MODIFIED LASSO SCREENING FOR AUDIO WORD-BASED MUSIC CLASSIFICATION USING LARGE-SCALE DICTIONARY

Ping-Keng Jao, Chin-Chia Michael Yeh and Yi-Hsuan Yang

Research Center for Information Technology Innovation, Academia Sinica, Taiwan

ABSTRACT

Representing music information using audio codewords has led to state-of-the-art performance on various music classification benchmarks. Comparing to conventional audio descriptors, audio words offer greater flexibility in capturing the nuance of music signals, in that each codeword can be viewed as a quantization of the music universe and that the quantization goes finer as the size of the dictionary (i.e., audio codebook) increases. In practice, however, the high computational cost of codeword assignment might discourage the use of a large dictionary. This paper presents two modifications of a LASSO screening technique developed in the compressive sensing field to speed up the codeword assignment process. The first modification exploits the repetitive nature of music signals, whereas the second one relaxes a screening constraint that is specific to reconstruction but not for classification. Our experiments show that the proposed method enables the use of a dictionary of 10,000 codewords with runtime close to the case of using a dictionary of 1,000 codewords. Moreover, using the larger dictionary significantly improves the mean average precision (MAP) from 0.219 to 0.246 for tagging thousands of tracks with 147 possible genre tags.

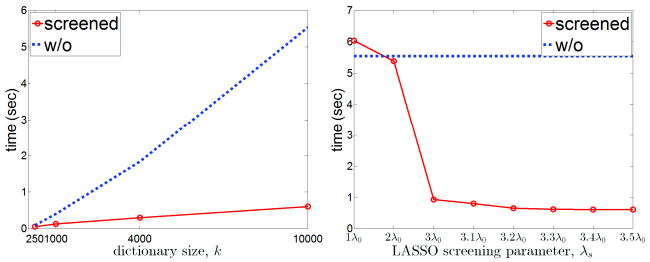
Index Terms— Sparse coding, feature learning, LASSO screening, music information retrieval, genre classification

1. INTRODUCTION

Feature learning is a burgeoning research topic in the broad signal processing and machine learning communities [1]. For music information retrieval (MIR), representing music information as a term-document structure comprising of elementary audio codewords has been found competitive or even superior to conventional, hand-crafted audio features [2–5].

Given a dictionary (audio codebook) $\mathbf{D} \in \mathbb{R}^{m\times k}$, which is a finite collection of k codewords $\mathbf{d}_j \in \mathbb{R}^m$, $j = 1, \ldots, k$, an input acoustic feature vector $\mathbf{x} \in \mathbb{R}^m$ can be replaced by a linear combination of the codewords, leading to the so-called *audio word* (AW), or *bag-of-frames* representation [5–10].

This work was supported in part by the National Science Council of Taiwan under Grant NSC 102-2221-E-001-004-MY3 and the Academia Sinica Career Development Award 102-CDA-M09.



(a) λ_s=3.5λ_0 (b) $k = 10,000$

Fig. 1. Time cost of performing LARS-LASSO sparse coding [12] with (red line with circles) or without (blue dashed line) the proposed modified LASSO screening (a) for fixed $\lambda_s = 3.5\lambda_0$ and varying dictionary size k (from 250 to 10,000) and (b) fixed k=10,000 and varying λ_s (cf. Section 4.1).

Specifically, the AW representation is a k-dimensional vector $\alpha \in \mathbb{R}^k$ computed by a *codeword assignment* function $\alpha = f(\mathbf{D}, \mathbf{x})$. For the case of $k \gg m$, it has been shown that the following *sparse coding* (SC) formulation empirically performs better than competing methods such as vector quantization (VQ) [6, 11].

$$\alpha^* = f_{\text{SC}}(\mathbf{D}, \mathbf{x}, \lambda_c) = \underset{\alpha}{\operatorname{argmin}} \frac{1}{2}\|\mathbf{x} - \mathbf{D}\alpha\|_2^2 + \lambda_c\|\alpha\|_1 \,. \quad (1)$$

This formulation leads to codes that are sparse (i.e., small $\|\alpha\|_1$) but are sufficient to reconstruct or to interpret the input signal (i.e., small $\|\mathbf{x} - \mathbf{D}\alpha\|_2^2$); the parameter λ_c controls the balance between these two terms. This problem can be efficiently solved by for example the least angle regression (LARS)-LASSO algorithm [12] given a dictionary of moderate size. Because the resulting AW representation α^* is high-dimensional (i.e., large k) but sparse, using linear support vector machine (SVM) [13] for training audio classifiers has been found effective and efficient [9, 11].

The dictionary $\mathbf{D}$ is usually generated from a possibly unlabeled music collection $\mathcal{X}$ in off-line [4, 5]. As each codeword $\mathbf{d}_j$ represents a sample of the acoustic space $\mathcal{U} \subseteq \mathbb{R}^m$, theoretically $\mathbf{D}$ can represent every nuance of music signals if 1) $\mathcal{X}$ is representative of the music universe $\mathcal{U}$ and 2) sufficiently large number of codewords are drawn from $\mathcal{X}$. The first condition can be approached by generating the dictionary from for example the million song dataset [2, 14, 15], a

publicly-available collection of audio features for a million contemporary popular music tracks. To meet the second condition, a straightforward approach is to increase the vocabulary size k by drawing more samples from $\mathcal{X}$. In practice, however, the approach is not feasible as SC using large vocabulary can be exceedingly time consuming [16].

The dotted line in Fig. 1(a) shows the time cost of encoding the 12-bin chromagram of a song (i.e., $m = 12$) using LARS-LASSO on a regular PC. It can be found that the runtime grows linearly along with k, and that it costs almost 6 seconds for the case of $k = 10,000$. Better efficiency is desirable for large-scale or mobile applications.

In this paper, we show that we can reduce the runtime remarkably (as shown by the solid line in Fig. 1(a)) with a modified LASSO screening technique without compromising the effectiveness of the AW representation. This amounts to eight-fold speed increase for $k = 10,000$. *LASSO screening* is a novel technique proposed in the compressive sensing field to filter redundant codewords for a given input without affecting the existence of an optimal solution [16–18]. We show that the existing screening techniques are less applicable to music signal processing and propose two modifications. Our evaluation shows that the proposed modifications effectively speed up SC and improve the accuracy of an AW-based music genre classification system.

Accelerating AW-based music classification has been studied recently. For example, Yeh *et al.* [19] employed dimension reduction techniques to reduce m and multi-frame or temporal sampling techniques to reduce the number of frames to be encoded for an audio signal. Yang [20] investigated the use of randomized clustering forest (RCF) to replace SC. On the other hand, for deep learning techniques distributed architecture is usually employed [2, 4, 21]. The present work differs from the prior arts in the following aspects. First, we focus on a system that can be built on a single machine. Second, the proposed method is designed specifically for SC, which usually leads to more effective AW representation than competing methods such as RCF or VQ [11, 20]. Lastly, this work represents an early attempt that applies screening to improve the efficiency of SC with a large-scale dictionary. In contrast, the dictionary size k used in previous work is usually less than 2,000 [4, 5, 10, 15].

2. MODIFIED LASSO SCREENING

2.1. LASSO screening

The principle idea of LASSO screening is to remove codewords in $\mathbf{D}$ that are unlikely to be useful for encoding a specific signal $\mathbf{x}$, before actually performing SC [16–18]. From the dual form of Eq. 1,

$$\theta^* = \arg\max_{\theta} \ \frac{1}{2}\|\mathbf{x}\|_2^2 - \frac{\lambda_c^2}{2}\ \|\theta - \frac{\mathbf{x}}{\lambda_c}\|_2^2\,, \qquad \text{s.t.} \quad |\theta^T \mathbf{d}_j| \le 1 \quad \forall j = 1, \ldots, k\,, \tag{2}$$

Algorithm 1 Modified LASSO screening based SC

Input: λ_c; λ_s; $\mathbf{X} \in \mathbb{R}^{m\times n} := [\mathbf{x}_1, \ldots, \mathbf{x}_n]$;
Output: $\mathbf{A} \in \mathbb{R}^{k\times n} := [\alpha_1, \ldots, \alpha_n]$;

```
Ω ← ∅;
for t = 1 to n do
    w_t ← screening(D, x_t, λ_s);
    Ω ← Ω ∪ w_t;
end for
for t = 1 to n do
    α_t ← f_SC(Ω, x_t, λ_c);
end for
```

rules for irrelevant codeword removal can be discovered by exploring the relation between the dual variable θ and the primal variable α. Given that the input $\mathbf{x}$ and each $\mathbf{d}_j$ are normalized to unit energy (i.e., $\mathbf{x}^T\mathbf{x} = 1$), Xiang *et al.* deduced a method called the "sphere test–3" to select codewords from $\mathbf{D}$ for encoding $\mathbf{x}$ [16, 17]. Specifically, the test calculates the $\mathbf{x}^T\mathbf{d}_j$, the correlation between the input and a codeword, and removes $\mathbf{d}_j$ if the following inequality holds,

$$|\mathbf{x}^T\mathbf{d}_j - (\lambda_{\max} - \lambda_s)\mathbf{d}_*^T\mathbf{d}_j| < \lambda_s \left(1 - \sqrt{\lambda_{\max}^{-2} - 1}\left(\frac{\lambda_{\max}}{\lambda_s} - 1\right)\right)\,, \tag{3}$$

where $\lambda_{\max} \equiv \max_j |\mathbf{x}^T\mathbf{d}_j|$ is the largest absolute correlation between the input and the codewords, $\mathbf{d}_* \in \{\pm\mathbf{d}_j\}_{j=1}^k$ is the codeword that leads to $\lambda_{\max}$, and $\lambda_s \le \lambda_{\max}$ is a parameter. After applying the above test to all the codewords, we obtain a dictionary subset $\mathbf{w} \in \mathbb{R}^{m\times k'}$, $k' \le k$, for $\mathbf{x}$,

$$\mathbf{w} = \text{screening}(\mathbf{D}, \mathbf{x}, \lambda_s)\,. \tag{4}$$

It has been proved that when $\lambda_s = \lambda_c$ the screening function will not affect the optimal solution of SC [16].

In practice, different $\mathbf{w}$ are used for different inputs. To make the dimension of the coding result consistent, we fill zeros to the entries corresponding to removed codewords to increase the dimension from k' to k.

2.2. Modifications

For a song with n frames, we have to perform SC for each frame $\mathbf{x}_t$ individually. In consequence, we would also have to use LASSO screening n times to compute $\mathbf{w}_t$ for each $\mathbf{x}_t$. However, it might not be necessary to encode each $\mathbf{x}_t$ with possibly a nearly totally different dictionary, given the repetition nature of music (i.e., the frames might be similar with one another). In addition, using different subsets of $\mathbf{D}$ to encode a song may suffer from either extra memory transfer or memory discontinuity problems. Our pilot study shows that the extra cost in memory transfer can be an important issue — we compared runtime of SC without screening and SC with any of the existing screening techniques [16, 17] and found no acceleration but severe de-acceleration in many cases.

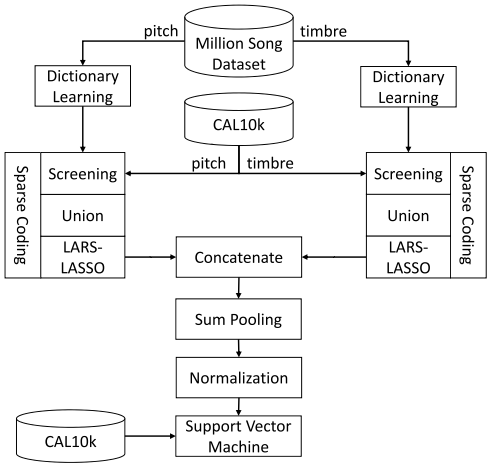


Fig. 2. System diagram of the proposed AW-based system.

To address this problem, we propose to use only one *song-level dictionary* $\mathbf{\Omega}$ for each song, using the union of $\mathbf{w}_t$ as the dictionary. As the pseudo code in Algorithm 1 shows, two passes through all the n frames are needed. The first pass accumulates the relevant codewords to obtain the song-level union $\mathbf{\Omega} \subseteq \mathbf{D}$. The second pass uses $\mathbf{\Omega}$ to encode each $\mathbf{x}_t$ by SC. No memory transfer is needed as the dictionary is shared.

The second modification we made is to relax the constraint $\lambda_s = \lambda_c$. Such a constraint is needed for minimizing the reconstruction error $\|\mathbf{x} - \mathbf{D}\alpha\|_2^2$, which is essential for compressive sensing applications. However, for classification applications, we are more concerned with the ability of the coding result α in representing the acoustic quality of $\mathbf{x}$, so perfect reconstruction might not be needed. Moreover, the constraint would lead to low rejection ratio (i.e., only a few codewords would be removed) [18] because the value of λ_c is usually small (e.g., the classic $\lambda_0 = 1/\sqrt{\min(m,k)}$ [22]) in SC. From the Fig. 2(c) of [16] we also see that the speed up brought up by screening is limited when λ_s is small.

As preserving the existence of the optimal solution might not be important for classification, we propose to use different values for λ_s and λ_c, which effectively breaks the connection of the primal and dual forms of Eq. 1. In this way, higher rejection ratio (and thus more efficient SC process) can be obtained by using a larger value for λ_s.

3. EXPERIMENT SETUP & SYSTEM OVERVIEW

Two datasets were utilized in the performance study: the million song dataset (MSD) [14] for dictionary generation and the CAL10k dataset [23] for evaluating the accuracy of AW-based music classification. The CAL10k dataset contains the annotation over 147 genre tags for 10,870 songs made by expert musicologists from Pandora (http://www.pandora.com).

For the low-level feature representation $\mathbf{x}$, we used the 12-D timbre descriptors (ENT) and 12-D pitch descriptors (ENP) computed by the EchoNest API (http://developer.echonest.com). ENT describes the timbre characteristics of the magnitude spectrogram, whereas ENP is chroma-like [23]. The features of a song are not in the frame-level but in the segment-level computed by the EchoNest API, where each segment corresponds to an acoustically homogenous fragment with multiple frames. In consequence, LARS-LASSO was employed to encode the segment-level features.

We respectively generated two dictionaries for ENT and ENP by using the "exemplar-based" approach, randomly selecting one segment-level feature vector from k random songs of MSD and considering the union of the vectors as a dictionary of size k. The same k is used for timbre and pitch. This approach bypasses the need of using dictionary learning algorithms [1, 22, 24], which is not the focus of this paper. Moreover, exemplar-based dictionary has been found effective for music genre classification [20, 25].

For classification, we computed $\alpha_t^{\text{ENT}} \in \mathbb{R}^k$ and $\alpha_t^{\text{ENP}} \in \mathbb{R}^k$ for each segment and used the concatenation $\hat{\alpha}_t = [\alpha_t^{\text{ENT}}; \alpha_t^{\text{ENP}}] \in \mathbb{R}^{2k}$ as the final segment-level feature. To obtain the song-level representation, which is used as input to SVM, we performed sum-pooling over a song by $\alpha^{\text{pool}} = \sum_{t=1}^{n} \hat{\alpha}_t$. In addition, L_1 and square-root power normalization were applied to α^{pool} for better performance [5].

Fig. 2 depicts the diagram of the proposed system. In our implementation, we cast the multilabel tagging problem into binary classification problems and used LIBLINEAR [13] for classifier training and prediction. The SVM parameter C was fixed to 8 throughout the experiments.

Table 1. The rejection ratio of codeword screening with or without (w/o) song-level union and the resulting speed up comparing to non-screening sparse coding

k	λ_s	rejection ratio (%)		speed up factor	
		w/o union	union	w/o union	union
10,000	λ_0	38.79	0.00	0.03x	0.88x
	$2\lambda_0$	88.38	1.11	0.19x	0.83x
	$3\lambda_0$	99.63	40.14	2.60x	1.63x
	$3.1\lambda_0$	99.80	51.90	3.21x	2.02x
	$3.2\lambda_0$	99.91	65.73	3.61x	2.99x
	$3.3\lambda_0$	99.96	80.29	3.91x	4.82x
	$3.4\lambda_0$	99.99	91.06	4.00x	7.27x
	$3.5\lambda_0$	99.9999	92.26	4.06x	8.03x

4. RESULT

4.1. Efficiency of SC with modified LASSO screening

We first present a preliminary efficiency evaluation of SC with or without screening, using a fixed $\lambda_c = \lambda_0 = 1/\sqrt{12}$ [22] and varying λ_s and k, for encoding a song randomly selected from CAL10k. On average, each song in CAL10k has 840 segments, so we have to perform nearly a thousand SC (and

Table 2. Accuracy of genre tagging for SC features without (w/o) or with the proposed screening versus baseline methods

method	dictionary size	rejection ratio	effective size	AUC		MAP		10-Prec		R-Prec	
				w/o	with	w/o	with	w/o	with	w/o	with
linear SVM using the screened SC (λ_s=3.5λ_0)	24	11%	21	0.856	0.853	0.179	0.176	0.232	0.228	0.193	0.189
	48	21%	38	0.859	0.852	0.205	0.195	0.268	0.254	0.219	0.209
	96	33%	65	0.864	0.855	0.218	0.199	0.281	0.265	0.228	0.216
	128	39%	78	0.865	0.852	0.220	0.201	0.285	0.266	0.234	0.213
	250	51%	123	0.866	0.851	0.227	0.204	0.293	0.262	0.243	0.220
	1000	75%	250	0.865	0.852	0.235	0.219	0.299	0.291	0.247	0.236
	4000	90%	403	0.868	0.851	0.256	0.242	0.326	0.313	0.270	0.256
	10000	95%	505	0.868	0.853	0.265	0.246	0.337	0.319	0.276	0.260
random guess [23]				0.501	—	0.018	—	0.015	—	0.015	—
linear SVM using ENT and ENP				0.757	—	0.070	—	0.115	—	0.092	—
'fastest' setting using SC [19]				0.789	—	0.122	—	0.164	—	0.140	—
'most accurate' setting using SC [19]				0.854	—	0.202	—	0.253	—	0.214	—
'GMM' using 'ENT+Δ' [23]				0.887	—	0.211	—	0.266	—	0.224	—

screening) for a song. Fig. 1(a) clearly shows that the proposed LASSO screening (with λ_s=3.5λ_0) greatly reduces the runtime for SC; the speed up gets more pronounced as k increases. This result shows that screening is important for a large-scale dictionary. On the other hand, Fig. 1(b) shows that LASSO screening is effective only when sufficiently large λ_s is used; if the original screening technique is applied (i.e., setting $\lambda_s = \lambda_0$), there is no speed up.

Table 1 shows how song-level fusion affects the efficiency. The following three important observations are made. First, the conventional LASSO screening (i.e., $\lambda_s = \lambda_0$ and without union) does not speed up but reduce the speed by a factor of 0.03, comparing to the runtime of SC without any screening. Second, higher λ_s leads to more speed up, as expected. Third, song-level union boosts the efficiency further. For $\lambda_s = 3.5\lambda_0$ and $k = 10,000$, the speed up attains 8.03x.

However, better efficiency in SC might come with lower accuracy of the resulting AW representation. To study this, we report the accuracy for genre tagging based on the proposed SC features, using the extreme case $\lambda_s = 3.5\lambda_0$.[1]

4.2. Accuracy of AW-based genre tagging

The accuracy of genre tagging for CAL10k was computed by averaging the result of the five train/test splits specified by Tingle *et al.* [23]. The following four measures are computed: the area under the receiver operating characteristic curve (AUC), mean average precision (MAP), 10-precision (10-Prec) and R-precision (R-Prec), according to [23].

The upper half of Table 2 shows the performance of SC features without (w/o) and with the modified LASSO screening as a function of dictionary size k. It can be found that better accuracy is obtained as k increases, as expected. The performance improvement is pronounced for MAP, 10-Prec and R-Prec. This result supports our interest in using a large-scale dictionary. From Table 2, we also see that screening does not affect the accuracy much. Although slightly better accuracy can be obtained without screening, the modified LASSO screening does not trade much accuracy for efficiency.

The lower half of Table 2 shows the result of two baseline methods (a random baseline and linear SVM using the concatenated ENT and ENP features without SC) and three existing methods [19,23] — the first two existing methods [19] applied SC on the raw magnitude spectrum (instead of EchoNest features) and employed dimension reduction and multi-frame representation, whereas the last one [23] used Gaussian Mixture Model (GMM) to model the 36-D 'ENT+Δ' feature with first- and second-order temporal derivatives. The dictionary size k for SC was set to 1,024 in [19] due to the computational cost of SC. We can find that the proposed SC features are competitive comparing with these prior arts. Remarkably higher accuracy in 10-Prec and R-Prec is obtained.[2]

5. CONCLUSIONS

In this paper, we have presented an efficient and effective sparse coding engine for MIR. It adds two modifications to a LASSO screening method to enable the use of a large dictionary, which in turn improves the ability of the codewords in representing the fine details of music signals. The first modification (song-level union) exploits the repetitive nature of music to reduce memory overhead, and the second one relaxes a constraint that is not required for classification. A large-scale dictionary generated from the million song dataset leads to state-of-the-art result in a genre tagging benchmark, the CAL10k dataset. In addition, our study provides empirical evidence showing that a larger dictionary is indeed favorable.

[1] In effect this sets $\lambda_s = 1$ as $3.5/\sqrt{12}$ exceeds 1, the maximal possible value for $\lambda_{\max}$. A prerequisite of Eq. 3 is $\lambda_s \leq \lambda_{\max}$ [16].

[2] Actually the comparison is not exactly fair as slightly different subsets of CAL10k were employed in this work, [19] and [23]. For example, only 7,799 songs (whose audio previews are available online) were considered in [19].

6. REFERENCES

[1] I. Tošić and P. Frossard, "Dictionary learning," *IEEE Signal Processing Magazine*, vol. 28, no. 2, pp. 27–38, 2011.

[2] S. Dieleman, P. Brakel, and B. Schrauwen, "Audio-based music classification with a pretrained convolutional network," in *Proc. Int. Society of Music Information Retrieval*, 2011, pp. 669–674.

[3] E. J. Humphrey, J. P. Bello, and Y. LeCun, "Deep architectures and automatic feature learning in music informatics," in *Proc. Int. Society of Music Information Retrieval*, 2012, pp. 403–408.

[4] J. Nam, J. Herrera, M. Slaney, and J. Smith, "Learning sparse feature representations for music annotation and retrieval," in *Proc. Int. Society of Music Information Retrieval*, 2012, pp. 565–560.

[5] L. Su, C.-C. M. Yeh, J.-Y. Liu, J.-C. Wang, and Y.-H. Yang, "A systematic evaluation of the bag-of-frames representation for music information retrieval," *IEEE Trans. Multimedia*, 2014.

[6] E. C. Smith and M. S. Lewicki, "Efficient auditory coding," *Nature*, vol. 439, no. 7079, pp. 978–982, 2006.

[7] M. D. Plumbley, T. Blumensath, L. Daudet, R. Gribonval, and M. E. Davies, "Sparse representations in audio and music: from coding to source separation," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 995–1005, 2010.

[8] J.-C. Wang, H.-S. Lee, H.-M. Wang, and S.-K. Jeng, "Learning the similarity of audio music in bag-of-frames representation from tagged music data," in *Proc. Int. Society of Music Information Retrieval*, 2011, pp. 85–90.

[9] B. Zhao, F.-F. Li, and E. P. Xing, "Online detection of unusual events in videos via dynamic sparse coding," in *IEEE Int. Conf. Computer Vision & Pattern Recognition*, 2011.

[10] P.-K. Jao, L. Su, and Y.-H. Yang, "Analyzing the dictionary properties and sparsity constraints for a dictionary-based music genre classification system," in *Proc. Asia Pacific Signal and Information Processing Association Annual Summit and Conf.*, 2013.

[11] C.-C. M. Yeh and Y.-H. Yang, "Supervised dictionary learning for music genre classification," in *ACM Int. Conf. Multimedia Retrieval*, 2012.

[12] B. Efron, T. Hastie, L. Johnstone, and R. Tibshirani, "Least angle regression," *Annals of Statistics*, vol. 32, pp. 407–499, 2004.

[13] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "LIBLINEAR: A library for large linear classification," *J. Machine Learning Research*, vol. 9, pp. 1871–1874, 2008.

[14] T. Bertin-Mahieux, D. P. W. Ellis, B. Whitman, and P. Lamere, "The million song dataset," in *Proc. Int. Society of Music Information Retrieval*, 2011.

[15] M. Henaff, K. Jarrett, K. Kavukcuoglu, and Y. LeCun, "Unsupervised learning of sparse features for scalable audio classification," in *Proc. Int. Society of Music Information Retrieval*, 2011, pp. 681–686.

[16] Z. J. Xiang, H. Xu, and P. J. Ramadge, "Learning sparse representations of high dimensional data on large scale dictionaries," in *Proc. Advances in Neural Information Processing Systems*, 2011, pp. 900–908.

[17] Z. J. Xiang and P. J. Ramadge, "Fast LASSO screening tests based on correlations," in *IEEE. Int. Conf. Acoustics, Speech & Signal Processing*, 2012, pp. 2137–2140.

[18] Y. Wang, Z. J. Xiang, and P. J. Ramadge, "LASSO screening with a small regularization parameter," in *IEEE. Int. Conf. Acoustics, Speech & Signal Processing*, 2013, pp. 3342–3346.

[19] C.-C. M. Yeh and Y.-H. Yang, "Towards a more efficient sparse coding based audio-word feature extraction system," in *Proc. Asia Pacific Signal and Information Processing Association Annual Summit and Conf.*, 2013.

[20] Y.-H. Yang, "Towards real-time music auto-tagging using sparse features," in *IEEE Int. Conf. Multimedia and Expo.*, 2013.

[21] H. Lee, Y. Largman, P. Pham, and A. Y. Ng, "Unsupervised feature learning for audio classification using convolutional deep belief networks," in *Advances in Neural Information Processing Systems*, 2009, pp. 1096–1104.

[22] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online dictionary learning for sparse coding," in *Int. Conf. Machine Learning*, 2009, pp. 689–696.

[23] D. Tingle, Y. E. Kim, and D. Turnbull, "Exploring automatic music annotation with acoustically-objective tags," in *Proc. ACM Int. Conf. Multimedia Information Retrieval*, 2010, pp. 55–62.

[24] K. Skretting and K. Engan, "Recursive least squares dictionary learning algorithm," *IEEE Trans. Signal Processing*, vol. 58, no. 4, pp. 2121–2130, 2010.

[25] Y. Panagakis, C. Kotropoulos, and G. R. Arce, "Music genre classification using locality preserving non-negative tensor factorization and sparse representations," in *Proc. Int. Society of Music Information Retrieval*, 2009, pp. 249–254.