Efficient ADMM-Based Algorithms for Convolutional Sparse Coding

Farshad G. Veshki and Sergiy A. Vorobyov, *Fellow, IEEE*

***Abstract*—Convolutional sparse coding improves on the standard sparse approximation by incorporating a global shift-invariant model. The most efficient convolutional sparse coding methods are based on the alternating direction method of multipliers and the convolution theorem. The only major difference between these methods is how they approach a convolutional least-squares fitting subproblem. In this letter, we present a novel solution for this subproblem, which improves the computational efficiency of the existing algorithms. The same approach is also used to develop an efficient dictionary learning method. In addition, we propose a novel algorithm for convolutional sparse coding with a constraint on the approximation error. Source codes for the proposed algorithms are available online.**

***Index Terms*—Convolutional sparse coding, constrained sparse approximation, dictionary learning, alternating direction method of multipliers.**

I. Introduction

Sparse representations are widely used in various applications of signal and image processing [1]–[6]. The sparse synthesis model admits that natural signals can be approximated using a linear combination of only a small number of atoms (columns) of a dictionary (matrix). A common formulation of the sparse coding problem is given as

$$\underset{\boldsymbol{x}}{\text{minimize}}\ \|\boldsymbol{x}\|_1 \quad \text{s.t.} \quad \|\boldsymbol{D}\boldsymbol{x}-\boldsymbol{s}\|_2^2 \leq \epsilon, \tag{1}$$

where $\boldsymbol{D}$ is the dictionary, $\boldsymbol{x} \in \mathbb{R}^m$ is the sparse representation vector, $\boldsymbol{s} \in \mathbb{R}^n$ is the signal, and ϵ represents the upper bound on the approximation error. Moreover, $\|\cdot\|_1$ and $\|\cdot\|_2$ denote the ℓ_1-norm and the Euclidean norm, respectively. The problem of finding sparsity promoting dictionaries is called dictionary learning [7], [8].

The applications of sparse representations and dictionary learning usually involve either or both extraction and estimation of local features. Typically, this is handled by a prior decomposition of the original signal into vectorized overlapping blocks (*e.g.*, patches in image processing). However, this strategy results in multi-valued representations. Moreover, since the relationships among neighboring blocks are ignored, dictionaries learned using this approach contain shifted versions of the same features.

Convolutional sparse coding (CSC) incorporates a single-valued and shift-invariant model that represents the entire signal. In this model, the product $\boldsymbol{D}\boldsymbol{x}$ in the standard sparse coding problem is replaced by a sum of convolutions. The convolutional form of the standard sparse coding problem (1) can be written as follows

$$\underset{\{\boldsymbol{x}_k\}_{k=1}^K}{\text{minimize}} \sum_{k=1}^K \|\boldsymbol{x}_k\|_1 \quad \text{s.t.} \quad \Big\|\sum_{k=1}^K \boldsymbol{d}_k * \boldsymbol{x}_k - \boldsymbol{s}\Big\|_2^2 \leq \epsilon, \tag{2}$$

where $*$ denotes the convolution operator (usually, with "$same$" padding) and $\boldsymbol{x}_k \in \mathbb{R}^n$ and $\boldsymbol{d}_k \in \mathbb{R}^m,\ k=1,\ldots,K$, are the sparse coefficient maps and the dictionary filters, respectively. Several applications have shown that CSC outperforms its standard version in modeling natural signals, such as audio and images [9]–[20].

A majority of available CSC algorithms, including [21]–[29], are based on the alternating direction method of multipliers (ADMM) [30]. ADMM breaks the CSC problem into two main sub-problems, one of which is a sparse approximation problem which can be efficiently addressed using a shrinkage operator, and the other entails a convolutional least-squares regression. An efficient solution to the second sub-problem based on the convolution theorem and the Sherman-Morrison formula is given in [23]. CSC problem (2) is typically addressed by solving its unconstrained equivalent

$$\underset{\{\boldsymbol{x}_k\}_{k=1}^K}{\text{minimize}}\ \frac{1}{2}\Big\|\sum_{k=1}^K \boldsymbol{d}_k * \boldsymbol{x}_k - \boldsymbol{s}\Big\|_2^2 + \lambda \sum_{k=1}^K \|\boldsymbol{x}_k\|_1, \tag{3}$$

where $\lambda > 0$ is a regularization parameter. It is known that there is a unique λ for each ϵ. However, the appropriate value of λ also depends on $\boldsymbol{s}$ and $\{\boldsymbol{d}_k\}_{k=1}^K$. Thus, despite being more convenient to solve, the unconstrained reformulation (3) introduces undesirable data dependency to the CSC algorithm.

A common approach for convolutional dictionary learning (CDL) entails optimizing the filters and the sparse coefficient maps using a batch of P training signals [22]–[25]. This problem can be formulated as

$$\underset{\{\boldsymbol{x}_k^p\}_{k=1}^K,\{\boldsymbol{d}_k\}_{k=1}^K}{\text{minimize}} \sum_{p=1}^P \left(\frac{1}{2}\Big\|\sum_{k=1}^K \boldsymbol{d}_k * \boldsymbol{x}_k^p - \boldsymbol{s}^p\Big\|_2^2 + \lambda \sum_{k=1}^K \|\boldsymbol{x}_k^p\|_1\right)$$
$$\text{s.t.} \quad \{\boldsymbol{d}_k\}_{k=1}^K \in \boldsymbol{\mathcal{D}}, \tag{4}$$

where $\boldsymbol{\mathcal{D}} = \{\boldsymbol{d}_k \mid \|\boldsymbol{d}_k\|_2 \leq 1,\ k=1,\ldots,K\}$. The CDL problem is usually addressed by alternating optimization with respect to $\{\boldsymbol{x}_k^p\}_{k=1}^K$ and $\{\boldsymbol{d}_k\}_{k=1}^K$ [21]–[23]. Several works have shown that (4) can be solved for $\{\boldsymbol{d}_k\}_{k=1}^K$ effectively and efficiently using ADMM in frequency domain [31].

The contributions of this letter are summarized as follows: (i) we present an efficient approach for solving the convolutional

Manuscript received December 3, 2021; accepted December 9, 2021. Date of publication December 14, 2021; date of current version January 28, 2022. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Shiwen He. *(Corresponding author: Sergiy A. Vorobyov.)*

The authors are with the Department of Signal Processing and Acoustics, Aalto University, Espoo 02150, Finland (e-mail: farshad.ghorbaniveshki@aalto.fi; svor@ieee.org).

Digital Object Identifier 10.1109/LSP.2021.3135196

least-squares fitting which leads to a constant improvement on the complexity of the existing CSC algorithms; (ii) we use the proposed solution to improve the efficiency of existing CDL methods; (iii) we propose a novel algorithm for solving the CSC problem with a constraint on the approximation error based on our solution to the unconstrained CSC problem. MATLAB implementations of the proposed algorithms are available at GitHub repository [32].

II. Proposed Algorithms

A. Unconstrained CSC

The ADMM formulation of the unconstrained CSC problem (3) can be written in the form

$$\underset{\{\boldsymbol{x}_k\}_{k=1}^K,\{\boldsymbol{z}_k\}_{k=1}^K}{\text{minimize}} \ \frac{1}{2}\Big\|\sum_{k=1}^K \boldsymbol{d}_k * \boldsymbol{z}_k - \boldsymbol{s}\Big\|_2^2 + \lambda\sum_{k=1}^K \|\boldsymbol{x}_k\|_1$$
$$\text{s.t.} \quad \boldsymbol{z}_k = \boldsymbol{x}_k, \quad k=1,\dots,K.$$

The ADMM iterations are

$$\{\boldsymbol{z}_k^{t+1}\}_{k=1}^K = \underset{\{\boldsymbol{z}_k\}_{k=1}^K}{\text{argmin}} \frac{1}{2}\Big\|\sum_{k=1}^K \boldsymbol{d}_k * \boldsymbol{z}_k - \boldsymbol{s}\Big\|_2^2 + \frac{\rho}{2}\sum_{k=1}^K \|\boldsymbol{z}_k - \boldsymbol{x}_k^t + \boldsymbol{u}_k^t\|_2^2$$

$$\{\boldsymbol{x}_k^{t+1}\}_{k=1}^K = \underset{\{\boldsymbol{x}_k\}_{k=1}^K}{\text{argmin}} \lambda\sum_{k=1}^K \|\boldsymbol{x}_k\|_1 + \frac{\rho}{2}\sum_{k=1}^K \|\boldsymbol{z}_k^{t+1} - \boldsymbol{x}_k + \boldsymbol{u}_k^t\|_2^2$$

$$\boldsymbol{u}_k^{t+1} = \boldsymbol{u}_k^t + \boldsymbol{z}_k^{t+1} - \boldsymbol{x}_k^{t+1}, \qquad k=1,\dots,K,$$

where $\rho > 0$ is the penalty parameter and $\{\boldsymbol{u}_k\}_{k=1}^K$ are the scaled Lagrangian multipliers. The second subproblem ($\boldsymbol{x}$-update step) can be addressed in an element-wise manner using the shrinkage operator

$$\boldsymbol{x}_k^{t+1} = \mathcal{S}_{\lambda/\rho}\left(\boldsymbol{z}_k{}^{t+1} + \boldsymbol{u}_k^t\right), \qquad k=1,\dots,K.$$

For completeness, it is reminded the shrinkage operator is defined as $\mathcal{S}_\kappa(a) = \text{sign}(a)\text{max}(0, |a| - \kappa)$.

The challenging step is solving the first subproblem ($\boldsymbol{z}$-update), which entails solving the optimization problem

$$\underset{\{\boldsymbol{z}_k\}_{k=1}^K}{\text{minimize}} \ \frac{1}{2}\left\|\sum_{k=1}^K \boldsymbol{d}_k * \boldsymbol{z}_k - \boldsymbol{s}\right\|_2^2 + \frac{\rho}{2}\sum_{k=1}^K \|\boldsymbol{z}_k - \boldsymbol{w}_k\|_2^2. \tag{5}$$

Using the convolution theorem, problem (5) in Fourier domain can be written as

$$\underset{\{\hat{\boldsymbol{z}}_k\}_{k=1}^K}{\text{minimize}} \ \frac{1}{2}\left\|\sum_{k=1}^K \hat{\boldsymbol{d}}_k \odot \hat{\boldsymbol{z}}_k - \hat{\boldsymbol{s}}\right\|_2^2 + \frac{\rho}{2}\sum_{k=1}^K \|\hat{\boldsymbol{z}}_k - \hat{\boldsymbol{w}}_k\|_2^2, \tag{6}$$

where $\hat{(\cdot)}$ denotes the discrete Fourier transform of a signal and $\odot$ stands for the element-wise multiplication operator. Note that the filters $\{\boldsymbol{d}_k\}_{k=1}^K$ are zero-padded to the size of $\{\boldsymbol{z}_k\}_{k=1}^K$ before performing the discrete Fourier transform.

Denoting $\boldsymbol{\delta}_i = [\hat{\boldsymbol{d}}_1(i),\dots,\hat{\boldsymbol{d}}_k(i)]^T$, $\boldsymbol{\zeta}_i = [\hat{\boldsymbol{z}}_1(i),\dots,\hat{\boldsymbol{z}}_k(i)]^T$, $\boldsymbol{\omega}_i = [\hat{\boldsymbol{w}}_1(i),\dots,\hat{\boldsymbol{w}}_k(i)]^T$, $i = 1,\dots,n$, where $(\cdot)^T$ is the (non-conjugate) transpose operator, we can see that problem (6) can be addressed as n independent problems

$$\underset{\boldsymbol{\zeta}_i}{\text{minimize}} \frac{1}{2}\left(\boldsymbol{\delta}_i^T\boldsymbol{\zeta}_i - \hat{s}_i\right)^2 + \frac{\rho}{2}\|\boldsymbol{\zeta}_i - \boldsymbol{\omega}_i\|_2^2.$$

Equating the derivative with respect to $\boldsymbol{\zeta}_i$ to zero, we have

$$\begin{aligned} 0 &= \bar{\boldsymbol{\delta}}_i\left(\boldsymbol{\delta}_i^T\boldsymbol{\zeta}_i - \hat{s}_i\right) + \rho\boldsymbol{\zeta}_i - \rho\boldsymbol{\omega}_i \\ &= (\bar{\boldsymbol{\delta}}_i\boldsymbol{\delta}_i^T + \rho\boldsymbol{I})\boldsymbol{\zeta}_i - \hat{s}_i\bar{\boldsymbol{\delta}}_i - \rho\boldsymbol{\omega}_i \\ &= (\bar{\boldsymbol{\delta}}_i\boldsymbol{\delta}_i^T + \rho\boldsymbol{I})\boldsymbol{\zeta}_i - (\hat{s}_i\bar{\boldsymbol{\delta}}_i - \bar{\boldsymbol{\delta}}_i\boldsymbol{\delta}_i^T\boldsymbol{\omega}_i) - (\bar{\boldsymbol{\delta}}_i\boldsymbol{\delta}_i^T + \rho\boldsymbol{I})\boldsymbol{\omega}_i, \end{aligned} \tag{7}$$

which gives

$$\begin{aligned} \boldsymbol{\zeta}_i^\star &= \boldsymbol{\omega}_i + (\hat{\boldsymbol{s}}_i - \boldsymbol{\delta}_i^T\boldsymbol{\omega}_i)(\bar{\boldsymbol{\delta}}_i\boldsymbol{\delta}_i^T + \rho\boldsymbol{I})^{-1}\bar{\boldsymbol{\delta}}_i \\ &= \boldsymbol{\omega}_i + (\hat{\boldsymbol{s}}_i - \boldsymbol{\delta}_i^T\boldsymbol{\omega}_i)(\|\boldsymbol{\delta}_i\|_2^2 + \rho)^{-1}\bar{\boldsymbol{\delta}}_i, \end{aligned} \tag{8}$$

where $(\cdot)^\star$ denotes the solution to an optimization problem and $\bar{(\cdot)}$ is the complex-conjugate of a complex number.

Denoting

$$\hat{\boldsymbol{c}}_k^\rho = \bar{\hat{\boldsymbol{d}}}_k \oslash \left(\rho + \sum_{k=1}^K \bar{\hat{\boldsymbol{d}}}_k \odot \hat{\boldsymbol{d}}_k\right), \quad \hat{\boldsymbol{r}} = \hat{\boldsymbol{s}} - \sum_{k=1}^K \hat{\boldsymbol{d}}_k \odot \hat{\boldsymbol{w}}_k \tag{9}$$

(here $\oslash$ stands for the element-wise division operator), the solution to the $\boldsymbol{z}$-update step based on (8) can be found as

$$\hat{\boldsymbol{z}}_k^\star = \hat{\boldsymbol{w}}_k + \hat{\boldsymbol{c}}_k^\rho \odot \hat{\boldsymbol{r}}. \tag{10}$$

Computational Complexity: The available ADMM-based CSC algorithms usually address the $\boldsymbol{z}$-update step as

$$\boldsymbol{\zeta}_i^\star = (\bar{\boldsymbol{\delta}}_i\boldsymbol{\delta}_i^T + \rho\boldsymbol{I})^{-1}(\hat{s}_i\bar{\boldsymbol{\delta}}_i + \rho\boldsymbol{\omega}_i), \tag{11}$$

which can be inferred from the second line of (7). Computing (11) using matrix inversion results in a complexity of $\mathcal{O}(K^3)$ [21]. However, the work of [23] demonstrated that this can be reduced to $\mathcal{O}(K)$ using the Sherman-Morrison formula. The complexity of the proposed method is also of $\mathcal{O}(K)$. However, using further simplifications, the proposed approach eliminates the need for explicit matrix inversion and requires fewer computations. In particular, performing the $\boldsymbol{z}$-update step on a batch of P images using the proposed method requires $((4K+1)P + 3K + 1)n$ flops, while it takes $(7KP + 3K + 1)n$ flops using the method of [23], indicating a considerable improvement provided by our method.

B. CSC With a Constraint on the Approximation Error

The ADMM reformulation of problem (2) is given as

$$\underset{\{\boldsymbol{x}_k\}_{k=1}^K,\{\boldsymbol{z}_k\}_{k=1}^K}{\text{minimize}} \ \boldsymbol{f}\left(\{\boldsymbol{z}_k\}_{k=1}^K\right) + \sum_{k=1}^K \|\boldsymbol{x}_k\|_1 \ \text{s.t.} \ \boldsymbol{z}_k = \boldsymbol{x}_k, \ \forall k,$$

where $\boldsymbol{f}(\cdot)$ is an indicator function of the constraint set in (3), that is,

$$\boldsymbol{f}\left(\{\boldsymbol{z}_k\}_{k=1}^K\right) = \begin{cases} 0, & \text{if } \ \boldsymbol{e}\left(\{\boldsymbol{z}_k\}_{k=1}^K\right) \le \epsilon \\ \infty, & \text{otherwise} \end{cases},$$

with

$$\boldsymbol{e}\left(\{\boldsymbol{z}_k\}_{k=1}^K\right) = \left\|\sum_{k=1}^K \boldsymbol{d}_k * \boldsymbol{z}_k - \boldsymbol{s}\right\|_2^2. \tag{12}$$

The ADMM iterations are

$$\{\boldsymbol{z}_k^{t+1}\}_{k=1}^K = \underset{\{\boldsymbol{z}_k\}_{k=1}^K}{\text{argmin}} \boldsymbol{f}\left(\{\boldsymbol{z}_k\}_{k=1}^K\right) + \frac{\rho}{2}\sum_{k=1}^K \|\boldsymbol{z}_k - \boldsymbol{x}_k^t + \boldsymbol{u}_k^t\|_2^2$$

$$\{\boldsymbol{x}_k^{t+1}\}_{k=1}^K = \underset{\{\boldsymbol{x}_k\}_{k=1}^K}{\text{argmin}} \sum_{k=1}^K \|\boldsymbol{x}_k\|_1 + \frac{\rho}{2}\sum_{k=1}^K \|\boldsymbol{z}_k^{t+1} - \boldsymbol{x}_k + \boldsymbol{u}_k^t\|_2^2$$

$$\boldsymbol{u}_k^{t+1} = \boldsymbol{u}_k^t + \boldsymbol{z}_k^{t+1} - \boldsymbol{x}_k^{t+1}, \qquad k=1,\dots,K.$$

The $\boldsymbol{z}$-update step requires solving the following optimization problem

$$\underset{\{\boldsymbol{z}_k\}_{k=1}^K}{\text{minimize}}\ \boldsymbol{f}\left(\{\boldsymbol{z}_k\}_{k=1}^K\right) + \frac{\rho}{2}\sum_{k=1}^K \|\boldsymbol{z}_k - \boldsymbol{w}_k\|_2^2. \tag{13}$$

Depending on $\{\boldsymbol{w}_k\}_{k=1}^K$, problem (13) either has a trivial solution or it is equivalent to an equality-constrained optimization problem. This can be expressed as

$$\{\boldsymbol{z}_k^\star\}_{k=1}^K = \begin{cases} \{\boldsymbol{w}_k\}_{k=1}^K, & \text{if } \boldsymbol{e}\left(\{\boldsymbol{w}_k\}_{k=1}^K\right) \leq \epsilon \\ \underset{\{\boldsymbol{z}_k\}_{k=1}^K}{\text{argmin}} \sum_{k=1}^K \|\boldsymbol{z}_k - \boldsymbol{w}_k\|_2^2 \ \text{ s.t. } \boldsymbol{e}\left(\{\boldsymbol{z}_k\}_{k=1}^K\right) = \epsilon, & \text{otherwise} \end{cases}. \tag{14}$$

Using a suitable regularization parameter ν, the problem in the second term of (14) can be reformulated as

$$\underset{\{\boldsymbol{z}_k\}_{k=1}^K}{\text{minimize}}\ \boldsymbol{e}\left(\{\boldsymbol{z}_k\}_{k=1}^K\right) + \nu\sum_{k=1}^K \|\boldsymbol{z}_k - \boldsymbol{w}_k\|_2^2, \tag{15}$$

which has the same form as problem (5). Finding the solution of (15) using (10) and plugging it into (12) leads to

$$\boldsymbol{e}\left(\{\boldsymbol{z}_k^\star\}_{k=1}^K\right) = \frac{\nu^2}{n}\left\|\hat{\boldsymbol{r}} \oslash \left(\nu + \sum_{k=1}^K \bar{\hat{\boldsymbol{d}}}_k \odot \hat{\boldsymbol{d}}_k\right)\right\|_2^2,$$

where the division by n is required by Parseval's theorem. Thus, problem (13) is simplified to a single-variable optimization problem for finding the optimal parameter $\nu^\star$, which satisfies $\nu^\star = \left\{\nu \mid \boldsymbol{e}\left(\{\boldsymbol{z}_k^\star\}_{k=1}^K\right) = \epsilon\right\}$. Considering that $\boldsymbol{e}(\{\boldsymbol{z}_k^\star\}_{k=1}^K)$ is strictly monotonically increasing in $\nu > 0$, this problem can be efficiently addressed, for example, using the *secant* method. Once $\nu^\star$ is known, the $\boldsymbol{z}$-update can be performed as $\hat{\boldsymbol{z}}_k^\star = \hat{\boldsymbol{w}}_k + \hat{\boldsymbol{c}}_k^{\nu^\star} \odot \hat{\boldsymbol{r}}, k = 1, \ldots, K$, where $\hat{\boldsymbol{c}}_k^{\nu^\star}$ and $\hat{\boldsymbol{r}}$ are calculated using (9).

C. Dictionary Update

Addressing CDL problem (4) over $\{\boldsymbol{d}_k\}_{k=1}^K$ is equivalent to solving the optimization problem

$$\underset{\{\boldsymbol{d}_k\}_{k=1}^K}{\text{minimize}}\ \frac{1}{2}\sum_{p=1}^P \left\|\sum_{k=1}^K \boldsymbol{d}_k * \boldsymbol{x}_k^p - \boldsymbol{s}^p\right\|_2^2 + \boldsymbol{\Omega}\left(\{\boldsymbol{d}_k\}_{k=1}^K\right), \tag{16}$$

where $\boldsymbol{\Omega}(\boldsymbol{d}_k)$ is an indicator function associated with the constraint set in (4). Problem (16) can be efficiently addressed using the consensus ADMM method [31]. The consensus ADMM formulation of (16) is given as

$$\underset{\{\boldsymbol{d}_k\}_{k=1}^K}{\text{minimize}}\ \frac{1}{2}\sum_{p=1}^P \left\|\sum_{k=1}^K \boldsymbol{g}_k^p * \boldsymbol{x}_k^p - \boldsymbol{s}^p\right\|_2^2 + \boldsymbol{\Omega}\left(\{\boldsymbol{d}_k\}_{k=1}^K\right)$$
$$\text{s.t.}\ \ \boldsymbol{g}_k^p = \boldsymbol{d}_k, \quad k = 1, \ldots, K, \quad p = 1, \ldots, P$$

with the ADMM iterations

$$\{\boldsymbol{g}_k^{p,t+1}\}_{k=1}^K = \underset{\{\boldsymbol{g}_k^p\}_{k=1}^K}{\text{argmin}}\left(\frac{1}{2}\left\|\sum_{k=1}^K \boldsymbol{g}_k^p * \boldsymbol{x}_k^p - \boldsymbol{s}^p\right\|_2^2 + \frac{\sigma}{2}\sum_{k=1}^K \|\boldsymbol{g}_k^p - \boldsymbol{d}_k^t + \boldsymbol{v}_k^{p,t}\|_2^2\right)$$

$$\{\boldsymbol{d}_k^{t+1}\}_{k=1}^K = \underset{\{\boldsymbol{d}_k\}_{k=1}^K}{\text{argmin}}\left(\boldsymbol{\Omega}\left(\{\boldsymbol{d}_k\}_{k=1}^K\right) + \frac{\sigma}{2}\sum_{k=1}^K \|\boldsymbol{d}_k - \frac{1}{P}\sum_{p=1}^P (\boldsymbol{g}_k^{p,t+1} + \boldsymbol{v}_k^{p,t})\|_2^2\right)$$

$$\boldsymbol{v}_k^{p,t+1} = \boldsymbol{v}_k^{p,t} + \boldsymbol{g}_k^{p,t+1} - \boldsymbol{d}_k^{t+1},\ k = 1, \ldots, K,\ p = 1, \ldots, P.$$

The first subproblem ($\boldsymbol{g}$-update) is similar to problem (5). Thus, it can be efficiently addressed using the proposed approach in Section II-A. The use of the Fourier domain-based approach requires $\{\boldsymbol{g}_k^p\}_{k=1}^K$ to be the same size as $\{\boldsymbol{x}_k^p\}_{k=1}^K$. Hence, the filters $\{\boldsymbol{d}_k\}_{k=1}^K$ are zero-padded to the size of $\{\boldsymbol{x}_k^p\}_{k=1}^K$ to be conformable with $\{\boldsymbol{g}_k^p\}_{k=1}^K$. Subproblem $\boldsymbol{d}$-update can be solved by projecting $\frac{1}{P}\sum_{p=1}^P(\boldsymbol{g}_k^{p,t+1} + \boldsymbol{v}_k^{p,t})$ onto the constraint set. This can be done simply by mapping the entries outside the constraint support to zero before normalizing the ℓ_2-norm. This approach can be also used for learning multiscale dictionaries, *i.e.*, filters with different sizes.

D. CDL Algorithm

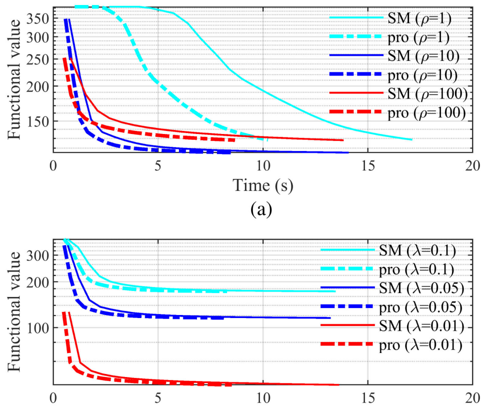
CDL problem given by (4) is addressed using alternation approach (by alternating between CSC (see Section II-A) and dictionary update (see Section II-C) subproblems) to find a local optimum. We use a single iteration for each subproblem (the updated ADMM variables are used to initiate the succeeding iterations). This approach has been shown to be effective while simplifying the algorithm [23], [31]. We also use the variable coupling approach suggested in [33] which is shown to provide a better numerical stability [23], [31]. Specifically, the sparse (shrunk) variable $\{\boldsymbol{x}_k^p\}_{k=1}^K$ and the constrained filters $\{\boldsymbol{d}_k\}_{k=1}^K$ are passed to the next subproblem. The dictionary can be initialized using norm-normalized Gaussian random filters. All other ADMM variables are initialized using zero arrays of appropriate sizes.

The performances of the proposed algorithms can be substantially improved using ADMM extensions such as *over-relaxation* [30, Section 3.4.3] and *varying penalty parameter* [30, Section 3.4.1]. The work of [23] provides detailed explanations of how these extensions can be incorporated into ADMM-based CSC and CDL algorithms.

III. Experimental Results

In this section, we first compare the proposed unconstrained CSC algorithm with the state-of-the-art method, which uses the Sherman-Morrison formula in convolutional fitting step (the SM method) [23]. Then, we compare our unconstrained and constrained CSC methods in terms of convergence speed. Finally, we compare the proposed CDL algorithm with three available methods. All methods are based on the same alternating approach explained in Section II-D and use ADMM in both phases (CSC and dictionary update). All compared methods use the SM method in CSC phase. The compared dictionary learning methods are based on the conjugate gradient method (CG) [23], the iterative Sherman-Morrison method (ISM) [23] and a method based on the consensus ADMM framework and the Sherman-Morrison formula (SM-cns) [31].

A 512×512 greyscale Lena image is used in the CSC experiments. The CDL experiments are performed using a dataset of 20 images taken from the USC-SIPI database [34]. All images in the dataset are converted to greyscale and resized to 256×256



(a)

(b)

Fig. 1. Functional values over time for the proposed unconstrained CSC method (pro) and the SM method using (a) different values of ρ for $\lambda = 0.05$, and (b) different values of λ for $\rho = 10$. A dictionary of 16 filters of size 8×8 is used in both cases.

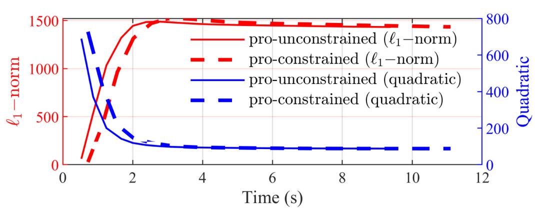


Fig. 2. The quadratic and ℓ_1-norm functional values for the proposed unconstrained and constrained CSC methods using $\lambda = 0.05$ ($\epsilon = 88.1886$), $\rho = 10$. A dictionary of 16 filters of size 8×8 is used.

pixels. All methods are implemented using MATLAB. All experiments are conducted on a PC equipped with an Intel(R) Core(TM) i5-8365 U 1.60 GHz CPU.

A. CSC Results

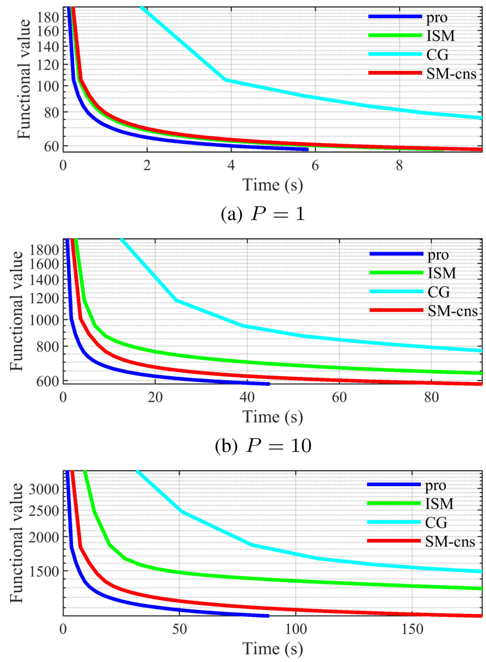
Fig. 1 shows the functional values over time for 25 iterations of the proposed unconstrained CSC method and the SM method using different values of ρ and λ tested. We use a fixed number of iterations to display the deference in efficiencies (the iterations of the two methods are equally effective). As it can be seen, the proposed method is significantly more efficient for all λ and ρ values. The algorithm complexities have been compared in Section II-A.

The proposed constrained and unconstrained CSC methods are compared in Fig. 2. Specifically, we executed the unconstrained CSC method using $\lambda = 0.05$, then we used the observed quadratic functional value ($\epsilon = 88.1886$) to run our constrained CSC method, while keeping the rest of the parameters unchanged. As it can be seen, the quadratic and the ℓ_1-norm functionals converge to the same values in both CSC methods. The constrained method results in a longer runtime, which accounts for optimization with respect to ν in each iteration.

B. CDL Results

In Fig. 3, the functional values over time for 50 iterations of all CDL methods using different dataset sizes P are compared. The complexity of the ISM method is of $\mathcal{O}(KP^2)$, which makes it inefficient when P is large. CG improves scalability, but slows down the convergence. The complexities of the proposed method and SM-cns are both of $\mathcal{O}(KP)$, and their iterations are equally effective. However, as it can be seen, the proposed method is substantially faster. This is achieved by using the method explained in Section II-A instead of the Sherman-Morrison formula, in both the $\boldsymbol{z}$-update step (CSC phase) and the $\boldsymbol{g}$-update step (dictionary update phase).

In Fig. 4, the convergence speeds of the proposed CDL method and SM-cns using different dictionary sizes K are compared. The improved computational efficiency of the proposed method can be clearly observed.



(a) $P = 1$

(b) $P = 10$

(c) $P = 20$

Fig. 3. Functional values over time using different values of P, $\rho = 10$, $\lambda = 0.05$, $K = 16$ filters of size 8×8.

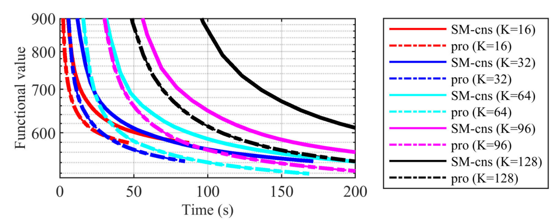


Fig. 4. Functional values over time using different K values, $P = 10$, $\rho = 10$, $\lambda = 0.05$ and filters of size 8×8.

IV. Conclusion

An efficient solution for the convolutional least-squares fitting problem has been presented. The proposed method has been used to substantially improve the efficiency of the state-of-the-art convolutional sparse coding and dictionary learning algorithms. In addition, a novel method for convolutional sparse approximation with a constraint on the approximation error has been proposed.

REFERENCES

[1] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–227, Feb. 2009.

[2] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Trans. Inf. Theory*, vol. 53, no. 12, pp. 4655–4666, Dec. 2007.

[3] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Trans. Image Process.*, vol. 15, no. 12, pp. 3736–3745, Dec. 2006.

[4] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *IEEE Trans. Image Process.*, vol. 19, no. 11, pp. 2861–2873, Nov. 2010.

[5] J. M. Bioucas-Dias *et al.*, "Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 5, no. 2, pp. 354–379, Apr. 2012.

[6] G. Wunder, H. Boche, T. Strohmer, and P. Jung, "Sparse signal processing concepts for efficient 5G system design," *IEEE Access*, vol. 3, pp. 195–208, 2015.

[7] K. Engan, S. O. Aase, and J. H. Husøy, "Method of optimal directions for frame design," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Phoenix, AZ, USA, 1999, pp. 2443–2446.

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

[9] A. Cogliati, Z. Duan, and B. Wohlberg, "Piano transcription with convolutional sparse lateral inhibition," *IEEE Signal Process. Lett.*, vol. 24, no. 4, pp. 392–396, Apr. 2017.

[10] P. Jao, L. Su, Y. Yang, and B. Wohlberg, "Monaural music source separation using convolutional sparse coding," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 11, pp. 2158–2170, Nov. 2016.

[11] P. Bao *et al.*, "Convolutional sparse coding for compressed sensing CT reconstruction," *IEEE Trans. Med. Imag.*, vol. 38, no. 11, pp. 2607–2619, Nov. 2019.

[12] Y. Liu, X. Chen, R. Ward, and Z. Wang, "Image fusion with convolutional sparse representation," *IEEE Signal Process. Lett.*, vol. 23, no. 12, pp. 1882–1886, Dec. 2016.

[13] X. Hu, F. Heide, Q. Dai, and G. Wetzstein, "Convolutional sparse coding for RGB NIR imaging," *IEEE Trans. Image Process.*, vol. 27, no. 4, pp. 1611–1625, Apr. 2018.

[14] S. Gu, W. Zuo, Q. Xie, D. Meng, X. Feng, and L. Zhang, "Convolutional sparse coding for image super-resolution," in *Proc. IEEE Int. Conf. Comput. Vision*, Santiago, Chile, 2015, pp. 1823–1831.

[15] M. Li *et al.*, "Video rain streak removal by multiscale convolutional sparse coding," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit.*, Salt Lake City, UT, USA, 2018, pp. 6644–6653.

[16] B. Wohlberg and P. Wozniak, "PSF estimation in crowded astronomical imagery as a convolutional dictionary learning problem," *IEEE Signal Process. Lett.*, vol. 28, pp. 374–378, 2021.

[17] M. Li, X. Cao, Q. Zhao, L. Zhang, and D. Meng, "Online rain/snow removal from surveillance videos," *IEEE Trans. Image Process.*, vol. 30, pp. 2029–2044, 2021.

[18] X. Feng, C. Fang, X. Lou, and K. Hu, "Research on infrared and visible image fusion based on tetrolet transform and convolution sparse representation," *IEEE Access*, vol. 9, pp. 23498–23510, 2021.

[19] J. Wei, L. Mi, Y. Hu, J. Ling, Y. Li, and Z. Chen, "Effects of lossy compression on remote sensing image classification based on convolutional sparse coding," *IEEE Geosci. Remote Sens. Lett.*, vol. 9, pp. 1–5, 2021.

[20] Y. Zhu, Y. Lu, Q. Gao, and D. Sun, "Infrared and visible image fusion based on convolutional sparse representation and guided filtering," *J. Electron. Imag.*, vol. 30, no. 4, pp. 1–17, 2021.

[21] H. Bristow, A. Eriksson, and S. Lucey, "Fast convolutional sparse coding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Portland, OR, USA, 2013, pp. 391–398.

[22] F. Heide, W. Heidrich, and G. Wetzstein, "Fast and flexible convolutional sparse coding," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit.*, Boston, MA, USA, 2015, pp. 5135–5143.

[23] B. Wohlberg, "Efficient algorithms for convolutional sparse representations," *IEEE Trans. Image Process.*, vol. 25, no. 1, pp. 301–315, Jan. 2016.

[24] G. Peng, "Adaptive ADMM for dictionary learning in convolutional sparse representation," *IEEE Trans. Image Process.*, vol. 28, no. 7, pp. 3408–3422, Jul. 2019.

[25] B. Choudhury, R. Swanson, F. Heide, G. Wetzstein, and W. Heidrich, "Consensus convolutional sparse coding," in *Proc. IEEE Int. Conf. Comput. Vision*, Venice, Italy, 2017, pp. 4290–4298.

[26] V. Papyan, Y. Romano, M. Elad, and J. Sulam, "Convolutional dictionary learning via local processing," in *Proc. IEEE Int. Conf. Comput. Vision*, Venice, Italy, 2017, pp. 5306–5314.

[27] I. Rey-Otero, J. Sulam, and M. Elad, "Variations on the convolutional sparse coding model," *IEEE Trans. Signal Process.*, vol. 68, pp. 519–528, 2020.

[28] V. Papyan, J. Sulam, and M. Elad, "Working locally thinking globally: Theoretical guarantees for convolutional sparse coding," *IEEE Trans. Signal Process.*, vol. 65, no. 21, pp. 5687–5701, Nov. 2017.

[29] Y. Wang, Q. Yao, J. T. Kwok, and L. M. Ni, "Scalable online convolutional sparse coding," *IEEE Trans. Image Process.*, vol. 27, no. 10, pp. 4850–4859, Oct. 2018.

[30] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2011.

[31] C. Garcia-Cardona and B. Wohlberg, "Convolutional dictionary learning: A comparative review and new algorithms," *IEEE Trans. Comput. Imag.*, vol. 4, no. 3, pp. 366–381, Sep. 2018.

[32] [Online]. Available: https://github.com/FarshadGVeshki/Convolutional-Sparse-Coding.git

[33] C. Garcia-Cardona and B. Wohlberg, "Subproblem coupling in convolutional dictionary learning," in *Proc. IEEE Int. Conf. Image Process.*, Beijing, China, 2017, pp. 1697–1701.

[34] [Online]. Available: http://sipi.usc.edu/database/