A Discriminative and Compact Audio Representation for Event Detection

Liping Jing[1], Bo Liu[1], Jaeyoung Choi[2,3], Adam Janin[2],
Julia Bernd[2], Michael W. Mahoney[2,4], and Gerald Friedland[2,4]

[1]Beijing Key Lab of Traffic Data Analysis and Mining, Beijing Jiaotong University, Beijing, China
[2]International Computer Science Institute, Berkeley, CA, USA
[3]Delft University of Technology, Delft, Netherlands
[4]University of California, Berkeley, CA, USA

{lpjing,12112082}@bjtu.edu.cn
{jaeyoung,janin,jbernd,mmahoney,fractor}@icsi.berkeley.edu

ABSTRACT

This paper presents a novel two-phase method for audio representation: Discriminative and Compact Audio Representation (DCAR). In the first phase, each audio track is modeled using a Gaussian mixture model (GMM) that includes several components to capture the variability within that track. The second phase takes into account both global structure and local structure. In this phase, the components are rendered more discriminative and compact by formulating an optimization problem on Grassmannian manifolds, which we found represents the structure of audio effectively. Experimental results on the YLI-MED dataset show that the proposed DCAR representation consistently outperforms state-of-the-art audio representations: i-vector, mv-vector, and GMM.

CCS Concepts

•Information systems → Multimedia and multimodal retrieval; •Computing methodologies → Machine learning;

Keywords

Event Detection; Audio Data; Discriminative and Compact Representation

1. INTRODUCTION

With the rapid increase in the number of user-generated videos shared on the Internet, it is becoming increasingly advantageous to explore new ways of retrieving them—for example, by automatically detecting events occurring in them.

Approaches to the event detection task have largely focused on visual-based methods—but audio content often provides complementary information. Multimodal approaches that use both visual and audio cues have recently gained traction. However, there has not been much in-depth exploration of how to best leverage audio information, especially unfiltered user-generated audio—though the focus is changing. For example, sound event detection was included in the 2013 DCASE challenge at IEEE AASP [18].

Major aspects of audio-based event detection include audio data representation and learning methodologies. In this work, we focus on the first aspect, audio data representation, which aims to extract specific features that can refine raw audio into higher-level information. These include low-level features (e.g., energy, cepstral, and harmonic features) and intermediate-level features [3]. The most popular low-level features are Mel-frequency cepstral coefficients (MFCCs) [7], as well as first-order statistics [9] and second-order statistics [15] (e.g., mv-vector [16]) derived from the MFCC features.

Another exciting recent approach is i-vectors, which use latent factor analysis to compensate for foreground and background variability [5, 6, 8]. Though these representation methods have shown promising performance, they have some limitations with regard to event detection. For example, most audio representations are derived unsupervised, i.e., they do not make use of existing label information—though label information is very useful in tasks such as image classification [11] and text classification [12]. In addition, these methods risk losing information about geometric structure within the data [17], and they do not capture signal variance within tracks, nor explicitly consider the local structure between Gaussian components, which may be useful for distinguishing events.

In this paper, we address these issues by introducing a Discriminative and Compact Audio Representation (DCAR) to model audio information. This method is implemented in two phases. First, each audio track is modeled using a Gaussian mixture model (GMM) with several components, to capture within-track variability and reduce storage space. Second, by integrating the labels for the audio tracks and the local structure among the Gaussian components, we identify an embedding to reduce the dimensionality of the mixture components and render them more discriminative. Consequently, the discriminative mixture components of the training data can be represented with low dimensionality.

MM '16, October 15-19, 2016, Amsterdam, Netherlands

DOI: http://dx.doi.org/10.1145/2964284.2970377

The DCAR model is presented in Sec. 2. Sec. 3 describes experiments with a real-world dataset, demonstrating that DCAR significantly improves event-detection performance. Conclusions and future work are discussed in Sec. 4.

2. DISCRIMINATIVE AND COMPACT AUDIO REPRESENTATION

2.1 Characterizing Per-Track Variability

Given a set of audio tracks, we first extract their low-level MFCC features. Let $\mathbf{X} = \{\mathbf{X}^i\}_{i=1}^n$ denote a set of n labeled audio files; each file is segmented into m_i frames. Each frame is modeled via a vector with d-dimension MFCC features ($d = 60$), i.e., $x_j^i \in \mathbb{R}^{60}$ (including the first 20 MFCC features and their first-order and second-order derivatives). Previous work has demonstrated that second-order statistics are much more appropriate for describing complicated multimedia data [5]. Therefore, we train a GMM with P components for each audio file (P can be tuned via cross-validation in experiments), and a set of components $G=\{g_i\}_{i=1}^N$ ($N=nP$) can be obtained from n training audio files. Each component has its weight w_i, mean μ_i, and covariance matrix Σ_i, i.e., $g_i = \{w_i, \mu_i, \Sigma_i\}$.

2.2 Identifying a Discriminative Embedding

The audio file can be represented using these mixture components, but this ignores the global structure of the data (e.g., label information) and the local structure among the components (e.g., nearest neighbors). Meanwhile, the original feature representation is usually large (since there are 60 MFCC features, each mean vector has 60 elements, and each covariance matrix contains 60×60 elements), which may be time-consuming in later processing. Therefore we propose a new method for generating a discriminative and compact representation from the high-dimensional mixture components. The DCAR method is summarized in Fig. 1.

Our main goal is to learn an embedding $\mathbf{W} \in \mathbb{R}^{d\times r}$ ($r < d$, here d is number MFCC features, r is embedding space size) based on N components generated from all labeled audio tracks, belonging to L event classes (i.e., $G = \{g_i, \ell_i\}_{i=1}^N$, where ℓ_i is the label for component g_i, based on the label of the corresponding audio file from which g_i was generated, and $L = |\{\ell_i\}_{i=1}^N|$). Therefore, the resulting low-dimensional GMM components should preserve the important structure of the original GMM components as much as possible. We introduce an embedding $\mathbf{W}$ and define new GMM components with mean

$$\hat{\mu} = \mathbf{W}^T\mu \tag{1}$$

and covariance matrix

$$\hat{\Sigma} = \mathbf{W}^T\Sigma\mathbf{W}. \tag{2}$$

The covariance matrix Σ can easily be transformed to a symmetric positive definite matrix by adding a small constant to its diagonal elements. To maintain this property, the embedding $\mathbf{W}$ is constrained to be full rank. A simple way of enforcing this requirement is to impose orthonormality constraints on $\mathbf{W}$ (i.e., $\mathbf{W}^T\mathbf{W} = \mathbf{I}_r$), so that the embedding can be identified by solving an optimization problem on the Grassmannian manifold.

For event detection, each training file has label information, which we also assign to its GMM components. This

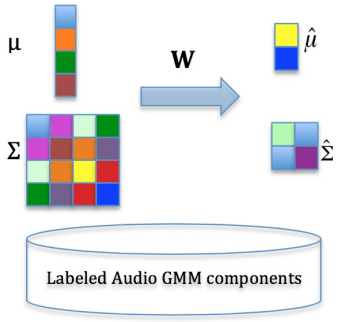

Figure 1: Framework for generating the discriminative and compact audio representation (DCAR). The left side shows the original d-dim GMM components (i.e., $\mu \in \mathbb{R}^d$ and $\Sigma \in \mathbb{R}^{d\times d}$); the right side shows the DCAR representation with r-dim ($r < d$) mixture components (i.e., $\hat{\mu} \in \mathbb{R}^r$ and $\hat{\Sigma} \in \mathbb{R}^{r\times r}$).

valuable information can be interpreted as global structure for those components. There is also intrinsic internal structure among the components, such as the affinity between each pair. When reducing the dimensionality of GMM components, it is necessary to maintain these two types of structure. Motivated by the idea of linear discriminative analysis [14] and the Maximum Margin Criterion [13], DCAR aims to minimize intra-class distance while maximizing inter-class distance, by solving the optimization problem in (3):

$$\begin{aligned}\mathbf{F}(\mathbf{W}) = \min_{\mathbf{W}^T\mathbf{W}=I_r} \sum_{i,j} A_{ij}\Big(\lambda\|\mathbf{W}^T(\mu_i - \mu_j)\|_2^2 \\ + \|\log(\mathbf{W}^T\Sigma_i\mathbf{W}) - \log(\mathbf{W}^T\Sigma_j\mathbf{W})\|_F^2\Big)\end{aligned} \tag{3}$$

The first term indicates the euclidean distance between mean vectors, and the second term is the distance between two covariance matrices measured by the Log-Euclidean Metric (LEM) [2]. The tunable trade-off parameter λ balances the effects of the two terms. The affinity matrix $\mathbf{A}$ is defined by building an intra- (within)-class similarity graph and an inter- (between)-class similarity graph.

$$\mathbf{A}_{ij} = \mathbf{S}_w - \mathbf{S}_b \tag{4}$$

$\mathbf{S}_w$ and $\mathbf{S}_b$ are two binary matrices describing the intra-class and inter-class similarity graphs respectively, formulated as:

$$\mathbf{S}_w(g_i, g_j) = \begin{cases}1 & \text{if } g_i \in \mathrm{NN}_w(g_j) \text{ or } g_j \in \mathrm{NN}_w(g_i) \\ 0 & \text{otherwise}\end{cases} \tag{5}$$

$$\mathbf{S}_b(g_i, g_j) = \begin{cases}1 & \text{if } g_i \in \mathrm{NN}_b(g_j) \text{ or } g_j \in \mathrm{NN}_b(g_i) \\ 0 & \text{otherwise,}\end{cases} \tag{6}$$

where $\mathrm{NN}_w(g_i)$ contains the n_w nearest neighbors of component g_i drawn from G that share the same label as ℓ_i, and $\mathrm{NN}_b(g_i)$ is the set of n_b nearest neighbors of g_i that have different labels. Here, the nearest neighbors of each component can be identified via their similarity. We use heat kernel weight with a self-tuning technique (for parameters σ_μ and σ_Σ) to measure the similarity between components:

$$S(g_i, g_j) = \lambda \exp\left(\frac{-\delta_\mu^2(\mu_i, \mu_j)}{2\sigma_\mu^2}\right) + \exp\left(\frac{-\delta_\Sigma^2(\Sigma_i, \Sigma_j)}{2\sigma_\Sigma^2}\right) \tag{7}$$

where λ is a trade-off parameter, δ_μ is the euclidean metric and δ_Σ is the LEM metric. This matrix $\mathbf{A}$ serves to effectively combine local structure (nearest neighbors) and global structure (label information).

The problem in (3) is a typical optimization problem with orthogonality constraints; it can therefore be formulated as a unconstrained optimization problem on Grassmannian manifolds [1]. Given that the objective function $\mathbf{F}(\mathbf{W})$ has the property that for any rotation matrix $\mathbf{R} \in SO(r)$ (i.e., $\mathbf{R}\mathbf{R}^T = \mathbf{R}^T\mathbf{R} = \mathbf{I}_r$), $\mathbf{F}(\mathbf{W}) = \mathbf{F}(\mathbf{W}\mathbf{R})$, this optimization problem is most compatible with a Grassmannian manifold. In other words, we can model the embedding $\mathbf{W}$ as a point on a Grassmannian manifold $\mathcal{G}(r, d)$, which consists of the set of all linear r-dimensional subspaces of $\mathbb{R}^d$.

We employ the conjugate gradient (CG) technique [1] to solve (3). On a Grassmannian manifold, CG performs minimization along geodesics with specific search directions. Here, the geodesic is the shortest path between two points on the manifold. For each point on the manifold $\mathcal{G}$, its tangent space is a vector space that contains the tangent vectors of all possible curves passing through that point. On the manifold, the tangent vectors must parallel transport along the geodesics. Optimizing $\mathbf{F}(\mathbf{W})$ results in a situation where the low-dimensional components are close if their corresponding original high-dimensional components are event-aware neighbors; otherwise, they will be as far apart as possible.

2.3 Detecting Events Using DCAR

To make use of the manifold structure, we adopted the Kernel Ridge Regression (KRR) method [17] to build the event classifiers. Let $\hat{G} = \{\hat{g}_i\}_{i=1}^N$ and $\hat{g}_i = \{\hat{\mu}_i, \hat{\Sigma}_i\}$ be the learned low-dimensional components. $Y \in \mathbb{R}^{N \times L}$ is the label information (where $\mathbf{Y}_{ij}$=1 if $\hat{g}_i$ belongs to the j^{th} event; otherwise $\mathbf{Y}_{ij}$=0). The KRR method aims to train a classifier by solving the following optimization problem:

$$\min_{\mathbf{H}} \mathbf{J}(\mathbf{H}) = \|\phi(\hat{G})^T\mathbf{H} - \mathbf{Y}\|_F^2 + \alpha\|\mathbf{H}\|_F^2 \qquad (8)$$

where the kernel function can be written as $K = \phi(\hat{G})^T\phi(\hat{G})$. Since each component $\hat{g}_i$ has a mean $\hat{\mu}_i$ and a covariance matrix $\hat{\Sigma}_i$, we adopt the similarity (7) as the kernel function, i.e., $K(\hat{g}_i, \hat{g}_j) = S(\hat{g}_i, \hat{g}_j)$. The problem in (8), as a quadratic convex problem, can be optimized by setting its derivative with respect to H to zero, and then computing H in closed form:

$$\mathbf{H} = \phi(\hat{G})(K + \alpha\mathbf{I})^{-1}\mathbf{Y}$$

Given a new test audio track, the P mixture components $\{g_p\}_{p=1}^P = \{w_p, \mu_p, \Sigma_p\}_{p=1}^P$ can be obtained. Then the corresponding discriminative, low-dimension mixture components $\{\hat{g}_p\}_{p=1}^P$ can be generated, as in (1) for $\hat{\mu}_p = W^T\mu_p$, and as in (2) for $\hat{\Sigma}_p = W^T\Sigma_p W$, where the embedding $\mathbf{W}$ is learned from the training data. The class membership matrix $M = \{M_p\}_{p=1}^P$ (where $M_p \in \mathbb{R}^{1 \times L}$ is the event membership of the p-th component) can be calculated:

$$M_p = \phi(\hat{g}_p)^T\mathbf{H} = \phi(\hat{g}_p)^T\phi(\hat{G})(K + \alpha\mathbf{I})^{-1}\mathbf{Y} = K_p(K + \alpha\mathbf{I})^{-1}\mathbf{Y}$$

Here $K_p = [K(\hat{g}_p, \hat{g}_i)]_{i=1}^N$, indicating the similarity between $\hat{g}_p$ and all of the training mixture components in $\hat{G}$. We can then make a final event prediction for the new audio track with P components using an average voting scheme

$$\ell = arg\max_j \sum_{p=1}^{P} w_p M_p(j)$$

where w_p is the weight of the p^{th} component.

3. EXPERIMENTAL RESULTS

3.1 Dataset and Methodology

Our experiments used YLI-MED [4], a recently released public video corpus with ten events, based on the YFCC100M [19]. Table 1 describes YLI-MED; the variation in track length makes event detection more challenging.

We compared the proposed DCAR with the state-of-the-art audio representations used for event detection: mv-vector [16], i-vector [6], and GMM. By "GMM", we mean the base GMMs extracting the GMM components from each audio file, but without discriminative dimensional reduction. For each method, we tuned the parameters using cross-validation on the training data to obtain the best result.

We describe results using the same classification method, KRR, with all representations.[1] For the l-th event in t testing tracks, we compared the prediction result to the ground truth to determine the number of true positives (TP_l), false positives (FP_l), true negatives (TN_l), and false negatives (FN_l). We evaluated event detection performance using four common metrics, $Accuracy$, $FScore$, $FalseAlarmRate$ (FAR), and $MissRate$. Higher $Accuracy$ and $FScore$ and lower FAR and $MissRate$ indicate better performance.

3.2 Results and Discussion

We evaluated DCAR and the three baseline representations on a ten-event detection task. Table 2 shows detection performance for each event and the average over the ten events, in terms of $FScore$ and $MissRate$. Combined $Accuracy$ for mv-vector, i-vector, GMM, and DCAR is 0.3907, 0.4640, 0.4923, and **0.5321**, respectively (p=0.01 for DCAR vs. each baseline [McNemar's two-tailed]), and the average FAR scores are 0.0674, 0.0593, 0.0570, and **0.0523**.

For each individual event and on average, DCAR achieves superior or competitive performance. In particular, DCAR consistently performs best for all four metrics on the *average* or *combined* scores (an average of more than 8% gain on all metrics relative to the second-best representation).

To explore DCAR's performance under different difficulty levels, we extracted two subsets (based on results from other tasks). Events in EC5 ("EasyCase") (Ev101, Ev104, Ev105, Ev108, and Ev109) are easy to distinguish from (most of) the others; those in HC4 ("HardCase") (Ev103, Ev106, Ev107, and Ev110) are more difficult. For both subsets, DCAR achieved the best results on each evaluation metric, for example, $FScores$ of **0.7067** on EC5 and **0.5283** on HC4 for DCAR, as opposed to 0.4773, 0.6415, and 0.6670 on EC5 and 0.4278, 0.2795, and 0.4821 on HC4 for mv-vector, i-vector, and GMM respectively. Interestingly, i-vector performs better than mv-vector on EC5, but worse on HC4.

We can make a number of observations about these results. First, it seems that modeling GMM components for each audio track (as in GMM, mv-vector, and DCAR) is

[1]Using SVM, KNN, or PLDA for this step does not change the performance rankings between representations.

Table 1: Dataset Composition

Event ID	Event Name	Training Data		Testing Data	
		# of Videos	length (ms)	# of Videos	length (ms)
Ev101	Birthday Party	99	6850∼248950	131	8380∼328960
Ev102	Flash Mob	91	8290∼325630	49	11710∼152560
Ev103	Getting a Vehicle Unstuck	89	5590∼591670	39	11170∼157690
Ev104	Parade	95	7840∼303850	127	5770∼216460
Ev105	Person Attempting a Board Trick	99	5950∼391150	88	5500∼254980
Ev106	Person Grooming an Animal	97	5950∼574300	38	7210∼292870
Ev107	Person Hand-Feeding an Animal	95	6850∼174880	113	7840∼244450
Ev108	Person Landing a Fish	99	7930∼363610	41	7480∼250120
Ev109	Wedding Ceremony	90	9640∼631630	108	9820∼646300
Ev110	Working on a Woodworking Project	98	5590∼373690	44	6760∼281080

Table 2: Per-event comparison of detection performance (as FScore and MissRate) using four representations: mv-vector, i-vector, GMM, and DCAR. (Best results in boldface; second-best underlined.)

	FScore (↑)				MissRate (↓)			
	mv-vector	i-vector	GMM	DCAR	mv-vector	i-vector	GMM	DCAR
Ev101	0.7259	**0.7842**	0.7303	0.7835	0.2824	0.1679	0.1527	**0.1298**
Ev102	0.2837	0.3396	0.3651	**0.4603**	0.5918	0.6327	0.5306	**0.4082**
Ev103	0.2178	0.2569	0.2410	**0.3820**	0.7179	0.6410	0.7436	**0.5641**
Ev104	0.4274	0.6206	0.6000	**0.6207**	0.6063	0.4331	0.3622	**0.3621**
Ev105	0.3354	0.3899	**0.5714**	0.5178	0.6932	0.6477	**0.3864**	0.4205
Ev106	0.1964	0.1835	0.2963	**0.3750**	0.7105	0.7368	0.6842	**0.6053**
Ev107	0.3850	0.3298	0.3250	**0.4024**	**0.6814**	0.7257	0.7699	0.7080
Ev108	0.3191	0.3853	0.3878	**0.4231**	0.6341	0.4878	0.5366	**0.4634**
Ev109	0.4211	0.5028	0.4286	**0.5176**	0.6667	**0.5833**	0.6667	0.5926
Ev110	0.0833	0.2299	**0.2857**	0.2162	0.9091	0.7727	**0.7500**	0.8182
Average	0.3395	0.4023	0.4231	**0.4699**	0.6494	0.5829	0.5583	**0.5072**

more effective than modeling a GMM on all the training audio tracks together (as in i-vector) when the events are closely related, as in HC4. We believe this is because, in real-world applications (e.g., with user-generated content), each track may have a large variance. The set of strategies that model each track via GMM capture the hidden structure within each audio track, while the i-vector strategy may smooth away that structure (even between events), leading to a less useful representation.

Second, GMM and DCAR perform better than mv-vector on both subsets, which indicates that one mixture component (as in mv-vector) may not sufficiently capture the full structure of the audio; in addition, vectorizing the mean and variance inevitably distorts the intrinsic geometrical structure among the data. Third, DCAR outperforms the base GMM, because DCAR takes into account the label information and the intrinsic nearest neighbor structure among the audio files when modeling the training data, and outputs a mapping function to effectively represent the test data. This result confirms that discriminative dimensionality reduction is beneficial for characterizing the distinguishing information for each audio file, leading to a better representation and significantly improving event detection performance.

4. CONCLUSIONS AND FUTURE WORK

We have presented a new audio representation, DCAR, and demonstrated its use in event detection.[2] The DCAR method stands out in its ability to capture the variability within each audio file. In addition, it achieves better discriminativity by integrating label information and the graph of the components' nearest neighbors among the audio files. In experiments, representing audio using the proposed DCAR notably improves performance on event detection. In a nutshell, the novelty of DCAR lies in its being a *compact representation* of an audio signal that *captures variability* and has *better discriminative ability* than other representations.

Videos are of course *multi*modal; visual content, captions, and other metadata can provide valuable information. We therefore plan to use these to extend the current model, along with complex information about temporal evolution [3]. We may also explore modeling only the most information-rich audio segments. Within audio, we also hope to evaluate the use of DCAR for other related tasks, such as audio scene classification (for example, testing it with the DCASE acoustic scenes dataset [18]).

[2]Further details about how DCAR is built and additional experimental results can be found in Jing et al. 2016 [10].

5. ACKNOWLEDGMENTS

This work was partially supported by the NSFC (61370129, 61375062), the PCSIRT (Grant IRT201206), and a collaborative LDRD led by Lawrence Livermore National Laboratory (U.S. Dept. of Energy contract DE-AC52-07NA27344). Any findings and conclusions are the authors', and do not necessarily reflect the views of the funders.

6. REFERENCES

[1] P. Absil, R. Mahony, and R. Sepulcher, editors. *Optimization algorithms on matrix manifolds*. Princeton University Press, 2008.

[2] V. Arsigny, P. Fillard, X. Pennec, and N. Apache. Geometric means in a novel vector space structure on symmetric positive definite matrices. *SIAM on Matrix Analysis*, 29(1):328–347, 2007.

[3] D. Barchiesi, D. Giannoulis, D. Stowell, and M. Plumbley. Acoustic scene classification: classifying environments from the sounds they produce. *Signal Processing Magazine*, 32(3):16–34, 2015.

[4] J. Bernd, D. Borth, B. Elizalde, G. Friedland, H. Gallagher, L. Gottlieb, A. Janin, S. Karabashlieva, J. Takahashi, and J. Won. The YLI-MED corpus: Characteristics, procedures, and plans (TR-15-001). Technical report, ICSI, 2015. arXiv:1503.04250.

[5] N. Dehak, P. Kenny, R. Dehak, P. Dumouchefl, and P. Ouellet. Front-end factor analysis for speaker verification. *IEEE Transactions on Audio, Speech and Language Processing*, 19(4):788–798, 2011.

[6] B. Elizalde, H. Lei, and G. Friedland. An i-vector representation of acoustic environment for audio-based video event detection on user generated content. In *Proceedings of the IEEE International Symposium on Multimedia*, 2013.

[7] A. Eronen, J. Tuomi, A. Klapuri, and S. Fagerlund. Audio-based context awareness - acoustic modeling and perceptual evaluation. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 529–532, 2003.

[8] Z. Huang, Y. Cheng, K. Li, V. Hautamaki, and C. Lee. A blind segmentation approach to acoustic event detection asked on i-vector. In *Proceedings of INTERSPEECH*, 2013.

[9] Q. Jin, P. Schulman, S. Rabat, S. Burger, and D. Ding. Event-based video retrieval using audio. In *Proceedings of INTERSPEECH*, pages 2085–2088, 2012.

[10] L. Jing, B. Liu, J. Choi, A. Janin, J. Bernd, M. W. Mahoney, and G. Friedland. DCAR: A discriminative and compact audio representation to improve event detection. *http://arxiv.org/abs/1607.04378*, 2016.

[11] L. Jing, C. Zhang, and M. Ng. SNMFCA: Supervised NMF-based image classification and annotation. *IEEE Transactions on Image Processing*, 21(11):4508–4521, 2012.

[12] M. Lan, C. Tan, J. Su, and Y. Lu. Supervised and traditional term weighting methods for automatic text categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(4):721–735, 2009.

[13] H. Li, T. Jiang, and K. Zhang. Efficient and robust feature extraction by maximum margin criterion. In *Proceedings of the Conference on Neural Information Processing Systems (NIPS)*, 2004.

[14] G. McLachlan, editor. *Discriminant analysis and statistical pattern recognition*. Wiley Interscience, 2004.

[15] R. Mertens, H. Lei, L. Gottlieb, G. Friedland, and A. Divakaran. Acoustic super models for large scale videos event detection. In *Proceedings of ACM Multimedia*, 2011.

[16] G. Roma, W. Nogueira, and P. Herrera. Recurrence quantification analysis features for auditory scene classification. In *Proceedings of IEEE AASP Challenge on DCASE*, 2013.

[17] B. Scholkopf and A. Smola, editors. *Learning with Kernels*. MIT Press, 2002.

[18] D. Stowell, D. Giannoulis, E. Benetos, M. Lagrange, and M. Plumbley. Detection and classification of audio scenes and events. *IEEE Transactions on Multimedia*, 17(10):1733–1746, 2015.

[19] B. Thomee, D. Shamma, G. Friedland, B. Elizalde, K. Ni, D. Poland, D. Borth, and L. Li. YFCC100M: The new data in multimedia research. *Communications of the ACM*, 59(2):64–73, 2016.