SYSTEM ANALYSIS AND PERFORMANCE TUNING FOR BROADCAST AUDIO FINGERPRINTING

Eloi Batlle, Jaume Masip, Pedro Cano

MTG-IUA, Universitat Pompeu Fabra
Pg. Circumvalació, 8
E-08003 Barcelona. Catalunya-Spain
{eloi,jmasip,pcano}@iua.upf.es

ABSTRACT

An audio fingerprint is a content-based compact signature that summarizes an audio recording. Audio Fingerprinting technologies have recently attracted attention since they allow the monitoring of audio independently of its format and without the need of meta-data or watermark embedding. These technologies need to face channel robustness as well as system accuracy and scalability to succeed on real audio broadcasting environments. This paper presents a complete audio fingerprinting system for audio broadcasting monitoring that satisfies the above system requirements. The system performance is enhanced with four proposals that required detailed analysis of the system blocks as well as extense system tuning experiments.

1. INTRODUCTION

Audio fingerprinting or Content-based audio identification (CBID) systems extract a perceptual digest of a piece of audio content, i.e. the fingerprint and store it in a database. When presented with unlabeled audio, its fingerprint is calculated and matched against those stored in the database. Using fingerprints and matching algorithms, distorted versions of a recording can be identified as the same audio content.

Plenty of audio fingerprinting algorithms have been proposed [1]. They differ mainly in the types of features being considered, the modeling of the fingerprint, the type of distance among fingerprints and the indexing mechanims for the database look-up. There is a trade-off between the fingerprint size from on side, which relates to system complexity and scalability, and granularity and robustness on the other side. The specific requirements of an application normally determine a choice in the trade-off.

The system presented here goes beyond the template matching paradigm to a statistical pattern matching paradigm. The goals are to incorporate robustness to the system by statistically modelling the audio evolution while reducing the size of the fingerprint by considering local and global perceptual redundancies in a corpus of music data.

This paper is structured as follows. Section 2 describes the content of each system block and their most significant mathematical behaviour. Section 3 analizes both the system requirements and the available system variables. Finally, Section 4 presents the experimental results achieved during system tuning.

2. SYSTEM OVERVIEW

2.1. Main System Blocks

The complete identification system and testbench is shown in Figure 1. The identification system contained within the grey frame has three inputs and one output: the distorted audio stream, the Hidden Markov Models, the fingerprint database and the identification labels respectively. The testbench is in charge of labeling the original audio stream, introducing audio distortion at the system input and comparing both the original and identified labels to measure the system accuracy.

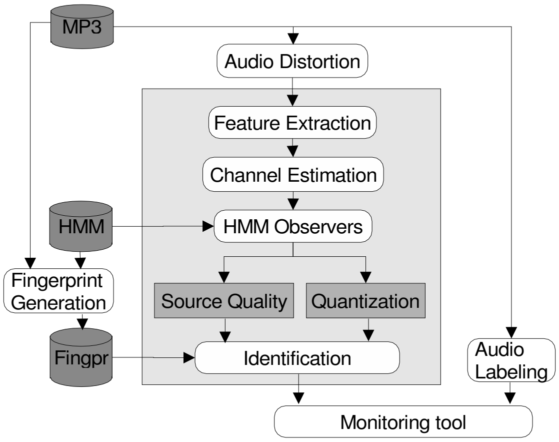


Figure 1: Audio identification system and testbench.

2.1.1. Feature extraction

The first step in an audio fingerprinting system is the extraction of feature parameters from raw audio samples. We use a filterbank based analysis procedure to approximate the human inner ear behaviour in the parametrization stage. Moreover, the well known Mel-cepstrum coefficients (MFCC) in speech recognition [2] can be used for music analysis [3].

2.1.2. Channel estimation

Techniques for dealing with known distortion are straightforward but real broadcast distortion is usually unknown. We can assume the channel as a combination of all possible distortions like equalizations, noise sources as well as DJ manipulations. To remove some effects of these distortions, we can assume that they are caused by a linear time-invariant (or slowly variant) channel which can be approximated by a linear filter $\mathcal{C}(\omega)$ that slowly changes in time. In the Fourier logarithmic space we can write

$$\ln|\mathcal{Y}(\omega)| = \ln|\mathcal{C}(\omega)| + \ln|\mathcal{X}(\omega)| \tag{1}$$

Where $\mathcal{X}$ and $\mathcal{Y}$ are the original and received signals respectively. If the distorting channel $\mathcal{H}(\omega)$ is slowly varying we can design a filter that, applied to the time sequence of parameters, is able to minimize the effects of the channel. The filter we designed for our system is

$$H(z) = 0.99\frac{1 - z^{-1}}{1 - 0.98z^{-1}} \tag{2}$$

This filter greatly improves the identification performance because it removes most equalization and transmission distortions [4].

2.1.3. Hidden Markov Model Observers

Monophonic audio can be seen as a sequences of acoustic events (notes). However, polyphonic music is much more complex since several events occur simultaneously. In [5], we defined a set of abstract events that do not have any physical meaning but allow to mathematically describe complex music as sequences of events. These events are captured by Hidden Markov Models[6] that model abstract audio generators.

Since it is almost impossible to assign a defined sound source to each HMM, a fixed number of HMMs is calculated such that each HMM maximizes the probability that if it really was a sound generator, it would generate that sound. The training stage uses the Expectation-Maximization algorithm [7] to include the unknown HMM song sequences as incomplete data, which leads to a modified Baum-Welch algorithm [8].

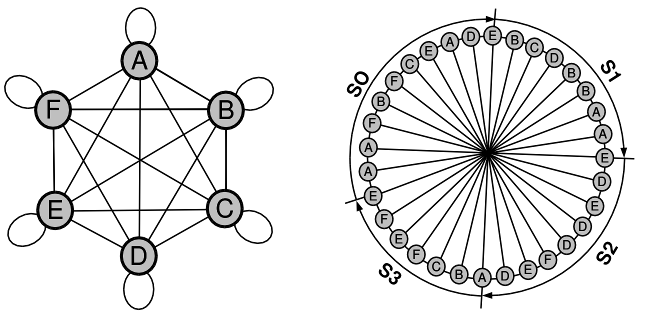
2.1.4. Identification Algorithm

Each HMM fingerprint in the song database uniquely identifies each song among the others. The fingerprint database is generated using the Viterbi algorithm [9]. This algorithm computes the highest probability path between HMMs on a complete HMM graph model (see Figure 2.a), resulting in the song observation probability given by

$$\begin{gathered} P(O|\lambda) = \sum_{allQ} p(O|Q,\lambda)P(Q|\lambda) = \\ \sum_{q_1,\cdots,q_T} \pi_{q_1} b_{q_1}(O_1) a_{q_1 q_2} b_{q_2}(O_2) ... a_{q_{T-1} q_T} b_{q_T}(O_T) \end{gathered} \tag{3}$$

where O_i is the audio sequence, λ are all the HMM, Q is the HMM sequence, $a_{i,j}$ is the jump probability from HMM i to j and $b_{q_i}(O_t)$ is the observation probability. This algorithm has been modified to allow control over the sequence length. In fact, smaller sequences are obtained if paths between neighbours are discouraged introducing probability penalties P_{ins}

$$\begin{cases} a'_{ii} = a_{ii} - P_{ins} \\ \quad a'_{ij} = 1 - a'_{ii} \end{cases} \tag{4}$$


A
B
C
D
E
F
SO
S1
S2
S3
E B C D B B A A E D E D D F E D A B C F E F E A A F B F C E A D

(a) Signature generation

(b) Database model

Figure 2: Complete HMM graph model and four song database model (S1,S2,S3,S4).

The identification algorithm matches an input streaming audio against all the fingerprints to determine whenever a song section has been detected. The Viterbi algorithm is used again with the purpose of exploiting the observation capabilities of the HMM models contained in the fingerprint sequences. Nevertheless, this time the model is not a complete graph but the HMM ring shown in Figure 2.b. In this structure, each HMM only has two links, one to itself and one toward its immediate neighbour. The identification algorithm scales linearly with the number of songs in the database because no backtracking is required for sigle path models.

2.2. New blocks

The main system blocks were first described in detail in [5]. Since then, two new blocks have been added previous to the identification block. This paper focuses on these blocks and their interaction with the identification block.

2.2.1. Quantization

The first system demostration used 8 byte floating point registers on all its building blocks. In particular, the identification algorithm, which also used floating point units, may outperform in many cases the original identification requirements. Bearing in mind that identification is the most computing intensive task of the system, and that it grows linearly with the song database (O(N)), one quantization block has been added to stress the resolution capabilities of the algorithm This block aims at reducing the space resources consumed by the algorithm (1 float = 2 int = 4 short = 8 bytes). As a result, potential speedup factors ranging 2-8 could be obtained with SIMD processors where the time performance may be inversely proportional to the number of data units contained within a large multimedia register [10].

2.2.2. Source information quality

Some song sections have lower discriminating capabilities than the others, such as common chord sequences in pop music. This block aims at mesuring the audio information quality and determine when the identification algorithm is fed with enough information to keep the uncertainty of the song identity below the system requirements. This block extracts the sequence of HMM contained in the source stream. Since all HMM in the sequence can be considered independent and equally probable in a first approach, the number of HMM in the sequence could be a reasonable aproximation of the identification quality (see Section 3.2.2).

3. SYSTEM ANALYSIS

The analysis of the system aims at providing the methodology for the proper design and optimization of the identification algorithm. This section is structured in two parts. First, we analize the system requirements and then the system variables to obtain a complete picture of their impact on the overall system performance.

3.1. Main system requirements

3.1.1. Broadcast robustness

The channel estimation block in charge of minimizing the effects of audio edition and transmission channels. This feature guarantees the applicability of the system within the framework of audio broacasting. Since this block has small impact on the system scalability it is granted far enough resources to accomplish its commitment. Therefore, it can be assumed that this block will perform properly without further optimization.

3.1.2. High accuracy

Given a fixed amount of resources for the identification block, the objective is to reach the highest possible accuracy. The quantization and the source information quality blocks contribute in this sense. On one side, the quantization noise should be as low as possible to keep good identification accuracies. On the other side, the source information quality analysis tags error prone fragments in order to warn the identification algorithm where the Viterbi algorithm should wait for more audio input to avoid ill-biased results.

3.1.3. Large scalablility

The feasibility of the system for real-time operation was demonstrated in [5]. With the advent of SIMD registers, computer performance boosted up to multimedia application requirements. Nevertheless, algorithm conversion for efficient parallel vector processing is not always immediate. Data structure redesign, coding over the SIMD instruction set as well as strong cache constraints are some of the challenges to be afforded. As a result, the potential 2-8 speedup factors achieved with these techniques have direct impact on the database scalability.

3.2. System variables

Four system variables allow managing the behaviour of the identification block. The analysis of each variable together with its effect over the system requirements are described next.

3.2.1. Descriptor and fingerprint depths

Two descriptor sets have been trained to capture generic instrument sources. Each set is composed of 256 and 1024 descriptors respectivelly. While bigger sets do not affect scalability they should improve system accuracy since the fingerprint information increases by a $log2(1024)/log2(256) = 10/8$ factor.

Song fingerprints were retrieved projecting the song feature vectors over complete 256 and 1024 HMM node graphs. Free fingerprints contained an average of 1038 descriptors/song while fingerprints compressed with $P_{ins} = -10$ (log scale) contained 448. Signature compression improves system scalability by factor 2.33 but conflicts with the accuracy requirements.

3.2.2. Quantization and source information quality

The quantization block projects HMM scores over an 8 bit code set. This code length is a trade off between quantization noise and data ranges. On one side, large code lengths reduce accuracy penalties introduced by quantization noise. On the other, minimal code length avoids unsigned short integer overflows (maximum value = 2^{16} = 65536) during Viterbi computations. Thanks to SIMD speedup, quantization can improve scalability by factor 4 at some accuracy expenses.

Regarding the audio source quality, it can be infered by a direct analysis of the fingerprint database. In fact, this is closely related to the discrimination power of a fingerprint and its sub-signatures. Given a song database of N songs and a finite alphabet ρ of size $|\Sigma| = \rho$, it is necessary to have at least a sequence of length m such that:

$$\rho^m \geq N$$

The presented system must recognize audio titles from an excerpt of audio–property known as *granularity*. The corresponding fingerprint extracted from a portion of audio is a subset the whole music title fingerprint. The size of the “subfingerprint” is not constant in our approach. It dependes on the richness of the audio when projected to the fingerprint. Figure 3 displays the time length pdf (200ms frames) of sequences containing 1 to 12 HMM (left-to-right).

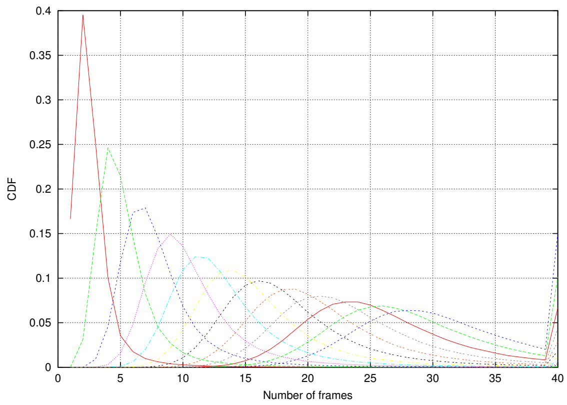


Figure 3: Modeling duration of descriptor subsequences.

The source quality block only requires a small and constant amount of resources which are independent of all previous variables, therefore it has no impact on scalability but it improves the system accuracy.

4. SYSTEM TUNING

The input for the system tuning experiments was a continuous audio stream generated appending 3902 songs contained in the mp3 audio database. The system was implemented using the C++ programming language (SIMD used MMX assembler code). Each experiment took less than 8 hours to complete on a computer cluster with 32 Pentium-III CPUs at 1GHz running C++ objects interconnected through MPI [11].

The distortion block present during the experiments introduced strong non-linear compression used in radio broadcasting. Compression is a dynamic range reduction that increases the overall loudness without changing the maximum signal peak level. This block combines a compressor and a limiter in order to achieve a fixed maximum level. The threshold, ratio, attack and release values used are 0.5, 40, 10ms and 2500ms respectively.

One experiment has been selected as reference to study the different system variables. This reference experiment used 256 HMM, 1038 length fingerprints, float registers and a 30 frame Viterbi window (6 seconds). Results are expressed as identification error rates over the song percentiles.

4.1. Descriptor and fingerprint depths

The reference experiment has been confronted against two system variables, the HMM and fingerprint depths. Figure 4 shows the accuracy results that confirm the conclusions reached during the system analysis. Although fingerprint compression penalyzed accuracy, the results show that bigger HMM sets can improve the system accuracy and compensate from fingerprint compression.

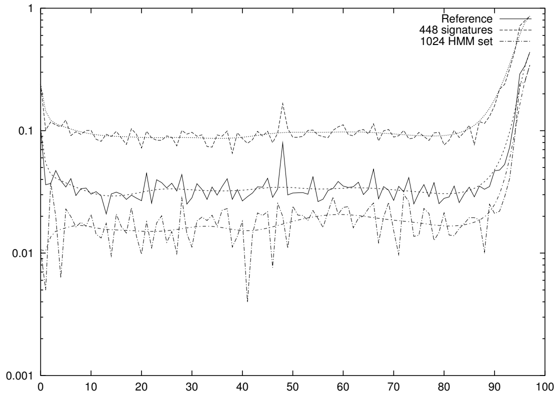


Figure 4: Accuracy impact of descriptor and fingerprint depths.

4.2. Quantization and source information quality

Again two system variables, namely 3 bit quantization and 10 HMM source quality measures, have been tested against the reference experiment. As already stated in the system analysis, the SIMD block introduced accuracy penalty while the source quality block improved its performance (see Figure 5).

5. CONCLUSIONS

HMMs that model generic audio sources have proved their utility as fingerprint descriptors and observers. Descriptor and fingerprint depths as well as quantization and source information quality measures have been proposed to improve the overall system performance. The preliminary results of the system analysis match the accuracy and scalability measures obtained with the system testbench. Fingerprint compression and SIMD improve the system scalability introducing accuracy penalties. Nevertheless, accuracy can be improved with bigger HMM sets and source quality measures, and both lack scalability impact. Therefore, we can conclude that the four proposals together with the system analisis provided the required tools and methodology to achieve the desired system tuning goals.

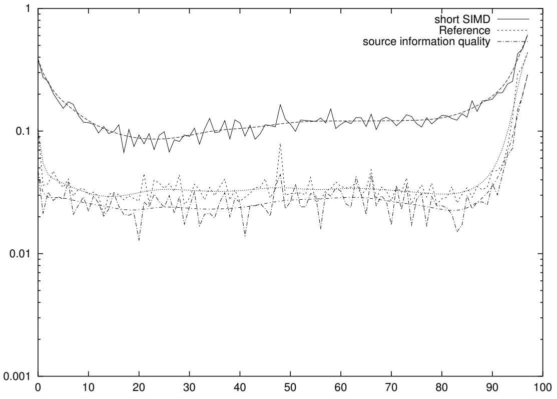


Figure 5: Acuracy impact of quantization and source quality.

6. REFERENCES

[1] P. Cano, E. Batlle, T. Kalker, and J. Haitsma, "Review of Audio Fingerprinting Algorithms," in *MMSP*, 2002.

[2] E. Batlle, C. Nadeu, and J. A. R. Fonollosa, "Feature Decorrelation Methods in Speech Recognition.," in *Int Conf on Spoken Language Processing*, 1998.

[3] B. Logan, "Mel Frequency Cepstral Coefficients for Music Modeling," in *ISMIR*, 2000.

[4] R. A. Bates, "Reducing the Effects of Linear Channel Distortion on Continuous Speech Recognition," M.S. thesis, Col. of Engineering. Boston University, 1996.

[5] E. Batlle, J. Masip, and E.Guaus, "Automatic Song Identification in Broadcast Audio," in *IASTED Signal and Image Processing Conference*, 2002.

[6] L. R. Rabiner, "A Tutorial on HMM and Selected Applications in Speech Recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[7] A. P. Dempster and et altri, "Maximum Likelihood from Incomplete Data via the EM Algorithm," *Journal of the Royal Statistical Society*, vol. 39, no. 1, pp. 1–38, 1977.

[8] L. E. Baum and J. A. Eagon, "An Inequality with Applications to Statistical Estimation for Probabilistic Functions of Markov Processes," *BAMS*, 1967.

[9] A. J. Viterbi, "Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Identification," *IEEE Trans. Info. Theory*, 1967.

[10] Kip R. Irvine, "Assembly Language for Intel-Based Computers," *Prentice Hall*, 2002.

[11] Gropp William, "MPI: The Complete Reference," *MIT Press*, 1998.