Regularized K-SVD

Bogdan Dumitrescu, Paul Irofti

Abstract

The problem of dictionary learning (DL) for sparse representations can be approximately solved by several algorithms. Regularization of the optimization objective (representation error) was proved useful since it avoids possible bottlenecks due to nearly linearly dependent atoms. We show here how the well known K-SVD algorithm can be adapted to the regularized DL problem, despite previous claims that such an adaptation seems impossible. We also provide numerical evidence that regularized K-SVD is better than Simultaneous Codeword Optimization (SimCO), the most prominent algorithm dedicated to the regularized DL problem.

Index Terms

sparse representations, dictionary learning, regularization

I. Introduction

Dictionary learning (DL) for sparse representations [1], [2], [3] has many applications in audio and image processing, compression, classification and computer vision, where it gives better performance than it can be obtained with fixed dictionaries like those built from popular transforms.

The DL problem can be stated as follows. The input consists of a data matrix $\boldsymbol{Y} \in \mathbb{R}^{m \times N}$, whose columns are the signals whose representation is sought with sparsity level $s \in \mathbb{N}$. The main output is the dictionary $\boldsymbol{D} \in \mathbb{R}^{m \times n}$, whose columns are named atoms. Each signal is represented as a linear

The authors are with Department of Automatic Control and Computers, University Politehnica of Bucharest, 313 Spl. Independenţei, 060042 Bucharest, Romania. E-mails: bogdan.dumitrescu@acse.pub.ro, paul@irofti.net.

combination of s atoms and the objective is to minimize the representation error:

$$\begin{aligned} \underset{\boldsymbol{D},\boldsymbol{X}}{\text{minimize}} \quad & \|\boldsymbol{Y}-\boldsymbol{D}\boldsymbol{X}\|_F^2 \\ \text{subject to} \quad & \|\boldsymbol{x}_\ell\|_0 \le s,\ \ell = 1:N \\ & \|\boldsymbol{d}_j\|_2 = 1,\ j = 1:n. \end{aligned} \tag{1}$$

We denote by $\boldsymbol{d}_j$ the j-th atom of the dictionary $\boldsymbol{D}$ and by $\boldsymbol{x}_\ell$ the ℓ-th column of the representation matrix $\boldsymbol{X} \in \mathbb{R}^{n\times N}$; $\|\cdot\|_0$ is the number of nonzero elements of a vector and $\|\cdot\|_F$ is the Frobenius norm of a matrix. The first constraint imposes the sparsity level: each column of $\boldsymbol{X}$ must have at most s nonzero elements. The second constraint imposes unit norm on the atoms, such that the multiplicative indetermination between $\boldsymbol{D}$ and $\boldsymbol{X}$ is removed. The DL optimization problem (1) is hard due to the discrete character of the constraints and the non-convexity of the objective.

Typical DL algorithms like MOD [4] and K-SVD [5] alternate two main optimization steps in the attempt of solving (1). The *sparse coding* step finds the sparse representations $\boldsymbol{X}$ for the current dictionary, usually with Orthogonal Matching Pursuit (OMP) [6]. The *dictionary update* step optimizes the dictionary $\boldsymbol{D}$ and, in the case of K-SVD, also the representations $\boldsymbol{X}$, keeping the support fixed (the nonzero structure of $\boldsymbol{X}$); this is the step that makes the main difference between algorithms.

Regularized DL [7] optimizes the objective function

$$f_\mu(\boldsymbol{D},\boldsymbol{X}) = \|\boldsymbol{Y}-\boldsymbol{D}\boldsymbol{X}\|_F^2 + \mu\|\boldsymbol{X}\|_F^2, \tag{2}$$

where $\mu > 0$ is a given parameter. This objective combines representation error with a term meant to avoid large values of the sparse representation coefficients, which is a symptom of nearly linearly dependent atoms that contribute to the same signal representation. In such a situation, the progress in the dictionary update step becomes very slow, due to the almost flat error function.

The Simultaneous Codeword Optimization (SimCO) algorithm proposed in [7] updates the whole dictionary using gradient (or second order) descent with optimal line search and always works with the optimal representations for the given support. While this strategy ensures good results, SimCO is computationally expensive. Optimal line search requires many calculations of the objective function. Also, several descent steps (called inner iterations) are made in each dictionary update step, which imply many calculations of the gradient.

It was stated in [7] that "it is not clear how to extend K-SVD to the regularized case"; a sequential version of SimCO was used there as a substitute for regularized K-SVD. We show in Section II that a regularized version of K-SVD does exist and actually has a very simple form, quite near from that

of the standard algorithm. A similar extension is provided in Section III for Approximated K-SVD [8]. In Section IV, we present numerical tests showing that the new regularized algorithms are faster than SimCO and also give better results.

Another approach to regularized DL is via elastic net regularization [9], [10], where sparsity is enforced by adding $\gamma\|\boldsymbol{X}\|_1$ to (2), where $\gamma > 0$ is a weight; in the sparse coding stage, OMP is replaced by algorithms specialized to the new objective, that are typically slower. Elastic net regularization is adapted for classification purposes in [11]. In a more general framework, regularized DL is solved in [12] with an alternating proximal algorithm. The regularized DL algorithm from [13], based on SimCO, proposes replacing OMP with regularized OMP (ROMP) [14] in sparse coding; note, however, that the regularization used in ROMP has no connection with the (traditional) least-squares regularization from (2). Finally, in other works like [15], [16] regularization is achieved by replacing the ℓ_2 term from (2) with a function depending on the mutual coherence of the atoms. We have compared simple algorithms for quadratic and coherence regularized DL in [17], showing their benefits when s is not small. We note that none of the above methods optimizes simultaneously both the atoms and the representations in the dictionary update stage, in K-SVD style, like the algorithm proposed here.

II. Regularized K-SVD algorithm

The central operation in K-SVD is to find the optimal value of an atom and of the corresponding representations when everything else is fixed. The atoms are updated one by one and dictionary update typically consists of a single sweep of the atoms.

Assume all atoms but $\boldsymbol{d}_j$ are fixed and denote by $\mathcal{I}_j$ the indices of the signals that use $\boldsymbol{d}_j$ in their representation. If atom $\boldsymbol{d}_j$ is ignored, then the representation error of these signals is

$$\boldsymbol{F} = \boldsymbol{Y}_{\mathcal{I}_j} - \sum_{i \neq j} \boldsymbol{d}_i \boldsymbol{X}_{i,\mathcal{I}_j}, \tag{3}$$

where $\boldsymbol{Y}_{\mathcal{I}_j}$ is the restriction of the signals matrix to the columns with indices in $\mathcal{I}_j$ and $\boldsymbol{X}_{i,\mathcal{I}_j}$ is the i-th row of the representation matrix restricted to the same columns. Similarly to K-SVD, regularized K-SVD aims to optimize the error function

$$\phi(\boldsymbol{d}_j, \boldsymbol{X}_{j,\mathcal{I}_j}) = \|\boldsymbol{F} - \boldsymbol{d}_j \boldsymbol{X}_{j,\mathcal{I}_j}\|_F^2 + \mu \|\boldsymbol{X}_{j,\mathcal{I}_j}\|^2 \tag{4}$$

and thus find the optimal atom $\boldsymbol{d}_j$ and the corresponding optimal representation coefficients $\boldsymbol{X}_{j,\mathcal{I}_j}$; this function results immediately from (2) when all other atoms and representations are fixed; in particular, since all rows of $\boldsymbol{X}$ but the j-th are fixed, the minimization of $\|\boldsymbol{X}\|_F^2$ can be reduced to that of $\|\boldsymbol{X}_{j,\mathcal{I}_j}\|^2$.

So, with lighter notation ($\boldsymbol{d}$ for $\boldsymbol{d}_j$ and $\boldsymbol{x}^T$ for $\boldsymbol{X}_{j,\mathcal{I}_j}$, where $\boldsymbol{x}$ is a column vector), our central problem is

$$\min_{\|\boldsymbol{d}\|=1,\boldsymbol{x}} \quad \phi(\boldsymbol{d}, \boldsymbol{x}^T). \tag{5}$$

Proposition 1: Using the SVD decomposition

$$\boldsymbol{F} = \sum_{i=1}^{r} \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^T, \tag{6}$$

where the rank $r \geq 1$ is irrelevant, the solution of (5) is

$$\boldsymbol{d} = \boldsymbol{u}_1, \quad \boldsymbol{x} = \frac{1}{1+\mu} \sigma_1 \boldsymbol{v}_1. \tag{7}$$

Proof. Remark first that

$$\phi(\boldsymbol{d}, \boldsymbol{x}^T) = (1+\mu)\|\boldsymbol{x}\|^2 - 2\boldsymbol{x}^T \boldsymbol{F}^T \boldsymbol{d} + \|\boldsymbol{F}\|_F^2. \tag{8}$$

We express $\boldsymbol{d}$ and $\boldsymbol{x}$ in the bases formed by the singular vectors of $\boldsymbol{F}$:

$$\boldsymbol{d} = \sum_{i=1}^{m} \rho_i \boldsymbol{u}_i, \quad \boldsymbol{x} = \sum_{i=1}^{N} \theta_i \boldsymbol{v}_i.$$

We denote $t = \|\boldsymbol{x}\|$ and (the atom being always normalized) note that

$$\sum_{i=1}^{m} \rho_i^2 = \|\boldsymbol{d}\|^2 = 1, \quad \sum_{i=1}^{N} \theta_i^2 = \|\boldsymbol{x}\|^2 = t^2. \tag{9}$$

Since

$$\boldsymbol{F}^T \boldsymbol{d} = \sum_{i=1}^{r} \sigma_i \boldsymbol{v}_i \boldsymbol{u}_i^T \boldsymbol{d} = \sum_{i=1}^{r} \sigma_i \rho_i \boldsymbol{v}_i,$$

the function to be minimized becomes

$$\phi(\boldsymbol{d}, \boldsymbol{x}^T) = (1+\mu)t^2 - 2\sum_{i=1}^{r} \sigma_i \rho_i \theta_i + \|\boldsymbol{F}\|_F^2.$$

Since σ_i form a decreasing sequence and the norm constraints (9) hold, the Cauchy-Schwarz inequality gives

$$\left(\sum_{i=1}^{r} \sigma_i \rho_i \theta_i \right)^2 \leq \left(\sum_{i=1}^{r} \sigma_i^2 \rho_i^2 \right) \sum_{i=1}^{N} \theta_i^2 \leq \sigma_1^2 t^2.$$

Equality is attained for $\rho_i = 0$, $\theta_i = 0$, for all $i > 1$, hence the objective function becomes

$$\phi(\boldsymbol{d}, \boldsymbol{x}^T) = (1+\mu)t^2 - 2\sigma_1 t + \|\boldsymbol{F}\|_F^2$$

and depends now on a single variable; its minimum is reached for $t = \sigma_1/(1+\mu)$, which gives (7). ■

Remark 1: In [7], the function (4) was written in the form

$$\phi(\boldsymbol{d}, \boldsymbol{x}^T) = \left\| \begin{bmatrix} \boldsymbol{F} \\ \boldsymbol{0} \end{bmatrix} - \begin{bmatrix} \boldsymbol{d} \\ \sqrt{\mu} \end{bmatrix} \boldsymbol{x}^T \right\|_F^2 \tag{10}$$

and it was argued that the minimum cannot be found via SVD: applying SVD directly to $\boldsymbol{F}$ gives the standard K-SVD solution; applying SVD to the extended matrix from (10) gives an extended atom with 0 on the last position instead of $\sqrt{\mu}$. Indeed, since minimizing (10) is a *structured* rank-1 approximation, one cannot expect SVD to supply a solution like in the standard approximation.

However, an alternative proof of Prop. 1 shows the relation with a usual rank-1 approximation. Since $\|\boldsymbol{d}\| = 1$, it follows that $\|\boldsymbol{x}\| = \|\boldsymbol{d}\boldsymbol{x}^T\|_F$. Introducing the rank-1 variable $\boldsymbol{\Upsilon} = \boldsymbol{d}\boldsymbol{x}^T$, it results that

$$\min_{\|\boldsymbol{d}\|=1, \boldsymbol{x}} \phi(\boldsymbol{d}, \boldsymbol{x}^T) = \min_{\text{rank}\boldsymbol{\Upsilon}=1} \|\boldsymbol{F} - \boldsymbol{\Upsilon}\|_F^2 + \mu \|\boldsymbol{\Upsilon}\|_F^2. \tag{11}$$

A few algebraic manipulations lead to the equality

$$\begin{aligned} &\|\boldsymbol{F} - \boldsymbol{\Upsilon}\|_F^2 + \mu \|\boldsymbol{\Upsilon}\|_F^2 = \\ &\left\| \frac{1}{\sqrt{1+\mu}} \boldsymbol{F} - \sqrt{1+\mu}\boldsymbol{\Upsilon} \right\|_F^2 + \left(1 - \frac{1}{1+\mu}\right) \|\boldsymbol{F}\|_F^2. \end{aligned}$$

Since the second term is constant, it results that (11) has the same solution as

$$\min_{\text{rank}\boldsymbol{\Upsilon}=1} \left\| \frac{1}{1+\mu} \boldsymbol{F} - \boldsymbol{\Upsilon} \right\|_F^2, \tag{12}$$

whose solution is (7). ■

Relation (7) says that the only difference between regularized and standard K-SVD is that the coefficients are smaller by a factor of $1+\mu$ in the former algorithm. So, the regularized objective (2) explicitly leads to diminished coefficients.

For the sake of completeness, Algorithm 1 summarizes the operations of regularized K-SVD. Its complexity is practically the same as that of standard K-SVD, since the number of divisions with $1+\mu$ is not significant. The behavior is also similar. The dictionary update stage is guaranteed to reduce the objective function (2), since it consists of optimal block coordinate descent steps on an atom and its representation. However, the sparse coding stage may occasionally not reduce the objective, due to the suboptimality of OMP (and in general of sparse representation algorithms). Despite this fact, regularized K-SVD typically displays a descending trend of the objective values.

III. Regularized AK-SVD

Approximated K-SVD (AK-SVD) [8] can be regularized easily by adapting the basic idea that the atom and its representation coefficients are optimized successively and not simultaneously like in K-SVD. So,

Input: initial dictionary $\boldsymbol{D} \in \mathbb{R}^{m \times n}$

signals set $\boldsymbol{Y} \in \mathbb{R}^{m \times N}$

number of iterations K

Output: new dictionary $\boldsymbol{D}$

for $k = 1$ **to** K **do**

Sparse coding: keeping $\boldsymbol{D}$ fixed, compute sparse representations $\boldsymbol{X}$ with OMP

Dictionary update:

for $j = 1$ **to** n **do**

Gather in $\mathcal{I}_j$ the indices of signals using atom $\boldsymbol{d}_j$

Compute σ_1, $\boldsymbol{u}_1$, $\boldsymbol{v}_1$, the first singular value and vectors, respectively, of matrix (3)

Set new atom $\boldsymbol{d}_j = \boldsymbol{u}_1$

Set new representations $\boldsymbol{X}_{j,\mathcal{I}_j} = \sigma_1 \boldsymbol{v}_1/(1+\mu)$

end for

end for

Algorithm 1: Regularized K-SVD algorithm

the problem (5) is not solved exactly; instead, the objective $\phi(\boldsymbol{d}, \boldsymbol{x}^T)$ is decreased by minimizing first on the atom $\boldsymbol{d}$ and then on the representation $\boldsymbol{x}$. The derivations below follow immediately by taking the appropriate derivatives in the expression (8) of the objective.

When the representations are fixed, the optimal atom in (5) is the same as in AK-SVD, namely

$$\boldsymbol{d} = \boldsymbol{F}\boldsymbol{x}/\|\boldsymbol{F}\boldsymbol{x}\|. \tag{13}$$

When the atom is fixed, the optimal representations are

$$\boldsymbol{x} = \frac{1}{1+\mu}\boldsymbol{F}^T\boldsymbol{d}. \tag{14}$$

Like in the regularized K-SVD formula (7), the diminishing factor $1+\mu$ applies to the AK-SVD relation for the representation.

To conclude, regularized AK-SVD consists of replacing (7) with (13) and (14) (in this order). Again, regularized AK-SVD has practically the same complexity as the standard version and similar convergence properties.

IV. NUMERICAL RESULTS

The purpose of the numerical experiments reported below is to compare the performances of SimCO and our regularized K-SVD as reflected by their different approaches to dictionary update through the regularized objective (2). Sparse coding is implemented via OMP; other sparse coding algorithms are certainly possible, but are outside the focus of this paper.

The first round of simulations is performed on synthetic data. Starting with a known dictionary, we randomly pick $s = 4$ atoms that we combine linearly to generate a sparse representation. The process is repeated N times to obtain the training data matrix $\boldsymbol{Y}$. We followed the experiment from [7] by using dictionaries of size $m = 16$ with $n = 32$ atoms and varying the training set size N. The DL process is iterated $K = 50$ times and for SimCO we use a single inner iteration. We executed 50 runs for each parametrization. The performance of each algorithm is compared in terms of its sparse representation distortion that is defined as $\|Y - DX\|_F^2/N$. When performing regularization, we set $\mu = 0.1$ for the first 30 DL iterations and $\mu = 0$ for the remaining 20.

The averaged results are presented in Figure 1 where we suffix the regularized versions with an 'r'. While regularized SimCO performs better than the standard algorithms, a behavior that is typical for low number N of signals, it is clearly visible that the proposed methods are always ahead.

We also tested the regularized DL algorithms on image representation tasks. We used 8×8 image patches from the USC-SIPI [18] database that we vectorized as signals of size $m = 64$. The following experiment was performed with $N = 4096$ random patches that we used for training $n = 256$ dictionary atoms for $K = 50$ DL iterations. For each method and parametrization we executed 10 runs each starting with the same initial dictionary and training set. The simulations were performed with Matlab R2015a on an Intel i7-3930K CPU running at a maximum clock frequency of 3.2GHz with 16GB of system memory.

In Table I we show the averaged RMSE $= \|Y - DX\|_F/\sqrt{mN}$ of KSVD, AK-SVD and SimCO. For SimCO we show multiple results with varied number of inner dictionary update iterations. We present regularization results with $\mu = 0.05$, like in [7].

Even though K-SVD variants are the clear winners for all sparsity targets, regularization starts to pay off only for larger values of s where finding linearly dependent atoms is more likely. As stated by the SimCO authors, plain SimCO is rather weak, but even with regularization it takes 50 inner dictionary update iterations to start obtaining results somewhat comparable to K-SVD.

In terms of complexity, our algorithms are clearly superior to SimCO. For example, with $s = 8$ in the test above, a single dictionary update (the "**for** j" loop in Algorithm 1) takes about 1 second for K-SVDr

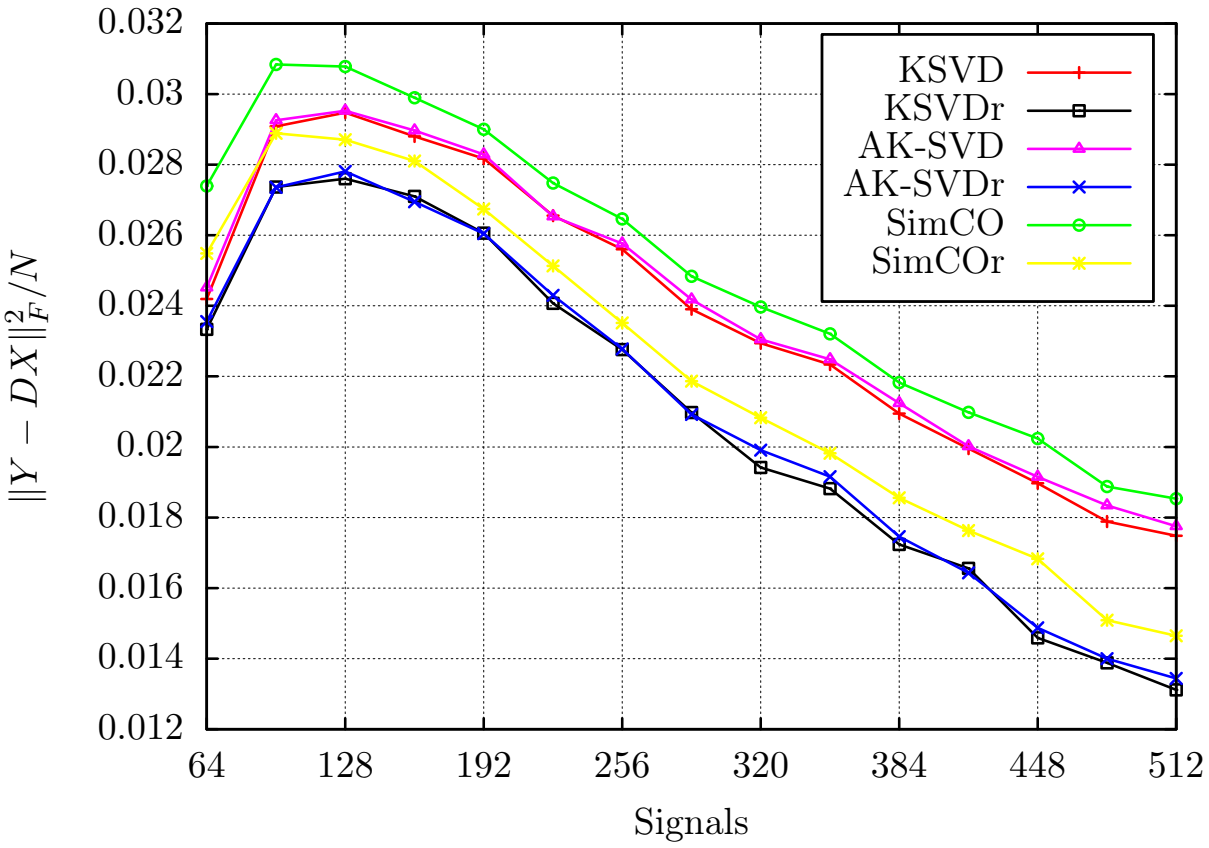

Fig. 1. Final errors averaged over 50 runs ($\mu = 0.1$).

and less than 0.2 seconds for AK-SVDr. For the same task, an inner SimCO update iteration takes about 1.5 seconds. However, SimCO needs many inner iterations to compete with regularized K-SVD in terms of achieved error level, hence the computational advantage of our algorithms. Of course, since we used plain Matlab implementations, these times give only a rough estimate of the performance that can be achieved. Still, the difference is too large in favor of our algorithms to be radically changed by any implementation.

We also performed image denoising experiments where we trained the dictionary with $n = 256$ atoms based on $N = 1000$ samples, each built by randomly picking 8×8 patches from the corrupted image. The DL process was performed with $K = 10$ iterations. SimCO also executed an extra 50 inner dictionary update iterations. Finally, we used the learned dictionary to reconstruct the image. The entire process, data dimensions and parametrization are identical to the ones presented in [7] except for the fact that they fix $\mu = 0.05$ where we chose to explore varied regularization values and also look at the standard

TABLE I

RMSE FOR THE STANDARD AND REGULARIZED ($\mu = 0.05$) K-SVD

Method	$s = 4$	$s = 6$	$s = 8$	$s = 10$	$s = 12$
KSVD	**0.0362**	0.0301	0.0266	0.0243	0.0227
KSVDr	0.0371	0.0318	0.0270	**0.0239**	**0.0219**
AK-SVD	0.0363	**0.0299**	**0.0262**	0.0240	0.0230
AK-SVDr	0.0373	0.0314	0.0270	0.0241	**0.0219**
SimCO(1)	0.0665	0.0628	0.0597	0.0556	0.0517
SimCOr(1)	0.0622	0.0562	0.0503	0.0463	0.0402
SimCOr(10)	0.0513	0.0455	0.0408	0.0369	0.0333
SimCOr(20)	0.0467	0.0418	0.0373	0.0337	0.0308
SimCOr(30)	0.0446	0.0391	0.0354	0.0323	0.0298
SimCOr(40)	0.0425	0.0373	0.0342	0.0310	0.0287
SimCOr(50)	0.0408	0.0359	0.0326	0.0300	0.0280

TABLE II

IMAGE DENOISING PSNR AND EXECUTION TIMES

	$\mu = 0$		$\mu = 0.05$		$\mu = 0.1$		$\mu = 0.5$	
Method	PSNR	Time	PSNR	Time	PSNR	Time	PSNR	Time
KSVD	28.35	15.2	28.40	15.9	28.38	16.2	28.37	17.1
AK-SVD	28.34	4.8	28.36	5.1	28.40	5.2	28.38	5.2
SimCO	28.13	170	28.27	200	28.36	184	28.32	197

versions.

The results are shown in Table II. Looking at the peak signal-to-noise ratio (PSNR) columns, we can see that the K-SVD methods are always providing better denoising, independent of how we choose μ. The time columns show the DL execution time measured in seconds. Here we note the significant advantage of the proposed methods: regularized AK-SVD is about 3 times faster that regularized K-SVD, which is more than 10 times faster than SimCO. We note that the differences between the regularized and standard algorithms are minor, a fact not mentioned in [7].

We have thus covered the numerical experiments from [7] that involved regularized DL and in all of them our regularized K-SVD was superior to SimCO.

V. Conclusions

Regularized dictionary learning is useful for its better numerical behavior, compared to the standard DL problem. We have derived explicit relations for solving the basic optimization problem involving a single atom and its representation coefficients, thus obtaining a regularized K-SVD algorithm. Like standard K-SVD, the algorithm requires the computation of the first singular vectors of a matrix, the only difference being a dampening of the representation coefficients. Numerical experiments show that our algorithm is preferable to SimCO (until now the algorithm of choice for regularized DL), being slightly superior in terms of end results but much faster.

Acknowledgment. We thank Prof. Wenwu Wang for sharing the implementation of SimCO.

References

[1] R. Rubinstein, A.M. Bruckstein, and M. Elad, "Dictionaries for Sparse Representations Modeling," *Proc. IEEE*, vol. 98, no. 6, pp. 1045–1057, June 2010.

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

[3] Q. Zhang and B. Li, *Dictionary Learning in Visual Computing*, Morgan & Claypool, 2015.

[4] K. Engan, S.O. Aase, and J.H. Husoy, "Method of Optimal Directions for Frame Design," in *IEEE Int. Conf. Acoustics Speech Signal Proc.*, 1999, vol. 5, pp. 2443–2446.

[5] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation," *IEEE Trans. Signal Proc.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.

[6] Y.C. Pati, R. Rezaiifar, and P.S. Krishnaprasad, "Orthogonal Matching Pursuit: Recursive Function Approximation with Applications to Wavelet Decomposition," in *27th Asilomar Conf. Signals Systems Computers*, Nov. 1993, vol. 1, pp. 40–44.

[7] W. Dai, T. Xu, and W. Wang, "Simultaneous Codeword Optimization (SimCO) for Dictionary Update and Learning," *IEEE Trans. Signal Proc.*, vol. 60, no. 12, pp. 6340–6353, Dec. 2012.

[8] R. Rubinstein, M. Zibulevsky, and M. Elad, "Efficient Implementation of the K-SVD Algorithm Using Batch Orthogonal Matching Pursuit," Tech. Rep. CS-2008-08, Technion Univ., Haifa, Israel, 2008.

[9] J. Mairal, F. Bach, and J. Ponce, "Task-Driven Dictionary Learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 4, pp. 791–804, Apr. 2012.

[10] B. Shen, B.D. Liu, and Q. Wang, "Elastic Net Regularized Dictionary Learning for Image Classification," *Multimed. Tools Appl.*, vol. 75, pp. 8861–8874, 2016.

[11] H.D. Liu, M. Yang, Y. Gao, Y. Yin, and L. Chen, "Bilinear Discriminative Dictionary Learning for Face Recognition," *Pattern Recognition*, vol. 47, no. 5, pp. 1835–1845, May 2014.

[12] A. Barla, S. Salzo, and A. Verri, "Regularized Dictionary Learning," in *Regularization, Optimization, Kernels, and Support Vector Machines*, J.A.K. Suykens, M. Signoretto, and A. Argyriou, Eds., pp. 27–52. CRC Press, 2015.

[13] J. Yang, X. Zhang, W. Peng, and Z. Liu, "A Novel Regularized K-SVD Dictionary Learning Based Medical Image Super-resolution Algorithm," *Multimed. Tools Appl.*, vol. 75, no. 21, pp. 13107–13120, 2016.

[14] D. Needell and R. Vershynin, "Uniform Uncertainty Principle and Signal Recovery via Regularized Orthogonal Matching Pursuit," *Found. Comp. Math.*, vol. 9, no. 3, pp. 317–334, June 2009.

[15] C.D. Sigg, T. Dikk, and J.D. Buhmann, "Learning Dictionaries With Bounded Self-Coherence," *IEEE Signal Proc. Letters*, vol. 19, no. 19, pp. 861–865, Dec. 2012.

[16] M. Nejati, S. Samavi, N. Karimi, S.M.R. Soroushmehr, and K. Najarian, "Boosted Dictionary Learning for Image Compression," *IEEE Trans. Image Proc.*, vol. 25, no. 10, pp. 4900–4915, Oct. 2016.

[17] P. Irofti and B. Dumitrescu, "Regularized Algorithms for Dictionary Learning," Int. Conf. Communications, Bucharest, Romania, pp.439–442, 2016.

[18] A.G. Weber, "The USC-SIPI Image Database," 1997.