Sparse Representation over Overcomplete Dictionary Based on Bayesian Nonparametric Approximation

Yan He, Donghui Wang, and Miaoliang Zhu

***Abstract*—Sparse representation of signals over overcomplete dictionaries shows state-of-art results in lots of applications. Though the problem is NP-hard, approximate solutions are proposed based on a wide variety of techniques. In this paper, we propose a method over a learned dictionary based on nonparametric methods for this problem. The structure follows the two steps of normal dictionary learning procedure. In one step we fix the dictionary and learn the sparse coefficient vector based on Bayesian nonparametric variable selection, while in the other step we minimize the objective based on the dictionary with the coefficient vector fixed by matrix-inversion free procedure.**

***Index Terms*—Sparse representation, nonparametric methods, dictionary learning**

I. Introduction

Sparse representation of signals over overcomplete dictionaries shows state-of-art results in signal processing, compression, and feature extraction [1], [2]. Suppose $\mathbf{y} \in \mathbb{R}^m$ and $\mathbf{x} \in \mathbb{R}^k$ are the input signal and the coefficient vector respectively, the canonical form of this problem is given by

$$\min_{\mathbf{x}} \| \mathbf{x} \|_0 \quad s.t. \ \mathbf{y} = \mathbf{D}\mathbf{x} \tag{1}$$

where D is a $m \times k$ matrix which is called dictionary and $\| \cdot \|_0$ is the sparsity measure that counts the number of nonzero coefficients. Though the problem is NP-hard, approximate solutions are proposed based on a wide variety of techniques. The dictionary D in the sparse representations can either be chosen as a pre-specified set of functions, such as Wavelets[3], Curvelets [4], or designed by adapting its content to fit a given set of signal examples. A pre-specified dictionary is high structured and leads to fast numerical implementations, with the drawback of inflexibility to adapt the representation to the data. The second approach suggests using machine learning techniques to infer the dictionary from a set of examples. *Dictionary learning* [5], [6] is the process of finding a dictionary in which a given set of training samples has sparse representations or approximations. In traditional dictionary learning, one often starts with some initial dictionary and finds sparse approximations of the set of training signals while keeping the dictionary fixed. This is followed by a second step in which the sparse coefficients are kept fixed and the dictionary is optimized. The altering evaluation runs for a specific number of alternating optimizations or until a specific approximation error is reached. Dictionary learning produces much finer-tuned dictionaries compared to the pre-specified approaches, and performs better results in applications. However, the generation of an unstructured dictionary and the evaluation of the sparse approximation are significant computational challenge.

A simple method for dictionary generation is to add two or more orthogonal bases. Blockwise orthogonality can then be exploited to find the sparse approximation [7]. However, the dictionary model itself is relatively restrictive, and its training algorithm shows somewhat weak performance. Another famous Gabor dictionary [1] designs a dictionary with sampling the parameters of an analytic function. These designed dictionaries are efficient when we have some a priori information about the signal's generative model. In the method of optimal directions(MOD) [8], the best dictionary is found using the pseudo inverse of, followed by re-normalization of each atom. This normalization step can increase the total approximation error. The K-SVD method [9] keeps the support of the coefficient vectors fixed in the dictionary update step. Updates for each atom are found as the best normalized elementary function that matches the error. M.Zhou [10] firstly proposed a dictionary learning method based on nonparametric models. They modeled the sparse approximation problem with beta process and the dictionary was modeled with a Gaussian distribution with the means and the covariance matrix, which are updated during the inference. The paper did not give enough proven to the inference procedure, and the experiment results did not show a pleasant performance.

In this paper, we follow the two steps of normal dictionary learning procedure. In one step we fix the dictionary and learn the sparse coefficient vector based on Bayesian non-parametric variable selection, while in the other step we minimize the objective based on D with X fixed by matrix-inversion free procedure. The structure is based on a sparsity model of the dictionary atoms over nonparametric methods, which leads to a flexible dictionary representation. Combine the feature of image denoising problem, fast Gibbs sampling algorithms are shown.

This paper is organized as follows. We begin in Section II with a description of dictionary learning problem. In Section III, we consider the task of training the dictionary for examples by the nonparametric method and analyze the inference of the model. Simulation results are provided in

Manuscript received June 12, 2012; revised July 21, 2012. This work was supported in part by the China 973 Program (project No. 2010CB327905) and the National Natural Science Foundation of China(project No. 61071218).

The authors are with College of Computer Science and Technology, Zhejiang University, Hangzhou, China, and with Zhejiang Zhejiang Financial College, Hangzhou, China (e-mail: heheyan@126.com, dhwang@zju.edu.cn, zhum@zju.edu.cn).

Section IV. We summarize and conclude in Section V.

II. Problem Statement

Classical dictionary learning techniques consider a finite training set of signals **Y** = [**y**$_1$, ..., **y**$_n$] in $\mathbb{R}^{m\times n}$ and optimize the empirical cost function

$$f_n(\mathbf{D}) \triangleq \frac{1}{n}\sum_{i=1}^{n} l(\mathbf{y}_i, \mathbf{D}) \tag{2}$$

where **D** in $\mathbb{R}^{m\times k}$ is the dictionary, each column representing a basis vector, and *l* is a loss function such that *l*(**y**,**D**) should be small if **D** is "good" at representing the signal **y**. The number of samples *n* is usually large, whereas the signal dimension *m* is relatively small. In general, we also have $k \ll n$, and each signal only uses a few elements of **D** in its representation. The convex sets of matrices with bounded Frobenius norm [11]

$$\mathcal{D} = \{\mathbf{D}_{d\times m} : \ \|\mathbf{D}\|_F \le C_F^{1/2}\} \tag{3}$$

where C_F is a constant and the convex set of matrices with bounded column norm,

$$\mathcal{D} = \{\mathbf{D}_{d\times m} : \ \|\mathbf{d}_i\|_F \le C_C^{1/2}\} \tag{4}$$

where d_i is the *i*th column of the dictionary **D** and C_C is a constant.

In practice, one is usually not interested in the minimization of the empirical cost f_n(**D**) with high precision, but instead in the minimization of the expected cost

$$f(\mathbf{D}) \triangleq \mathbb{E}_{\mathbf{y}}[l(\mathbf{y}, \mathbf{D})] = \lim_{n\to\infty} f_n(\mathbf{D}) \tag{5}$$

where the expectation is taken relative to the probability distribution *p*(**y**) of the data.

III. Bayesian Methods for Sparse Dictionary Learning

The problem of minimizing the empirical cost fn(D) can be rewritten as a joint optimization problem with respect to the dictionary D and coefficients *x*, which is not jointly convex, but convex with respect to each of the two variables D and *x* when the other one is fixed [12]. We use the block relaxation technique to solve the problem [11], that is, in one step we fix D and minimize the coefficients *x*, while in the other step we minimize the objective based on D with *x* fixed. This alternating minimization continues until the algorithm converges to an accumulation point.

A. Sparse Coding

With the fixed dictionary **D**, the image denoising problem is described as

$$f(\mathbf{y}\,|\,\mathbf{x},\sigma^2) = \mathcal{N}_n(\mathbf{D}\mathbf{x}, \sigma^2\mathbf{I}) \tag{6}$$

where *y* is *n*×1, D is an *n*×*p* matrix, *x* is a *p*×1 vector of unknown regression coefficients, and *σ* is an unknown positive scalar. We follow the Bayesian variable selection approach [13], [14], which is similar to the variable selection methods. The variable selection problem arises when there is some unknown subset of the predictors with regression coefficients so small that it would be preferable to ignore them.

We index each of these possible 2^p subset choices by the vector $\gamma = (\gamma_1, ..., \gamma_p)^{\mathrm{T}}$, where $\gamma_i = 0$ or 1 if x_i is small or large, respectively. Following [15], we model the uncertainty underlying variable selection by a mixture prior $p(\mathbf{x}, \sigma^2, \gamma) = p(\mathbf{x}\,|\,\sigma^2, \gamma)p(\sigma^2\,|\,\gamma)p(\gamma)$ which can be conditionally specified as follows:

$$\begin{gathered} p(\mathbf{x}\,|\,\sigma^2,\gamma) = N(0, \sigma^2 \textstyle\sum_\gamma^* R_\gamma \sum_\gamma^*) \\ p(\sigma^2\,|\,\gamma) = \mathrm{IG}(\nu/2, \nu\lambda_\gamma/2) \\ p(\gamma) = \prod \omega_i^{\gamma_i}(1-\omega_i)^{(1-\gamma_i)} \end{gathered} \tag{7}$$

where $\sum_\gamma^*$ is a diagonal and R_γ is a correlation matrix. $\nu\lambda_\gamma/\gamma^2 \sim \chi_\nu^2$ and λ_γ is set constant in this paper which leads to reasonable results in our experience. $p(\gamma_i = 1) = 1 - p(\gamma_i = 0) = \omega_i$ is the prior probability which indicate that **x**$_i$ is large enough to select d_i in the dictionary D.

The *i*th diagonal element of $\sum_\gamma^*$ is denoted by

$$(\textstyle\sum_\gamma^{*2})_{ii} = \begin{cases} \upsilon_{0\gamma_{(i)}}^* & \text{when } \gamma_i = 0 \\ \upsilon_{1\gamma_{(i)}}^* & \text{when } \gamma_i = 1 \end{cases} \tag{8}$$

Then each component of **x** is modeled as a mixture of normals

$$p(x_i\,|\,\sigma,\gamma) = (1-\gamma_i)\mathcal{N}(0, \sigma^2\upsilon_{0\gamma_i}^*) + \gamma_i\mathcal{N}(0, \sigma^2\upsilon_{1\gamma_i}^*) \tag{9}$$

When the data supports $\gamma_i = 0$ over $\gamma_i = 1$, then x_i is probably small enough so that d_i will not be needed in the model. Then the marginal distribution of x_i given $\boldsymbol{\gamma}$ is

$$p(x_i\,|\,\gamma) = (1-\gamma_i)T(\nu, 0, \lambda_\gamma\upsilon_{0\gamma_i}^*) + \gamma_i T(\nu, 0, \lambda_\gamma\upsilon_{1\gamma_i}^*) \tag{10}$$

where $T(\nu, 0, \lambda_\gamma\upsilon_{j\gamma_i}^*)$ is the *t* distribution with *v* degrees of freedom and scale parameter $\lambda_\gamma\upsilon_{j\gamma_i}^*$.

Margin out of **x** and *σ* from $p(\mathbf{x},\sigma,\gamma\,|\,\mathbf{y}) = p(\mathbf{y}\,|\,\mathbf{x},\sigma)\bullet p(\mathbf{x}\,|\,\sigma,\gamma)p(\sigma\,|\,\gamma)p(\gamma)$, we will get $p(\gamma\,|\,\mathbf{y}) \propto g(\gamma) = |\tilde{\mathbf{D}}^{\mathrm{T}}\tilde{\mathbf{D}}|^{-1/2}|\sum_\gamma^* R_\gamma \sum_\gamma^*|^{-1/2}(\nu\lambda + S_\gamma^2)^{-(n+\nu)/2}p(\gamma)$, where

$$\begin{gathered} \tilde{\mathbf{D}} = \begin{bmatrix} \mathbf{D} \\ (\sum_\gamma^* R_\gamma \sum_\gamma^*)^{-1/2} \end{bmatrix}, \\ S_\gamma^2 = \mathbf{y}^{\mathrm{T}}\mathbf{y} - \mathbf{y}^{\mathrm{T}}\mathbf{D}(\mathbf{D}^{\mathrm{T}}\mathbf{D} + (\textstyle\sum_\gamma^* R_\gamma \sum_\gamma^*)^{-1})^{-1}\mathbf{D}^{\mathrm{T}}\mathbf{y} \end{gathered} \tag{11}$$

g (γ) in the formulation give *p*(γ |**y**) up to a normalization constant, which would require evaluation of the sum of *g*(γ) over all possible γ values.

B. Update the Dictionary

We alternate between the updates of D and x, minimizing over one while keeping the other variable fixed. We have chosen to follow the matrix-inversion free procedure of Mairal[12] for updating the dictionary. For natural image patches, the dictionary elements are usually constrained to be in

$$\mathcal{D}=\{\mathbf{D}_{d\times m}:\ \|\mathbf{d}_i\|_2\le 1\} \tag{12}$$

The update of dictionary amounts to performing

$$\begin{aligned} D_t &= \arg\min_{D\in\mathcal{D}} \frac{1}{t}\sum_{i=1}^{t}\frac{1}{2}\|\mathbf{y}_i-\mathbf{D}\mathbf{x}_i\|_2^2+\lambda\|\mathbf{x}_i\|_1 \\ &= \arg\min_{D\in\mathcal{D}} \frac{1}{t}(\frac{1}{2}\mathrm{Tr}(\mathbf{D}^{\mathrm{T}}\mathbf{D}\mathbf{A}_t)-\mathrm{Tr}(\mathbf{D}^{\mathrm{T}}\mathbf{B}_t)) \end{aligned} \tag{13}$$

where $\mathbf{A}_t=\sum_{i=1}^{t}\mathbf{x}_i\mathbf{x}_i^{\mathrm{T}}$ and $\mathbf{B}_t=\sum_{i=1}^{t}\mathbf{y}_i\mathbf{x}_i^{\mathrm{T}}$. Mairal [12] shows an online algorithm to update each column of the dictionary by

$$\mathbf{u}_i\leftarrow\frac{1}{\mathbf{A}[i,i]}(\mathbf{b}_i-\mathbf{D}\mathbf{a}_i)+\mathbf{d}_i,\ \ \mathbf{d}_i\leftarrow\frac{1}{\max(\|\mathbf{u}_i\|_2,1)}\mathbf{u}_\mathbf{i} \tag{14}$$

C. Gibbs Sampling

The Gibbs sampler is very widely applicable to a broad class of Bayesian problems has sparked a major increase in the application of Bayesian analysis. Rather than calculate all 2^p posterior probabilities in $p(\gamma\,|\mathbf{y})$, which would involve the same kind of computational burden we originally sought to avoid, Gibbs sampler is used here to generate a sequence ***x**$_1$*, ..., ***x**$_m$*, which in many cases converges rapidly. Such a sequence can be obtained quickly and efficiently, with far less effort than required to compute the entire posterior. This is because that *x* with highest probability will also appear most frequently and hence will be easiest to identify. Those *x* that appear infrequently or not at all are simply not of interest and can be disregarded.

To generate an Gibbs sequence

$$\mathbf{x}^0,\sigma^0,\gamma^0,\mathbf{x}^1,\sigma^1,\gamma^1,...,\mathbf{x}^j,\sigma^j,\gamma^j,... \tag{15}$$

which is an ergodic Markov chain. $\mathbf{x}^0,\sigma^0$ and γ^0 are initialized as **1** vectors. The subsequent values of $\mathbf{x}^j,\sigma^j,\gamma^j$ are obtained by successively simulation values according to the following iterated sampling scheme.

$$\mathbf{x}^j\sim p(\mathbf{x}^j\,|\,\mathbf{y},\sigma^{j-1},\gamma^{j-1})=\mathcal{N}_p(\mu_{\mathbf{x}^j},\Sigma_{\mathbf{x}^j}), \tag{16}$$

where

$$\begin{aligned} \mu_{\mathbf{x}^j} &= \mathbf{I}+\Sigma^*_{\gamma^{j-1}}R_{\gamma^{j-1}}\Sigma^*_{\gamma^{j-1}}\mathbf{y}^{\mathrm{T}}\mathbf{y} \\ \Sigma_{\mathbf{x}^j} &= ((\sigma^{j-1})^{-2}\mathbf{y}^{\mathrm{T}}\mathbf{y}+\Sigma^*_{\gamma^{j-1}}R_{\gamma^{j-1}}\Sigma^*_{\gamma^{j-1}})^{-1} \end{aligned} \tag{17}$$

Next, the variance σ^j is obtained by sampling from

$$\begin{aligned} \sigma^j &\sim p(\sigma^j\,|\,\mathbf{y},\mathbf{x}^j,\gamma^{j-1}) \\ &= \mathrm{IG}((n+\nu_{\gamma^{j-1}})/2,|\,y-Dx^j\,|^2+\nu_{\gamma^{j-1}}\lambda_{\gamma^{j-1}}) \end{aligned} \tag{18}$$

Finally, the vector γ^j is obtained by sampling from

$$\gamma^j\sim p(\gamma\,|\,\mathbf{y},\mathbf{x}^j,\sigma^j) \tag{19}$$

which can be calculated by equation (11).

By repeated successive sampling, the Gibbs sequence is obtained. It follows from [16] that the subsequence is a homogeneous ergodic Markov chain that converges geometrically to its unique stationary distribution. A practical consequence of this property is that as the length of the subsequence is increased, the empirical distribution of the realized values of γ will converge to the actual posterior $p(\gamma\,|\mathbf{y})$. The proposed algorithm is shown in Fig. 1.

```
Algorithm 1
input : The input signals $\mathbf{Y}=[\mathbf{y}_1,..,\mathbf{y}_n]$, each $\mathbf{y}_i\in\mathbb{R}^m$, the
    initial dictionary $\mathbf{D}_0\in\mathbb{R}^{m\times k}$;
$\mathbf{A}_0\in\mathbb{R}^{k\times k})\leftarrow 0$, $\mathbf{B}_0\in\mathbb{R}^{m\times k})\leftarrow 0$;
for $t=1$ to $n$ do
    Initial $\gamma_t^{(0)}=[\gamma_{t1}^{(0)},...,\gamma_{tk}^{(0)}]$;
    repeat
        for $i=1$ to $r$ do
            Compute $p(\gamma_i|\mathrm{y},\gamma_{j\neq i})$ from equation (11);
        end for
    until $\gamma_t$ convergence
    Compute $p(\mathrm{x}_t|\gamma_t)$ from equation (10);
    $\mathbf{A}_t\leftarrow\mathbf{A}_{t-1}+\mathrm{x}_t\mathrm{x}_t^T$;
    $\mathbf{B}_t\leftarrow\mathbf{B}_{t-1}+\mathrm{y}_t\mathrm{x}_t^T$;
    repeat
        for $i=1$ to $k$ do
            Update the $i$th column by equation (14);
        end for
    until $\mathbf{D}_t$ convergence
end for
output : Return $[\gamma_1,...,\gamma_n]$ and $\mathbf{D}_T$
```

Fig. 1. The proposed algorithm

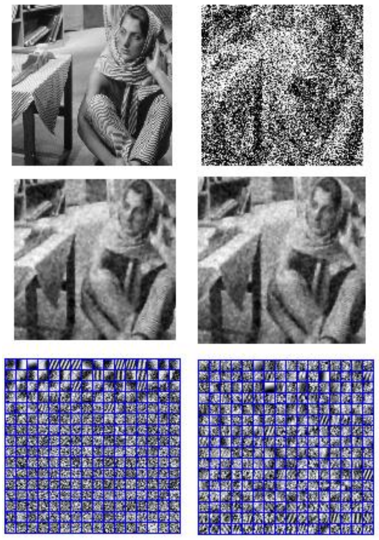

Fig. 2. The left image in the first line is the original barbara image, and the second is the noisy and incomplete barbara image with the noise standard deviation of 15 and 70% of ites pixels missing at random. The third and the fourth are the restored images by BPFA and our algorithm, and the second line are the inferred dictionaries by BPFA and our algorithm

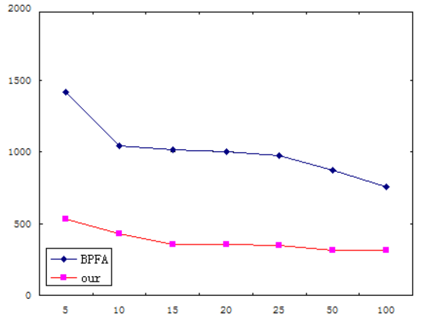

Fig. 3. The sampling speed between BPFA and our algorithm.

IV. Experiment Results

In this section we demonstrate the results achieved by applying the above methods on several test images. The tested images, as also the tested noise levels, are all the same ones as those used in the denoising experiments reported in [17], in order to enable a fair comparison.

TABLE I: Image Denoising PSNR Results. Comparing with KSVD and BPFA. The Top Are Results of KSVD, next Are Results of BPFA and the Bottom Are Results of Our Algorithm, Respectively.

σ	House	Lena	Barbara	Boats	Couple	Hill
5	39.37	38.60	38.08	37.22	37.31	37.02
	39.18	38.20	37.94	36.43	36.77	36.24
	38.56	37.68	37.05	35.76	36.16	35.72
10	35.98	35.47	34.42	33.64	33.52	33.37
	36.29	35.62	34.63	33.70	33.63	33.31
	35.33	34.71	33.86	33.49	32.96	33.01
15	34.32	33.70	32.37	31.73	31.45	31.47
	34.52	33.93	32.61	31.97	31.73	31.64
	33.93	33.42	31.77	31.91	31.23	30.88
20	33.20	32.38	30.83	30.36	30.00	30.18
	33.25	32.65	31.10	30.70	30.34	30.47
	32.79	31.75	30.58	30.21	29.91	29.58
25	32.15	31.32	29.60	29.28	28.90	29.18
	32.24	31.63	29.88	29.70	29.28	29.57
	31.75	30.81	29.09	28.83	28.75	29.04
50	27.95	27.79	25.47	25.95	25.32	26.27
	28.49	28.29	26.03	26.50	25.94	26.81
	27.64	27.60	25.12	26.14	25.85	26.38
100	23.71	24.46	21.89	22.81	22.60	23.98
	24.37	24.95	22.13	23.32	23.01	24.22
	23.12	23.42	21.65	22.97	22.86	23.75

V. Conclusion

This work has presented Bayesian nonparametric method for sparse dictionary learning, whose results have a state-of-the-art performance in image denoising. The proposed method is based on local operations and involves nonparametric sparse coding of each input signals, corresponding with the dictionary updating. The experiments have shown that a dictionary trained on patches of the noisy image itself performs very well.

In the experiments, the dictionary used was of size 64 × 256, designed to handle image patches of size 8 × 8 pixels (n = 64, k = 256). The redundant DCT dictionary was used as an initialization. The stopping rule was an average error passing a threshold, chosen empirically to be $\epsilon = 1.15\times\sigma$, where σ is assumed to be known [17]. The results shown correspond to 8 iterations of the sparse-coding and dictionary update process. Fig. 2 shows the results of the proposed algorithms for ’Barbara’ and the dictionary obtained by our algorithm for this experiment, comparing with the results of the BPFA [10].

The results shown in the pictures are unconspicuous between our algorithm and the BPFA, with the dictionaries are much different, where our result shows frequent uses of some dictionary patches.

As one representative example of the model's ability to infer the noise variance, we consider the PSNR comparing with KSVD and BPFA, which are shown in Table I. The mean inferred noises are respective standard deviations of 5, 10, 15, 20, 25, 50 and 100. Each of these noise variances were automatically inferred using exactly the same model. BPFA performs very similarly to KSVD, and our algorithm is some lower than the others. The calculation speeds of our algorithm comparing with BPFA are shown in Fig. 3, which shows apparently improvements, based on ‘barbara’ image with noise standard vary from 5 to 100. The iterations in the algorithms are set 10, each with 100 sampling sequences.

References

[1] S. Mallat and Z. Zhang, “Matching pursuits with time-frequency dictionaries,” *IEEE Trans. Signal Process*, vol. 41, no. 12.

[2] A. M. Bruckstein, D. L. Donoho, and M. Elad, “From sparse solutions of systems of equations to sparse modeling of signals and images,” *SIAM Rev*, vol. 51, no. 1.

[3] S. Mallat, “A wavelet tour of signal processing, second edition,” *Academic Press*, New York, 2nd ed, 1999.

[4] E. J. Cands and D. L. Donoho, *Curvelets-A Surprisingly Effective* Nonadaptive *Representation for Objects with Edges*, 1999.

[5] M. Zibulevsky and B. A. Pearlmutter, “Blind source separation by sparse decomposition in a signal dictionary,” *Neural Comput*, vol. 13, no. 4.

[6] K. Schnass and P. Vandergheynst, “Dictionary preconditioning for greedy algorithms,” *IEEE Trans. Signal Process*, vol. 56, no. 5.

[7] S. A. B. Sardy, and P. Tseng, “Block coordinate relaxation methods for nonparametric wavelet denoising,” *J. Comput. Graph. Stat*, vol. 9, no. 2.

[8] K. Engan, S. O. Aase, and J. H. Husoy, “Method of optimal directions for frame design,” in *Proc. Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*, vol. 5, pp. 2443–2446, 1999.

[9] M. Aharon, E. Elad, and Bruckstein, A. K. Svd, “An algorithm for designing of overcompletes dictionaries for sparse representation,” *IEEE Trans. Signal Process*, vol. 54, no. 11.

[10] M. Zhou, H. Chen, J. Paisley, L. Ren, G. Sapiro, and L. Carin, “Non-parametric bayesian dictionary learning for sparse image representations,” in *Proc. Neural Information Processing Systems*, 2009.

[11] M. Yaghoobi, T. Blumensath, and M. E. Davies, “Dictionary learning for sparse approximations with the majorization method,” *IEEE Transactions onSignal Processing*, vol. 57, no. 6.

[12] J. Mairal, F. R. Bach, J. Ponce, and G. Sapiro, “online learning for matrix factorization and sparse coding,” *CoRR*, 2009.

[13] P. J. Brown, M. Vannucci, and T. Fearn, “Multivariate bayesian variable selection and prediction,” *Journal of the Royal Statistical Society*, vol. 60, pp. 627–641, 1998.

[14] N. Sha, M. Vannucci, M. G. T. Bayesian, “variable selection in multinomial probit models to identify molecular signatures of disease stage,” *Biometrics*, vol. 6, pp. 812–819, 2004.

[15] E. I. George and R. E. M. Culloch, “Approaches for bayesian variable selection,” *Statistica Sinica*, vol. 7, pp. 339–374, 1997.

[16] J. Diebolt and C. P. Robert, “Estimation of finite mixture distributions through bayesian sampling,” *J. R. Statist. Soc B*, vol. 56, no. 2, pp. 363–375, 1994.

[17] J. Portilla, V. Strela, M. Wainwright, and E. Simoncelli, “Image denoising using scale mixtures of gaussians in the wavelet domain,” *IEEE Transactions On Image Processing*, vol. 12, no. 11, pp. 1338–1351, 2003.