



**HAL**  
open science

# Multivariate dictionary learning and shift & 2D rotation invariant sparse coding

Quentin Barthélemy, Anthony Larue, Aurélien Mayoue, David Mercier,  
Jerome I. Mars

► **To cite this version:**

Quentin Barthélemy, Anthony Larue, Aurélien Mayoue, David Mercier, Jerome I. Mars. Multivariate dictionary learning and shift & 2D rotation invariant sparse coding. SSP 2011 - 2011 IEEE Workshop on Statistical Signal Processing, Jun 2011, Nice, France. pp.649-652, 10.1109/SSP.2011.5967783 . hal-00625352

**HAL Id: hal-00625352**

**<https://hal.science/hal-00625352>**

Submitted on 21 Sep 2011

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# MULTIVARIATE DICTIONARY LEARNING AND SHIFT & 2D ROTATION INVARIANT SPARSE CODING

Q. Barthélemy\*, A. Larue\*, A. Mayoue\*, D. Mercier<sup>o</sup>

J. I. Mars

CEA, LIST

\*Laboratoire d’Outils pour l’Analyse de Données

<sup>o</sup>Laboratoire Information, Modèles et Apprentissage  
Gif-sur-Yvette, F-91191, France

GIPSA-lab, DIS

UMR 5216 CNRS, Grenoble INP  
Grenoble, F-38402, France

## ABSTRACT

In this article, we present a new tool for sparse coding : Multivariate DLA which empirically learns the characteristic patterns associated to a multivariate signals set. Once learned, Multivariate OMP approximates sparsely any signal of this considered set. These methods are specified to the 2D rotation-invariant case. Shift and rotation-invariant cases induce a compact learned dictionary. Our methods are applied to 2D handwritten data in order to extract the elementary features of this signals set.

**Index Terms**— Sparse coding ; Multivariate DLA ; Multivariate OMP ; Shift-invariant ; Rotation-invariant ; Levenberg-Marquardt ; Handwritten characters.

## 1. INTRODUCTION

In signal processing, sparsity is a very interesting property which becomes more and more popular. Usually, it is used as a criterion in a transformed domain for compression, compress sensing, denoising, etc [1]. As we will then consider, sparsity can also be used as a feature extraction method, to make emerge from data elements containing relevant information. In our application, we extract motion primitives of the handwriting.

In a Hilbert space, we define the matrix inner product<sup>1</sup> as  $\langle A, B \rangle = \text{trace}(B^H A)$  and its associated Frobenius norm denoted  $\|\cdot\|$ . We consider a signal  $y \in \mathbb{C}^N$  of  $N$  samples and a normed dictionary  $\Phi \in \mathbb{C}^{N \times M}$  composed of  $M$  atoms  $\{\phi_m\}_{m=1}^M$ . The decomposition of the signal  $y$  is done on the dictionary  $\Phi$  such that  $y = \Phi x + \epsilon$ , assuming  $x \in \mathbb{C}^M$  the coding coefficients and  $\epsilon \in \mathbb{C}^N$  the residual error. The dictionary is said redundant when  $M > N$  : the linear system is thus under-determined and accepts several possible solutions. The introduction of constraints such as positivity, sparsity or other, allows to regularize the solution. The decomposition under sparsity constraint is formalized by :  $\min_x \|x\|_0$  s.t.  $\|y - \Phi x\|^2 \leq C_0(P_0)$ , in which  $C_0$  is a constant and  $\|x\|_0$  the  $\ell_0$  pseudo-norm defined as the cardinal of the  $x$  support<sup>2</sup>. In order to solve  $(P_0)$ , we want to determine the dictionary  $\Phi$  which fits the set of the studied signals. That means  $\Phi$  contains dedicated atoms allowing to sparsely code any signal of this set. To provide the decomposition sparsity, a first approach consists in the union of several classical dictionaries such as wavelets, curvelets and others with fast transforms [1] : the main drawback is the choice of these dictionaries. A second approach, called *sparse coding*, is a data driven learning method which

adapts atoms to elementary patterns characteristic of the studied set [2, 3, 4]. The obtained atoms do not belong to classical dictionaries : they are appropriate to the considered application.

In this paper, we briefly present the existing sparse approximation and dictionary learning algorithms. We look at the multivariate and shift-invariant cases. We then expose our new methods : Multivariate OMP (Orthogonal Matching Pursuit) and Multivariate DLA (Dictionary Learning Algorithm), and their specifications to the 2D rotation-invariant case. As a validation, proposed methods are applied to handwritten signals : results are shown and then discussed.

## 2. STATE OF THE ART

### 2.1. Sparse Approximation Algorithms

In general, finding the sparsest solution of the coding problem  $(P_0)$  is NP-hard. To overcome this difficulty, one way consists in relaxing  $(P_0)$  with a  $\ell_1$  norm :  $\min_x \|x\|_1$  s.t.  $\|y - \Phi x\|^2 \leq C_1(P_1)$ , with  $C_1$  a constant.  $(P_1)$  is a convex optimization problem having a single minimum. Different algorithms solving this problem are presented in [5] such as methods based on Interior Point, Homotopy, Iterative Thresholding, etc. A high coherence<sup>3</sup>  $\mu_\Phi$  does not ensure that these algorithms recover the optimal  $x$  support, and if it is the case, the convergence is long.

Another way consists in simplifying  $(P_0)$  in a sub-problem :  $\min_x \|y - \Phi x\|^2$  s.t.  $\|x\|_0 \leq C'_0(P'_0)$ , with  $C'_0$  a constant. Pursuit algorithms [5] tackle sequentially  $(P'_0)$ , but this optimization is non-convex. The OMP algorithm [6] solves the least squares problem increasing iteratively the constant  $C'_0$ . The obtained solution is sub-optimal because the support recovery is not guaranteed especially for a high coherence  $\mu_\Phi$ . However, it is fast when one searches very few coefficients.

### 2.2. Dictionary Learning Algorithms

The aim of DLA is to find a dictionary adapted to the signals set we want to code. Classical learning methods alternate between two steps : i)  $\Phi$  is fixed and  $x$  is obtained by sparse approximation (Section 2.1), ii)  $x$  is fixed and  $\Phi$  is updated. The update is based on criteria such as Maximum Likelihood (ML) [2], Maximum A Posteriori [3] or Majorization-Minimization. There are also simultaneous learning methods such as K-SVD presented in [4], the bibliography of which synthesizes well the state of the art. Some of these methods have been modified to deal with atoms overlappings in the shift-invariant case : extensions of MOD [7] and of K-SVD [8].

1. The conjugate transpose operator is denoted  $(\cdot)^H$ .

2. The support of  $x$  is  $\text{support}(x) = \{i \in \mathbb{N}_N : x_i \neq 0\}$ .

3. The coherence of the dictionary  $\Phi$  is  $\mu_\Phi = \max_{i \neq j} |\langle \phi_i, \phi_j \rangle|$ .

### 3. MULTIVARIATE AND SHIFT-INVARIANT CASES

In the multivariate case, the studied signal becomes  $y \in \mathbb{C}^{N \times V}$ , denoting  $V$  the number of components. Two problems can be considered depending on the  $\Phi$  and  $x$  natures :

- $\Phi \in \mathbb{C}^{N \times M}$  univariate and  $x \in \mathbb{C}^{M \times V}$  multivariate, the common case handled by multichannel sparse approximation algorithms [9, 10, 11, 12].
- $\Phi \in \mathbb{C}^{N \times M \times V}$  multivariate and  $x \in \mathbb{C}^M$  univariate<sup>4</sup>, case only evoked in [13] for sparse approximation, but with a particular dictionary template. In the present work, we will focus on this case, with  $\Phi$  multivariate and normed.

In the shift-invariant case, we want to code the signal  $y$  as a sum of few structures, named kernels, characterized independently of their positions. The  $L$  shiftable kernels of the compact  $\Psi$  dictionary are replicated at all positions to provide the  $M$  atoms of the  $\Phi$  dictionary. Kernels  $\{\psi_l\}_{l=1}^L$  can have different lengths : zero-padding is done to make them all have  $N$  samples. The  $N$  samples of the signal  $y$ , the residue  $\epsilon$ , the atoms  $\phi_m$  and the kernels  $\psi_l$  are indexed by  $t$ . Considering a kernel  $\psi_l$ ,  $\sigma_l$  is a subset of the  $N$  indexes  $t$ . Translated at all positions  $\tau \in \sigma_l$ , the kernels  $\psi_l(t)$  generate all atoms  $\phi_m(t)$  :

$$y(t) = \sum_{m=1}^M x_m \phi_m(t) + \epsilon(t) = \sum_{l=1}^L \sum_{\tau \in \sigma_l} x_{l,\tau} \psi_l(t - \tau) + \epsilon(t) \quad (1)$$

To sum up, the multivariate signal  $y$  is approximated as a weighted sum of few shiftable multivariate kernels  $\psi_l$ .

### 4. METHODS PRESENTATION

We now expose our new methods for sparse coding : Multivariate OMP (M-OMP) for multivariate sparse approximation, Multivariate DLA (M-DLA) for multivariate dictionary learning and their specifications to the 2D Rotation-Invariant (2DRI) case.

#### 4.1. Multivariate OMP

Sparse approximation can be achieved by any algorithm able to overcome the high coherence due to the shift-invariant case. OMP is chosen for its speed : a more precise description is given in [6]. We modify it to handle the multivariate case described previously (Section 3) and we name it Multivariate OMP (Algorithm 1). Denoting  $k$  the current iteration, the inner product between  $\epsilon^{k-1}$  and each atom is now replaced by the correlation with each kernel (step 1), generally computed by FFT. The orthogonal projection (step 4) is often computed recursively by different methods : we choose the block matrix inversion one [6]. The obtained vector  $x^k$  is reduced to its active (*i.e.* nonzero) coefficients. The multivariate case is taken account in selection (step 2) and in the orthogonal projection where the multivariate signal  $y$  (*resp.* dictionary  $D$ ) is unfolded along the components dimension in a univariate vector  $y_j$  (*resp.* matrix  $D_j$ ). Compared with the original OMP, the complexity of the M-OMP is only increased by a factor of  $V$ , the number of components.

#### 4.2. Multivariate DLA

Our learning method named Multivariate DLA (Algorithm 2) is an alternation between two steps : a sparse approximation step done

4.  $\Phi x$  is considered as a element-wise product along dimension  $M$ .

5. The complex correlation between the  $u^{th}$  components of the multivariate signals  $a$  and  $b$  is denoted  $\Gamma \{a[u], b[u]\}$ .

---

#### Algorithm 1: $x = \text{Multivariate\_OMP}(y, \Psi)$

---

**initialization** :  $k = 1$ ,  $\epsilon^0 = y$ , dictionary  $D^0 = \emptyset$   
**repeat**  
    1. Correlations<sup>5</sup> : **for**  $l \leftarrow 1$  **to**  $L$  **do**  
         $\left[ C_l^k(\tau) \leftarrow \sum_{u=1}^V \Gamma \{ \epsilon^{k-1}[u], \psi_l[u] \}(\tau) \right.$   
    2. Selection :  $(l_{max}^k, \tau_{max}^k) \leftarrow \arg \max_{l,\tau} |C_l^k(\tau)|$   
    3. Active Dictionary :  $D^k \leftarrow D^{k-1} \cup \psi_{l_{max}^k}(t - \tau_{max}^k)$   
    4. Active Coefficients :  $x^k \leftarrow \arg \min_x \|y - D^k x\|^2$   
    5. Residue :  $\epsilon^k \leftarrow y - D^k x^k$   
    6.  $k \leftarrow k + 1$   
**until** convergence

---

by M-OMP and a dictionary update step. M-DLA is applied on training signals  $\{y_p\}_{p=1}^P$  representative of the studied set. Our update step is based on the ML criterion [2], usually optimized by the Stochastic Gradient method. To achieve this optimization, we choose a Levenberg-Marquardt 2<sup>nd</sup> order Gradient Descent [14] which increases the convergence speed, blending Stochastic Gradient and Gauss-Newton methods. The current iteration is denoted  $i$ . For each multivariate kernel  $\psi_l$ , the update rule is given by :

$$\psi_l^i(\underline{t}) = \psi_l^{i-1}(\underline{t}) + (H_l^i + \lambda^i \cdot I)^{-1} \cdot \sum_{\tau \in \sigma_l} x_{l,\tau}^* \epsilon^{i-1}(\underline{t} + \tau) \quad (2)$$

with  $\underline{t}$  the indexes limited to the  $\psi_l$  temporal support,  $(\cdot)^*$  the conjugate operator,  $\lambda$  the adaptive descent step and  $H_l$  the average hessian computed for the kernel (and not for each sample). In atoms overlappings cases, the learning method can become unbalanced due to the error done on the gradient estimation. We slightly overestimate the hessian  $H_l$  to compensate this phenomenon. The update step, which now stabilizes the method, is called LM-modif (step 2). Moreover, kernels are normalized at the end of each iteration and their lengths are modified depending on the energy present in their edges.

Added to the non-convex optimization of the M-OMP, the convergence of the M-DLA towards the global minimum is not guaranteed owing to its alternating minimization. However we find a minimum, local or global, which assures the solution sparsity.

---

#### Algorithm 2: $\Psi = \text{Multivariate\_DLA}(\{y_p\}_{p=1}^P)$

---

**initialization** :  $i = 1$ ,  $\Psi^0 = \{\text{L kernels of white noise}\}$   
**repeat**  
    **for**  $p \leftarrow 1$  **to**  $P$  **do**  
        1. Sparse approximation :  $x^i \leftarrow \text{M-OMP}(y_p, \Psi^{i-1})$   
        2. Dictionary update :  $\Psi^i \leftarrow \text{LM-modif}(y_p, x^i, \Psi^{i-1})$   
        3.  $i \leftarrow i + 1$   
**until** convergence

---

#### 4.3. 2D Rotation-Invariant case

Studying bivariate real signals, 2D movements for example, we aspire to characterize them independently of their orientations. The rotation-invariance implies introducing a  $\theta_{l,\tau}$  angle rotation matrix  $R$  for each bivariate kernel  $\psi_l(t - \tau)$ . Equation (1) becomes :

$$y(t) = \sum_{l=1}^L \sum_{\tau \in \sigma_l} x_{l,\tau} R(\theta_{l,\tau}) \psi_l(t - \tau) + \epsilon(t) \quad (3)$$

Now, in the selection step (Algorithm 1, step 2), the aim is to find the angle  $\theta_{l,\tau}^k$  which maximizes the correlations  $|C_l^k(\tau, \theta_{l,\tau})|$ . A naive approach consists in sampling  $\theta_{l,\tau}$  into  $\Theta$  angles and to add a degree of freedom in the correlations computation (Algorithm 1, step 1). The complexity is increased by a factor of  $\Theta$  with respect to the M-OMP used in the bivariate real case.

To avoid this additional cost, we transform the  $y$  signal from  $\mathbb{R}^{N \times 2}$  to  $\mathbb{C}^N$  and we apply M-OMP : coding coefficients  $x$  are now complex. The modulus gives the coefficient amplitude and the argument gives the rotation angle. Now able to rotate, kernels are no longer learned through a particular orientation as in the previous approach said *oriented* ( $V = 2, y \in \mathbb{R}^{N \times 2}$ ). Thus, kernels are shift and rotation-invariant, providing a *non-oriented* decomposition ( $V = 1, y \in \mathbb{C}^N$ ). This 2DRI specification of the approximation (*resp.* learning) method is further denoted 2DRI-OMP (*resp.* 2DRI-DLA).

## 5. APPLICATION DATA

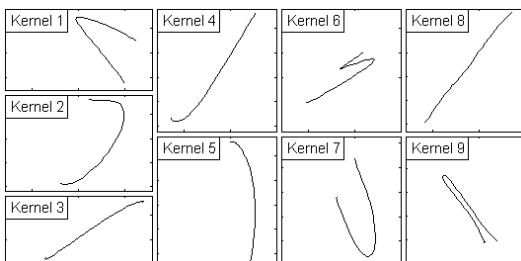
Our methods are applied to the *Character Trajectories* signals available on the UCI database [15] and initially dealt with a probabilistic model and an EM learning method [16], but without real sparsity in the resulting decompositions. Data are composed of a hundred occurrences of 20 letters written by the same person. The temporal signals are the cartesian pen tip velocities  $v_x$  and  $v_y$ .

We aim at learning an adapted dictionary in order to code sparsely velocity signals. Dictionary is learned on the first 20 occurrences of each letter and the approximation method is tested on the remaining ones. Velocity signals, on which our methods are applied, are integrated only to display the associated trajectories.

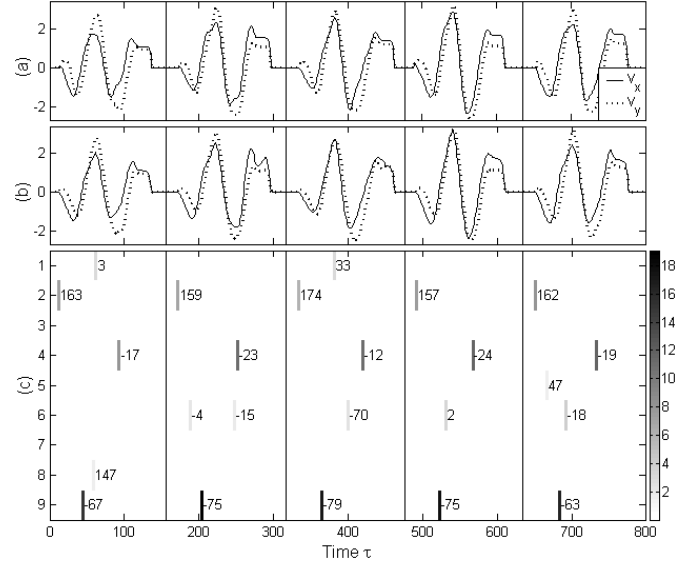
## 6. RESULTS

Results are directly presented for the non-oriented case. The integrated kernels dictionary (Fig 1) shows that the 2DRI-DLA has successfully extracted motion primitives. Indeed, straight and curved strokes correspond to the elementary patterns of the set of handwritten signals.

To evaluate the sparse coding qualities, decompositions of 5 occurrences of the letter  $d$  on this dictionary are considered in (Fig 2). The velocity signal (Fig 2a) (*resp.* Fig 2b)) is the original (*resp.* reconstructed *i.e.* approximated) signal, composed of the real part  $v_x$  (in solid line) and the imaginary part  $v_y$  (dotted). The relative mean square error on velocities is around 12% with 4-5 atoms used for the reconstruction. Coding coefficients  $x_{l,\tau}$  are displayed by a time-kernel representation called spikegram (Fig 2c). It condenses



**Fig. 1.** Dictionary, learned by 2DRI-DLA, of the trajectories associated to kernels.



**Fig. 2.** Original (a) and reconstructed (b) velocity signals of 5 occurrences of the letter  $d$ , and their associated spikegram (c).

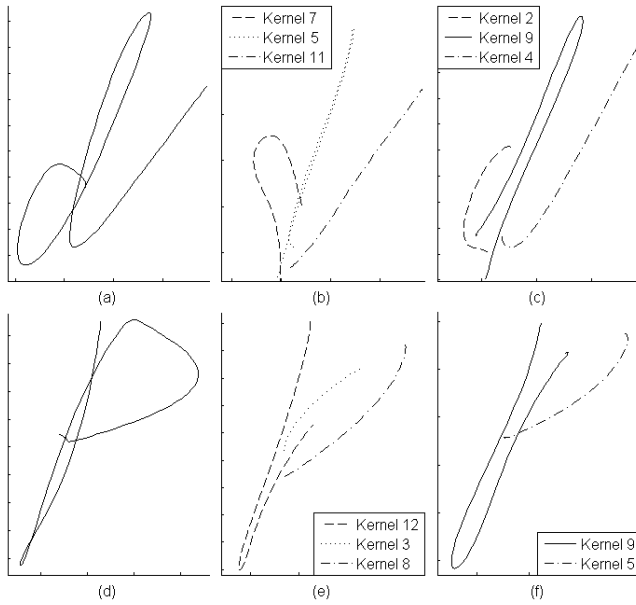
four indications : the temporal position  $\tau$  (abscissa), the kernel index  $l$  (ordinate), the coefficient amplitude  $|x_{l,\tau}|$  (gray level) and the rotation angle  $\theta_{l,\tau}$  (number next to spike, in degree). The low number of atoms used for signals reconstruction shows the decompositions sparsity. Primary atoms are the large amplitude ones : they concentrate relevant information. Secondary atoms code variabilities between different realizations. The decompositions reproducibility, highlighted by the primary atoms repetition (amplitudes and angles values) of the different occurrences, is the proof of an adapted dictionary.

The trajectory of the original letter  $d$  (Fig 3a) (*resp.*  $p$  (Fig 3d)) is reconstructed with its primary atoms, comparing the oriented case (Fig 3b) (*resp.* Fig 3e)) and the non-oriented one (Fig 3c) (*resp.* Fig 3f)). For instance, letter  $d$  (Fig 3c) is rebuilt as the sum of the kernels 2, 4 and 9 (Fig 1) specified by the amplitudes and the angles of the spikegram (Fig 2c). We now focus on the principal vertical stroke common to letters  $d$  and  $p$  (Fig 3a and Fig 3d). To code it, the oriented case uses two different kernels : kernel 5 for  $d$  (Fig 3b, dotted line) and kernel 12 for  $p$  (Fig 3e, dashed line). Whereas, the non-oriented case needs only one for the two letters : kernel 9 (Fig 3c and Fig 3f, solid line) used with an average difference of  $180^\circ$ . Thus, the non-oriented approach reduces the dictionary redundancy providing a kernel dictionary even more compact.

## 7. DISCUSSION

The dictionary learning allows to recover signals primitives. The resulting dictionary can be thought of as a catalog of elementary patterns dedicated to the considered application and having a physical meaning as opposed to classical dictionaries such as wavelets, curvelets, etc. Therefore, decompositions based on such a dictionary are made sparsely on the characteristic components of the studied signals set. Considering the reconstruction mean square error, the few kernels used shows the efficiency of this sparse coding approach.

The 2DRI approach reduces the dictionary size in two ways :



**Fig. 3.** Letter *d* (resp. *p*) : original (a) (resp. (d)), oriented reconstructed (b) (resp. (e)) and non-oriented reconstructed (c) (resp. (f)) trajectories.

- when the studied signals cannot rotate, like in the presented application. The non-oriented approach detects rotational invariants (vertical stroke of letters *d* and *p* for example) that reduces the dictionary size : from 12 to 9.
- when the studied signals can rotate : for example, when the acquiring tablet is revolved. To provide an adapted dictionary for sparse coding, the oriented approach needs to learn motion primitives for each of possible angles as opposed to the non-oriented case. That is the noticeable reduction of the dictionary size.

Thereby, shift and rotation-invariant cases provide a compact learned dictionary  $\Psi$  (Fig 1). Moreover, the non-oriented approach allows to be robust to any writing direction (tablet rotation) and to any writing inclination (intra and inter-users variabilities).

## 8. CONCLUSION

We have presented new tools : M-DLA for automatically learning the patterns characteristic of a multivariate signals set, with the dictionary update done by a ML criterion, and M-OMP for coding sparsely all signal of this set. They are specified to the 2D Rotation-Invariant case, respectively named 2DRI-DLA and 2DRI-OMP. Shift and rotation-invariant cases induce a compact learned dictionary. Their applications are dimension reduction, compression, denoising, gestures representation and analysis, and all other processing which is multivariate feature extraction based. Moreover, we apply these methods to motion signals that is new with regards to custom sparse coding applications.

The considered prospects are to compare our method with other learning methods appropriate to the shift-invariant case and to integrate a dilatation parameter to take in consideration the movement execution speed. These methods provide sparse descriptors, we also project to add a classification step to make gestures recognition.

## 9. REFERENCES

- [1] J.L. Starck, M. Elad, and D.L. Donoho, “Redundant multiscale transforms and their application for morphological component analysis,” *Advances in Imaging and Electron Physics*, vol. 132, pp. 287–348, 2004.
- [2] B.A. Olshausen and D.J. Field, “Sparse coding with an overcomplete basis set : a strategy employed by V1 ?,” *Vision Research*, vol. 37, pp. 3311–3325, 1997.
- [3] K. Kreutz-Delgado, J.F. Murray, B.D. Rao, K. Engan, T. Lee, and T.J. Sejnowski, “Dictionary learning algorithms for sparse representation,” *Neural Comput.*, vol. 15, pp. 349–396, 2003.
- [4] M. Aharon, M. Elad, and A. Bruckstein, “K-SVD : An algorithm for designing overcomplete dictionaries for sparse representation,” *IEEE Trans. on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, 2006.
- [5] J.A. Tropp and S.J. Wright, “Computational methods for sparse solution of linear inverse problems,” *Proc. of the IEEE*, vol. 98, no. 6, pp. 948–958, 2010.
- [6] Y.C. Pati, R. Rezaifar, and P.S. Krishnaprasad, “Orthogonal Matching Pursuit : recursive function approximation with applications to wavelet decomposition,” in *Asilomar Conf. on Signals, Systems and Comput.*, 1993.
- [7] K. Skretting, J.H. Husøy, and S.O. Aase, “General design algorithm for sparse frame expansions,” *Signal Proc.*, vol. 86, pp. 117–126, 2006.
- [8] B. Mailhé, S. Lesage, R. Gribonval, F. Bimbot, and P. Vandergheynst, “Shift-invariant dictionary learning for sparse representations : Extending K-SVD,” in *Proc. Eur. Signal Process. Conf., Lausanne*, 2008.
- [9] A. Lutoborski and V.N. Temlyakov, “Vector greedy algorithms,” *J. Complex.*, vol. 19, pp. 458–473, August 2003.
- [10] S.F. Cotter, B.D. Rao, K. Engan, and K. Kreutz-Delgado, “Sparse solutions to linear inverse problems with multiple measurement vectors,” *IEEE Trans. on Signal Processing*, vol. 53, no. 7, pp. 2477–2488, 2005.
- [11] J.A. Tropp, A.C. Gilbert, and M.J. Strauss, “Algorithms for simultaneous sparse approximation ; Part I : Greedy pursuit,” *Signal Proc.*, vol. 86, pp. 572–588, 2006.
- [12] J. Chen and X. Huo, “Theoretical results on sparse representations of multiple-measurement vectors,” *IEEE Trans. on Signal Processing*, vol. 54, no. 12, pp. 4634–4643, 2006.
- [13] R. Gribonval and M. Nielsen, “Beyond sparsity : Recovering structured representations by  $\ell_1$ -minimization and greedy algorithms. -Application to the analysis of sparse underdetermined ICA-,” Tech. Rep. PI-1684, IRISA, 2005.
- [14] K. Madsen, H.B. Nielsen, and O. Tingleff, “Methods for non-linear least squares problems, 2nd edition,” Tech. Rep., Technical University of Denmark, 2004.
- [15] A. Frank and A. Asuncion, “UCI machine learning repository,” <http://archive.ics.uci.edu/ml>, 2010.
- [16] B.H. Williams, M. Toussaint, and A.J. Storkey, “Modelling motion primitives and their timing in biologically executed movements,” *Advances in Neural Information Processing Systems*, vol. 20, pp. 1609–1616, 2008.