G^{i,j}) + g(G^{i,j}). \end{aligned} \tag{A.23}$$

Using the result of (A.22), the inequality becomes:

$$\begin{aligned} L_{c_1^i, c_2^i}(D^{i,j+1}, G^{i,j+1}, \boldsymbol{\lambda}^i, H^i) + \tfrac{t_1 - L_1}{2}\|D^{i,j+1} - D^{i,j}\|_F^2 \\ + \tfrac{t_2 - L_2 - 2L'}{2}\|G^{i,j+1} - G^{i,j}\|_F^2 \leq L_{c_1^i, c_2^i}(D^{i,j}, G^{i,j}, \boldsymbol{\lambda}^i, H^i). \end{aligned} \tag{A.24}$$

Setting $a = \min\left(\tfrac{t_1 - L_1}{2}, \tfrac{t_2 - L_2 - 2L'}{2}\right)$, we obtain

$$\begin{aligned} L_{c_1^i, c_2^i}(D^{i,j+1}, G^{i,j+1}, \boldsymbol{\lambda}^i, H^i) + a\|(D^{i,j+1}, G^{i,j+1}) - (D^{i,j}, G^{i,j})\|_F^2 \\ \leq L_{c_1^i, c_2^i}(D^{i,j}, G^{i,j}, \boldsymbol{\lambda}^i, H^i). \end{aligned} \tag{A.25}$$

Thus, the sequence $((D^{i,j}, G^{i,j}))_{j\in\mathbb{N}}$ satisfies the condition **H1**.

To prove the condition **H2**, it is necessary to compute the subdifferential of problem (22) of the pair of matrix variables $(D^{i,j+1}, G^{i,j+1})$, which is denoted by $\partial L_{(c_1^i, c_2^i)}(D^{i,j+1}, G^{i,j+1}, \boldsymbol{\lambda}^i, H^i)$. With the results obtained in (A.15) and (A.21), we use again the triangle inequality, then

$$\begin{aligned}
\|\partial L_{(c_1^i,c_2^i)}(D^{i,j+1}, G^{i,j+1}, \boldsymbol{\lambda}^i, H^i)\| &= \|\nabla f(D^{i,j+1}) + \nabla h(D^{i,j+1}, G^{i,j+1}) + \partial g(G^{i,j+1})\| \\
&\leq \|\nabla_D f(D^{i,j+1}) + \nabla_D h(D^{i,j+1}, G^{i,j+1})\| \\
&\quad + \|\partial g(G^{i,j+1}) + \nabla_G h(D^{i,j+1}, G^{i,j+1})\| \\
&\leq \|\nabla_D f(D^{i,j+1}) + \nabla_D h(D^{i,j+1}, G^{i,j})\| \\
&\quad + \|\partial g(G^{i,j+1}) + \nabla_G h(D^{i,j+1}, G^{i,j+1})\| \\
&\quad + \|\nabla_D h(D^{i,j+1}, G^{i,j+1}) - \nabla_D h(D^{i,j+1}, G^{i,j})\|.
\end{aligned}$$

Using the expressions of the partial derivatives in (27), then the following inequality holds

$$\begin{aligned}
\|\nabla_D h(D^{i,j+1}, G^{i,j+1}) - \nabla_D h(D^{i,j+1}, G^{i,j})\| &\leq \|\nabla h(D^{i,j+1}, G^{i,j+1}) - \nabla h(D^{i,j+1}, G^{i,j})\| \\
&\leq L\|(D^{i,j+1}, G^{i,j+1}) - (D^{i,j}, G^{i,j})\|,
\end{aligned}$$

where L is the Lipchitz constant of the function h. Combining the inequalities (A.15) and (A.20), the condition **H2** of the global sequence $(D^{i,j}, G^{i,j})_{j\in\mathbb{N}}$ is obtained

$$\begin{aligned}
\|\partial L_{(c_1^i,c_2^i)}(D^{i,j+1}, G^{i,j+1}, \boldsymbol{\lambda}^i, H^i)\| &\leq (L_1 + b)\|D^{i,j+1} - D^{i,j}\| + (t_2 + L_2)\|G^{i,j+1} - G^{i,j}\| \\
&\quad + L\|(D^{i,j+1}, G^{i,j+1}) - (D^{i,j}, G^{i,j})\|.
\end{aligned}$$

Let $t = \max(L_1 + b + L, L_2 + t_2 + L)$, then

$$\begin{aligned}
&\|\partial L_{(c_1^i,c_2^i)}(D^{i,j+1}, G^{i,j+1}, \boldsymbol{\lambda}^i, H^i)\| \\
&\leq t\|(D^{i,j+1}, G^{i,j+1}) - (D^{i,j}, G^{i,j})\|
\end{aligned}$$

The condition **H3** is straightforward by considering the continuity of the function. □

References

References

[1] J. Mairal, F. Bach, J. Ponce, et al., Sparse modeling for image and vision processing, Foundations and Trends® in Computer Graphics and Vision 8 (2-3) (2014) 85–283.

[2] M. Antonini, M. Barlaud, P. Mathieu, I. Daubechies, Image coding using wavelet transform, IEEE Transactions on image processing 1 (2) (1992) 205–220.

[3] L. Jacques, L. Duval, C. Chaux, G. Peyré, A panorama on multiscale geometric representations, intertwining spatial, directional and frequency selectivity, Signal Processing 91 (12) (2011) 2699 – 2730.

[4] K. Huang, S. Aviyente, Sparse representation for signal classification, in: NIPS, Vol. 19, 2006, pp. 609–616.

[5] J. Mairal, F. Bach, J. Ponce, Task-driven dictionary learning, IEEE Transactions on Pattern Analysis and Machine Intelligence 34 (4) (2012) 791–804.

[6] M. Aharon, M. Elad, A. Bruckstein, K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation, IEEE Transactions on signal processing 54 (11) (2006) 4311–4322.

[7] P. Honeine, Approximation errors of online sparsification criteria, IEEE Transactions on Signal Processing 63 (17) (2015) 4700 – 4709. doi:10.1109/TSP.2015.2442960.
URL http://dx.doi.org/10.1109/TSP.2015.2442960

[8] P. Honeine, Analyzing sparse dictionaries for online learning with kernels, IEEE Transactions on Signal Processing 63 (23) (2015) 6343 – 6353. doi:10.1109/TSP.2015.2457396.
URL http://dx.doi.org/10.1109/TSP.2015.2457396

[9] J. A. Tropp, Greed is good: Algorithmic results for sparse approximation, IEEE Transactions on Information theory 50 (10) (2004) 2231–2242.

[10] B. Mailhé, D. Barchiesi, M. D. Plumbley, Ink-svd: Learning incoherent dictionaries for sparse representations, in: 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2012, pp. 3573–3576.

[11] D. Barchiesi, M. D. Plumbley, Learning incoherent dictionaries for sparse approximation using iterative projections and rotations, IEEE Transactions on Signal Processing 61 (8) (2013) 2055–2065.

[12] J. Wu, M. Bai, Incoherent dictionary learning for reducing crosstalk noise in least-squares reverse time migration, Computers Geosciences 114 (2018) 11 – 21. doi:https://doi.org/10.1016/j.cageo.2018.01.010.
URL http://www.sciencedirect.com/science/article/pii/S0098300417309275

[13] C. Bao, Y. Quan, H. Ji, A convergent incoherent dictionary learning algorithm for sparse coding, in: European Conference on Computer Vision, Springer, 2014, pp. 302–316.

[14] H. Tang, X. Li, X. Zhang, D. Zhang, L. Mao, T. Liu, Coherence-regularized discriminative dictionary learning for histopathological image classification, Signal, Image and Video Processing 13 (5) (2019) 923–931.

[15] M. Rebollo, L. F. Escudero, A mixed integer programming approach to multi-spectral image classification, Pattern Recognition 9 (1) (1977) 47–57.

[16] S. Garrido-Jurado, R. Muñoz-Salinas, F. Madrid-Cuevas, R. Medina-Carnicer, Generation of fiducial marker dictionaries using mixed integer linear programming, Pattern Recognition 51 (2016) 481 – 491.

[17] X. Zhou, K. Jin, Q. Chen, M. Xu, Y. Shang, Multiple face tracking and recognition with identity-specific localized metric learning, Pattern Recognition (2017) accepted.

[18] L. Cao, F. Luo, L. Chen, Y. Sheng, H. Wang, C. Wang, R. Ji, Weakly supervised vehicle detection in satellite images via multi-instance discriminative learning, Pattern Recognition 64 (2017) 417 – 424.

[19] S. Bourguignon, J. Ninin, H. Carfantan, M. Mongeau, Exact sparse approximation problems via mixed-integer programming: Formulations and computational performance, IEEE Transactions on Signal Processing 64 (6) (2016) 1405–1419.

[20] Y. Liu, S. Canu, P. Honeine, S. Ruan, Mixed integer programming for sparse coding: Application to image denoising, IEEE Transactions on Computational Imaging 5 (2019) 1–8. doi:10.1109/TCI.2019.2896790.

[21] D. P. Bertsekas, Nonlinear programming, Athena scientific Belmont, 1999.

[22] H. Zhu, X. Zhang, D. Chu, L.-Z. Liao, Nonconvex and nonsmooth optimization with generalized orthogonality constraints: An approximate augmented lagrangian method, Journal of Scientific Computing 72 (1) (2017) 331–372.

[23] X. Chang, S. Liu, P. Zhao, D. Song, A generalization of linearized alternating direction method of multipliers for solving two-block separable convex programming, Journal of Computational and Applied Mathematics 357 (2019) 251 – 272. doi:https://doi.org/10.1016/j.cam.2019.02.028. URL http://www.sciencedirect.com/science/article/pii/S0377042719301013

[24] H. Attouch, J. Bolte, P. Redont, A. Soubeyran, Proximal alternating minimization and projection methods for nonconvex problems: An approach based on the kurdyka-łojasiewicz inequality, Mathematics of Operations Research 35 (2) (2010) 438–457.

[25] I. Ramírez, F. Lecumberry, G. Sapiro, Sparse modeling with universal priors and learned incoherent dictionaries, in: University of Minnesota. Institute for Mathematics and Its Applications, 2009.

[26] V. Abolghasemi, S. Ferdowsi, S. Sanei, Fast and incoherent dictionary learning algorithms with application to fmri, Signal, Image and Video Processing 9 (1) (2015) 147–158.

[27] V. Abolghasemi, M. Chen, A. Alameer, S. Ferdowsi, J. Chambers, K. Nazarpour, Incoherent dictionary pair learning: Application to a novel open-source database of chinese numbers, IEEE Signal Processing Letters 25 (4) (2018) 472–476.

[28] Z. Li, T. Hayashi, S. Ding, X. Li, An efficient algorithm for incoherent analysis dictionary learning based on proximal operator, in: Systems, Man, and Cybernetics (SMC), 2016 IEEE International Conference on, IEEE, 2016, pp. 003546–003549.

[29] Z. Li, S. Ding, T. Hayashi, Y. Li, Incoherent dictionary learning with log-regularizer based on proximal operators, Digital Signal Processing 63 (2017) 86–99.

[30] L. Li, S. Li, Y. Fu, Learning low-rank and discriminative dictionary for image classification, Image and Vision Computing 32 (10) (2014) 814–823.

[31] Y. Peng, L. Li, S. Liu, X. Wang, J. Li, Weighted constraint based dictionary learning for image classification, Pattern Recognition Letters . (in press) (2018) .

[32] Y. Zhang, Z. Jiang, L. S. Davis, Learning structured low-rank representations for image classification, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2013, pp. 676–683.

[33] M. Ulbrich, Z. Wen, C. Yang, D. Klockner, Z. Lu, A proximal gradient method for ensemble density functional theory, SIAM Journal on Scientific Computing 37 (4) (2015) A1975–A2002.

[34] Z. Lin, M. Chen, Y. Ma, L. Wu, The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices, Tech. rep. (2010).

[35] W. Chen, H. Ji, Y. You, An augmented lagrangian method for 1-regularized optimization problems with orthogonality constraints, SIAM Journal on Scientific Computing 38 (4) (2016) B570–B592.

[36] J. Mairal, F. Bach, J. Ponce, G. Sapiro, Online dictionary learning for sparse coding, in: Proceedings of the 26th annual international conference on machine learning, ACM, 2009, pp. 689–696.

[37] M. Elad, M. Aharon, Image denoising via sparse and redundant representations over learned dictionaries, IEEE Transactions on Image processing 15 (12) (2006) 3736–3745.

[38] K. L. Hoffman, T. K. Ralphs, Integer and combinatorial optimization, in: Encyclopedia of Operations Research and Management Science, Springer, 2013, pp. 771–783.

[39] A. Neumaier, O. Shcherbina, Safe bounds in linear and mixed-integer linear programming, Mathematical Programming 99 (2) (2004) 283–296.

[40] N. Parikh, S. Boyd, et al., Proximal algorithms, Foundations and Trends® in Optimization 1 (3) (2014) 127–239.

[41] S. G. Mallat, Z. Zhang, Matching pursuits with time-frequency dictionaries, IEEE Transactions on signal processing 41 (12) (1993) 3397–3415.

[42] R. Rubinstein, M. Zibulevsky, M. Elad, Efficient implementation of the k-svd algorithm using batch orthogonal matching pursuit, Cs Technion 40 (8) (2008) 1–15.

[43] S. S. Chen, D. L. Donoho, M. A. Saunders, Atomic decomposition by basis pursuit, SIAM review 43 (1) (2001) 129–159.

[44] R. Jenatton, J. Mairal, F. R. Bach, G. R. Obozinski, Proximal methods for sparse hierarchical dictionary learning, in: Proceedings of the 27th international conference on machine learning (ICML-10), 2010, pp. 487–494.

[45] P. Belotti, P. Bonami, M. Fischetti, A. Lodi, M. Monaci, A. Nogales-Gómez, D. Salvagnin, On handling indicator constraints in mixed integer programming, Computational Optimization and Applications 65 (3) (2016) 545–566.

[46] R. T. Rockafellar, R. J.-B. Wets, Variational analysis, Vol. 317, Springer Science & Business Media, 2009.

[47] H. Attouch, J. Bolte, B. F. Svaiter, Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward–backward splitting, and regularized gauss–seidel methods, Mathematical Programming 137 (1-2) (2013) 91–129.