Analysis of Environmental Sounds

Keansub Lee

Submitted in partial fulfillment of the
requirements for the degree
of Doctor of Philosophy
in the Graduate School of Arts and Sciences

COLUMBIA UNIVERSITY

2009

Abstract

Analysis of Environmental Sounds

Keansub Lee

Environmental sound archives - casual recordings of people's daily life - are easily collected by MP3 players or camcorders with low cost and high reliability, and shared in the web-sites. There are two kinds of user generated recordings we would like to be able to handle in this thesis : Continuous long-duration personal audio and Soundtracks of short consumer video clips.

These environmental recordings contain a lot of useful information (semantic concepts) related with activity, location, occasion and content. As a consequence, the environment archives present many new opportunities for the automatic extraction of information that can be used in intelligent browsing systems. This thesis proposes systems for detecting these interesting concepts on a collection of these real-world recordings.

The first system is to segment and label personal audio archives - continuous recordings of an individual's everyday experiences - into 'episodes' (relatively consistent acoustic situations lasting a few minutes or more) using the Bayesian Information Criterion and spectral clustering.

The second system is for identifying regions of speech or music in the kinds of energetic and highly-variable noise present in this real-world sound. Motivated by psychoacoustic evidence that pitch is crucial in the perception and organization of sound, we develop a noise-robust pitch detection algorithm to locate speech or music-like regions. To avoid false alarms resulting from background noise with strong periodic components (such as air-conditioning), a new scheme is added in order to

suppress these noises in the domain of autocorrelogram.

In addition, the third system is to automatically detect a large set of interesting semantic concepts, which we chose for being both informative and useful to users, as well as being technically feasible. These 25 concepts are associated with people's activities, locations, occasions, objects, scenes and sounds, and are based on a large collection of consumer videos in conjunction with user studies. We model the soundtrack of each video, regardless of its original duration, as a fixed-sized clip-level summary feature. For each concept, an SVM-based classifier is trained according to three distance measures (Kullback-Leibler, Bhattacharyya, and Mahalanobis distance).

Detecting the time of occurrence of a local object (for instance, a cheering sound) embedded in a longer soundtrack is useful and important for applications such as search and retrieval in consumer video archives. We finally present a Markov-model based clustering algorithm able to identify and segment consistent sets of temporal frames into regions associated with different ground-truth labels, and at the same time to exclude a set of uninformative frames shared in common from all clips. The labels are provided at the clip level, so this refinement of the time axis represents a variant of Multiple-Instance Learning (MIL).

Quantitative evaluation shows that the performance of our proposed approaches tested on the $60h$ personal audio archives or 1900 YouTube video clips is significantly better than existing algorithms for detecting these useful concepts in real-world personal audio recordings.

Contents

1 Introduction **1**

1.1 Motivation 1

1.2 Contribution 6

1.3 Organization 7

2 Background **10**

2.1 Segmenting and Clustering 10

2.2 Speech and Music Detection 12

2.3 Generic Concept Detection 15

2.4 Markov Model-based Local Concept Detection 18

2.5 Summary 19

3 Segmenting and Clustering **20**

3.1 Features, segmenting and clustering algorithm 20

3.1.1 Features 21

3.1.1.1 Short-time features 22

3.1.1.2 Long-time features 24

3.1.2 Unsupervised Segmentation 26

3.1.3 Clustering 28

3.2 Evaluations 31

3.2.1 The data of personal audio recordings 31

3.2.2 Features and Segmentation Results 32
3.2.3 Clustering Results 35
3.2.4 Varying the time-frame 37
3.3 Discussion 39
3.3.1 Visualization and browsing 39
3.3.2 Scavenging other data sources 40
3.3.3 Speech and privacy 43
3.4 Summary 45

4 Speech and Music Detection **46**

4.1 Voice Activity Detection in Personal Audio Recordings Using Autocorrelogram Compensation 47
4.1.1 Noise-robust Voiced Pitch Detection 49
4.1.1.1 Multichannel Autocorrelogram 50
4.1.1.2 Autocorrelogram Compensation 51
4.1.1.3 Linear compensation 51
4.1.1.4 Non-linear compensation 52
4.1.1.5 Cross-channel Integration and HMM Pitch Tracking 54
4.2 Detecting Music in Ambient Audio by Long-window Autocorrelation 55
4.2.1 Noise-robust Musical Pitch Detection 56
4.2.1.1 LPC Whitening and ACF 57
4.2.1.2 ACF Compensation 57
4.2.1.3 Aperiodic Noise Suppression 59
4.2.1.4 Long-time Stationary Periodic Noise Suppression . . 59
4.2.2 Pitch Dynamics Estimation 60
4.3 Evaluations 60
4.3.1 Speech Detection Results 61

4.3.2 Music Detection Results 62
4.4 Discussion 66
4.5 Summary 67

5 Generic Concept Detection **68**
5.1 Data and Labels 69
5.1.1 The Semantic Concepts 69
5.1.2 Video Data 70
5.2 Audio Concept Detection Algorithms 73
5.2.1 Support Vector Machines (SVMs) 74
5.2.2 Single Gaussian Modeling (1G) 75
5.2.3 Gaussian Mixture Models (GMM) 77
5.2.4 Probabilistic Latent Semantic Analysis (pLSA) 78
5.3 Evaluations 82
5.4 Discussion 87
5.5 Summary 94

6 HMM-based Local Concept Detection **95**
6.1 Detecting multiple Local Concepts From Global Annotations 95
6.1.1 HMM-based Clustering 96
6.2 Evaluations 100
6.3 Discussion 103
6.4 Summary 106

7 Conclusions **107**

Bibliography **110**

List of Figures

1.1 Data capture equipment. In the middle of the picture the iRiver flash memory recorder. The larger unit to the right is a data logger recording ambient temperature, which we have considered as a proxy for more specific ground truth on location changes. 2

2.1 Spectrogram of example of noisy speech from a personal audio recording. Individual prominent syllables are visible as sets of parallel harmonics below 1 kHz, but this signal would be much too noisy for current speech activity detection or speaker identification techniques. 13

3.1 Examples of the six long-time-frame statistic features based on 21-band auditory (Bark-scaled) spectra. The underlying data is eight hours of recordings including a range of locations. White vertical lines show our hand-marked episode boundaries (see text). 25

3.2 Affinity matrix for the 127 automatic segments. Segments are ordered according to the dominant ground-truth label in each segment. . . . 30

3.3 ROC curves for segment boundary detection based on different summaries and combinations of the 21 bin Auditory Spectral features. . 34

3.4 Confusion matrix for the sixteen segment class labels, calculated over the 3753 one-minute frames in the test data. 36

3.5 Example of segmenting and clustering 8-hour personal audio clip base on the Average Log Energy and Entropy Deviation features. 37

3.6 Effect on labeling frame accuracy of varying the basic time-frame duration. 38

3.7 Screenshot from our experimental browser. Recorded audio is shown by a pseudocolor spectrogram with a vertical time axis. Next to this are the automatically derived segments along with their per-cluster manual labels. The display also shows appointments read from the user's online calendar - a useful prompt in navigating the recordings and interpreting the automatic segments. 41

4.1 Example of noisy speech from a personal audio recording. The pitch tracks in (b) and (c) are extracted by a noise-robust PDA as described in the text; pane (d) shows the result of our algorithm with the same input signal. The pitch of a stationary periodic air-conditioning noise appears as flat contours around lags 105 and 210 in (b), and tends to be more dominant around 4-6 s in (c) due to the failure of a noise estimation of the spectral subtraction, but is clearly deleted by our method in (d). Shadowed regions indicate manually-labeled voiced segments. 48

4.2 Block diagram of our proposed system. 50

4.3 SACs for the input signal from figure 4.1 with and without compensation using the local-average ACF over a 1 s window. Stationary harmonic air-conditioning noise appears as a sequence of strong peaks at lags of 105, 210 and 315 samples in the original SAC, but is clearly deleted in the non-linear compensated SAC (panel (c)), which also preserves speech information lost in the linear compensated SAC of panel (b). The non-linear compensated SAC is the basis of the enhanced pitch tracks shown in figure 4.1 (d). 54

4.4 Example of clean music sound showing the pitch (panel(b) and temporal rhythm (panel (c)). 56

4.5 Examples of noisy speech, music and machine sound from a consumer audio recording. 58

4.6 Distribution of 750 sound clips of a testing set along the musical pitch and rhythm spaces 66

5.1 Co-occurrence matrix for the 25 manually-annotated labels within the 1,873 video set. Co-occurrence counts within each row are normalized by the total number of instances of that row's concept to give the conditional probability of observing the overlapped concept given the labeled concept. 72

5.2 The process of calculating clip-level features via a single Gaussian model per clip, and using them within an SVM classifier. 75

5.3 Illustration of the calculation of pLSA-based features and clip-level comparisons, based on GMM component histograms. Top left shows the formation of the global GMM; bottom left shows the formation of the topic profiles, $P(g|z)$ and topic weights, $P(z|c_{train})$ in training data; top right shows the analysis of each testing clip into topic weights, $P(z|c_{test})$ by matching each histogram to a combination of topic profiles estimated by training data, and bottom right shows the final classification by an SVM. 81

5.4 Average precision (AP) across all 25 classes for the Single Gaussian models (1G), using each of the three distance measures, KL, Mahalanobis, and Bhattacharyya. Labels are sorted by the guessing baseline performance (shown). Bars and error-bars indicate the mean and standard deviation over 5-fold cross-validation testing respectively. . 83

5.5 As Fig. 5.4, but using Gaussian Mixture models (GMMs) with 2, 4, 8, and 16 components, and approximated Bhattacharyya distance. . 84

5.6 As Fig. 5.4, but using pLSA modeling of component-use histograms for GMMs of 256, 512, and 1024 components. Also shown is performance using the 256 component histogram directly, without pLSA modeling. 84

5.7 AP averaged across all classes for pLSA models using different numbers of 'topics' (latent dimensions) and different treatments for the inferred per-clip topic strengths, $p(z|c)$. 85

5.8 The best results from Figs. 5.4, 5.5, and 5.6, illustrating the relative performance of each representation. 86

5.9 Confusion matrix of classified labels within 750 testing clips according to three approaches. 88

5.10 The result with MFCC and LEnergy + Entropy. 89
5.11 Examples of retrieval results for the "cheer" concept. Shown are the top 15 results for each of the best-performing detection systems, 1G+KL2, 8GMM+Bha, and pLSA500+lognorm. Highlighted results are correct according to manual labeling; the number of correct results is shown in the heading for each pane. 90
5.12 Example pLSA topic weights (i.e. $p(z|c)$) across all concepts for a 100-topic model. Topic columns are sorted according to the concept for which they have the largest weight. 91
5.13 LDA projection of 25 concepts into two-dimensional subspace. 93

6.1 Example of clustering consistent a set of temporal frames into segments corresponding to each concept using the first-order 3-state Markov process. 97
6.2 Example analysis of a soundtrack consisting of a conversation at the beach. Speech is clustered into the global background, but cheers, and background beach noises are correctly identified. 101
6.3 Log-scaled transition matrix trained and modified. 106

List of Tables

1.1 Relationship between dataset and our proposed algorithms. 7

3.1 Segment classes, counts, and average duration (in minutes) from the manual annotation of the 62 hour test set. 32

3.2 Sensitivity @ Specificity = 0.98 for each feature set. Values greater than 0.8 are shown in bold. 33

3.3 Sensitivity @ Specificity = 0.98 for different combinations of the three best statistics based on the 21- bin Auditory Spectrum. 34

4.1 Voice detection performance. The accuracy rate is the proportion of voiced frames correctly detected, and d' (threshold-independent measure of class separation). The best value in each row is shown in bold. The best threshold for zero-pitch probability was estimated as the 61^{st} percentile of the SAC for the Binary Decision with Pitch Tracks system. 62

4.2 Speech / Music (with or w/o vocals) classification accuracy of broadcasting recordings with one Gaussian classifier. Each value indicates how many of the 2.4 second segments out of a total of 120 are correctly classified as speech or music. The best performance of each column is shown in bold. 63

4.3 Music/Non-music Classification Performance on YouTube consumer environmental recordings. Each data point represents the mean and standard deviation of the clip-based performance over 10 cross-validated experiments. Where d' is a threshold-independent measure of the separation between two unit-variance Gaussian distributions and AP is the Average of Precisions calculated for each of relevant examples separately to be a higher value when more relevant examples, i.e. music clips, is returned earlier. The best performance of each column is shown in bold. 65

5.1 *Definition of the* 25 *concepts, and counts of manually-labeled examples of each concept from* 1,873 *videos.* 71

5.2 *Consumer (*59.8%*) / Non-consumer (*40.2%*) Classification Performance on* 3,134 *YouTube recordings based on the single Gaussian model with the KL distance measure. Each data point represents the mean and standard deviation of the clip-level performance over* 5 *cross-validated test folds.* 83

6.1 *Supervised (using frame-scaled hand-labels) concept classification performance on YouTube videos. The second column indicates how many of the clips tagged with the concept actually contained relevant sounds; the third column gives the average duration of relevant sound within those clips. The fourth column shows the (frame-scale) prior of these concepts. Values in columns* 6 *through* 9 *represent means of the frame-level performance over* 5 *cross-validated experiments. Concepts are evaluated with accuracy, d′, and average precision (AP), and the best performance of each row is shown in bold. Note that accuracy rate isn't the good measure of performance when the prior of non-concept's frames is severely high. Different columns indicate different experimental conditions, as explained in the text.* 102

6.2 *Semi-supervised (using clip-scaled hand-labels) concept classification performance on YouTube videos. Values in columns 3 through 8 represent means of the frame-level performance over 5 cross-validated experiments. Different columns indicate different experimental conditions, as explained in the text.* 105

Acknowledgments

I would like to express my sincere appreciation to my advisor, Professor Daniel P. W. Ellis. His excellent guidance and warm support have made my research fun and rewarding during my graduate study. He is the most erudite advisor and gentle mentor who has satisfied my interest and curiosity, and who has encouraged me to overcome a lot of difficulties.

I would also like to express my gratitude to Professors John R. Kender, Shih-Fu Chang, Xiaodong Wang and Dr. Alexander C. Loui at Kodak company for serving on my Ph.D committee and for their insightful comments and discussions on this thesis.

I am very grateful to all the members of the LabROSA for sharing a moment of fun and friendship; Adam Berenzweig, Manuel Reyes-Gomez, Marios Athineos, Michael Mandel, Graham Poliner and Sambarta Bhattacharjee, Ron Weiss, Xanadu Halkias, Courtenay Cotton and Christine Smit.

Finally, I would like to thank my parents and my wife Sunoung Jang. Their endless love and optimistic mind help me to do my best to peacefully and steadily step forward to gain my goal.

Chapter 1

Introduction

This thesis presents a computational frameworks for analyzing the environmental sounds.

1.1 Motivation

Preservation and recollection of facts and events are central to human experience and culture, yet our individual capacity to recall, while astonishing, is also famously fallible. As a result, technological memory aids date back to cave paintings and beyond; more recent trends include the shift from specific, active records (such as making notes) to transparent, comprehensive archives (such as the 'sent-mail' box of an email application) - which become increasingly valuable as the tools for retrieving the contents improve.

We have been investigating what we see as a natural extension of this trend to the large-scale collection of daily personal experiences in the form of audio recordings, striving to capture everything heard by the individual user during the time archives are collected. Our interest in this problem stems from work in content-based retrieval, which aims to make multimedia documents such as movies and

Figure 1.1: Data capture equipment. In the middle of the picture the iRiver flash memory recorder. The larger unit to the right is a data logger recording ambient temperature, which we have considered as a proxy for more specific ground truth on location changes.

videos searchable in much the same way that current search engines allow fast and powerful retrieval from text documents and archives. However, automatic indexing of movies has to compete with human annotations (e.g. subtitles) - if the ability to search is important enough to people, it will be worth the effort to perform manual annotation. But for data of speculative or sparse value, where manual annotation would be out of the question, automatic annotation is a much more compelling option. Recordings of daily experiences - which may contain interesting material in much less than 1% of their span - are a promising target for automatic analysis.

The second spur to our interest in this project was the sudden availability of devices capable of making these kinds of 'environmental' audio archives - recordings of an individual's daily life - at low cost, with high reliability, and with minimal impact to the individual. There are two kinds of user generated recordings we would like to be able to handle : Personal audio and Soundtracks of consumer video.

The continuous long-duration recordings of 'Personal audio' - storing essentially everything heard by the owner - are easily collected by a body-worn MP3 player with 1GB of flash memory and a built-in microphone, able to record continuously for about 16 hours, powered by a single rechargeable AA battery as shown in Figure 1.1. This kind of technology, along with the plummeting cost of mass storage (i.e., a year's worth of recordings is maybe 60GB or a small stack of writable DVDs), makes the collection of large personal audio archives astonishingly cheap and easy. The 'consumer videos' are also in some cases replacing still-image snapshots as a medium for the causal recording of daily life. More and more people are capturing and recording their experiences using the video recording functions of small and inexpensive digital cameras and camcorders. These short video clips are commonly shared with others via sharing sites such as YouTube [2].We are particularly interested in exploiting the acoustic information - the soundtrack of a video - and in seeing what useful information can be reliably extracted from these consumer clips.

We have chosen to use audio recordings, instead of video, as the foundation for our environmental archive system. While the information captured by audio and video recordings is clearly complementary, we see several practical advantages to using audio only: Firstly, an omnidirectional microphone is far less sensitive to positioning or motion than a camera. It is possible to capture information from all directions and are largely robust to sensor position and orientation (and lighting), allowing data collection without encumbering the user. Secondly, because audio data rates are at least an order of magnitude smaller than video, the recording devices can be much smaller and consume far less energy. Thirdly, the nature of audio is distinct from video, making certain kinds of information (e.g. what is said) more accessible, and other information (e.g. the presence of nonspeaking individuals) unavailable [54].

Potentially, processing the content of an audio archive could provide a wide range of useful information:

- **Location**: Particular physical locations frequently have characteristic acoustic ambiences that can be learned and recognized, as proposed in [14](e.g. beach has a water-wave sound). The particular sound may even reveal finer gradations than pure physical location (e.g. the same restaurant empty vs. busy), although at the same time it is vulnerable to different confusions (e.g. mistaking one restaurant ambience for another).

- **Activity**: Different activities are in many cases easily distinguished by their sounds e.g. typing on a computer vs. having a conversation vs. reading. Skiing activity would generate a skier noise.

- **People**: Speaker identification based on the acoustic properties of voice is a mature and successful technology [56]. However, it requires some adaptation to work with the variable quality and noise encountered in real-world audio.

- **Words**: The fantasy of a personal memory prosthesis is the machine that can fulfill queries along the lines of: "This topic came up in a discussion recently. What was that discussion about?", implying not only that the device has recognized all the words of the earlier discussion, but that is can also summarize the content and match it against related topics. This seems ambitious, although similar applications are being pursued for recordings of meetings [51, 55]. Capturing casual discussions also raises serious privacy concerns.

  A more palatable and possibly more feasible approach is to mimic the pioneering Forget-me-not system [37] in capturing tightly-focused 'encounters' or events, such as the mention of specific facts like telephone numbers, web ad-

dresses etc. [26]. This could work as an automatic, ubiquitous version of the memo recorders used by many professionals to capture momentary ideas.

- **Contents**: For example, one attribute that we see as both informative and useful to users, and at the same time technically feasible, is the detection of background music. If a user is searching for the video clip of a certain event, they are likely to able to remember (or guess) if there was music in the background, and thereby limit the scope of a search. In a manual labeling of a database of over 1000 video clips recorded by real users of current digital cameras (which include video capability),approximately 18% were found to include music - enough to be a generally-useful feature, while still retaining some discriminative power.

While the collection of large environmental audio archives provide a wide range of surely valuable information such as the daily locations and activities of the user, no tools currently exist to make such recordings remotely worthwhile. To review a particular event would require loading the whole file into an audio browser and making some kind of linear search: guessing the approximate time of the event of interest, then listening to little snippets and trying to figure out whether to scan forwards or backwards. The time required for this kind of search begins to approach the duration of the original recording, and renders any but the most critical retrieval completely out of the question. Our purpose is to develop tools and techniques that could turn these easily-collected environmental audio archives into something useful and worthwhile.

1.2 Contribution

In this dissertation, we have described a vision of personal audio archives and presented our work on providing automatic indexing based on the statistics of frequency-warped short-time energy spectra calculated over windows of seconds or minutes. Our automatically clustered segments based on the Bayesian Information Criterion and spectral clustering can be grouped into similar or recurring classes which, once the unknown correspondence between automatic and ground-truth labels is resolved, gives frame-level accuracies of over 80% on our 62 h hand-labeled test set.

In addition, we have proposed a robust pitch detection algorithm for identifying the presence of speech or music in the noisy, highly-variable personal audio collected by body-worn continuous recorders. In particular, we have introduced a new technique for estimating and suppressing stationary periodic noises such as air-conditioning machinery in the autocorrelation domain. The performance of our proposed algorithm is significantly better than existing speech or music detection systems for the kinds of data we are addressing.

Subsequently, we have described several variants of a system for classifying consumer videos into a number of semantic concept classes, based on features derived from their soundtracks. Specifically, we have experimented with various techniques for summarizing low-level MFCC frames into fixed-size clip-level summary features, including Single Gaussian Models, Gaussian Mixture Models, and probabilistic Latent Semantic Analysis of the Gaussian Component Histogram. We constructed SVM classifiers for each concept using the Kullback-Leibler, Bhattacharyya, and Mahalanobis distances. In spite of doubts over whether soundtrack features can be effective in determining content classes such as “picnic” and “museum” that do not have obvious acoustic correlates, we show that our classifiers are able to achieve APs far above chance, and in many cases at a level likely to be useful in real retrieval

Domain	Segmentation	Generic Concepts	Specific Concepts
Personal Audio	Chap. 3		Chap. 4
Consumer Video	Chap. 6	Chap. 5	

Table 1.1: Relationship between dataset and our proposed algorithms.

tasks.

However, the concepts have diverse characteristics in terms of consistency, frequency and interrelationships. For example, the "music" and "crowd" typically persist over a large proportion if not the entirety of any clip to which they apply, and hence should be well represented in the global feature patterns (e.g., mean and covariance of entire frames of a clip). However, the concept "cheer" manifests as a relatively small segment within a clip (at most a few seconds within 1 minute clip) which means that the global patterns of an entire clip may fail to distinguish it from others. We have developed a Markov model based clustering algorithm for detecting the local patterns (at the frame scale) embedded in a global background based on the acoustic information - soundtrack of a video - while annotations are presented at the level of a clip.

The work directly related to this thesis was reported in two journal articles [23, 40] and seven conference proceedings [21, 22, 38, 39, 17, 10, 9].

1.3 Organization

Table 1.1 shows how our proposed algorithms are related on two different dataset respectively. The segmentation, 16-location concept and speech detection algorithms are tested on the personal audio archives, and local concept segmentation, 25 generic concept and music detection methods are evaluated on the soundtrack of consumer videos.

The remainder of the thesis is organized as follows:

In Chapter 2, we provide background information and a discussion of prior works on segmenting and clustering of personal audio, detecting speech or music in an environmental sound, and generic or local concepts detecting on consumer videos based on their soundtracks.

In Chapter 3, we describe our approaches in segmenting and labeling personal audio archives - continuous recordings of an individual's everyday experiences - into 'episodes' (relatively consistent acoustic situations lasting a few minutes or more)using the Bayesian Information Criterion [11] and spectral clustering [53].

In Chapter 4, we presents a novel method for identifying regions of speech or music in the kinds of energetic and highly-variable noise present in a real-world sound collected by body-worn recorders. Motivated by psychoacoustic evidence that pitch is crucial in the perception and organization of sound, we develop a noise-robust pitch detection algorithm to locate speech or music-like regions. To avoid false alarms resulting from background noise with strong periodic components (such as air-conditioning), we add a new scheme to suppress these noises in the domain of autocorrelogram.

In Chapter 5, we describe a system to automatically detect a large set of interesting semantic concepts, which we chose for being both informative and useful to users, as well as being technically feasible. These concepts are associated with people's activities, locations, occasions, objects, scenes and sounds, and are based on a large collection of consumer videos in conjunction with user studies. We model the soundtrack of each video, regardless of its original duration, as a fixed-sized clip-level summary feature. For each concept, an SVM-based classifier is trained according to three distance measures (Kullback-Leibler, Bhattacharyya, and Mahalanobis distance) and tested on $1,900$ consumer clips.

In Chapter 6, we develop a novel Multiple Instance Learning (MIL) approach, namely a Markov model-based clustering algorithm able to segment a set of temporal frames into regions associated with different ground-truth labels tagged at the clip level, and at the same time to exclude uninformative "background" frames shared in common from all clips.

Finally, in Chapter 7, we make concluding remarks regarding the merits and limitations of our frameworks and propose directions for future work.

Chapter 2

Background

In this chapter, we provide background information and a discussion of prior works on segmenting and clustering of personal audio, detecting speech or music in an environmental sound, and generic or local concept detecting on consumer videos based on their soundtracks.

2.1 Segmenting and Clustering

The idea of using technology to aid human memory extends back as far as the earliest precursors to writing; more recently, the idea of literal recordings of particular events has extended from photographs and recordings of 'special events' to the possibility of recording everything experienced by an individual, whether or not it is considered significant. The general idea of a device to record the multitude of daily experience was suggested in 1945 by Bush [8], who noted:

> Thus far we seem to be worse off than before - for we can enormously extend the record; yet even in its present bulk we can hardly consult it.

The prospect of complete records of an individual's everyday environment raises many deep issues associated with our assumptions and beliefs about with what authority past events can be described, yet the ease with which such recordings can be made with current technology would almost certainly lead to their widespread collection if they were in fact useful; on the whole they are not, because the process of actually locating any particular item of information in, for instance, a complete audio record of the past year, or week, or even hour, is so excruciatingly burdensome as to make it unthinkable except in the most critical of circumstances. This is Bush's problem of consultation, and the problem we consider in this thesis.

A number of recent projects have worked along these lines, an explicit attempt to meet Bush's vision. Early experiments in live transmission from body-worn cameras developed into independent wearable computers [48], but it was still several years before researchers could seriously propose comprehensive capture and storage portions of personal experience such as "MyLifeBits" [29].

In some early experiments of [14, 15] which focused on analyzing "ambulatory audio" to make environment classifications to support context sensitive applications. This work eventually led to a project in which a continuous waking-hours record was collected for 100 days with a special backpack, and then segmented and clustered into recurring locations and situations; rank-reduced features from the fish-eye video capture were most useful for this task.

Our work in segmenting and clustering based on recorded sound draws on work in audio segmentation. Early work on discriminating between speech and music in radio broadcasts [59] became important for excluding non-speech segments from speech recognizers intended to work with news broadcasts [61]. Since speech recognizers are able to 'adapt' their models to specific speakers, it was also important to segment speech into different speakers' turns and cluster the disjoint segments orig-

inating from the same speaker, by agglomerative clustering across likelihood ratios or measures such as the Bayesian Information Criterion (BIC), which is better able to compare likelihoods between models with differing numbers of parameters [11].

Other work in multimedia content analysis spans a number of projects to segment sound tracks into predefined classes such as speech, music, environmental sounds, and various possible mixtures [68]. Predefined classes allow model-based segmentation e.g. with hidden Markov models (HMMs), but local measures of segment dissimilarity permit segmentation even when no prior classes are assumed [36].

2.2 Speech and Music Detection

It has become clear that the richest and most informative content in these recordings is the speech, and thus it is important to be able to distinguish which segments of the sound contain speech via Voice Activity Detection (VAD). For example, dividing into speech and nonspeech allows both purer modeling of background ambience (for location recognition) and more focused processing of speech (for speaker identification, or for privacy protection by rendering detected speech unintelligible).

We would like to be able to identify segments in which anyone is speaking, and where possible to identify who is speaking - a process called 'diarization' in the speech recognition community [1]. Both of these functions - speech activity detection and speaker identification - are well established for telephony and broadcast audio, and have recently begun to be considered in the domain of meeting recordings [63, 24], which have some resemblance to personal audio. However, existing techniques are completely inadequate to deal with the bulk of our data because of the very high levels of background noise and/or reverberation.

Figure 2.1 shows a typical example of the kind of noisy signal we would like to be able to handle. This is from a belt-mounted recorder worn during a discussion

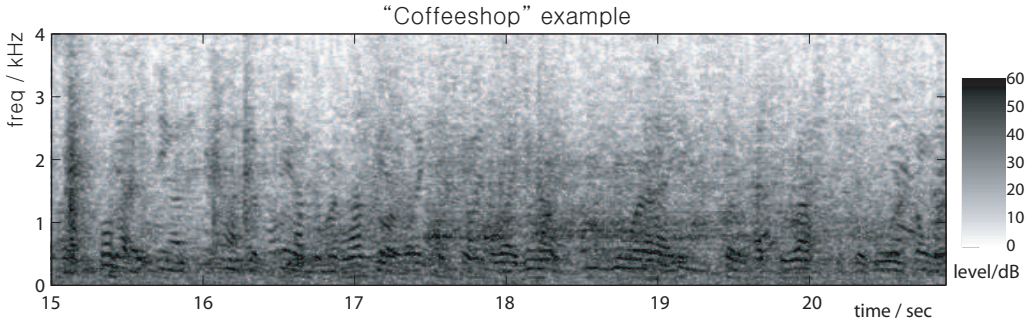

Figure 2.1: Spectrogram of example of noisy speech from a personal audio recording. Individual prominent syllables are visible as sets of parallel harmonics below 1 kHz, but this signal would be much too noisy for current speech activity detection or speaker identification techniques.

in a coffee shop. In this mono recording, several voices can be heard, but only for a word or two at a time - it is not possible to follow the conversation. It is, however, possible to identify the different speakers, given that they are familiar to the listener.

Speech activity detection has been addressed in telephony (where detected inactivity can be exploited to reduce bandwidth) and speech recognition systems (since a recognizer will often find 'words' in crosstalk or background noise, leading to insertion errors). In the telephony domain the standard approach amounts to an energy threshold: there is no effort to distinguish between voice and other energetic signals. Speech recognition systems designed to work with broadcast audio must take a richer view and be prepared to exclude sounds such as music and other effects that may nonetheless have significant energy. The most successful approaches employ classifiers similar to, or based upon, the acoustic models of the speech recognizer itself to decide which segments resemble speech and are thus likely to be appropriate to pass on to the recognition engine [65].

Prior work on soundtrack analysis has typically focused on detecting or distinguishing between a small number of high level categories such as speech, music, silence, noise, or applause. The application domain has most often relatively care-

fully produced sources, such as broadcast audio or movie soundtracks. Saunders [58] presented a speech/music Discrimination (SMD) based on simple features such as zero-crossing rate and short-time energy and a multivariate Gaussian classifier for use with radio broadcasts. This work reported an accuracy rate of 98% with 2.4 second segments. Scheirer et al. [59] tested 13 temporal and spectral features followed by a Gaussian Mixture Model (GMM) classifier, and reported an error rate of 1.4% in classifying 2.4 second segments from a database of randomly recorded radio broadcasts of speech and music. Williams et al. [65] approached SMD by estimating the posterior probability of around 50 phone classes on the same data, and achieved the same performance. Zhang et al. [68] proposed a system to segment and classify audio from movies or TV programs into more classes such as speech, music, song, environmental sound, speech with music background, environmental sound with music background, silence, etc. Energy, zero-crossing rate, pitch, and spectral peak tracks were used as features, and heuristic rule based classifier achieved an accuracy rate of more than 90%. Ajmera et al. [3] used entropy and dynamism features based on posterior probabilities of speech phonetic classes (as obtained at the output of an artificial neural network (ANN)), and developed a SMD based on a Hidden Markov Model (HMM) classification framework. Karneback [35] showed the best result with combining low-frequency modulation features and Mel-frequency Cepstral Coefficients(MFCCs). Thoshkahna et al. [6] had an accuracy of about 97% on the same data by using HILN (Harmonics, Individual Lines and Noise) model based features .

Neither of these approaches can be used for personal audio or the soundtrack of 'consumer video'which has many characteristics that distinguish it from the broadcast audio that has most commonly been considered in this work. Casual recordings made with small, hand-held cameras will very often contain a great deal of spuri-

ous, non-stationary noise such as babble, crowd, traffic, or handling artifacts. This unpredictable noise can have a great impact on detection algorithms, particularly if they rely on the global characteristics of the signal (e.g. the broad spectral shape encoded by MFCC features) which may now be dominated by noise. There is no consistent energy level for the kind of speech or music we want to be able to detect, and the highly variable background noise will often be as loud or louder than target speech or music. And because of the significant noise background, the features used for conventional acoustic classifiers (e.g. Mel Cepstra) will represent a hopelessly entangled mixture of aspects of the speech and the background interference: short of training a classifier on examples of speech in every possible background noise we anticipate, any conventional classifier will have very poor performance.

2.3 Generic Concept Detection

For less constrained environmental sounds, research has considered problems such as content-based retrieval, surveillance applications, or context-awareness in mobile devices. A popular framework is to segment, cluster, and classify environmental recordings into relatively simple concepts such as “animal”, “machine”,“walking”, “reading”, “meeting”, and “restaurant”, with testing performed on a few hours of data. Wold et al. [66] presented a content based audio retrieval system called ’Muscle Fish’. This work analyzed sounds in terms of perceptual aspects such as loudness, pitch, brightness, bandwidth and harmonicity, and adopted the nearest neighbor (NN) rule based on Mahalanobis distance measure to classify the query sound into one of predefined sound classes broadly categorized into animals, machines, musical instrument, speech and nature. Foote [27] proposed a music and sound effects retrieval system where 12 Mel-Frequency Cepstral Coefficients (MFCCs) plus energy were used as feature vectors. A tree-based vector quantizer (VQ) was applied on the

feature vector space to partition it into regions. Sounds were classified by calculating the Euclidean or cosine distances between the histograms of VQ codeword usage within each sound. Guo et al. [30] used SVM classifiers with perceptual and cepstral features on the 'Muscle Fish' data and roughly halved the errors in comparison to [66]. Malkin et al. [44] used linear autoencoding neural networks to achieve a lower error rate than a standard gaussian mixture model (GMM) for classifying environments such as restaurant, office, and outdoor. A linear combination of autoencoders and GMMs yielded still better performance. Ma et al. [43] considered the problem of classifying the acoustic environment on a portable device, for instance to provide a record of daily activities. They used MFCC features classified by an adapted speech recognition HMM to achieve over 90% accuracy distinguishing 3 second excerpts of 11 environments; humans averaged only 35% correct on the same data. Chu et al. [13] investigate acoustic context recognition for an autonomous robot. They compared nearest-neighbor (NN), GMM, and SVM classifiers with a wide range of features on a five-way classification task, and found best performance using the SVM and a subset of features selected by a greedy scheme.

The work most directly comparable to the proposed method is that by Eronen et al. [25]. Similar to [43], they investigated the classification of 24 contexts such as restaurant, office, street, kitchen with a view to applications in portable devices that could alter their behavior to best match an inferred situation. They compared a variety of features and classifiers, and achieved best performance with a simple approach of training a 5-component GMM on the MFCCs for each class, then classifying a test sample according to the GMM under which it achieves the highest likelihood. We take this as our baseline comparison system in the results below.

None of this prior work has directly addressed the classification of consumer videos by their soundtracks, and this domain raises a number of novel issues that

are addressed for the first time in this thesis. Firstly, we are dealing with the relatively large number of 25 concepts, comparable only to the 24 contexts in [25]; other systems used only between 2 and 12 concepts. Secondly, our concepts are drawn from a user study of photography consumers [9], and thus reflect actual types of queries that users would wish to make rather than simply the distinctions that we expect to be evident in the data. Thirdly, in all previous work there has been exactly one ground-truth label for each clip example (i.e. the data were exclusively arranged into a certain number of examples of each category). Consumer-relevant concepts cannot be so cleanly divided, and in our data most clips bear multiple labels, requiring a different approach to classification; our approach is inspired by similar work in music clip tagging, which has a similarly unpredictable number of relevant tags per item [47]. Finally, our data set is larger than any previously reported in environmental sounds, consisting of the soundtracks from 1873 distinct videos obtained from YouTube. These soundtracks are typically rather poor quality, often contain high levels of noise, and frequently have only sparse instances of "useful" (i.e. category-relevant) sounds. Thus, this is a much more demanding task than has been addressed in earlier work.

In addition to the novelty of the problem, the proposed method makes a number of specific technical contributions. Firstly we illustrate the viability of classifying, based only on soundtrack data, concepts like "beach" or "night" that on first sight seem unrelated to audio. Secondly, we show how to address the problem of overlapping concepts through the use of multiple, independent classifiers. Finally, we introduce a novel technique based on probabilistic Latent Semantic Analysis (pLSA) which outperforms our baseline Gaussian-SVM classifiers.

2.4 Markov Model-based Local Concept Detection

The concepts have diverse characteristics in terms of consistency, frequency and interrelationships. For example, concepts such as "music" or "crowd" typically persist over a large proportion (if not the entirety) of any clip to which they apply, and hence should be well represented in the global feature patterns – for example, the mean and covariance of per-frame features of a clip. However, the concept "cheer" manifests as a relatively small segment within a clip (at most a few seconds within a one-minute clip) which means that the global statistics of the clip may fail to distinguish it from others. In this paper, we address the problem of detecting such local patterns embedded in a global background soundtrack. In particular, we examine the case where we have training labels to indicate where examples of the concepts are present, but these labels are available only at the clip level (such as tags applied to YouTube videos), and therefore do not provide any more detailed information on the timing of the local events within the clip.

Multiple instance learning (MIL) has been successfully used to learn robust models from this kind of weak annotation across different levels of granularity. In MIL, each bag (e.g. an entire image or soundtrack) is a collection of instances (e.g. local feature vectors). Annotation is given at the bag level actually reflecting the label of one or more instances in that bag. If at least one instance is positive, the corresponding bag is labeled as positive. On the other hand, a bag is tagged as negative only when all instances in the bag are negative. The goal is to learn a set of instance points that are close to positive bags and simultaneously far away from negative bags. MIL, originally developed for applications in drug discovery [49], has been applied to content-based image retrieval, classification, and object detection [12, 52, 64], as well as music labeling [46].

2.5 Summary

In this chapter, we discuss the background information and prior works on segmenting and clustering of personal audio, detecting speech or music in an environmental sound, and generic or local concept detecting on consumer videos based on their soundtracks.

Chapter 3

Segmenting and Clustering

This chapter describes our approaches in segmenting and labeling personal audio archives - continuous recordings of an individual's everyday experiences - into 'episodes' (relatively consistent acoustic situations lasting a few minutes or more) using the Bayesian Information Criterion [11] and spectral clustering [53].

In the next section, we describe our processing of these recordings, considering the features appropriate for long-duration recordings, identifying segmentation points, and clustering and classifying the resulting segments. Evaluations and discussions on efforts at displaying and interacting with this data, and in integrating it with other 'scavenged' data such as online calendars are presented in Section 3.2 and 3.3 respectively. Finally, we summarize this in Section 3.4.

3.1 Features, segmenting and clustering algorithm

To ease the problem of locating and reviewing a particular event in a lengthy recording, we seek automatic means to generate a coarse index into the recording. At the broadest level, this index can divide a multi-hour recording into episodes consisting of, say, 5 minutes to an hour, during which statistical measures of the audio indi-

cate a consistent location or activity. By segmenting the recording at changes in an appropriate statistic, then clustering the resulting segments to identify similar or repeated circumstances, a user could identify and label all episodes of a single category (for instance, attending lectures by Professor X) with minimal effort. Below, we describe our approaches for extracting features, locating segmentation points, and clustering the resulting episodes.

3.1.1 Features

For the automatic diary application, temporal resolution on the order of one minute (for example) is plenty: most of the events we wish to identify are at least a quarter-hour long. We therefore construct a system where the temporal frame rate is greater than the 10 or 25 ms common in most audio recognition approaches. 25 ms is popular because even a dynamic signal like speech will have some stationary characteristics (e.g. pitch, formant frequencies) at that time scale. For characterizing acoustic environments, however, it is the stationary properties at a much longer timescale that concern us - the average level and degree of variation of energy at different frequency bands, measured over a window long enough to smooth out short-term fluctuations. Thus, we are interested in segmenting and classifying much longer segments, and not becoming distracted by momentary deviations.

We opted for a two-level feature scheme, with conventional short-time features (calculated over 25 ms windows) being summarized by statistics over a longer basic time-frame of up to 2 min. Long time-frames provide a more compact representation of long-duration recordings and also have the advantage that the properties of the background ambience may be better represented when transient foreground events are averaged out over a longer window.

3.1.1.1 Short-time features

Our data consists of single-channel recordings resampled to 16 kHz. All our features started with a conventional Fourier magnitude spectrum, calculated over 25 ms windows every 10 ms, but differed in how the 201 short-time Fourier transform (STFT) frequency values were combined together into a smaller per-time-frame feature vector, and in how the 6000 vectors per second were combined into a single feature describing each longer time-frame (e.g., from 0.25 seconds to 2 minutes).

We used several basic short-time feature vectors, each at two levels of detail.

- **Energy Spectrum**, formed by summing the STFT points across frequency in equal-sized blocks. The Energy Spectrum for time step n and frequency index j is:

$$A[n, j] = \sum_{k=0}^{N_F} w_{jk} X[n, k] \tag{3.1}$$

  where $X[n, k]$ are the squared-magnitudes from the N point STFT, $N_F = N/2 + 1$ is the number of non-redundant points in the STFT of a real signal, and the w_{jk} define a matrix of weights for combining the STFT samples into the more compact spectrum. To match the dimensionality of the auditory features below, we created two versions of the Energy Spectrum; the first combined the 201 STFT values into 21 Energy Spectrum bins (each covering about 380 Hz or about 10 STFT bins); the second Energy Spectrum had 42 bins (of about 190 Hz).

- **Auditory Spectrum**, similarly formed as weighted sums of the STFT points, but using windows that approximate the bandwidth of the ear - narrow at low frequencies, and broad at high frequencies - to obtain spectrum whose detail approximates, in some sense, the information perceived by listeners. We used the Bark axis, so a spacing of 1 Bark per band gave us 21 bins, and

0.5 Bark/band gave 42 bins. Each of these variants simply corresponds to a different matrix of w_{jk} in eqn. 3.1 above.

- **Entropy Spectrum**: The low-dimensional spectral features collapse multiple frequency bands into one value; the intuition here is that although the auditory bands are wide, the entropy value will distinguish between energy that is spread broadly across the whole band, versus one or two narrow energy peaks or sinusoids providing the bulk of the energy in the band. Humans are of course very sensitive to this distinction [34].

  We use entropy (treating the distribution of energy within the subband as a PDF) as a measure of the concentration (low entropy) or diffusion (high entropy) within each band, i.e. we define the short-time *entropy* spectrum at each time step n and each spectral channel j as:

$$H[n,j] = -\sum_{k=0}^{N_F} \frac{w_{jk}X[n,k]}{A[n,j]} \cdot log\left(\frac{w_{jk}X[n,k]}{A[n,j]}\right) \tag{3.2}$$

  where the the band magnitudes $A[n,j]$ from eqn. 3.1 serve to normalize the energy distribution within each weighted band to be PDF-like.

  The entropy can be calculated for the bins of both the Energy Spectrum and the Auditory Spectrum; for the Auditory Spectrum, since the w_{jk} define unequal width bands, it is convenient to normalize each channel by the theoretical maximum entropy (of a uniform initial spectrum X) to aid in visualizing the variation in entropy between bands.

- **Mel-frequency Cepstral Coefficients (MFCCs)** use a different (but similar) frequency warping, then apply a decorrelating cosine transform on the log magnitudes. We tried the first 21 bins, or all 40 bins from the implementation we used. MFCCs are the features most commonly used in speech recognition

and other acoustic classification tasks. (For the 'linear' averaging described below, we first exponentiated the cepstral values to obtain nonnegative features comparable to the spectral energies above).

3.1.1.2 Long-time features

To represent longer time frames of up to 2 minutes, we tried a number of statistics to combine the set of short-time feature vectors (calculated at 10-ms increments described above) into a single vector. We calculated the mean and standard deviation for each dimension before or after conversion to logarithmic units (dB), giving four summary vectors, μ_{lin}, σ_{lin} , μ_{dB}, σ_{dB} respectively, all finally expressed in dB units. We also calculate the average of the entropy measure μ_H, and the entropy deviation normalized by its mean value, σ_H/μ_H. Figure 3.1 illustrates each of these statistics, based on the Bark-scaled auditory spectrum, for 8 hours of audio recorded on one day.

- **Average Linear Energy**, μ_{lin} : The mean of the vector of energies for a long-time frame of data from each individual channel. This value is then converted to logarithmic units (dB).
- **Linear Energy Deviation**, σ_{lin} : The standard deviation of long-time's worth of each feature dimension, converted to dB.
- **Normalized Energy Deviation**, σ_{lin}/μ_{lin}: Energy Deviation divide by Average Energy, in linear units. If two temporal profiles differ only by a gain constant, this parameter is unchanged.
- **Average Log Energy**, μ_{dB} : We take the logarithm of the energy features first, then calculate the mean within each dimension over the full long-time frame.

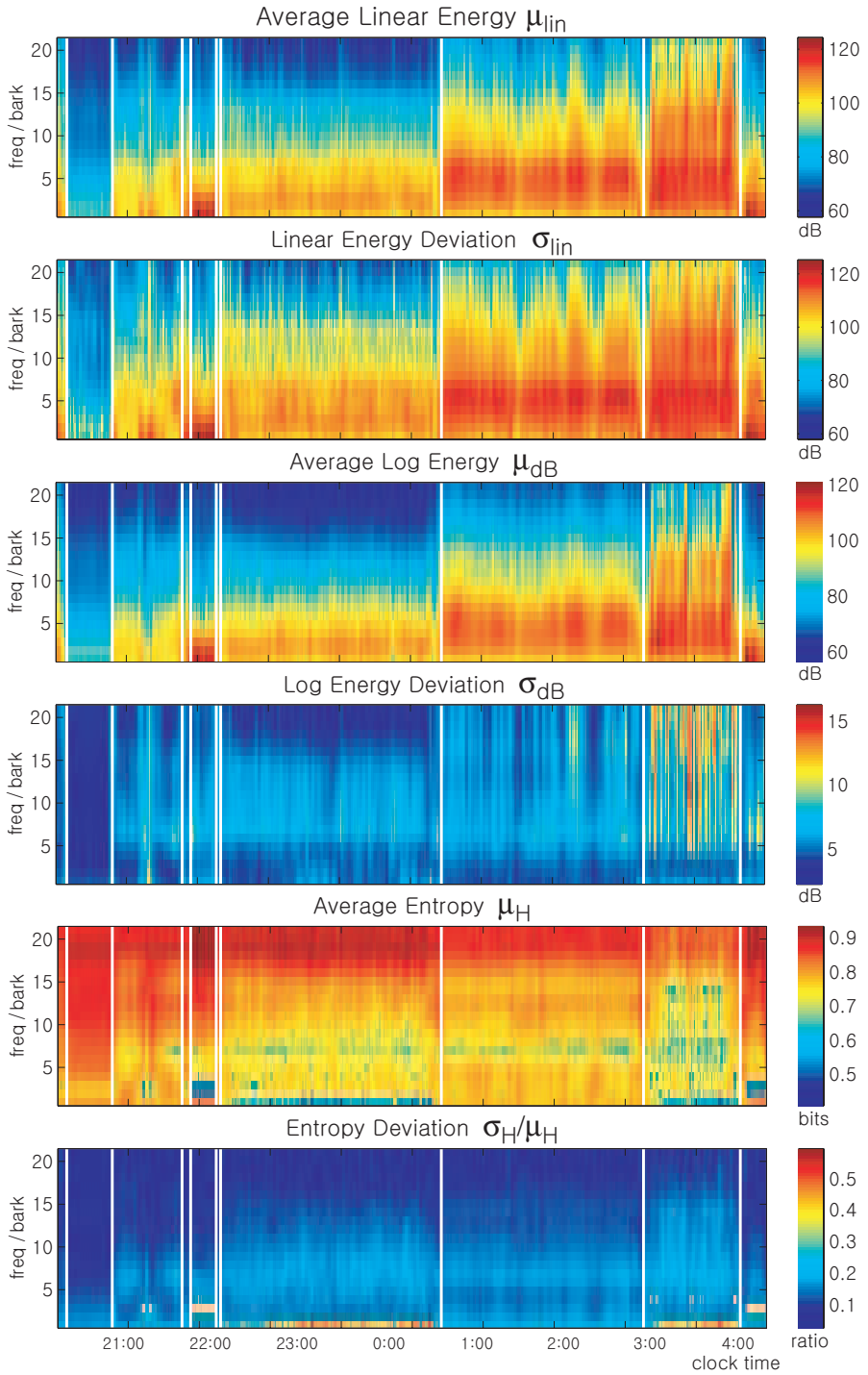

Figure 3.1: Examples of the six long-time-frame statistic features based on 21-band auditory (Bark-scaled) spectra. The underlying data is eight hours of recordings including a range of locations. White vertical lines show our hand-marked episode boundaries (see text).

- **Log Energy Deviation**, σ_{dB} : The standard deviation of the log-domain version of the feature values. In general, log-domain mean and deviation place excessive emphasis on very quiet sections (which can become arbitrarily negative in log units) but the relatively high noise background in this data avoided this problem. Note that this feature is already invariant to overall gain changes.

- **Average Entropy**, μ_H : We calculate the Average Entropy by taking the mean of each dimension of the Entropy Spectrum feature $H[n, j]$ of eqn. 3.2 over the diverse long-time window.

- **Entropy Deviation**, σ_H/μ_H : The standard deviation of $H[n, j]$ within each window, normalized by its average.

3.1.2 Unsupervised Segmentation

To segment an audio stream we must detect the time indices corresponding changes in the nature of the signal, in order to isolate segments that are acoustically homogeneous. One simple approach is to measure dissimilarity (e.g. as log likelihood ratio or KL divergence) between models derived from fixed-size time windows on each side of a candidate boundary. However, the fixed window size imposes both a lower limit on detected segment duration, and an upper bound on the accuracy with which the statistical properties of each segment can be measured, limiting robustness. In contrast, the Bayesian Information Criterion (BIC) provides a principled way to compare the likelihood performance of models with different numbers of parameters and explaining different amounts of data e.g. from unequal time windows. The speaker segmentation algorithm presented in [11] uses BIC to compare every possible segmentation of a window that is expanded until a valid boundary is found, so that the statistics are always based on complete segments.

The BIC is a likelihood criterion penalized by model complexity as measured by the number of model parameters. Let $\chi = \{x_i : i = 1, ..., N\}$ be the data set we are modeling and $M = \{m_i : i = 1, ..., K\}$ be the candidate models we wish to choose between. Let $\#(M_i)$ be the number of parameters in model M_i, and $L(\chi, M_i)$ be the total likelihood of χ under the optimal parameterization of M_i. The BIC is defined as:

$$BIC(M) = log(L(\chi, M)) - \frac{\lambda}{2}\#(M) \cdot log(N) \tag{3.3}$$

where λ is a weighting term for the model complexity penalty which should be 1 according to theory. By balancing the expected improvement in likelihood for more complex models by the penalty term, choosing the model with the highest BIC score is, by this measure, the most appropriate fit to the data.

The BIC-based segmentation procedure described in [11] proceeds as follows. We consider a sequence of d-dimensional audio feature vectors $\chi = \{x_i \in R^d : i = 1, ..., N\}$ covering a portion of the whole signal as independent draws from one or two multivariate Gaussian processes. Specifically, the null hypothesis is that the entire sequence is drawn from a single distribution:

$$H_0 : \{x_1, ..., x_N\} \sim N(\mu_0, \Sigma_0) \tag{3.4}$$

which is compared to the hypothesis that the first i points are drawn from one distribution and that the remaining points come from a different distribution, i.e. there is a segment boundary after sample t:

$$H_1 : \{x_1, ..., x_t\} \sim N(\mu_1, \Sigma_1), \;\; \{x_{t+1}, ..., x_N\} \sim N(\mu_2, \Sigma_2) \tag{3.5}$$

where $N(\mu, \Sigma)$ denotes a multivariate Gaussian distribution with mean vector μ and full covariance matrix Σ.

The difference in BIC scores between these two models is a function of the candidate boundary position t:

$$BIC(t) = log\left(\frac{L(\chi|H_0)}{L(\chi|H_1)}\right) - \frac{\lambda}{2}\left(d + \frac{d(d+1)}{2}\right) log(N) \tag{3.6}$$

where $L(\chi|H_0)$ is the likelihood of χ under hypothesis H_0 etc., and $d + d(d+1)/2$ is the number of extra parameters in the two-model hypothesis H_1. When $BIC(t) > 0$, we place a segment boundary at time t, and then begin searching again to the right of this boundary, and the search window size N is reset. If no candidate boundary t meets this criteria, the search window N is increased, and the search across all possible boundaries t is repeated. This continues until the end of the signal is reached.

The weighting parameter λ provides a 'sensitivity' control which can be adjusted to make the overall procedure generate a larger or smaller number of boundaries for a given signal.

3.1.3 Clustering

Given a data stream divided into self-consistent segments, an automatic diary application needs to make some kind of labeling or classification for each segment. These labels will not be meaningful without some kind of supervision (human input), but even without that information, the different sequential segments can be clustered together to find recurrences of particular environments - something which is very common in a continuous, daily archive. We performed unsupervised clustering on segments generated by the BIC segmentation scheme from the previous section to

identify the sets of segments that corresponded to similar situations, and which could therefore all be assigned a common label (which can be obtained from the user in a single interaction).

We used the spectral clustering algorithm [53]. First, a matrix is created consisting of the distance between each pair of segments. We use the symmetrized Kullback-Leibler (KL) divergence between single, diagonal-covariance Gaussian models fit to the feature frames within each segment. For Gaussians, the symmetrized KL divergence is given by:

$$D_{KLS}(i,j) = \frac{1}{2}\left((\mu_i - \mu_j)^T(\Sigma_i^{-1} + \Sigma_j^{-1})(\mu_i - \mu_j) - tr(\Sigma_i^{-1}\Sigma_j + \Sigma_j^{-1}\Sigma_i - 2I)\right) \tag{3.7}$$

where Σ_i is the unbiased estimate of the feature covariance within segment i, μ_i is the vector of per-dimension means for that segment, I is the identity matrix, and $tr()$ is the trace of a matrix. (Since some segments can be just a few frames long, we regularized our covariance estimates with a small empirically-optimized constant added to the leading diagonal.) D_KLS is zero when two segments have identical means and covariances, and progressively larger as the distributions become more distinct.

These distances are then converted to an 'affinity matrix' consisting of elements a_{ij} which are close to 1 for similar segments (that should be clustered together), and close to zero for segments with distinct characteristics. The a_{ij} is formed as a Gaussian-weighted distance i.e.

$$a_{ij} = exp\left(-\frac{1}{2}\frac{D_{KLS}(i,j)^2}{\sigma^2}\right) \tag{3.8}$$

where σ is a free parameter controlling the radius in the distance space over which points are considered similar; increasing σ leads to fewer, larger clusters. We tuned

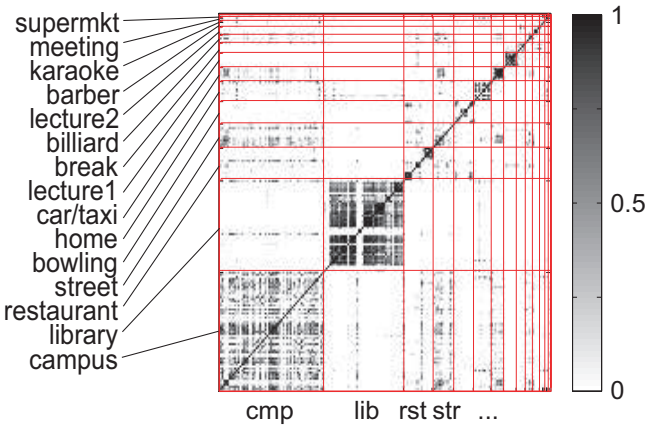

Figure 3.2: Affinity matrix for the 127 automatic segments. Segments are ordered according to the dominant ground-truth label in each segment.

it by hand to give reasonable results. Figure 3.2 shows the affinity matrix between the 127 automatic segments.

Clustering then consists in finding the eigenvectors of the affinity matrix, which are the vectors whose outer products with themselves, scaled by the corresponding eigenvalues, sum up to give the affinity matrix. When the affinity matrix indicates a clear clustering (most values close to zero or one), the eigenvectors will tend to have binary values, with each vector contributing a block on the diagonal of the reconstructed affinity matrix whose rows and columns have been reordered to make the similar segments adjacent; in the simplest case, the nonzero elements in each of the top eigenvectors indicate the dimensions belonging to each of the top clusters in the original data.

Assigning the data to K clusters can then be achieved by fitting K Gaussian components to the data using the standard EM estimation for Gaussian mixture models. This fit is performed on a set of K-dimensional points formed by the rows of the first K eigenvectors (taken as columns). Similar segments will have similar projections in this space - along each of the axes in the simplest case - and will cluster

together. The choice of K, the desired number of clusters, is always problematic: we considered each possible value of K up to some limit, then evaluated the quality of each resulting clustering using the BIC criterion introduced above, penalizing the overall likelihood achieved by describing the data with K Gaussians against the number of parameters involved.

3.2 Evaluations

We have experimented with several different short-time features and several different statistics, and compared them empirically for their ability to support segmentation and clustering of our 'episodes'.

3.2.1 The data of personal audio recordings

Evaluating and developing our techniques required test data including ground truth for segmentation points and episode categories. We used a "Neuros" personal audio computer [16], which has a built-in microphone, a 20G hard disk, and battery power sufficient to record for over 8 hours without recharging. By carrying this device on a belt hook for a week, we collected a database of more than 62 hours. This single channel data was originally recorded as a 64 Mbps MPEG-Audio Layer 3 file, then downsampled to 16 kHz.

We manually annotated some 62 h of 9 audio recorded over 8 successive days (by author KL), marking boundaries wherever there was a clear change in environment we were hoping detect, e.g. entering and leaving buildings and vehicles etc. This resulted in 139 segments (average duration 27 min) which we assigned to 16 broad classes such as 'street', 'restaurant', 'class', 'library' etc. We note the risk of experimenter bias here, since the labeling was performed by the researchers who were already aware of the kinds of distinctions that would be possible or impossible

Labels	Minutes	Segments	Avg. duration
Library	981	27	36.3
Campus	750	56	13.4
Restaurant	560	5	112.0
Bowling	244	2	122.0
Lecturer 1	234	4	58.5
Car/Taxi	165	7	23.6
Street	162	16	10.1
Billiards	157	1	157.0
Lecturer 2	157	2	78.5
Home	138	9	15.3
Karaoke	65	1	65.0
Class break	56	4	14.0
Barber	31	1	31.0
Meeting	25	1	25.0
Subway	15	1	15.0
Supermarket	13	2	6.5
total	3753	139	27.0

Table 3.1: Segment classes, counts, and average duration (in minutes) from the manual annotation of the 62 hour test set.

for the system. Thus, although our results may be optimistic for this reason, we believe they are still indicative of the viability of these approaches. Table 3.1 lists the different segment classes identified in the data along with the number of such segments and their average durations.

3.2.2 Features and Segmentation Results

Our six short-time spectral representations - linear frequency, auditory spectrum, and Mel cepstra, each at either 21 or 42 (40 for MFCC) elements per frame - summarized by our seven "long-time" feature summarization functions gave us 38 different compact representations of our 62 hour dataset. (The entropy-based measures were not calculated for the cepstral features, since the concept of 'subband' does not apply in that case.) The BIC segmentation was applied to each version, and the λ

	Long-time features						
Short-time features	μ_{lin}	σ_{lin}	$\frac{\sigma_{lin}}{\mu_{lin}}$	μ_{dB}	σ_{dB}	μ_H	$\frac{\sigma_H}{\mu_H}$
21-bin Energy Spectrum	0.723	0.676	0.386	0.355	0.522	0.734	0.744
42-bin Energy Spectrum	0.711	0.654	0.342	0.368	0.505	0.775	0.752
21-bin Auditory Spectrum	0.766	0.738	0.487	**0.808**	0.591	**0.811**	**0.816**
42-bin Auditory Spectrum	0.761	0.731	0.423	0.792	0.583	0.800	**0.816**
21-bin MFCC	0.734	0.736	0.549	0.145	0.731	N/A	N/A
40-bin MFCC	0.714	0.640	0.498	0.166	0.699	N/A	N/A

Table 3.2: Sensitivity @ Specificity = 0.98 for each feature set. Values greater than 0.8 are shown in bold.

parameter was varied to control the trade-off between finding too many boundaries (false alarms in the boundary detection task) and too few boundaries (false rejection of genuine boundaries). A boundary placed within 3 min of the ground-truth position was judged correct, otherwise it was a false alarm, as were boundaries beyond the first near to a ground-truth event.

Table 3.2 shows the Sensitivities (Correct Accept rate, the probability of marking a frame as a boundary given that it is a true boundary) of each system when λ is adjusted and the results interpolated to achieve a Specificity of 98% (the probability of marking a frame as a non-boundary given that it is not a boundary, or equivalently a False Alarm rate of 2%).

There is a wide range of performance among the different features; mean and deviation of the linear energy perform quite well across all underlying representations, and their ratio does not. Log-domain averaging performs very well for the auditory spectrum but not for the other representations, and log-domain deviation is most useful for MFCCs. However, the spectral entropy features, describing the sparsity of the spectrum within each subband give the best overall performance, particularly when based on the auditory spectra.

Long-time Feature Set	Sensitivity
μ_{dB}	0.808
μ_H	0.811
σ_H/μ_H	0.816
$\mu_{dB} + \mu_H$	0.816
$\mu_{dB} + \sigma_H/\mu_H$	0.840
$\mu_H + \sigma_H/\mu_H$	0.828
$\mu_{dB} + \mu_H + \sigma_H/\mu_H$	0.836

Table 3.3: Sensitivity @ Specificity = 0.98 for different combinations of the three best statistics based on the 21- bin Auditory Spectrum.

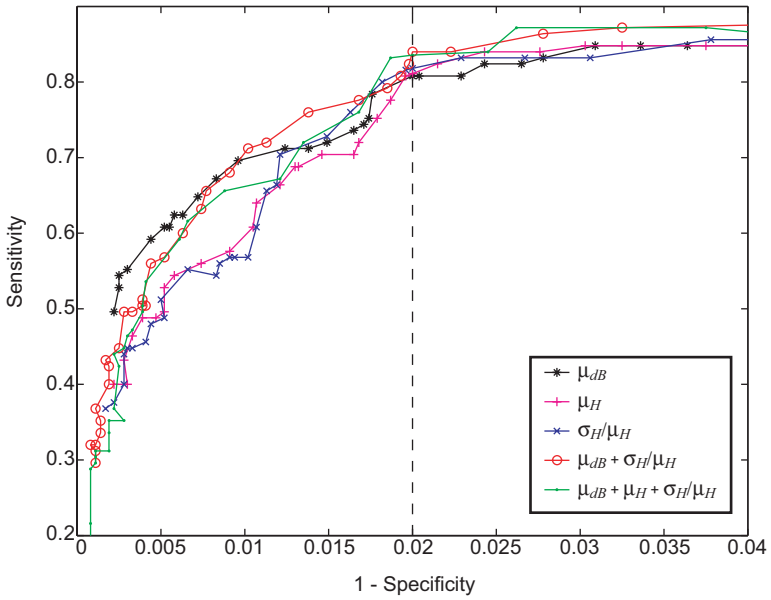

Figure 3.3: ROC curves for segment boundary detection based on different summaries and combinations of the 21 bin Auditory Spectral features.

Since the 21 bin Auditory Spectrum were the best underlying short-term features, our remaining results use only this basis. We experimented with using combinations of the best individual feature sets, to perform BIC segmentation on a higher-dimensional feature formed by simple concatenation. Table 3.3 shows the results of all possible combinations of the three best features, Average Log Energy

μ_{dB}, Average Entropy μ_H, and Entropy Deviation σ_H/μ_H. Although all the combinations yield broadly similar results, our best combination involves just two of the three features, namely the Average Log Energy plus the Entropy Deviation.

Figure 3.3 shows the Receiver Operating Characteristic (ROC) curve for our best performing detectors, illustrating the trade-off between false alarms and false rejects as the BIC penalty weight λ is varied. (A better performance lies closer to the top-left corner, and random guessing follows the leading diagonal). We see that the $\mu_{dB} + \sigma_H/\mu_H$ combination is the best overall, although the differences from the best individual feature sets are quite small.

Our feature vectors are relatively large, particularly when feature combinations are used. We are currently pursuing rank-reduction of the features using Principal Component Analysis (PCA) prior to the BIC segmentation. In an initial investigation, we obtained a Sensitivity of 0.874 (at Specificity = 0.98) for a combination of the first 3 principal components of μ_{dB} combined with the first 4 principal components of μ_H (which proved more useful than σ_H/μ_H in this case).

3.2.3 Clustering Results

Our best segmentation scheme produced 127 automatically-generated segments for our 62 h data set. Spectral clustering (using the same average spectrum features as used for segmentation) then arranged these into 15 clusters. We evaluated these clusters by comparing them against the 16 labels used to describe the 139 ground-truth segments. Since there is no a priori association between the automatically-generated segments and the hand-labeled ones, we chose this association to equate the most similar clusters in each set, subject to the constraint of a one-to-one mapping. This resulted in one ground-truth class ("street") with no associated automatic cluster,

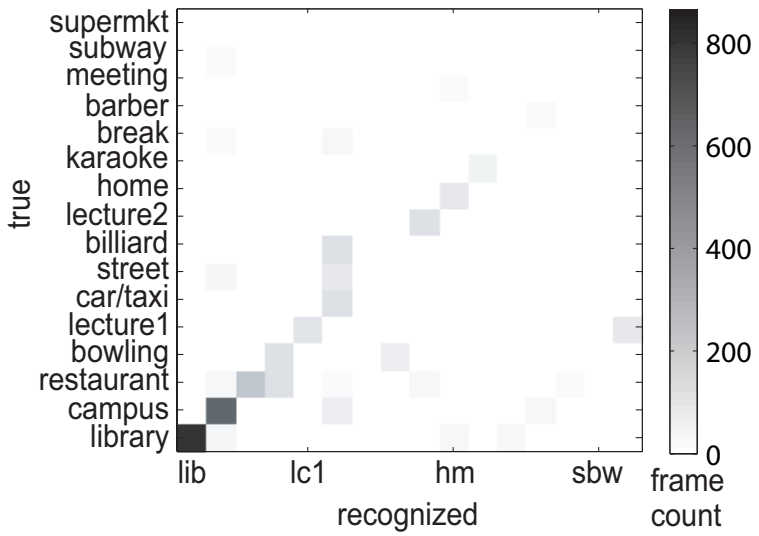

Figure 3.4: Confusion matrix for the sixteen segment class labels, calculated over the 3753 one-minute frames in the test data.

and five more ("billiards", "class break", "meeting", "subway", and "supermkt") for which no frames were correctly labeled, meaning the correspondences are arbitrary.

Since the automatic and ground-truth boundaries will not correspond, we evaluate the clustering at the frame level i.e. for each 1 min time-frame, the ground-truth and automatic labels were combined. Overall, the labeling accuracy at the frame level was 67.3% (which is also equal to the weighted average precision and recall, since the total number of frames is constant). Figure 3.4 shows an overall confusion matrix for the labels.

For comparison, direct clustering of one-minute frames without any prior clustering, and using an affinity based on the similarity of feature statistic distributions among 1s subwindows, gave a labeling accuracy of 42.7% - better than the a priori baseline of guessing all frames as a single class (26.1%), but far worse than our

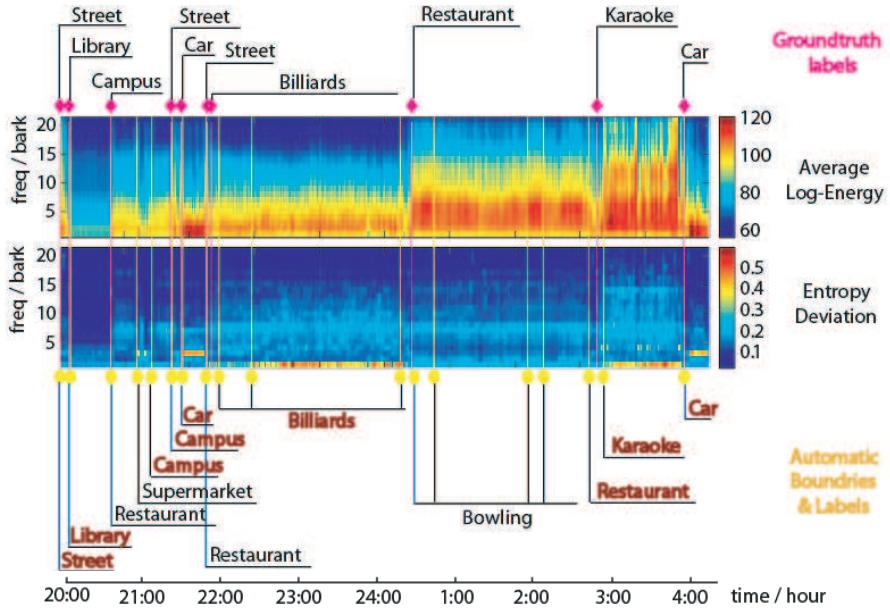

Figure 3.5: Example of segmenting and clustering 8-hour personal audio clip base on the Average Log Energy and Entropy Deviation features.

segmentation based approach.

Figure 3.5 shows the example of segmenting and clustering 8-hour personal audio clip using both Average Log Energy and Entropy Deviation features. Segmentation errors exist within transient regions between places, e.g., campus. If same place has variable background ambience in time, for example, restaurant when is busy or not in time, there are some errors in clustering results.

3.2.4 Varying the time-frame

The results above are based on 60 s windows, our arbitrary initial choice motivated by the granularity of the task. Returning to this parameter, we ran the entire system (both segmentation and clustering) for time-frames varying from 0.25 s to 120 s to see how this affected performance, holding other system parameters constant.

Figure 3.6 shows the overall frame accuracy of the clustering as a function of

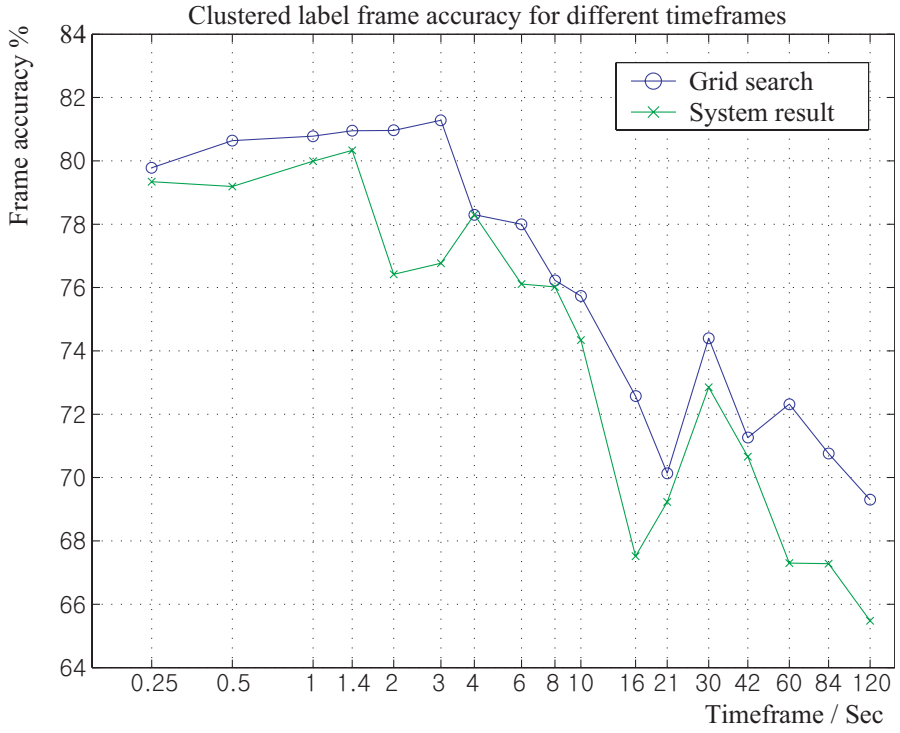

Figure 3.6: Effect on labeling frame accuracy of varying the basic time-frame duration.

time-frame length. The lower trace gives the system results, showing variation from 65% to over 80% frame accuracy, with the best results achieved at the shortest time frames, and significant degradation for time-frames above 10 s. The upper trace shows the best result from Frame an exhaustive grid search over the clustering parameters K and σ, giving an upper bound in performance. We see that 3 s is the time-frame with the best performance - arguably still long enough to capture background ambience statistics by averaging over foreground transients, but much shorter than (and distinctly superior to) the 60 s window we had used thus far.

We also experimented with basing the clustering on different features, which of course need not be the same as those used in segmentation. The results above are based on the 21-dimensional log-domain average auditory spectrum μ_{dB}, which

achieved a 76.8% frame-level labeling accuracy with the 3 s window. Using the normalized entropy deviation, σ_H/μ_H increased this to 82.5%, and combining both features with the mean entropy achieved the best result of 82.8%.

Note, however, that we have not reported the segmentation performance - shorter time frames gave many more inserted segmentation points, which did not, however, impact labeling accuracy since the resulting short segments were still correctly clustered on the whole. For the indexing application, however, excess segment boundaries are a problem, so labeling frame accuracy is not the only metric to consider. Larger numbers of segments also severely impact the running time of spectral clustering, which is based on the eigen-solution of an $N \times N$ affinity matrix.

3.3 Discussion

3.3.1 Visualization and browsing

We have developed a prototype browsing interface, shown in Figure 3.7. A day-by-day pseudo-spectrogram visualization of the audio, where each pixel's intensity reflects the average log energy, the saturation (vividness of the color) depends on the mean spectral entropy, and the hue (color) depends on the entropy deviation, lies alongside the automatically-derived segments and cluster labels, as well as the user's calendar items.

Audio can be reviewed by clicking on the spectrogram, along with the usual fast forward/rewind transport controls. Our informal experiences with this interface have been mixed. It greatly facilitates finding particular events in a recording compared to the timeline slider provided by a basic media player. However, the interface has an effective resolution no better than a minute or two, and having to listen through even this much audio to reach the desired moment is still painful and

boring, and would benefit from the addition of time-scaling techniques for faster review.

3.3.2 Scavenging other data sources

Given the minimal impact of collecting audio archives, we have looked for other data sources to exploit. Since users are resistant to changing their work patterns, including the software they use, our goal was to find existing information streams that could be 'scavenged' to provide additional data for a personal history/diary. The basic framework of a time-line provided by the audio recordings can be augmented by annotations derived from any time-stamped event record. This is the idea of "chronology as a storage model" proposed in Lifestreams [28] as a method of organizing documents that exploits human cognitive strengths. While our interest here is more in recalling the moment rather than retrieving documents, the activities are closely related.

Some of the time-stamped data we have identified includes:

- **Online calendars** : Many users keep maintain their calendars on their computers, and this data can usually be extracted. The calendar is of course the most familiar interface for accessing and browsing time-structured data extending over long periods, and forms the basis of our preliminary user interface.

- **E-mail logs** : E-mail interaction typically involves a large amount of time-stamped information. We have extracted all the dates from a user's sent messages store to build a profile of when (and to whom) email messages were being composed.

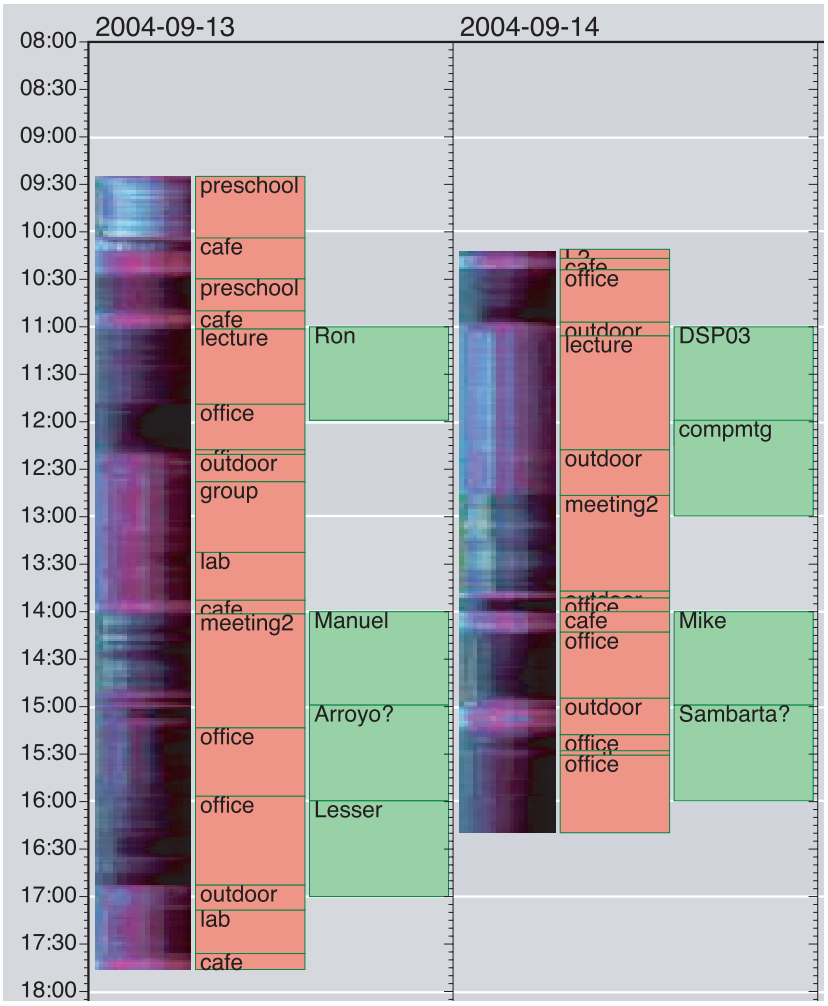

Figure 3.7: Screenshot from our experimental browser. Recorded audio is shown by a pseudocolor spectrogram with a vertical time axis. Next to this are the automatically derived segments along with their per-cluster manual labels. The display also shows appointments read from the user's online calendar - a useful prompt in navigating the recordings and interpreting the automatic segments.

- **Other computer interactions** : There are many other activities at the computer keyboard than can lead to useful time logs. For instance, web browser histories constitute a rich, easily reconstituted, record of the information seen by the user. As a more specific example, the popular outliner NoteTaker [4] is frequently used for real-time note taking, and records a datestamp (down to one-second resolution) for each line entered into the outline. Dense note-taking activity can thus be extracted and presented on the calendar interface, along with the titles of pages being modified, effortlessly providing a topic description. Moreover, instead of using the time-line to organize outline entries, the outline itself - a hierarchic structuring of information created by the user - can also be used, when available, as an alternative interface to the recorded archive. Replaying the recording from immediately prior to a particular entry being made in the outline would be a very useful enhancement to written notes of talks and lectures, along the lines of the Audio Notebook [62] - but without requiring users to acquire special hardware or change their current practice, and involving only a small amount of additional software to link the existing records.

- **GPS TrackLogs** : Inexpensive personal Global Positioning System (GPS) receivers can keep a log of their positions, synchronized to a highly-accurate clock, and with minimal impact to the user as long as the device is carried (and periodically uploaded). We initially investigated this as a way to collect ground-truth tags for the segmentation experiments described in section 3.1, but since GPS does not work indoors (and only intermittently on the streets of built-up cities), it was not so useful. None the less, when available, GPS information giving exact location as well as motion data can provide a rich input to an automatic diary.

- **Phone records** : Phone companies typically provide detailed logs of every phone call placed (as well as calls received on mobile phones), and this information is usually available in electronic form via the web; such data can be parsed and included in the timeline view.

- **Digital photos** : Cheap digital cameras have by now almost completely eliminated analog formats, at least for casual photographers. Since these pictures are usually datestamped and uploaded to the user's personal computer, the information about when pictures were taken - and thumbnails of the images themselves - can be added to the timeline.

The common theme, in addition to the temporally-based indexing, is that each of these data streams already exists and requires only minimal additional processing to be incorporated. By the same token, since the data is being opportunistically scavenged rather than carefully collected expressly for the diary, it may offer unreliable, partial coverage; users will take photographs or make phone calls only sporadically. Even in our focused e.orts to collect baseline audio archives, the recorders will be used only for a few hours each day, and certain files may become corrupted or lost. These are realities of personal information, and practical applications and user interfaces should be built to accommodate them, for instance by o.ering multiple, partially-redundant data streams, rather than being useful only when everything 'works as planned'.

3.3.3 Speech and privacy

Initially, our interest was in the non-speech background ambience in the audio signals as we consider this a neglected topic in audio analysis. However, it has become clear that the speech content is the richest and most engaging information in our

recordings . both for information and 'reminiscence' purposes. To this end, we are developing a robust speech detector that we intend to be able to identify fragments of speech amid noisy and reverberant backgrounds as encountered in our data. Dividing into speech and non-speech segments allows both 'purer' modeling of background ambience (for location recognition) as well as more focused processing of speech. Identifying interactions with particular speakers would be useful for access, as, of course, would recognizing the spoken content - e.g. by making use of the techniques being developed for meeting transcription [55].

This, however, brings us squarely into the domain of privacy concerns. This project readily arouses resistance and suspicion from acquaintances who find the idea of recording conversations threatening and creepy. We must address such concerns before an application of this kind can become widely accepted and useful. While segmentation requires only the long-time-frame statistics (which do not contain sufficient information for resynthesis to audio), much of the usefulness of the data is lost unless users have the ability to listen to the original audio. Sufficiently accurate speaker identification could enable the retention of intelligible utterances only if the speaker has given explicit permission, along the lines of the "revelation rules" in the location-tracking system of Lamming and Flynn [37]. If recorders become more pervasive, they could be made to respect an "opt-out" (or opt-in) beacon along the lines of [7].

We are also looking at ways of securing the recordings against unauthorized access. An intriguing technique for co-operative computing breaks the data into two individually-useless parts (e.g. by adding and subtracting the same random sequence to the original waveform) which are distributed to two agents or locations, then permits computation of derived features (such as our time-frame statistics) without either party having access to the full data [18].

3.4 Summary

We have described a vision of personal audio archives and presented our work on providing automatic indexing based on the statistics of frequency-warped short-time energy spectra calculated over windows of seconds or minutes. Our automatically clustered segments can be grouped into similar or recurring classes which, once the unknown correspondence between automatic and ground-truth labels is resolved, gives frame-level accuracies of over 80% on our 62 h hand-labeled test set.

Ubiquitous, continuous recordings seem bound to become a part of our arsenal of personal records as soon as the retrieval and privacy issues are tackled, since, for audio-only recordings, the collection technology is already quite mature. While the most compelling applications for this data remain to be clarified, we are intrigued and encouraged by our investigations so far.

Chapter 4

Speech and Music Detection

In this chapter, we presents a novel method for identifying regions of speech or music in the kinds of energetic and highly-variable noise present in a real-world sound collected by body-worn recorders. Motivated by psychoacoustic evidence that pitch is crucial in the perception and organization of sound, we develop a noise-robust pitch detection algorithm to locate speech or music-like regions. To avoid false alarms resulting from background noise with strong periodic components (such as air-conditioning), we add a new scheme to suppress these noises in the domain of autocorrelogram.

In the next section, we describe our Voice Activity Detection (VAD) algorithms to identify the presence of speech. In Section 4.2, our proposed music detection features for detecting the stable periodicities of musical pitch are presented. Evaluations and discussions are presented in section 4.3 and 4.4 respectively. Finally, we summarize this in section 4.5.

4.1 Voice Activity Detection in Personal Audio Recordings Using Autocorrelogram Compensation

To detect regions of speech in this kind of high-noise, high-variability sound, we draw inspiration from the particular sensitivity of listeners to pitch, and to its dynamics. The first few harmonics of pseudoperiodic vowels have the greatest energy of any part of a speech signal, and thus are the most likely to be detectible in poor signal-to-noise ratios (SNRs). Also, the redundancy of multiple harmonics derived from a single underlying periodicity gives rise to robust coding of the fundamental frequency for more accurate detection in noise. As a result, our approach is based on a class of noise-robust Pitch Detection Algorithms (PDAs) that perform nonlinear combination of periodicity information in different spectral regions to best exploit locally-favorable SNRs, and can thus identify periodicity present across the entire spectrum even when the evidence in any single frequency channel is weak [19].

However, to use such PDAs to detect speech implicitly assumes that any periodicity present in the signal corresponds to voice. When the signal contains interference that is itself periodic – such as the steady hum of an air-conditioning unit, which is particularly common in some of our outdoor recordings – this approach to VAD raises many false alarms. In figure 4.1 (b), there is a fair number of obviously erroneous nonspeech pitches, as well as distortions of the voiced pitches, due to air-conditioning noise. Even multi-pitch trackers (like [67]) cannot separate such noise because voiced pitches are often weaker and/or intertwined (or overlapped) with non-voice, interfering pitch. Moreover, because these noises sometimes have higher spectral energy than speech, conventional spectral subtraction methods fail to estimate the correct local noise model for them and are thus unable to effectively eliminate them in the domain of spectral energy, as seen in figure 4.1 (c).

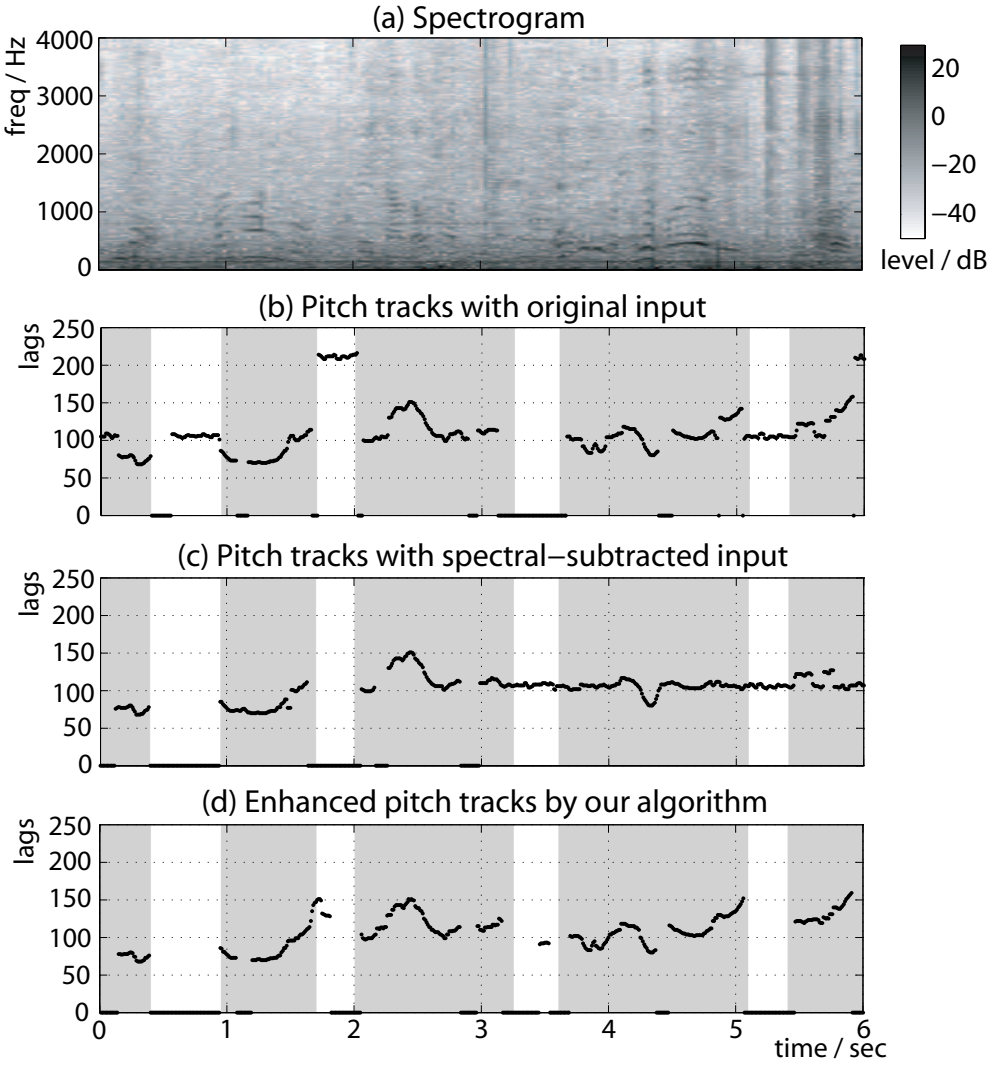

Figure 4.1: Example of noisy speech from a personal audio recording. The pitch tracks in (b) and (c) are extracted by a noise-robust PDA as described in the text; pane (d) shows the result of our algorithm with the same input signal. The pitch of a stationary periodic air-conditioning noise appears as flat contours around lags 105 and 210 in (b), and tends to be more dominant around 4-6 s in (c) due to the failure of a noise estimation of the spectral subtraction, but is clearly deleted by our method in (d). Shadowed regions indicate manually-labeled voiced segments.

In the next section, we describe a new method to remove long-time stationary periodic noises in the domain of autocorrelogram seen in figure 4.1 (d). Based on the fact that the autocorrelation function (ACF) of these noises has a more slowly-changing shape compared to speech over long durations, subbands corrupted with such noise can be excluded from the summary autocorrelation (SAC) by estimating whether the current ACF and the local average ACF are similar.

4.1.1 Noise-robust Voiced Pitch Detection

Our system is based on a noise-robust PDA [67] that estimates dominant periodicities from an SAC formed by summing the normalized short-time ACFs of multiple subbands (based on a perceptual model filterbank). Critically, ACFs are excluded from the SAC if they appear to be dominated by aperiodic noise, so the SAC describes the periodicities present only in relatively noise-free portions of the spectrum, chosen frame by frame. Specifically, the SAC is built from only those subbands whose normalized ACF has a peak above 0.945, where a peak of 1.0 would corresponds to a perfectly periodic signal, and added noise reduces this value (this threshold was established empirically in [67]). Finally an HMM is used to extract the most probable pitch track from the SAC.

As described below, our modification is to further exclude channels in which similarity between the current ACF and its average over a longer time window exceeds a threshold automatically adapted to differentiate between dynamic periodic signals such as voiced speech, and stationary periodic noises like air-conditioning. A simplified block diagram of our system is illustrated in figure 4.2.

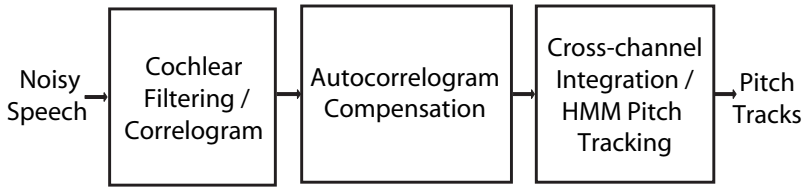

Figure 4.2: Block diagram of our proposed system.

4.1.1.1 Multichannel Autocorrelogram

Single-channel (mono) input recordings are resampled to 16 kHz, and then passed through a bank of 55 gammatone filters uniformly spaced on an ERB scale. We used the channels spanning 80 Hz to 800 Hz to capture the strongest pitched-voice energy. Then, the envelope is calculated by half-wave rectifying these outputs.

The ACF $r_{yy}(c, n, \tau)$ and its energy $e_{yy}(c, n, \tau)$ for each subband envelope output $y(c, n)$ at a given frequency channel c and time index n may be defined as:

$$r_{yy}(c, n, \tau) = \sum_{i=n+1}^{n+W} y(c, i)y(c, i + \tau) \tag{4.1}$$

$$e_{yy}(c, n, \tau) = \sqrt{\sum_{i=n+1}^{n+W} y^2(c, i) \sum_{i=n+1}^{n+W} y^2(c, i + \tau)} \tag{4.2}$$

where W is an integration window size, and $r_{yy}(c, n, \tau)$ and $e_{yy}(c, n, \tau)$ are calculated over 25 ms windows every 10 ms for lag $\tau = 0 \ldots 400$ samples (i.e. up to 25 ms for a lowest pitch of 40 Hz). $r_{yy}(c, n, \tau)$ has a large value when $y(c, n)$ is similar to $y(c, n+\tau)$, i.e. if $y(c, n)$ has a period of P, then $r_{yy}(c, n, \tau)$ has peaks at $\tau = lP$ where l is an integer. The normalized ACF $r_{yy}(c, n, \tau)/e_{yy}(c, n, \tau)$ always falls between 0 and 1 (for our nonnegative envelopes), and thus a value of 1 at nonzero lag implies perfect repetition of a signal periodic within the window. To simplify notation, variables c, n, and τ are henceforth dropped.

4.1.1.2 Autocorrelogram Compensation

Let us assume that noisy speech y consists of a clean voiced signal s and stationary periodic noise n i.e. $y(c,n) = s(c,n) + n(c,n)$. In this case, the ACF given by:

$$r_{yy} = r_{ss} + 2r_{sn} + r_{nn} \tag{4.3}$$

For large W, if we assume that $n(c,n)$ is zero mean and uncorrelated with $s(c,n)$, so $r_{sn} = 0$ i.e. $r_{yy} = r_{ss} + r_{nn}$. Taking the expected value of both sides gives:

$$E\{r_{yy}\} = E\{r_{ss}\} + E\{r_{nn}\} \tag{4.4}$$

Given an estimate of the autocorrelation of the noise $\hat{r}_{nn}$, we could derive an estimate of the uncorrupt speech signal as:

$$\hat{r}_{ss} = r_{yy} - \hat{r}_{nn} \tag{4.5}$$

4.1.1.3 Linear compensation

Theoretically, the ACF of a stationary periodic noise r_{nn} could be estimated during periods when the speech is inactive and then subtracted (or cancelled) from the ACF of the current frame r_{yy} resulting in the ACF of the clean speech $\hat{r}_{ss}$. However, there is no simple way to detect pure-noise segments in a highly noisy signal. Instead, we introduce a new method based on our assumption, supported by observation, that r_{nn} for the kinds of noise we are trying to remove changes very little with time. Consequently, the long-time average of the ACF r_{yy} tends to be close to r_{nn}. Thus, we can attempt to estimate the autocorrelation of the less stationary voice signal by, for each time frame and each channel, estimating $\hat{r}_{nn}$ as the average ACF over

M adjacent frames $avg\{r_{yy}\}$, and then subtracting it from r_{yy}:

$$\hat{r}_{ss} = max(0, r_{yy} - avg\{r_{yy}\}) \tag{4.6}$$

where $max()$ ensures that the estimated ACF cannot be negative.

Compared with the original SAC, the stationary periodic noise is effectively suppressed in a linear-compensated SAC, as shown in figure 4.3 (b), but at the cost of some speech information, particularly at lags below 100 samples. The basic assumption on this linear compensation is that the expected (average) value of r_{ss} in equation 4.4 is zero. However, since autocorrelations of bandlimited signals will always be positive in the vicinity of zero lag, r_{ss} does not have a zero-mean distribution, and $avg\{r_{yy}\}$ does not provide an unbiased estimate of r_{nn} for these lags. As a result, even with a large averaging window (e.g. 10 s), our estimate of the noise ACF is greater than the actual value of the distortion at these lags, and thus some speech information is removed by the compensation.

4.1.1.4 Non-linear compensation

To avoid the noise over-estimation problems of linear compensation, for each time frame and each channel, we compare every r_{yy} to $avg\{r_{yy}\}$ by cosine similarity, and use this to make a hard decision to include or exclude that ACF from the SAC. If the similarity is greater than a threshold θ_1, the subband is considered noisy for that frame, and is thus excluded from contributing to the SAC.

$$k = Sim_{cos}(r_{yy}, avg\{r_{yy}\}) \tag{4.7}$$

$$\hat{r}_{ss} = \begin{cases} r_{yy} & \text{if } k \leq \theta_1 \\ 0 & \text{otherwise} \end{cases} \tag{4.8}$$

where $Sim_{cos}()$ is the cosine similarity (dot product divided by both magnitudes) between the two ACF vectors.

θ_1 is automatically tuned based on voice pitch dynamics and harmonic spacing. Changes in target pitch cause r_{ss} to be smoothly varying along time, making r_{yy} differ from $avg\{r_{yy}\}$. Channels containing clean speech will thus exhibit local-minima in similarity k compared to their noise-dominated neighbors. Since voiced speech spectra will have equidistant harmonics with noise energy in-between [57], during speech segments, we may see clean voiced ACFs with noisy ACFs between them. If speech is corrupted by stationary, periodic noise, ACFs dominated by this noise are likely persistent in some channels over long time frames. Therefore, θ_1 is chosen as the mean of a set of cosine similarity values of entire channels over M frames. Decreasing the value of M makes it easier to identify periodic noise with shorter duration (or some variability), but risks making gross errors of mistaking speech with small pitch variation as background noise. A value of $M = 100$ (e.g. 1 s window) is a good compromise between robustness and the ability to catch short-duration stationary harmonic noises.

After excluding the frequency bands judged to be dominated by periodic noise, the SAC is calculated based only on channels with a strong peak in the normalized ACF that exceeds a second threshold θ_2 (e.g. 0.945). θ_2 is chosen by examining the statistics from sample utterances mixed with interference [67]. Thus, the selected normalized ACF R_{yy} for every frame and channel is given by:

$$R_{yy} = \begin{cases} \hat{r}_{ss}/e_{yy} & \text{if } \hat{r}_{ss}/e_{yy} \geq \theta_2 \\ 0 & \text{otherwise} \end{cases} \tag{4.9}$$

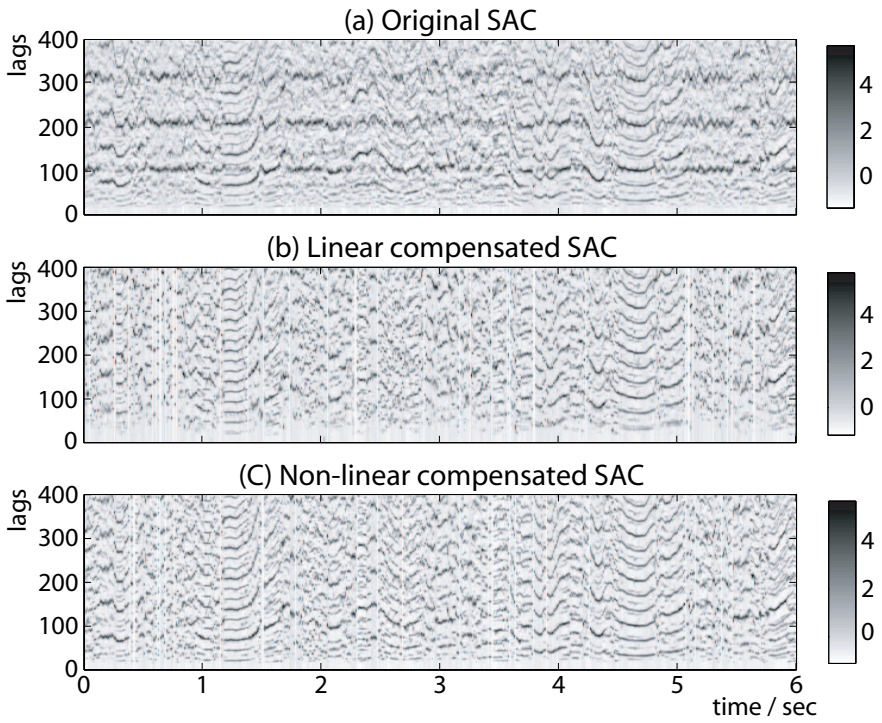

Figure 4.3: SACs for the input signal from figure 4.1 with and without compensation using the local-average ACF over a 1 s window. Stationary harmonic air-conditioning noise appears as a sequence of strong peaks at lags of 105, 210 and 315 samples in the original SAC, but is clearly deleted in the non-linear compensated SAC (panel (c)), which also preserves speech information lost in the linear compensated SAC of panel (b). The non-linear compensated SAC is the basis of the enhanced pitch tracks shown in figure 4.1 (d).

4.1.1.5 Cross-channel Integration and HMM Pitch Tracking

As in [67], the R_{yy}s are integrated across frequency channels to obtain an SAC. Finally, an HMM is used to extract continuous pitch tracks. We define the pitch state as the union of two subspaces, one pitch or no pitch. In each frame, a hidden node represents the set of observed peaks. While the transition behavior with the same pitch subspace is modeled by a Laplacian distribution, the transition between different subspaces can be determined by training given a constant probability of a zero pitch. The Viterbi algorithm is used to find the most likely sequence of pitch

states. We allow the probability of the no pitch state to vary according to the level of noise. Given a transition matrix estimated for relatively clean speech, we calculate pitch tracks with multiple different values for the zero-pitch probability, set as the n^{th} percentile of the SAC in each frame, and then determine the best percentile value by training. We also used the complete set of HMM posterior probabilities across all thresholds as a feature vector for SVM classification (below).

4.2 Detecting Music in Ambient Audio by Long-window Autocorrelation

In trying to design robust features, we focus on the two key characteristics of music worldwide shown in Figure 4.4 : Pitch and Rhythm. Pitch refers to the perceived musical notes that build up melodies and harmony, and is generally conveyed by locally-periodic signals (thus possessing a spectrum with harmonic peaks); musical instruments are usually designed to have relatively stable periods, and musical notes typically last for hundreds of milliseconds before the pitch is changed. Rhythm is the regular temporal structuring of note events giving rise to a sense of beat or pulse, usually at several hierarchically-related levels (beat, bar, etc.). While a given musical instance may lack clear pitch (e.g. percussion music) or a strong rhythm (e.g. an extremely 'romantic' piano style), it is difficult to imagine music possessing neither.

In the next section, we describe a music detection feature for detecting the stable periodicities of pitch that is robust to high levels of background noise.

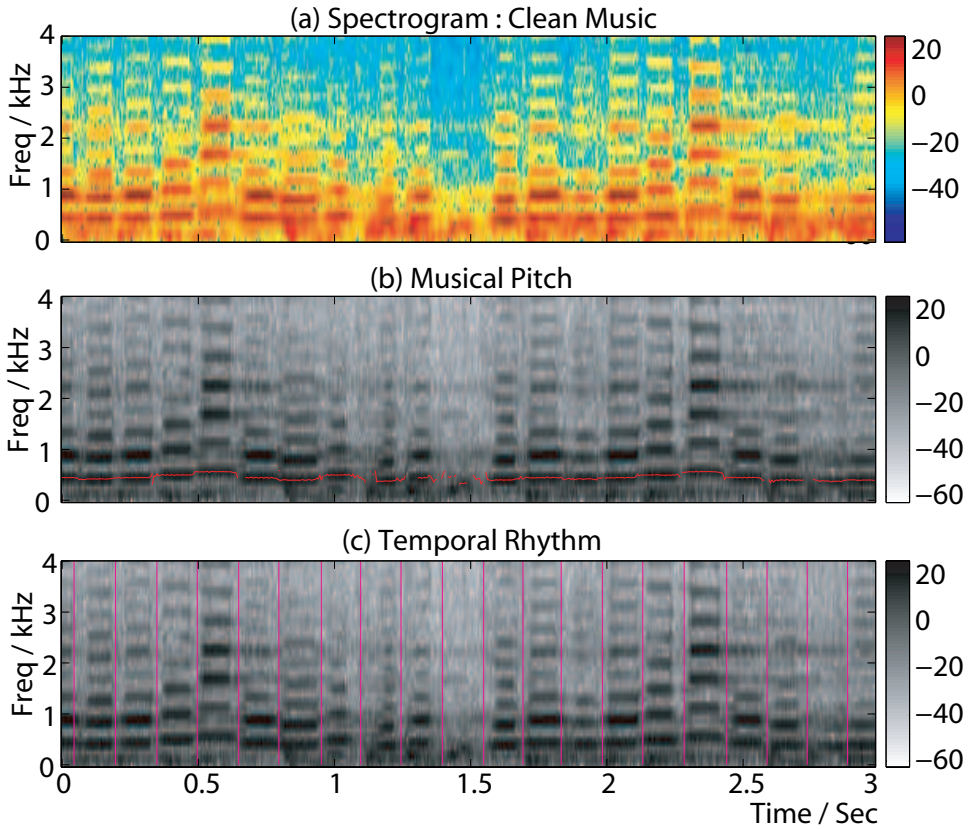

Figure 4.4: Example of clean music sound showing the pitch (panel(b) and temporal rhythm (panel (c)).

4.2.1 Noise-robust Musical Pitch Detection

Our strategy for detecting musical pitches is to identify the autocorrelation function (ACF) peaks resulting from the periodic, pitched energy that are stationary for around 100..500 ms, but to exclude aperiodic noise and stationary periodicity arising from background noise. Whitening by Linear Predictive (LP) inverse filtering prior to ACF concentrates aperiodic noise energy around zero lag, so we use only higher-lag coefficients to avoid this energy. Calculating the ACF over 100 ms windows emphasizes periodicities stable on that time scale, but we then subtract the long-term average ACF to remove any stationary, periodic background. Finally, the

stability (or dynamics) of pitch content is estimated by a feature composed of the cosine similarity between successive frames of the compensated ACF.

4.2.1.1 LPC Whitening and ACF

Mono input recordings are resampled to 16 kHz, and fit with a 12th order LPC model over 64 ms windows every 32 ms. Further processing is applied to residual of this modeling, which is a spectrally flat (whitened) version of the original signal, preserving any pitch-rate periodicity. The short-time ACF $r_{ee}(n,\tau)$ for each LPC residual envelope output $e(n)$ at a given time index n may be defined as:

$$r_{ee}(n,\tau) = \sum_{i=n+1}^{n+W} e(i)e(i+\tau) \tag{4.10}$$

where W is an integration window size, and $r_{ee}(n,\tau)$ is calculated over 100 ms windows every 5 ms for lag $\tau = 0 \ldots 200$ samples (i.e. up to 12.5 ms for a lowest pitch of 80 Hz). $r_{ee}(n,\tau)$ has a large value when $e(n)$ is similar to $e(n+\tau)$, i.e. if $e(n)$ has a period of P, then $r_{ee}(n,\tau)$ has peaks at $\tau = lP$ where l is an integer.

4.2.1.2 ACF Compensation

Assume that residual $e(n)$ consists of a clean musical signal $m(n)$ and a background aperiodic noise $a(n)$ and stationary periodic noise $b(n)$ i.e. $e(n) = m(n)+a(n)+b(n)$. If the noise $a(n)$ and $b(n)$ are zero-mean and uncorrelated with $m(n)$ each other for large W, the ACF is given by:

$$r_{ee}(n,\tau) = r_{mm}(n,\tau) + r_{aa}(n,\tau) + r_{bb}(n,\tau) \tag{4.11}$$

To simplify notation, variables n and τ are henceforth dropped.

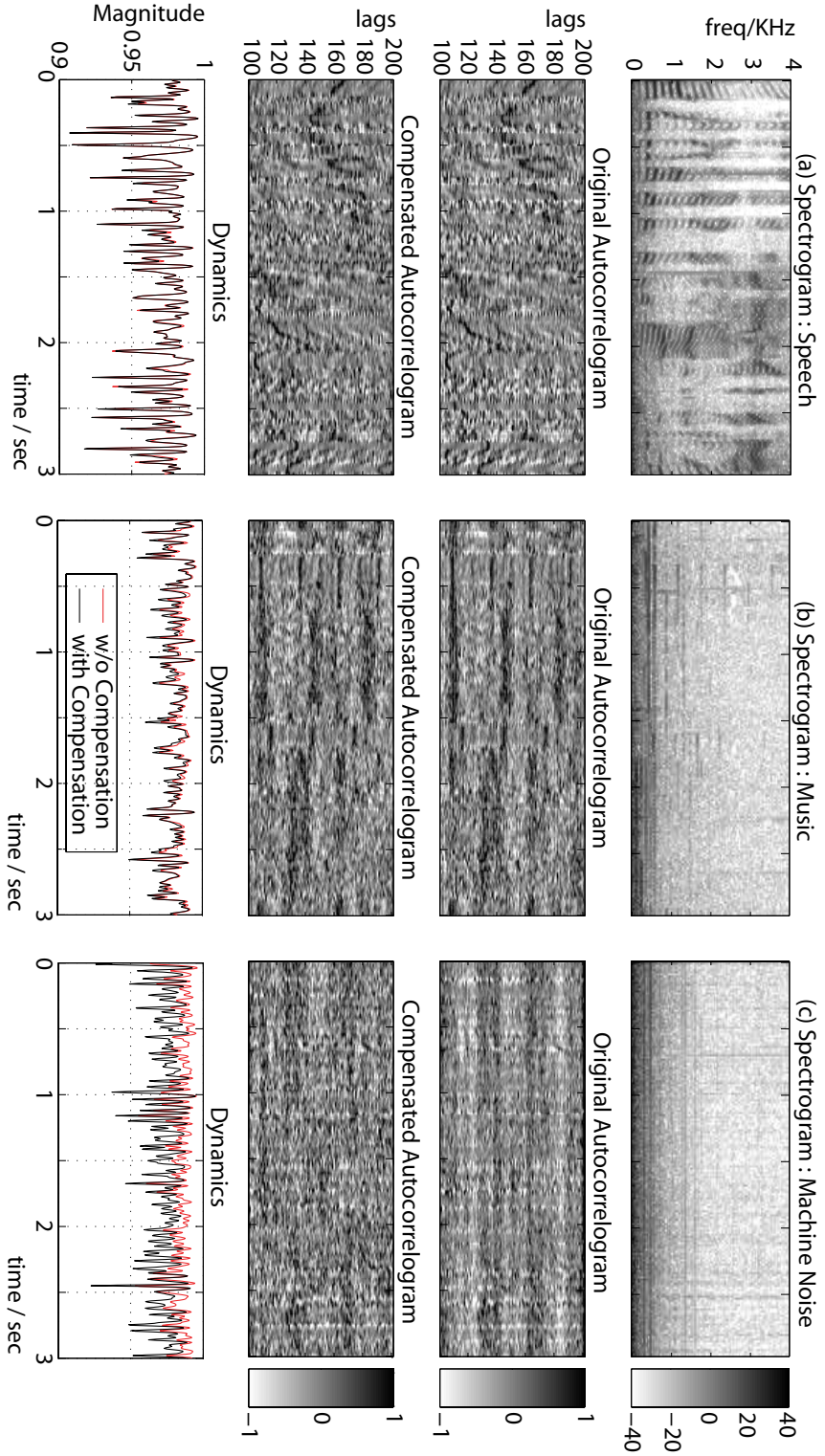

Figure 4.5: Examples of noisy speech, music and machine sound from a consumer audio recording.

4.2.1.3 Aperiodic Noise Suppression

The effect of LPC whitening is to concentrate the ACF of unstructured noise, r_{aa} at or close to the zero-lag bins. We can remove the influence of aperiodic noise from our features by utilizing only the coefficients of r_{ee} for lag $\tau \geq \tau_1$ samples (i.e. in our system, $\tau \geq 100$).

$$r_{ee} = r_{mm} + r_{bb},\ for\ \tau \geq 100 \tag{4.12}$$

Once the low-lag region has been removed, ACF r_{ee} is normalized by its energy $||r_{ee}||$ to lie in the range from -1 and 1.

4.2.1.4 Long-time Stationary Periodic Noise Suppression

A common form of interference in environmental recordings is a stationary periodic noise such as the steady hum of a machine as shown in the third column of Figure 4.5, resulting in ACF ridges that are not, in fact, related to music [38]. The ACF contribution of this noise r_{bb} will change very little with time, so it can be approximated as the long-time average of r_{ee} over M adjacent frames (covering around 10 second). We can estimate the autocorrelation of the music signal, $\hat{r}_{mm}$, as the difference between the local ACF and its long-term average,

By approximately estimating r_{bb} as the average ACF over M ($>$ 10 second) adjacent frames $avg\{r_{ee}\}$, the aperiodic noise-free ACF r_{ee} at these high lags can assume to be represented by a linear combination of weighted $avg\{r_{ee}\}$ and an estimated uncorrupted music ACF $\hat{r}_{mm}$ for each frame.

$$\hat{r}_{mm} = r_{ee} - \gamma \cdot \hat{r}_{bb} = r_{ee} - \gamma \cdot avg\{r_{ee}\} \tag{4.13}$$

γ is a scaling term to accommodate the per-frame normalization of the high-lag ACF and is calculated as the best projection of the average onto the current frame:

$$\gamma = \frac{\sum_\tau r_{ee} avg\{r_{ee}\}}{\sum_\tau avg\{r_{ee}\} avg\{r_{ee}\}} \quad for \tau \geq 100 \tag{4.14}$$

This estimated music ACF $\hat{r}_{mm}$ is shown in the third row of Figure 4.5.

4.2.2 Pitch Dynamics Estimation

The stability of pitch in time can be estimated by comparing temporally adjacent pairs of the estimated music ACFs:

$$\Upsilon(n) = S_{cos}(\hat{r}_{mm}(n), \hat{r}_{mm}(n+1)) \tag{4.15}$$

where S_{cos} is the cosine similarity (dot product divided by both magnitudes) between the two AC vectors.

Υ is shown in the fourth row of Figure 4.5. The sustained pitches of music result in flat pitch contours in the ACF, and values of Υ that approach 1, as shown in the second column of Figure 4.5. By contrast, speech (column 1) has a constantly-changing pitch contour, resulting in a generally smaller Υ, and the initially larger Υ of stationary periodic noise from e.g. machine is attenuated by our algorithm (column 3).

4.3 Evaluations

Our proposed algorithms for detecting speech or music are evaluated on personal audio and the soundtracks of consumer videos respectively.

4.3.1 Speech Detection Results

A 15 min test set was collected by a belt-mounted recorder worn during an outdoor discussion with four people (in front of the campus library), and thus was highly contaminated by noises including other people's voices and air-conditioning noise. We manually annotated it into three categories: foreground speech (FS), background speech (BS) and nonspeech (NS). In our experiments, we compared four discrimination tasks: FS versus BS+NS, FS+BS versus NS, BS versus NS and FS versus NS.

The data set was divided into a 5 min training and a 10 min testing set. For our experiments, we computed the pitch track contour and the HMM posterior probabilities using every 5^{th} percentile of the SAC at each frame as the zero-pitch probability. We used these features as the basis for two voice detector systems: For the first system, after choosing the best fixed zero-pitch threshold on training set, we took the presence of a non-zero pitch track as indicating speech. The second system detected speech with a 2-way SVM classifier based on the 20-dimensional feature set of the HMM posterior probabilities across all zero-pitch probability settings.

As shown in figure 4.1, within speech regions labeled manually, there are many unvoiced segments between prominent syllables or words. Using pitch to detect the presence of voice cannot, of course, directly recognize these unpitched speech segments, but we smoothed the output of the pitch detector with a 1 s median filter to provide labels more directly comparable to the hand-labeled ground-truth.

The overall performance on the testing data is presented in table 4.1 in terms of the accuracy rate and d' (a threshold-independent measure, taken as the separation between two unit-variance Gaussian distributions that would exhibit the same level of performance). For comparison, we also used a baseline of guessing all frames as a single class. The accuracy and d' with the non-linear ACF compensation are sig-

Tasks	Guessing (Accuracy)	Binary Decision with Pitch Tracks (Accuracy, d')	
		Without Non-linear AC Compensation	With Non-linear AC Compensation
FS/BS+NS	51.7%	73.8%, 1.66	**83.9**%, 1.99
FS+BS/NS	68.0%	76.9%, 1.26	**81.0**%, **2.07**
BS/NS	66.2%	57.8%, 0.48	**75.7**%, **1.24**
FS/NS	61.8%	79.4%, 1.74	**88.0**%, **2.44**
Tasks	Guessing (Accuracy)	SVM Classification with HMM Posterior (Accuracy, d')	
		Without Non-linear AC Compensation	With Non-linear AC Compensation
FS/BS+NS	51.7%	75.9%, 1.73	83.7%, **2.05**
FS+BS/NS	68.0%	74.2%, 1.60	80.2%, 2.00
BS/NS	66.2%	59.3%, 0.63	71.9%, 1.17
FS/NS	61.8%	76.5%, 1.96	85.8%, 2.36

Table 4.1: Voice detection performance. The accuracy rate is the proportion of voiced frames correctly detected, and d' (threshold-independent measure of class separation). The best value in each row is shown in bold. The best threshold for zero-pitch probability was estimated as the 61^{st} percentile of the SAC for the Binary Decision with Pitch Tracks system.

nificantly better than those without, which improves FS/BS+NS discrimination by about 10% absolute, and BS/NS discrimination by about 20%. Thus, the proposed algorithm is effective even for weak speech. The decision based on nonzero pitch track was simpler and by almost every measure (marginally) superior to the SVM classifier, and is thus preferred on the basis of its lower computational cost.

4.3.2 Music Detection Results

The pitch dynamics feature Υ was summarized by its mean (mDyn) and variance (vDyn) for the purpose of classifying clips. We compared these features with others that have been successfully used in music detection [59], namely the 4Hz Modulation Energy (4HzE), Variance of the spectral Flux (vFlux) and Rhythm (Rth) which we

	2.4s Segment	
	Speech vs. Music w/ vocals	Speech vs. Music w/o vocals
T	96/120, 65/120	96/120, 62/120
mDyna	114/120, 99/120	114/120, 104/120
vDyna	89/120, 115/120	89/120, 116/120
4HzE	**106/120**, **118/120**	**106/120**, **120/120**
vFlux	106/120, 116/120	**106/120**, **120/120**
T+mDyna	111/120, 109/120	**111/120**, **114/120**
4HzE+vFlux	**104/120**, **118/120**	104/120, 120/120
T+mDyna+vDyan	112/120, 114/120	**112/120**, **117/120**
T+4HzE+vFlux	103/120, 119/120	103/120, 120/120
T+mDyna+4HzE	**108/120**, **119/120**	108/120, 120/120
T+mDyna+vFlux	108/120, 117/120	108/120, 120/120

Table 4.2: Speech / Music (with or w/o vocals) classification accuracy of broadcasting recordings with one Gaussian classifier. Each value indicates how many of the 2.4 second segments out of a total of 120 are correctly classified as speech or music. The best performance of each column is shown in bold.

took as the largest peak value of normalized ACF of an 'onset strength' signal over the tempo range (50-300 BPM) [20].

Table 4.2 compares performance on a data set of random clips captured from broadcast radio, as used in [59]. The data was randomly divided into a 15 s segments, giving 120 for training and a 60 for testing (20 each of speech, music with vocals, and music without vocals). Classification was performed by a likelihood ratio test of single Gaussians fit to the training data. 4HzE and vFlux have the best performance among single features, but Rth + mDyn + vDyn has the best performance (by a small margin) in distinguishing speech from vocal-free music.

However, classification of clean broadcast audio is not the main goal of our current work. We also tested these features on the soundtracks of 1873 video clips from the YouTube [2], returned by consumer-relevant search terms such as 'animal', 'people', 'birthday', 'sports' and 'music', then filtered to retain only unedited,

raw consumer video. Clips were manually sorted into 653 (34.9%) that contained music, and 1220 (65.1%) that did not. We labeled a clip as music if it included clearly-audible professional or quality amateur music (regardless of vocals or other instruments) throughout. These clips are recorded in a variety of locations such as home, street, park and restaurant, and frequently contain noise including background voices and many different types of a mechanical noise.

We used a 10 fold cross-validation to evaluate the performance in terms of the accuracy, d' (the equivalent separation of two normalized Gaussian distributions), and Average Precision (the average of the precision of the ordered returned list truncated at every true item). We compared two classifiers, a single Gaussian as above, and an SVM with an RBF kernel. At each fold, the classifier is trained on 40% of the data, tuned on 20%, and then are tested on the remaining 40% selected at random. For comparison,we also report the performance of the '1G+KL with MFCC' system from [10], which simply takes the mean and covariance matrix of MFCC features over the entire clip, and then uses an SVM classifier with a symmetrized Kullback-Leibler (KL) kernel.

As shown in table 4.3, the new mDyn feature is significantly better than previous features 4HzE or vFlux, which are less able to detect music in the presence of highly-variable noise. The best 2 and 3 feature combinations are 'Rth + mDyn' and 'Rth + mDyn + vFlux' (which slightly outperforms 'Rth + mDyn + vDyn' on most metrics). This confirms the success of the pitch dynamics feature, Υ, in detecting music in noise.

Features	One Gaussian Classifier		
	Accuracy(%)	d'	AP(%)
T	$\mathbf{82.1 \pm 1.08}$	$\mathbf{1.84 \pm 0.08}$	$\mathbf{74.9 \pm 1.66}$
mDyna	80.3 ± 1.2	1.65 ± 0.09	71.3 ± 3.12
vDyna	63.3 ± 1.35	0.78 ± 0.07	48.1 ± 2.77
4HzE	64.2 ± 0.86	0.82 ± 0.09	51.6 ± 1.82
vFlux	63.5 ± 1.03	0.84 ± 0.06	52.9 ± 2.77
T+mDyna	$\mathbf{86.9 \pm 0.92}$	$\mathbf{2.17 \pm 0.09}$	$\mathbf{86.4 \pm 1.41}$
T+vDyna	84.1 ± 1.17	1.96 ± 0.12	77 ± 2.37
mDyna+vDyna	83.8 ± 1.02	1.88 ± 0.09	76.4 ± 2.69
4HzE+vFlux	65.5 ± 1.28	0.87 ± 0.07	55 ± 3.53
T+mDyna+vDyna	89.5 ± 0.54	2.44 ± 0.06	87.5 ± 0.92
T+4HzE+vFlux	83.2 ± 1.53	1.98 ± 0.2	80.4 ± 3.18
T+mDyna+4HzE	89.2 ± 1.1	2.4 ± 0.12	87.9 ± 1.46
T+mDyna+vFlux	$\mathbf{90 \pm 1.35}$	$\mathbf{2.51 \pm 0.13}$	$\mathbf{89.3 \pm 1.28}$
T+mDyna+4HzE+vFlux	89.3 ± 1.01	2.41 ± 0.1	87.4 ± 1.07
All	89.8 ± 0.94	2.48 ± 0.11	87.6 ± 2.06
1G+KL with MFCC	N/A	N/A	N/A

Features	SVM Classifier		
	Accuracy(%)	d'	AP(%)
T	$\mathbf{82.6 \pm 1.11}$	$\mathbf{1.77 \pm 0.09}$	$\mathbf{80.8 \pm 1.7}$
mDyna	80.3 ± 1.36	1.66 ± 0.1	78.6 ± 2.33
vDyna	65.8 ± 1.25	0.73 ± 0.06	51.6 ± 2.98
4HzE	64.9 ± 1.87	0.87 ± 0.08	53.8 ± 2.12
vFlux	66 ± 1.31	0.88 ± 0.07	52.7 ± 3.47
T+mDyna	$\mathbf{88 \pm 1.09}$	$\mathbf{2.31 \pm 0.14}$	$\mathbf{89.7 \pm 1.97}$
T+vDyna	85.3 ± 0.85	2 ± 0.06	84.7 ± 1.31
mDyna+vDyna	82.5 ± 0.98	1.83 ± 0.1	82.6 ± 2.55
4HzE+vFlux	66.8 ± 1.08	0.91 ± 0.09	53 ± 5.19
T+mDyna+vDyna	89.3 ± 1.16	2.45 ± 0.13	$\mathbf{92.9 \pm 1.63}$
T+4HzE+vFlux	85.2 ± 1.43	2.03 ± 0.13	84.9 ± 2.83
T+mDyna+4HzE	$\mathbf{90.3 \pm 1.04}$	$\mathbf{2.58 \pm 0.13}$	92.2 ± 1.79
T+mDyna+vFlux	90.1 ± 1.01	2.51 ± 0.15	91.2 ± 2.13
T+mDyna+4HzE+vFlux	$\mathbf{90.9 \pm 1.21}$	$\mathbf{2.63 \pm 0.17}$	92.2 ± 2.04
All	89.9 ± 1.57	2.53 ± 0.17	91.2 ± 2.1
1G+KL with MFCC	80.2 ± 0.75	1.68 ± 0.007	80.4 ± 1.82

Table 4.3: Music/Non-music Classification Performance on YouTube consumer environmental recordings. Each data point represents the mean and standard deviation of the clip-based performance over 10 cross-validated experiments. Where d' is a threshold-independent measure of the separation between two unit-variance Gaussian distributions and AP is the Average of Precisions calculated for each of relevant examples separately to be a higher value when more relevant examples, i.e. music clips, is returned earlier. The best performance of each column is shown in bold.

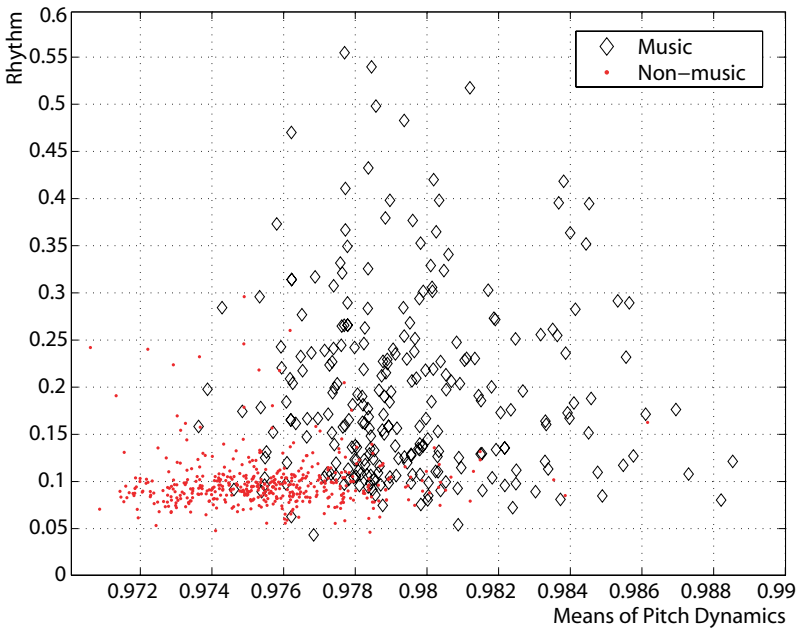

Figure 4.6: Distribution of 750 sound clips of a testing set along the musical pitch and rhythm spaces

4.4 Discussion

Subsequent informal experiments have revealed that the sustained notes of background music can also be removed by this technique, which is a direction for further investigation e.g. for applications involving the recognition of broadcast speech: Detected voice pitch can be used for harmonic filtering to remove much of the nonspeech energy, to provide a drop-in replacement ASR feature. The multipitch tracker may also be helpful to suppress weak background voices after deleting strong stationary harmonic noises; this aspect is also currently under investigation.

As shown in Figure 4.6, we can more fully understand the behavior of algorithm by investigating the distribution of errors. First, there are many errors due to the ambiguity of manually labeling music. based on the perception of annotators, some clips containing relatively short or weak music (i.e. severally corrupted by the

background speech at concert) and poor quality music (i.e. a baby beats a piano or blow a flute) are tagged into non-music, but some of them are classifying into the music. Second, Many singing (i.e. kids' singing at birthday party) without instruments labeled into non-music due to a limited musical value are sometimes classified into music because of a flat singing pitch. Finally, some errors occur in cheering (or screaming) voices and alarm sounds such as a whistling, honk sound and telephone ringing sound, which also generate a relatively flat contour similar to musical pitch.

4.5 Summary

In this chapter, we have proposed a robust pitch detection algorithm for identifying the presence of speech or music in the noisy, highly-variable personal audio collected by body-worn continuous recorders. In particular, we have introduced a new technique for estimating and suppressing stationary periodic noises such as air-conditioning machinery in the autocorrelation domain. The performance of our proposed algorithm is significantly better than existing speech or music detection systems for the kinds of data we are addressing.

Chapter 5

Generic Concept Detection

In this chapter, we develop a system to automatically detect a large set of interesting semantic concepts, which we chose for being both informative and useful to users, as well as being technically feasible. These concepts are associated with people's activities, locations, occasions, objects, scenes and sounds, and are based on a large collection of consumer videos in conjunction with user studies. We model the soundtrack of each video, regardless of its original duration, as a fixed-sized clip-level summary feature. For each concept, an SVM-based classifier is trained according to three distance measures (Kullback-Leibler, Bhattacharyya, and Mahalanobis distance) and tested on $1,900$ consumer clips.

Concepts have diverse characteristics in terms of consistency, frequency, and interrelationships. For example, the labels "music" and "crowd" typically persist over a large proportion if not the entirety of any clip to which they apply, and hence should be well represented in the global feature patterns (e.g., mean and covariance of a clip's frame-level features). However, the concept "cheer" manifests as a relatively small segment within a clip (at most a few seconds), which means that the global patterns of an entire clip may fail to distinguish it from others.

This points to the need for methods that can emphasize local patterns embedded in a global background, such as the probabilistic latent semantic analysis approach described below.

We briefly review the selection, definition and annotation of semantic concepts for consumer videos in Section 5.1. Audio-based detectors are described in Section 5.2. The evaluation and discussion of experimental results are included in Section 5.3 and 5.4 respectively. Finally, we summarize this in Section 5.5.

5.1 Data and Labels

5.1.1 The Semantic Concepts

Our goal is to provide classification that is relevant to users browsing personal video collections, thus our concepts must reflect the actual needs of this target group. In previous work [10], we defined the set of 25 concepts used here by starting from a full ontology of over 100 concepts obtained through user studies conducted by the Eastman Kodak company [42]. For our experiments, we further pared down to 25 concepts based on three criteria: (1) usefulness – whether a concept is useful in real-world consumer media applications; (2) detectability – whether a concept is practically anticipated to be detected in terms of the signal content features; and (3) observability – whether a concept is sufficiently clearly expressed to be observable by the third-person annotators. These selected concepts fall into several broad categories including activities, occasions, locations, or particular objects in the scene, as shown in Table 5.1. Most concepts are intrinsically visual, although some concepts, such as music and cheering, are primarily acoustic. Since most of the selected concepts are dominated by the visual cues, using visual cues achieved higher accuracy for most concepts than using audio cues. However, audio models provided

significant benefits. For example, by their nature, concepts like "music", "singing", and "cheer" can primarily be detected in the acoustic domain. Even for some visually dominated concepts (like "museum" and "animal"), audio methods were found to be more reliable than visual counterparts, implying that the soundtracks of video clips from these concepts provide rather consistent audio features for classification. By combining visual baseline detectors and audio baseline detectors through context fusion, the proposed Audio-Visual Boosted Conditional Random Field (AVBCRF) method algorithm improves the performance by more than 10% compared with the visual baseline. The improvements over many concepts are significant, e.g. 40% for "animal", 51% for "baby", 228% for "museum", 35% for "dancing", and 21% for "parade". This thesis describes for the first time the detail of the audio-based detectors used in that work.

5.1.2 Video Data

We downloaded 4,539 videos (about 200 videos for each concept) from YouTube [2] by using most relevant keywords (queries) associated with the definition of these 25 concepts. For these downloaded videos, we first manually filtered them to discard commercial videos, which are not consistent with the consumer video genre, or low-quality videos (especially those having poor sound quality).

Non-consumer videos are mainly composed of two kinds of videos: broadcasting content, and user-edited videos. The "sports" videos downloaded with using keywords like soccer, basketball, football, baseball, volleyball and ping-pong contain many commercial videos captured from TV sports. Some consumer videos are also extensively edited, e.g. the highlights of a field trip can have many abrupt changes of locations in single video clip. Some clips look like music videos, and have largely

Category	Concept	Definition	Examples
Activities	dancing	one or people dancing	189
	singing	singers visible and audible	345
	ski	one or people skiing	68
Locations	beach	sand and water visible	130
	museum	exhibitions of arts, antiques	45
	park	some greenery in view	118
	playground	swings, slides in view	96
Occasions	birthday	birthday cake, caps, songs	68
	graduation	caps and gowns visible	72
	picnic	people and food outdoors	54
	parade	people or vehicles moving	91
	show	concerts, plays, recitals	211
	sports	soccer, basketball, football, baseball, volleyball, ping-pong	84
	wedding	bride and groom in view	57
Objects	animal	dogs, cats, birds, wild animals	61
	baby	infant, 12 months or younger	112
	boat	boats in the water	89
	group of 3+	three or more people	1126
	group of 2	two people	252
	one person	single person	316
Scenes	crowd	many people in the distance	533
	night	outdoors at night	300
	sunset	the sun in view	68
Sounds	cheer	acclamation, hurrah	388
	music	clearly audible professional or amateur music	653
Total	25 concepts		1873 clips

Table 5.1: *Definition of the* 25 *concepts, and counts of manually-labeled examples of each concept from* 1,873 *videos.*

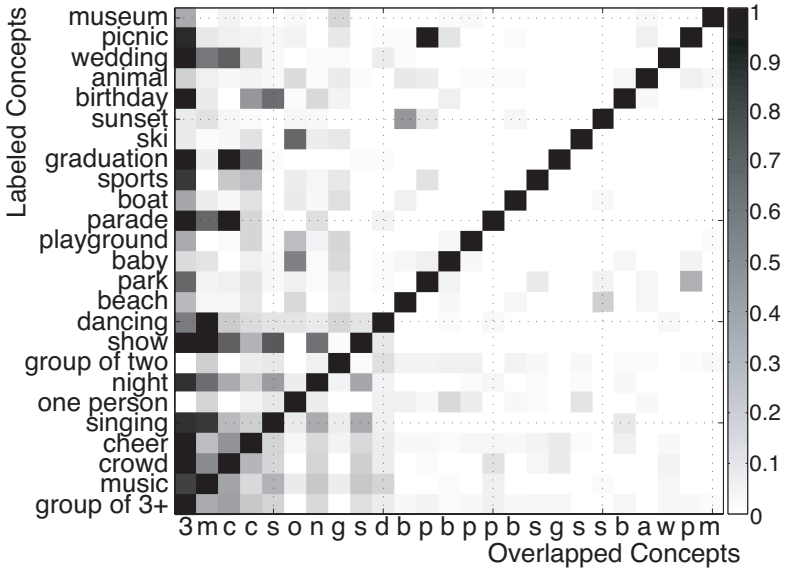

Figure 5.1: Co-occurrence matrix for the 25 manually-annotated labels within the 1,873 video set. Co-occurrence counts within each row are normalized by the total number of instances of that row's concept to give the conditional probability of observing the overlapped concept given the labeled concept.

replaced the original soundtrack with added background music. These types are also excluded.

In consequence, the 4,539 YouTube videos were reduced to 1,873 (41%) relevant, consumer-style clips, 1,261 (28%) irrelevant (non-consumer), and 1,405 (31%) poor-quality videos whose soundtracks had bandwidth less than 8kHz. We used only the 1,873 relevant videos with adequate sound quality as our experimental data set. The average duration of a clip from this set was 145 s.

Videos were downloaded based on the tags and description provided by their owners. However, people will generally tag their videos according to subjective definitions (e.g., labeling an entire ski trip as relating to the concept "skiing"). To ensure accurate labels, we manually reviewed every one of the 1,873 videos, and

tagged it with the concepts that it contained, as defined in Table 5.1. On average, each video ended up with 3 concept labels, and some labels were very likely to co-occur with others, as illustrated in Figure 5.1. For example, "group of three or more", "music", "crowd", and "cheer", are all highly overlapped with other concepts. The video collections and labels are described in [9] in detail and now available at http://labrosa.ee.columbia.edu/projects/consumervideo/.

5.2 Audio Concept Detection Algorithms

Our fundamental frame-level feature is the Mel-frequency Cepstral Coefficients (MFCCs) commonly used in speech recognition and other acoustic classification tasks. The single-channel (mono) soundtrack of a video is first resampled to 8kHz, and then a short-time Fourier magnitude spectrum is calculated over 25ms windows every 10ms. The spectrum of each window is warped to the Mel frequency scale, and the log of these auditory spectra is decorrelated into MFCCs via a discrete cosine transform.

After the initial MFCC analysis, each video's soundtrack is represented as a set of $d = 21$ dimensional MFCC feature vectors, where the total number of frames depends on the duration of the original video. (21 dimensions were chosen based on results from our earlier experiments [22]; general audio classification usually benefits from using more MFCC dimensions than are commonly encountered e.g. in speech recognition.) To reduce this set of MFCC frames, regardless of its original size, to a single fixed-dimension clip-level feature vector, we experimented with three different techniques: Single Gaussian Modeling (1G), Gaussian Mixture Modeling (GMM), and probabilistic Latent Semantic Analysis of a Gaussian Component Histogram (pLSA). Each of these is discussed in more detail below.

These fixed-size representations are then compared to one another distance measures including Kullback-Leibler divergence (KL), Bhattacharyya distance (Bha),

and Mahalanobis distance (Mah). Distances between clips form the input to a Support Vector Machine classifier as described in the next subsection.

5.2.1 Support Vector Machines (SVMs)

The SVM is a supervised learning method used for classification and regression that has many desirable properties [60]. Data items are projected into a high-dimensional feature space, and the SVM finds a separating hyperplane in that space that maximizes the margin between sets of positive and negative training examples. Instead of working in the high-dimensional space directly, the SVM requires only the matrix of inner products between all training points in that space, also known as the kernel or gram matrix. with a method similar to [33], we exponentiate the matrix of distances between examples, $D(f, g)$, to create a gram matrix $K(f, g)$:

$$K(f, g) = \exp(-\gamma \cdot D(f, g)) \tag{5.1}$$

where $\gamma = \{2^{10}, 2^9, ..., 2^{-10}\}$, and f and g index the video clips. We use the so-called slack-SVM that allows a trade-off between imperfect separation of training examples and smoothness of the classification boundary, controlled by a constant C that we vary in the set $\{10^1, 10^2, ...10^{10}\}$. Both tunable parameters γ and C are chosen to maximize classification accuracy over a held-out set of validation data. After training an independent SVM model for each concept, we apply the classifiers to summary features derived from the test video clips. The resulting distance-to-boundary is a real value that indicates how strongly the video is classified as reflecting the concept. The test videos are then ranked according to this value as a retrieval result for the relevant concept. Following conventions in information retrieval, we evaluate classifiers by calculating their average precision (AP), which is the proportion of

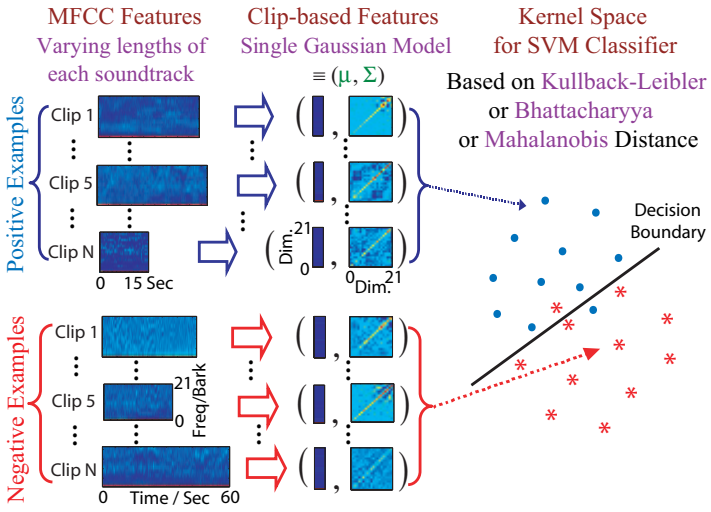

Figure 5.2: The process of calculating clip-level features via a single Gaussian model per clip, and using them within an SVM classifier.

true results in a ranked list truncated at the n^{th} true item, averaged over all n.

5.2.2 Single Gaussian Modeling (1G)

The basic assumption of Single Gaussian Modeling is that different activities (or concepts) are associated with different sounds whose average spectral shape and variation, as calculated by the cepstral feature statistics, will be sufficient to discriminate categories. This approach is based on common practice in speaker recognition and music genre identification, where the distribution of cepstral features, collapsed across time, is found to be a good basis for classification [56, 45]. Specifically, to describe a clip's sequence of MFCC features as a single feature vector, we ignore the time dimension and treat the set as a "bag of the frames" in MFCC feature space, which we then model as a single, full-covariance Gaussian distribution. This Gaussian is parameterized by its 21-dimensional mean vector μ and 21×21-dimensional (full) covariance matrix Σ. The overall process of the Single Gaussian Modeling is illustrated in Figure 5.2.

To calculate the distance between two Gaussians, as required for the gram-matrix input (or kernel matrix) for the SVM, we have experimented with three different distance measures. First is the Kullback-Leibler (KL) divergence: If two clips f and g are modeled by single Gaussians as:

$$f(x) = \mathcal{N}(\mu_f, \Sigma_f), \quad g(x) = \mathcal{N}(\mu_g, \Sigma_g) \tag{5.2}$$

respectively, then the distance between the clips is taken as the KL divergence between Gaussians $f(x)$ and $g(x)$ i.e.:

$$\begin{aligned} D_{KL}(f,g) &= (\mu_f - \mu_g)^T(\Sigma_f^{-1} + \Sigma_g^{-1})(\mu_f - \mu_g) \\ &\quad + trace(\Sigma_f^{-1}\Sigma_g + \Sigma_g^{-1}\Sigma_f) - 2d \end{aligned} \tag{5.3}$$

The second distance measure is the Bhattacharyya (Bha) distance, defined by:

$$\begin{aligned} D_B(f,g) &= \frac{1}{4}(\mu_f - \mu_g)^T(\Sigma_f + \Sigma_g)^{-1}(\mu_f - \mu_g) \\ &\quad + \frac{1}{2} log \left| \frac{\Sigma_f + \Sigma_g}{2} \right| - \frac{1}{4} log|\Sigma_f \Sigma_g| \end{aligned} \tag{5.4}$$

The final approach simply treats the d-dimensional mean vector μ concatenated with the $d(d+1)/2$ independent values (diagonal and upper triangular elements) of the covariance matrix Σ as a point in a new $21 + 231$ dimensional feature space describing the clip. These 252-dimensional features, denoted by h_f and h_g for videos f and g, are compared to one another using the Mahalanobis (i.e. covariance-normalized Euclidean) distance to build the gram matrix:

$$D_M(f,g) = (h_f - h_g)^T \Sigma_h^{-1} (h_f - h_g) \tag{5.5}$$

where Σ_h is the covariance of these features taken across the entire training set. We assumed Σ_h to be diagonal i.e. consisting only of the variance of each dimension.

5.2.3 Gaussian Mixture Models (GMM)

In order to capture details of feature distributions that may not be well fit by a single Gaussian, we also experimented with using a mixture of diagonal-covariance Gaussians, estimated via the EM algorithm, to describe the bag-of-frames distribution. To compare GMMs, we use just one distance measure, an approximation to the Bhattacharyya distance that was shown to give good performance in tasks requiring the comparison of GMMs [31]: Assume that the distributions of two clips, $f(x)$ and $g(x)$, are represented by two different GMMs:

$$f(x) = \sum_a \pi_a \mathcal{N}(\mu_a, \Sigma_a), \quad g(x) = \sum_b \pi_b \mathcal{N}(\mu_b, \Sigma_b) \tag{5.6}$$

where π_a, μ_a, and Σ_a are the prior weight, mean, and covariance of each Gaussian mixture component used to approximate clip f, and the b-subscripted values are for clip g. To simplify notation, we call $f_a = \mathcal{N}(\mu_a, \Sigma_a)$ and $g_b = \mathcal{N}(\mu_b, \Sigma_b)$ henceforth.

Although there is no closed-from expression for the Bhattacharyya divergence between two GMMs, it can be approximated by variational methods [31]. The Bhattacharyya similarity between two distributions $f(x)$ and $g(x)$ is:

$$\begin{aligned} B(f,g) &\equiv \frac{1}{2}\int \sqrt{f(x)g(x)}dx \\ &\geq \sqrt{\sum_{ab} \pi_a \pi_b B^2(f_a, g_b)} \equiv \hat{B}^v(f,g) \end{aligned} \tag{5.7}$$

where $B(f_a, g_b)$ is the Bhattacharyya distance between a particular pair of single Gaussians, one from each mixture. To preserve the identity property that $\hat{B}(f,g) =$

$\frac{1}{2}$ if and only if $f = g$, the variational Bhattacharyya similarity $\hat{B}^v$ is normalized using the geometric mean of $B(f, f)$ and $B(g, g)$:

$$\hat{B}_{norm}(f, g) = \frac{\hat{B}^v(f, g)}{\sqrt{\hat{B}^v(f, f)\hat{B}^v(g, g)}} \tag{5.8}$$

With this normalized Bhattacharyya approximation, the corresponding Bhattacharyya divergence is defined as: $D_B(f, g) = -\log(2\hat{B}_{norm}(f, g))$.

5.2.4 Probabilistic Latent Semantic Analysis (pLSA)

Unlike the Gaussian models' assumption that each concept is distinguished by the global distribution of all short-time feature vectors, this approach recognizes that each soundtrack will consist of many different sounds that may occur in different proportions even for the same category, leading to variations in the global statistics. If, however, we could decompose the soundtrack into separate descriptions of those specific sounds, we might find that the particular palette of sounds, but not necessarily their exact proportions, would be a more useful indicator of the content. Some kinds of sounds (e.g. background noise) may be common to all classes, whereas some sound classes (e.g. a baby's cry) might be very specific to a particular class of videos.

To build a model better able to capture this idea, we first construct the vocabulary (or palette) of sounds by constructing a large GMM, composed of M Gaussian components; we experimented with M in the range 256 to 1024. This large GMM was trained on MFCC frames subsampled from all videos from the training set, regardless of label. (We observed a marginally better performance after training the GMM on a set of frames selected as the central points of about 100 groups, clustered by the K-means algorithm on each clip, instead of a random sampling method). The

resulting M Gaussians are then considered as anonymous sound classes from which each individual soundtrack is assembled – the analogues of words in document modeling. We assign every MFCC frame in a given soundtrack to the the most likely mixture component from this 'vocabulary' GMM, and describe the overall soundtrack with a histogram of how often each of the M Gaussians was chosen when quantizing the original clip's frames.

Suppose that we have given a collection of training clips $C = \{c_1, c_2, ...c_N\}$ and an M-mixture of Gaussians $G = \{g_1, g_2, ...g_M\}$. We summarize the training data as a $N \times M$ co-occurrence matrix of counts O with elements $o_{ij} = o(c_i, g_j)$, the number of times mixture component g_j occurred in clip c_i. Normalizing this within each clip gives an empirical conditional distribution $P(g|c)$. Note that this representation also ignores temporal structure, but it is able to distinguish between nearby points in cepstral space provided they were represented by different Gaussians in the vocabulary model. The idea of using histograms of acoustic tokens to represent the entire soundtrack is also similar to that of using visual token histograms for image representation [50, 41].

We could use this histogram $P(g|c)$ directly, but to remove redundant structure and to give a more compact description, we go on to decompose the histogram with probabilistic Latent Semantic Analysis (pLSA) [32]. This approach, originally developed to generalize the distributions of individual words in documents on different topics $Z = \{z_1, z_2, ...z_K\}$, models the histogram as a mixture of a smaller number of 'topic' histograms, giving each document a compact representation in terms of a small number of topic weights. The individual topics are defined automatically to maximize the ability of the reduced-dimension model to match the original set of histograms. (This technique has been used successfully in an audio application by Arenas-Garcia et al. [5], who use pLSA as a way to integrate and condense different

features of music recordings for applications in similarity and retrieval.)

Specifically, the histogram-derived probability $P(g|c)$ that a particular component g will be used in clip c is approximated as the sum of contributions from topics z, $p(g|z)$, weighted by the specific contributions of each topic to the clip, $p(z|c)$, i.e.

$$P(g|c) = \sum_{z \in Z} P(g|z)P(z|c) \tag{5.9}$$

which embodies the assumption that conditioning on a topic z makes clip c and component g independent. During training, the topic profiles $P(g|z)$ (which are shared between all clips), and the per-clip topic weights $P(z|c)$, are optimized by using the Expectation Maximization (EM) algorithm. In the Expectation (E) step, posterior probabilities are computed for the latent variables:

$$P(z|c,g) = \frac{P(z)P(c|z)P(g|z)}{\sum_{z' \in Z} P(z')P(c|z')P(g|z')} \tag{5.10}$$

Parameters are updated in the maximization (M) step:

$$\begin{aligned} P(g|z) &\propto \sum_{c} o(c,g)P(z|c,g) \\ P(c|z) &\propto \sum_{g} o(c,g)P(z|c,g) \\ P(z) &\propto \sum_{c}\sum_{g} o(c,g)P(z|c,g) \end{aligned} \tag{5.11}$$

The number of distinct topics determines how accurately the individual distributions can be matched, but also provides a way to smooth over irrelevant minor variations in the use of certain Gaussians. We tuned it empirically on the development data, as described in Section 5.3. Representing a test item similarly involves finding the best set of weights to match the observed histogram as a (nonnegative)

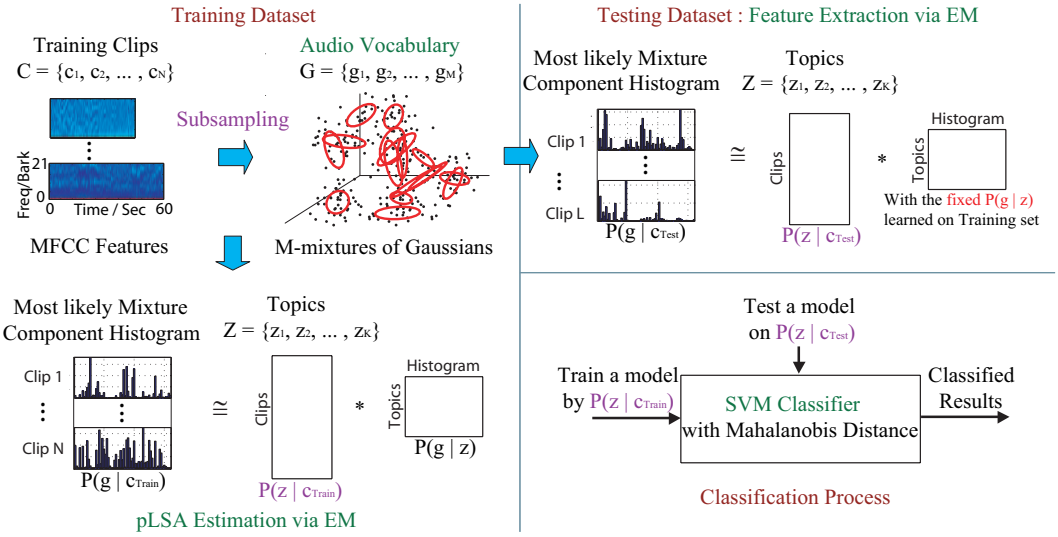

Figure 5.3: Illustration of the calculation of pLSA-based features and clip-level comparisons, based on GMM component histograms. Top left shows the formation of the global GMM; bottom left shows the formation of the topic profiles, $P(g|z)$ and topic weights, $P(z|c_{train})$ in training data; top right shows the analysis of each testing clip into topic weights, $P(z|c_{test})$ by matching each histogram to a combination of topic profiles estimated by training data, and bottom right shows the final classification by an SVM.

combination of the topic profiles; we minimizing the KL distance via an iterative solution i.e., the per-clip topic weights $P(z|c)$ of testing data sets are optimized by using the EM algorithm with fixed the topic profiles $P(g|z)$ that is already estimated on training set.

Finally, each clip is represented by its vector of topic weights and the SVM's gram matrix is calculated as the Mahalanobis distance in that topic weight vector space. (Again, we assumed the feature covariance matrix was diagonal.) We compared several different variants of the topic weight vector: unmodified $P(z|c)$, log-transformed $\log(P(z|c))$ and log-normalized $\log(P(z|c)/P(z))$, which normalizes the topic weight by the prior of topics and then takes the logarithm. The process of pLSA feature extraction is illustrated in Figure 5.3.

5.3 Evaluations

We evaluate our approaches using 5-fold cross validation on our labeled collection of 1873 videos: At each fold, SVM classifiers for each concept are trained on 40% of the data, tuned on 20%, and then tested on the remaining 40%, selected at random.

In a preliminary experiment, we used the Single Gaussian Model with KL distance for the purpose of classifying 3,134 YouTube videos into the 1,873 (59.8%) consumer and 1261 (40.2%) non-consumer videos. The performance is evaluated in terms of the accuracy, d' (equivalent separation of two normalized Gaussian distributions) and Average Precision (the average of the precision of the ranked retrieved list truncated at every true item). The soundtrack of 'consumer video' has many characteristics that distinguish it from non-consumer audio: Casual recordings made with small, hand-held cameras will very often contain a great deal of spurious, non-stationary noise such as babble, crowd (e.g., many overlapped background voices), traffic (e.g., car, boat and machine noises), or handling artifacts (e.g. camera zooming sound). This unpredictable noise can have a great impact on the global characteristics of the signal – the broad spectral shape encoded by MFCC features – which may be dominated by noise. While this is a disadvantage from the point of view of classifying the clip's contents, it may help to discriminate between the kinds of consumer videos we are addressing and the more benign acoustic environments of professional video content. Our results as shown in Table 5.2.

We then evaluated all our approaches in terms of the AP for detecting the 25 concepts across the 1,873 consumer-style videos. Figure 5.4 shows the results of the Single Gaussian modeling (1G) with the three different distance measures, KL, Mahalanobis, and Bhattacharyya. 1G+KL gives better performance for location-related concepts such as "park", "playground", and "ski"; by contrast, audio-dominated concepts such as "music", "cheer", and "singing" are best with the 1G+Mah. Con-

Approach	Accuracy (%)	d'	AP for detecting consumer videos (%)
1G + KL	79.2 ± 2.7	1.75 ± 0.16	89.7 ± 0.5

Table 5.2: *Consumer (59.8%) / Non-consumer (40.2%) Classification Performance on 3,134 YouTube recordings based on the single Gaussian model with the KL distance measure. Each data point represents the mean and standard deviation of the clip-level performance over 5 cross-validated test folds.*

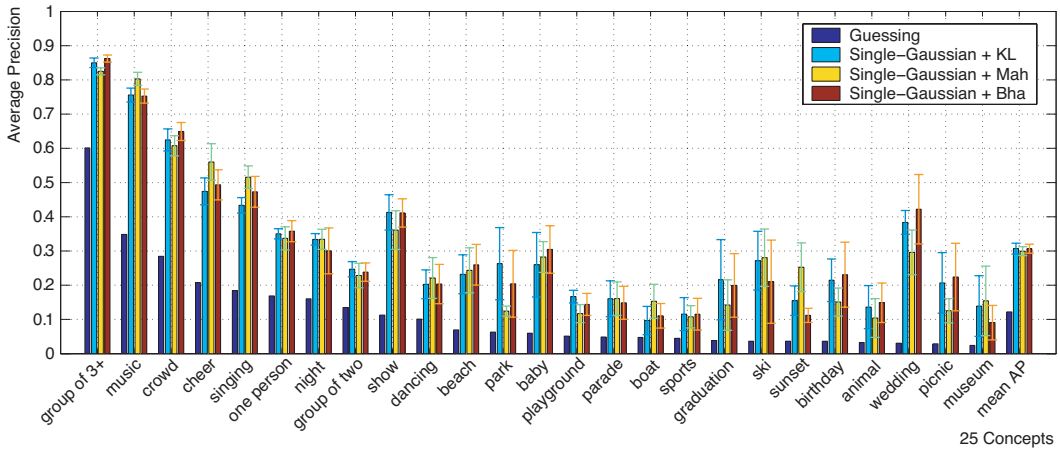

Figure 5.4: Average precision (AP) across all 25 classes for the Single Gaussian models (1G), using each of the three distance measures, KL, Mahalanobis, and Bhattacharyya. Labels are sorted by the guessing baseline performance (shown). Bars and error-bars indicate the mean and standard deviation over 5-fold cross-validation testing respectively.

cepts "group of 3+", "crowd", and "baby" are well detected by 1G+Bha, possibly because human speech plays an important role in discriminating them from other concepts. On average, 1G+KL performs the best among the three distance measures.

Figure 5.5 shows the results for Gaussian Mixture Models with between 2 and 16 Gaussian components per model. Between-model distance is calculated by the approximated Bhattacharyya divergence. Although the optimal number of Gaussian is strongly dependent on the total duration of positive examples of the class, the

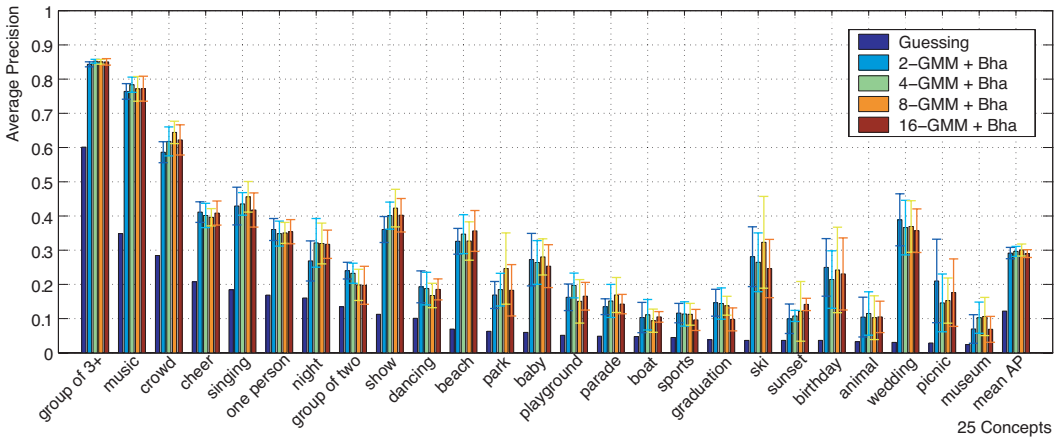

Figure 5.5: As Fig. 5.4, but using Gaussian Mixture models (GMMs) with 2, 4, 8, and 16 components, and approximated Bhattacharyya distance.

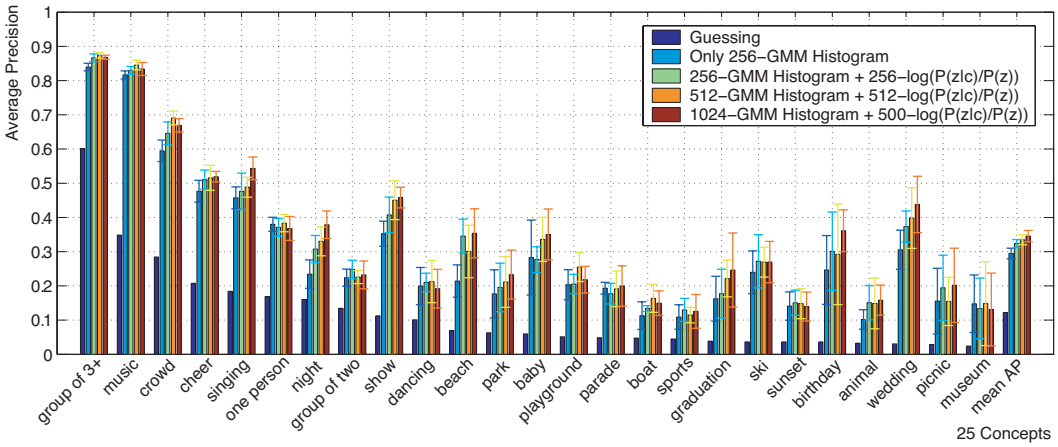

Figure 5.6: As Fig. 5.4, but using pLSA modeling of component-use histograms for GMMs of 256, 512, and 1024 components. Also shown is performance using the 256 component histogram directly, without pLSA modeling.

8-GMM is a good compromise (the best AP), able to capture detail across all the classes.

The performance of the probabilistic Latent Semantic Analysis of the GMM histogram is shown Figures 5.6 and 5.7. To build the gram matrix for the SVM, we tested various summary features, including the raw histogram counts $P(g|c)$ (i.e. without decomposition into pLSA topics), the per-clip topic weights $P(z|c)$,

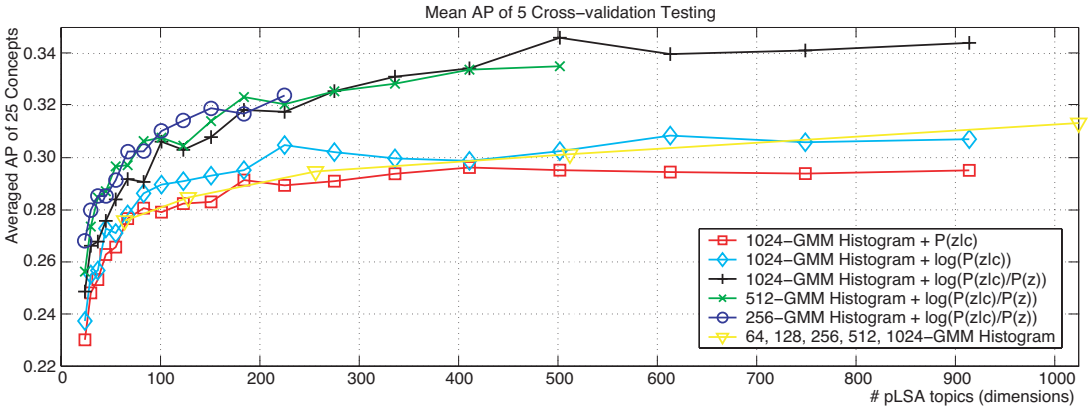

Figure 5.7: AP averaged across all classes for pLSA models using different numbers of 'topics' (latent dimensions) and different treatments for the inferred per-clip topic strengths, $p(z|c)$.

log-topic weights $\log(P(z|c))$, and log-normalized topic weights $\log(P(z|c)/P(z))$. In each case, the gram matrix contained Mahalanobis distances i.e. normalized by the variance of the features across the entire training set. By comparing the three curves for 1024-GMM histograms in Figure 5.7, we see that log-normalized topic weights perform significantly better than the raw histogram or unnormalized weights. As we increase the number of Gaussian components used to build the histograms, we see increasing benefits for the less-frequent (lower-prior) concepts. The best performance is obtained by using around 500 topics to model component use within a 1024 mixture GMM.

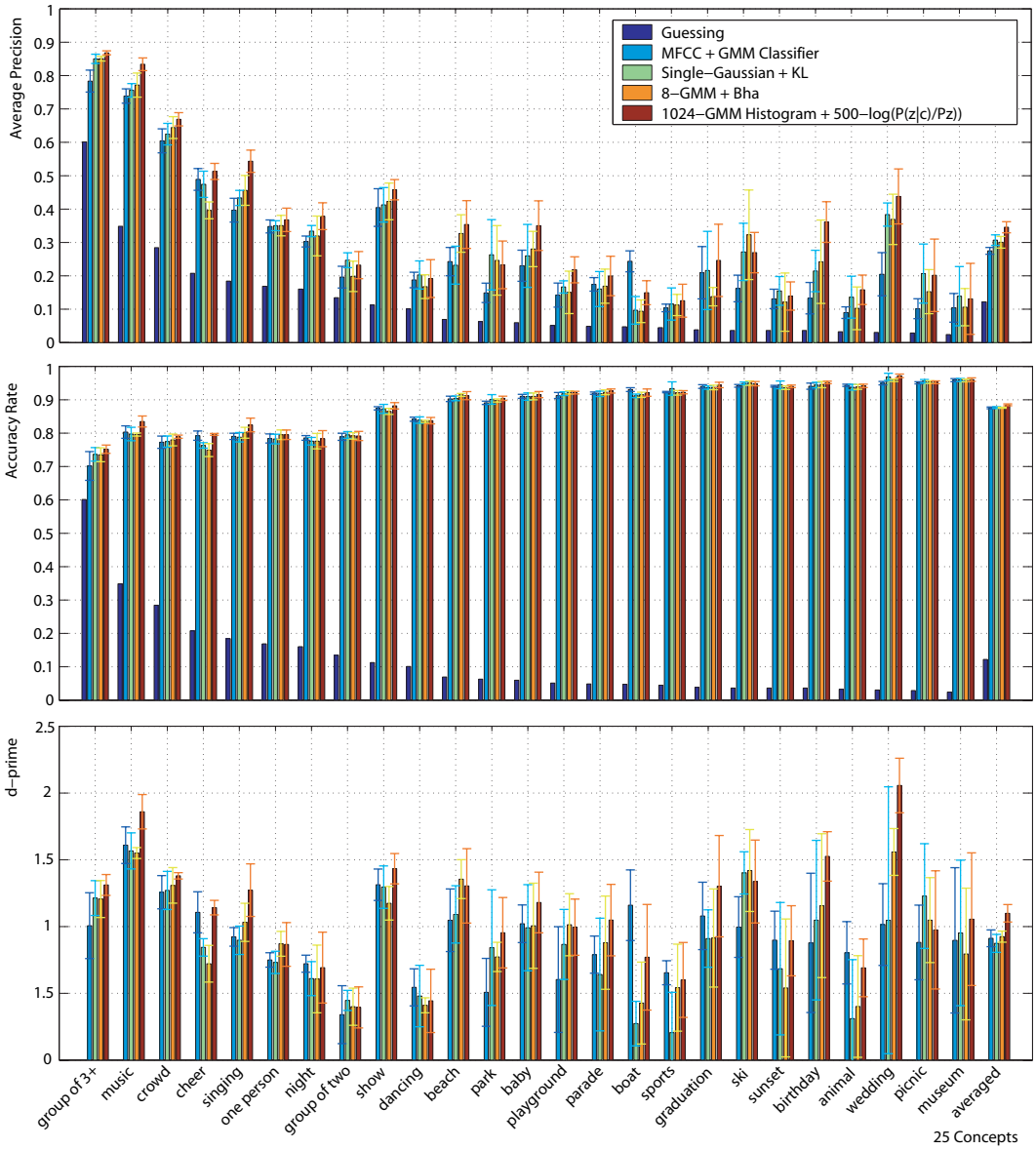

Figure 5.8: The best results from Figs. 5.4, 5.5, and 5.6, illustrating the relative performance of each representation.

5.4 Discussion

Figure 5.8. compares the best results for each of the three modeling approaches, (1G+KL, 8-GMM+Bha, and pLSA-500+lognorm) along with comparison system based on [25]. The comparison system builds an 8-component diagonal-covariance GMM for the MFCC features of clips bearing each label, and ranks items based on the likelihood under that GMM i.e. it lacks the final SVM stage of the other systems. The figure compares the systems in terms of average precision (AP), accuracy rate, and d'. Note that accuracies can be made very high for concepts with small prior probabilities simply by labeling all clips as negative; d' and AP are less vulnerable to this bias. To obtain a hard classification (for accuracy and d' calculation) from our SVM-based rankings, we need to choose a threshold for the distance-to- boundary values. We set this threshold independently for each class at the point at which the number of positive classifications matched the prior of the class.

Most striking is the wide variation in performance by concept, which is to be expected since different labels will be more or less evident in the soundtrack as well as being supported by widely differing amounts of training data. Indeed, the main determinant of performance of these classifiers appears to be the prior likelihood of that label, suggesting that a large amount of training data is the most important ingredient for a successful classifier – although this is confounded by the correspondingly higher baseline. In some cases these factors may be distinguished: a less frequent concept "ski" has AP similar to that of the more frequent concept "beach", suggesting that it is more easily detected from the audio. However, the error bars show that the AP varies much more widely among the 5-fold cross-validation, presumably because a smaller number of positive training examples will lead to less consistency between the different subsets of positive examples chosen in each fold to train the SVM classifier.

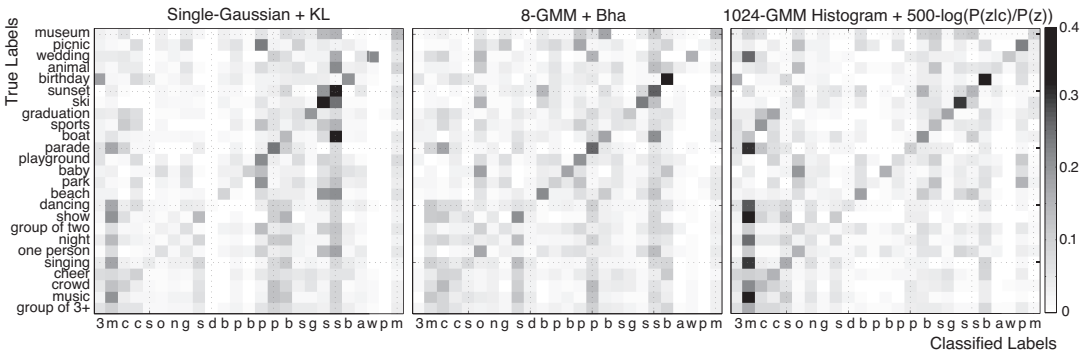

Figure 5.9: Confusion matrix of classified labels within 750 testing clips according to three approaches.

Some concepts consist of a few distinct, representative sounds that may be more successfully modeled by GMMs than by a single Gaussian. For instance, we have noticed that "beach" is mainly composed of two sound types, 'wind' and 'water' sounds; the AP for this concept is noticeably larger with the GMM than with 1G. This also suggests that performance could be improved by dividing some classes into more specific and hence more consistent subclasses (e.g., "animal" refined to "dog" and "cat" etc).

In addition, we have noticed that some concepts such as "cheer", "people", and "music" may be predominantly contained in other concepts such as "birthday", "sports", and "show". It is damaging to use such highly overlapped labels for SVM training with the 1G or GMM approaches because it is impossible to separate pure positive and negative segments at the scale of whole clips. The pLSA model is less sensitive to this problem, since it is able to represent the clip-level summary features directly as combinations of "topics", rather than trying to assign them to a single class. This may explain why its performance, averaged over all classes, appears superior to the other approaches.

Figure 5.9 shows confusion matrices for each classification approach obtained

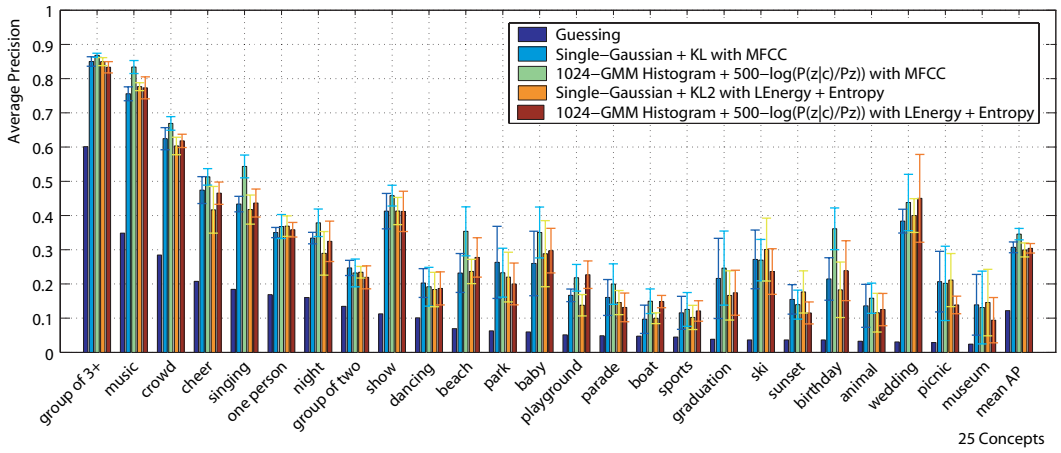

Figure 5.10: The result with MFCC and LEnergy + Entropy.

by assigning each clip to the single class whose SVM gave the largest distance-to-margin, then looking at the distribution of labels assigned to all clips tagged with each specific class to obtain each row of the matrix. Because this approach does not allow the classifier to assign the multiple tags that each clip may bear, perfect performance is not possible and confusions may reflect label co-occurrence as well as imperfect classifiers. The 1G and GMM confusion patterns are more similar to each other than either is to the pLSA approach.

Figure 5.10 shows the result of the 1G and pLSA approaches using MFCC or Log Energy and Entropy features developed for segmenting and clustering the Personal Audio in Chapter 3. On average, both approaches with MFCC achieve a better performance than with the combination of Log Energy and Entropy features.

Figure 5.11 gives example results for detecting the concept "cheer". Most "cheer" clips contain speech, music, and other background sounds that are more predominant than any cheering sound. On average, cheer sounds account for around 28% of the time within corresponding clips.

We have argued that pLSA is successful because it can represent soundtracks as mixtures of "topics" that may correspond to varying kinds of sounds within the

Figure 5.11: Examples of retrieval results for the "cheer" concept. Shown are the top 15 results for each of the best-performing detection systems, 1G+KL2, 8GMM+Bha, and pLSA500+lognorm. Highlighted results are correct according to manual labeling; the number of correct results is shown in the heading for each pane.

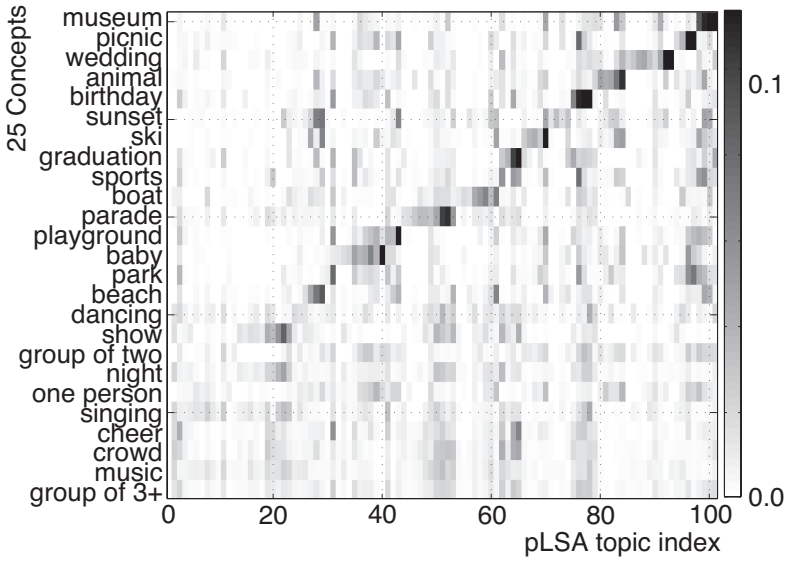

Figure 5.12: Example pLSA topic weights (i.e. $p(z|c)$) across all concepts for a 100-topic model. Topic columns are sorted according to the concept for which they have the largest weight.

overall soundtrack duration. As a possible way to substantiate this, Figure 5.12 shows the weights associated with each class for each of the anonymous topics for a 100 topic model based on 1024 component GMM occupancy histograms. While many pLSA topics are strongly identified with a single concept, many others make significant contributions to several classes, such as topics 26 to 28 that occur in both "beach" and "sunset", or topics 96 and 97 that contribute to "park" and "picnic". The conjecture is that these topics correspond to the GMM states that cover the common sounds that occur in these classes; however, this needs to be confirmed by a closer examination of the time frames corresponding to the GMM states associated with these topics.

While the pLSA approach gives consistently the best results, the margin of improvement is relatively small and might not be important in some applications. The

baseline single-Gaussian, or likelihood-based GMM systems perform relatively well in comparison and are much simpler to construct and to evaluate. Thus, depending on the nature of the database and the value of the highest possible precision, these may be valid approaches. However, this pattern could change with larger training databases and needs to be reevaluated.

Figure 5.13 shows the distribution of 25 concepts over 2-dimensional space projected by LDA (Linear Discriminant Analysis). The covariance matrix of each concept is calculated on a set of MFCC or pLSA (500-dimensional $\log(P(z|c)/P(z))$) frames randomly selected from clips labeled with the appropriate concept. The "dancing" and "park" concepts are less overlapped with other in MFCC feature space than in pLSA feature space, so their performance are better with MFCC. On the other hand, "playground" and "wedding" concepts have better results with pLSA approach because they are more discriminant in the pLSA feature space.

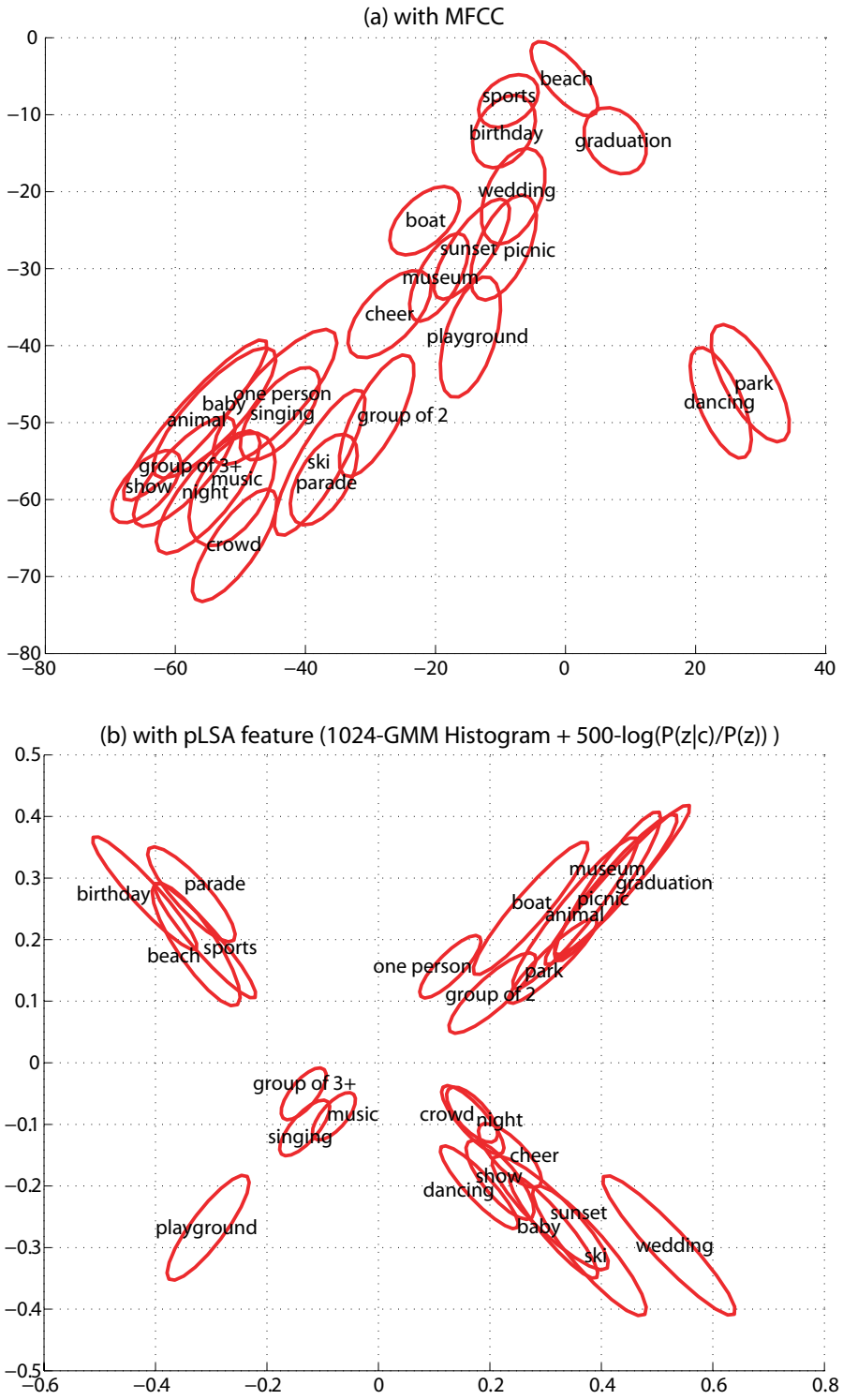

Figure 5.13: LDA projection of 25 concepts into two-dimensional subspace.

5.5 Summary

We have described several variants of a system for classifying consumer videos into a number of semantic concept classes, based on features derived from their soundtracks. Specifically, we have experimented with various techniques for summarizing low-level MFCC frames into fixed-size clip-level summary features, including Single Gaussian Models, Gaussian Mixture Models, and probabilistic Latent Semantic Analysis of the Gaussian Component Histogram. We constructed SVM classifiers for each concept using the Kullback-Leibler, Bhattacharyya, and Mahalanobis distances. In spite of doubts over whether soundtrack features can be effective in determining content classes such as "picnic" and "museum" that do not have obvious acoustic correlates, we show that our classifiers are able to achieve APs far above chance, and in many cases at a level likely to be useful in real retrieval tasks.

Chapter 6

HMM-based Local Concept Detection

In this chapter, we develop a novel MIL approach, a Markov model-based clustering algorithm able to segment a set of temporal frames into regions associated with different ground-truth labels tagged at the clip level, and at the same time to exclude uninformative "background" frames shared in common from all clips.

In the next section, we describe detecting multiple local concepts from global annotation using Markov models. Evaluation, discussion and summary are presented in section 6.2, 6.3 and 6.4 respectively.

6.1 Detecting multiple Local Concepts From Global Annotations

Our system starts with a basic frame-level feature, the Mel-frequency Cepstral Coefficients (MFCCs) that are commonly used in speech recognition and other acoustic classification tasks. The single-channel (mono) soundtrack of a video is first resam-

pled to 8kHz, and then a short-time Fourier magnitude spectrum is calculated over 25ms windows every 10ms. The spectrum of each window is warped to the Mel frequency scale, and the log of these auditory spectra is decorrelated into MFCCs via a discrete cosine transform. After the initial MFCC analysis, each video's soundtrack is represented as a set of $d = 21$ dimensional MFCC feature vectors.

We then train a hidden Markov model with Gaussian mixture emission models to learn the concepts. Each concept is a distinct state in the model, and in addition one or more "global background" states are included. The assumption here is that each feature vector can be associated with a particular concept (state), but through the time sequence of features in an entire clip, multiple different concepts may be expressed. The model is learned via conventional Baum-Welch Expectation Maximization (EM), but for each clip the transition matrix is modified to ensure that only the global background states and the states for the concepts specified in the clip-level labeling of that video will be updated; transitions to all other states are set to zero. Figure 6.1 uses a 3-state Markov model to illustrate this idea.

The training process maximizes the likelihood of all frames using only the states allowed by the relevant clip-level annotations (and the global background states). It should result in states being used to model frames that are most relevant to those labels, with less informative frames being absorbed by the background models. Thus, the procedure achieves both clustering of frames that relate to each state, and produces a model that can be used to identify relevant sounds in test examples. This process is described in more detail in the next section.

6.1.1 HMM-based Clustering

The hidden Markov model (HMM) assumes that each feature vector is generated by a particular state, meaning that it reflects a particular concept, but the states will

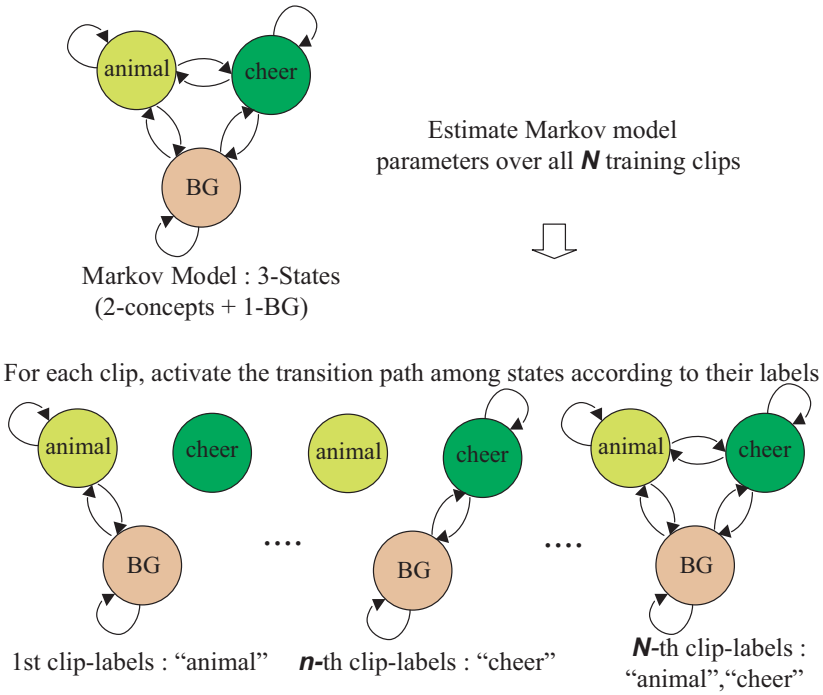

Figure 6.1: Example of clustering consistent a set of temporal frames into segments corresponding to each concept using the first-order 3-state Markov process.

change with time. The HMM is parameterized by a set of parameters, $\theta = \{\pi, A, \phi\}$ where π, A and ϕ indicate the prior, transition and emission probabilities of states.

We begin by considering a single clip n. Let assume that C_n denotes a $K-$dimensional annotation vector for a clip n in which each component, $C_n(k) \in \{0, 1\}$ for $k = 1, \cdots, K$, indicates the presence or absence of the k^{th} concept tagged by a human, and the K is the total number of concepts. Each concept can be present or absent independently in a clip. In our system, we annotated each training clip with 25 concepts as described in the section 6.2. We add 1, 2, or 4 states for the global background whose labels are set to be true (1) for all training clips; adding more background states allows for greater variety for this category, which we expect to account for the majority of the data. Thus, K is 26, 27 or 29.

The $K \times K$- dimensional transition matrix A is controlled by the ground truth annotation C_n of a clip n to be able to selectively train only the parameters of states whose corresponding concepts appear in C_n. The transition matrix A_n of clip n is modified from the original A so that:

$$A_n(i,j) = A(i,j), \;\; iff \; C_n(i) \; and \; C_n(j) \; == \; 1. \tag{6.1}$$

All other values are set to zero. $A_n(i,j)$ is then normalized by rows to satisfy

$$\sum_{j=1}^{K} A_n(i,j) = 1. \tag{6.2}$$

The remaining process is to estimate the parameters, $\theta = \{\pi, A, \phi\}$, using the EM (Expectation Maximization) algorithm for all N training clips. Assume that X_n denotes the observations for clip n comprising a set of MFCC feature vectors $\{x_{nt}\}$ for $t = 1, \cdots, T_n$, where T_n is the total number of frames in clip n and depends on the duration of the original video.

For every clip, we apply the forward-backward algorithm on X_n, with the corresponding modified transition matrix A_n, to evaluate the marginal posterior distribution $\gamma(z_{ntk})$ of the latent variable z_{ntk} which indicates that frame t of clip n was emitted by state k. We also estimate the joint posterior distribution $\xi(z_{n,t-1,k}, z_{ntk})$ of two successive latent variables in the E-step. The parameters $\theta = \{\pi, A, \phi\}$ are updated for the M-step:

$$\pi_k = \frac{\sum_{n=1}^{N} \gamma(z_{n1k})}{\sum_{j=1}^{K} \sum_{n=1}^{N} \gamma(z_{n1j})} \tag{6.3}$$

$$A_{jk} = \frac{\sum_{n=1}^{N}\sum_{t=2}^{T_n}\xi(z_{n,t-1,j},\ z_{ntk})}{\sum_{l=1}^{K}\sum_{n=1}^{N}\sum_{t=2}^{T_n}\xi(z_{n,t-1,j},\ z_{ntl})} \tag{6.4}$$

with the π parameters set analogously. The k^{th}'s state emission probability, $p(X;\phi_k)$, is modeled by an M-component Gaussian mixture model (GMM):

$$p(X;\phi_k) = \sum_{m=1}^{M} w_{km} N(X|\mu_{km}, \Sigma_{km}) \tag{6.5}$$

The posterior probability of m^{th} component of GMM is given by

$$\tau_{ntkm} = \frac{w_{km} N(x_{nt}|\mu_{km}, \Sigma_{km})}{\sum_{m'=1}^{M} w_{km'} N(x_{nt}|\mu_{km'}, \Sigma_{km'})} \tag{6.6}$$

Per-component weights w_{km}, means μ_{km}, and covariances Σ_{km} are also updated using $\gamma(z_{ntk})$.

$$\mu_{km}^{new} = \frac{\sum_n \sum_t \gamma(z_{ntk})\tau_{ntkm} x_{nt}}{\sum_n \sum_t \gamma(z_{ntk})\tau_{ntkm}} \tag{6.7}$$

$$\Sigma_{km}^{new} = \frac{\sum_n \sum_t \gamma(z_{ntk})\tau_{ntkm}(x_{nt} - \mu_{km}^{new})(x_{nt} - \mu_{km}^{new})^T}{\sum_n \sum_t \gamma(z_{ntk})\tau_{ntkm}} \tag{6.8}$$

$$w_{km}^{new} = \frac{\sum_n \sum_t \gamma(z_{ntk})\tau_{ntkm}}{\sum_n \sum_t \gamma(z_{ntk})} \tag{6.9}$$

We use $M = 16$ for each state. The GMMs are initialized with a set of MFCC frames randomly selected from clips labeled with the appropriate concept.

After learning the HMM given the clip-level labels, the Viterbi algorithm is used

to find the most probable sequence of states for a given sequence of MFCC frame in each testing clip as shown in Figure 6.2.

6.2 Evaluations

We tested our HMM-based clustering algorithm on the soundtracks of 1,873 videos clips and the hand-labeled 25 concepts used in our previous work on consumer video classification described in the Chapter 5 in detail.

As shown in Figure 6.2, after Viterbi decoding each frame is assigned to one of 27 concepts (25 primary plus two background states), and most voiced frames are assigned into the global background (BG). Owing to the limitations of visual-based annotation, the speech from unseen people in a scene (e.g. narration from a person who is recording a video, or voices from the TV at home) is often not explicitly labeled, and so voice tends to fall into the global background as a sound common to all clips regardless of label.

To evaluate frame level performance, we further annotated the soundtracks of four object-related concepts (animal, baby, boat and cheer) to indicate the precise time segments that contain the sounds of those objects. The overall frame-level performance on this test data is presented in table 6.1 and 6.2 in terms of the frame-level accuracy, d' and Average Precision (AP). The accuracy rate is the proportion of 10 ms frames correctly labeled; d' is a threshold-independent measure of the separation between the two classes (presence and absence of the label) when mapped to two unit-variance Gaussian distributions, and AP is the Average of Precisions calculated separately for each true frame. Note that accuracy figures are high since in most cases there is a strong prior probability that any frame is negative (no relevant sound), so even labeling all frames negative would achieve high accuracy; d' and AP are less vulnerable to this bias. We used 5-fold cross-validation to evaluate

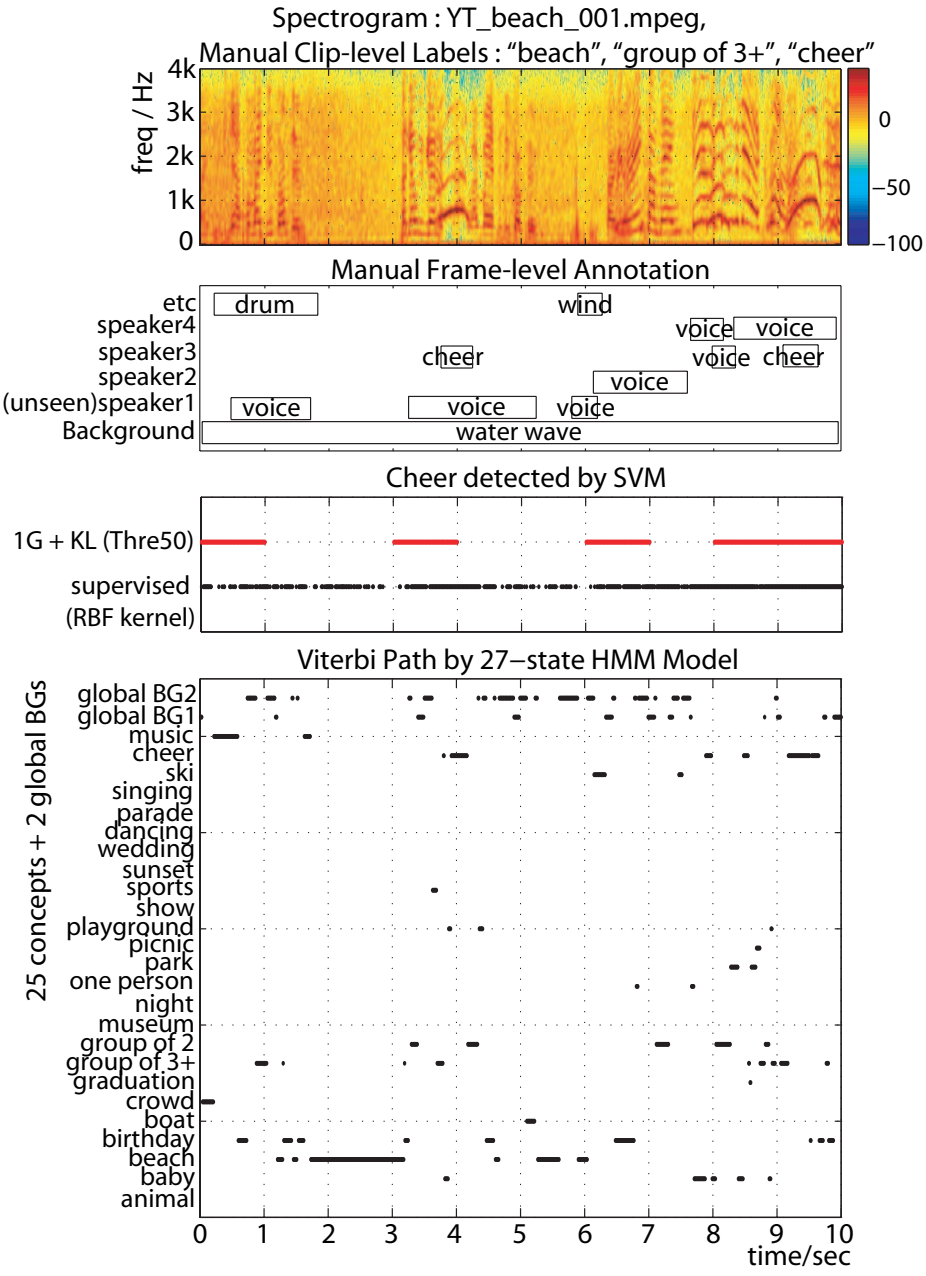

Figure 6.2: Example analysis of a soundtrack consisting of a conversation at the beach. Speech is clustered into the global background, but cheers, and background beach noises are correctly identified.

Concept	# with sound	Avg. Dur.	Prior (frames)		SVM with RBF kernel	HMM Clustering		
						26S	27S	29S
animal	21/61 clips	8.0s	0.22%	acc.	74.8%	98.1%	98.5%	**99.2**%
				d'	0.24	0.29	**0.38**	0.12
				AP	0.2%	0.25%	**0.35**%	0.25%
baby	43/112 clips	7.3s	0.4%	acc.	86%	96.9%	97.3%	**97.7**%
				d'	1.12	1.2	1.26	**1.3**
				AP	**4.5**%	3%	2.7%	3.5%
boat	41/89 clips	30.8s	1.62%	acc.	92.7%	97.3%	97.7%	**97.9**%
				d'	0.88	**1.47**	1.34	1.24
				AP	5.4%	**12.2**%	11.2%	9.1%
cheer	388/388 clips	5.1s	2.44%	acc.	46.8%	94.8%	95.2%	**95.4**%
				d'	1.62	**1.92**	1.91	**1.92**
				AP	20.2%	29.5%	29.5%	**29.8**%

Table 6.1: *Supervised (using frame-scaled hand-labels) concept classification performance on YouTube videos. The second column indicates how many of the clips tagged with the concept actually contained relevant sounds; the third column gives the average duration of relevant sound within those clips. The fourth column shows the (frame-scale) prior of these concepts. Values in columns* 6 *through* 9 *represent means of the frame-level performance over* 5 *cross-validated experiments. Concepts are evaluated with accuracy, d′, and average precision (AP), and the best performance of each row is shown in bold. Note that accuracy rate isn't the good measure of performance when the prior of non-concept's frames is severely high. Different columns indicate different experimental conditions, as explained in the text.*

the performance: At each fold, the classifier is trained on 40% of the data, tuned on 20%, and then tested on the remaining 40% selected at random at the clip level.

Table 6.1 show the concept classification result when two classifiers (SVM and HMM-based clustering) are learned in a supervised way. The SVM classifier with using RBF kernel is trained on a set of frames manually annotated with the corresponding concept, and then tested on testing videos with using a zero as threshold of deciding whether a concept or not. In HMM clustering system, the GMMs are also initialized with a set of hand-labeled true frames of each concept. Severely biased prior between concept and non-concept frames, i.e., 2.4% for cheer and 97.6% for

non-cheer, gives a negative effect on correctly training a 2-way SVM classifier, so a lot of non-concept frames are wrongly detected into a concept as shown in the third panel of the Figure 6.2. The HMM-based clustering system outperforms a SVM classifier in that a concept (corresponding to single state) can be successfully discriminated from other remaining many concepts.

For comparison, we also report the frame-level performance of the '1G+KL with SVM' system from [10], which trains an SVM classifier using a symmetrized Kullback-Leibler (KL) distance calculated on single, full-covariance Gaussian distributions fit to MFCC features over the entire clip. Here, to get a comparable sub-clip level time labeling, we divide the soundtrack into 1 s segments and classify each one. The resulting distance-to-boundary values from the SVM are shifted due to the change of segment's length, so we try several different thresholds. The "Thre0" column in the Table 6.2 gives the results when classification is based on the standard SVM threshold of 0, which show the negative impact of this shift. Thus, we experiment with various other set the threshold, shown in the subsequent columns: "Thre50" sets the threshold at the 50^{th} percentile of the values within the clip, meaning that exactly half the labels in each test clip will be labeled positive. "Thre26S", "Thre27S", and "Thre29S" instead choose the percentile as the actual number of frames detected by the HMM-based system with the corresponding number of states, as an upper-bound comparison. The posterior probabilities of frames calculated through the Viterbi decoding are used for evaluating the AP in HMM-based clustering system.

6.3 Discussion

The HMM based system results (using a clip-level annotation) are given in Table 6.2, for systems with 26, 27, or 29 states (i.e. 1, 2, or 4 background states). In-

correct annotations (i.e. clips labeled with a concept that contain zero soundtrack frames relevant to the concept because the object makes no sound) are a major factor degrading performance, and we see that performance varies depending on the proportion of tagged clips that contain relevant audio frames (column 2). The animal concept, which has the worst result of $d' = 0.65$, contains relevant sound in only 34% (21/61) clips. Performance improves as the proportion of clips containing relevant sounds increases. Thus, "baby" with $43/112 = 38\%$ relevant-sounding clips has $d' = 0.92$, "boat" (46%) has $d' = 1.35$, and "cheer" (100%) has $d' = 1.76$.

Another factor determining performance is the consistency of representative sounds for each concept. The "animal" concept covers many kinds of animal (e.g. dog, cat, fish etc.), and tends to have a very broad range of corresponding content. Compared with "animal", "baby" is better because baby sounds (e.g. crying and laughing) are more specific than animal sounds. In the case of "boat", the representative sound is the relatively consistent engine noise, and a large proportion (46%) of relevant clips contain it, leading to much better overall performance.

The best performance of the HMM system occurs for cheering segments. We infer this is because the cheer concept is conveyed by acoustic information (leading to correct annotations), and its sound is consistent between different clips. The performance of our HMM system with a random initialization ($d' = 1.76$ and $AP = 25.5\%$ for cheer detection) is significantly better than a supervised learning method, 'SVM with RBF kernel' ($d' = 1.62$ and $AP = 20.2\%$), and even is much similar to manually initialized HMM system ($d' = 1.91$ and $AP = 29.5\%$). This provides the best illustration of the success of our HMM-based clustering in detecting local objects.

To inspect the effect of the transitions between states (concepts), we compare an original transition trained on a training data with ones in which off-diagonal values

Concept		"1G+KL"+ SVM				
		Thre0	Thre50	Thre26S	Thre27S	Thre29S
animal	acc.	36.6%	72.9%	98.5%	98.6%	**99.0**%
	d'	0.15	0.47	0.38	0.31	0.3
	AP	**0.67**%				
baby	acc.	99.6%	50.3%	96.9%	97.1%	97.9%
	d'	0	**1.1**	0.65	0.71	0.59
	AP	0.73%				
boat	acc.	58.8%	50.7%	96.5%	97.1%	97.6%
	d'	0.47	0.61	0.17	0.13	0.2
	AP	2.23%				
cheer	acc.	97.6%	52.0%	92.7%	93.2%	93.5%
	d'	0.15	1.38	0.1	0.11	0.11
	AP	4.37%				

Concept		HMM Clustering					
		trained Transition			uniform Transition		
		26S	27S	29S	26S	27S	29S
animal	acc.	98.4%	98.5%	**98.9**%	98.3%	98.8%	**98.9**%
	d'	0.57	**0.65**	0.3	0.54	0.56	0.36
	AP	0.47%	0.52%	0.37%	**0.53**%	0.51%	0.39%
baby	acc.	97.0%	97.2%	**98.0**%	97.0%	97.3%	**98.0**%
	d'	0.93	0.92	0.96	0.92	0.94	**0.97**
	AP	1.64%	1.83%	1.65%	1.73%	**2.01**%	1.66%
boat	acc.	97.1%	97.6%	**97.9**%	97.0%	97.7%	**97.9**%
	d'	1.3	1.35	1.3	1.28	**1.38**	1.34
	AP	9.7%	10.8%	9.0%	10.6%	**11.2**%	9.8%
cheer	acc.	95.0%	95.3%	95.4%	95.0%	95.3%	**95.5**%
	d'	1.77	1.76	1.72	**1.78**	1.77	1.73
	AP	25.4%	25.5%	24.1%	26.3%	**26.5**%	24.4%

Table 6.2: *Semi-supervised (using clip-scaled hand-labels) concept classification performance on YouTube videos. Values in columns 3 through 8 represent means of the frame-level performance over 5 cross-validated experiments. Different columns indicate different experimental conditions, as explained in the text.*

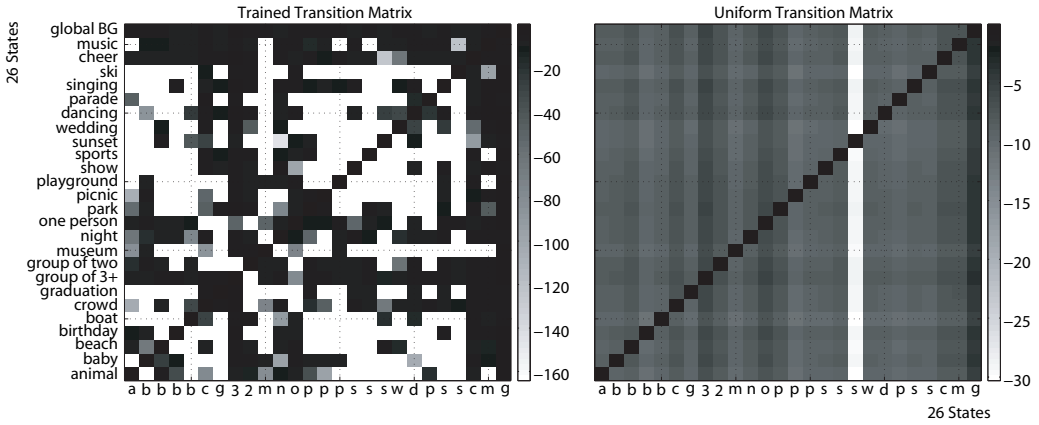

Figure 6.3: Log-scaled transition matrix trained and modified.

A_{jk} (transition probabilities among different states) of each state are modified to be a value of $\pi_k(1 - A_{kk})$ as shown in Figure 6.3. The performances with an original and modified transition are almost same in Table 6.2. The transition probabilities may be underestimated due to the incorrect annotation.

6.4 Summary

In this chapter, we develop Markov model-based clustering in order to segment consistent sets of temporal frames into regions associated with different ground-truth labels tagged at the clip level, and at the same time excluding uninformative "background" frames shared by all clips in common. Quantitative evaluation shows that local concepts are effectively detected by this clustering technique even based only on coarse clip-level labels, and that detection performance is significantly better than existing algorithms for real-world consumer recordings.

Chapter 7

Conclusions

In this thesis, we described a vision of 'environmental' audio archives. It is central for human experience and culture to record an individual's daily life as a medium for preserving and recollecting events and facts. With the availability of devices making these kinds of 'environmental' recordings at low cost, with high reliability, and with minimal impact to the individual, people can easily collect and share a large collection of personal recordings. Moreover, these recordings contain much richer information closely related with human life, and consequently present many new opportunities for the automatic extraction of information that can be used in intelligent browsing systems. Therefore, we are particularly interested in exploiting the acoustic information, and in seeing what useful information can be reliably extracted from these kinds of data.

The segmenting/clustering algorithm for continuous long-duration personal audio archives is first presented. This is to provide automatic indexing based on the statistics of frequency-warped short-time energy spectra calculated over windows of seconds or minutes. Our automatically clustered segments can be grouped into similar or recurring classes which, once the unknown correspondence between automatic

and ground-truth labels is resolved, gives frame-level accuracies of over 80% on our 62 h hand-labeled test set.

We also proposed a robust pitch detection algorithm for identifying the presence of speech or music in the noisy, highly-variable personal audio collected by body-worn continuous recorders. In particular, we have introduced a new technique for estimating and suppressing stationary periodic noises such as air-conditioning machinery in the autocorrelation domain. The performance of our proposed algorithm is significantly better than existing speech or music detection systems for the kinds of data we are addressing.

In addition, we have described several variants of a system for classifying consumer videos into 25 semantic concept classes, based on features derived from their soundtracks. Specifically, we have experimented with various techniques for summarizing low-level MFCC frames into fixed-size clip-level summary features, including Single Gaussian Models, Gaussian Mixture Models, and probabilistic Latent Semantic Analysis of the Gaussian Component Histogram. We constructed SVM classifiers for each concept using the Kullback-Leibler, Bhattacharyya, and Mahalanobis distances. In spite of doubts over whether soundtrack features can be effective in determining content classes such as "picnic" and "museum" that do not have obvious acoustic correlates, we show that our classifiers are able to achieve APs far above chance, and in many cases at a level likely to be useful in real retrieval tasks.

Finally, we develop the HMM-based clustering algorithm in order to segment consistent set of temporal frames into regions associated with different ground-truth labels tagged at the clip level, and at the same time to exclude a set of uninformative frames shared in common from all clips. Quantitative evaluation shows that local concepts are effectively detected by this clustering technique even based on the coarse labeling scheme, and that detection performance is significantly better than

existing algorithms in the real-world consumer recordings.

For future works, we will test the hmm-based clustering on varying longer-time windowed features and other features such as Log-energy and Entropy. Based on the model of concepts estimated through this HMM clustering, we will try to separate foreground and background sounds. Definition of concepts more directly related to audio and integration with video information will be helpful to improve the performance of detecting semantics in real-world recordings.

Bibliography

[1] National institutes of science and technology. meeting recognition diarization evaluation. 2005. (Cited on page 12.)

[2] Youtube - broadcast yourself. 2006. (Cited on pages 3, 63, and 70.)

[3] J. Ajmera, I. McCowan, and H. Bourlard. Speech/music segmentation using entropy and dynamism features in a hmm classification framework. *Speech Communication*, 40:351–363, 2003. (Cited on page 14.)

[4] AquaMinds Software. NoteTaker: An outlining program, 2003. (Cited on page 42.)

[5] J. Arenas-Garcia, A. Meng, K. Petersen, T. Lehn-Schioler, L. Hansen, and J. Larsen. Unveiling music structure via plsa similarity fusion. In *IEEE Workshop on Machine Learning for Signal Processing*, pages 419–424, Thessaloniki, Aug. 2007. (Cited on page 79.)

[6] Sudha. V B. Thoshkahna and K. R. Ramakrishnan. A speech-music discriminator using hiln model based features. In *Proc. IEEE Int. Conf. on Acous., Speech, and Sig. Proc. (ICASSP)*, France, May 2006. (Cited on page 14.)

[7] J. Brassil. Using mobile communications to assert privacy from video surveillance. In *Proc. 1st Intl. Workshop on Security in Systems and Networks*, April 2005. (Cited on page 44.)

[8] V. Bush. As we may think. *The Atlantic Monthly*, July 1945. (Cited on page 10.)

[9] S.-F. Chang, D. Ellis, W. Jiang, K. Lee, A. Yanagawa, A. Loui, and J. Luo. Kodak consumer video benchmark data set: concept definition and annotation. In *MIR workshop, ACM Multimedia*, Germany, Sep. 2007. (Cited on pages 7, 17, and 73.)

[10] S.-F. Chang, D. Ellis, W. Jiang, K. Lee, A. Yanagawa, A. Loui, and J. Luo. Largr-scale multimodal semantic concept detection for consumer video. In *MIR*

workshop, ACM Multimedia, Germany, Sep. 2007. (Cited on pages 7, 64, 69, and 103.)

[11] S. Chen and P. Gopalakrishnan. Speaker, environment and channel change detection and clustering via the Bayesian Information Criterion. In *Proc. DARPA Broadcast News Transcription and Understanding Workshop*, 1998. (Cited on pages 8, 12, 20, 26, and 27.)

[12] Y. Chen and Z. Wang. Image categorization be learning and reasoning with regions. In *Journal of Machine Learning Research*, 2004. (Cited on page 18.)

[13] S. Chu, S. Narayanan, and C.-C. J. Kuo. Content analysis for acoustic environment classification in mobile robots. In *AAAI Fall Symposium, Aurally Informed Performance: Integrating Machine Listening and Auditory Presentation in Robotic Systems*, pages 16–21, 2006. (Cited on page 16.)

[14] B. Clarkson, N. Sawhney, and A. Pentland. Auditory context awareness via wearable computing. In *Proc. Perceptual User Interfaces Workshop*, 1998. (Cited on pages 4 and 11.)

[15] B. P. Clarkson. *Life patterns: structure from wearable sensors.* PhD thesis, MIT Media Lab, 2002. (Cited on page 11.)

[16] Digital Innovations. The Neuros digital audio computer, 2003. (Cited on page 31.)

[17] A.R. Doherty, A. F. Smeaton, K. Lee, and D. P. W. Ellis. Multimodal segmentation of lifelog data. In *Proc. RIAO 2007-Large-scale Semantic Access to Content(Text, Image, Video and Sound)*, Pittsburgh, USA, June 2007. (Cited on page 7.)

[18] W. Du and M. J. Atallah. Privacy-preserving co-operative statistical analysis. In *Proc. 17th Annual Computer Security Applications Conf.*, pages 102–110, New Orleans, Louisiana, USA, December 10-14 2001. (Cited on page 44.)

[19] D. P. W. Ellis. The weft: A representation for periodic sounds. In *Proc. IEEE Int. Conf. Acous., Speech, and Sig. Proc.*, pages II–1307–1310, 1997. (Cited on page 47.)

[20] D. P. W. Ellis. Beat tracking with dyanmic programming. In *MIREX 2006 Audio Beat Tracking Contest*, Sep. 2007. (Cited on page 63.)

[21] D. P. W. Ellis and K. Lee. Features for segmenting and classifying long-duration recordings of "personal" audio. In *Proc. ISCA Tutorial and Research Workshop on Statistical and Perceptual Audio Processing SAPA-04*, pages 1–6, Jeju, Korea, October 2004. (Cited on page 7.)

[22] D. P. W. Ellis and K. Lee. Minimal-impact audio-based personal archives. In *Proceedings of the 1st ACM Workshop on Continuous Archival and Retrieval of Personal Experiences*, New York, NY, October 2004. (Cited on pages 7 and 73.)

[23] D. P. W. Ellis and K. Lee. Accessing minimal-impact personal audio archives. *IEEE MultiMedia*, 13(4):30–38, Oct-Dec 2006. (Cited on page 7.)

[24] D. P. W. Ellis and J. Liu. Speaker turn segmentation based on between-channel differences. In *Proceedings of NIST Meeting Recognition Workshop*, Montreal, Mrach 2004. (Cited on page 12.)

[25] A. Eronen, V. Peltonen, J. Tuomi, A. Klapuri, S. Fagerlund, T. Sorsa, G. Lorho, and J. Huopaniemi. Audio-based context recognition. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(1):321–329, Jan 2006. (Cited on pages 16, 17, and 87.)

[26] M. Flynn, 2004. Personal communication. (Cited on page 5.)

[27] J. Foote. Content-based retrieval of music and audio. In *Proc. SPIE*, pages vol. 3229, pp. 138–147, 1997. (Cited on page 15.)

[28] E. Freeman and D. Gelernter. Lifestreams: A storage model for personal data. *SIGMOD Record (ACM Special Interest Group on Management of Data)*, 25(1):80–86, March 1996. (Cited on page 40.)

[29] J. Gemmel, G. Bell, R. Lueder, S. Drucker, and C. Wong. MyLifeBits: Fulfilling the Memex vision. In *Proc. ACM Multimedia*, pages 235–238, Juan-les-Pins, France, Dec 2002. (Cited on page 11.)

[30] G. Guo and S. Z. Li. Content-based audio classification and retrieval by support vector machines. *IEEE Tr. on Neural Networks*, 14(1):209–215, 2003. (Cited on page 16.)

[31] J. Hershey and P. Olsen. Variational bhattacharyya divergence for hidden markov models. In *Proc. IEEE Int. Conf. on Acous., Speech, and Sig. Proc. (ICASSP)*, Las Vegas, Nevada, April 2008. (Cited on page 77.)

[32] T. Hoffmann. Probabilistic latent semantic idexing. In *Proc. 1999 Int. Conf. on Research and Development in Information Retrieval(SIGIR'99)*, Berkeley, CA, August 1999. (Cited on page 79.)

[33] T. Jebara, R. Kondor, and A. Howard. Probability product kernels. *Journal of Machine Learning Research, JMLR, Special Topic on Learning Theory.*, pages 819–844, 2004. (Cited on page 74.)

[34] J. D. Johnston. Transform coding of audio signals using perceptual noise criteria. *IEEE J. Selected Areas in Comm.*, 6(2):314–323, Feb 1988. (Cited on page 23.)

[35] S. Karneback. Expanded examinations of a low frequency modulation feature for speech/music discrimination. In *Proc. Int. Conf. on Spoken Language Processing (ICSLP)*, USA, Sep. 2002. (Cited on page 14.)

[36] T. Kemp, M. Schmidt, M. Westphal, and A. Waibel. Strategies for automatic segmentation of audio data. In *Proc. IEEE Int. Conf. on Acous., Speech, and Sig. Proc. (ICASSP)*, pages III–1423–1426, Istanbul, 2000. (Cited on page 12.)

[37] M. Lamming and M. Flynn. Forget-me-not: Intimate computing in support of human memory. In *Proc. FRIEND21, 1994 Int. Symp. on Next Generation Human Interface*, Meguro Gajoen, Japan, 1994. (Cited on pages 4 and 44.)

[38] K. Lee and D. P. W. Ellis. Voice activity detection in personal audio recordings using autocorrelogram compensation. In *Proc. Proc. Interspeech*, Pittsburgh, 2006. (Cited on pages 7 and 59.)

[39] K. Lee and D. P. W. Ellis. Detecting music in ambient audio by long-window autocorrelation. In *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 9–12, Las Vegas, Apr 2008. (Cited on page 7.)

[40] K. Lee and D. P. W. Ellis. Audio-based semantic concept classification for consumer video. *Submitted to Transactions on Audio, Speech and Language Processing*, 2009. (Cited on page 7.)

[41] R. Lienhart and M. Slaney. plsa on large scale image database. In *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Hawaii, USA, April 2007. (Cited on page 79.)

[42] A. C. Loui and et al. Kodak consumer video benchmark data set : Concept definition and annotation. In *ACM Multimedia Information Retrieval Workshop*, Sept 2007. (Cited on page 69.)

[43] L. Ma and B. Milnerand D. Smith. Acoustic environment classification. *ACM Trans. Speech Lang. Pro.*, 3(2):1–22, 2006. (Cited on page 16.)

[44] R. G. Malkin and A. Waibel. Classifying user environment for mobile applications using linear autoencoding of ambient audio. In *Proc. IEEE Int. Conf. on Acous., Speech, and Sig. Proc. (ICASSP)*, Pittsburgh, PA, USA, Mar. 2005. (Cited on page 16.)

[45] M. I. Mandel and D. P. W. Ellis. Song-level features and support vector machines for music classification. In *Proc. International Conference on Music Information Retrieval ISMIR*, pages 594–599, London, Sep 2005. (Cited on page 75.)

[46] M. I. Mandel and D. P. W. Ellis. Multiple-instance learning for music information retrieval. In *Proc. International Conference on Music Information Retrieval ISMIR*, pages 577–582, Philadelphia, Sep 2008. (Cited on page 18.)

[47] M. I. Mandel and D. P. W. Ellis. A web-based game for collecting music metadata. *J. New Music Research*, 37(2):151–165, 2008. (Cited on page 17.)

[48] S. Mann. Wearable computing: A first step toward personal imaging. *IEEE Computer Magazine*, pages 25–32, Feb 1997. (Cited on page 11.)

[49] O. Maron and T. Lozano-Perez. A framework for multiple instance learning. In *Neural Information Processing Systems*, 1998. (Cited on page 18.)

[50] F. Monay and D. Gatica-Perez. Plsa-based image auto-annotation: constraining the latent space. In *Proc. of the 12th annual ACM international conference on Multimedia*, New York, NY, USA, Oct. 2004. (Cited on page 79.)

[51] N. Morgan, D. Baron, J. Edwards, D. Ellis, D. Gelbart, A. Janin, T. Pfau, E. Shriberg, and A. Stolcke. The meeting project at ICSI. In *Proc. Human Lang. Tech. Conf.*, pages 246–252, 2001. (Cited on page 4.)

[52] M. Naphade and J. Smith. A generalized multiple instance learning algorithm for large scale modeling of multiple semantics. In *Proc. IEEE Int. Conf. on Acous., Speech, and Sig. Proc. (ICASSP)*, Pittsburgh, PA, USA, Mar. 2005. (Cited on page 18.)

[53] A. Ng, M. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in NIPS*. MIT Press, Cambridge MA, 2001. (Cited on pages 8, 20, and 29.)

[54] N. Oliver and E. Horvitz. Selective perception policies for limiting computation in multimodal systems: A comparative analysis. In *Proc. Int. Conf. on Multimodal Interfaces (ICMI'03)*, Vancouver, CA, Nov 2003. (Cited on page 3.)

[55] S. Renals and D. P. W. Ellis. Audio information access from meeting rooms. In *Proc. IEEE Int. Conf. on Acous., Speech, and Sig. Proc. (ICASSP)*, Hong Kong, 2003. (Cited on pages 4 and 44.)

[56] D. Reynolds. An overview of automatic speaker recognition technology. In *Proc. IEEE Int. Conf. on Acous., Speech, and Sig. Proc. (ICASSP)*, Orlando, FL, 2002. (Cited on pages 4 and 75.)

[57] C. Ris and S. Dupont. Assessing local noise level estimation methods: Application to noise robust ASR. *Speech Communication*, 34(1–2):141–158, 2001. (Cited on page 53.)

[58] J. Saunders. Real-time discrimination of broadcast speech/music. In *Proc. IEEE Int. Conf. on Acous., Speech, and Sig. Proc. (ICASSP)*, pages 993–996, May 1996. (Cited on page 14.)

[59] E. Scheirer and M. Slaney. Construction and evaluation of a robust multifeature speech/music discriminator. In *Proc. IEEE Int. Conf. on Acous., Speech, and Sig. Proc. (ICASSP)*, 1997. (Cited on pages 11, 14, 62, and 63.)

[60] John Shawe-Taylor and Nello Cristianini. *Support Vector Machines and other kernel-based learning methods*. Cambridge University Press, 2000. (Cited on page 74.)

[61] M. A. Siegler, U. Jain, B. Raj, and R. M. Stern. Automatic segmentation, classification and clustering of broadcast news audio. In *Proc. DARPA Broadcast News Workshop*, 1997. (Cited on page 11.)

[62] Lisa Stifelman, Barry Arons, and Chris Schmandt. The audio notebook: Paper and pen interaction with structured speech. In *Proc. ACM SIGCHI Conf. on Human Factors in Comp. Sys.*, pages 182–189, Seattle, WA, 2001. (Cited on page 42.)

[63] D. P. W. Ellis T. Pfau and A. Stolcke. Multispeaker speech activity detection for the icsi meeting recorder. In *Proc. IEEE Workshop on Auto. Speech Recog. and Understanding*. Italy, December 2001. (Cited on page 12.)

[64] I. Ulusoy and C. M. Bishop. Generative versus discriminative models for object recognition. In *Proc. IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, San Diego, USA, 2005. (Cited on page 18.)

[65] G. Williams and D. P. W. Ellis. Speech/music discrimination based on posterior probability features. In *Proc. Eurospeech*, Budapest, September 1999. (Cited on pages 13 and 14.)

[66] E. Wold, T. Blum, and J. Wheaton. Content-based classification, search and retrieval of audio. In *IEEE Multimedia*, pages vol.3, no.3, pp. 27–36, 1996. (Cited on pages 15 and 16.)

[67] M. Wu, D.L. Wang, and G. J. Brown. A multipitch tracking algorithm for noisy speech. *IEEE Transactions on Speech and Audio Processing*, 11:229–241, 2003. (Cited on pages 47, 49, 53, and 54.)

[68] T. Zhang and C.-C. J. Kuo. Audio content analysis for online audiovisual data segmentation and classification. *IEEE Tr. Speech and Audio Proc.*, 9(4):441–457, 2001. (Cited on pages 12 and 14.)