AUTOMATIC MUSIC TAGGING BY LOW-RANK REPRESENTATION

Yannis Panagakis and Constantine Kotropoulos

Department of Informatics
Aristotle University of Thessaloniki
Box 451, Thessaloniki 54124, GREECE
email: {panagakis,costas}@aiia.csd.auth.gr

ABSTRACT

A novel multi-label annotation method is proposed and applied to music tagging. Each music recording is represented by its auditory temporal modulations (ATMs). Given a set of training music recordings represented by the tag-music recording matrix having zero-one (indicator) vectors of the tags associated with each recording in its columns along with the matrix of the ATM representations in its columns, a low-rank weight matrix is sought, such that the tag-music recording matrix is expressed as the product of the weight matrix and the matrix of the ATM representations plus an error matrix. Clearly, such a weight matrix captures the relationships between the labels (i.e., tags) and the audio features. It can be derived by solving a convex nuclear norm minimization problem, if the tag-music recording matrix and the matrix of the ATM representations are assumed to be jointly low-rank. Having found the weight matrix, the annotation vector for labeling any test music recording can be obtained by multiplying the weight matrix with its ATM representation. The just outlined method is referred to as *low-rank representation based multi-label annotation (LRRMA)*. The LRRMA outperforms the state-of-the-art auto-tagging systems, when applied to the CAL500 dataset in a 5-fold cross-validation experimental protocol.

Index Terms— Automatic Music Tagging, Multi-label Classification, Low-Rank Representation, Nuclear Norm Minimization.

1. INTRODUCTION

The development of powerful large-scale semantic music discovery engines is of paramount importance in Web 2.0, since such engines allow efficient music browsing and recommendation [1]. Current music oriented recommendation services, such as *last.fm*[1] and *Pandora*[2] employ *social tags* for semantic music representation. Social tags are text-based labels, provided by other users, that encode semantic information related to music (e.g., instrumentation, genres, emotions). The major drawbacks of the aforementioned services are: 1) a newly added music recording must be tagged manually, before it can be retrieved [2], which is a time consuming and expensive process and 2) unpopular music recordings may not be tagged at all [1]. Content-based automatic tagging of music could be exploited to mitigate the just mentioned drawbacks and complement the set of tags provided by humans.

A considerable volume of research in *automatic music tagging* (also known as *automatic multi-label music annotation* or *autotagging*) has been accumulated [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]. The majority of such autotagging systems consists of two stages, namely a music representation stage and a machine learning one. In the first stage, the music signal is represented by either low- or mid-level audio features. Music is frequently modeled by the long-term statistical distribution of short-time features. Such features include timbral texture features, rhythmic features, pitch content, or their combinations yielding a *bag-of-features (BOF)* representation [1, 2, 3, 4, 5, 6, 7]. This BOF representation ignores the temporal structure of music and thus fails to capture music dynamics (e.g., tempo and beat) or temporal structures (e.g., arpeggios), which are important characteristics of music. On the contrary, mid-level music representations, such as the auditory temporal modulations (ATMs) [11] capture both the timbral and the temporal structure of music [8, 9]. At the machine learning stage, the automatic music tagging is treated as multi-label classification problem and a variety of algorithms has been exploited in order to associate the tags with the audio features. For instance, music tag prediction may be treated as a set of binary classification problems, where standard classifiers, such as the support vector machines [5, 7] or ada-boost [4] can be applied. Furthermore, probabilistic autotagging systems have been proposed, attempting to infer the correlations or joint probabilities between the tags and the audio features [2, 10, 6]. Recently, autotagging systems have been proposed based on subspace learning [8, 9]. That is, linear and multilinear subspace learning algorithms efficiently harness the multi-label information for feature extraction, while the multiple labels of training music recordings are propagated to the test music recordings, assuming either sparse or dense representations.

In this paper, a novel automatic music tagging method is proposed. Given a set of training music recordings represented by the tag-music recording matrix having zero-one (indicator) vectors of the tags associated with each recording in its columns along with the matrix of the ATM representations in its columns, a low-rank weight matrix is sought, such that the tag-music recording matrix is expressed as the product of the weight matrix and the matrix of the ATM representations plus an error matrix. The weight matrix can be derived by solving a convex nuclear norm minimization problem, if the tag-music recording matrix and the matrix of the ATM representations are assumed to be jointly low-rank. In particular, an algorithm for nuclear norm minimization is developed by employing the alternating direction augmented Lagrange multiplier method [12, 13]. The derived low-rank weight matrix is expected to capture the relationships between the feature space defined by the ATMs and the semantic space defined by the labels. Accordingly, the annotation vector for labeling any test music recording can be obtained by multiplying the weight matrix with its ATM representation (or any audio feature vector in general). The proposed method is referred to as *low-rank representation-based multi-label annotation (LRRMA)*. The performance of the LRRMA is assessed by conducting exper-

[1] http://www.last.fm/
[2] http://www.pandora.com/

iments on the CAL500 dataset [2]. The reported experimental results demonstrate the superiority of the proposed framework over the state-of-the-art autotagging systems on the CAL500 dataset, when 5-fold cross-validation is applied.

The paper is organized as follows. In Section 2, basic notation conventions are introduced. The multi-label annotation framework, that is based on the low-rank representations is detailed in Section 3. Experimental results are demonstrated in Section 4. Conclusions are drawn and future research direction are indicated in Section 5.

2. NOTATIONS

Throughout the paper, matrices are denoted by uppercase boldface letters (e.g., $\mathbf{X}, \mathbf{Y}$), vectors are denoted by lowercase boldface letters (e.g., $\mathbf{x}$), and scalars appear as lowercase letters (e.g., i, μ, ϵ). $\mathbf{I}$ denotes the identity matrix of compatible dimensions. The ith column of $\mathbf{X}$ is denoted as $\mathbf{x}_i$. The set of real numbers is denoted by $\mathbb{R}$, while the set of nonnegative real numbers is denoted by $\mathbb{R}_+$.

A variety of matrix norms will be used. The matrix ℓ_0 and ℓ_1 norms are denoted by $\|\mathbf{X}\|_0$ (i.e., the number of nonzero entries in $\mathbf{X}$) and $\|\mathbf{X}\|_1 = \sum_i \sum_j |x_{ij}|$, respectively. $\|\mathbf{X}\|_F = \sqrt{\sum_i \sum_j x_{ij}^2} = \sqrt{\text{tr}(\mathbf{X}^T\mathbf{X})}$ is the Frobenius norm, where tr(.) denotes the trace of a square matrix. The nuclear norm of $\mathbf{X}$ (i.e., the sum of singular values of a matrix) is denoted by $\|\mathbf{X}\|_*$. The ℓ_∞ norm of $\mathbf{X}$, denoted by $\|\mathbf{X}\|_\infty$, is defined as the element of $\mathbf{X}$ with the maximum absolute value.

3. MULTI-LABEL MUSIC ANNOTATION BY LOW-RANK REPRESENTATION

Each music recording is modeled by its slow auditory temporal modulations (ATMs) [11]. The ATMs are obtained by modeling the path of human auditory processing as a two-stage process. In the first stage, which models the early auditory system, the acoustic signal is converted into a time-frequency distribution along a logarithmic frequency axis, the so-called *auditory spectrogram*. In this paper, the early auditory system is modeled by employing Lyons' passive ear model [14]. The auditory spectrogram is then downsampled along the time axis by a factor of 5. The underlying temporal modulations of the music signal are derived by applying a biorthogonal wavelet filter along each temporal row of the auditory spectrogram, where its mean has been previously subtracted, for a set of 8 discrete rates $r \in \{2, 4, 8, 16, 32, 64, 128, 256\}$ Hz ranging from slow to fast temporal rates [11]. Thus, the entire auditory spectrogram is modeled by a three-dimensional (3D) representation of frequency, rate, and time, which is averaged along the time axis yielding a two-dimensional representation.

An ensemble of N training music recordings is represented by a 3D nonnegative array (i.e., 3rd order nonnegative tensor) of dimensions $96 \times 8 \times N$, which is then converted into the data matrix $\hat{\mathbf{X}} \in \mathbb{R}_+^{768\times N}$ by taking the transpose of the unfolded tensor along the samples mode[1]. The entries of $\hat{\mathbf{X}}$ are further post-processed as follows: First, each row of $\hat{\mathbf{X}}$ is normalized to the range $[0, 1]$ by subtracting from each entry the row minimum and then by dividing it with the difference between the row maximum and the row minimum. The columns of $\hat{\mathbf{X}}$ are next normalized in order to have unit ℓ_2 norm. Accordingly, the nth music recording is now represented by the nth column of the normalized ATM representation matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N] \in \mathbb{R}_+^{d\times N}$ with $d = 768$, that is $\mathbf{x}_n \in \mathbb{R}_+^d$, $n = 1, 2, \ldots, N$. Let $\mathbf{Y} \in \{0, 1\}^{V\times N}$ be the tag-music recording matrix, where V indicates the cardinality of the tag vocabulary. Obviously, $y_{ij} = 1$ if the jth recording is labeled with the ith tag in the vocabulary and 0 otherwise. Since, $\mathbf{x}_j$ can be labeled by multiple labels, more than one non-zero elements may appear in $\mathbf{y}_j$.

Given a set of training recordings along with the associated label vectors, the goal is to infer how the label vectors are associated to the ATM representation of the recordings. To this end, it is assumed that the jth recording label vector $\mathbf{y}_j$ can be obtained by $\mathbf{y}_j = \mathbf{W}\mathbf{x}_j + \mathbf{e}_j$, where $\mathbf{W} \in \mathbb{R}^{V\times d}$ is a weight matrix, which captures the relationships between the audio feature space and the semantic space defined by the labels and $\mathbf{e}_j \in \mathbb{R}^V$ is a bias or error term. Therefore, for the entire training set, the label vectors are modeled by $\mathbf{Y} = \mathbf{W}\mathbf{X} + \mathbf{E}$. Building on recent advances in low-rank representations [13] and matrix completion [15], we further assume that the weight matrix $\mathbf{W}$ is low-rank and the error matrix $\mathbf{E}$ is sparse. The underling assumption here is that the tag-recording matrix $\mathbf{Y}$ and the matrix of the ATM representations $\mathbf{X}$ are jointly low-rank. This assumption is possible in many pattern recognition problems (e.g., clustering, classification). Indeed, in content-based audio analysis, one assumes that the data matrix formed by the audio features is low-rank in order to apply say principal component analysis, while in context-based audio analysis (e.g., the tags-based retrieval in social networks) one assumes that the tag-recording matrix is low-rank in order to apply say latent semantic analysis. Here, we assume that content and context are interrelated, which naturally leads to assume that the tag-recording matrix and the matrix of ATM representation are jointly low-rank.

Based on the aforementioned assumptions, the low-rank matrix $\mathbf{W}$ and the sparse matrix $\mathbf{E}$ can be found by solving the optimization problem:

$$\underset{\mathbf{W},\mathbf{E}}{\text{argmin}} \quad \text{rank}(\mathbf{W}) + \lambda\|\mathbf{E}\|_0 \quad \text{subject to } \mathbf{Y} = \mathbf{W}\,\mathbf{X} + \mathbf{E}, \quad (1)$$

where $\lambda > 0$ is a regularization parameter. The optimization problem (1) is difficult to be solved due to the discrete nature of the rank function and the ℓ_0 matrix norm. A convex relaxation of (1) is [13, 15]:

$$\underset{\mathbf{W},\mathbf{E}}{\text{argmin}} \quad \|\mathbf{W}\|_* + \lambda\|\mathbf{E}\|_1 \quad \text{subject to } \mathbf{Y} = \mathbf{W}\,\mathbf{X} + \mathbf{E}. \quad (2)$$

Problem (2) can be solved iteratively by employing the Alternating Direction Augmented Lagrange Multiplier (ADALM) method [12], i.e.,:

$$\underset{\mathbf{W},\mathbf{J},\mathbf{E}}{\text{argmin}} \quad \|\mathbf{J}\|_* + \lambda\|\mathbf{E}\|_1 \quad \text{subject to } \mathbf{Y} = \mathbf{W}\,\mathbf{X} + \mathbf{E},\ \mathbf{W} = \mathbf{J}, \quad (3)$$

which can be solved by minimizing the augmented Lagrange function [12]:

$$\begin{aligned} f(\mathbf{W}, \mathbf{J}, \mathbf{E}, \mathbf{\Lambda}_1, \mathbf{\Lambda}_2) &= \|\mathbf{J}\|_* + \lambda\|\mathbf{E}\|_1 \\ &+ \text{tr}\left(\mathbf{\Lambda}_1^T(\mathbf{Y} - \mathbf{W}\mathbf{X} - \mathbf{E})\right) + \text{tr}\left(\mathbf{\Lambda}_2^T(\mathbf{W} - \mathbf{J})\right) \\ &+ \frac{\mu}{2}\|\mathbf{Y} - \mathbf{W}\mathbf{X} - \mathbf{E}\|_F^2 + \frac{\mu}{2}\|\mathbf{W} - \mathbf{J}\|_F^2, \end{aligned} \quad (4)$$

where $\mathbf{\Lambda}_1, \mathbf{\Lambda}_2$ are the Lagrange multipliers and $\mu > 0$ is a penalty parameter. The minimization of (4) with respect to $\mathbf{W}$, $\mathbf{J}$, and $\mathbf{E}$ can be performed in an alternating fashion by first fixing $\mathbf{W}$ and $\mathbf{E}$ and updating $\mathbf{J}$, next by fixing $\mathbf{J}$ and $\mathbf{E}$ and updating $\mathbf{W}$, then by fixing

[1]The tensor unfolding can be implemented in Matlab by employing the `tenmat` function of the MATLAB Tensor Toolbox available at: `http://csmr.ca.sandia.gov/~tgkolda/TensorToolbox/`.

$\mathbf{J}$ and $\mathbf{W}$ and updating $\mathbf{E}$, and finally updating the Lagrange multipliers. The ADALM method for the minimization of (2) is outlined in Algorithm 1. Its convergence has been demonstrated in [16].

Algorithm 1 Solving (2) by ADALM.

Input: Training matrix $\mathbf{X} \in \mathbb{R}^{d \times N}$, the label matrix $\mathbf{Y} \in \mathbb{R}^{V \times N}$, and the parameter λ.
Output: Weight matrix $\mathbf{W} \in \mathbb{R}^{V \times d}$ and error matrix $\mathbf{E} \in \mathbb{R}^{V \times N}$.

1: Initialize: $\mathbf{W} = \mathbf{J} = \mathbf{E} = \mathbf{0}$, $\mathbf{\Lambda}_1 = \mathbf{0}$, $\mathbf{\Lambda}_2 = \mathbf{0}$, $\mu = 10^{-6}$, $\epsilon = 10^{-8}$, $\mathbf{Q} = \left(\mathbf{I} + \mathbf{X}\mathbf{X}^T\right)^{-1}$.
2: **while** not converged **do**
3: Fix $\mathbf{W}, \mathbf{E}, \mathbf{\Lambda}_1, \mathbf{\Lambda}_2$ and update $\mathbf{J}$ by $\mathbf{J} = \text{argmin } \frac{1}{\mu}\|\mathbf{J}\|_* + \frac{1}{2}\|\mathbf{J} - (\mathbf{W} + \mathbf{\Lambda}_2/\mu)\|_F^2$.
4: Fix $\mathbf{J}, \mathbf{E}, \mathbf{\Lambda}_1, \mathbf{\Lambda}_2$ and update $\mathbf{W}$ by $\mathbf{W} = \left((\mathbf{Y} - \mathbf{E})\mathbf{X}^T + \mathbf{J} + (\mathbf{\Lambda}_1\mathbf{X}^T - \mathbf{\Lambda}_2)/\mu\right)\mathbf{Q}$.
5: Fix $\mathbf{J}, \mathbf{W}, \mathbf{\Lambda}_1, \mathbf{\Lambda}_2$ and update $\mathbf{E}$ by $\mathbf{E} = \text{argmin } \frac{\lambda}{\mu}\|\mathbf{E}\|_1 + \frac{1}{2}\|\mathbf{E} - (\mathbf{Y} - \mathbf{W}\mathbf{X} + \mathbf{\Lambda}_1/\mu)\|_F^2$.
6: Update the Lagrange multipliers by $\mathbf{\Lambda}_1 = \mathbf{\Lambda}_1 + \mu(\mathbf{Y} - \mathbf{W}\mathbf{X})$, $\mathbf{\Lambda}_2 = \mathbf{\Lambda}_2 + \mu(\mathbf{W} - \mathbf{J})$.
7: Update μ by $\mu = \min(1.2 \cdot \mu, 10^6)$.
8: Check convergence conditions $\|\mathbf{Y} - \mathbf{W}\mathbf{X}\|_\infty < \epsilon$ and $\|\mathbf{W} - \mathbf{J}\|_\infty < \epsilon$.
9: **end while**

In Algorithm 1, Step 3 can be solved via the singular value thresholding operator [17], Step 4 amounts to solving the following unconstrained least-squares problem $\mathbf{W} = \text{argmin}_{\mathbf{W}} f(\mathbf{W}, \mathbf{J}, \mathbf{E}, \mathbf{\Lambda}_1, \mathbf{\Lambda}_2)$ whose solution involves $\mathbf{Q} = \left(\mathbf{I} + \mathbf{X}\mathbf{X}^T\right)^{-1}$, while Step 5 is solved by the shrinkage operator [15]. The singular value thresholding operator is defined for any matrix $\mathbf{M}$ as [17]: $\mathcal{D}_\tau[\mathbf{M}] = \mathbf{U}\mathcal{S}_\tau\mathbf{V}^T$ with $\mathbf{M} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ being the singular value decomposition (SVD) and $\mathcal{S}_\tau[m] = \text{sgn}(m)\text{max}(|m| - \tau, 0)$ being the shrinkage operator [15], which can be applied to a matrix in an element-wise manner. The computational cost of Algorithm 1 is that of the SVD (i.e., $O(d \cdot N^2)$) being its most computationally demanding task.

The weight matrix $\mathbf{W}$, obtained by Algorithm 1, captures the semantic relationships between the label space and the audio feature space. In music tagging, the semantic relationships are expected to propagate from the feature space to the label vector space. Let us denote by $\tilde{\mathbf{x}} \in \mathbb{R}^d$ the audio feature representation (ATMs here) of a test music recording and by $\mathbf{a} \in \mathbb{R}^V$ the label vector of this recording. Having found $\mathbf{W}$, $\mathbf{a}$ can be obtained by: $\mathbf{a} = \mathbf{W}\tilde{\mathbf{x}}$. The labels associated with the largest values in $\mathbf{a}$ form the tag vector recommended for annotating the test music recording.

4. EXPERIMENTAL EVALUATION

The performance of the proposed method for automatic music tagging is assessed by conducting experiments on the CAL500 dataset [2]. The CAL500 is a corpus of 500 tracks of Western popular music, each of which has been manually annotated by at least three human annotators, who employ a vocabulary of 174 tags. The tags used in CAL500 dataset annotation span 6 semantic categories, namely instrumentation, vocal characteristics, genres, emotions, acoustic quality of the song, and usage terms (e.g., I would like to listen this song while driving, sleeping) [2].

The length of the tag vector returned by the proposed method is set to 10. That is, each test music recording is annotated with 10 tags. Three metrics, namely the mean per-word precision and the mean per-word recall and the F_1 score are employed in order to assess the annotation performance of the proposed automatic music tagging system whose definitions are as follows [2]: Per-word precision is defined as the fraction of songs annotated by the system with label w that are actually labeled with word w. Per-word recall is defined as the fraction of songs actually labeled with word w that the system annotates with label w. The F_1 score is the harmonic mean of precision and recall. That is, $F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$ yielding a scalar measure of overall annotation performance.

Since the LRRMA requires a significant number of training samples, the evaluation procedure defined in [10] is adopted. That is, the experiments are restricted to 78 tags, which have been employed to tag at least 50 music recordings in the CAL500 dataset.

In Figure 1, the mean precision, the mean recall, and the F_1 score is plotted as a function of the parameter λ, which involved in the LRRMA. The performance of the LRRMA is relatively constant

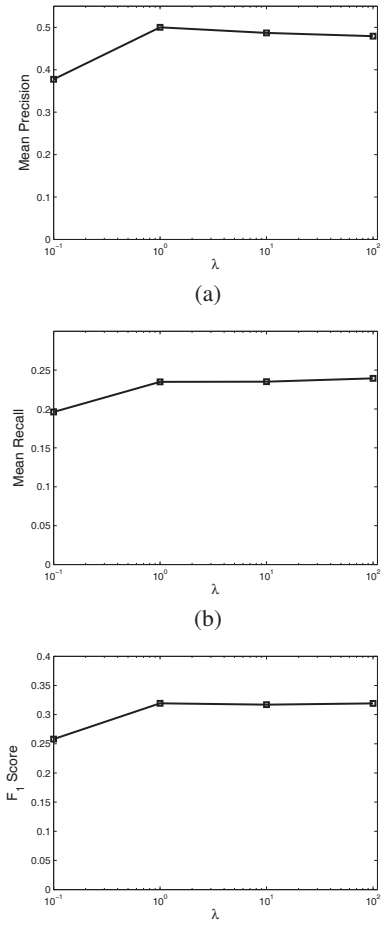

Fig. 1. Mean LRRMA results for various λ values on the CAL500 dataset: (a) mean precision, (b) mean recall, and (c) F_1 score.

for a wide range of λ values, while the best performance in terms of F_1 score is achieved for $\lambda = 1$.

In Table 1, quantitative results for automatic music tagging based on audio features only are summarized. The reported per-

Table 1. Mean annotation results on the CAL500 Dataset.

System	Protocol	Precision	Recall	F_1 Score
LRRMA	5FCV, V = 78	**0.500 (0.004)**	**0.234 (0.0005)**	**0.319**
PARAFAC2 [9]	5FCV, V = 78	0.445 (0.002)	0.223 (0.004)	0.297
HEM-GMM [10]	5FCV, V =78	0.490	0.230	0.260
HEM-DTM [10]	5FCV, V =78	0.470	0.250	0.300
CBA [6] as evaluated in [10]	5FCV, V =78	0.410	0.240	0.290

formance metrics are mean and standard errors (i.e., the sample standard deviation divided by the sample size) inside parentheses computed from 5-fold cross-validation (5FCV) with a vocabulary size $V = 78$ on the CAL500 dataset. By inspecting Table 1, it is seen that the LRRMA clearly exhibits the best performance with respect to the per-word precision, per-word recall, and F_1 score among the state-of-the-art auto-tagging systems, that is compared to, when 5-fold cross-validation is applied. Unlike, the direct performance comparisons for the methods listed in Table 1, wich employ the same protocol, fair comparisons cannot be made between the proposed method and others (e.g., [2, 8]) due to different protocols used.

5. CONCLUSIONS

An effective automatic music tagging method has been proposed that resorts to auditory temporal modulations for music representation, while the relationships between the tags and the features are inferred by a low-rank weight matrix. The results reported advance the state-of-the-art auto-tagging systems in the CAL500 dataset when 5-fold cross-validation is employed.

In the future, the performance of the LRRMA will be investigated by exploiting conventional audio representations such as mel-frequency cepstral coefficients and chroma features. Furthermore, the evaluation of the LRRMA in tag-based music retrieval will be conducted.

Acknowledgements
This research has been co-financed by the European Union (European Social Fund - ESF) and Greek national funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF) - Research Funding Program: Heraclitus II. Investing in Knowledge Society through the European Social Fund.

6. REFERENCES

[1] T. Bertin-Mahieux, D. Eck, and M. Mandel, "Automatic tagging of audio: The state-of-the-art," in *Machine Audition: Principles, Algorithms and Systems*, W Wang, Ed. IGI Publishing, 2010.

[2] D. Turnbull, L. Barrington, D. Torres, and G. Lanckriet, "Semantic annotation and retrieval of music and sound effects," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 16, no. 2, pp. 467–476, 2008.

[3] R. Miotto, L. Barrington, and G. Lanckriet, "Improving auto-tagging by modeling semantic co-occurrences," in *Proc. 11th Int. Symp. Music Information Retrieval*, Utrecht, The Netherlands, 2010, pp. 297–302.

[4] T. Bertin-Mahieux, D. Eck, F. Maillet, and P. Lamere, "Autotagger: A model for predicting social tags from acoustic features on large music databases," *Journal of New Music Research*, vol. 37, no. 2, pp. 115–135, 2008.

[5] M. I. Mandel and D. P. W. Ellis, "Multiple-instance learning for music information retrieval," in *Proc. 9th Int. Symp. Music Information Retrieval*, Philadelphia, USA, 2008, pp. 577–582.

[6] M. Hoffman, D. Blei, and P. Cook, "Easy as CBA: A simple probabilistic model for tagging music," in *Proc. 10th Int. Symp. Music Information Retrieval*, Kobe, Japan, 2009, pp. 369–374.

[7] S. R. Ness, A. Theocharis, G. Tzanetakis, and L. G. Martins, "Improving automatic music tag annotation using stacked generalization of probabilistic svm outputs," in *Proc. 17th ACM Int. Conf. Multimedia*, Beijing, China, 2009, pp. 705–708.

[8] Y. Panagakis, C. Kotropoulos, and G. R. Arce, "Sparse multi-label linear embedding within nonnegative tensor factorization applied to music tagging," in *Proc. of 11th Int. Symp. Music Information Retrieval*, Utrecht, The Netherlands, 2010, pp. 393–398.

[9] Y. Panagakis and C. Kotropoulos, "Automatic music tagging via PARAFAC2," in *Proc. of 2011 IEEE Int. Conf. Acoustics, Speech and Signal Processing*, 2011, pp. 481–484.

[10] E. Coviello, A.B. Chan, and G. Lanckriet, "Time series models for semantic music annotation," *IEEE Tran. Audio, Speech, and Language Processing*, vol. 19, no. 5, pp. 1343 –1359, 2011.

[11] Y. Panagakis, C. Kotropoulos, and G. R. Arce, "Non-negative multilinear principal component analysis of auditory temporal modulations for music genre classification," *IEEE Trans. Audio, Speech, and Language Technology*, vol. 18, no. 3, pp. 576–588, 2010.

[12] D. P. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods*, Athena Scientific, Belmont, MA, 2nd edition, 1996.

[13] G. Liu, Z. Lin, S. Yan, J. Sun, and Y. Ma, "Robust recovery of subspace structures by low-rank representation," *arXiv:1010.2955v4*, 2010.

[14] R. Lyon, "A computational model of filtering, detection, and compression in the cochlea," in *IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Paris, France, 1982, pp. 1282–1285.

[15] E. J. Candes, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?," *Journal of ACM*, vol. 58, no. 3, pp. 1–37, 2011.

[16] J. Eckstein and D. P. Bertsekas, "On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators," *Mathematical Programming*, vol. 55, pp. 293–318, 1992.

[17] J. F. Cai, E. J. Candes, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM Journal Optimization*, vol. 2, no. 2, pp. 569–592, 2009.