Efficient Convolutional Dictionary Learning using Fast Iterative Shrinkage Thresholding Algorithm

Gustavo Silva, Paul Rodríguez
{gustavo.silva, prodrig}@pucp.edu.pe
Electrical Engineering Department, Pontificia Universidad Católica del Perú

Abstract

We present a **novel algorithm for solving the convolutional dictionary learning problem**.

In this work, we include **two improvements over existing methods**:

i An accelerated Proximal Gradient (APG) approach calculated in the frequency domain in order to efficiently solve the dictionary update stage.

ii A partial computation of the coefficient maps through a new update model reminiscent of the Block Gauss Seidel (BGS) method.

Experimental results: The proposed method is significantly faster than the state-of-the-art methods. Speedup of $1.5 \sim 12.5$.

Denoising task: The performance is comparable to the existing methods in terms of PSNR, SSIM and sparsity metrics.

Introduction

Convolutional dictionary learning (CDL)

$$\underset{\{x_{k,m}\},\{d_m\}}{\arg\min} \frac{1}{2}\sum_k \left\| \sum_m d_m * x_{k,m} - s_k \right\|_2^2 + \lambda \sum_k \sum_m \|x_{k,m}\|_1 \quad \text{s.t.} \quad \|d_m\|_2 = 1 \quad (1)$$

where $\{x_{k,m}\}$ represents the K sets of M coefficient maps, $\{d_m\}$ a set of M dictionary filters, and $\{s_k\}$ the K training images.

Used on the fields of:	Drawbacks:
• Signal/image processing • Computer vision	• High memory requirements • High computational complexity

- **Coefficient update (Sparse coding or SC):**

$$\underset{\{x_m\}}{\arg\min} \frac{1}{2}\| \sum_m d_m * x_m - s\|_2^2 + \lambda \sum_m \|x_m\|_1 \,. \quad (2)$$

ADMM: [1] proposed an ADMM-based solution, in which the most expensive step is handled in the frequency domain via the Sherman-Morrison method.

FISTA: Recent work [2] proposed to compute the gradient step in the frequency domain, thus reducing the computational cost associated with the convolution operators.

- **Dictionary update (Dictionary learning or DL):**

$$\underset{\{d_m\}}{\arg\min} \frac{1}{2}\sum_k \left\| \sum_m x_{k,m} * d_m - s_k \right\|_2^2 + \sum_m i_{C_{PN}}(d_m) \,. \quad (3)$$

ADMM [2]: Extension of [1] for DL problem, in which most expensive step can be solved directly using conjugate gradient **(CG)** or arranged and then solved with the Iterative Sherman-Morrison **(ISM)** method.

$$\underset{\{d_m\},\{g_m\}}{\arg\min} \frac{1}{2}\sum_k \left\| \sum_m x_{k,m} * d_m - s_k \right\|_2^2 + \sum_m i_{C_{PN}}(g_m) \quad \text{s.t.} \quad d_m - g_m = 0 \quad (4)$$

ADMM-Consensus [3],[4]: The consensus approach **(CSS)** allows to obtain K independent systems that can be solved with the simple Sherman-Morrison technique [1].

$$\underset{\{d_{k,m}\},\{g_m\}}{\arg\min} \frac{1}{2}\sum_k \left\| \sum_m x_{k,m} * d_{k,m} - s_k \right\|_2^2 + \sum_m i_{C_{PN}}(g_m) \quad \text{s.t.} \quad d_{1,m} = d_{2,m} = \cdots = g_m \quad (5)$$

References

[1] B. Wohlberg, "Efficient convolutional sparse coding," *Acoustics, Speech and Signal Processing (ICASSP), IEEE International Conference*, 2014.

[2] B. Wohlberg, "Efficient algorithms for convolutional sparse representations," *IEEE Transactions on Image Processing*, vol. 25, pp. 301–315, Jan. 2016.

[3] M. Ŝorel and F. Ŝroubek, "Fast convolutional sparse coding using matrix inversion lemma," *Digital Signal Processing*, vol. 55, pp. 44 – 51, 2016.

[4] C. Garcia and B. Wohlberg, "Subproblem coupling in convolutional dictionary learning," *International Conference on Image Processing (ICIP)*, 2017.

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

Proposed Method and Results

Frequency Domain Accelerated Proximal Gradient (APG)

Our algorithm consists of an APG-based solution for each update of the CDL problem, in which most steps are computed in the frequency domain.

The **dictionary update stage** is given by the following steps:

- Defining the linear operator $X_{k,m}$, such that $x_{k,m} * d_m = X_{k,m} d_m$ and denoting $\hat{X}_{k,m}$, d_m and s_k in the DFT domain as $\hat{X}_{k,m}$, $\hat{d}_m$ and $\hat{s}_k$, respectively. The fidelity term is arranged as

$$\sum_k \left\| \sum_m \hat{X}_{k,m}\hat{d}_m - \hat{s}_k \right\|_2^2 = \sum_k \left\| \hat{X}_k d_f - \hat{s}_k \right\|_2^2 = \|X_f d_f - s_f\|_2^2 \quad (6)$$

where $\hat{X}_k = (\hat{X}_{k,1}\ \hat{X}_{k,2}\ \cdots)$, $\quad X_f = (\hat{X}_1\ \hat{X}_2\ \cdots)^T$,

$d_f = (\hat{d}_1\ \hat{d}_2\ \cdots)^T$ and $s_f = (\hat{s}_1\ \hat{s}_2\ \cdots)^T$.

Resulting gradient: $\nabla F(d_f) = (X_f)^T (X_f D_f - s_f)$

- In constrast with the spatial domain formulation, where an exact line search is usually computationally prohibitive, the frequency domain formulation does allow an effective exact line search procedure via

$$\underset{\{\rho\}}{\arg\min} \frac{1}{2}\|X_f(d_f - \rho\nabla F(d_f)) - s_f\|_2^2 \,, \quad (7)$$

Algorithm 1: Frequency domain APG

for $k = 1 : \text{maxIter}$ **do**

- **Coefficient update (I):**
  1: Compute X_f^{k+1} via the frequency domain APG approach.
- **Dictionary update (II) :**
  2: Gradient calculation
  $\nabla \mathbf{F}(\mathbf{g_f^k}) = (\mathbf{X_f^{k+1}})^{\mathbf{H}}(\mathbf{X_f^{k+1}}\mathbf{g_f^k} - \mathbf{s_f})$
  3: Step size calculation
  $\rho = \|\nabla\mathbf{F}(\mathbf{g_f^k})\|_2^2 / \|\mathbf{X_f^{k+1}}\nabla\mathbf{F}(\mathbf{g_f^k})\|_2^2$
  4: Dictionary computation
  $h^{k+1} = IFFT2\{g_f^k - \rho \cdot \nabla F(g_f^k)\}$
  $d^{k+1} = \text{prox}_{i_{C_{PN}}}(h^{k+1})$
  $d_f^{k+1} = FFT2\{d^{k+1}\}$
  5: Auxiliary dictionary g_f^{k+1} (Nesterov accelerated method)
  $\gamma^{k+1} = (1 + \sqrt{1 + 4(\gamma^{k+1})^2})$
  $g_f^{k+1} = d_f^{k+1} + \frac{\gamma^k - 1}{\gamma^{k+1}}(d_f^{k+1} - d_f^k)$
  6: Normalization of auxiliary dictionary
  $\mathbf{g_f^{k+1}} = \sqrt{\mathbf{N}} \cdot \mathbf{g_f^{k+1}} / \|\mathbf{g_f^{k+1}}\|_2$

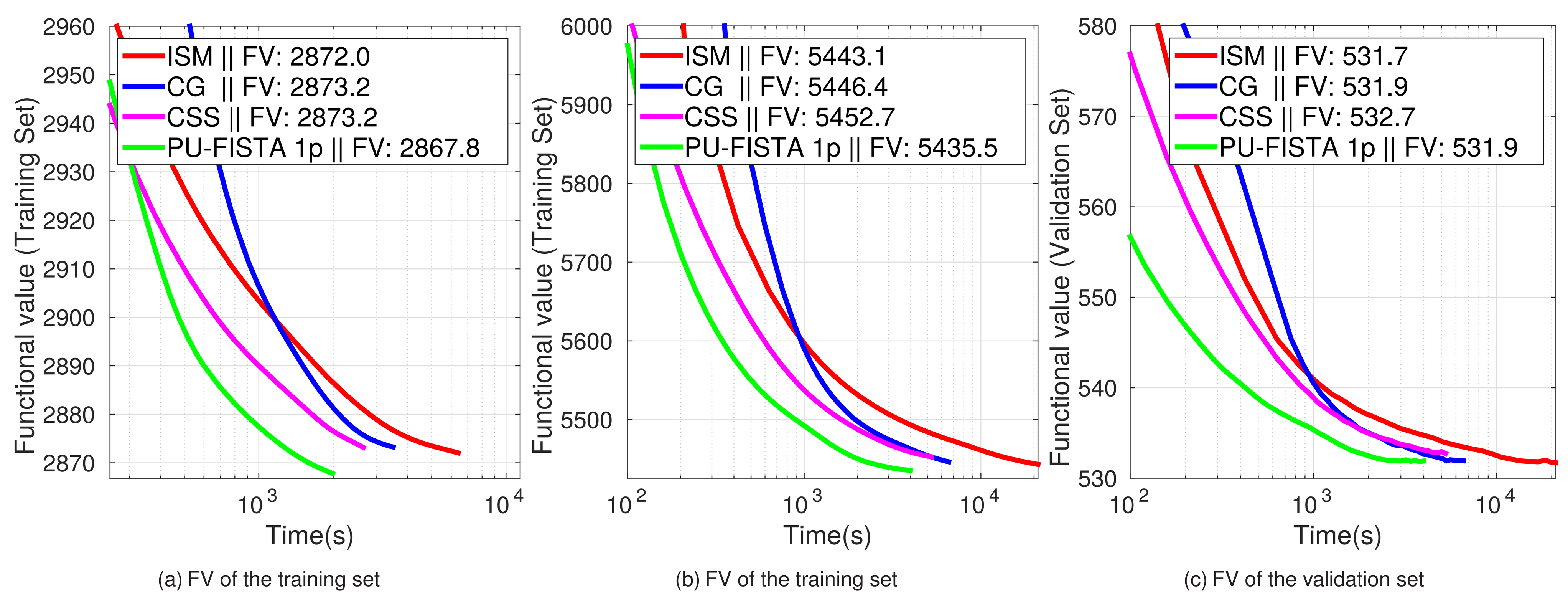

Figure 1 : Value of **the training and validation functional** vs. execution time for the CDL methods on a set of (a) 20 and (b),(c) 40 training images.

Partial Update model

The BGS [5] (a.k.a Alternating Optimization) formulation to minimize a function $f(x)$ is posed as:

$$x_i^{k+1} = \underset{y \in x_i}{\arg\min} f(x_1^k,\ \cdots,\ x_{i-1}^k,\ y,\ x_{i+1}^k,\ \cdots,\ x_r^k) \,, \quad (8)$$

where y is the single partition of interest.

Adapting this model to the CDL problem, the SC update can be written as

$$x_{k,m}^{(i,r)} = \underset{\{x_{k,m}\}}{\arg\min} \frac{1}{2}\sum_{k=1}^{P_r} \left\| \sum_m d_m * x_{k,m} - s_k^{(r)} \right\|_2^2 + \lambda \sum_{k=1}^{P_r}\sum_m \|x_{k,m}\|_1, \quad (9)$$

where the dataset $\{s_k\}$ was divided into R partitions, with denoting P_r is the partition size.

$$s_k = \{s_k^{(1)},\ s_k^{(2)}, \ldots, s_k^{(R)}\}$$

The complete set of coefficient maps $x_{k,m}$ is composed from the current estimated partition and the previous values of the other ones.

$$x_{k,m}^{(i)} = [x_{k,m}^{(i-1,1)},\ \ldots,\ \mathbf{x_{k,m}^{(i,r)}},\ \ldots,\ x_{k,m}^{(i-1,R)}] \quad (10)$$

This complete set of coefficients is used to estimate the current dictionary given by (4).

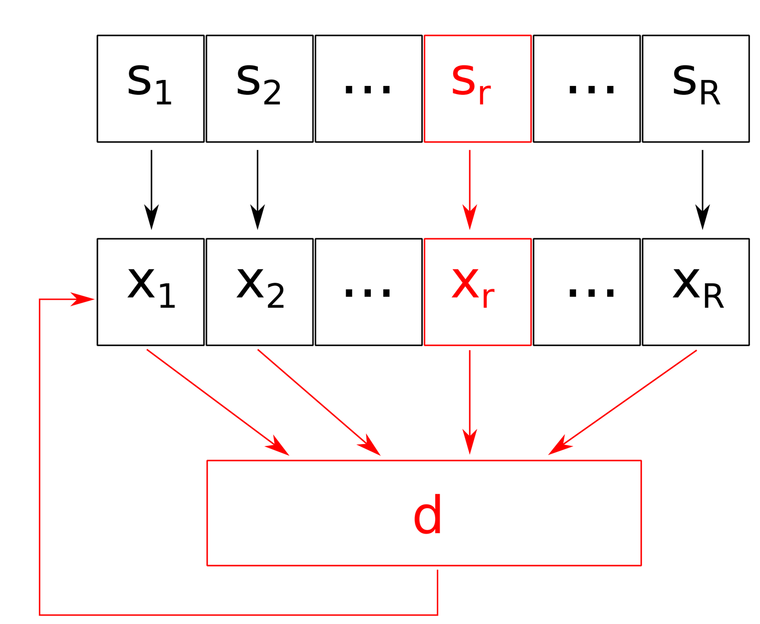

Figure 2 : Partial Update model of the CDL problem.

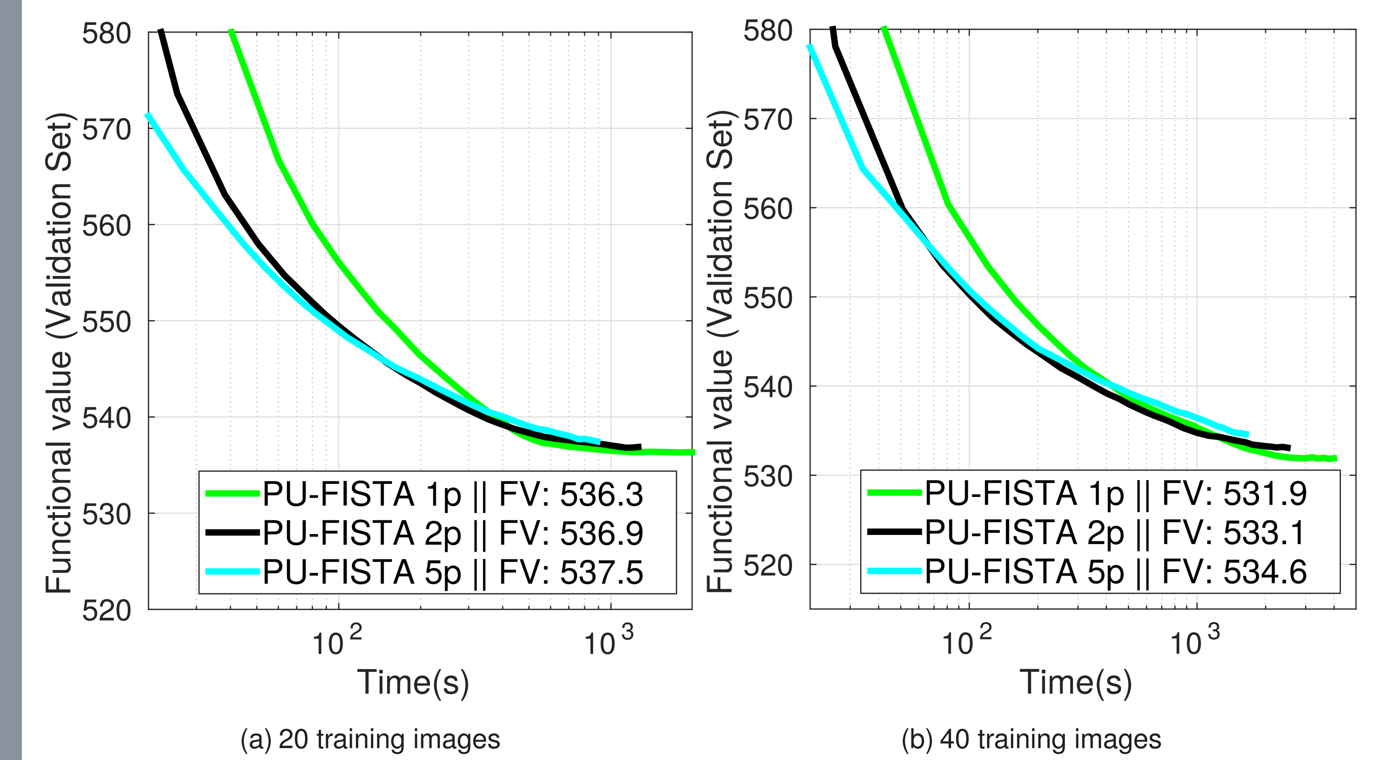

Figure 3 : Value of **validation functional** vs. execution time of our proposed method with 1, 2 and 5 partitions for training set sizes of (a) 20 and (b) 40.

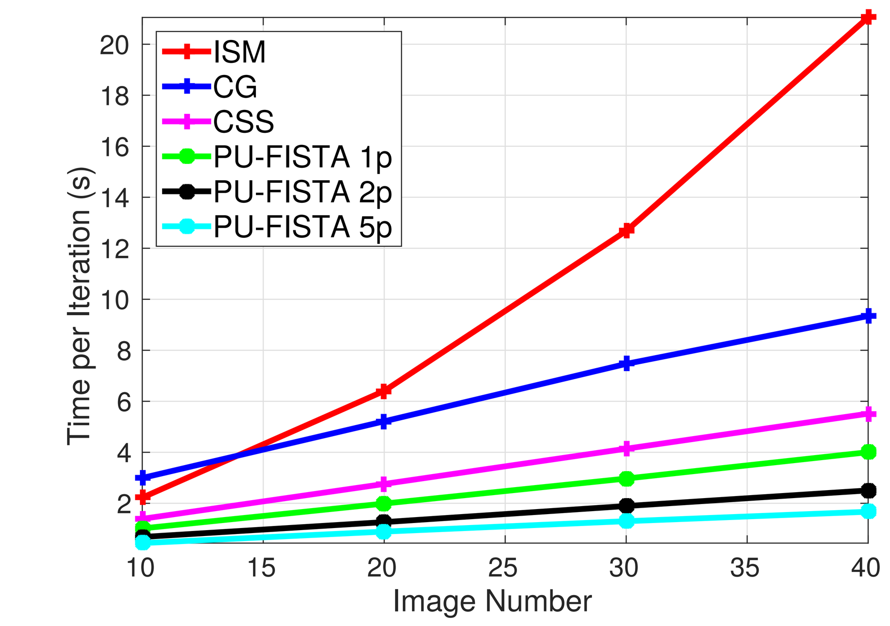

Figure 4 : Average runtime per iteration for different training set sizes.

Training set	ISM	GS	CSS	PU-FISTA 1P	PU-FISTA 2P	PU-FISTA 5P
10	2222	2993	1475	1015	678	439
40	21051	9343	5866	4003	2504	1668

Table 1 : Execution time in seconds of the CDL methods

Speedup:

- Our **AGP algorithm (PU-FISTA-1P)** is about **2.2** $\sim$ **5.3** times faster than **ISM**, **2.5** times faster than **CG**, and **1.5** times faster than **CSS**.
- The complementary update model provides additional speedup of $1.6 \sim 2.5$ times when using $2 \sim 5$ partitions.

Results: Denoising task

	Mandrill			Barbara			Peppers		
	PSNR	SSIM	L0 %	PSNR	SSIM	L0 %	PSNR	SSIM	L0 %
ISM	21.08	0.5286	7.46	23.15	0.6091	5.68	25.36	0.6818	1.58
GC	21.08	0.5282	7.48	23.15	0.6091	5.69	25.35	0.6816	1.60
CSS	21.09	0.5293	7.63	23.14	0.6082	5.73	25.34	0.6805	1.58
PU-FISTA 1p	21.08	0.5293	7.43	23.15	0.6093	5.61	25.36	0.6829	1.55
PU-FISTA 2p	21.08	0.5293	7.38	23.11	0.6084	5.56	25.37	0.6834	1.53
PU-FISTA 5p	21.08	0.5280	7.38	23.08	0.6076	5.57	25.34	0.6825	1.57

Table 2 : Denoising (best λ) of standard images corrupted with AWGN $\sigma = 0.2$

Conclusions

New computationally efficient algorithm for solving the CDL problem considering two complementary formulations. Its speedup is around $1.5 \sim 12.5$. The reconstruction performance in the denoising task is equivalent as the existing methods.

Key contributions:

- APG-based solution for both CDL subproblems that has proved to be significantly faster than state-of-the-art methods.
- Novel update model, which reduces the computations in our sparse coding update.