COLLABORATIVE SOURCES IDENTIFICATION IN MIXED SIGNALS VIA HIERARCHICAL SPARSE MODELING

Pablo Sprechmann, Ignacio Ramirez, Pablo Cancela, and Guillermo Sapiro

University of Minnesota and Universidad de la Republica, Uruguay

ABSTRACT

A collaborative framework for detecting the different sources in mixed signals is presented in this paper. The approach is based on C-HiLasso, a convex collaborative hierarchical sparse model, and proceeds as follows. First, we build a structured dictionary for mixed signals by concatenating a set of sub-dictionaries, each one of them learned to sparsely model one of a set of possible classes. Then, the coding of the mixed signal is performed by efficiently solving a convex optimization problem that combines standard sparsity with group and collaborative sparsity. The present sources are identified by looking at the sub-dictionaries automatically selected in the coding. The collaborative filtering in C-HiLasso takes advantage of the temporal/spatial redundancy in the mixed signals, letting collections of samples collaborate in identifying the classes, while allowing individual samples to have different internal sparse representations. This collaboration is critical to further stabilize the sparse representation of signals, in particular the class/sub-dictionary selection. The internal sparsity inside the sub-dictionaries, as naturally incorporated by the hierarchical aspects of C-HiLasso, is critical to make the model consistent with the essence of the sub-dictionaries that have been trained for sparse representation of each individual class. We present applications from speaker and instrument identification and texture separation. In the case of audio signals, we use sparse modeling to describe the short-term power spectrum envelopes of harmonic sounds. The proposed pitch independent method automatically detects the number of sources on a recording.

1. INTRODUCTION AND MOTIVATION

Sparse signal modeling has been shown to lead to numerous state-of-the-art results in signal processing, in addition to being very attractive at the theoretical level. The standard model assumes that a signal can be efficiently represented by a sparse linear combination of atoms from a given or learned dictionary. The selected atoms form the *active set*, whose cardinality is significantly smaller than the size of the dictionary and the dimension of the signal. Adding structural constraints to this active set has value both at the level of representation robustness and at the level of signal interpretation; e.g., [1, 2, 3]. This leads to *group* or *structured* sparse coding, the atoms are grouped and a few groups are active at a time. An alternative way to add structure (and robustness) to the problem is to consider the simultaneous and collaborative encoding of multiple signals, requesting that they all share the same active set; e.g., [4, 5, 6].

In the (linear) source separation problem, an observed signal is assumed to be a linear superposition (mixture) of several sources, and the primary task is to estimate from it each of the unmixed sources. If the task is only to identify the active sources, the problem is called source identification. In this case, since the original sources do not need to be recovered, the modeling can be done in terms of features extracted from the original signals in a non-bijective way.

We propose to first use traditional sparse modeling tools to learn a dictionary $\mathbf{D}_i$ for each one of the $\mathcal{G}$ possible classes. Concatenating these dictionaries, $\mathbf{D} = [\mathbf{D}_1 | \mathbf{D}_2 | \dots | \mathbf{D}_\mathcal{G}]$, any mixture signal produced by that ensemble will be accurately represented as a sparse linear combination of the atoms of this larger dictionary $\mathbf{D}$. In this case one expects the resulting sparsity patterns to have a particular structure, with sub-dictionaries active following the classes present in the mixture. In addition, the time correlation in audio signals and the spatial correlation in images suggests that there is an important correlation between neighboring samples that should be exploited. Consider for example a piece of music, where a few out of many potential instruments are playing simultaneously at different times. For small time windows, we can assume that the same few instruments are playing at all instants (each instant represents a mixture sample), so that the corresponding same few groups or sub-dictionaries will be active in all samples. However, we do not expect the sound produced by each instrument to be the same at each instant, the internal activation per sub-dictionary will be sample dependent.

We propose to use the Collaborative Hierarchical Lasso (C-HiLasso) model that combines the benefits of structured and collaborative sparse coding in a hierarchical sparse model, with sparsity both at the group and single coefficient levels, and where multiple signal samples/instances collaborate in the recovery of their common active groups. However, for each signal, the active atoms within the shared active groups are particular to that signal realization. The internal sparsity inside the blocks, which is not present in standard structured/block sparsity models, is critical, since each block corresponds to a sub-dictionary $\mathbf{D}_i$ learned to efficiently represent signals of one of the possible classes in a sparse coding fashion. Not considering such in-block sparsity will then be contradictory to the essence of the dictionary model, while not considering sparsity and collaboration at the block level will contradict the fact that only a few classes are active per instance of the signal, and such classes are shared, e.g., in audio as explained above. Previously proposed sparsity models, e.g., group or collaborative sparsity, or elastic net [7], don't have these characteristics, which are critical for the problem at hand and consistent with the realistic assumptions about the signal. In [8] we provide additional details and variations of the proposed model, including a detailed comparison of C-HiLasso with recent literature, theoretical results regarding recovery guarantees, and an efficient optimization techniques that ensure convergence to the global optimum. The goal of this work is to show how the this framework can be successfully applied to several types of signals by appropriately selecting the features.

In Section 2 we briefly describe the CHiLasso model. In Section 3 we address the problem of single-channel speaker and instrument identification. The feature selection is crucial for the success and the efficiency of the model. The proposed method uses the spectral envelope as feature vectors and does not require the estimation of the fundamental frequency of the sources. In Section 4 we address the problem of texture separation and identification.

---

2. COLLABORATIVE HIERARCHICAL SPARSE CODING

We have a set of data samples $\mathbf{x}_j \in \mathbb{R}^m, j = 1, \ldots, n$, and a dictionary of p atoms in $\mathbb{R}^m$, assembled as a matrix $\mathbf{D} \in \mathbb{R}^{m \times p}$. Each sample $\mathbf{x}_j$ can be written as $\mathbf{x}_j = \mathbf{D}\mathbf{a}_j + \epsilon$, $\mathbf{a}_j \in \mathbb{R}^p$, $\epsilon \in \mathbb{R}^m$, $\|\epsilon\|_2 \ll \|\mathbf{x}_j\|_2$. The underlying assumption in sparse modeling is that the "optimal" reconstruction $\mathbf{a}_j$ has only a few nonzero elements. The convex formulation of this representation, known in the literature as *Lasso* [9], can be efficiently solved using general purpose or specialized optimization techniques. A popular variant is the unconstrained version,

$$\min_{\mathbf{a}} \frac{1}{2} \|\mathbf{x}_j - \mathbf{D}\mathbf{a}\|_2^2 + \lambda \|\mathbf{a}\|_1, \qquad (2.1)$$

where λ is an parameter value, usually found by cross-validation.

In many situations, one has prior knowledge that certain groups of atoms are simultaneously selected in the coding. Designing a model that takes this into account naturally leads to a better result. Suppose that a dictionary of p atoms is divided into $\mathcal{G}$ groups.[1] We refer to the *sub-dictionary* of atoms of $\mathbf{D}$ belonging to a group G as $\mathbf{D}_G$, and the corresponding set of linear reconstruction coefficients as $\mathbf{a}_G$. The *Group Lasso* problem is, [1],

$$\min_{\mathbf{a}} \frac{1}{2} \|\mathbf{x}_j - \mathbf{D}\mathbf{a}\|_2^2 + \sum_{G=1}^{\mathcal{G}} \|\mathbf{a}_G\|_2. \qquad (2.2)$$

Note that (2.2) reduces to (2.1) when the groups contain only one atom, and its effect on the groups of $\mathbf{a}$ is a natural generalization of Lasso: it turns on/off coefficients in groups.

In numerous applications, one expects that certain collections of samples, $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathbb{R}^{m \times n}$, share the same active components from the dictionary, that is, the indexes of the corresponding nonzero coefficients, $\mathbf{A} = [\mathbf{a}_1, \ldots, \mathbf{a}_n] \in \mathbb{R}^{p \times n}$, are the same for all the samples in the collection. Imposing such dependency in the ℓ_1 regularized regression problem gives rise to the so called *collaborative* (also called "multitask" or "simultaneous") sparse coding problem [10]. The model is given by

$$\min_{\mathbf{A}} \frac{1}{2} \|\mathbf{X} - \mathbf{D}\mathbf{A}\|_F^2 + \lambda \sum_{k=1}^{p} \left\|\mathbf{a}^k\right\|_2, \qquad (2.3)$$

where $\mathbf{a}^k \in \mathbb{R}^n$ is the k-th row of $\mathbf{A}$, that is, the vector of the n different values that the coefficient associated to the k-th atom takes for each sample $j = 1, \ldots, n$. If we extend this idea to the Group Lasso, we obtain a *collaborative Group Lasso* (C-GLasso),

$$\min_{\mathbf{A}} \frac{1}{2} \|\mathbf{X} - \mathbf{D}\mathbf{A}\|_F^2 + \lambda \sum_{G=1}^{\mathcal{G}} \left\|\mathbf{A}^G\right\|_F, \qquad (2.4)$$

where $\mathbf{A}^G$ is the sub-matrix formed by all the rows belonging to group G.

As explained in Section 1, in our proposed strategy for performing source separation, each $\mathbf{D}_G$ is trained for sparsely representing one of the possible sources in the mixture. Hence, the sparsity pattern of the coefficients of the mixture signals is hierarchical: sparsity at the group and atom levels. In this situation the C-GLasso would fail to recover the true sparsity pattern, since it promotes all the atoms in the sub-dictionary to be selected simultaneously. We also need to consider collaboration at the group level, as in (2.4), but

---

[1]For simplicity we assume that all the groups have the same size. The extension to the general case is straightforward, see [1] for details.

not at the individual atoms level. The only alternative that can handle all these requirements simultaneously is C-HiLasso (see [8] for details on how to automatically set the regularizer parameters $\lambda_{1,2}$ and also details on the optimization),

$$\min_{\mathbf{A}} \frac{1}{2} \|\mathbf{X} - \mathbf{D}\mathbf{A}\|_F^2 + \lambda_2 \sum_{G=1}^{\mathcal{G}} \left\|\mathbf{A}^G\right\|_F + \lambda_1 \sum_{j=1}^{n} \|\mathbf{a}_j\|_1. \quad (2.5)$$

The regularizer in (2.5) is a combination of the ones used in C-GLasso and Lasso and as such encourages the signals to share the same groups (classes), while the active set inside each group is signal dependent. Note that the last term in (2.5) can be replaced by a group sparsifying norm, e.g., if atoms on the sub-dictionary have some correlation. Previous approaches have only considered particular cases, such as structured coding [2], hierarchy without collaboration [11], or collaboration without hierarchy [12, 13]. The comprehensive new model, [8], is needed for the important applications presented next.

3. SOURCE IDENTIFICATION IN AUDIO

Source identification is a classic problem in audio analysis, see [14, 15] and references therein. Here is addressed with the C-HiLasso model.

3.1. Feature Selection for Speaker Identification

A challenging aspect when identifying audio sources is to obtain features that are specific to each source and at the same time invariant to changes in the fundamental frequency (tone) of the sources. In the case of speech, sounds can be divided into two main groups, *voiced* and *unvoiced* sounds. Of the two, only the former contains information useful for identifying the speaker. Since unvoiced sounds have much less energy than voiced ones, we can easily remove them from the feature extraction process, so that the identification is performed solely with the voiced sounds. To describe voiced sounds, we use their short-term power spectrum envelopes (SE) as feature vectors, which is a common choice in speaker recognition tasks [16].

The SE in human speech varies along time, producing different patterns for each phoneme. Then, a speaker does not produce a unique SP for voiced sounds, but a set that lives in a union of manifolds. Since such manifolds are well represented by sparse models, the SE characteristics are well suited for the sparse modeling framework. For C-HiLasso, the feature extraction process needs to be linear, and extracting the SE is not a linear operation. To overcome this, we propose a method inspired on the *Mel Frequency Coefficients (MFCC)* technique [16], exploiting the harmonic properties of voiced sounds.

Assume that we observe a signal $y(n)$ that is a linear mixture of c harmonic sources,

$$y(n) = \sum_{i=1}^{c} \alpha_i \sum_{k=1}^{K} E_i \left(\frac{2\pi}{f_s} k f_i \right) \cos \left(\frac{2\pi}{f_s} k f_i n + \phi_{ik} \right),$$

where α_i are the mixing coefficients, E_i and f_i are the SE and fundamental frequency of the i-th source respectively, f_s is the sampling frequency, and ϕ_{ik} is the phase of the k-th harmonic (or partial), out of K, of that source. As with MFCC, we start the analysis of an audio window by performing a short-term Fourier transform (STFT) on it. Since phase information is irrelevant for computing the spectral envelope, we only keep the magnitude of the obtained STFT. In order to amplify the frequency range of interest, we apply an emphasis
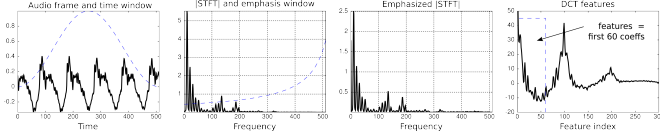
Fig. 1. *Feature extraction for audio signals. From left to right: sample analysis window, its STFT magnitude and emphasis curve, emphasized STFT magnitude, DCT and low frequency samples used as features (dotted).*

window $W(\theta)$ to the STFT magnitude, obtaining

$$|Y(\theta)| = \sum_{i=1}^{c} \alpha_i \sum_{k=1}^{K} W(\theta) E_i \left(\frac{2\pi}{f_s} k f_i \right) \delta \left(\theta - \frac{2\pi}{f_s} k f_i \right), \quad (3.6)$$

where δ is the Dirac distribution.[2] Finally, we perform a discrete cosine transform (DCT) on the emphasized STFT magnitude (which is a real function), obtaining, for each source, the convolution of a low frequency lobe that approximates the spectrum of the emphasized SE, with a sequence of spikes corresponding to the fundamental frequency and its partials. Keeping only the low-frequency coefficients of the computed DCT in (3.6) , we obtain $\sum_{i=1}^{c} \alpha_i DCT \{W(\theta) E_i(\theta)\}$, which is a linear combination of the spectrums of the emphasized SE of the present sources. We then obtained a linear relation between the sources and their spectral envelopes. The feature extraction process is summarized in Figure 1.

The audio files were re-sampled at $f_s = 16 kHz$. The features were taken based on the STFT with a frame length of 512 samples and an overlapping of 75% using a Hanning window. The emphasis in frequency is $E(f) = 1 + \alpha f$, $\alpha = 2/f_s$. After the DCT, the lowest 60 coefficients form the features.

3.2. Speaker Identification via C-HiLasso

The data for this case consists of six minutes long recordings of five different German radio speakers, two female and three male.[3] One quarter of the samples are used for training and the rest for testing.

First we want to ensure that the proposed features and the sparse modeling framework are well suited for this application. We analyzed the dataset using two very simple classifiers: k nearest neighbors, and the classifier proposed in [17]. In the latter case, following standard dictionary learning techniques, a dictionary $\mathbf{D}_i$ is learned for each class using the corresponding training samples. Each normalized testing sample, $\mathbf{x}_j$, is then assigned to the class for which the risk $R(\mathbf{D}_i, \mathbf{x}_j) = \min_{\mathbf{a}_j} \|\mathbf{x}_j - \mathbf{D}_i \mathbf{a}_j\| + \lambda \|\mathbf{a}_j\|$ is minimized (λ is learned via cross validation). The error rate obtained for each speaker by each method is shown in Figure 2(left). Although a per-sample error rate of 17.0% is not small, each sample corresponds to a time window of 32ms and we can safely assume that the same speaker will be active during several consecutive samples. Thus a simple voting scheme on top of any of these classifiers would reduce the error significantly. The collaboration, naturally included in C-HiLasso, is crucial for the identification, also when mixtures are present as detailed next.

In the analysis above we assumed the strong hypothesis that only one speaker is active at a time. We now relax this and test the performance of C-HiLasso in identifying speakers in mixture signals. Clearly, k nearest neighbors can't be used in this case. For each speaker, a sub-dictionary of 90 atoms was learned from the training dataset (we observed that the exact dictionary size is not critical to the results of the algorithm). We extracted 10 non-overlapping

frames of 15 seconds each, and encoded them using C-HiLasso. The experiment was repeated for all possible combinations of two speakers, and all the speakers talking alone. In order to quantify the performance we measured the Hamming distance between the detected active sources and the ground truth. We compared the results only against the Lasso algorithm. This is the canonical experiment since the dictionaries where trained to sparsely represent the data (C-GLasso assumes that all the atoms in the sub-dictionary are active simultaneously). C-HiLasso obtained a Hamming distance of 0.053, showing a very good capability of automatically detecting the active sources (speakers) on each frame without having the number of active sources as prior knowledge. Lasso gives a Hamming distance of 3.33. The Hamming distance when there is only one speaker in the mixture signal is 0.08 and 0.04 when there are two of them. Again here the Lasso performs worse giving 4 and 3 respectively.

In Figure 2 we show the results obtained for each frame. One could think of adding robustness to this method by evaluating overlapping time frames and doing time regularization.

3.3. Instrument Identification

Unlike the case of the human voice where the fundamental frequency can vary over a small range of values, in musical instruments it can vary considerably from one instrument to the other. For example in the experiment bellow, the fundamental frequencies vary from 80Hz (bassoon) to 1600 Hz (flute). The above proposed features represent a good description of the spectral envelope for low fundamental frequency sources. When considering sounds with high fundamental frequency, the descriptor represents a mixture of information of the envelope and fundamental frequency together. This happens because a non-adaptive linear operation can't separate them for a very wide range of fundamental frequencies. At first glance this may appear as a drawback, but in fact it becomes an advantage as it includes some information from the fundamental frequency into the descriptor, still keeping reasonable dictionary size.

We used the *Development Set for MIREX 2007 MultiF0 Estimation Tracking Task*,[4] which consists of a 52 second long musical piece played by five different wind instruments (bassoon, clarinet, flute, horn and oboe), and a set of tracks where each instrument plays individually. We used the first half of these audio tracks for training, and the rest for testing. In some passages of this piece, the instruments are arranged harmonically (forming chords), meaning that the notes one plays are partials of the fundamental note played by others. Thus, in these passages, the partials of the intervening instruments superimpose.

The experiment for this case is analogous to the one with speakers, with the testing tracks divided in frames of 3 seconds each. The results are shown in Figure 2. The average Hamming distance between the identified sources and the ground truth for C-HiLasso and Lasso was respectively 0.16 and 2.46 when only one source was active, and 0.18 and 2.76 for all combinations of two sources, for a total average of 0.17 and 2.56. Once again, this demonstrates the power of C-HiLasso in collaboratively identifying the correct instruments (classes or sub-dictionaries). The hierarchical component is critical since, while all the signals share the active instruments (and the sub-dictionaries), each time frame is different, meaning they are represented using different atoms of the detected sub-dictionaries.

---

[2] We only write the positive part of the spectrum.

[3] The dataset is available from the authors upon request.

[4] http://www.music-ir.org/mirex/wiki/.

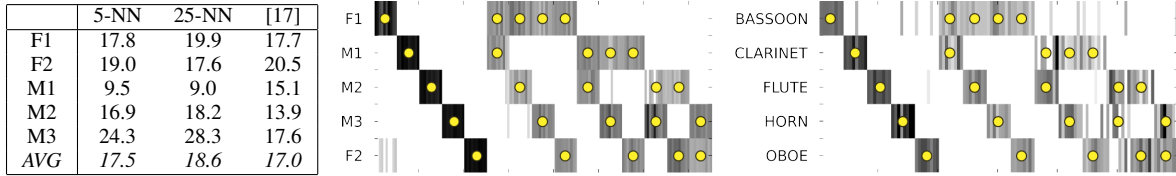|	5-NN	25-NN	[17]
F1	17.8	19.9	17.7
F2	19.0	17.6	20.5
M1	9.5	9.0	15.1
M2	16.9	18.2	13.9
M3	24.3	28.3	17.6
AVG	*17.5*	*18.6*	*17.0*

Fig. 2. *Results for audio source identification. The leftmost table shows single source detection results for a simple k-NN classifier (with $k = 5$ and 25) and the sparse model classifier presented in [17]. The center and right figures show identification results obtained with C-HiLasso when the sources are speakers (left) and wind instruments (right). Each column of the graph corresponds to the sources identified for a specific time frame, with the true ones marked by yellow dots. The vertical axis indicates the estimated activity of the different sources, where darker colors indicate higher energy. In the speaker identification case, we have 10 frames (15 seconds of audio) for each possible combination and in the instrument case, 8 frames (3 seconds). For speakers we used $(\lambda_1, \lambda_{2_0}) = (0.8, 0.008)$ and for the instruments $(\lambda_1, \lambda_{2_0}) = (0.8, 0.015)$. We observe how the number and type of classes are correctly identified.*

4. SOURCE SEPARATION IN TEXTURE IMAGES

Using sparse modeling for addressing the source separation problem in images has been addressed in [18, 19]. The methods are designed for non-collaborative separation of mixtures of two given classes. In this section, we explore the capabilities of C-HiLasso for source separation in images which are mixtures of a few out of several possible textures drawn from the Brodatz dataset, Figure 3.[5] The columns of **X** contain the pixel values of all possible square windows of 10×10 pixels in the mixture image as $m = 10^2$ dimensional vectors. The sub-dictionaries $\mathbf{D}_G$ for each texture source were obtained off-line from training samples taken from the left halves of the texture images, while the samples used in the tests were taken from the right halves. Clearly, if the image is a mixture of a number of source texture images, then every sample will also be a mixture of the same corresponding classes in the source images, and the hypothesis of C-HiLasso will hold. The experiment was repeated for all possible 28 combinations of 2 out of 8 possible source textures. In terms of detected groups, the C-HiLasso achieves near perfect performance, with an average Hamming error between the true and estimated active sets of 0.14. Lasso is clearly not designed for this task, yielding an average Hamming error of 2.8. The best average PSNR (APSNR) obtained with C-HiLasso for all combinations was $23.7dB$, which is $2dB$ larger than the $21.7dB$ obtained with Lasso. The C-GLasso obtains a Hamming error of 0.62 (three times larger than C-HiLasso), and gives a significantly lower APSNR of $19.8dB$, clearly showing that the model is not good for representing the data.

We conclude that C-HiLasso is efficient for collaboratively identifying sources in a set of mixture signals. The framework is capable of identifying sources in audio and identifying and recovering mixed sources in images, always detecting the number of sources present in the mixture.

5. REFERENCES

[1] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables," *J. Royal Stat. Society B*, vol. 68, pp. 49–67, 2006.

[2] R. Jenatton, J. Audibert, and F. Bach, "Structured variable selection with sparsity-inducing norms," Tech. Rep. arXiv:0904.3523v1, 2009.

[3] Y. C. Eldar and M. Mishali, "Robust recovery of signals from a structured union of subspaces," *IEEE TIT*, vol. 55, pp. 5302–5316, 2009.

[4] J. Tropp, "Algorithms for simultaneous sparse approximation. part II: Convex relaxation," *Signal Processing*, vol. 86, pp. 589–602, 2006.

[5] S. Cotter, B. Rao, K. Engan, and K. Kreutz-Delgado, "Sparse solutions to linear inverse problems with multiple measurement vectors," *IEEE Trans. SP*, vol. 53, no. 7, pp. 2477–2488, July 2005.

[6] J. Chen and X. Huo, "Theoretical results on sparse representations of multiple-measurement vectors," *IEEE TSP*, vol. 54, pp. 4634–4643, 2006.
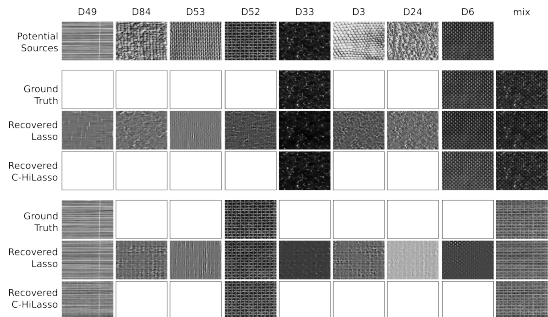
Fig. 3. *Texture separation results. We show only two sample combinations out of the 28 possible combinations of 2 active sources out of 8 (inactive ones are shown in white). Standard Lasso, which is not designed for this task, recovers residual sources which were not present in the original mixtures. The quality of the recovered active textures, although not evident from the thumbnails above, is also 2dB better in both examples, and 1dB on average for all combinations, compared to what we can obtain with Lasso.*

[7] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *J. of the Royal Stat. Soc. B*, vol. 67, pp. 301–320, 2005.

[8] P. Sprechmann, I. Ramírez, G. Sapiro, and Y. C. Eldar, "C-hilasso: A collaborative hierarchical sparse modeling framework," *http://arxiv.org/abs/1003.0400*, June 2010.

[9] R. Tibshirani, "Regression shrinkage and selection via the LASSO," *J. Royal Stat. Society B*, vol. 58, no. 1, pp. 267–288, 1996.

[10] B. Turlach, W. Venables, and S. Wright, "Simultaneous variable selection," *Technometrics*, vol. 27, pp. 349–363, 2004.

[11] J. Friedman, T. Hastie, and R. Tibshirani, "A note on the group Lasso and a sparse group Lasso," preprint, 2010.

[12] J. Peng et al., "Regularized multivariate regression for identifying master predictors with application to integrative genomics study of breast cancer," *Annals of Applied Statistics*, vol. 4, no. 1, pp. 53–77, 2010.

[13] P. Boufounos, G. Kutyniok, and H. Rauhut, "Sparse recovery from combined fusion frame measurements," *arXiv:0912.4988v1*, 2010.

[14] M. N. Schmidt and R. K. Olsson, "Single-channel speech separation using sparse non-negative matrix factorization," in *Proc. of Interspeech*, 2006.

[15] P. Mowlaee, R. Saeidi, Z. H. Tan, M. G. Christensen, P. Fränti, and S. H. Jensen, "Joint single-channel speech separation and speaker identification," in *Proc. ICASSP*, 2010.

[16] L. Rabiner and B.-H. Juang, *Fundamentals of Speech Recognition*, Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1993.

[17] P. Sprechmann and G. Sapiro, "Dictionary learning and sparse coding for unsupervised clustering," in *Proc. ICASSP*, Mar. 2010.

[18] N. Shoham and M. Elad, "Alternating KSVD-denoising for texture separation," in *IEEE 25-th Convention, Israel*, 2008.

[19] J. Bobin, J. Starck, J. Fadili, and Y. Moudden, "Sparsity and morphological diversity in blind source separation," *IEEE Trans. IP*, vol. 16, no. 11, pp. 2662–2674, 2007.

[5] http://www.ux.uis.no/~tranden/brodatz.html