Dictionary learning for sparse coding: Algorithms and convergence analysis

Chenglong Bao, Hui Ji, Yuhui Quan and Zuowei Shen

Abstract—In recent years, sparse coding has been widely used in many applications ranging from image processing to pattern recognition. Most existing sparse coding based applications require solving a class of challenging non-smooth and non-convex optimization problems. Despite the fact that many numerical methods have been developed for solving these problems, it remains an open problem to find a numerical method which is not only empirically fast, but also has mathematically guaranteed strong convergence. In this paper, we propose an alternating iteration scheme for solving such problems. A rigorous convergence analysis shows that the proposed method satisfies the global convergence property: the whole sequence of iterates is convergent and converges to a critical point. Besides the theoretical soundness, the practical benefit of the proposed method is validated in applications including image restoration and recognition. Experiments show that the proposed method achieves similar results with less computation when compared to widely used methods such as K-SVD.

Index Terms—dictionary learning, sparse coding, non-convex optimization, convergence analysis

1 Introduction

Sparse coding aims to construct succinct representations of input data, i.e. a linear combination of only a few atoms of the dictionary learned from the data itself. Sparse coding techniques have been widely used in applications, e.g. image processing, audio processing, visual recognition, clustering and machine learning [1]. Given a set of signals $Y := \{y_1, y_2, \dots, y_p\}$, sparse coding aims at finding a dictionary $D := \{d_1, d_2, \dots, d_m\}$ such that each signal $y \in Y$ can be well-approximated by a linear combination of $\{d_j\}_{j=1}^m$, i.e., $y = \sum_{\ell=1}^m c_\ell d_\ell$, and most coefficients c_ℓs are zero or close to zero. Sparse coding can be typically formulated as the following optimization problem:

$$\min_{D,\{c_i\}_{i=1}^p} \sum_{i=1}^{p} \frac{1}{2}\|y_i - Dc_i\|^2 + \lambda\|c_i\|_0, \qquad (1)$$

subject to $\|d_i\| = 1, 1 \le i \le m$. The dictionary dimension m is usually larger than the signal dimension n.

1.1 Overview of the problem

The problem (1) is a non-convex problem whose non-convexity comes from two sources: the sparsity-promoting ℓ_0-norm, and the bi-linearity between the dictionary D and codes $\{c_i\}_{i=1}^p$ in the fidelity term. Most sparse coding based applications adopt an alternating iteration scheme: for $k = 1, 2, \dots,$

(a) *sparse approximation*: update codes $\{c_i\}_{i=1}^p$ via solving (1) with the dictionary fixed from the previous iteration, i.e. $D := D^k$.

(b) *dictionary refinement*: update the dictionary D via solving (1) with codes fixed from the previous iteration, i.e. $c_i := c_i^{k+1}$ for $i = 1, \dots, p$.

Thus, each iteration requires solving two non-convex sub-problems (a) and (b).

The sub-problem (a) is an NP-hard problem [2], and thus only a sub-optimal solution can be found in polynomial time. Existing algorithms for solving (a) either use greedy strategies to obtain a local minimizer (e.g. orthogonal matching pursuit (OMP) [3]), or replace the ℓ_0-norm by its convex relaxation, the ℓ_1-norm, to provide an approximate solution (e.g. [4], [5], [6], [7]).

The sub-problem (b) is also a non-convex problem due to the existence of norm equality constraints on atoms $\{d_i\}_{i=1}^m$. Furthermore, some additional non-convex constraints on D are used for better performance in various applications, e.g. compressed sensing and visual recognition. One such constraint is an upper bound on the *mutual coherence* $\mu(D) = \max_{i\neq j} |\langle d_i, d_j\rangle|$ of the dictionary, which measures the correlation of atoms. A model often seen in visual recognition (see e.g. [8], [9], [10]) is defined as follows,

$$\min_{D,C} \|Y - DC\|^2 + \lambda\|C\|_0 + \frac{\mu}{2}\|D^\top D - I\|^2, \qquad (2)$$

subject to $\|d_i\| = 1, 1 \le i \le m$. Due to the additional term $\|D^\top D - I\|^2$, the problem (2) is harder than (1).

1.2 Motivations and our contributions

Despite the wide use of sparse coding techniques, the study of algorithms for solving (1) and its variants with rigorous convergence analysis has been scant in the literature. The most popular algorithm for solving the constrained version of (1) is the K-SVD

• *C. Bao, H. Ji, Y. Quan and Z. Shen are with the Department of Mathematics, National University of Singapore, Singapore, 119076. E-mail: matbc,matjh,matquan,matzuows@nus.edu.sg*

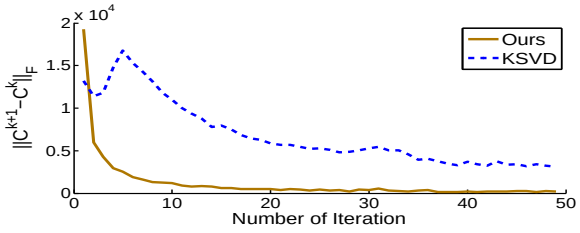


Fig. 1. Convergence behavior: the increments of the coefficient sequence C^k generated by K-SVD and by the proposed method in image denoising.

method [11], which calls OMP for solving the sparse approximation sub-problem. The OMP method is a greedy algorithm known for its high computational cost. For problem (2), existing applications usually call some generic non-linear optimization solver. Although these alternating iteration schemes generally can guarantee that the objective function value is decreasing, the generated sequence of iterates may diverge. Indeed, the sequence generated by K-SVD is not always convergent; see Fig. 1 for the convergence behavior of the sequence generated by K-SVD in a typical image denoising problem. Recently, the so-called proximal alternating method (PAM) [12] and the proximal alternating linearized method (PALM) [13] were proposed to solve a class of non-convex optimization problems, with strong convergence. However, problems considered in [12] and [13] are rather general—a direct call of these two methods is not optimal when being applied to sparse coding.

There certainly is a need for developing new algorithms to solve (1) and its variants. The new algorithms should not only be computationally efficient in practice, but also have strong convergence guaranteed by theoretical analysis, e.g. the *global convergence property*: the whole sequence generated by the method converges to a critical point of the problem.

This paper proposes fast alternating iteration schemes satisfying the global convergence property, applicable to solving the non-convex problems arising from sparse coding based applications, including (1), (2), and discriminative extensions of the K-SVD method [14], [15]. Motivated by recent work on multi-block coordinate descent [16], PAM [12] and PALM [13], we propose a multi-block hybrid proximal alternating iteration scheme, which is further combined with an acceleration technique from the implementation of the K-SVD method. The proposed dictionary learning methods have their advantages over existing dictionary learning algorithms. Unlike most existing sparse coding algorithms, e.g. K-SVD, the proposed method satisfies the global convergence property and is more computationally efficient with comparable results. Compared to some recent generic methods, e.g. PALM [13]), for solving these specific non-convex problems, the proposed dictionary learning method decreases the objective function value faster than PALM and yields better results in certain applications such as image denoising.

The preliminary version of this work appeared in [17], whereas this paper introduces several extensions. One is the extension of the two-block alternating iteration scheme to the multi-block alternating iteration scheme, which has wider applicability. Another improvement over the original is that the new scheme allows choosing either the proximal method or the linearized proximal method to update each block, which makes it easier to optimize the implementation when applied to solving specific problems. Furthermore, this paper adds more visual recognition experiments.

2 Related Work

In this section, we briefly review the most related sparse coding methods and optimization techniques.

Based on the choice of sparsity-promoting function, existing sparse coding methods fall into one of the following three categories: (a) ℓ_0-norm based methods, (b) ℓ_1-norm based methods, and (c) methods based on some other non-convex sparsity-promoting function. One prominent existing algorithm for solving ℓ_0-norm based problems is the so-called K-SVD method [11]. The K-SVD method considers the constrained version of (1) and uses an alternating iteration scheme between D and $\{c_i\}$: with the dictionary fixed, it uses the OMP method [18] to find sparse coefficients $\{c_i\}$, and then with sparse coefficients fixed, atoms in the dictionary D are sequentially updated via the SVD. The K-SVD method is widely used in many sparse coding based applications with good performance. However, the computational burden of OMP is not trivial, and thus there exists plenty of room for improvement. In addition, there is no convergence analysis for K-SVD.

Anther approach to sparse coding is using the ℓ_1-norm as the sparsity-promoting function. Many ℓ_1-norm based sparse coding methods have been proposed for various applications; see e.g. [5], [6], [19], [20]. The variational model considered in these works can be formulated as follows,

$$\min_{D\in\mathcal{D},C\in\mathcal{C}} \sum_{i=1}^{p} \frac{1}{2}\|y_i - Dc_i\|^2 + \lambda\|c_i\|_1, \qquad (3)$$

where $\mathcal{D},\mathcal{C}$ are predefined feasible sets of the dictionary D and coefficients C, respectively. It is evident that the sparse approximation sub-problem now only requires solving a convex problem. Many efficient numerical methods are available for ℓ_1-norm based sparse approximation, e.g. the homotopy method [21] used in [5] and the fast iterative shrinkage thresholding algorithm [22] used in [6]. Methods for dictionary refinement either sequentially updates atoms (e.g. [5], [6]) or simultaneously updates all atoms using the projected gradient method [7]. None of the methods mentioned above has any convergence analysis. Recently, an algorithm with convergence analysis was

proposed in [23], based on the multi-block alternating iteration scheme [16].

The ℓ_1-norm based approach has its drawbacks, e.g. it results in over-penalization on large elements of a sparse vector [24], [25]. To correct such biases caused by the ℓ_1-norm, several non-convex relaxations of ℓ_0-norm were proposed for better accuracy in sparse coding, e.g., the non-convex minimax concave in [26] and the smoothly clipped absolute deviation in [24]. Proximal algorithms have been proposed in [27], [28], [29] to solve these problems containing non-convex relaxations. Again, these methods can only guarantee that sub-problems during each iteration can be solved using some convergent method. The question of global convergence of the whole iteration scheme remains open.

The block coordinate descent (BCD) method was proposed in [30] for solving multi-convex problems, which are generally non-convex but convex in each block of variables. It is known that the BCD method may cycle and stagnate when being applied to solve non-convex problems; see e.g. [31]. A multi-block coordinate descent method was proposed in [16] which updates each block via either the original method, the proximal method, or the linearized proximal method. Its global convergence property was established for multi-convex problems, which are not applicable to the cases discussed in this paper. The recently proposed PAM [32] updates each block using the proximal method. The sub-sequence convergence property was established in [32], and the global convergence property was established in [12] for the case of two-block alternating iterations. In [13], PALM, which satisfies the global convergence property, was proposed to solve a class of non-convex and non-smooth optimization problems; it updates each block using the linearized proximal method. PALM is applicable to problems in sparse coding.

The work presented in this paper is closely related to these block coordinate descent methods. The proposed scheme is also a multi-block alternating iteration scheme, but it is different from these previous methods in several aspects, owing to it being tailored for sparse coding problems. It enables block-wise granularity in the choice of update scheme (i.e. between the proximal method and the linearized proximal method). Such flexibility is helpful to develop efficient numerical methods that are optimized for the specific problems in practical applications. In addition, motivated by the practical performance gain of an acceleration technique used in the K-SVD method, we developed an accelerated plain dictionary learning method. The proposed dictionary learning methods show their advantages over existing ones in various sparse coding based applications. The global convergence property is also established for all the algorithms proposed in this paper.

3 Numerical Algorithm

3.1 Preliminaries on non-convex analysis

In this section, we introduce some notation and preliminaries which will be used in the remainder of this paper. Vectors and matrices are denoted by lower and uppercase letters, respectively. Sets are denoted by calligraphic letters. Given a vector $y \in \mathbb{R}^n$, y_j denotes the j-th entry. For a matrix $Y \in \mathbb{R}^{m\times n}$, $Y_j \in \mathbb{R}^n$ denotes the j-th column and Y_{ij} denotes the i-th entry of Y_j. Given a matrix $Y \in \mathbb{R}^{m\times n}$, its infinity norm is defined as $\|Y\|_\infty = \max_{i,j} |Y_{ij}|$, and its ℓ_0 norm, denoted by $\|Y\|_0$, is defined as the number of nonzero entries in Y. The ℓ_2 norm of vectors and the Frobenius norm of matrices are uniformly denoted as $\|\cdot\|$. Given a positive constant $\lambda > 0$, the so-called hard-thresholding operator $T_\lambda(Y)$ is defined as

$$T_\lambda(x) = \begin{cases} x, & \text{if } |x| > \lambda; \\ \{0, \lambda\}, & \text{if } |x| = \lambda; \\ 0, & \text{otherwise,} \end{cases}$$

when applied to scalar variables. When applied to matrix Y, $T_\lambda(Y)$ applies T_λ on each entry of Y. For a set $\mathcal{S}$, its associate indicator function $\delta_\mathcal{S}$ is defined by

$$\delta_\mathcal{S}(Y) = \begin{cases} 0, & \text{if } Y \in \mathcal{S}; \\ +\infty, & \text{if } Y \notin \mathcal{S}. \end{cases}$$

For a proper and lower semi-continuous (PLS) function, denoted as $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$, the domain of f is defined by $\mathrm{dom} f = \{x \in \mathbb{R}^n : f(x) < +\infty\}$. Next, we define the critical points of a PLS function.

Definition 3.1 ([13]). *Consider a PLS function f.*

- *The* Fréchet subdifferential *of f is defined as*

$$\hat{\partial} f(x) = \left\{ u : \liminf_{y\to x, y\neq x} \frac{f(y) - f(x) - \langle u, y - x\rangle}{\|y - x\|} \geq 0 \right\}$$

  if $x \in \mathrm{dom} f$, and $\emptyset$ otherwise.
- *The* limiting subdifferential *of f is defined as*

$$\partial f(x) = \{u : \exists\ x^k \to x,\ f(x^k) \to f(x) \text{ and } u^k \in \hat{\partial} f(x^k) \to u\}.$$

- *x is a* critical point *of f if $0 \in \partial f(x)$.*

It can be seen that if x is a local minimizer of f, then $0 \in \partial f(x)$. If f is convex, then

$$\partial f(x) = \hat{\partial} f(x) = \{u | f(y) \geq f(x) + \langle u, y - x\rangle, \forall y\},$$

i.e., $0 \in \partial f(x)$ is the first order optimal condition. If $(\{c_i\}, D)$ is a critical point of (1), then it satisfies

$$(D^\top D c_i)_j = (D^\top y_i)_j, \text{ if } (c_i)_j \neq 0.$$

Definition 3.2 (Lipschitz Continuity). *A function f is a Lipschitz continuous function on the set Ω, if there exists a constant $L_0 > 0$ such that*

$$\|f(x_1) - f(x_2)\| \leq L_0 \|x_1 - x_2\| \quad \forall x_1, x_2 \in \Omega.$$

L_0 is called the Lipschitz constant.

Definition 3.3. *A function H is called m-strongly convex if and only if $H(x) - \frac{m}{2}\|x\|^2$ is convex.*

If H is m-strongly convex and differentiable, then

$$H(x) \geq H(y) + \langle \nabla f(y), x - y\rangle + \tfrac{m}{2}\|x-y\|^2, \ \forall x, y. \quad (4)$$

In the following, we introduce the so-called *proximal operator* ([33]) defined as

$$\mathrm{Prox}_{\lambda}^{f}(x) := \underset{y\in\mathbb{R}^n}{\operatorname{argmin}} f(y) + \tfrac{\lambda}{2}\|y - x\|^2. \quad (5)$$

For any PLS function F, the proximal operator defined in (5) is non-empty and compact for all $\lambda \in (0, +\infty)$; see e.g. [13]. For certain functions, the proximal operator (5) is explicitly defined, e.g., $\mathrm{Prox}_{\lambda}^{f}(x) = T_{\sqrt{2/\lambda}}(x)$ when $f = \|\cdot\|_0$.

3.2 Problem Formulation

The optimization arising from most existing sparse coding based approaches can be expressed as follows,

$$\begin{aligned} &\min_{D,C,W} Q(D,C,W) + \lambda\Psi(C) + \mu\Phi(D) + \tau\Gamma(W) \\ &\text{subject to} \quad D \in \mathcal{D}, C \in \mathcal{C}, \end{aligned} \quad (6)$$

where $D = [D_1, \ldots, D_m]$ denotes the dictionary, $C = [C_1, \ldots, C_p]$ denotes sparse codes, W denotes an optional variable such as a linear classifier and $\mathcal{D}, \mathcal{C}$ are feasible sets for D and C, respectively. The most often used feasible set $\mathcal{D}$ is the normalized dictionary

$$\mathcal{D} = \{D \in \mathbb{R}^{n\times m} : \|D_i\| = 1, i = 1, \ldots, m\}. \quad (7)$$

In this paper, we also define a feasible set for the code C for better stability of the model (6):

$$\mathcal{C} = \{C \in \mathbb{R}^{p\times m} : \|C\|_\infty \leq M\}, \quad (8)$$

where M is the upper bound, which can be set arbitrarily large to make it applicable in any application.

The terms in the objective function of (6) vary among different approaches. The fidelity term $Q(D, C, W)$ is usually based on the Frobenius norm. The term $\Psi(\cdot)$ is a sparsity promoting function such as $\|\cdot\|_0$. The term $\Phi(D)$ is some regularizer for the dictionary, e.g. a regularizer based on mutual coherence $\|D^\top D - I\|^2$. The last term is a regularizer for the optional variable, e.g. $\Gamma(W) = \|W\|^2$, used in some sparse coding based classifiers.

Example 3.4. *A list of some instances of* (6) *that have appeared in sparse coding based applications.*

(a) In the K-SVD method for sparse image modeling [34],

$$Q(D, C) = \frac{1}{2}\|Y - DC^\top\|^2; \Psi(C) = \|C\|_0, \quad (9)$$

where Y denotes the collection of image patches and $\mu = \tau = 0$.

(b) In discriminative K-SVD based recognition [15],

$$Q(D, C, W) = \tfrac{1}{2}\|Y - DC^\top\|^2 + \tfrac{\alpha}{2}\|L - WC^\top\|^2 \quad (10)$$

where Y denotes the training samples, W denotes a multi-class linear classifier and L denotes the class labels of training samples. $\Gamma(W) = \|W\|^2$, $\Psi(C) = \|C\|_0$ or $\delta_{\mathcal{K}_0}(C)$ where $\mathcal{K}_0$ denotes the set of all vectors with k_0 non-zero elements.

(c) In label consistent K-SVD based visual recognition [14], the function Q has the same form as (10) *but with different definitions of W, and L—the variable W contains both a linear classifier and a linear transform and L contains both class labels of training samples and label consistency of atoms.*

(d) In incoherent dictionary learning for signal processing and face recognition, besides the same term Q as (b), we have an additional non-convex term for lowering mutual coherence:

$$\Phi(D) = \|D^\top D - I\|^2. \quad (11)$$

In this paper, we propose a method for solving a class of ℓ_0-norm related optimization problems which covers all examples listed in Example 3.4.

3.3 Multi-block proximal alternating iterations

We first rewrite most existing sparse coding related optimization problems in the following manner:

$$\min_{x=(x_0,\ldots,x_N)} H(x) = P(x) + \sum_{i=0}^{N} r_i(x_i), \quad (12)$$

where $x_i \in \mathbb{R}^{n_i}, \ i = 0, 1, \ldots, N$. Let

$$P_i^k(\cdot) := P(x_0^k, \cdots, x_{i-1}^k, \cdot, x_{i+1}^{k-1}, \ldots, x_N^{k-1})$$

be a function with respect to variable x_i when $x_j = x_j^k, j \neq i$. Throughout this paper, we make the following assumptions about the objective function H.

Assumption 3.5. *Let $\mathrm{dom}(H) = \mathcal{X}_0 \times \mathcal{X}_1 \times \ldots \times \mathcal{X}_N$. The function $H = P + \sum_{i=1}^{N} r_i$ defined in* (12) *satisfies the following conditions:*

1) *The function H is a semi-algebraic function.*
2) *$r_i, \ i = 0, 1, \ldots, N$ are PLS functions.*
3) *$\inf H > -\infty$, $\inf P > -\infty$ and $\inf r_i > -\infty, \forall i$.*
4) *P is a C^1 function and ∇P is Lipschitz continuous on any bounded set.*
5) *For each block of variables x_i, $\nabla_i P$ is L_i-Lipschitz continuous in $\mathcal{Y}_i$ where L_i is a function of $(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_N)$, and $\mathcal{Y}_i = \{x : \|x\| \leq 2M\}$ if $\mathcal{X}_i$ is bounded in a volume with diameter M and $\mathbb{R}^{n_i}$ otherwise.*

We propose a multi-block hybrid proximal alternating method for solving the optimization problem (12), which allows updating each block of variables using either the proximal method or the linearized proximal method. In other words, there are two schemes available for updating x_i^k:

$$x_i^{k+1} \in \begin{cases} \mathrm{Prox}_{\mu_i^k}^{P_i^k + r_i}(x_i^k), & (13a) \\ \mathrm{Prox}_{\mu_i^k}^{r_i}(x_i^k - \nabla P_i^k(x_i^k)/\mu_i^k), & (13b) \end{cases}$$

During each iteration, each block can be either updated via the proximal method (13a) or via the linearized proximal method (13b). Such flexibility facilitates optimizing for performance when applied to specific problems in practice, an advantage over methods such as PALM, which updates each block using the linearized proximal method. The proposed algorithm is outlined in Alg. 1.

Algorithm 1 Multi-block hybrid proximal alternating method for solving (12)

Main Procedure:
1. Initialization: x_i^0 and μ_i^0, i=0,…,N.
2. For $k = 0, \ldots, K$,
   (a) For $0 = 1, \ldots, N$,
   $x_i^{k+1} \in \mathrm{Prox}_{\mu_i^k}^{P_i^k + r_i}(x_i^k) \cup \mathrm{Prox}_{\mu_i^k}^{r_i}(x_i^k - \nabla P_i^k(x_i^k)/\mu_i^k)$
   End
   (b) Update μ_i^{k+1}.
   End

Remark 3.6 (Parameter Updating)**.** *Let Ω_1 denote the set of variables using* (13a) *and let Ω_2 denote the set of variables using* (13b)*. Then, μ_i^k is updated according to the following criteria:*

1) *For $x_i \in \Omega_1$, $\mu_i^k \in (a, b)$ where $a, b > 0$.*
2) *For $x_i \in \Omega_2$, $\mu_i^k \in (a, b)$ and $\mu_i^k > L_i^k$, where L_i^k denotes the Lipschitz constant of ∇P_i^k.*

The details of updating μ_i^k will be discussed when applying Alg. 1 to solving specific problems.

Theorem 3.7. *[Global Convergence] The sequence $\{x^k\}$ generated by Alg. 1 converges to a critical point of* (12)*, if the following two conditions are both satisfied:*

1) *the objective function H defined in* (12) *satisfies Assumption 3.5.*
2) *the sequence $\{x^k\}$ is bounded.*

Proof: see Appendix A. □

As we will show in the next section, Theorem 3.7 is applicable to all of the cases listed in Example 3.4.

3.4 Applications of Algorithm 1 in Sparse Coding

In this section, based on Alg. 1, we present two dictionary learning methods for sparse coding based applications. The main one is the *accelerated plain dictionary learning method* which covers case (a) in Example 3.4, as well as the cases (b) and (c) with very minor modifications. It is not applicable to case (d) owing to the existence of the term $\|D^\top D - I\|^2$. The other is the *discriminative dictionary learning method* which covers all four cases in Example 3.4, including the case (d). Under the same alternating iteration scheme, these two methods differ from each other in how the blocks of variables are formed and how they are updated.

3.4.1 Accelerated plain dictionary learning

Recall that the minimization problem for plain dictionary learning can be expressed as

$$\min_{D \in \mathbb{R}^{n \times m}, C \in \mathbb{R}^{p \times m}} \tfrac{1}{2}\|Y - DC^\top\|^2 + \lambda \|C\|_0, \tag{14}$$

subject to $\|D_i\|_2 = 1, i = 1, \ldots, m$ and $\|C\|_\infty \leq M$. we split (C, D) into the following variable blocks:

$$(x_0, x_1, \ldots, x_N) := (C; D_1, D_2, \ldots, D_m).$$

Then, Alg. 1 can be applied to solve (14), in which

$$\begin{cases} r_0(C) = \lambda \|C\|_0 + \delta_{\mathcal{C}}(C), \\ r_i(D_i) = \delta_{\mathcal{D}}(D_i), i = 1, 2 \ldots, m, \\ P(C, D_1, \ldots, D_m) = \frac{1}{2}\|Y - [D_1, D_2, \ldots, D_m]C^\top\|^2, \end{cases}$$

where $\mathcal{D}, \mathcal{C}$ are defined in (7) and (8) respectively.

During each iteration, we propose the following update strategy: code C is updated via the linearized proximal method and the dictionary atoms D_is are updated via the proximal method. In other words,

$$\begin{cases} C^{k+1} \in \mathrm{Prox}_{\mu^k}^{r_0}(C^k - \nabla P_0^k(C^k)/\mu^k), & \text{(15a)} \\ D_i^{k+1} \in \mathrm{Prox}_{\lambda_i^k}^{P_i^k + r_i}(D_i^k), i = 1, 2, \ldots, m. & \text{(15b)} \end{cases}$$

Both sub-problems, (15a) and (15b), have closed-form solutions. Define

$$\begin{cases} U^k = C^k - \frac{1}{\mu^k}\nabla P_0^k(C^k), \\ C^{k,i} = (C_1^{k+1}, \ldots, C_{i-1}^{k+1}, C_{i+1}^k, \ldots, C_p^k), \\ D^{k,i} = (D_1^{k+1}, \ldots, D_{i-1}^{k+1}, D_{i+1}^k, \ldots, D_m^k), \\ R^{k,i} = Y - D^{k,i}(C^{k,i})^\top, \\ p^{k,i} = R^{k,i}C_i^k + \lambda_i^k D_i^k. \end{cases} \tag{16}$$

Then we have

Proposition 3.8. *Suppose M is chosen such that $M > \sqrt{2\lambda/\mu^k}$. Then, both* (15a) *and* (15b) *have closed form solutions which are given by*

$$\begin{cases} C^{k+1} = \mathrm{sign}(U^k) \odot \min(\left|T_{\sqrt{2\lambda/\mu^k}}(U^k)\right|, M), \\ D_i^{k+1} = (\|p^{k,i}\|_2)^{-1} p^{k,i}, \quad i = 1, , 2 \ldots, m, \end{cases} \tag{17}$$

where $\odot$ denotes Hadamard product, and $U^k, p^{k,i}$ are given by (16)*.*

Proof: By direct computation, we know minimization problems (15a) and (15b) are equivalent to

$$\begin{cases} C^{k+1} \in \mathrm{argmin}_{C \in \mathcal{C}} \frac{\mu_k}{2\lambda}\|C - U^k\|^2 + \|C\|_0, \\ D_i^{k+1} \in \mathrm{argmin}_{\|d\|_2 = 1} \frac{c_0}{2}\|d - p^{k,i}/c_0\|^2, \end{cases} \tag{18}$$

where $c_0 = \lambda_j^k + \|C_j^k\|_2^2$. Then, it can be seen that the solutions of two sub-problems are given by (17). □

Remark 3.9 (Setting of step sizes $\mu^k, \{\lambda_i^k\}$)**.** *There are $m+1$ step sizes that need to be set: μ^k in* (15a) *and $\{\lambda_i^k\}_{i=1}^m$ in* (15b)*. Let $0 < a < b$ be two constants; step size μ^k can*

be chosen as $\mu^k = \max(\rho L(D^k), a)$, *where* $\rho > 1$ *and* $L(D^k)$ *satisfies*

$$\|\nabla_C P(C^1, D^k) - \nabla_C P(C^2, D^k)\| \leq L(D^k)\|C^1 - C^2\|.$$

The step sizes λ_i^k *are simply chosen as* $\lambda_i^k \in (a, b)$. *Moreover, we can set* $L(D^k)$ *to be the maximum eigenvalue of the matrix* $D^{k\top}D^k$. *It can be seen that* $L(D^k)$ *is a bounded sequence as each column in* D *is of unit norm.*

The iterative scheme (18) can be further improved by adding an additional acceleration step in each iteration. Such an acceleration technique was first introduced in the approximated K-SVD method [35]. In the approximated K-SVD method, after updating one atom during dictionary refinement, one immediately updates the associated coefficients to further decrease the objective function value. Thus, we can also incorporate such a technique into the iterative scheme (18) for faster convergence.

Let R_I denote the sub-matrix of R whose columns are indexed in the index set I. Then, we immediately update C_i via solving the following optimization problem:

$$\widehat{C}_i^{k+1} \in \underset{\|c\|_\infty \leq M}{\operatorname{argmin}} \frac{1}{2}\|R^{k,i} - D_i^{k+1}c^\top\|^2 \tag{19}$$

subject to $c_\ell = 0, \ell \in I_i$, where $I_i = \{\ell \in \mathbb{Z}^N : C_{\ell,i}^k = 0\}$ and $R^{k,i}$ is defined in (16). The minimization problem (19) has a closed form solution given by

$$\widehat{C}_{\ell,i}^{k+1} = \operatorname{sign}(g_\ell)\min(|g_\ell|, M), \tag{20}$$

where $g = (R_{I_i}^{k,i})^\top D_i^{k+1}$ if $\ell \notin I_i$ and 0 otherwise.

A detailed description of the accelerated plain dictionary method for solving (14) is given in Alg. 2. Even with the additional acceleration step (b) (ii), Alg. 2 remains global convergent.

Theorem 3.10. *The sequence,* (C^k, D^k), *generated by Alg. 2 is bounded and converges to a critical point of* (14).

Proof: see Appendix B. □

Remark 3.11. *Alg. 2 can also be applied to solving cases (b)-(c) in Example 3.4 by including the update of block* W. *The update strategy is the same as that of the discriminative dictionary learning method discussed in the next section. However, Alg. 2 is not suitable for solving case (d) in Example 3.4. The existence of the term* $\|D^\top D - I\|^2$ *in the objective function of the case (d) makes the iterative scheme* (18) *not efficient as the sub-problems no longer have closed form solutions.*

Algorithm 2 Accelerated plain dictionary learning

1: **INPUT:** Training signals Y;
2: **OUTPUT:** Learned dictionary D;
3: **Main Procedure:**
1. Initialization: D^0, $\rho > 1$, $K \in \mathbb{N}$ and $b > a > 0$.
2. For $k = 0, 1, \ldots, K$,
(a) update sparse code C:

$$\begin{cases} \mu^k = \max(\rho\|D^{k\top}D^k\|_2, a), \\ C^{k+1} = \operatorname{sign}(U^k) \odot \min(|T_{\sqrt{2\lambda/\mu^k}}(U^k)|, M), \end{cases}$$

where U^k is defined by (16).
(b) update dictionary D: for $i = 1, \ldots, m$,
(i). Update D_i via

$$D_i^{k+1} = (\|p^{k,i}\|_2)^{-1}p^{k,i},$$

where $p^{k,i}$ is defined in (16) with $\lambda_i^k \in (a, b)$.
(ii). re-update the coefficients C_i

$$C_i^{k+1} := \widehat{C}_i^{k+1},$$

where $\widehat{C}_i^{k+1}$ is given by (20).

3.4.2 Discriminative incoherent dictionary learning

Discriminative incoherent dictionary learning is based on the following model:

$$\begin{aligned} \min_{D,C,W} \tfrac{1}{2}\|Y - DC^\top\|^2 + \tfrac{\alpha}{2}\|L - WC^\top\|^2 \\ + \tfrac{\mu}{2}\|D^\top D - I\|^2 + \lambda\|C\|_0 + \tfrac{\tau}{2}\|W\|^2, \end{aligned} \tag{21}$$

where $D \in \mathcal{D}, C \in \mathcal{C}$ and $\mathcal{D}, \mathcal{C}$ are defined in (7) and (8) respectively. Clearly, all four cases in Example 3.4 are covered by this model. We propose forming the blocks of variables by splitting (C, D, W) into

$$(W, C_1, C_2, \ldots, C_m, D_1, D_2, \ldots, D_m).$$

Recall that the term $\|D^\top D - I\|^2$ in (21) is equal to $2\sum_{i\neq j}(D_i^\top D_j)^2$ since $\|D_i\| = 1, \forall i = 1, \ldots, m$. Then we have

$$P(\cdots) = \tfrac{1}{2}\|Y - DC^\top\|^2 + \tfrac{\alpha}{2}\|L - WC^\top\|^2 + \mu\sum_{i\neq j}(D_i^\top D_j)^2$$

and

$$\begin{cases} r_0(W) = \tau\|W\|^2/2, \\ r_i(C_i) = \lambda\|C_i\|_0 + \delta_{\mathcal{C}}(C_i),\ i = 1, 2\ldots, m, \\ r_{i+m}(D_i) = \delta_{\mathcal{D}}(D_i),\ i = 1, 2\ldots, m, \end{cases} \tag{22}$$

where $\mathcal{D}, \mathcal{C}$ are defined in (7) and (8) respectively.

Based on Alg. 1, we propose the following update strategy: both the linear classifier W and the sparse code C are updated using the proximal method, and the dictionary D is updated using the linearized proximal method. In other words,

$$\begin{cases} W^{k+1} \in \operatorname{Prox}_{\gamma^k}^{P_0^k + r_0}(W^k); \\ C_i^{k+1} \in \operatorname{Prox}_{\mu_i^k}^{P_i^k + r_i}(C_i^k), i = 1, 2, \ldots, m; \\ D_i^{k+1} \in \operatorname{Prox}_{\lambda_i^k}^{r_{i+m}}(d_i^k), i = 1, \ldots, m, \end{cases} \tag{23}$$

where $d_i^k = D_i^k - \nabla P_i^k(D_i^k)/\lambda_i^k$. In (23), all three sub-problems have closed form solutions. Define

$$\begin{cases} V^k = \alpha C^{k\top}C^k + (\tau + \gamma^k)I, \\ q^{k,i} = R^{k,i\top}D_i^k + \mu_i^k C_i^k + S^{k,i\top}W_i^{k+1}, \\ D^{k,i} = (D_1^{k+1}, \ldots, D_{i-1}^{k+1}, D_i^k, \ldots, D_m^k), \end{cases} \tag{24}$$

where $R^{k,i}$ is defined in (16) and

$$S^{k,i} = L - \sum_{j<i} W_i^{k+1} C_i^{k+1\top} - \sum_{j>i} W_i^{k+1} C_i^{k\top}.$$

Then, we have

Proposition 3.12. *Suppose M is chosen such that $M > \sqrt{2\lambda/a_i^k}$, where $a_i^k = \|D_i^k\|^2 + \mu_i^k$. Then, all sub-problems in (23) have closed form solutions given by*

$$\begin{cases} W^{k+1} = (\alpha LC^k + \gamma^k W^k)(V^k)^{-1}, \\ C_i^{k+1} = \text{sign}(q^{k,i}) \odot \min(\left|T_{\sqrt{2\lambda/a_i^k}}(q^{k,i}/a_i^k)\right|, M), \\ D_i^{k+1} = (\|d^{k,i}\|_2)^{-1} d^{k,i}, \end{cases} \tag{25}$$

Proof: By direct computation, the minimization problems in (23) are equivalent to

$$\begin{cases} \min_W \frac{\alpha}{2}\|L - WC^{k\top}\|^2 + \frac{\gamma^k}{2}\|W - W^k\|^2 + \frac{\tau}{2}\|W\|^2, \\ \min_{\|c\|_\infty \le M} \frac{a_i^k}{2\lambda}\|c - q^{k,i}/a_i^k\|^2 + \|c\|_0, 1 \le i \le m, \\ \min_{\|d\|_2=1} \|d - d^{k,i}\|^2, 1 \le i \le m. \end{cases}$$

It can be seen that the solutions of the above minimization problems are given by (25). □

Remark 3.13 (Updating step sizes $\gamma^k, \mu_i^k, \lambda_i^k$)**.** *There are $2m+1$ step sizes. Let $a > b$ be two positive constants; we simply set $\gamma^k, \mu_i^k \in (a, b)$. Step sizes λ_i^k can be set as $\lambda_i^k = \max(\rho L_i^k, a)$, where L_i^k is the Lipschitz constant of ∇P_{i+m}^k in $\mathcal{X} = \{d \in \mathbb{R}^n : \|d\| \le 2\}$. Although it is not easy to compute L_i^k, $\|C_i^k\|^2 + 2\mu\|D^{k,i\top} D^{k,i}\|$ is no smaller than the Lipschitz constant L_i^k.*

A detailed description of the discriminative dictionary learning method for solving (21) is given in Alg. 3. The global convergence property of Alg. 3 can be shown using similar analysis as that of Alg. 1.

Corollary 3.14. *The sequence, (W^k, C^k, D^k), generated by Alg. 3 is bounded and converges to a critical point of (21).*

Proof: see Appendix C. □

Remark 3.15. *The acceleration step used in Alg. 2 is not helpful for further improving the performance of Alg. 3, as the coefficients C are sequentially updated in Alg. 3, while they are updated in Alg. 2 as one block.*

4 Experiments

In this section, the two proposed dictionary learning methods are evaluated in two applications: image denoising and visual recognition. Most existing sparse coding based image denoising approaches are based on model (9) of case (a) in Example 3.4. The three models in cases (b)–(d) in Example 3.4 have been used in various visual recognition applications.

Algorithm 3 Discriminative incoherent dictionary learning

1: **INPUT:** Training signals Y;
2: **OUTPUT:** Learned Incoherent Dictionary D;
3: **Main Procedure:**
1. Initialization: D^0, C^0, $\rho > 1$, and $b > a > 0$.
2. For $k = 0, 1, \ldots,$
(a) Update W: $\gamma^k \in (a, b)$ and

$$W^{k+1} = (\alpha LC^k + \gamma^k W^k)(V^k)^{-1},$$

where V^k is defined in (24).
(b) Update sparse code C_i: for $i = 1, \ldots, m$,

$$C_i^{k+1} = \text{sign}(q^{k,i}) \odot \min(|T_{\sqrt{2\lambda/a_i^k}}(q^{k,i}/a_i^k)|, M),$$

where $q^{k,i}$ is defined in (24) with $\mu_i^k \in (a, b)$.
(c) Estimate D_i: for $i = 1, \ldots, m$,

$$\begin{cases} \lambda_i^k = \max(\rho(\|C_i^k\|_2^2 + 2\mu\|D^{k,i\top} D^{k,i}\|_2), a), \\ D_i^{k+1} = d^{k,i}/\|d^{k,i}\|_2, \end{cases}$$

where $d^{k,i}$ is defined in (24).

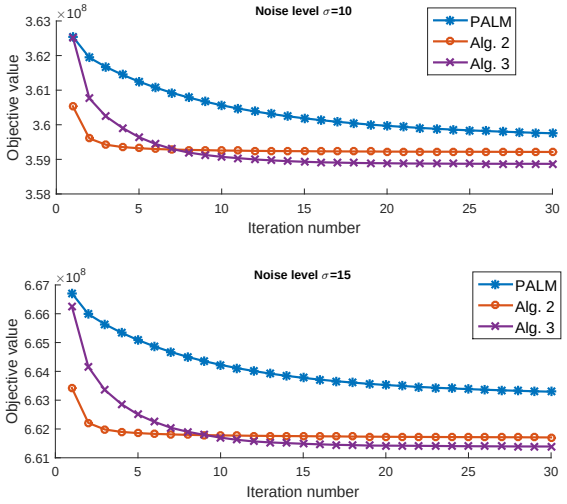


Fig. 2. Objective function value versus iteration number in sparse coding based image de-noising.

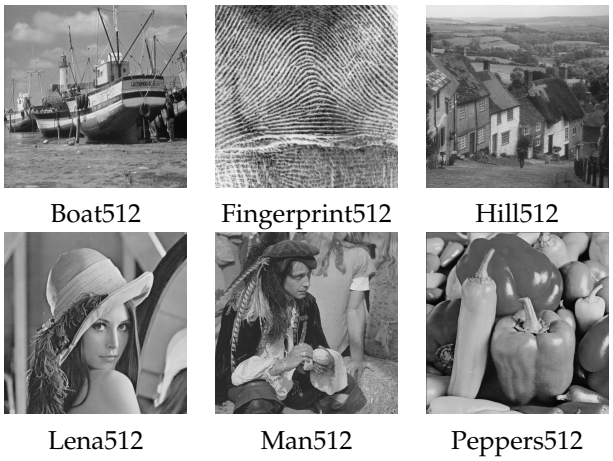


Fig. 3. Six test images for image denoising.

4.1 Image Denoising

In image denoising, we follow the same procedure in [11]. Through all the experiments in image denoising, the dimension of the dictionary is set to be the same as the K-SVD method [34], i.e. $m = 4n$, The dictionary is learned from 4×10^4 image patches randomly chosen from the input noisy image. The patch size is 8×8. The parameter λ is set to $15\sigma^2$ for the dictionary learning process, where σ denotes noise standard deviation. level, the parameter ρ is set to $1 + 10^{-3}$. All methods used in experiments were set to run for at most 30 iterations. All experiments were preformed in the Linux version of MATLAB R2011b (64 bit) running on a PC workstation with an INTEL CPU (2.4 GHZ) and 48 GB of RAM. The experiments are done on six test images (see Fig. 3) with different noise standard deviations.

Four dictionary learning methods were tested in image denoising: the K-SVD method [35][1], PALM [13], Alg. 2 and Alg. 3. Same as the K-SVD method, the dictionary is initialized using an over-complete DCT dictionary (see [11] for more details). Alg. 3 was applied to solving (1) by setting the weight of the incoherence term and the weight of discriminative term to zero and removing the corresponding computational steps. The implementation of PALM is done by splitting (C, D) into the blocks $(C, D_1, D_2, \ldots, D_m)$ and updating each block using the linearized proximal method.

4.1.1 Computational efficiency

Fig. 2 shows how fast the objective function value is reduced by each of the three methods. The K-SVD method is not included as it considers an unconstrained model whose objective function is different from the other three. It can be seen that both Alg. 2 and Alg. 3 reduce the objective function value noticeably faster than PALM. The difference between Alg. 2 and Alg. 3 is rather minor.

A comparison of running time is shown in Tab. 1. It can be seen that Alg. 2 and PALM are the fastest one, while the K-SVD method and Alg. 3 are noticeably slower. The speed of Alg. 2 and PALM on running time agrees with the theoretical computational complexity. Let K denote the average number of nonzero entries in each column of C. By direct counting, the total number of the dominant operations per iteration in Alg. 2 is

$$T_{\text{Alg. 2}} = p(2nm + 6Km + 4Kn) + 6nm^2.$$

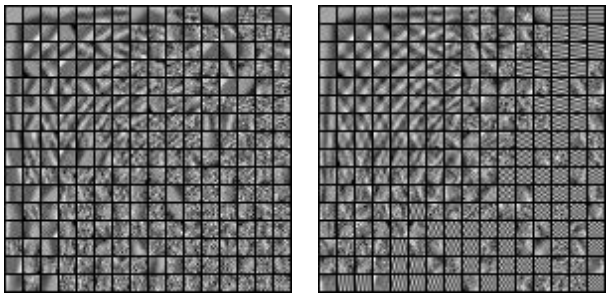
When $K \ll n \sim m \ll p$, it is about $2pnm$, while it is about $2pnm + pK^2m$ in the K-SVD method ([35]).

Overall, Alg. 2 is the best performer as it is noticeably faster at reducing the objective function value per iteration while at the same time not requiring significantly more time per iteration.

1. http://www.cs.technion.ac.il/~ronrubin/software.html

TABLE 1
Running time (seconds) versus dimension of dictonary atom

atom dimension	6x6	8x8	10x10	12x12	14x14	16x16
K-SVD	39	70	114	164	228	308
PALM	9	16	28	42	60	86
Alg. 2	10	18	30	45	66	96
Alg. 3	71	217	465	1011	1848	3094



K-SVD Alg. 2

Fig. 4. The dictionaries learned from the image "Lena512" with noise level $\sigma = 25$ using the K-SVD method and Alg. 2.

4.1.2 Quality of results

The denoising performance is measured in terms of the PSNR value. See Tab. 2 for a comparison of the PSNR values of the denoised results from five methods: the DCT-based thresholding method, the K-SVD method [34], PALM, Alg. 2 and Alg. 3. It can be seen that in terms of the average PSNR value, the K-SVD method, Alg. 2 and Alg. 3 are comparable, and they are all better than the other two methods. Fig. 4 shows the dictionaries learned from noisy image by both the K-SVD method and Alg. 2, and Fig. 5 gives a visual illustration of the results from Alg. 2. Given these results, it is evident that Alg. 2 yields results very close to K-SVD while at the same time requiring significantly less computation.

The proposed algorithms only can guarantee finding a critical point of the relating non-convex problem. Thus, same as the K-SVD method, they will yield different outcomes when using different initializations. See Tab. 3 for a comparison of the average PSNR value

noisy image ($\sigma = 25$) restored image by Alg. 2

Fig. 5. Visual illustration of a noisy image and the denoised one by Alg. 2.

TABLE 2
PSNR values of the denoised results

Image	Boat512					Fingerprint512					Hill512				
σ	5	10	15	20	25	5	10	15	20	25	5	10	15	20	25
DCT	36.79	33.49	31.34	29.96	28.90	36.34	32.25	29.68	28.29	26.85	36.54	32.93	31.11	30.02	29.00
K-SVD	37.17	33.64	31.73	30.36	29.28	36.59	32.39	30.06	28.47	27.26	36.99	33.34	31.43	30.17	29.19
PALM	37.08	33.48	31.46	30.05	28.95	36.50	32.21	29.84	28.18	26.85	36.98	33.28	31.35	30.07	29.06
Alg. 3	37.11	33.58	31.63	30.18	29.07	36.58	32.27	29.87	28.24	26.94	36.91	33.36	31.44	30.04	29.11
Alg. 2	36.97	33.53	31.65	30.31	29.18	36.59	32.35	30.03	28.44	27.17	36.94	33.31	31.29	30.02	29.06
Image	Lena512					Man512					Peppers512				
σ	5	10	15	20	25	5	10	15	20	25	5	10	15	20	25
DCT	38.29	35.25	33.39	32.03	30.96	37.16	33.12	31.01	29.65	28.67	37.06	34.48	33.02	31.89	30.95
K-SVD	38.59	35.47	33.70	32.38	31.32	37.61	33.62	31.45	30.13	29.11	37.77	34.72	32.37	32.26	31.39
PALM	38.46	35.35	33.50	32.15	31.08	37.42	33.45	31.31	29.92	28.86	37.50	34.58	33.02	31.79	30.80
Alg. 3	38.48	35.37	33.55	32.21	31.16	37.46	33.53	31.45	30.09	29.02	37.57	34.67	33.19	31.99	31.02
Alg. 2	38.49	35.41	33.57	32.25	31.19	37.46	33.47	31.43	30.02	29.00	37.68	34.64	33.22	32.14	31.18

TABLE 3
Average PSNR value of the denoised results using different initializations

Initialization	$\sigma = 5$	$\sigma = 10$	$\sigma = 15$	$\sigma = 20$	$\sigma = 25$
Alg. 2, DCT	37.36	33.79	31.87	30.53	29.46
Alg. 2, RND	37.17	33.65	31.70	30.31	29.25
Alg. 3, DCT	37.35	33.80	31.86	30.46	29.38
Alg. 3, RND	37.16	33.64	31.68	30.33	29.27

of the denoised results from the proposed methods using two different initial dictionaries: DCT and RND. DCT refers to the aforementioned over-complete DCT dictionary, and RND refers to a random subset of the collection of image patches. It can be seen that the denoising performance is influenced by how the dictionary is initialized, but such influence is not significant.

4.2 Image Recognition

In this section, the proposed methods were tested in sparse coding based recognition tasks, composed of three methods in Example 3.4, cases (b–d). Case (b) is the D-KSVD method [15], case (c) is the LC-KSVD method [14], and case (d) is the dictionary learning method with structured incoherence [8]. Both the D-KSVD method and the LC-KSVD method simultaneously perform dictionary learning and classifier training using the K-SVD method. The dictionary learning method with structured incoherence uses some standard non-linear optimization solver.

Alg. 2 was applied to solving the dictionary learning problems in both the D-KSVD method and the LC-KSVD method, and Alg. 3 was applied to solving the optimization problem in case (d). Throughout the experiments in this sub-section, the model parameters of each model were set the same [14], independent of the choice of numerical algorithm. The sparsity level was also fixed by only keeping coefficients with the k_0 largest magnitudes when thresholding.

TABLE 4
Classification accuracies (%) on four datasets.

Dataset	Training size	Case (b)			Case (c)			Case (d)
		K-SVD	Alg. 2	PALM	K-SVD	Alg. 2	PALM	Alg. 3
Yale B	1216	94.10	94.04	94.12	95.00	95.02	95.05	95.12
AR	2000	88.80	88.48	88.52	93.70	93.58	93.80	93.88
Caltech	5	49.6	49.9	49.8	54.0	54.2	54.2	54.8
	10	59.5	59.9	60.1	63.1	63.1	63.2	63.6
	15	65.1	65.2	65.0	67.7	67.5	67.6	68.3
	20	68.6	68.7	68.5	70.5	70.2	70.2	72.2
	25	71.1	70.8	71.0	72.3	72.3	72.1	72.7
	30	73.0	73.2	73.2	73.6	73.4	73.5	73.9
Scene	1500	89.1	88.8	89.2	92.9	92.7	92.9	93.1

4.2.1 *Face Recognition*

The methods are evaluated on two face datasets: Extended YaleB dataset [36] and AR face dataset [37].

Extended YaleB Dataset: the dataset [36] contains 2,414 images of 38 human frontal faces, with approximately 64 images (representing different illumination conditions and facial expressions) for each person and original images were cropped to 192×168 pixels. Following [15], we projected each face image into a 504-dim feature vector using a zero-mean random Gaussian matrix. The database was randomly split into two halves: one half containing 32 images per person used for training, and the remaining for validation.

AR Face Dataset: the dataset [37] consists of over 4000 frontal images from 126 individuals. For each individual, 26 pictures were taken in two separate sessions. Following the standard evaluation procedure from [14], [15], we used a subset of the database consisting of 2,600 images from 50 male subjects and 50 female subjects. For each person, 20 images were randomly chosen for training and the remaining images were used for test. Each image was cropped to 165×120 and then is projected itoto a 540-dim vector.

4.2.2 *Object Classification*

The Caltech-101 dataset [38] is a data set with 8677 images from 101 object categories and 467 images

from an additional background category. Same as [39], for each image, the SIFT feature based spatial pyramid feature [40] was extracted and further reduced to 3000-dim via PCA. Following standard protocol, we randomly picked $\{5, 10, 15, 20, 25, 30\}$ samples per category for training and used the rest for test.

4.2.3 Scene classification

The experiments were done on the Scene-15 dataset [40], which contains both outdoor and indoor scenes. The number of images per category varies from 210 to 410, and the resolution of each image is about 250×300. For each image, the SIFT feature based spatial pyramid feature [40] was extracted and further reduced to 3000-dim via PCA. Following the experimental settings of [14], we randomly selected 100 images per category for training and used the rest for test.

4.2.4 Results and Discussion

The results are listed in Tab. 4. It can be seen that the performance of Alg. 2 is at least comparable to that of the K-SVD method or PALM in all scenarios. Overall, the classification performance using the sparse coding model in the case (d) of Example 3.4 is better than the other three models, and Alg. 3 can be used for solving the non-convex problem in case (d) of Example 3.4.

5 Summary and Future Work

In this paper, we proposed a multi-block alternating proximal method with global convergence property for solving a class of ℓ_0-norm related optimization problems arising from sparse coding. The proposed algorithms are not only theoretically sound for non-convex problems arising from sparse coding based applications, but were also shown to be computationally efficient in practical sparse coding based applications. In future, we will further investigate stochastic methods for solving the optimization problems in sparse coding with the aim of converging to global minimizers.

References

[1] I. Tosic and P. Frossard, "Dictionary learning," *IEEE Signal Process. Mag.*, vol. 28, no. 2, pp. 27–38, 2011.

[2] A. L. Chistov and D. Y. Grigor'ev, "Complexity of quantifier elimination in the theory of algebraically closed fields," in *Mathematical Foundations of Computer Science*. Springer, 1984, pp. 17–31.

[3] A. Tropp, "Greed is good: algorithmic results for sparse approximation," *IEEE Trans. Inf. Theory*, 2004.

[4] S. Chen, D. Donoho, and M. Saunders, "Atomic decomposition by basis pursuit," *SIAM J. Sci. Comput.*, 1999.

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

[6] R. Jenatton, J. Mairal, F. R. Bach, and G. R. Obozinski, "Proximal methods for sparse hierarchical dictionary learning," in *ICML*, 2010.

[7] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Supervised dictionary learning," in *NIPS*, 2009.

[8] I. Ramirez, P. Sprechmann, and G. Sapiro, "Classification and clustering via dictionary learning with structured incoherence and shared features," in *CVPR*, 2010.

[9] B. Mailhé, D. Barchiesi, and M. D. Plumbley, "INK-SVD: Learning incoherent dictionaries for sparse representations," in *ICASSP*, 2012.

[10] D. Barchiesi and M. D. Plumbley, "Learning incoherent dictionaries for sparse approximation using iterative projections and rotations," *IEEE Trans. Signal Process.*, 2013.

[11] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, 2006.

[12] H. Attouch, J. Bolte, P. Redont, and A. Soubeyran, "Proximal alternating minimization and projection methods for nonconvex problems: An approach based on the kurdyka-lojasiewicz inequality," *Math. Oper. Res.*, vol. 35, no. 2, pp. 438–457, 2010.

[13] J. Bolte, S. Sabach, and M. Teboulle, "Proximal alternating linearized minimization for nonconvex and nonsmooth problems," *Math. Program.*, pp. 1–36, 2013.

[14] Z. Jiang, Z. Lin, and L. Davis, "Learning a dicscriminative dictionary for sparse coding via label consistent K-SVD," in *CVPR*, 2011.

[15] Q. Zhang and B. Li, "Discriminative K-SVD for dictionary learning in face recognition," in *CVPR*, 2010.

[16] Y. Xu and W. Yin, "A block coordinate descent method for multi-convex optimization with applications to nonnegative tensor factorization and completion," *SIAM J. Imaging. Sci.*, vol. 6, no. 3, pp. 1758–1789, 2013.

[17] C. Bao, H. Ji, Y. Quan, and Z. Shen, "ℓ_0 norm based dictioanry learning by proximal method with global convergence," in *CVPR*, 2014.

[18] A. Tropp, "Greed is good: algorithmic results for sparse approximation," *IEEE Trans. Inf. Theory*, 2004.

[19] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Non-local sparse models for image restoration," in *ICCV*, 2009.

[20] J. Mairal, M. Elad, and G. Sapiro, "Sparse representation for color image restoration," *IEEE Trans. Image Process.*, vol. 17, no. 1, pp. 53–69, 2008.

[21] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *Ann. statist.*, vol. 32, no. 2, pp. 407–499, 2004.

[22] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imaging. Sci.*, 2009.

[23] Y. Xu and W. Yin, "A fast patch-dictionary method for the whole image recovery," *UCLA CAM report*, 2013.

[24] J. Fan and R. Li, "Variable selection via nonconcave penalized likelihood and its oracle properties," *J. Am. Statist. Assoc.*, 2001.

[25] C.-H. Zhang, "Nearly unbiased variable selection under minimax concave penalty," *Ann. Statist.*, 2010.

[26] J. Shi, X. Ren, G. Dai, J. Wang, and Z. Zhang, "A non-convex relaxation approach to sparse dictionary learning," in *CVPR*, 2011.

[27] A.Rakotomamonjy, "Direct optimization of the dictionary learning," *IEEE Trans. Signal Process.*, 2013.

[28] S. Sra, "Scalable nonconvex inexact proximal splitting," in *NIPS*, 2012.

[29] P. Gong, C. Zhang, Z. Lu, J. Huang, and J. Ye, "A general iterative shrinkage and thresholding algorithm for non-convex regularized optimization problems," in *ICML*, 2013.

[30] J. M. Ortega and W. C. Rheinboldt, *Iterative solution of nonlinear equations in several variables*. Siam, 2000, vol. 30.

[31] M. J. Powell, "On search directions for minimization algorithms," *Math. Program.*, vol. 4, no. 1, pp. 193–201, 1973.

[32] L. Grippo and M. Sciandrone, "On the convergence of the block nonlinear gauss–seidel method under convex constraints," *Oper. Res. Lett.*, vol. 26, no. 3, pp. 127–136, 2000.

[33] R. T. Rockafellar and R. J.-B. Wets, *Variational analysis: grundlehren der mathematischen wissenschaften*. Springer, 1998, vol. 317.

[34] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Trans. Image Process*, 2006.

[35] R. Rubinstein, M. Zibulevsky, and M. Elad, "Efficient implementation of the K-SVD algorithm using batch orthogonal matching pursuit," *CS Technion*, 2008.

[36] A. S. Georghiades, P. N. Belhumeur, and D. J. Kriegman, "From few to many: Illumination cone models for face recognition under variable lighting and pose," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2001.
[37] A. Martínez and R. Benavente, "The ar face database," Computer Vision Center, Tech. Rep., 1998.
[38] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories," in *CVPR WGMBV*, 2004.
[39] L. Zhang, W. Dong, D. Zhang, and G. Shi, "Two-stage image denoising by principle component analysis with local pixel grouping," *Pattern Recogn.*, 2011.
[40] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *CVPR*, 2006.
[41] H. Attouch, J. Bolte, and B. F. Svaiter, "Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward–backward splitting, and regularized gauss–seidel methods," *Math. Program.*, vol. 137, no. 1-2, pp. 91–129, 2013.

APPENDIX A
PROOF OF THEOREM 3.7

At first, we define KL functions and semi-algebraic functions used for the convergence analysis.

Definition A.1 (Kurdyka-Łojasiewicz property [13])**.** *Let $f:\mathbb{R}^d\to(-\infty,+\infty]$ be a PLS function. The function is said to have the KL property at $\bar{x}\in \mathrm{dom}\partial f:=\{x\in\mathbb{R}^d:\partial f\neq\emptyset\}$ if there exist $\eta>0$, a neighborhood X of $\bar{x}$ and a concave and continuous function $\psi:[0,\eta)\to\mathbb{R}_+$ which satisfies $\psi(0)=0$, ψ is C^1 on $(0,\eta)$ and continuous at 0 and $\psi^{'}(s)>0,\forall s\in(0,\eta)$, such that for all*

$$x\in X\cap\{x:f(\bar{x})<f(x)<f(\bar{x})+\eta\},$$

the following inequality holds:

$$\psi^{'}(f(x)-f(\bar{x}))\mathrm{dist}(0,\partial f(x))\geq 1. \tag{26}$$

If f satisfy the KL property at each point of $\mathrm{dom}\partial f$ then f is called a KL function.

Definition A.2. *(Semi-algebraic sets and functions [12]) A subset S of $\mathbb{R}^n$ is called the semi-algebraic set if there exists a finite number of real polynomial functions g_{ij},h_{ij} such that $S=\bigcup_j\bigcap_i\{x\in\mathbb{R}^n:g_{ij}(x)=0,h_{ij}(x)<0\}$. A function f is called the semi-algebraic function if its graph $\{(x,t)\in\mathbb{R}^n\times\mathbb{R},t=f(x)\}$ is a semi-algebraic set.*

The main tool for the proof is the following theorem.

Theorem A.3 ([41])**.** *Assume $H(z)$ is a PLS function with $\inf H>-\infty$, the sequence $\{z^k\}_{k\in\mathbb{N}}$ is a Cauchy sequence and converges to a critical point of $H(z)$, if the following four conditions hold:*

(P1) Sufficiently decreasing: *there exists some positive constant ρ_1, such that*

$$H(z^k)-H(z^{k+1})\geq\rho_1\|z^{k+1}-z^k\|^2,\ \forall k.$$

(P2) Relative error*: there exists some positive constant $\rho_2>0$, such that for any $w^k\in\partial H(z^k)$,*

$$\|w^{k+1}\|_F\leq\rho_2\|z^{k+1}-z^k\|,\quad\forall k.$$

(P3) Continuity: *there exists a subsequence $\{z^{(k_j)}\}_{j\in\mathbb{N}}$ and $\bar{z}$ such that*

$$z^{(k_j)}\to\bar{z},\ H(z^{k_j})\to H(\bar{z}),\quad as\ j\to+\infty.$$

(P4) KL property*: H satisfies the KL property in its effective domain.*

By the theorem above, we only need to check that the sequence generated by Alg. 1 satisfies the conditions (P1)-(P4). Let Ω_1,Ω_2 denote the index sets of the variables that use proximal update (13a), linearized proximal update(13b) respectively, and define

$$\begin{cases} P_i^k(\cdot):=P(x_0^{k+1},\cdots,x_{i-1}^{k+1},\cdot,x_{i+1}^k,\ldots,x_N^k),\\ \widetilde{P}_i^k(\cdot):=P_i^k(x_i^k)+\langle\nabla P_i^k(x_i^k),\cdot-x_i^k\rangle.\end{cases}$$

Condition (P1). Before proceeding, we first present a lemma about continuous differentiable functions which can be derived from [13, Lemma 3.1].

Lemma A.4. *Let $h:\mathbb{R}^n\to\mathbb{R}$ be a continuously differentiable function and ∇h is L_h-Lipschitz continuous in $\Omega=\{x:\|x\|\leq M\}$. Then, we have*

$$h(u)\leq h(v)+\langle u-v,\nabla h(v)\rangle+\frac{L_h}{2}\|u-v\|_F^2,\forall u,v\in\Omega_1,$$

where $\bar{\Omega}=\{x:\|x\|\leq M/2\}$.

Proof: For any $x,y\in\bar{\Omega}$, by the triangular inequality, we know $x+\alpha y\in\Omega$ where $0\leq\alpha\leq 1$. Define $g(\alpha)=h(x+\alpha y)$. Then, we have

$$h(x+y)-h(x)=g(1)-g(0)=\int_0^1\frac{dg}{d\alpha}(\alpha)d\alpha$$
$$\leq\int_0^1 y^\top\nabla h(x)d\alpha+|\int_0^1 y^\top(\nabla h(x+\alpha y)-\nabla h(x))d\alpha|$$
$$\leq y^\top\nabla h(x)+\|y\|\int_0^1 L_h\alpha\|y\|d\alpha=y^\top\nabla h(x)+L_h\|y\|^2/2$$

which completes the proof. □

When $i\in\Omega_1$, the term $P_i^k(x_i^k)+r_i(x_i^k)$ is no less than

$$P_i^k(x_i^{k+1})+r_i(x_i^{k+1})+\frac{\mu_i^k}{2}\|x_i^{k+1}-x_i^k\|^2. \tag{27}$$

When $i\in\Omega_2$, the term $P_i^k(x_i^k)+r_i(x_i^k)$ is no less than

$$\widetilde{P}_i^k(x_i^k)+r_i(x_i^{k+1})+\frac{\mu_i^k}{2}\|x_i^{k+1}-x_i^k\|^2. \tag{28}$$

By the Lipschitz continuity of $\nabla_i P$ and lemma A.4,

$$P_i^k(x_i^{k+1})\leq\widetilde{P}_i^k(x_i^k)+\frac{L_i^k}{2}\|x_i^{k+1}-x_i^k\|^2. \tag{29}$$

The combination of (28) and (29) leads to the fact that $P_i^k(x_i^k)+r_i(x_i^k)$ is no less than

$$P_i^k(x_i^{k+1})+r_i(x_i^k)+\frac{\mu_i^k-L_i^k}{2}\|x_i^{k+1}-x_i^k\|^2. \tag{30}$$

Summing up (27) and (30) gives the term

$$H(x^k)-H(x^{k+1})=P_0^k(x_0^k)-P_N^k(x_N^{k+1})$$

is no less than

$$\sum_{i\in\Omega_1}\frac{\mu_i^k}{2}\|x_i^{k+1}-x_i^k\|^2+\sum_{i\in\Omega_2}\frac{\mu_i^k-L_i^k}{2}\|x_i^{k+1}-x_i^k\|^2,$$

as $P_{i+1}^k(x_{i+1}^k)=P_i^k(x_i^{k+1})$. Let $\rho_1=\min\{(\mu_i^k-L_i^k)/2: k\in\mathbb{N}, i\in\Omega_2\}$. Then, $\rho_1>0$ since $\mu_i^k>L_i^k$ which gives $\mu_i^k\in(a,b)$. Thus, Condition (P1) is satisfied.

Condition(P2). If $i\in\Omega_1$, we have

$$0\in\nabla P_i^k(x_i^{k+1})+\mu_i^k(x_i^{k+1}-x_i^k)+\partial r_i(x_i^{k+1}). \quad (31)$$

Define $V_i^k=-\nabla P_i^k(x_i^{k+1})-\mu_i^k(x_i^{k+1}-x_i^k)$. Then,

$$\omega_i^k:=V_i^k+\nabla_i P(x^{k+1})\in\partial_i H(x^{k+1}).$$

If $\{x^k\}$ is bounded, since ∇P is Lipschitz continuous on any bounded set, there exists $M_1>0$ such that

$$\|w_i^k\|\le M_1\|x^{k+1}-x^k\|,\forall i\in\Omega_1. \quad (32)$$

Similarly, if $i\in\Omega_2$, we have

$$0\in\nabla P_i^k(x_i^k)+\mu_i^k(x_i^{k+1}-x_i^k)+\partial r_i(x_i^{k+1}).$$

Define $V_i^k=-\nabla P_i^k(x_i^k)-\mu_i^k(x_i^{k+1}-x_i^k)$. Then,

$$\omega_i^k:=V_i^k+\nabla_i P(x^{k+1})\in\partial_i H(x^{k+1}).$$

By the boundedness of $\{x^k\}$ and Lipschitz continuity of ∇P, we know there exists $M_2>0$ such that

$$\|w_i^k\|\le M_2\|x^{k+1}-x^k\|,\forall i\in\Omega_2. \quad (33)$$

Define $M=N\max(M_1,M_2)$. Then (32) and (33) lead to $\|\omega^k\|\le M\|x^{k+1}-x^k\|$, where $\omega^k=(\omega_1,\ldots,\omega_N)$ such that $\omega_i=\omega_i^k$ when $i\in\Omega_1$ or $i\in\Omega_2$. Therefore, Condition (P2) is satisfied.

Condition (P3). Consider two convergent subsequences $x^{k_j}\to\bar{x}$ and $x^{k_j-1}\to\bar{y}$ of a bounded sequence $\{x^k\}$. We first show that $\bar{x}=\bar{y}$. Given any positive integer j, from Condition (P1), we have

$$H(x^0)-H(x^{j+1})>\rho\sum_{k=0}^{j}\|x^k-x^{k+1}\|^2. \quad (34)$$

Since $\{H(x^j)\}$ is decreasing and $\inf H>-\infty$, there exist some $\bar{H}$ such that $H(x^j)\to\bar{H}$ as $j\to+\infty$. Let $j\to+\infty$ in (34), we have

$$\sum_{k=0}^{+\infty}\|x^k-x^{k+1}\|^2<H(x^0)-\bar{H}<+\infty.$$

which implies $\lim\|x^k-x^{k-1}\|=0$. Then, we have $\lim\|x^{k_j+1}-x^{k_j}\|=0$ and $\bar{x}=\bar{y}$.

Denote $\bar{x}=(\bar{x}_1,\ldots,\bar{x}_N)$. For $i\in\Omega_1$, we have for all $x_i\in\mathcal{X}_i$

$$\begin{aligned}&P_i^k(x_i^{k+1})+r_i(x_i^{k+1})+\frac{\mu_i^k}{2}\|x_i^{k+1}-x_i^k\|^2\\&\le P_i^k(x_i)+r_i(x_i)+\frac{\mu_i^k}{2}\|x_i-x_i^k\|^2.\end{aligned} \quad (35)$$

Let $k=k_j-1$, $x_i=\bar{x}_i$ in (35) and $j\to+\infty$, we have then $\limsup_{j\to+\infty}r_i(x_i^{k_j})\le r_i(\bar{x}_i)$.

For $i\in\Omega_2$, we have

$$\begin{aligned}&\widetilde{P}_i^k(x_i^{k+1})+r_i(x_i^{k+1})+\frac{\mu_i^k}{2}\|x_i^{k+1}-x_i^k\|^2\\&\le\widetilde{P}_i^k(x_i)+r_i(x_i)+\frac{\mu_i^k}{2}\|x_i-x_i^k\|^2.\end{aligned} \quad (36)$$

Let $k=k_j-1$, $x_i=\bar{x}_i$ in (36) and $j\to+\infty$, by the Lipschitz continuity of ∇P and Condition (P1), we have $\limsup_{j\to+\infty}r_i(x_i^{k_j})\le r_i(\bar{x}_i)$. Together with the fact that r_i is lower semi-continuous, we have $\lim_{j\to+\infty}r_i(x_i^{k_j})=r_i(\bar{x}_i),\forall i=1,2\ldots,N$. Therefore, by the continuity of P, we conclude that

$$\lim_{j\to+\infty}P(x^{k_j})+\sum_{i=1}^{N}r_i(x_i^{k_j})=P(\bar{x})+\sum_{i=1}^{N}r_i(\bar{x}_i).$$

4. Condition (P4). The function H in Theorem 3.7 is a semi-algebraic function [13], which automatically satisfies the so-called KL property according to the following theorem in [13].

Theorem A.5. *([13]) Let f is a PLS and semi-algebraic function, then f satisfies the KL property in* $\mathrm{dom}f$.

APPENDIX B
PROOF OF THEOREM 3.10

Due to space limitation, we only prove the convergence of Alg. 2 for $m=1$. The proof can be easily extended to the case of $m>1$ with small modifications. For $m=1$, the objective function in (14) can be rewritten as $H(c,d)=F(c)+Q(c,d)+G(d), c\in\mathbb{R}^n, d\in\mathbb{R}^p$, where F,Q,G are defined as

$$\begin{cases}F(c)=\sum_{i=1}^{p}F_i(c_i)=\sum\limits_{i=1}^{p}\lambda\|c_i\|_0+\delta_{\mathcal{X}}(c_i),\\ G(d)=\delta_{\mathcal{U}}(d),\quad Q(c,d)=\frac{1}{2}\|Y-dc^\top\|^2.\end{cases} \quad (37)$$

where $\mathcal{U}=\{d:\|d\|=1\}$ and $\mathcal{X}=\{c:|c_i|\le M\}$. For a vector c, let c_I denote the sub-vector of c contains the entries indexed in I. Define $Q_d^k=Q(v^k,d)$. Then, Alg. 2 can be re-written as

$$\begin{cases}v^k\in\mathrm{Prox}_{2\lambda/\mu^k}^{F}(c^k-\dfrac{1}{\mu^k}\nabla_c Q(c^k,d^k)), & (38a)\\ d^{k+1}\in\mathrm{Prox}_{\lambda^k}^{G+Q_d^k}(d^k), & (38b)\\ c^{k+1}:\ c_{I_k^c}^{k+1}=0\text{ and }c_{I_k}^{k+1}\in\underset{\tilde{c}\in\mathcal{X}}{\mathrm{argmin}}\ f_i^k(\tilde{c}), & (38c)\end{cases}$$

where $I_k=\{i:v_i^k\neq0\}$, $\widehat{Y}=Y_{I_k}$ and $f^k(\tilde{c})=\frac{1}{2}\|\widehat{Y}-d^{k+1}\tilde{c}^\top\|^2$. It is noted that f^k is strongly convex. Define $z^k=(c^k,d^k)$ and

$$u_{k+1}=\|v^k-c^k\|+\|c^{k+1}-v^k\|+\|d^{k+1}-d^k\|.$$

In the next, we introduce a series of lemmas which are the main ingredients of the proof.

Lemma B.1. *Let $\{z^k\}$ denote the sequence generated by* (38a)-(38c). *Then, there exists $\rho>0$ such that*

$$H(z^k)-H(z^{k+1})\ge\rho u_{k+1}^2 \quad (39)$$

and

$$\sum_{k=1}^{\infty} u_k^2 < \infty, \quad \lim_{k\to+\infty} u_k = 0. \tag{40}$$

Proof: By (27) and (30), the updates (38a) and (38b) imply that there exists $\rho_1 > 0$ such that

$$\begin{cases} H(c^k, d^k) - H(v^k, d^k) \geq \rho_1 \|c^{k+\frac{1}{2}} - c^k\|^2, \\ H(v^k, d^k) - H(v^k, d^{k+1}) \geq \rho_1 \|d^{k+1} - d^k\|^2, \end{cases} \tag{41}$$

From (38c) and (4), we have

$$f^k(v_{I_k}^k) - f^k(c_{I_k}^{k+1}) \geq \frac{1}{2}\|v_{I_k}^k - c_{I_k}^{k+1}\|^2 = \frac{1}{2}\|c^{k+1} - v^k\|^2$$

as $\|d^{k+1}\|_2 = 1$ and $c_{I_k^c}^{k+1} = v_{I_k^c}^k$, which implies

$$H(v^k, d^{k+1}) - H(c^{k+1}, d^{k+1}) \geq \frac{1}{2}\|c^{k+1} - v^k\|^2.$$

Together with (41), we have

$$H(z^k) - H(z^{k+1}) \geq \rho u_{k+1}^2 \tag{42}$$

by the Cauchy-Schwarz inequality, where $\rho = \min(\rho_1, 1/2)/3$. Thus, $\{H(Z^k)\}$ is a decreasing sequence and $H(Z) \geq 0$. Let $\bar{H}$ be the limit of $H(z^k)$. Telescoping the inequality (42) gives

$$\sum_{k=1}^{\infty} u_k^2 \leq \frac{1}{\rho}(H(z^0) - \bar{H}) < \infty,$$

which leads to $\lim_{k\to+\infty} u_k = 0$. □

Lemma B.2. *Let $\{z^k\}$ denote the sequence generated by* (38a)-(38c)*. Then, there exists*

$$w^{k+1} := (w_c^{k+1}, w_d^{k+1}) \in \partial H(z^{k+1})$$

and $M > 0$, such that $\|w^{k+1}\| \leq M u_{k+1}$.

Proof: By (31), the scheme (38b) implies

$$-\nabla_d Q(v^k, d^{k+1}) - \lambda^k(d^{k+1} - d^k) \in \partial G(d_i^{k+1}),$$

Then, we have

$$\begin{aligned} \omega_d^{k+1} :=& \nabla_d Q(z^{k+1}) - \nabla_d Q(v^k, d^{k+1}) - \lambda^k(d^{k+1} - d^k) \\ \in& \nabla_d Q(z^{k+1}) + \partial G(d_i^{k+1}) = \partial_d H(z^{k+1}), \end{aligned}$$

and

$$\|\omega_d^{k+1}\| \leq L\|c^{k+1} - v^k\| + b\|d^{k+1} - d^k\| \tag{43}$$

by the Lipschitz continuity of ∇Q. Additionally, we have $-(\nabla_c Q(z^k) + \mu^k(v^k - c^k)) \in \partial F(v^k)$ from (38a), and

$$\partial_{c_i} F(c_i^{k+1}) = \partial_{c_i} F(v^k), \forall i \in I_k^c,$$

from (38c). So, define

$$\omega_{c_{I_k^c}}^{k+1} := \partial_{c_{I_k^c}} Q(z^{k+1}) - \partial_{c_{I_k^c}} Q(z^k) - \mu^k(v_{I_k^c}^k - c_{I_k^c}^k)),$$

then, $\omega_{c_{I_k^c}}^{k+1} \in \partial_{c_{I_k^c}} H(z^{k+1})$. By the Lipschitz continuity of ∇H and the boundedness of z^k, there exists $M_1 > 0$ such that

$$\|\omega_{c_{I_k^c}}^{k+1}\| \leq M_1 u_{k+1}. \tag{44}$$

For any $i \in I_k$ we have

$$-\partial_{c_i} f^k(c_{I_k}^{k+1}) \in \partial_{c_i} F(c^{k+1}),$$

as $0 \in \partial\|x\|_0, \forall x$. Consequently, we have

$$\omega_{c_{I_k}}^{k+1} := \partial_{c_{I_k}} Q(z^{k+1}) - \partial_{c_{I_k}} f^k(c_{I_k}^{k+1}) \in \partial_{c_{I_k}} H(z^{k+1}).$$

It is easy to know that $\omega_{c_{I_k}}^{k+1}$=0. Let

$$\omega^{k+1} = (\omega_c^{d+1}, \omega_d^{k+1}) = (\omega_{c_{I_k}}^{k+1}, \omega_{c_{I_k^c}}^{k+1}, \omega_d^{k+1}).$$

Then, from (43), (44), we have $\omega^{k+1} \in \partial H(z^{k+1})$ and $\|\omega^{k+1}\| \leq M u_{k+1}$, where $M = \max(L, b, M_1)$. □

Lemma B.3. *Let $\{z^k\}$ denote the sequence generated by* (38a)-(38c)*. For any convergent sub-sequence $z^{k_j} \to \bar{z} = (\bar{c}, \bar{d})$ of $\{z^k\}$, then $\bar{z}$ is a critical point of* (14)*.*

Proof: Recall that $\lim_{j\to+\infty} u_{k_j} = 0$. Thus, $v^{k_j-1} \to \hat{c}$, $z^{k_j-1} \to \hat{z}$ and $\hat{c} = \bar{c}, \hat{z} = \bar{z}$. From (38a), we have

$$\widehat{Q}_c^{k_j-1}(v^{k_j-1}) + F(v^{k_j-1}) \leq \widehat{Q}_c^{k_j-1}(c) + F(c), \tag{45}$$

where

$$\widehat{Q}_c^k(c) = \langle \nabla_c Q(c, d^k), c - c^k \rangle + \frac{\mu^k}{2}\|c - c^k\|^2.$$

Replacing c by $\bar{c}$ in (45) and let $j \to +\infty$, we have

$$\limsup_{j\to+\infty} F(v^{k_j-1}) \leq F(\bar{c}),$$

by the Lipschitz continuity of ∇Q, the boundedness of $\{z^k\}$ and (40). As $\|c^{k_j}\|_0 \leq \|v^{k_j-1}\|_0$, we have

$$\limsup_{j\to+\infty} F(c^{k_j}) \leq \limsup_{j\to+\infty} F(v^{k_j-1}) \leq F(\bar{c}).$$

Together with the fact that F is lower semi-continuous, we have $\lim_{j\to+\infty} F(c^{k_j}) = F(\bar{c})$. Since $d^k \in \mathcal{U}$ and Q is continuous, $\lim_{j\to+\infty} H(z^{k_j}) = H(\bar{z})$. From Lemma B.2, there exists $\omega^{k_j} \in \partial H(z^{k_j})$ such that $\omega^{k_j} \to 0$ by (40), which means $\bar{z}$ is a critical point of (14). The proof is complete. □

The next lemma [13] presents a uniformized KL property related to KL functions which will be used to prove the global convergence of the sequence $\{z^k\}$.

Lemma B.4 ([13])**.** *Let Ω be a compact set and let σ be a PLS function. Assume that σ is constant on Ω and satisfies the KL property at each point of Ω. Then, there exist $\epsilon > 0$, $\eta > 0$ and a concave $\psi : [0, \eta] \to \mathbb{R}_+$ with $\psi(0) = 0$, $\psi'(s) > 0$ for all $s \in (0, \eta)$ and $\psi \in C^1$, continuous at 0, such that for all $\bar{u}$ in Ω and all u in the following intersection:*

$$\{u : \mathrm{dist}(u, \Omega) \leq \epsilon\} \cap \{u : \sigma(\bar{u}) < \sigma(u) \leq \sigma(\bar{u}) + \eta\},$$

one has, $\psi'(\sigma(u) - \sigma(\bar{u}))\mathrm{dist}(0, \partial\sigma(u)) \geq 1$.

The global convergence of the sequence $\{z^k\}$ is established the following theorem, whose proof is similar to that of [13, Theorem 1].

Theorem B.5. *The sequence $\{z^k\}$ generated by* (38a)-(38c) *converges to a critical point of H.*

Proof: As shown in Appendix C, $H(z)$ is a semi-algebraic function and thus is a KL function. Let $w(z^0)$ be the set of limit points of the sequence $\{z^k\}$ starting from the point z^0. By the boundedness of $\{z^k\}$, $w(z^0)$ is a nonempty, compact set as $w(z^0) = \bigcap_{q\in\mathbb{N}} \overline{\bigcup_{k\geq q}\{z^k\}}$. Furthermore, as $H(z^k)$ is decreasing and bounded below, there exists H such that $\bar{H} = \lim_{k\to+\infty} H(z^k)$. Then, for any $\bar{z} \in w(z^0)$, there exists a sub-sequence z^{k_j} converging to $\bar{z}$ as $j \to +\infty$. First of all, we know $H(z^{k_j})$ converges to $\bar{H}$ as $H(z^k)$ converges to $\bar{H}$. From lemma B.3, we have $\bar{H} = \lim H(z^{k_j}) = H(\bar{Z})$. It implies that $H(z) = \bar{H}$ for all $z \in w(z^0)$.

In the next, we assume $H(z^k) < H(\bar{z})$. Otherwise, assume $H(z^{k_0}) = \bar{H}$, from the decreasing property of the sequence $\{z^k\}$, we know $z_k = z_{k_0}$ for all $k > k_0$. Then, from lemma B.4 with $\Omega = w(z^0)$, there exists ℓ, such that for $k > \ell$, we have

$$\psi^{'}(H(z^k) - H(\bar{z}))\mathrm{dist}(0, \partial H(z^k)) \geq 1. \tag{46}$$

From Lemma B.2, we have

$$\psi^{'}(H(z^k) - H(\bar{z})) \geq \frac{1}{M}u_k, \tag{47}$$

where $M > 0$. Meanwhile, as ψ is concave, we have

$$\begin{aligned}&\psi(H(z^k) - H(\bar{z})) - \psi(H(z^{k+1}) - H(\bar{z}))\\ &\geq \psi^{'}(H(z^k) - H(\bar{z}))(H(z^k) - H(z^{k+1})).\end{aligned} \tag{48}$$

Define $\triangle_{p,q} := \psi(H(z^p) - H(\bar{z})) - \psi(H(z^q) - H(\bar{z}))$. From lemma B.1, (47) and (48), there exists $c_0 > 0$, such that for $k > \ell$, $\triangle_{k,k+1} \geq u_{k+1}^2/c_0 u_k$. Thus,

$$2u_{k+1} \leq u_k + c_0\ \triangle_{k,k+1} \tag{49}$$

by Cauchy-Schwartz inequality. Summing (49) over i, we have

$$2u_{k+1} + \sum_{i=l+1}^{k} u_i \leq u^{\ell} + C\ \triangle_{\ell+1,k+1},$$

as $\triangle_{p,q} + \triangle_{q,r} = \triangle_{p,r}$. Then, for any $k > \ell$,

$$\sum_{i=\ell+1}^{k} u_i \leq u_\ell + C\psi(H(z^{\ell+1}) - H(\bar{z})).$$

Therefore,

$$\sum_{k=1}^{\infty} \|z^{k+1} - z^k\| \leq \sum_{k=1} u_k < \infty,$$

which implies that $\{z^k\}$ is a convergent sequence. Since $z^{k_j} \to \bar{z}, j \to +\infty$, we have $z^k \to \bar{z}$. □

APPENDIX C
PROOF OF COROLLARY 3.14

Let $Z^k := (C^k, D^k)$ to be the sequence generated by Alg. 3. First of all, Z^k is a bounded sequence as $D^k \in \mathcal{D}$ and $C^k \in \mathcal{C}$. Moreover, it can be seen that all conditions in Assumption 3.5 are satisfied. It is noted that H is a semi-algebraic function as polynomial functions are semi-algebraic, since $\|L - WC^\top\|^2 + \|W\|^2$ and $\|D^\top D - I\|^2$, are semi-algebraic, which is true as both are polynomials, $\mathcal{D}, \mathcal{C}$ are semi-algebraic set and $\|\cdot\|_0$ is semi-algebraic [13, Example 5.2]).