Friedrich-Alexander-Universität Erlangen-Nürnberg

FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG

Master Thesis

Audio Fingerprinting Techniques for Sample Identification in Electronic Music

submitted by

Pedro Solórzano

submitted

October 31, 2016

Supervisor / Advisor

M.Sc. Patricio López-Serrano
Prof. Dr. Meinard Müller

Reviewers

Prof. Dr. Meinard Müller

AUDIO LABS

International Audio Laboratories Erlangen
A joint institution of the
Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU) and
the Fraunhofer-Institut für Integrierte Schaltungen IIS.

Fraunhofer IIS

Erklärung

Hiermit versichere ich an Eides statt, dass ich die vorliegende Arbeit selbstständig und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Die aus anderen Quellen oder indirekt übernommenen Daten und Konzepte sind unter Angabe der Quelle gekennzeichnet. Die Arbeit wurde bisher weder im In- noch im Ausland in gleicher oder ähnlicher Form in einem Verfahren zur Erlangung eines akademischen Grades vorgelegt.

Erlangen, October 31, 2016

Pedro Solórzano

Acknowledgements

I would like to express my gratitude to Prof. Dr. Meinard Müller who introduce me to music processing analysis. He made me to consider to do my thesis in music signal analysis. I learned through his support, patience, and directions during master thesis.

I also would like to express my gratitude to my supervisor M.Sc. Patricio López-Serrano who gave me a lot of support during my master thesis. I appreciate his patience and directions through the thesis. Also, I spent good times with him and the team during my stay in Audio labs.

I would like to thank to all friends that I've obtained during this 2 years of masters studies. I spend good time them and I got from them personal and professional learning. I got many support from them during the thesis.

At last but not less important, I would like to express my immense gratitude to my family, my boyfriend Carlos and friends who are living far from me. They gave me support and motivation to grow professionally and personally. I miss them and I'm willing to meet them soon :).

Abstract

Many tasks in Music Information Retrieval use audio samples (short fragments taken from larger musical pieces) to organize, extract or search for music information. In order to find these samples within a collection that contains them, a technique called audio fingerprinting is often used. The main goal of this thesis is to obtain a better understanding of the various components and parameters of a fingerprinting-based sample identification system. In this context, electronic music constitutes an interesting test scenario, but the application of the techniques studied is not limited to this genre. The main assumption made in this thesis is that musically meaningful sections in audio recordings can often be characterized by the presence or absence of certain sound events or patterns (such as samples), which, however, may be superimposed with other sound sources and/or appear in modified forms. We use a collection of electronic music samples within the genres: dance, deep house, dubstep, hip-hop, techno, and trap. We apply sample combinations, audio degradation, and time shift differences on these samples and match the resulting fingerprints to their original versions. Furthermore, we applied audio matching with shifted queries. Evaluation is made with a variant of precision, recall, and F-measure. The obtained results guide us to identify particular behaviors of the information captured by the fingerprinting process which can lead to interesting research approaches in music structure analysis.

Contents

Chapter 1

Introduction

1.1 General Background

In many cultures, music is an important part of people's lives and it spans a wide range of forms and styles such as folk songs, electronic music, symphonies, etc. With the technological advances of music production, researchers are becoming more interested in developing computational methods not only for storage, distribution, and production of music but also for the field of *music information retrieval* (MIR) which aims to organize, extract, and search musical information, e.g., browsing personal collections, automatically categorizing music, copyright monitoring [14].

One important task in MIR is *music structure analysis*. As mentioned in [14], one main goal of music structure analysis is to divide a given music representation into temporal segments that have some musical meaning and to group these segments into appropriate categories. Music structure analysis tasks involve complex and generally ill-defined problems since the concept of structure is ambiguous. The musical structure of one piece of music may be explained by repeating melodies, musical sections while in other pieces may be characterized by a certain instrumentation or tempo [14].

Musical information is usually described through a small number of examples, e.g., given in the form of audio segments. These examples may specify a rhythmic pattern, a harmonic progression, a certain timbre, or some other type of audio event. In MIR, audio recordings are often analyzed or structured by means of such examples using *content-based* retrieval techniques related to audio identification, audio matching [10], and version identification [5]. In the audio identification process, it is often used audio fingerprinting techniques since they can give a compact and descriptive feature representation [7] [20]. A common application of this technique is searching an audio recording within a large database using a small sample (query). Typical applications of audio fingerprinting are: automatic playlist recognition, automatic music library organization,

and digital rights management (DRM) monitoring for file sharing [7]. One important issue in the retrieval process is the question of how to deal with acoustic and musical variations. In particular, the identification of specific sound events becomes challenging when they are superimposed with other sound sources.

An interesting scenario for applying content-based techniques is the family of genres of electronic music (EM). The devices and/or software commonly used to produce EM, such as sequencers, digital audio workstations (*Ableton Live*, *MAGIX Music Maker Premium*, *Reason*) [1] [12] [16], impose a musical structure in which musical patterns are repeatedly triggered and overlaid. According to [11], these patterns or audio samples, whose length can span several seconds, may be superimposed with other sound sources and/or appear in modified forms. This particular musical structure has increasingly received attention in research since it allows new approaches on important tasks in MIR. For example, there are studies on modeling and decomposing EM (music structure analysis) [11], sample identification in hip-hop music [19], segmentation and timbre similarity [17], and downbeat detection [8].

As mentioned above, EM is often based on sampling, where existing recorded sounds or audio samples are reused and they may appear in different forms. Furthermore, many iterative information retrieval tasks such as automatic music identification extract and analyze musical information with the help of small samples. Motivated by this scenario, we want to investigate a fingerprint-based retrieval approach to sound event detection in complex mixtures. More specifically, we want to use EM audio samples in order to analyze the components and parameters of a fingerprint-based sample identification system under the scenarios of complex mixtures, audio degradation, and time shift differences. The fingerprinting process used in this thesis involves choosing relevant frequency components (with maximum intensity values) along an EM sample (usually with length of 4 to 15 seconds). In this context, electronic music also constitutes an interesting test scenario, but the application of the techniques studied here are not limited to this genre. A general outline of the scenario in which this thesis is developed is shown in Figure 1.1.

1.2 Main Contributions

The main contributions of this thesis are as follows.

First, we use an fingerprint retrieval matrix in order to compare audio samples with each other and investigate their performance under specific combinations.

Then, we study the behavior of the features of these audio samples under 3 general scenarios: complex combination, time shift differences, and audio degradation such as white noise, adding external sounds, and adding effects. In addition, we applied fingerprint-based sample identification

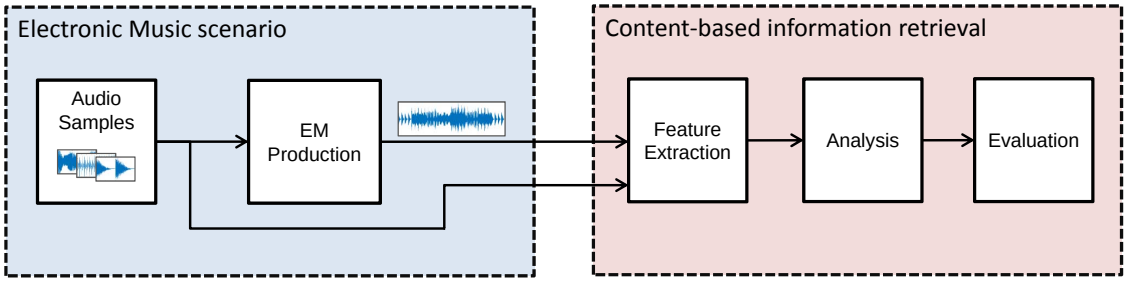

Figure 1.1: Outline scenario in which this thesis is developed.

approach with time-shifted queries.

Finally, we identify important components and parameters of the fingerprinting process which significantly affect the behavior of features.

1.3 Thesis Organization

This thesis is organized in 4 chapters.

In Chapter 2, we introduce the electronic music scenario based on audio samples. Also, we give a brief description of basic techniques and concepts that are going to be used through the following chapters. We review concepts of music processing analysis such as audio feature representations, content-based retrieval techniques and evaluation.

In Chapter 3, we describe our processing pipeline. We use a small data-set of audio samples in order to explain the fingerprint-based feature extraction system, the methods, and experiments applied.

In Chapter 4, we use a data-set of 111 EM audio samples in our processing pipeline and discuss the obtained results.

In Chapter 5, we make conclusions about our main results and discuss about recommendations and future work.

Chapter 2

Background and Fundamentals

In this chapter, we elaborate on theoretical background and fundamental concepts that are being used in this thesis. The main idea is to give a general description and mathematical notations rather than technical details. First, we describe the electronic music scenario, then we review important concepts related to music processing such as audio representation, short time Fourier transform, spectrogram representations, fingerprinting, audio matching, and evaluation.

2.1 Electronic Music Scenario

Electronic Music (EM) is a general term that covers genres like techno, trance, dance, house music, hip-hop, etc. which are often produced by a combination of several musical patterns. In many production examples, the hook or the main musical sequence of an EM track results in 20 second or so [18]. Several musical patterns are added or removed along the musical composition in order to shape a musical tension and to keep interest on the listener throughout the track.

A common characteristic among different genres of EM is the loop-based music production technique. A *loop* is a piece of audio that represents a sound event, e.g., an instrument sound, piece of music, an audio sample. It is often found with a duration of one or few bars [1] and in some cases it can suggest a predetermined music structure [18], [3]. Digital audio workstations (DAWs), multi-track layout sequencers, and music production softwares allow to compose loop-based EM, e.g., *Ableton* [1], *Magix Music Maker* [12], and *Reason* [16]. Loop-based EM tracks are produced by the combination of different channels that activate or deactivate these audio samples with a musical relationship. Besides, such samples may appear in a modified form by an addition of audio effects, e.g., delay, reverberation, volume changes, etc.

[1] In music theory, the term *bar* refers to the period of time corresponding to a specific number of beats or onset notes. It is used to mark the metrical units of a piece of music [4].

As said at the beginning of this section, the core or the hook last a few seconds when it is compared to the total duration of the track [18]. Thus, EM producers shape the musical tension structure in order to maintain the listener's attention. Butler in [3] and Snoman in [18] gave a general structure of an EM track that leads the listener to the core of the composition and then to an end. This structure is not a rule to create an EM, but it can be identified or perceived in the majority of EM instances.

A musical energy structure of an EM track can be divided in 4 parts: intro, build-up, break-down, core and the outro. The *intro* is composed with one or two loops and usually gives the main beat tempo of the piece of music. Then, a *build-up* part comes increasing the energy or musical tension with a combination of more loops. The *break-down* follows to create an expectation to the listener of what is coming, usually musical sequences with a lower musical energy are played. After the breakdown, the *core* appears where the main hook of the track is present and a climax is reached (usually most of the loops are activated). As a final state, the *outro* decrease the musical energy after the climax and defines the coming end of the track. Figure 2.1 shows a block structure using 3 loops with same length (8 seconds) and a synchronized activation time. The general structure of this figure shows that the intro last 16 second, the build-up follows with a duration of 32 seconds, the brake down continues with a duretion of 16 seconds, the core last 16 seconds, and the outro follows with a duration of 448 seconds. In Figure 2.2, we show the block structure of an EM track which I produced using the software and data set of Magix [12].

2.2 Audio Signal Representation

Music can be represented as an audio signal that encodes certain aspect of a piece of music such as temporal, dynamic and specific tonal micro-deviations. The encoded information is modeled as a *continuous-time* signal (CT-signal) or *analog* signal that reflect infinitesimally small changes in both the amplitude and the time. However, in the digital signal processing field, *analog* signals must be converted into *digital* signals through a *digitization* process. Following the definitions in [14], in most cases, an analog-to-digital conversion consists of two steps called *sampling*[2] and *quantization*. Given a CT-signal defined to be a function $f : \mathbb{R} \to \mathbb{R}$, a *discrete-time* signal (DT-Signal) is defined to be a function $x : \mathbb{Z} \to \mathbb{R}$ and is obtained by

$$x(n) := f(n \cdot T) \tag{2.1}$$

where, the value $x(n)$ is called *sample*. The positive and real value $T > 0$ is referred as the *sampling period* and its inverse as the *sampling frequency* $F_s := 1/T$.

[2]In the context of audio digitalization, the term sampling refers to the process of converting continuous-time signals into discrete-time signals.

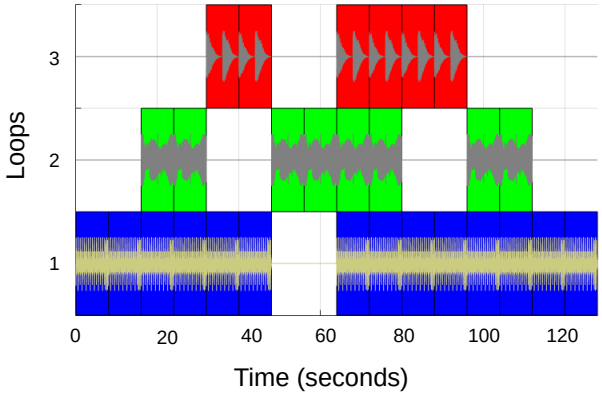

Figure 2.1: Block Structure of an EM track produced with 3 loops. These audio samples have a duration of 8 seconds and represent a particular sound event. Blue, green and red blocks represent the activation of a drum, a melody and a chord composition respectively.

Figure 2.2: Example of an EM track produced with 9 loops. Each row represents an audio sample, where colored blocks indicate their presence in the track. Each audio sample has different bar duration and represents a particular sound event. This example is a piece of music that I mixed using the software and loop data-set of Magix [12].

Quantization allows to restrict the amplitude of the signal to a limited set of values. However, this step is omitted in the above definition of DT-Signals for the sake of simplicity. In addition, the definition does not take into account a finite representation of samples that is required for digital signal processing. For music processing, the signal $x(n)$ consist of a set of samples $x(1), x(2), ..., x(N)$ where $N \in \mathbb{N}$ is the audio's length. Also, along this thesis, the terms *audio signal*, *audio recording*, *track signal* and *loop signal* refer to DT-Signal. Figure 2.3 shows an audio representation of an audio signal; the sampling rate is $F_s = 22050\,Hz$ and the duration of the signal is 8 seconds.

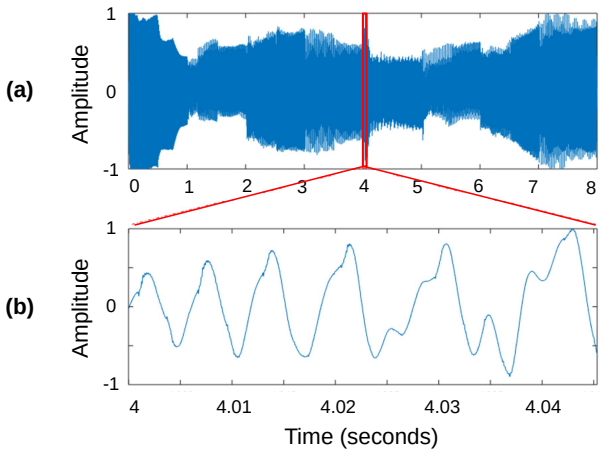

Figure 2.3: Audio representation of an audio signal of an EM with a sampling rate of 22050 Hz and duration of 8sec ($N = 176400$ samples). Vertical axes represent the amplitude o the signal. Horizontal axes correspond to the time (in seconds). Figure (a) represents the waveform of the complete signal whilst figure (b) shows a zoomed version of 1000 samples (45.4 ms of the signal).

2.3 Fourier Transform (FT)

A music signal is the superposition of sound components over time. The audio representation in time can show some explicit information and it can have other hidden elements. The Fourier transform is the most important tool in audio signal processing. It maps a time-dependent signal into a frequency dependent function allowing to obtain additional information for further audio processing. The content of this thesis deals with *Discrete Fourier Transform* (DFT) since we compute audio signals using the fast Fourier transform algorithm (FFT). Thus, we omitted other definitions of Fourier transform.

Given a DT-Signal x of length N, the discrete Fourier transform X of x is defined by

$$X(k) = \sum_{n=0}^{N-1} x(n) \cdot e^{-2\pi ikn/N} \tag{2.2}$$

for $k \in [0 : N-1]$. The index k of $X(k)$ corresponds to the physical frequency

$$F_{\text{coef}}(k) = \frac{k \cdot F_s}{N} \, [Hz] \tag{2.3}$$

The *inverse discrete Fourier transform* (IDFT) is defined as

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) \cdot e^{2\pi ikn/N} \tag{2.4}$$

for $n \in [0 : N-1]$.

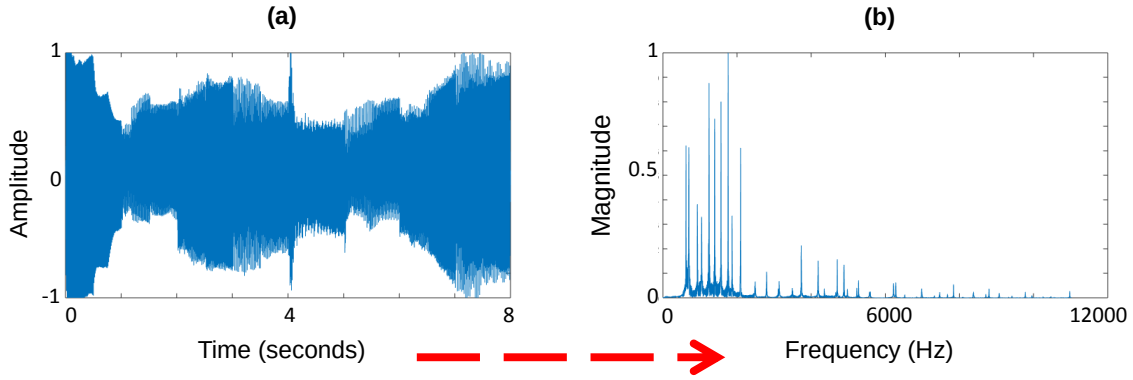

Figure 2.4: Discrete Fourier transform (DFT) (b) of the audio signal in (a).

For a visual understanding, Figure 2.4 shows the magnitude of the DFT (b) of the waveform in (a). The Fourier transform gives the frequency components that are present in the audio signal, however, the time information is hidden.

2.3.1 Short Time Fourier Transform (STFT) and Spectrogram Representations

For audio analysis, the *short time Fourier transform* (STFT) is used to know where the time-frequency components of an audio music appears. The main idea of this technique is to compute the FT of small sections or *frames* of the entire signal instead of computing the FT of the complete signal. The STFT is a compromise between a time- and frequency-based representation and allows to obtain the frequencies occurred on each computed frame. The mathematical definition of the discrete STFT $\mathcal{X}(m,k)$ is

$$\mathcal{X}(m,k) := \sum_{n=0}^{N-1} w(n) \cdot x(n+mH) \cdot e^{-2\pi ikn/N} \tag{2.5}$$

where $k \in [0 : N-1]$ is the frequency index, $m \in \mathbb{Z}$ is the frame index, w is the window function, H the distance between each frame (usually called *hop* size), and N is the window's length. The index m corresponds to the physical time

$$T_{coef}(m) = \frac{m \cdot H}{F_s} \, [sec] \tag{2.6}$$

and the index k to the physical frequency

$$F_{\text{coef}}(k) = \frac{k \cdot F_s}{N} \, [Hz] \tag{2.7}$$

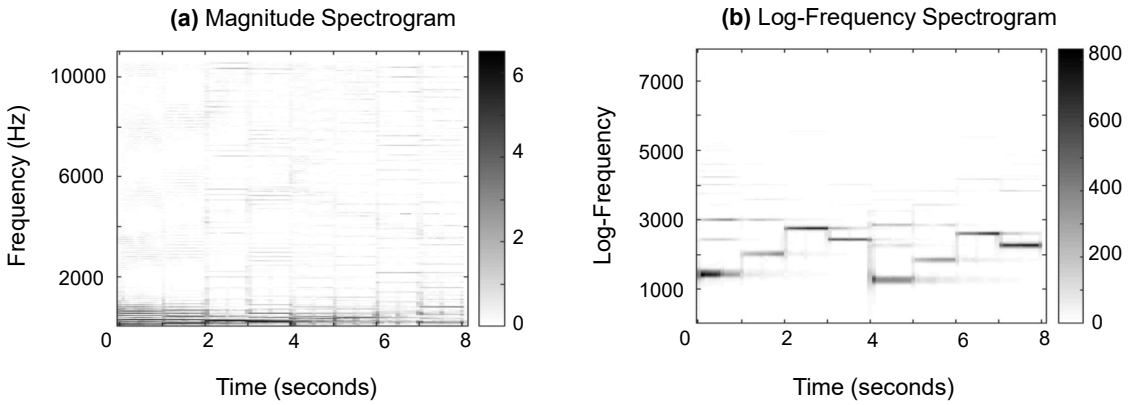

Figure 2.5: Spectrogram representations of the audio signal described in Figure 2.4. The STFT was computed using a Hann window with length of 4096 samples and a hop size of 2048 samples. (a) Spectrogram defined by Equation (2.8), (b) Log-frequency spectrogram with a resolution of 36 bins per octave.

2.3.2 Spectrogram Representation

The STFT can be visualized by a two-dimensional (time-frequency) feature representation called *spectrograms*. A STFT's magnitude spectrogram is defined by

$$\mathcal{Y}(m,k) = \mid \mathcal{X}(m,k) \mid^2 \tag{2.8}$$

Figure 2.5 (a) shows the Spectrogram of audio signal in Figure 2.4. The horizontal axis represents time frames, the vertical axis the frequency coefficients, and the color (see color bar) refers to the intensity of a particular frequency at a particular time.

From the equation 2.7 we can see that the spectral coefficients in the frequency dimension are equally separated. Motivated by the natural perception of the human, it is often computed spectrograms with the frequency axis logarithmic spaced. The *log-frequency spectrogram* allows to emphasize musical or tonal relationships of an audio signal. Figure 2.5(b) shows the log-frequency spectrogram of the audio signal in Figure 2.4. When comparing both spectrogram representations, the log-frequency gives a new resolution on the vertical axis where the relevant frequency components of the human auditory system are emphasized.

2.4 Content-Based Audio Retrieval

As said in Section 1.1, audio samples are used to describe an audio recording information. The topic of this thesis deals with an important content-based audio retrieval task known as *audio identification*. Thus, in this section, we focus on the concepts of related techniques such as

fingerprinting with peak maps, similarity measures and audio matching.

2.4.1 Fingerprinting with Peak Map

Fingerprinting has become a powerful technique since they are a compacted and descriptive feature representation that follow the important properties of *robustness*, *reliability*, *fingerprint size*, *ganularity*, *search speed and scalability* [7]. The way of designing and computing a fingerprint varies depending of the requirements imposed by the application in hand. Philips and Shazam system developed two of the most important audio fingerprint techniques; they are still in use nowadays. The *Philips* system was developed by Haitsma and Kalker in 2002 [7] and is based on energy differences of neighboring frequency bins in a time frame, called sub-fingerprint. A fingerprint block is a sequence of 256 sub-fingerprints and it can be used to identify an audio music. On the order hand, the *Shazam* system, developed by Wang in 2003 [20], was motivated by smartphone-based applications and it is based on spectral peaks constellation and hashing techniques.

The main idea of fingerprinting with peak maps is to select maximum values (peaks) in the spectrogram. An analysis window is used to define regions in the spectrogram in which a peak may be selected. An amplitude limitation for choosing a peak in the spectrogram can be defined in order to avoid noisy or irrelevant results. The overall selected peaks yield to a peak map representation. Figure 2.6 shows an example of how a peak selection is done with an analysis windows size of 4x4 (4 frequency indexes and 4 time frames). Maximum values below 1 are not chosen. In Figure 2.7, we show a peak map PM (a) and a log-frequency peak map LPM (b) using the spectrograms in Figure 2.5.

2.4.2 Similarity measure and Matching Curve

In the context of this thesis, two audio samples are similar if all information of the query is contained in the other audio. Thus, a similarity measure $s \in [0, 1]$ aims to quantify how much information of a query is contained in the other audio signal. For example, in [11], it was used such similarity definition and the similarity measures applied for the audio identification tasks were: *Cosine*, *Inclusion* and *Jaccard* measure.

The comparison between a query (audio sample) and an audio recording results in a matching curve. Peaks in the matching curve indicate activation time positions of the sample within the recording. One of the most common matching techniques is the *diagonal matching* DM. The main idea of DM is to shift the query over the audio recording and locally compare the feature representation of the musical pattern by means of the similarity measure.

Following the diagonal matching definition in [14]. Given the feature representation Q and V of

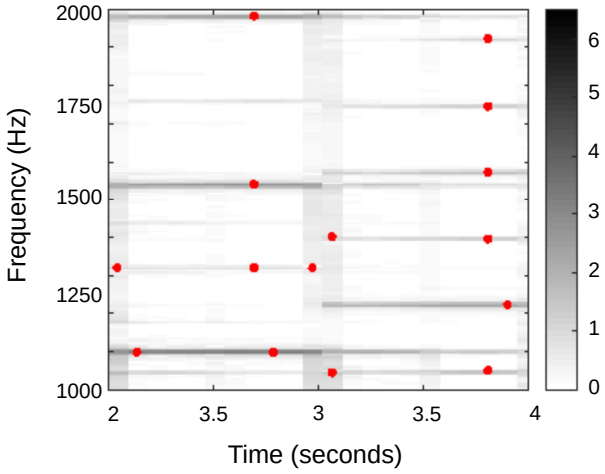

Figure 2.6: Selection of peaks from the spectrogram. The analysis windows has a size of 4 frequency indexes (16.15 Hz) and 4 time frames (0.372 ms). Red points indicate selected peaks. Maximum values in the Spectrogram which are less than 1 are not chosen.

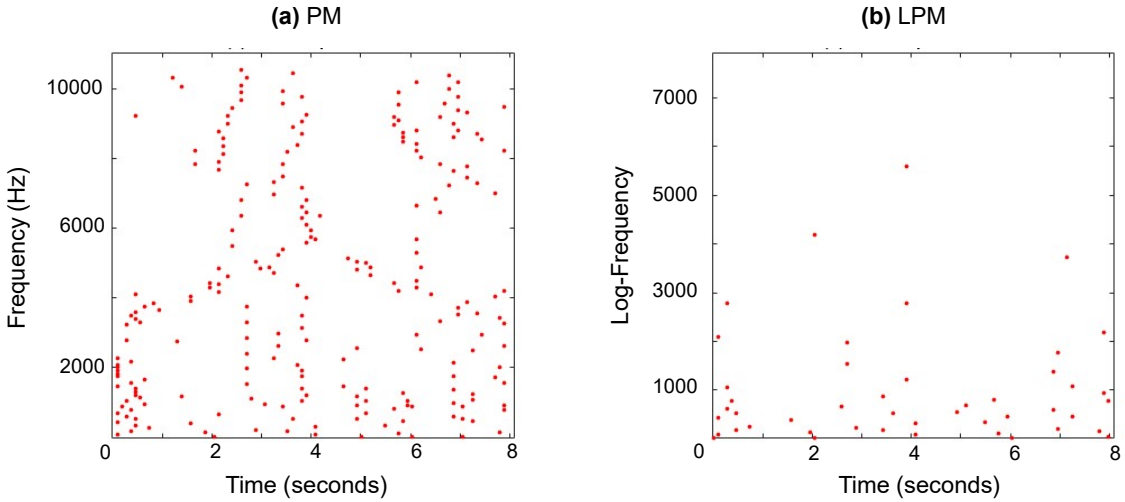

Figure 2.7: Peak Map PM (a) and log-frequency peak map LPM (b) of the magnitude- and log-frequency- spectrogram described in 2.3.2.

a loop pattern and the audio recording respectively. Let be similarity measure $s(Q, V_m) \in [0, 1]$, where V_m is the feature representation section starting at the frame m and with the same size of Q. The local comparison $\Delta_{Diag}(m)$ is defined by,

$$\Delta_{Diag}(m) = s(Q, V_m) \tag{2.9}$$

where $\Delta_{Diag}(m) \in [0, 1]$. Figure 2.8 shows an example of the comparison procedure. Red points represent the peak map of the query described in Figure 2.7(a) and black points the peak map of the track described in Figure 2.1. The PM of the query is shifted over the time frames of

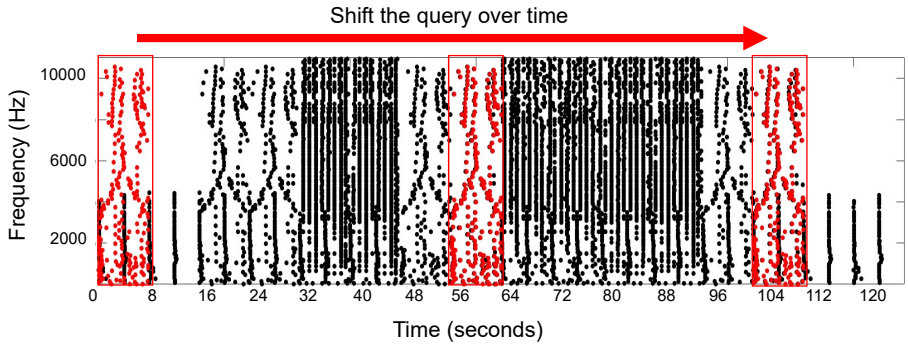

Figure 2.8: Audio Matching. Red points denotes the peaks of our query described in Figure 2.7(a). Black points indicates the peaks of the track described in Figure 2.1. The matching curve is made by shifting the PM of the query over the time frames of the PM of the audio recording. A local similarity measure is computed for every shift. Red rectangles shows 3 different comparisons within the track.

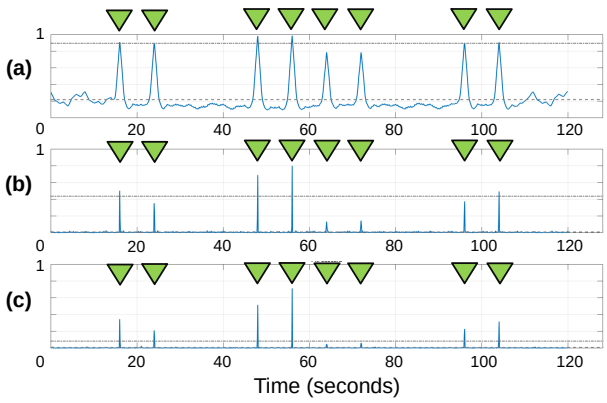

Figure 2.9: Matching curve results of Figure 2.8. The curves (a), (b), and (c) correspond to the matching curves using the *cosine*, *inclusion* and *jaccard* measure respectively.

the PM of the track. For each shift, a similarity measure is computed. Red rectangles denote 3 different time-frames m positions where a comparison is made. In Figure 2.9 is shown the resulting matching curve of Figure 2.8. The peaks of the matching curves reveal the instances where the audio sample is present. Also, these instances are indicated by green triangles on the top of each matching curve.

2.5 Evaluation

Now that we have described various matching techniques, we need to deal measures which can help us to determine how well a given procedure performs the task in hand or how reliable is the result of a process in specific tasks. In this section, we addressed the *gain* and *Pearson correlation* coefficients used in [11] and the *precision, recall, F-measure* definitions.

Similarity Measure	Gain Ratio	Pearson Correlation
Cosine	4.529	0.305
Inclusion	9.174	0.645
Jaccard	30.084	0.776

Table 2.1: Gain Ratio and Pearson Correlation of the matching curves in Figure 2.9.

2.5.1 Gain Ratio and Pearson correlation coefficient

The *gain ratio* is a measure of the relation between the average gain of the relevant values (results in loop activation times) and the average of the curve. Thus, in this thesis, the gain ratio $\text{Gain}_{\text{Ratio}}$ is define by

$$\text{Gain}_{\text{Ratio}} = \frac{\text{Average of values in activation times}}{\text{Average of the matching curve}} \tag{2.10}$$

where high $\text{Gain}_{\text{Ratio}}$ indicates that peaks in the matching curve are noticeable. If $\text{Gain}_{\text{Ratio}}$ is low, the values in the activation times are hard to find in the matching curve, which means that the similarity measure does not performs well for the comparison in hand. Table 2.1 shows the gain ratios from Figure 2.9. The Jaccard measure shows a better performance comparing with the others.

In this thesis, the *Pearson Correlation* $\text{Pcorr}_{\text{coef}} \in [0, 1]$ is computed by the Pearson correlation coefficient between the matching curve and the corresponding ideal matching curve. This ideal curve consist of ones in the activation times and zeros in the remaining time. A $\text{Pcorr}_{\text{coef}}$ close to 1 indicates that the matching curve is highly correlated to the ideal matching curve and there is a clear identification of peaks in the curve. If $\text{Pcorr}_{\text{coef}}$ is close to zero, peaks in the activation times are difficult to identify in the matching curve, which means that the similarity measure does not performs well for the task in hand. In Table 2.1 we show the Pearson correlation coefficients of Figure 2.9. The Jaccard similarity measure shows the highest performance and the cosine measure having the lowest.

2.5.2 Precision, Recall, F-Measure

Precision, recall, and F-measure are concepts from the field of information retrieval and pattern recognition. Given a classification type as in Figure 2.10, where the estimation and reference is compared, the *precision* P of the estimation is defined as the number of true positives divided by the total number of items estimated as positive:

$$\text{P} = \frac{\#\text{TP}}{\#\text{TP} + \#\text{FP}} \tag{2.11}$$

		Reference	
		Positive	Negative
Estimation	Positive	True Positive (TP)	False Positive (FP)
	Negative	False Negative (FN)	True Negative (TN)

Figure 2.10: Classification parameters when 2 events are compared [14].

The recall R is defined as the number of true positives divided by the total number of positive items in the reference:

$$\mathrm{R} = \frac{\#\mathrm{TP}}{\#\mathrm{TP} + \#\mathrm{FN}} \tag{2.12}$$

Both precision and recall are bounded to the values $[0, 1]$. Precision $\mathrm{P} = 1$ means that there is no false positive and all items estimated as positives are indeed positives. In contrast, a perfect recall $\mathrm{R} = 1$ means that there is no false negative but there may be false positives. When $\mathrm{P} = 1$ and $\mathrm{R} = 1$, it means that the estimation values are exactly classified as to the references annotation.

The F-measure F is defined as the harmonic mean between precision and recall:

$$\mathrm{F} = \frac{2 \cdot \mathrm{P} \cdot \mathrm{R}}{\mathrm{P} + \mathrm{R}} \tag{2.13}$$

where $\mathrm{F} \in [0, 1]$ with $\mathrm{F} = 1$ if and only if $\mathrm{P} = 1$ and $\mathrm{R} = 1$.

During this thesis, we applied this concepts to compare audio sample features. In this case, the reference is the query and the estimation is a track. A positive is defined as a peak in the fingerprint representation whereas a negative absence. Thus, true positives refer to common peaks between the reference and the estimation, false positives refer to peaks in the estimation that are not in the reference, and false negative are peaks in the reference that are not in the estimation. Furthermore, the precision P shows a relation between common peaks and the total peaks in the estimation. The recall shows a relation between the common peaks and the peaks in the reference. The F-measure can be seen as a measure that tell us if the true positives are significant in both reference and estimation.

At this moment, we reviewed some background and basic concepts such as audio representation, audio feature representation, audio matching and evaluation. Now, the following chapter aims to explain the methods applied, and the analysis of results. For detailed description of this fundamental concepts, we recommend [14] for the fundamentals of music processing. For electronic music theory, one can find interesting information in[18] and [3].

Chapter 3

Processing Pipeline

As we said, in this thesis, we apply fingerprinting techniques to structure and analyze electronic music. In this context, we use electronic music samples (loops) to obtain a better understanding of the parameters and components of fingerprint-based identification systems under the scenarios of: complex superposition, audio degradation, shift variances, and matching with shifted-queries. In this chapter, we describe our processing pipeline. First, we introduce our example data-set that we used for our initial experiments and then described the methods applied.

3.1 Example Data-set

For our initial experiments we collected a set of 12 loops which can be roughly categorized in 3 types: *percussive* (P), *melodic* (M), and *other* (O). Each loop is a mono (singel audio channel) or stereo (two audio channels) audio signal with a sampling frequency $F_s = 44100\,\text{Hz}$, a length of 8 seconds, and a musical tempo of 120 BPM [1]. This example data-set is a compilation of loops from the websites [6] and [9]. Table 3.1 shows a general description of each loop and it gives an idea of the kind of audio signals that we are working with.

3.2 Feature Extraction Configuration

Each loop is loaded, re-sampled to a sampling frequency $F_s = 22050\,\text{Hz}$, and converted to mono (if necessary) in order to reduce the computational work load. Closely following [11], we use a fingerprint-based feature representation. We compute peak maps using a modified version of the *Shazam* system described in [20]. First, we calculated the STFT of an audio signal using

[1] The term tempo refers to the musical time of a piece of music. A common measure of the tempo is the BPM (Beats per minute)[4].

Audio Name	Label	Category	Description
Drums 1	L-P1	Percussive	A music sample of drums [9]
Kick Snare	L-P2	Percussive	A music sample of drums [6]
Disco Drums	L-P3	Percussive	A music sample of drums [6]
Melody 1	L-M1	Melodic	Melody made with a piano [9]
Zen Synth 2	L-M2	Melodic	Synthetic melodic sequence [6]
120 Aftermath	L-M3	Melodic	Synthetic melodic sequence [6]
Kalm B4 the Storm	L-M4	Melodic	Synthetic melodic sequence [6]
Bass 1	L-O1	Other	A bass composition [9]
Plingers-Delight	L-O2	Other	A melody with special sound effects [6]
cm005 monastry phrase	L-O3	Other	Vocal phrases [6]
NewsJingle-Intro	L-O4	Other	Lead sequence [6]
Jazz Chords Piano	L-O5	Other	Two chords sound [6]

Table 3.1: Description of our example data-set.

a *Hann* window of size 4096 samples and a hop size $H = 2048$ samples. Then, the magnitude spectrogram and log-frequency spectrogram are computed as decribed in Section 2.3.2.

Following the general description in Section 2.4.1, a peak map $\mathcal{P} \in \mathbb{B}^{K \times M}$ with $\mathbb{B} := \{0, 1\}$ is constructed from a spectrogram representation $\mathcal{Y}$. Before the peak selection, an exponential decay is introduced in the spectogram $\mathcal{Y}$, which helps to select the maximum frequency value at the moment of appearance. A rectangular analysis window is made for each time-frequency bin $\mathcal{Y}(m, k)$. For our initial experiments, the windows size is 15x15 which means that each window covers 15 time-frames (m) and 15 frequency-indexes (k). Within each window, the maximum value $\mathcal{Y} > 1$ will set the bin output $\mathcal{P}(m, k)$ to 1 and the neighbor output to 0; peaks below 1 are considered noisy or irrelevant information.

A peak map (PM) and a log-frequency peak map (LPM) is computed using the magnitude spectrogram and the log-frequency spectrogram respectively. Figure 3.1 shows the magnitude spectrogram peak maps (PM) of our example data-set. The loops are organized along 3 rows labeled as P (percussive), M (melodic) and O (Other) respectively. For each PM shown, the frequency axis spans the range between 0 and 11025 Hz; all loops have a length of 8 seconds. Maximum peaks in the magnitude spectrogram of each loop are represented with black points. Note that in the top row, the peak maps show predominantly vertical structures. This is due to the fact that percussive sounds have noise-like onsets that cover a wide range of frequency bands. In the middle row, we see *melodic* loops, which are characterized by horizontal structures. In the bottom row, we show peak maps corresponding to the *other* category. This category contains loops with both *percussive* and *melodic* properties.

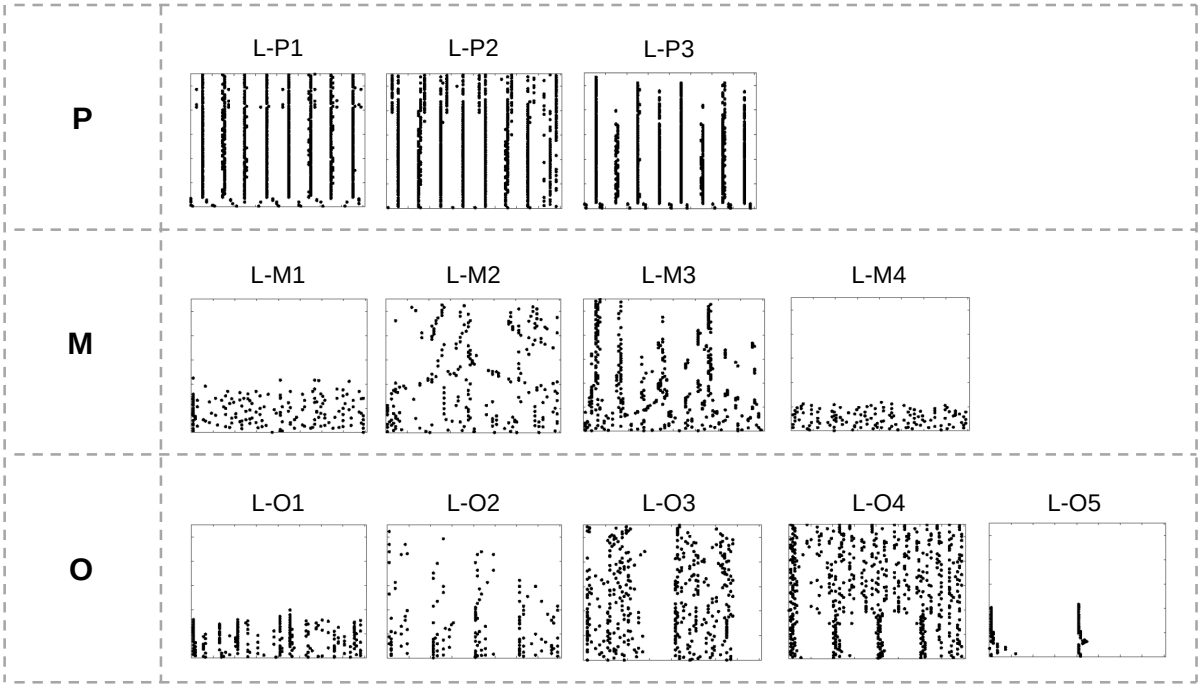

Figure 3.1: Magnitude spectrogram peak maps (PM) of the example data-set. This data-set consists of 12 loops where each of them has a duration of 8 seconds. The frequency axis of each peak map representation goes from 0 to 11025 Hz. Black points represent maximum peaks in the magnitude spectrogram of the corresponding loop. Loops are grouped among 3 types: P (*percussive* sound), M (*melodic* sound) and O (*other*). Each row represents one of these types.

3.3 Loop Combination

Our first task involve the study of fingerprint behavior under loop combination scenarios. For this, we constructed tracks with a duration of 8 seconds (same duration as the loops) in order to investigate the peak information in specific combinations. In Figure 3.2, we show the resulting peak map of a track with a duration of 8 seconds which was produced by the sum of 2 audio loops: L-O5 and L-M2.

When we match a loop query with tracks similar to the one described in Figure 3.2, we can see that the information related to the query may only be partially present. Additional information also appears, mostly due to the presence of other loops. In order to measure the query information contained within these tracks, we computed the *precision*, *recall* and *f-measure* retrieval information. As said in Section 2.5.2, *true positives* correspond to common peaks between the query and the track, *false positives* are peaks in the track which do not appear in the query, and *false negatives* are peaks in the query and do not appear in the track. For a visual example, Figure 3.3 shows a peak map retrieval representation of the track described in

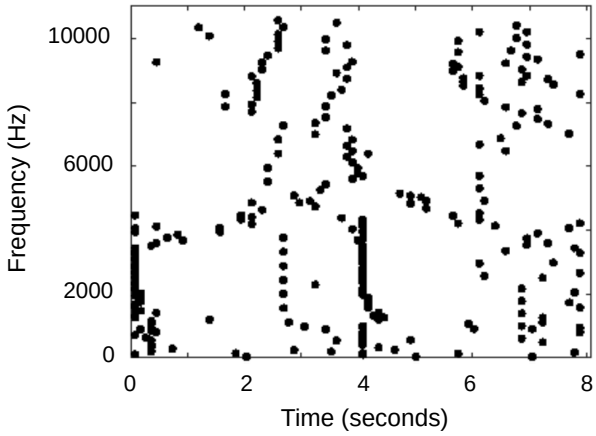

Figure 3.2: Peak map of an audio track produced by the sum of L-O5 and L-M2. The audio track has a duration of 8 seconds. Black points denote the maximum peaks found in the magnitude spectrogram of the audio track.

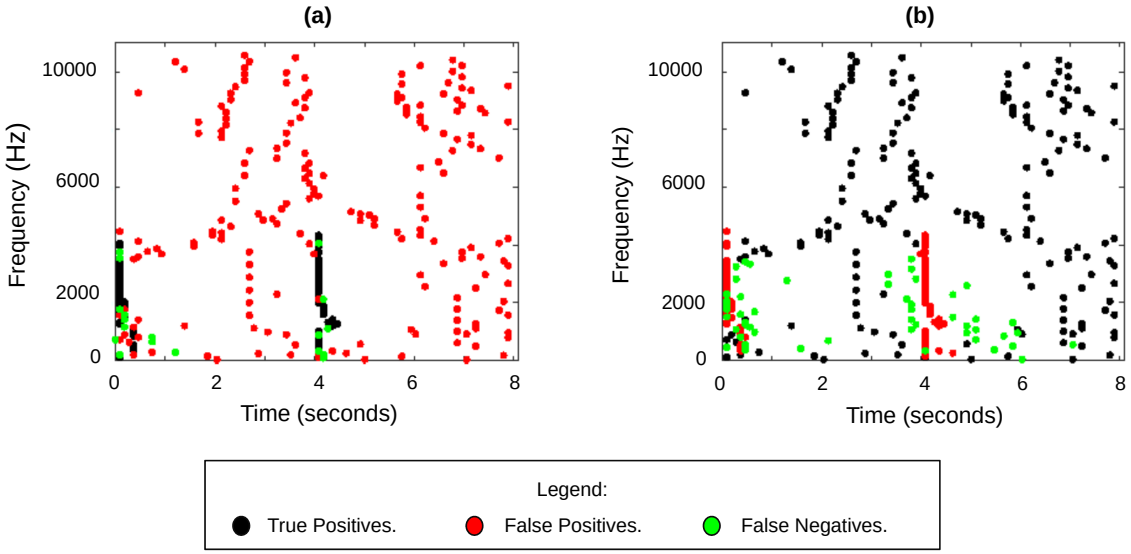

Figure 3.3: Peak map retrieval representation of an audio track. Black points represent common peaks between the query and the audio track (true positives), red points correspond to peaks in the audio track but not in the query (false positives) and green points denote the peaks in the query but not in the audio track (false negatives). In both (a) and (b), we used the track from the Figure 3.2. The query in (a) is the loop L-O5 and the query in (b) is the loop L-M2.

Figure 3.2. In (a), the loop L-O5 is the query whereas in (b), L-M2 is the query. In both (a) and (b), black points represent common peaks between the query and the track (true positives), red points correspond to peaks in the audio track but not in the query (false positives) and green points denote the peaks in the query which are not present in the track (false negatives).

When an audio combination is done, it may happen that some peaks of the query in the

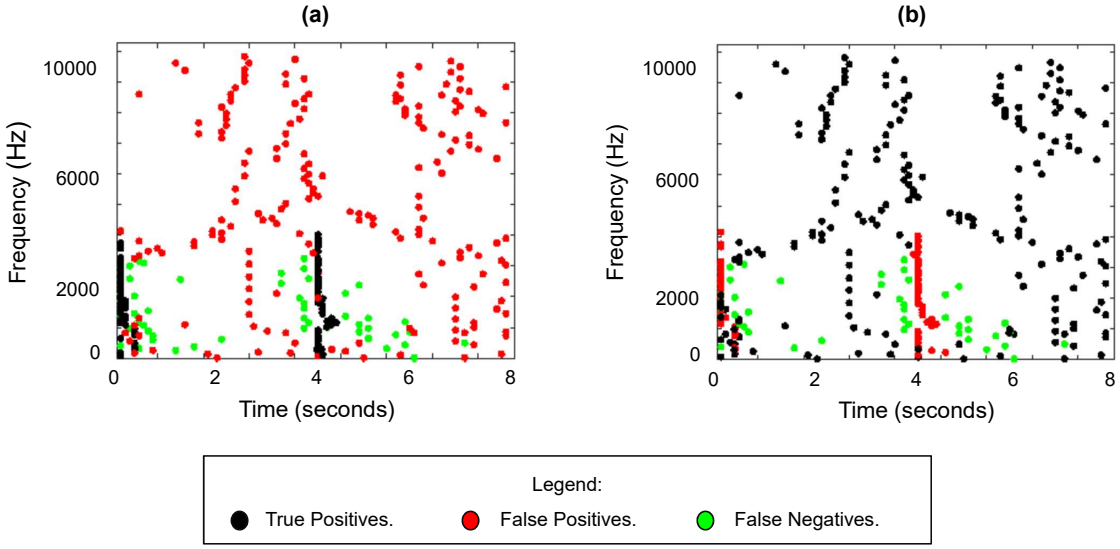

Figure 3.4: Peak map retrieval representation of an audio track where peaks with one time-shift (left and right) or one frequency index (up and down) differences can be a match. Black points represent common peaks between the query and the audio track (true positives), red points correspond to peaks in the audio track but not in the query (false positives) and green points denote the peaks in the query but not in the audio track (false negatives). In both (a) and (b), we used the track from the Figure 3.2. The query in (a) is the loop L-O5 and the query in (b) is the loop L-M2.

combination might be shifted. Thus, during this thesis, we try to match a peak at the exact time-frequency component, one time frame on the left, one time frame on the right, one frequency index up, and one frequency index down. Figure 3.4 shows the peak map retrieval representation of Figure 3.3 when one frame shift is taken into account. As we can see, there are more true positives and less missing peaks (false negatives).

In our first experiments, we constructed tracks using 4 loops of our example data-set. Each track has a duration of 8 seconds and it is produced by an unique combination among the 4 chosen loops. The retrieval information is computed using each loop as a query. With this experiment, we introduce our audio sample retrieval matrix representation shown in Figure 3.5. Rows are are assigned to the loops L-P2, L-M2, L-O4, and L-O2. Columns represent the 15 possible tracks. Colored points in each cell denotes the presence of the loop in the corresponding track. For a better visualization, different colors have been assigned to each loop: blue (L-P2), green(L-M2), red(L-O4), and cyan (L-O2). Precision (a), recall (b) and F-measure (c) are represented by colored squares. Color-bars were adjusted to enhance the visualization.

From the audio sample retrieval matrix, one can easily compare the peak survival information between loops when they are combined. In Figure 3.5, columns 1 to 4 show a comparison between loops because tracks 1 to 4 are a replica of the corresponding loop. A perfect match (the retrieval

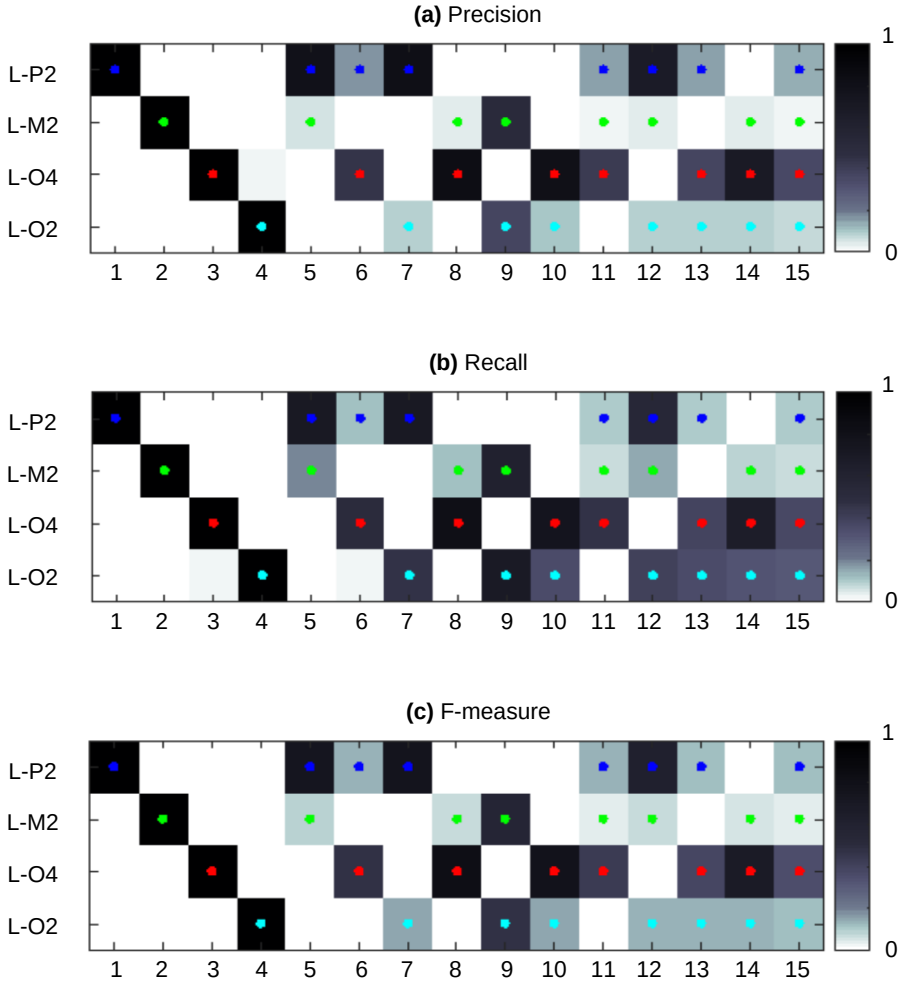

Figure 3.5: Audio sample retrieval matrices. In all matrix representations, rows represent audio samples (loops) of our example data-set: L-P2, L-M2, L-O4 and L-O2. Columns represent tracks with a duration of 8 seconds which were produced by possible unique superposition among the previously mentioned 4 loops. Colored points indicate the presence of the loop in the track. A different color has been assigned to each loop: blue, green, red and cyan to L-P2, L-M2, L-O4 and L-O2 respectively. All color-bars were adjusted in order to enhance the visualization. Each cell contains the retrieval results when the corresponding loop is the query. The matrix representations describe the *precision* (a), the *recall* (b), and the *F-measure* (c).

result is equal to 1) happened when the loop is matched to its replica. It might happen that loops have common information. If we see closely in (a), when L-O4 (the query) is matched to the track 4 (replica of L-O2) we see that there is a considerable amount of L-O4's information contained in L-O2. In the case that L-O2 is the query and is matched to track 3 (replica of L-O4), the precision is irrelevant since the amount of information of L-O2 in L-O4 is low. However, in (b) the recall is relevant which lead us to the idea that a significant ammount of peaks in L-O4 are also in L-O2. In other cases of loops comparison, there is no relevant common information

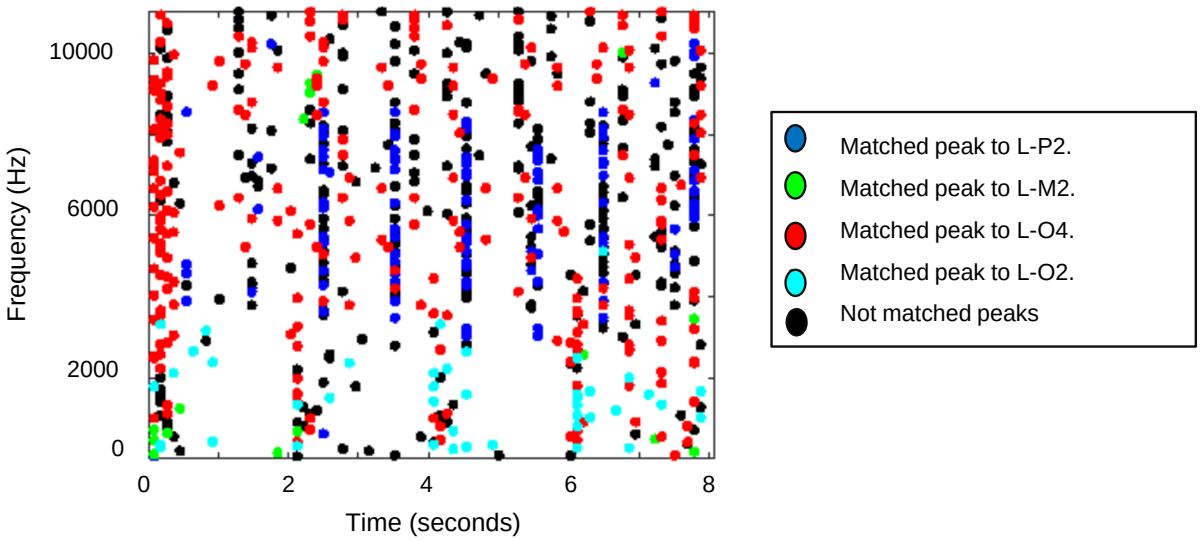

Figure 3.6: Peak map representation of track 15. The frequency axis spans the range between 0 and 11025 Hz. Blue, green, red and cyan matched peaks when the query is L-P2, L-M2, L-O4 and L-O2 respectively. Black peaks denote information produced by the overall combination.

(retrieval results are nearly close to zero). Columns 5 to 15 in Figure 3.5 can help to analyze the survival information of loops in different possible combinations. For example, from the results in track 15 (Figure 3.5), we can see that in (a) around 30% of the information in the track matches to L-O4. From (b), we can see that more than 25% of the peaks in L-O4 and L-O2 survived the combination. The F-measures in (c) shows the accuracy of the combination, in other words, this measure give an evaluation of the survival peaks with respect to the track and query. If F-measure is high, true positives are a relevant amount not only in the query but in the track. In (c), we can see that most of the original peaks in L-O4 survived the combination of the track 15 (around 30%) whereas the survival peaks of L-O2 represents low information in track 15(5%) and the query (7%). This information can hardly be seen using a peak map retrieval representation. In Figure 3.6, we shows the peak map representation of track 15. Each loop was matched to the track and a different color was assigned to the matched peaks; blue (L-P2), green (L-M2), red (L-O4) and cyan (L-O2). Unmatched peaks produced by the combination of loops are denoted as black points.

Another interesting case in Figure 3.5 is track 6 when is matched to the query L-O2 (which is not present in the track). In this case, the result in (b) shows that the query has relevant amount of information produced by the combination. This can indicate that when L-O2 is added to track 6, significant information in L-O2 will may be positive affected (see recall of track 13).

A second experiment consists of measuring the survival information when the query is matched to different complex mixes. During this experiment, the term *complex mix* refers to the number

of loops overlapped (including the query). The complexity increase with the number of loops combined in the track. Figure 3.7 gives an example of 10 tracks with different complexity (from 1 to 10); the length of each track is 8 seconds. Track 1 has the lowest complexity whereas track 10 has the highest complexity.

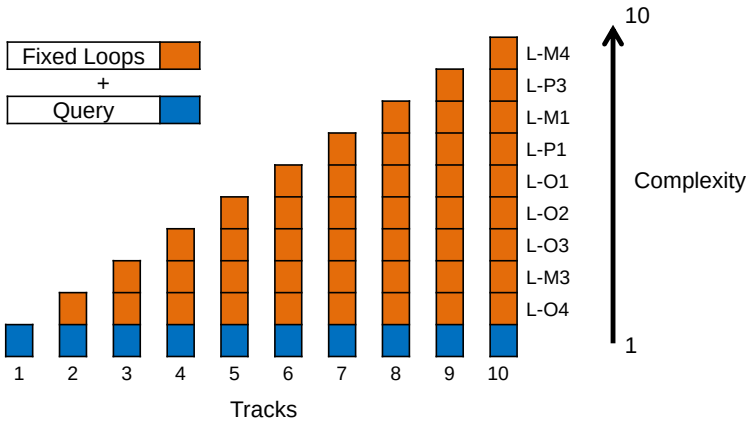

Figure 3.7: Complex mixtures. A total of 10 tracks were generated with different complexity. Tracks were constructed by the superposition of loops of our example data-set according to the following order: query, L-O4, L-M3, L-O3, L-O2, L-O1, L-P1, L-M1, LP3, L-M4. The complexity is increasing from track 1 to track 10. Track 1 has the lowest complexity whereas track 10 has the highest complexity. Blue blocks indicate the presence of a loop query. Orange blocks denote the presence of additional loop.

The retrieval information was computed for these complex tracks. In Figure 3.8, the F-measure results for the loops L-P2, L-M2, and L-O4 are shown. Vertical axes are in a logarithmic scale and represent the F-measure. Horizontal axes correspond to the *complexity* (number of loops contained in the track). Loops were added according to the order described in Figure 3.7. The orange, green and blue lines correspond to the F-measure when the query is L-O5, L-M2 and L-P2 respectively. The dashed line is a base-line computed by averaging F-measure results of 50 random queries (noise audio loops and real audio loops) that are not contained in the constructed tracks. In (a), magnitude square peak maps (PM) were used to compute the F-Measure. In (b), log-frequency peak maps (LPM) were used. In both (a) and (b), we can see that when we add the second loop L-O4 to the query (see Figure 3.7 complexity 2), less than 30% of the spectral peaks survived the combination. From complexity 2 to 10, results continues slowly decreasing or increasing, depending on the complexity and the types of loops combined. In the case where the F-measure of a query goes below the base-line (see figure (b), loop L-O5 with complexity 7 and 8), it means that relevant information of that query didn't survive to that kind of complex configuration. Thus, this loop might be hardly recognizable in the matching process when that particular combination is presented.

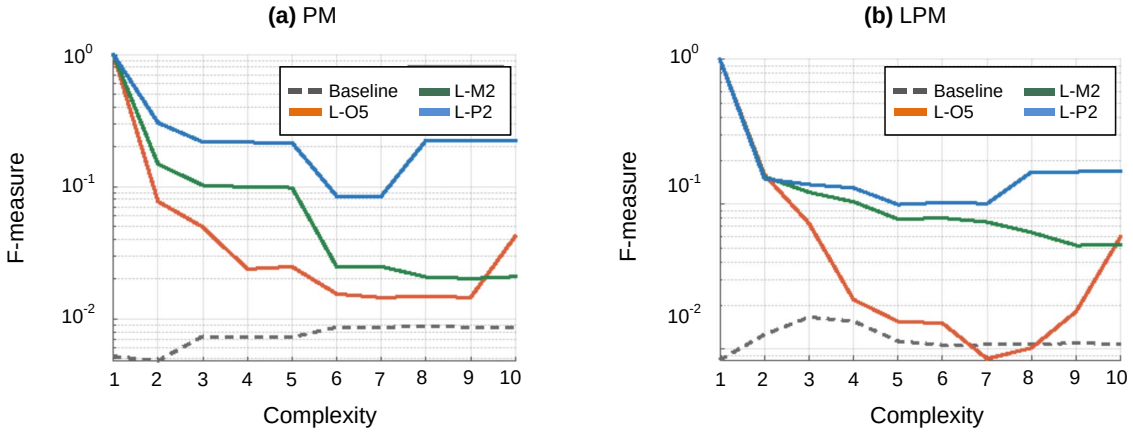

Figure 3.8: F-measure performance when the loops L-O5, L-M2 and L-P2 are matched to tracks with different complexity. Horizontal axes represents the complexity of the track. The term complexity states to the number of loops superimposed in the track. The track was constructed by the superposition of loops of our example data-set according to the following order: query, L-O4, L-M3, L-O3, L-O2, L-O1, L-P1, L-M1, LP3, L-M4. Vertical axes represents the F-measure in a logarithmic scale. (a) shows the results using magnitude square peak maps. (b) shows the results using log-frequency peak maps.

3.4 Audio Degradation

For the study of fingerprinting under audio degradation scenarios, we applied 9 different types of degradation techniques which are grouped into two subsections: adding external sound and adding effects.

3.4.1 Adding External Sound

In our experiments of adding external sounds, we used the audio degradation toolbox described in [13], as well as 4 different sounds from the toolbox data-set: random white noise, vinyl sounds (old-dusty-vinyl-recording.wav), pub environment (restaurant08.wav), and headphone noise (hum-50Hz-from-headphone-plug.wav). We constructed tracks by adding an external sound to a loop query with a specific signal to noise ratio (SNR)[2]; each track has a length of 8 seconds. Figure 3.9 gives a visual example of how each track was constructed. There are a total of 22 tracks where the first track contains only the original sound. Tracks 2 to 22 are a combination of the query and the external sound with SNR from 60 dB to 0 dB respectively.

Figure 3.10 shows the F-measure for the 4 audio degradation experiments. The results correspond the loops: L-P2 (blue), L-M2 (orange) and L-O4 (green). Solid lines correspond to magnitude

[2]The signal to noise ratio (SNR) is defined as the ratio of the signal power to the noise power [15]. In this thesis, the SNR is expressed in decibels (dB). A ratio higher than 0 dB means higher signal power than noise power.

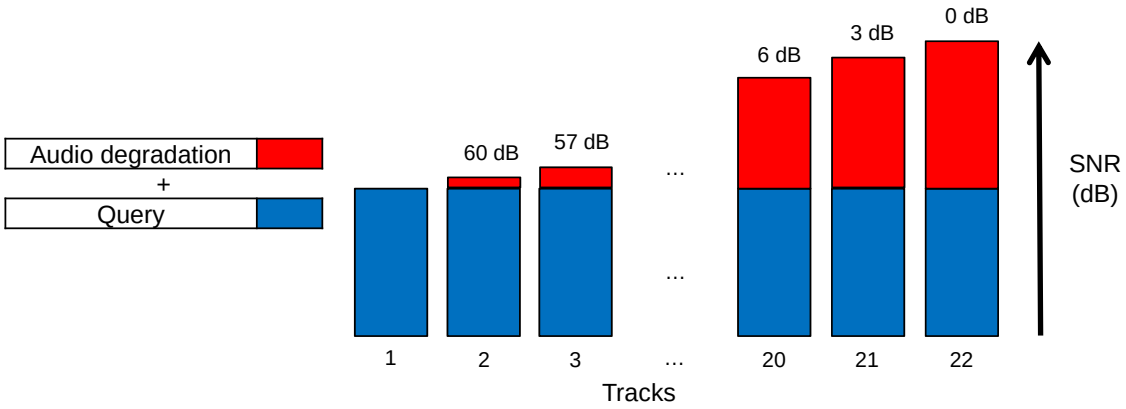

Figure 3.9: Adding external sound degradation. A total of 22 tracks were generated, each of them with a duration of 8 seconds. Track 1 contains the original track whereas the rest contains the query and an audio degradation with a specific SNR. Tracks 2 to 22 span the range of SNR from 60 dB to 0 dB respectively. Blue blocks indicate the presence of a loop query. Red blocks denote the presence of an external sound.

spectrogram peak maps (PM) while dashed lines indicate the use of log-frequency spectrograms (LPM). Figures (a), (b), (c), and (d) correspond to the experiments of: white noise, pub environment, vinyl noise sound, and headphone noise sound. Horizontal axes spans the range of SNR 60 dB to 0 dB. The first value is always one, because the track corresponds to the replica of the loop. For all types of degradation, we can see that L-M2 decreases faster when compared to the other loops. This is due to the fact that spectral peaks in L-M2 are more spread which means peaks are more vulnerable to noisy distortions. Loops L-O4 and L-P2 are more resistant to noise changes, where in most cases L-O4 presents a higher F-measure. Furthermore, in cases (a) and (c), the decreases in all loops start around 50 dB while in cases (b) and (d), the decrease started around 40 dB. When SNR = 0 dB, the external sound has the same intensity as the loop, and in most of our results the F-measures are below 0.5. For a comparison among the noisy sounds, we can see that loops are more resistant to headphone noise since the spectral peaks of the query start to decrease from SNR 40 dB. In addition, white and vinyl noises add more distortion to the loops with SNR < 55 dB. Peak maps and log-frequency peak maps have a similar behavior within the pub environment and headphone degradation. However, LPMs have a noticeable higher performance for the vinyl and white noise degradation.

3.4.2 Adding Effects

We conducted further degradation experiments with the following audio effects: reverberation, delay, volume changes, linearly increasing the volume and linearly decreasing the volume. The methods applied to construct tracks for each experiment are described in the following sections.

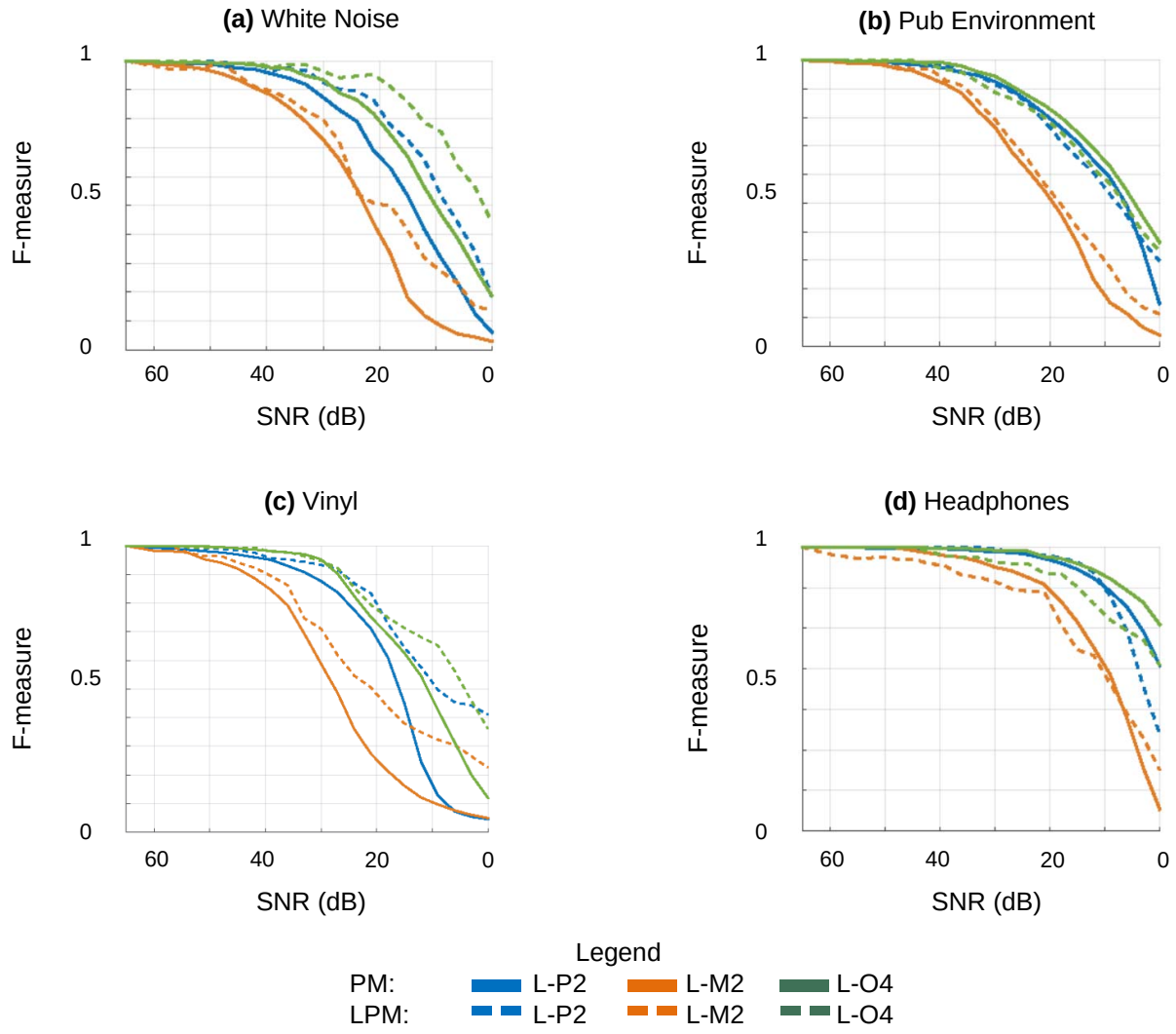

Figure 3.10: F-measure vs. SNR (dB) results when an external sound is added to the loop. Vertical axes show the F-measure on a linear scale. Horizontal axes span the SNR range from 60 dB to 0 dB. The first value is always one, because the track corresponds to the replica of the loop. Solid lines indicate the use of magnitude spectrogram peak maps (PM). Dashed lines indicate the use of log-frequency peak maps (LPM). The results of 3 loop queries of our example data-set are shown. Blue, orange and green correspond to the loop queries L-P2, LM2 and L-O4 respectively. In (a), white noise was added to the loops. In (b), a sound that simulates a pub environment was superimposed. In (c), vinyl noise sound was added. In (d), a sound that simulates noise from headphones was added to the loops.

3.4.2.1 Reverberation and Delay

Reverberation is one of the most common effects applied in electronic music [3], [18], [2]. There is a wide range of reverberation configurations. However, we modeled a simple effect using the

Matlab function *reverb* (available on Matlab version R2016b). A total of 21 tracks were built with different *diffusion* factors, from 0 to 1. This parameter is associated to the density of the reverb tail. A diffusion closer to 1 means that all reflections are pushed together. If the diffusion is closer to 0, more discrete echoes are created. The other input parameters of the function were set to their default values: $Pre\text{-}delay = 0$, $HighCutFrequency = 20000\,\text{Hz}$, $DecayFactor = 0.5$, $HighFrequencyDamping = 0.0005$, $WetDryMix = 0.3$. The F-measure for the loops L-P2 (blue), L-M2 (orange) and L-O4 (green) are shown in Figure 3.11 (a). In this particular experiment, the melodic loop shows a higher loss of information (low F-measure). In the case of diffussion = 0.9, the F-measure for L-P2 and L-O2 start to decrease, for L-M2 slowly increase; and they end with 0.6 (diffussion = 1) because at this point the reflections are pushed together and produce a noisy sound combination that affects the spectral peaks of loops. However, when the reflections are more spread from each other, F-measures slowly decrease and increase. From a diffusion factor of 0 to 0.9, L-P2 shows a higher F-measure whereas L-M2 shows the lowest. LPMs have higher performance for L-P2 while PMs have higher performance for L-M2.

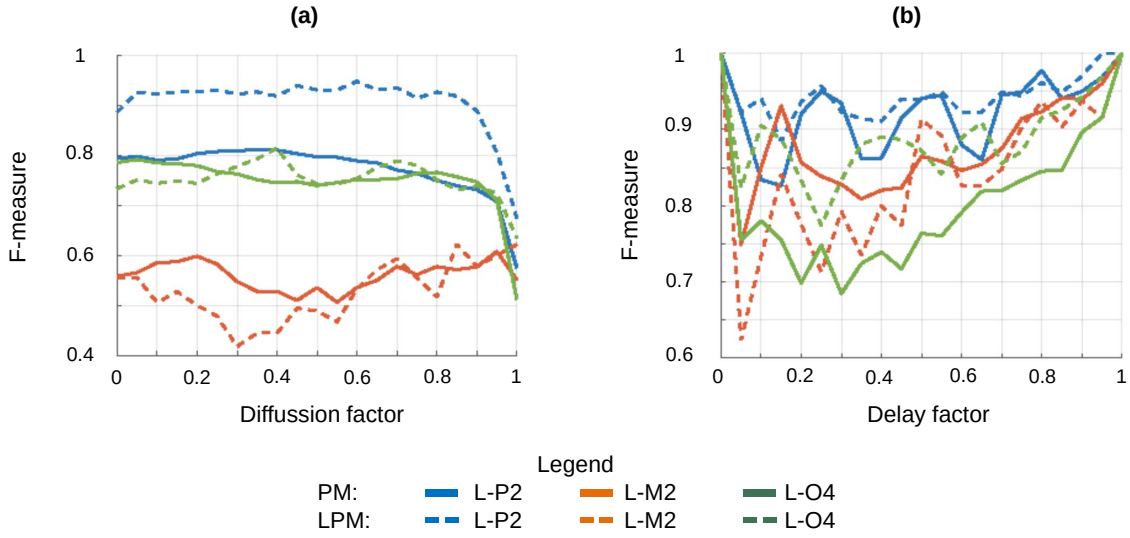

Figure 3.11: F-measures for reverberation and delay. Vertical axes show the F-measure on a linear scale. Solid lines indicate that the magnitude spectrogram peak maps (PM) were used. Dashed lines indicate that log-frequency peak maps (LPM) were used. The results of 3 loop queries of our example data-set are shown: L-P2 (blue), LM2 (orange) and L-O4 (green). In (a), a reverberation effect was applied. The horizontal axis spans the diffusion factor range of 0 to 1 and the vertica axis spans the range of 0.4 to 1. In (b), delay was applied. The horizontal axis spans the delay factor range of 0 to 1 and the vertical axis spans the range of 0.6 to 1.

Adding Delay is another common effect applied in electronic music in order to cause an impact on the music [3], [18], [2]. One can find a wide variety of delay configurations. For our initial experiments, we focused on a simple scenario which consist of adding one delayed version of the loop withing its activation time. Therefore, a total of 21 tracks were built with a different *delay*

factor, from 0 to 1. The delayed version is activated with half of the original volume level and linearly decreases within the loop activation time. In Figure 3.11 (b), we can see the F-measure results for L-P2, L-M2, and L-O4. The same notation as in (a) were used in the figure. There is a sharp drop in F-measure between the delay 0 and 0.5. After this initial drop, the F-measure oscillates with an overall increasing behavior. The oscillation are due to the fact that the delayed copy contains the same musical relations which in certain delay values can be severe to the loop (minima) or less destructive (maxima). As the delay factor is increasing the overall F-measure is also increasing since less sections of the loop are affected by the delayed copy.

3.4.2.2 Volume Changes

In electronic music, loops might appear with an adjusted volume to create tension and variety in the music. As a first experiment with volume changes, we computed tracks using the loops with different *volume-level* factors $\in [0, 1]$. The resulting track is the loop query with a different intensity level; the volume goes from 0 to the original volume of the query. The F-measure results of the loops L-P2 (blue), L-M2 (orange) and L-O4 (green) are shown in Figure 3.12 (a). The horizontal axis spans the range of 0 to 1. Solid lines indicate the use of magnitude spectrogram peak maps (PM) and dashed lines indicate the use of log-frequency peak maps (LPM). In (a), we can see that LPM performs better when compared to PM. When we have around 20% of the original volume, more than half of the loop's information is captured by the feature representation. In this case, the amplitude limitations in the peak selection process of a peak map plays an important role. As said in Section 3.2, the exponential decay and the amplitude threshold avoids to choose noise or irrelevant peaks. Thus, when the volume-level factor is low, peaks are not chosen because of lower amplitude values in the spectrogram. If we apply such amplitude limitation with a lower exponential decay factor and lower amplitude threshold, more information is captured with lower volume levels and the F-measure increase faster than in (a).

As a second experiment, we applied linearly increasing volume changes in a track. Following the ADSR envelope modell [14], we can define the increasing volume phase of a loop query as an *attack phase*, and the steady section as a *sustain phase* (portion with the original volume). We constructed tracks using the loops with an attack factor. This factor indicates what starting portion of the loop will be modified. All resulting tracks start with an increasing section (attack phase), then continue with its original intensity until the end of the loop is reached (sustain phase). In the attack phase, the volume-level increases linearly from 0 to the original query volume. For example, a factor of 0.5 means that from the beginning of the audio loop until its midpoint, the volume increases. The rest of the audio loop is not affected. Using the same loops and notation in Figure 3.12 (a), the F-measure results are shown in Figure 3.12 (b). The vertical axis spans the range of 0.5 to 1. The results of this experiment show a decreasing F-measure.

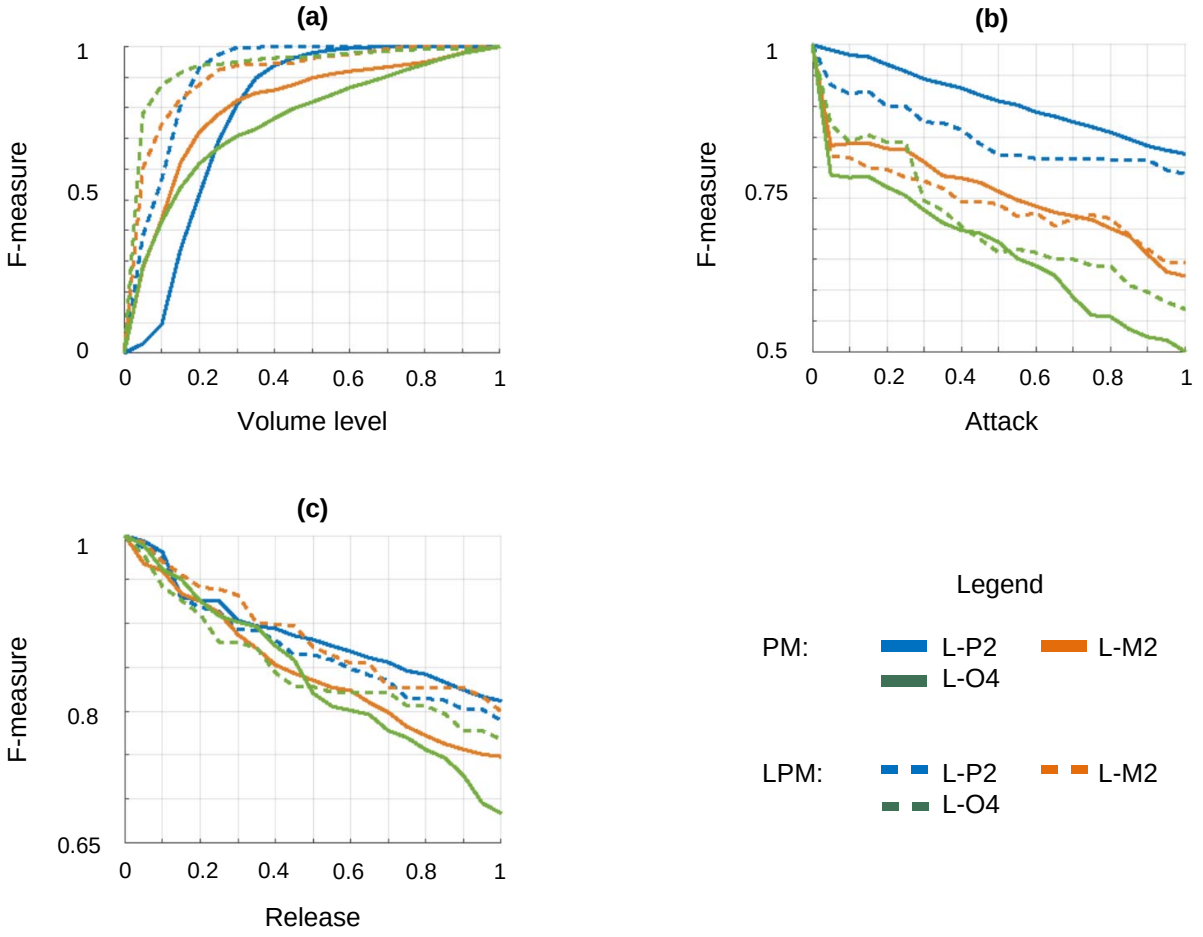

Figure 3.12: F-measure for 3 different volume changes scenarios. The queries are: L-P2 (blue), L-M2 (orange), and L-O4 (green). Solid lines indicate the use of magnitude spectrogram peak maps (PM) and dashed lines indicate the use of log-frequency peak maps (LPM). Figure (a) shows results after uniformly adjusting the volume level of the query. When volume-level factor is equal to 1, the original intensity level of the loop is used in the track. A volume-level of 0 means there is no sound. The vertical axis spans the range of 0 to 1. Figure (b) shows the results after an attack phase on the loop is applied. The attack factor $\in$ [0,1] indicates what starting section of the loop will be increasing. The vertical axis spans the range of 0.5 to 1. In (c), an ending section of the loop decreases the volume intensity to 0 (release phase). The vertical axis spans the range of 0.65 to 1.

This is due to the fact that when the attack factor is close to 1, less information is captured by the peak maps. However, in the case where the increasing factor is equal to 1, the F-measure is greater than 0.5. Also, the amplitude limitations in the peak selection play an important role since peaks with lower intensity are not chosen. As the attack fator increase, more peaks are not captured by the peak selection process.

As a third experiment of volume changes, we created tracks using the loops with a decreasing

section. Following the ADSR envelope modell [14], we can define this section as an *release phase*. This method is similar to the previous experiment. All tracks start with a sustain phase and end with a release phase. The *release* factor indicates what portion of the loop will be modified. For all the values of *release*, the volume-level reduces from the loop's original volume to 0. For example, a release factor of 0.5 means that from the midpoint of the audio, the volume is linearly reducing (release phase). The rest of the audio loop is not affected (sustain phase). Following the same notations of previous experiments, the F-measure results of the loops L-P2, L-M2 and L-O4 are shown in Figure 3.12 (c). The F-measures decrease and reach a value more than 0.65 when release = 1. Here again, we see the effects of amplitude limitations in the peak selection process. Less peaks are chosen when the release factor is increasing. The performances in both cases (b) and (c) have the same behavior, but different decreasing rates. From these particular results, one can see that more feature information survived when a release phase is used rather than when an attack phase is used. In other words, relevant features are captured when the beginning of the loop is in its original form (decreasing-volume case), rather than having an unmodified ending section (increasing-volume case).

3.5 Time Shift Differences within a STFT window frame

In real world cases, a loop can be activated several times within a track. STFT framing has an effect on the resulting feature representation. When the STFT is computed, in both query and track, the information of the isolated loop query and the information of the loop within the track can be different since the position of the signal inside the window-frames can also be different. Figure 3.13 shows a simple example of this case. Signal samples are denoted by blue points whilst zero-padding is indicated by black points. Dashed lines denote the information captured by a frame in the STFT. The signal on the top has a total of 11 samples and the resulting STFT of this signal, with a window size of 9 samples and hop size of 4 samples, has 2 frames. The signal in the middle row is a shifted version of the signal above; the shift step is 2 samples. These two signals can have considerably different feature representations. This is because the frames in both signal contain different information inside. The STFT of the signal at the bottom results in 3 frames, where the last 2 frames contain almost all the original information of the signal.

Motivated by this case, we want to study the effects of shifting the audio loops in time. following the same procedure as in Figure 3.13, we created tracks which contain a loop signal shifted in time. The loop signal, with sampling frequency $F_s = 22050\,\text{Hz}$, is shifted by increments of 64 samples (2.9 ms) until the sample 4096 (185,8 ms) is reached, yielding a total of 64 tracks (shifted loop versions). The STFT is computed for each shift step by means of zero padding. The first frame is centered with respect to the time $t = 0$ seconds of the signal, using a Hann window with a size of 4096 samples, and a hop size $H = 2048\,\text{samples}$. The query is matched

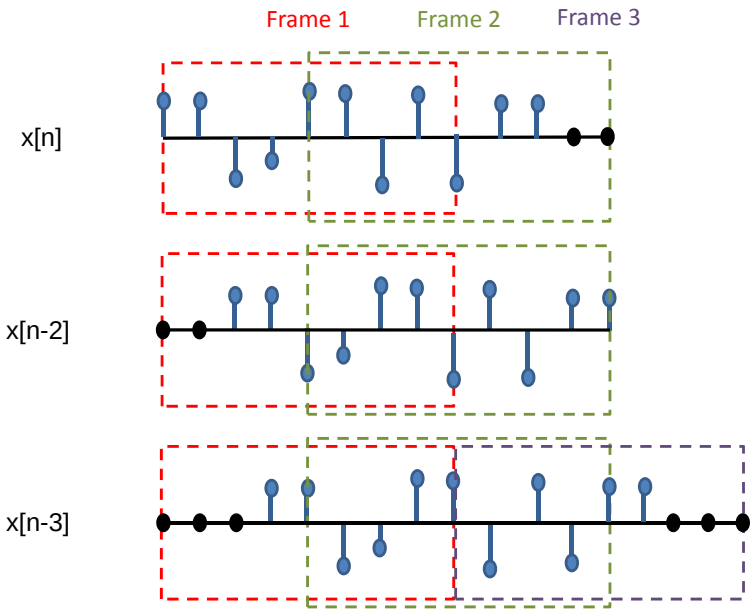

Figure 3.13: Effects of shifting a signal by zero-padding. The signal samples are denoted as blue points. The STFT is computed by means of zero-padding (black points). A signal $x[n]$, $x[n-2]$ and $x[n-3]$ are shown from top to bottom. Dashed lines denote the information captured by a frame in the STFT.

to the resulting tracks. When we do a comparison, the first 2 and last 2 frames are not taken into account in order to avoid distortions from the zero-padding. In Figure 3.14, we show the resulting F-measure for the loops L-P2 (blue), L-M2 (orange), and L-O4 (green). Following the same notation as previous figures, solid lines correspond to PM and dashed lines to LPM. In (a), we considered a true positive as the peak in the query that matches to a peak in the track at the same time-frequency index. In (b), we considered a true positive as the peak in the query that matches to a peak in the track at the same time-frequency index, one time-frame on the left, one time-frame on the right, one frequency index on up, or one frequency-index down. The STFT is increasingly different to those computed with shift zero until the *hop* size is reached in the case of (a), and until half of the *hop* size in the case of (b). A minimum is reached in shift 1984 in (a) and in shift 1024 in (b). As we can see, if we search for matched not only at the exact time-frame position but also at neighbor time-frequency indexes, the similarities between shifted versions increase.

3.6 Matching with Shifted Queries

In Section 3.5, it was interesting to see that a loop computed with different time-frame positions can results in such different STFTs. With this in mind, we came to the idea of finding a loops within a track by using several shifted versions of the query. First, we created tracks which

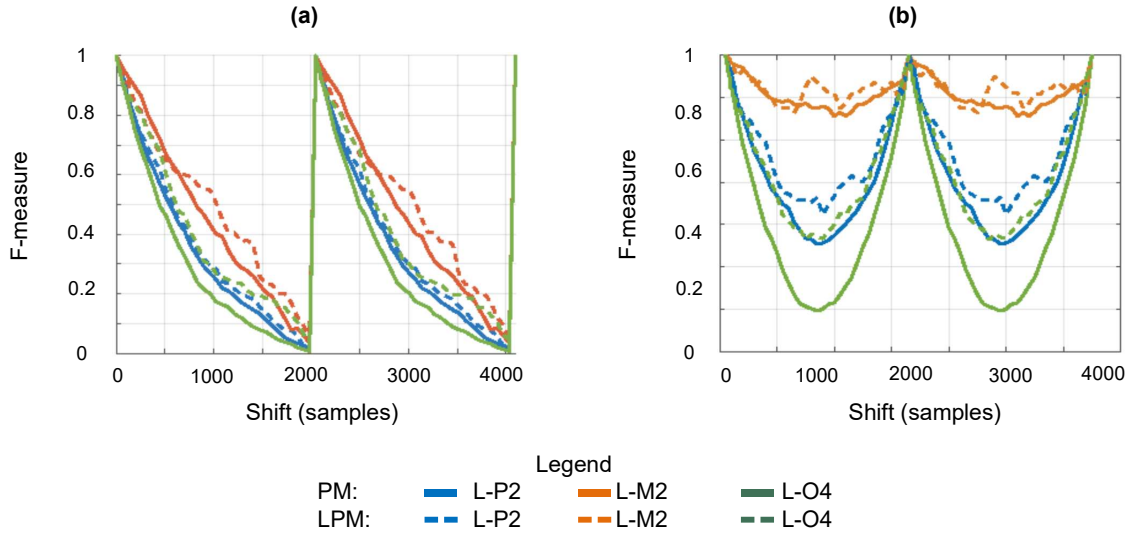

Figure 3.14: F-measure results when an audio sample is compared with a shifted-version. The results of the loops L-P2 (blue), L-M2 (orange) and L-O4 (green) are shown. Solid lines indicate the use of magnitude spectrogram peak maps (PM) and dashed lines indicate the use of log-frequency peak maps (LPM). In (a), we considered a true positive as the peak in the query that matches to a peak in the track at the same time-frequency index. In (b), we considered a true positive as the peak in the query that matches to a peak in the track at the same time-frequency index, one time-frame on the left, one time-frame on the right, one frequency index on up, or one frequency-index down.

contain one loop repeated a number of consecutive times. Then, we computed several shifted versions of the loop query, form shift 0 (zero) to 1024 sample (half of the Hop size). The original loop is compared with tracks by means of the F-measure since we want to evaluate the matched peaks in both track and query; the matching procedure is applied as described in Section 2.4.2. We try to find matches not only with peaks in the same time-frequency index but also with the neighbor peaks. The results are saved as a *query similarity* variable. Features from the query and track do not include the first 2 and last 2 frames. This decision was made because these two frames, in the query STFT, can contain distorted information caused by the zero-padding.

Secondly, we proceed to do a matching between the original loop query (no shift added) and the track. In Figure 3.15, we can see the F-measure results for 3 different shifted-query versions of L-P2 when they are compared with the other shifted versions. The queries correspond to the loop versions $x[n]$ (blue), $x[n-1024]$ (red) and $x[n-1984]$ (green). Blue points indicate the F-measure results when the original query ($x[n]$) is compared to the versions $x[n-1024]$ and $x[n-1984]$. As we can see, the maximum diference between the versions is 0.55 which can help us to avoid irrelevant matching values.

As a third step, we proceed to compute a matching curve as described in (see Section 2.4.2). In this case, the similarity measures aims to quantify how much information of the query is

contained withing a track. We use the F-measure as a similarity measure. When the similarity measure between the original loop (shift zero) and the track is close to a *query similarity* value, we compare again using the corresponding shifted version. This procedure is done because we assume that a similarity measure with a shifted version will result in a higher value or similarity than using the original loop. The resulting matching curve will contain enhanced peaks that can help to locate the activation times of the query. Furthermore, a threshold of similarity can be set by using the query similarity value between the shift 0 and the shift 512. If a similarity is lower than this value, it can be considered irrelevant (this assumption is made because this is a simple case scenario where no other sounds are added).

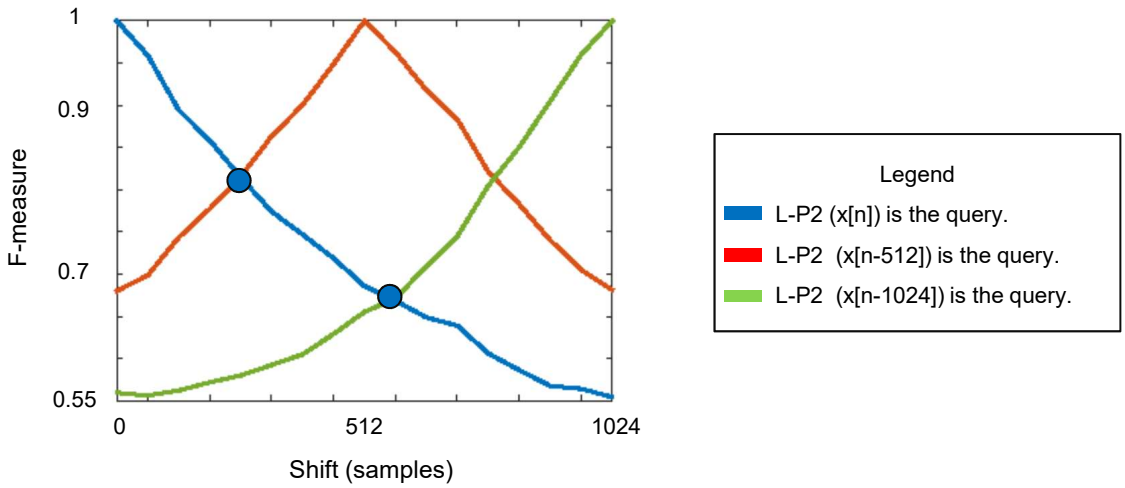

Figure 3.15: F-measure results for time-shift comparison of the audio signal L-P2. Blue, red and green lines denote the results when the original loop ($x[n]$), the loop shifted by 512 samples ($x[n-512]$) and the loop shifted by 1024 samples ($x[n-1024]$)are the respective queries. Blue dots indicate the F-measure when the original loop is compared to the other queries.

Using our example data-set, we created 12 tracks which contain one loop repeated 10 times with no information in between repetitions. We computed the matching curve when different shifted versions of the query were used. The shift is between the samples 0 and half of the *hop* size (1024) where the first query is the original loop and the second query version correspond to the shift 1024. Then, the *query similarity* is computed by means of the F-measure (similarity measure). Figure 3.16 shows the matching curve for the loops L-P2, LM2, and L-O4. Triangles denote the ground truth activation times. In the curves on the left (a) only the original query was used in the identification process. On the right side, the curves in (b) show the results when 3 versions of the query were used. When we compare both (a) and (b), we can see that the peaks in (b) are enhanced. For percussive loops like L-P2, the curves in both cases do not show clear activation times. This kind of loops contain similar repetitive patterns (see Figure 3.1) which can give high similarity results on local comparison along the track.

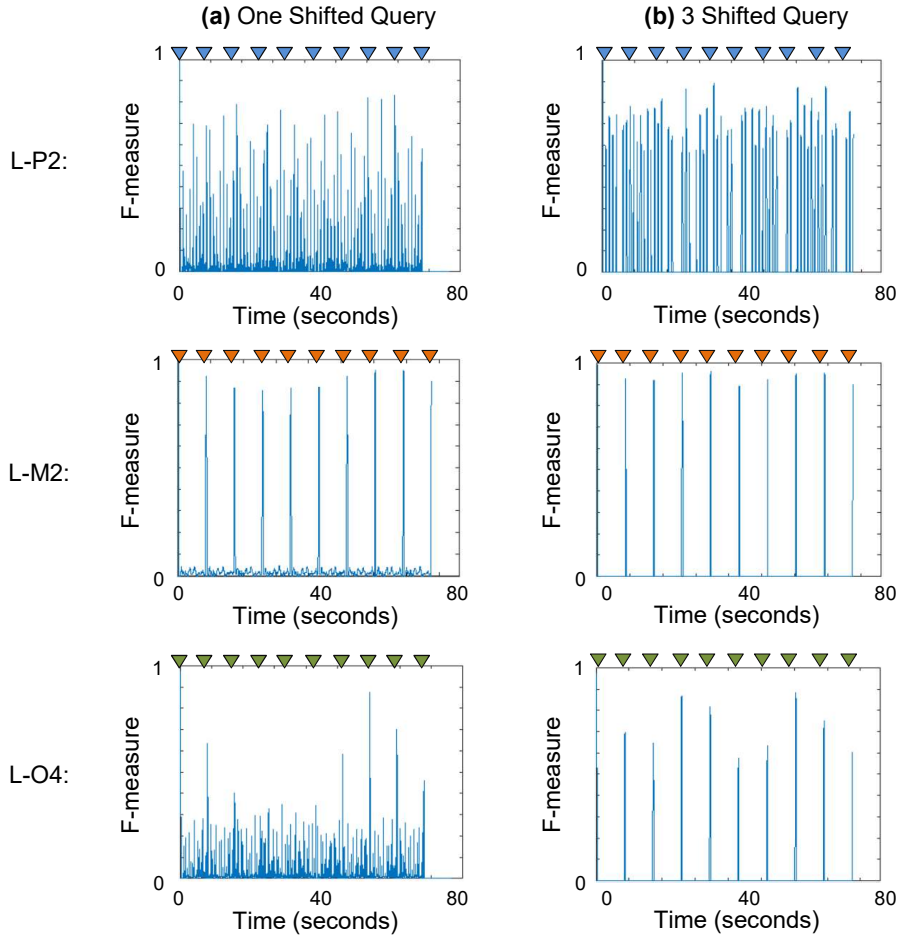

Figure 3.16: Matching curve for the loops L-P2, L-M2 and L-O4. For the curves on the left side (a), we used only the original query. For the curves on the right side (b), we used three shifted queries. Triangles indicate the activation times of a query in the track.

In order to compare the matching performance in each track with respect to the number of time-shifted queries used, we computed the gain ratio $\text{Gain}_{\text{Ratio}}$ and Pearson correlation $\text{Pcorr}_{\text{coef}}$ mentioned in Section 2.5. Figure 3.17 shows the corresponding Pearson correlation (a) and gain ratio (b) results for loops L-P2 (blue), L-M2 (orange), and L-O3 (green). Magnitude spectrogram peak maps were used. As we can see in the figure, both the gain ratio and Pearson correlation for PMs increases when a 2 loop version is used in the matching procedure. For the loop L-P2, both gain and Pearson correlation slowly increase because of the repetitive peak patterns within the loop PM [11]. On the other hand, for L-M2 and L-O2, the increase is stronger in both (a) and (b) with 2 shifted queries, then the increase is slower for higher number of queries.

At this moment, we have discussed methods such as the feature extraction based on peak maps, F-measure for audio sample evaluation, and the gain ratio and Pearson correlation for evaluation. Furthermore, we described the experiments applied for loop combination, audio degradation,

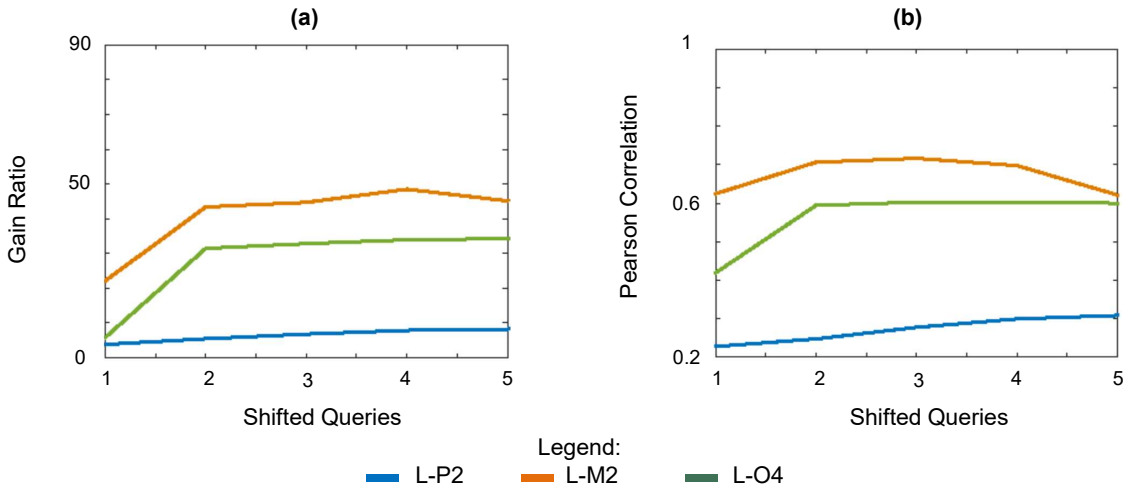

Figure 3.17: Gain ratio (a) and Pearson Correlation (b) for the loops L-P2, L-M2 and L-O4. Horizontal axes span the range of 1 to 10 and correspond to the number of shifted versions used in the matching process.

shift differences, and matching with shifted queries. The results showed in this chapters aimed to illustrate all the task done in this thesis and they don't lead to a relevant conclusion of a general peak map feature behavior in electronic music samples. However, in the following chapter we will discuss the results obtained by using a bigger data set which will guide us to meaningful results and analyses.

Chapter 4

Larger-Scale Experiments

Having described the processing pipeline in Chapter 3, we proceed to study its performance with a larger, more realistic data-set. Following the structure of the previous chapter, we start explaining our data-set. Then, we give a description of the feature extraction configuration. The results and analysis are described in the sections: loop combination, audio degradation, and time shift differences within a STFT window frame.

4.1 Loop data-set

The loop data-set consists of 111 audio samples. We collected these samples from the included data-base of *MAGIX Music Maker Premium* software [12]. Loops are organized into 6 electronic music genres: *dance*, *deep house*, *dubstep*, *hip-hop*, *techno* and *trap*. Within each genre, loops are categorized by instrumental families: bass, brass, drums, fx, guitar, keys, lead, mallet, pad, percussion, sequence, special, strings, synth, vocals, and vocal rap. All loops are stereo with a sampling frequency of $F_s = 44100\,\text{Hz}$. The length of a loop can be of 1, 2, 4 or 8 bars. The musical tempo is indicated by the BPM. Table 4.1 shows a summary of the data-set information based on genre categories. Table 4.2 shows a summary of the data-set information based on instrumental families. In Appendix A, there is a detailed description of each loop in the data-set.

4.2 Feature Extraction Configuration

Following the same steps as described in Section 3.2, for each stereo audio sample, we did a conversion from stereo to mono by means of the average of the channels. Each loop was re-sampled to the sampling frequency $F_s = 22050$. The STFT was computed using a *Hann* window of size 4096 samples and a hop size $H = 2048$ samples. The magnitude spectrogram and

	# of loops	# of instrumental families	BPM	Bar measure
Dance	10	10	130	2, 4, 8
Deep House	19	11	120	1, 2, 4
Dubstep	19	8	135	4, 8
HipHop	20	12	90	1, 2, 4
Techno	23	8	130	1, 2
Trap	20	8	75	1, 2, 4

Table 4.1: General description of the audio loop data-set organized by genres.

	# of loops	# of Genres	BPM	Bar measure
Bass	13	6	75, 90, 120, 130, 135	1, 2, 4
Brass	3	3	75, 90, 120	2, 4
Drums	22	6	75, 90, 120, 130, 135	2, 4, 8
Fx	11	6	75, 90, 120, 130, 135	2, 4
Guitar	2	2	90, 130	2, 8
Keys	4	4	90, 130, 135	2, 4
Lead	2	1	135	4, 8
Mallet	1	1	130	2
Pad	7	6	75, 90, 120, 130, 135	2, 4
Percussion	8	3	120, 130	1, 2, 4
Sequence	15	6	75, 90, 120, 130, 135	1, 2, 4
Special	3	1	120	2
Strings	3	2	90, 120	2, 4
Synth	7	5	75, 90, 120, 130, 135	1, 2, 4
Vocals	9	4	75, 90, 120, 130	2, 4, 8
Vocal Rap	1	1	90	1

Table 4.2: General description of the audio loop data-set organized by instrumental families.

log-frequency spectrogram were calculated. The peak map (PM) and the log-frequency peak map (LPM) of each audio were computed using an analysis window of 4x4 (see Section 3.2). This window size was chosen in order to obtain peaks at the beat times of the audio sample. Maximum values in the spectrogram below 1 are not chosen as peaks. In all retrieval calculations we considered a *true positive* as a peak in the query that matches to a peak in the track at the exact position, one time frame position to the left, one time frame position to the right, one frequency index position up, or one frequency index position down (see Section 3.3).

4.3 Loop Combination

In this experiment, we applied the audio sample retrieval matrix on loops with the same genre. Then, we study the behavior of features for different complex mix scenarios. The results and analysis are shown in the following sections.

4.3.1 Audio retrieval matrices

In this section, we describe the audio retrieval matrices results obtained for each EM genre in Table 4.1. Within each genre, loops can have different length, thus, we extended the loops to the maximum length found within the genre by means of repetitions. For example, if the maximum length of a loop is 8 bars, all loop within the same genre will be extended by means of repetitions until a length of 8 bars is reached.

4.3.1.1 Dance

A total of 10 audio samples (loops) were used. Each loop belongs to an instrumental family: bass, drums, fx, guitar, keys, pad, percussion, sequence, synth, and vocals. The musical tempo is 130 BPM where loops can have a duration of 2, 4 or 8 bars (3.69, 7.38, or 14.76 seconds respectively). A detailed description of these loops can be seen in Table A.1. Audio sample retrieval matrices were computed for all 10 loops (extended to 14.76 seconds) and the results are shown in Figure 4.1. Rows in Figure 4.1 represent the loops which are sorted as in Table A.1. Columns represent 27 tracks with length of 8 bars (14.76 seconds). Red points in all matrices denote the presence of a loop.

In Figure 4.1, columns 1 to 10 show that there is common information among loops. In (a), rows 1, 2, 4, and 9 correspond to the loops with the category bass, drums, guitar, and synth respectively. All loops contain in most cases around 20% of the peaks in the aforementioned 4 loops (columns 1 to 10). In (b), we can see the same were columns 1 (bass), 2 (drums), 4 (guitar), and 9 (synth) show significant values on recall (around 20%). In (c), we can see which loops share a similar amount (in query and in track) of common peaks since the F-measure gives a relation between the precision and recall (Section 2.5.2).

Columns 11 to 27 describe some specific combinations of loops (we show different combinations of 2 to 10 loops) which were randomly computed. In precision matrix (a), we can see how much information survived in a combination since we compute the ratio of matched peaks and peaks in the track. In most cases, loops 2 (drums) and 9 (synth) present large amount of peaks in the combination (around 50% of the peaks in the query), which can mean that these loops have a considerable amount of peaks, and/or the other loops in the combination have few feature components. An example of these cases is track 15, which is the combination of 4 loops: loop 1 (bass), loop 2 (drums), loop 5 (keys), and loop 8 (Sequence). Most of the peaks in track 15 matched to peaks in loop 2; these peaks (true positives) represent 63% of the peaks in track 15. On the other hand, loop 5 has the lowest relevance in track 15 since true positives represent 5% of the peaks in the track. In Figure 4.2, we show the peak map retrieval representation for the recall of track 15. In (a) the query is loop 2 (drums) and in (b) the query is loop 5 (keys).

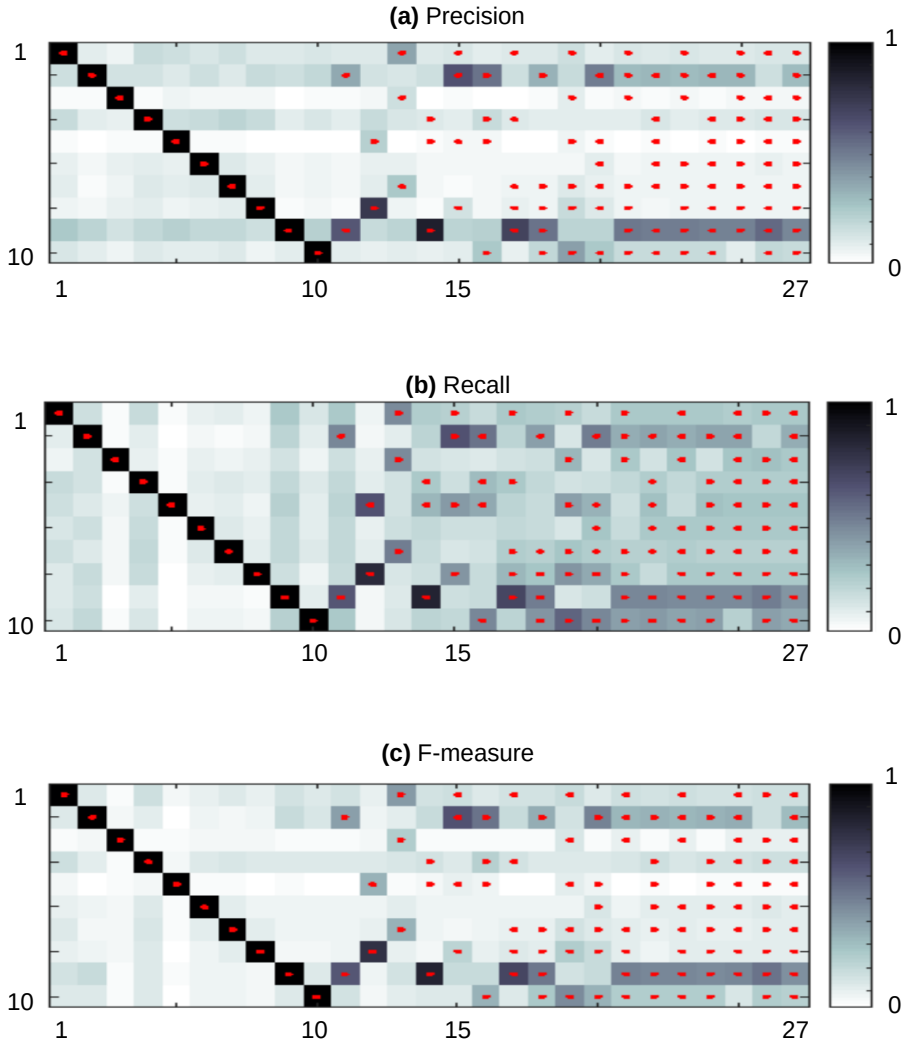

Figure 4.1: Audio sample retrieval matrices for the genre *Dance*. In all matrix representations, rows represent audio samples (loops) which are sorted (from top to bottom) by the instrumental families: bass, drums, fx, guitar, keys, pad, percussion, sequence, synth, and vocals (see Table A.1). Columns represent tracks with a duration of 8 bars (14.76 seconds). Red points indicate the presence of the loop in the track. Each cell contains the retrieval results when the corresponding loop is the query. The matrix representation in (a) describes the precision computed by the ratio of true positives to peaks in the track. The figure (b) shows the recall computed by the ratio of true positives to peaks in the loop. (c) represents the F-measure between the results in (a) and (b).

From the recall results in Figure 4.1(b), in columns 11 to 14, we can see that matched peaks of queries included in the combination (red points) represent in the track more than 35%. This peak information can decrease to 20% when the number of combined loops is increasing. This is due to the fact that new information is added to the track, thus, some peaks of the query may be distorted by the combination and they may not be relevant in the fingerprinting process. Also,

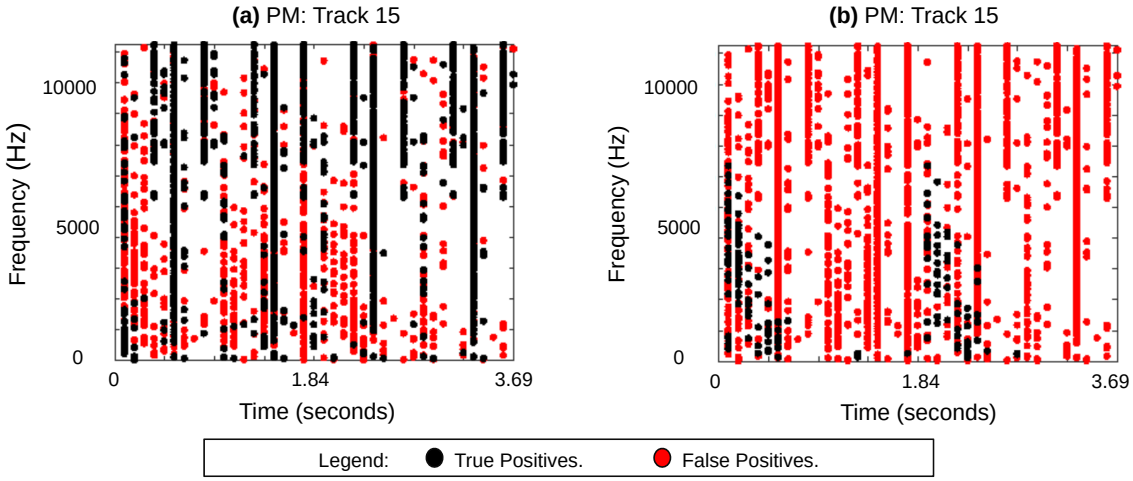

Figure 4.2: Peak map retrieval representation of track 15. The horizontal axis spans the range of 0 to 3.69 seconds (2 bars). Black points represent common peaks between the query and the audio track (true positives). Red points correspond to peaks in the audio track but not in the query (false positives). The query in (a) is loop 2 (drums) and the query in (b) is loop 5 (keys).

we can see that tracks contain information that is also present in loops which are not in the corresponding combination. The reason is that loops share information among them (columns 1 to 10) and combinations of them may help to emphasize features that can also be in loops not present in the mixture. As a result, in the case of few combination of loops (columns 11 to 14), we can see at least 5% of common peaks between a track an a loop (not included in the combination), and in the case of higher loop combinations, it increases to around 15%.

In (c), we can see that the features in loops 2 (drums) and 9 (synth) represent a large amount of peaks in both query and track. The other loops have low representative information in combination experiments since true positives (common peaks between track and query) represent a small portion of features in both track and query.

4.3.1.2 Deep House

A total of 19 audio samples were used (see Table 4.1). The instrumental families included in this genres are: bass, brass, drums, fx, pad, percussion, sequence, special, strings, synth, and vocals. The musical tempo is 120 BPM where loops can have a duration of 1, 2 or 4 bars (2, 4, or 8 seconds respectively). A detailed description of these loops can be seen in Table A.2. Audio sample retrieval matrices were computed for the 19 loops (extended to 8 seconds) and the results are shown in Figure 4.3. Rows in Figure 4.3 represent the loops which are sorted as in Table A.2. Red points in all matrices denote the presence of a loop. Columns represent 54 tracks with duration of 4 bars (8 seconds).

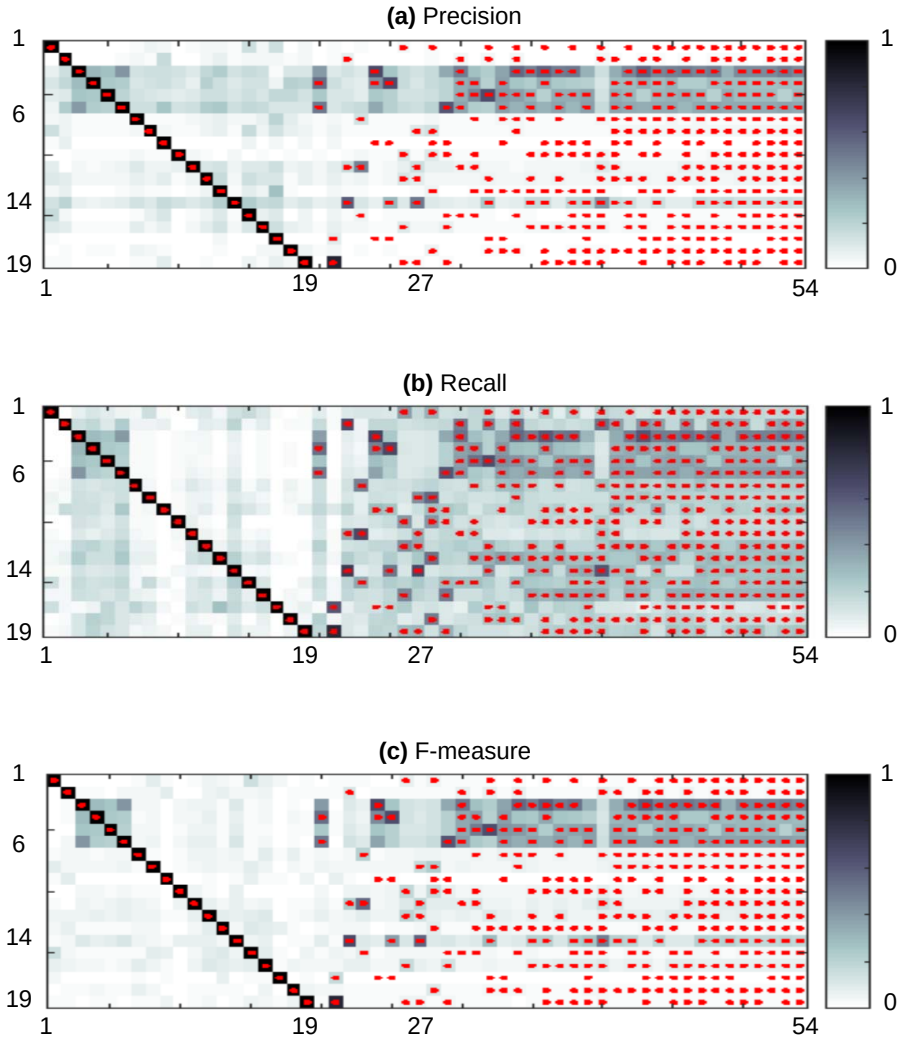

Figure 4.3: Audio sample retrieval matrices for the genre *Deep House*. In all matrix representations, rows represent audio samples (loops) which are sorted (from top to bottom) by the instrumental families: bass, brass, drums, fx, pad, percussion, sequence, special, strings, synth, and vocals (see Table A.2). Columns represent tracks with a duration of 4 bars (8 seconds). Red points indicate the presence of the loop in the track. Each cell contains the retrieval results when the corresponding loop is the query. The matrix representation in (a) describes the precision computed by the ratio of true positives to peaks in the track. The figure (b) shows the recall computed by the ratio of true positives to peaks in the loop. (c) represents the F-measure between the results in (a) and (b).

Loops 3, 4, 5, and 6 show a significant similarity since they are members of the same category drums; in columns 1 to 19, they show in precision (a), recall (b) and F-measure (c) more than 25% of representation of matched peaks. There are cases where these 4 loops are not present in the track but they yield high precision values (a). In addition, loop 3 (drums 1), 4 (drums 2), 5 (drums 3), 6 (drums 4), and 14 (special 2) contain significant information that is also present in

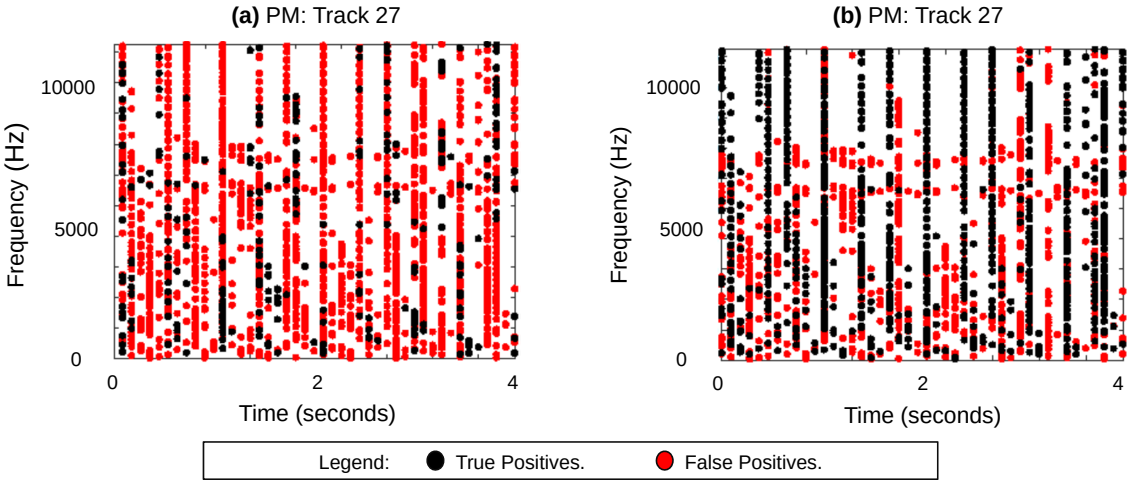

Figure 4.4: Peak map retrieval representation of track 27. The horizontal axis spans the range of 0 to 4 seconds (2 bars). Black points represent common peaks between the query and the audio track (true positives). Red points correspond to peaks in the audio track but not in the query (false positives). The query in (a) is loop 3 (drums 1) and the query in (b) is loop 14 (special 2).

other loops; we can see clear regions in precision (horizontal regions) and recall (vertical regions).

In the combination examples of Figure 4.3 (columns 20 to 57), loops 3 (drums 1), 4 (drums 2), 5 (drums 3), 6 (drums 4), and 14 (special 2) are audio samples that represent a large portion of peaks in tracks. These 5 loops show high values in the F-measure (c) since more than 10% of matched peaks is in both query and track. The other loops have low precision values which indicate that matched peaks represent a small amount of peaks in the track (less than 6%), thus the F-measure is low. In addition, we can see high values in the recall (b) which means that matched peaks represent at least 20% of peaks in the query. In cases were the query is not present in the combination, we can see that tracks match more than 15% of peaks in the query, e.g., track 27. In Figure 4.4, we show the peak map retrieval representation of track 27 where the query in (a) is loop 3 (drums 1) which is not present in the combination, however, it shows a significant amount of matched peaks (21%). The query in (b) is loop 14 (special 2) which is present in the mixture (62%).

4.3.1.3 Dubstep

A total of 19 audio samples were used (see Table 4.1). The instrumental families included in this genres are: bass, drums, fx, keys, lead, pad, sequence, and synth. Loops can have a length of 4 or 8 bars (7.11 or 14.22 seconds) and they have a musical tempo of 135 BPM. A detailed description of these loops can be seen in Table A.3. In the audio sample retrieval matrices shown in Figure 4.5, all 19 loops (extended to 14.22 seconds) were used. Rows in Figure 4.5 represents

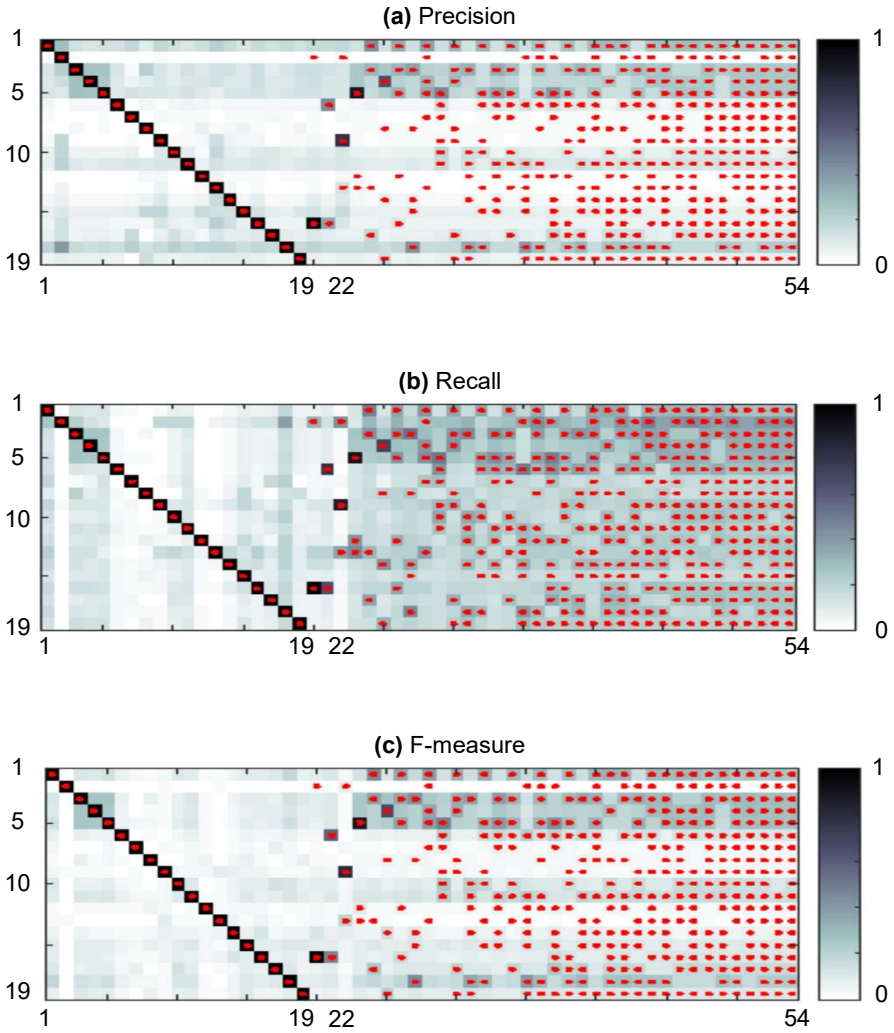

Figure 4.5: Audio sample retrieval matrices for the genre *Dubstep*. In all matrix representation, rows represent audio samples which are sorted (from top to bottom) by the instrumental families: bass, drums, fx, keys, lead, pad, sequence, and synth (see Table A.3). Columns represent tracks with a duration of 8 bars (14.22 seconds). Red points indicate the presence of the loop in the track. Each cell contains the retrieval results when the corresponding loop is the query. The matrix representation in (a) describes the precision computed by the ratio of true positives to peaks in the track. The figure (b) shows the recall computed by the ratio of true positives to peaks in the loop. (c) represents the F-measure between the results in (a) and (b).

the loops which are organized as described in Table A.3. Red points in all matrices denote the presence of a loop. Columns represent 54 tracks with duration of 8 bars (14.22 seconds).

In Figure 4.5, columns 1 to 19 show horizontal regions in (a) and vertical regions in (b) which indicate that there are loops containing relevant information included also in other loops. These regions correspond to loops: 1 (bass 1), 3 (drums 1) , 4 (drums 2), 5 (drums 3), 10 (lead 1), 11

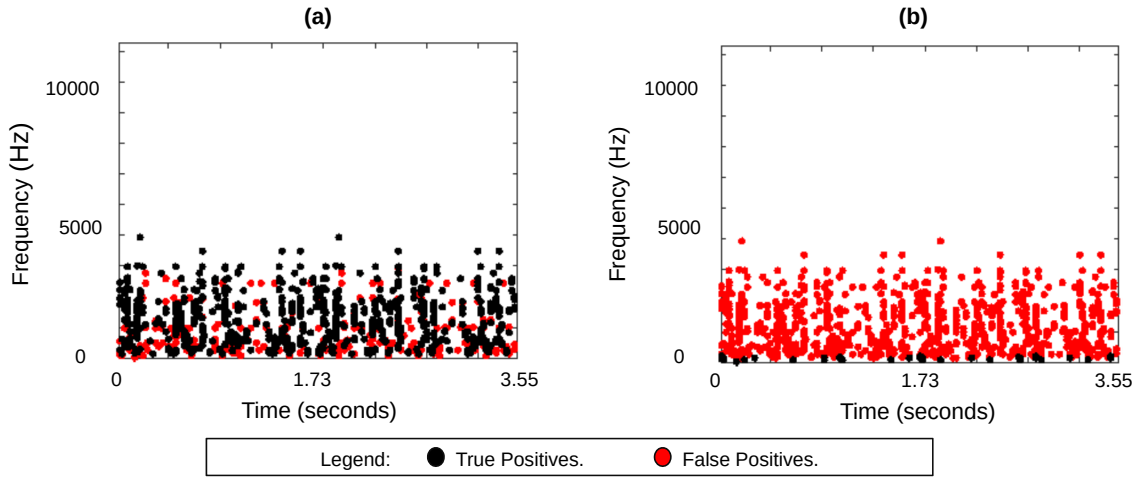

Figure 4.6: Peak map retrieval representation of track 22. The horizontal axis spans the range of 0 to 3.55 seconds (2 bars). Black points represent common peaks between the query and the audio track (true positives). Red points correspond to peaks in the audio track but not in the query (false positives). The query in (a) is loop 9 (keys) and the query in (b) is loop 2 (bass 2).

(lead 2), 15 (sequence 2), and 18 (synth 2). At least these loops contain 15% of peaks that are also in other loops.

In the combination examples (columns 20 to 54), the precision (a) shows that the peaks of loops 1, 3, 4, 5, 10, 11, 15 and 18 represent more than 15% of the peaks in tracks; they are a representative portion of peaks in tracks. In the case of the other loops, matched peaks represent in most cases less that 12% of the peaks in the query. The recall (c) shows that matched peaks represent more than 20% of the peaks in the query, even if the query is not present in the corresponding combination. From the f-measure (c), we can see that features of loops 1 (bass 1), 3 (drums 1) , 4 (drums 2), 5 (drums 3), 10 (lead 1), 11 (lead 2), 15 (sequence 2), and 18 (synth 2) are loops which contain a relevant amount of peaks since the precision and recall are high. On the other hand, we can also see loops which do not have a representative amount of peaks and they may be easily distorted. For example, track 22 is a combination of 3 loops: loop 2 (bass 2), loop 9 (keys), and loop 13 (pad 2). Loop 2 presents the lowest F-measure because of its irrelevancy with respect of peaks in track 22; few peaks in track 22 matched to loop 2 (low precision), however, matched peaks represent 60% of peaks in the loop. On the contrary, loop 9 is the relevant loop in track 22 since most of the peaks in the audio recording matched the loop. In Figure 4.6, it is shown the peak map retrieval representation of track 22. The query in (a) is the loop 9 (keys) and the query in (b) is the loop 2 (bass 2).

4.3.1.4 Hip-Hop

A total of 20 audio samples were used (see Table 4.1). The instrumental families included in this genres are: bass, brass, drums, fx, guitar, keys, pad, sequence, strings, synth, vocals and vocal raps. Loops can have a length of 1, 2, or 4 bars (2.66, 5.32, or 10.64 seconds) and and they have a musical tempo of 90 BPM. A detailed description of these loops can be seen in Table A.4. In the audio sample retrieval matrices shown in Figure 4.7, all 20 loops (extended to 10.64 seconds) were used. Rows in Figure 4.7 represents the loops which are organized as described in Table A.4. Red points in all matrices denote the presence of a loop. Columns represent 54 tracks with duration of 4 bars (10.64 seconds).

The precision results (a) show that there are loops that represent less than 6% of the peaks in combination examples, e.g, loop 1 to 3 (bass), 4 (brass), 15 (sequence), etc. In columns 20 to 57, the recall results (b) show that matched peaks represent in most cases more than 20% in the query, even if it is not present in the track. Peaks from the category drums are more relevant in the track than others. Also, in columns 1 to 20, loops 5 to 13 (drums, fx, guitar, and keys) show in most cases more than 13% of peaks which are also in other loops. In the F-measure (columns 1 to 20), loops 1 to 3 share information since belongs to the category bass. Also, loops 5 to 9 have common information because they are members of the category drums. In addition, matched peaks in drums queries and loop 13 (keys) represent in most cases more than 17% in both query and tracks. As a visual example of a loop combination, Figure 4.8 shows an peak map retrieval representation of track 25 where the query in (a) is loop 14 (pad) with a precision of 0.09 and recall 0.25. The query in (b) is loop 19 (vocals) with a precision of 0.23 and a recall of 0.51.

4.3.1.5 Techno

A total of 23 audio samples were used (see Table 4.1). The instrumental families included in this genres are: bass, drums, fx, keys, mallet, pad, percussion, and sequence. Loops can have a length of 1, 2, or 4 bars (1.84, 3.68, and 7.36 seconds) and they have a musical tempo of 130 BPM. A detailed description of these loops can be seen in Table A.5. The audio sample retrieval matrices in Figure 4.9 were computed for the 23 loops (extended to 7.36 seconds). Rows in Figure 4.9 represents the loops which are organized as described in Table A.5. Red points in all matrices denote the presence of a loop. Columns represent 54 tracks with duration of 4 bars (7.36 seconds).

From these matrices, we can see similarity between loops which in most cases is because they belong to the same instrumental family. Loops 1 to 3 are members of the category bass, loops 4 to 8 belong to the category drums, loops 13 to 16 belongs to the category percussion, and loops 17 to 23 are members of the category sequence (see precision, recall). In the categories drums, percussion, and sequence, the matched peaks represent a significant amount of information in

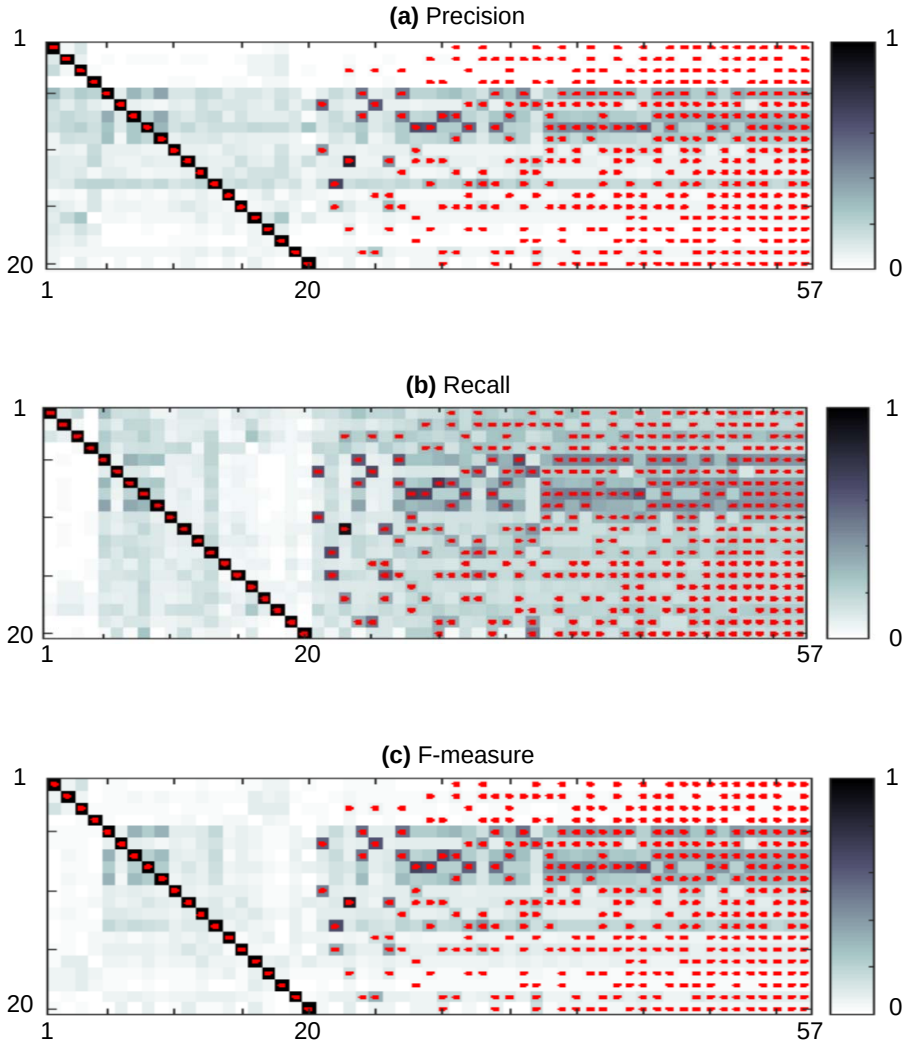

Figure 4.7: Audio sample retrieval matrices for the genre *Hip-Hop*. In all matrix representations, rows represent audio samples which are sorted (from top to bottom) by the instrumental families: bass, brass, drums, fx, guitar, keys, pad, sequence, strings, synth, vocals and vocal raps (see Table A.4). Columns represent tracks with a duration of 4 bars (10.64 seconds). Red points indicate the presence of the loop in the track. Each cell contains the retrieval results when the corresponding loop is the query. The matrix representation in (a) describes the precision computed by the ratio of true positives to peaks in the track. The figure (b) shows the recall computed by the ratio of true positives to peaks in the loop. (c) represents the F-measure between the results in (a) and (b).

both the track (precision) and the query (recall); they represent more than 35%, 20%, 15% and respectively (see F-measure). One example of a loop combination within this genre is the track 25 which is a mix of loop 11 (mallet) and loop 15 (percussion 3). The query in (a) is loop 11 with a precision 0.24 of and recall of 0.53. The query in (b) is loop 15 with a precision 0.37 of and recall of 0.42.

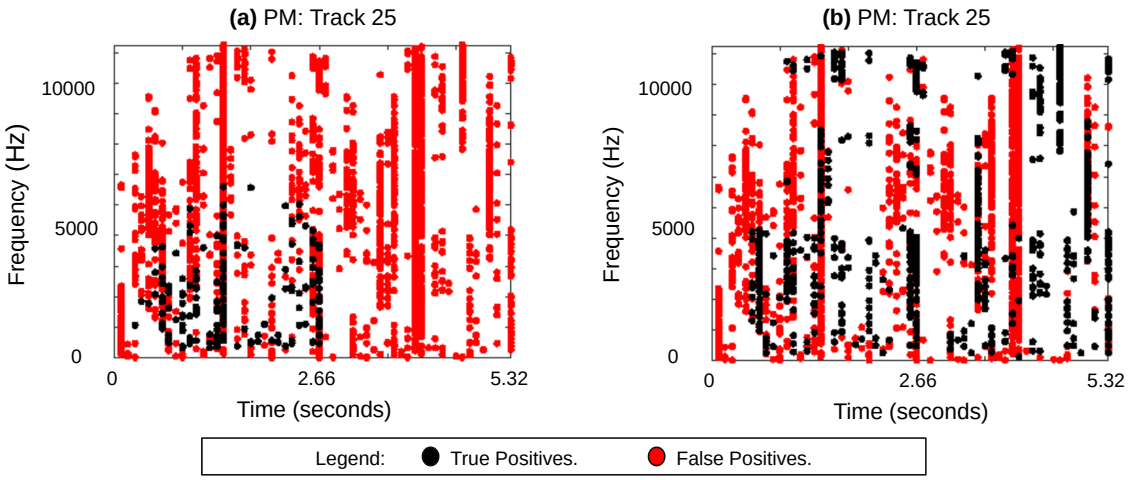

Figure 4.8: Peak map retrieval representation of track 25. The horizontal axis spans the range of 0 to 5.32 seconds (2 bars). Black points represent common peaks between the query and the audio track (true positives). Red points correspond to peaks in the audio track but not in the query (false positives). The query in (a) is loop 19 (vocals) and the query in (b) is loop 14 (pad).

4.3.1.6 Trap

As we can see in Table 4.1, a total of 20 audio samples were used. The instrumental families included in this genres are: bass, brass, drums, fx, pad, sequence, synth and vocals. Loops can have a length of 1, 2, or 4 bars (3.2, 6.4, and 12.8 seconds) and they have a musical tempo of 75 BPM. A detailed description of these loops can be seen in Table A.6. The audio sample retrieval matrices in Figure 4.11 were computed for the 20 loops (extended to 12.8 seconds). Rows in Figure 4.11 represents the loops which are organized as described in Table A.6. Red points in all matrices denote the presence of a loop. Columns represent 54 tracks with duration of 12.8 seconds.

From the F-measure (c), we can see that loops 5 to 8 share a similar amount of information since they belong to the category drums. There is common information among loops with different categories; loops 13 to 20 have similarities in features and they are organized in 3 different categories: sequence, synth, and vocals. Loops 5 to 8 (drums), 11 (fx), and loops 13 to 20 (sequence, synth, and vocals) show common information among loops with more than 12% of matched peaks (see recall and precision). All these mentioned loops have in most of combination a significant amount of matched peaks in both track and query. From the recall (b), we can see that matched peaks represent at leas 15% of the peaks in query, even if the query is not present in the corresponding combination. There are loops with a significant amount of matched peaks in the query (around 40% recall) but they are not relevant in the track (low precision). Thus, their F-measure decrease, e.g., track 23. Figure 4.12 shows the peak map retrieval representation of track 23 which is the combination of loop 2 (bass 2), 12 (pad), and 13 (sequence). In this case,

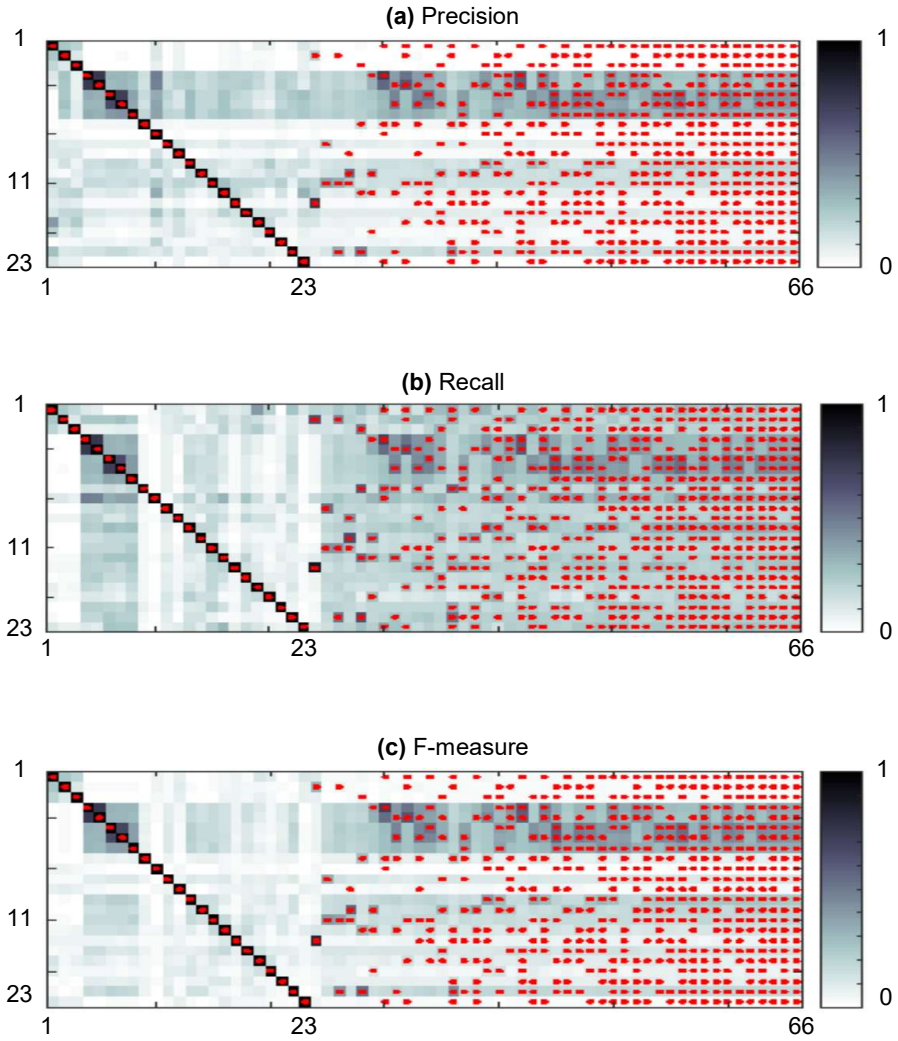

Figure 4.9: Audio sample retrieval matrices for the genre *Techno*. In all matrix representations, rows represent audio samples which are sorted (from top to bottom) by the instrumental families: bass, drums, fx, keys, mallet, pad, percussion, and sequence (see Table A.5). Columns represent tracks with a duration of 4 bars (7.36 seconds). Red points indicate the presence of the loop in the track. Each cell contains the retrieval results when the corresponding loop is the query. The matrix representation in (a) describes the precision computed by the ratio of true positives to peaks in the track. The figure (b) shows the recall computed by the ratio of true positives to peaks in the loop. (c) represents the F-measure between the results in (a) and (b).

loop 2 (bass) has a precision of 0.02 and a recall of 0.39 while loop 12 (pad) has a precision of 0.12 and a recall of 0.45.

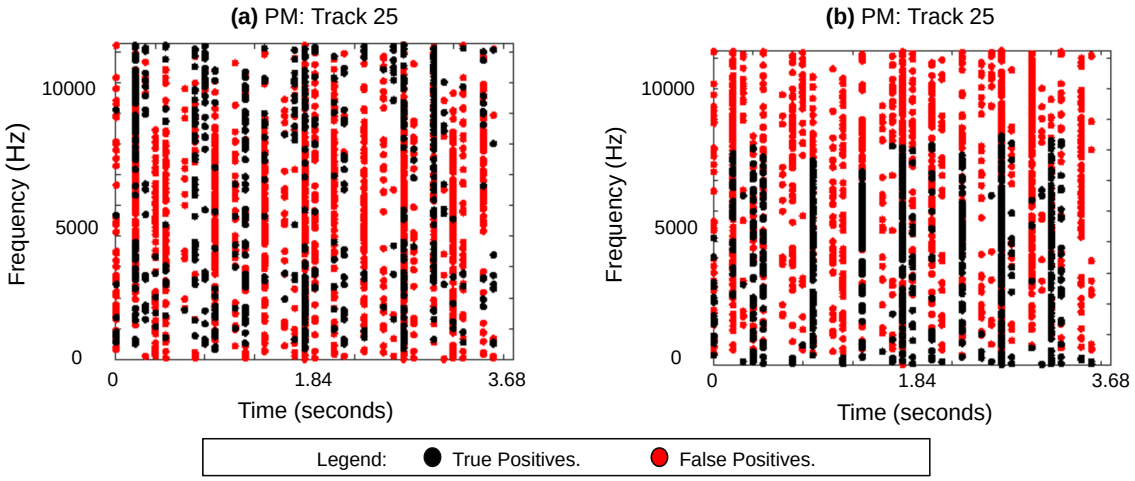

Figure 4.10: Peak map retrieval representation of track 25. The horizontal axis spans the range of 0 to 3.68 seconds (2 bars). Black points represent common peaks between the query and the audio track (true positives). Red points correspond to peaks in the audio track but not in the query (false positives). The query in (a) is loop 11 (mallet) and the query in (b) is loop 15 (percussion 3).

4.3.2 Complex mixtures

In this experiment, we computed the retrieval information (precision, recall, and F-measure) of tracks with different complexity level. As said in Section 3.3, the complexity level increases when a loop is added (including the query). We randomly chose 20 different combinations of 9 loops with same genre as the query. For each combination we proceed to construct tracks following the procedure described in Figure 3.7; for each complexity greater than 1 we compute 2220 tracks.

Figures 4.13, 4.14, and 4.15 show the retrieval information results by means of boxplots. Boxes indicate the interquartile range where the red line inside each box is the median. Magnitude spectrogram peak maps (PM) were used in (a) whilst log-frequency peak maps (LPM) were used in (b). The dashed horizontal line in both (a) and (b) denote the base-line computed by the average of retrieval information results when the query is white noise.

In Figure 4.13, we can see a decreasing behavior of the precision as the complexity increases. This decrease starts rapidly when a second loop is added; the median in the precision is 0.42 (PM) and 0.49 (LPM) for complexity 2. The decrease is slower for a complexity higher than 5 in both LPM and PM. There are particular cases that show a lower decrease (red crosses above boxes) that are related to loop queries with a large amount of features and they can easily be relevant in a track combination, e.g., drums in the techno genre. On the other hand, there are cases in which the precision goes below the base-line (horizontal dashed line). These cases are related to loop with a small amount of peaks, e.g. loop 2 (bass 2) in dubstep genre.

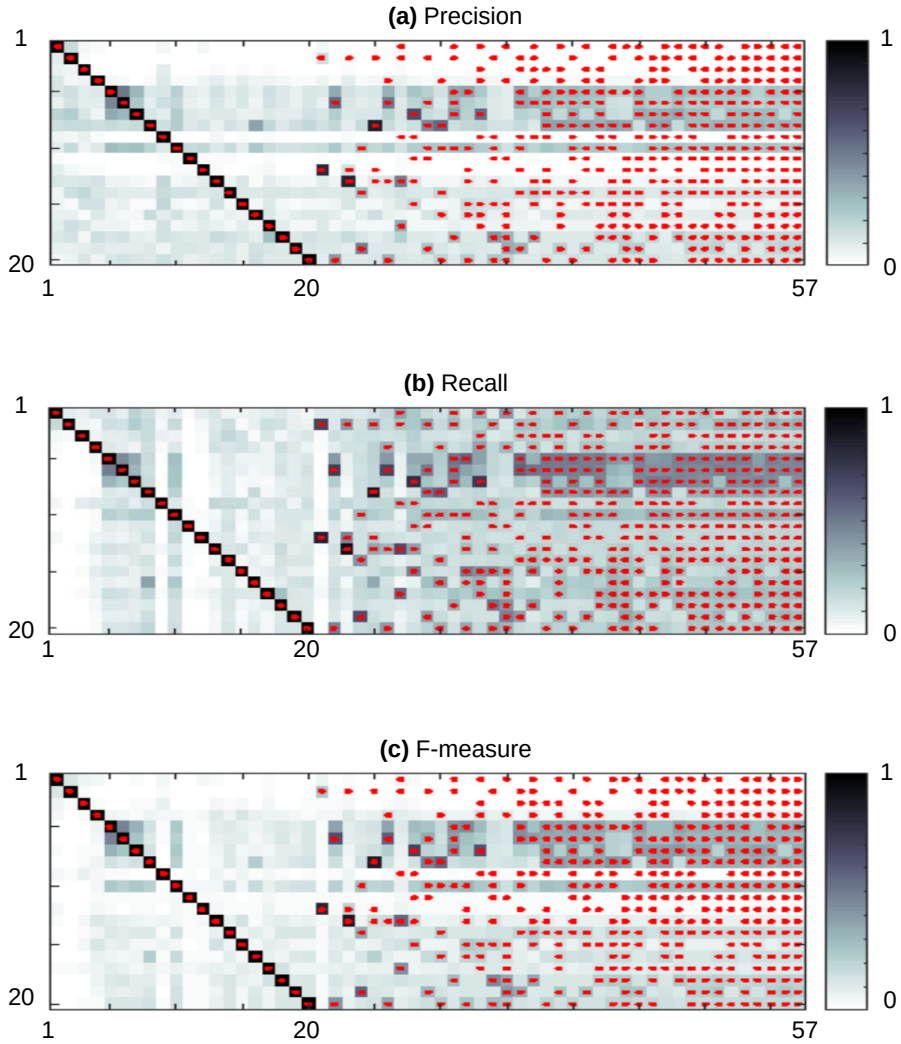

Figure 4.11: Audio sample retrieval matrices for the genre *Trap*. In all matrix representation, rows represent audio samples which are sorted (from top to bottom) by the instrumental families: bass, brass, drums, fx, pad, sequence, synth, and vocals (seeTable A.6). Columns represent tracks with a duration of 4 bars (12.8 seconds). Red points indicate the presence of the loop in the track. Each cell contains the retrieval results when the corresponding loop is the query. The matrix representation in (a) describes the precision computed by the ratio of true positives to peaks in the track. The figure (b) shows the recall computed by the ratio of true positives to peaks in the loop. (c) represents the F-measure between the results in (a) and (b).

The recall is shown in Figure 4.14. These results indicate that less peaks in the query are matched to tracks with higher complexity. The recall decreases from a median of 0.69 to 0.22 for complexity 2 to 6 in the case of PMs (a). When LPMs are used (b), the recall decreases from a median of 0.63 to 0.19 for complexity 2 to 6. Median values of 0.17 (for PMs) and 0.19 (for LPMs) are reached for complexity 10 meaning that in most cases around 17% of peaks in the query are matched. The red crosses are related to loops with a small amount of peaks where a

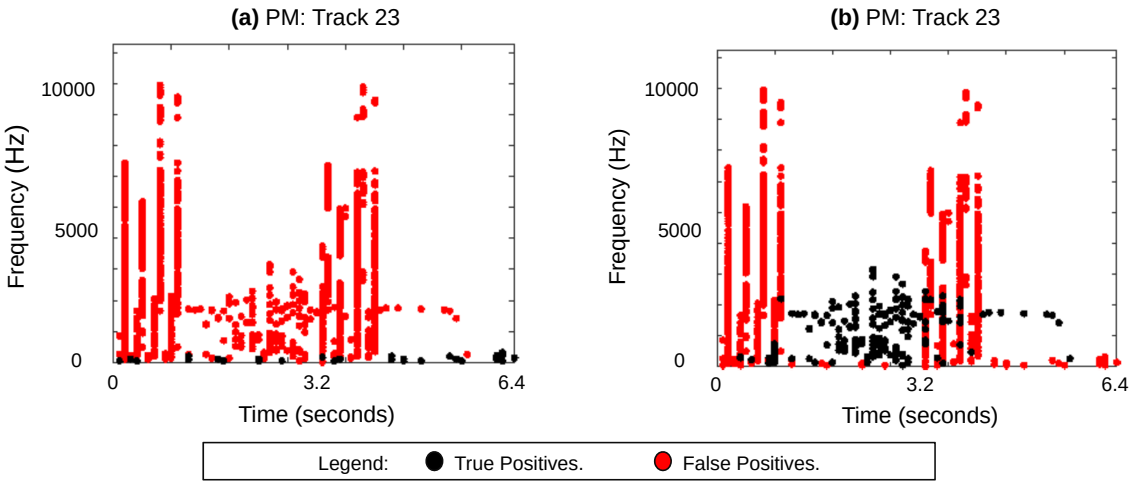

Figure 4.12: Peak map retrieval representation of track 23. The horizontal axis spans the range of 0 to 6.4 seconds (2 bars). Black points represent common peaks between the query and the audio track (true positives). Red points correspond to peaks in the audio track but not in the query (false positives). The query in (a) is loop 2 (bass) and the query in (b) is loop 12 (sequence).

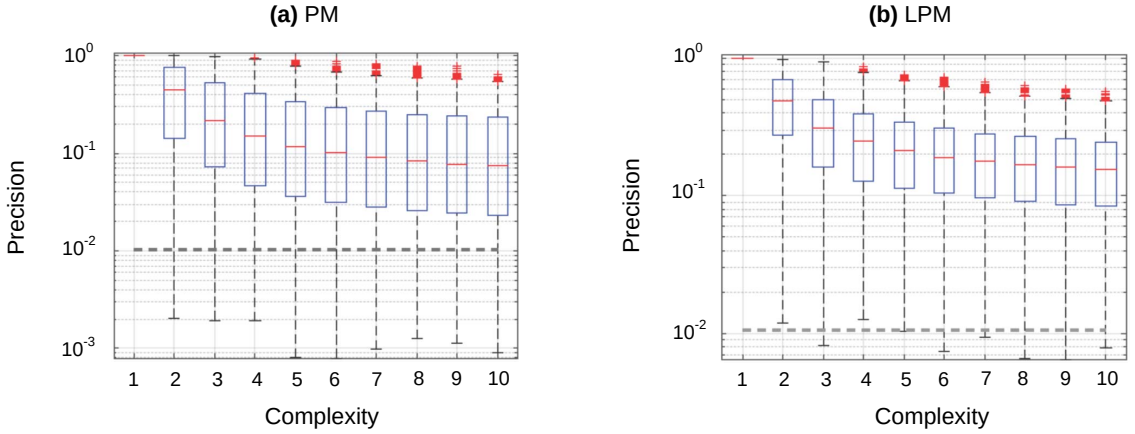

Figure 4.13: Precision with respect to different complexity. Horizontal axes represent the complexity of the track. The term complexity states the number of loops superimposed in the track. Vertical axes represent the F-measure in a logarithmic scale. (a) shows the results using magnitude square peak maps (PM). (b) shows the results using log-frequency peak maps (LPM).

few distortions produced by combinations can severely affect the recognition of these kind of loops.

Figure 4.14 shows that the F-measure in both (a) and (b) has a similar behavior. The results decrease with the increase of complexity. Log- frequency peak maps (b) show a better performance when they are compared with magnitude spectrogram peak maps (a). Matched peaks of LPMs represent a big portion in both track and query. In addition, LPMs present values above the base-line denoting that recognition can be based on a significant amount of matched peaks in

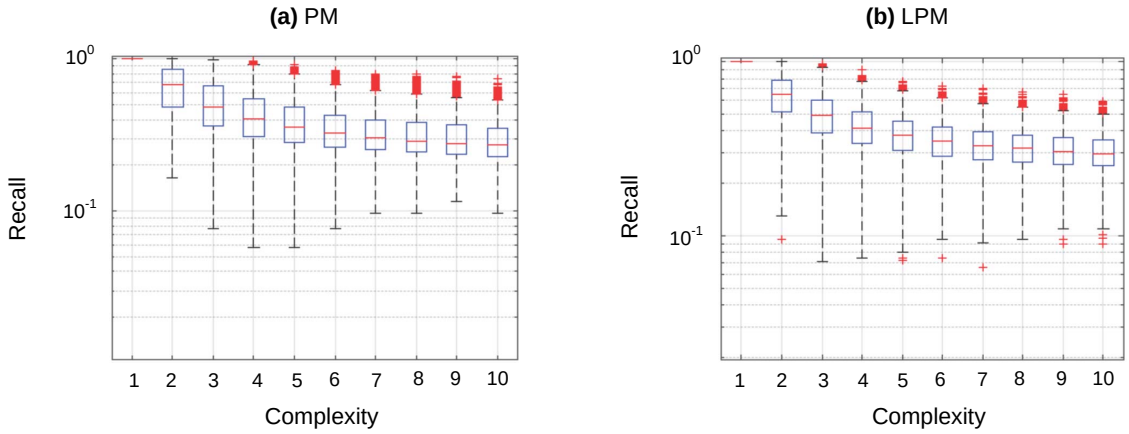

Figure 4.14: Recall with respect to different complexity. Horizontal axes represent the complexity of the track. The term complexity states the number of loops superimposed in the track. Vertical axes represent the F-measure in a logarithmic scale. (a) shows the results using magnitude square peak maps (PM). (b) shows the results using log-frequency peak maps (LPM).

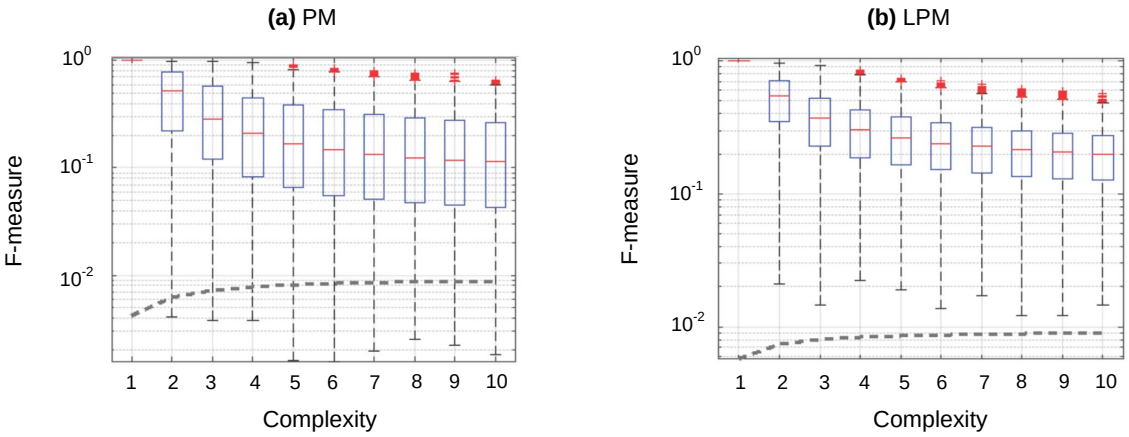

Figure 4.15: F-measure with respect to different complexity. Horizontal axes represent the complexity of the track. The term complexity states the number of loops superimposed in the track. Vertical axes represent the F-measure in a logarithmic scale. (a) shows the results using magnitude square peak maps (PM). (b) shows the results using log-frequency peak maps (LPM).

both query and track. For example, if matched peaks are in a noisy environment (precision below base-line) the query can be recognizable.

4.4 Audio Degradation

In the following sections we describe the results for the scenarios of adding external sounds.

4.4.1 Adding External Sound

As we discussed in Section 3.4.1, we constructed tracks with an external sound added to a loop with a specific SNR. These sounds are: white noise, pub environment, and vinyl noise. Precision, recall and F-measure were computed for each case and the results are discussed below with respect to SNR.

4.4.1.1 White Noise

Figures 4.16, 4.18, and 4.20 show the retrieval information precision, recall, F-measure respectively. Magnitude spectrogram peak maps (PM) were used in (a) whilst log-frequency peak maps (LPM) were used in (b).

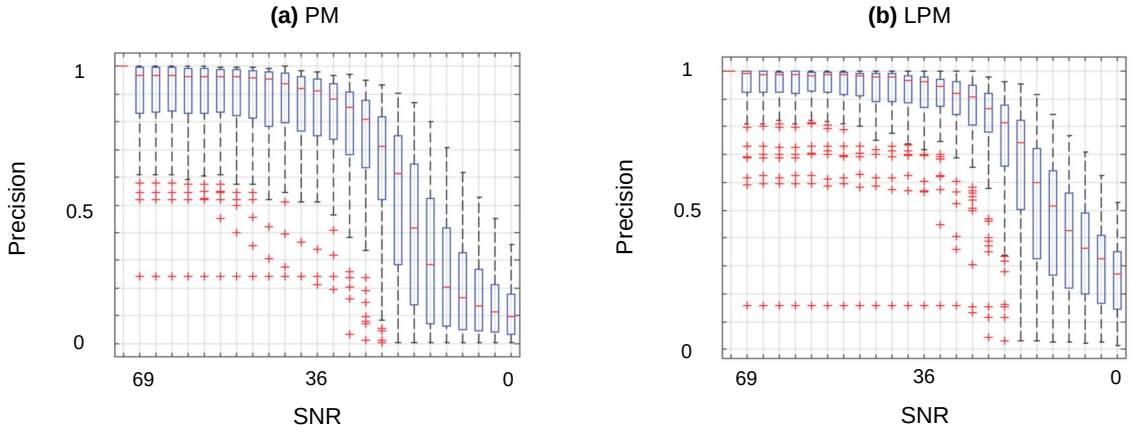

Figure 4.16: Precision with respect of SNR (dB). White noise was added to the query as described in Figure 3.9. Horizontal axes span the SNR range of 69 dB to 0 dB. The first value (on the left side) is always one, because the track corresponds to the replica of the query. Boxes indicate the intequartiles where the red line inside each box is the median. In (a), the magnitude spectrogram peak map (PM) was used and in (b) the log-frequency peak map (LPM) was used.

The precision in Figure 4.16 shows that once white noise is added to a loop (form SNR 69 dB to 0 dB), there are noisy peaks in the track which decrease the ratio of matched peaks (true positives) and peaks in the audio recording. In the cases of adding external sound, we have to consider the effects of amplitude limitations in the peak map selection process (see Section 2.4.1). As we said, in the feature configuration section (Section 4.2), we applied an exponential decay in order to obtain peaks at the moment of appearance, and in addition, maximum values in the spectrogram which are below to 1 are not chosen so that irrelevant information is not considered. If we eliminate these amplitude limitations, the precision results may be lower (there are more noisy peaks in the track) and the decrease may start with higher values of SNR (peaks with low amplitude are chosen). For PMs (a), the decrease starts with a SNR of 45 dB with a median 0.96 and continues to SNR of dB with a median of 0.097. For LPMs (b), the decrease starts

with 39 dB with a median of 0.97 and continues to SNR with a median of 0.37. The decrease is stronger for PMs (a) than LPMs (b). Figure 4.17 shows an example of the white noise effect for the case of $SNR = 40\,\text{dB}$ (a) and $SNR = 20\,\text{dB}$. Noisy peaks in the track (false negatives) increase when they are compared to the matched peaks (true positives).

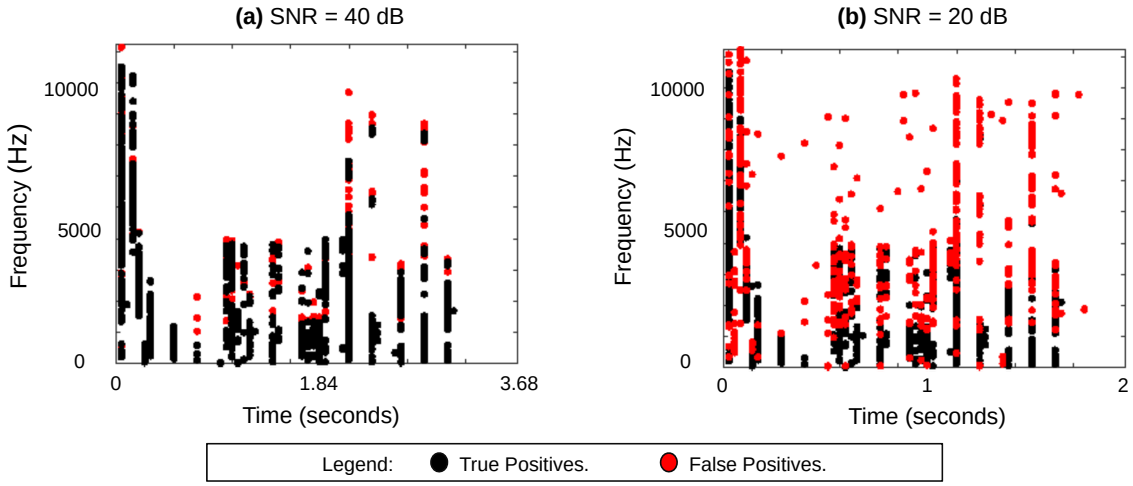

Figure 4.17: Retrieval representation of a track produced by a loop (vocals 2 of the genre trap, see Table A.6) and white noise. White noise was added to the query as described in Figure 3.9. Horizontal axes span the duration range of 0 to 3.68 seconds (2 bars). Black points denote the matched peaks (true positives). Red points indicate peaks in the track but not in the query (false positives).

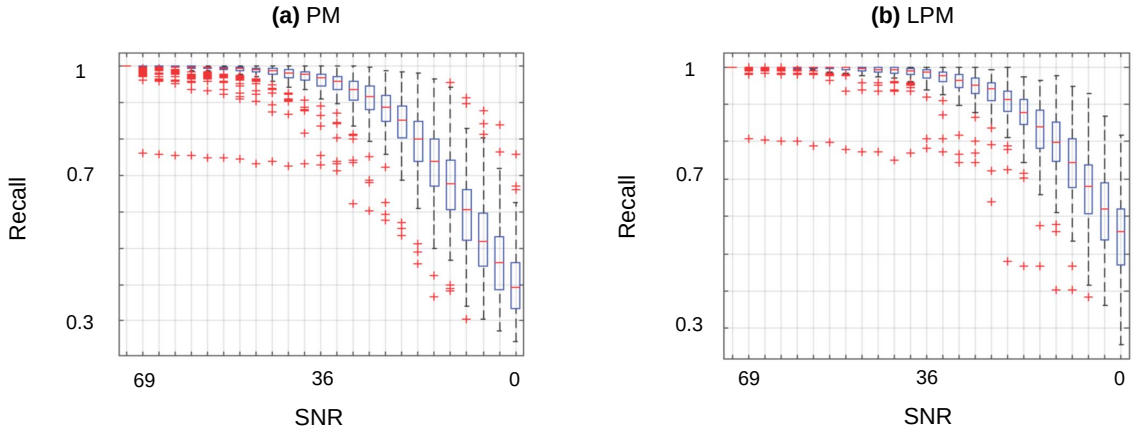

Figure 4.18: Recall with respect of SNR (dB). White noise was added to the query as described in Figure 3.9. Horizontal axes span the SNR range of 69 dB to 0 dB. The first value (on the left side) is always one, because the track corresponds to the replica of the query. Boxes indicate the intequartiles where the red line inside each box is the median. In (a), the magnitude spectrogram peak map (PM) was used and in (b) the log-frequency peak map (LPM) was used.

The recall shown in Figure 4.18 indicates how much information survived the noisy recording. As we can see, there are loops whose features are vulnerable to noisy scenarios (red crosses below

boxes) and there are also loops with resistant peaks (crosses above boxes). However, in most loops, the decrease of matched peaks in the query (true positives) starts from SNR of 45 dB with a median of 0.98 for peak maps (a) and from SNR of 33 dB with a median of 0.984 for log-frequency spectrogram (b). When SNR is equal to 0 dB, the recall has a median of 0.39 in (a) and 0.56 in (b). In this case, the recall is above 0.24 in both (a) and (b), which means that at least 25% of the peak in the query survived the noisy environment. In Figure 4.19 we can see an example of the decrease in matched peaks. Loop 5 of the genre dance is the query and presents a decrease of matched peaks (true positives) when the SNR goes from 40 dB (a) to 20 dB (b).

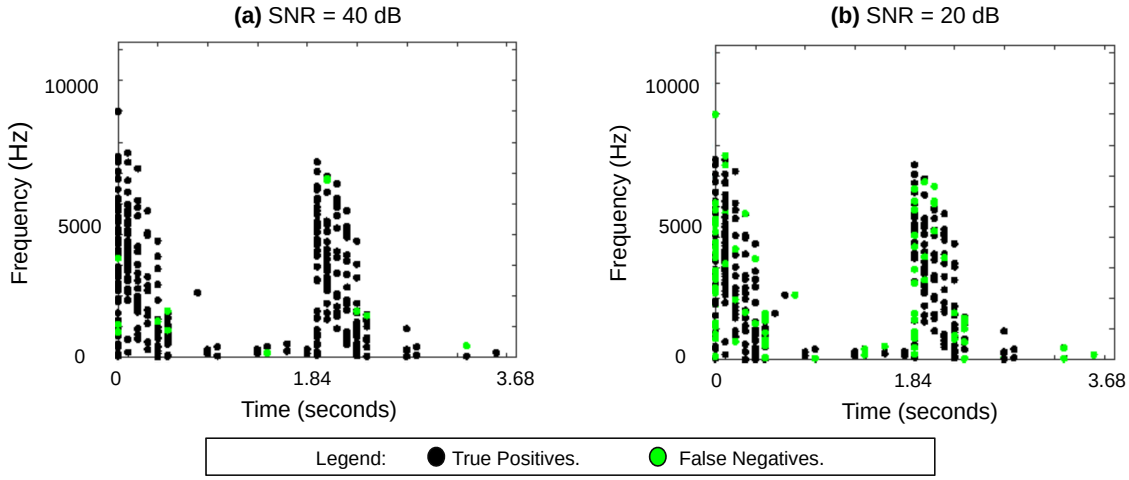

Figure 4.19: Retrieval representation of a loop query (keys of the genre dance, see Table A.1). The track consist of white noise added to the query as described in Figure 3.9. Horizontal axes span the duration range of 0 to 3.68 seconds (2 bars). Black points denote the matched peaks (true positives). Green points indicate peaks in the query but not in the track (false negatives).

The F-measure in Figure 4.20 gives a notion of how the noise affects the features in both track and query. There are 8 particular loops which are vulnerable to white noise environments since they presented F-measures below the minimum of typical values in high SNR values (red crosses). They can be easily distorted (low recall) and/or they can have a small amount of representative features (low precision) . PMs have a stronger decrease in matched peaks than LPMs, this is due to the fact that LPMs emphasize features in a logarithmic scale, thus, noisy peaks are decreased and the distortion is less powerful.

4.4.1.2 Pub environment

Figures 4.21, 4.22, and 4.23 show the retrieval information precision, recall, F-measure respectively. Magnitude spectrogram peak maps (PM) were used in (a) whilst log-frequency peak maps (LPM) were used in (b).

In Figure 4.21, we can see a similar decreasing behavior as in the previous case. The precision

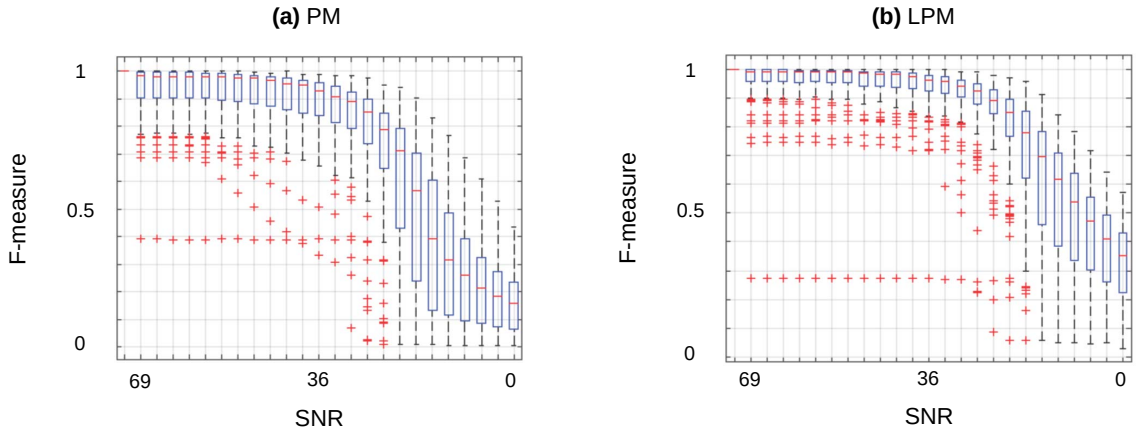

Figure 4.20: F-measure with respect of SNR (dB). White noise was added to the query as described in Figure 3.9. Horizontal axes span the SNR range of 69 dB to 0 dB. The first value (on the left side) is always one, because the track corresponds to the replica of the query. Boxes indicate the intequartiles where the red line inside each box is the median. In (a), the magnitude spectrogram peak map (PM) was used and in (b) the log-frequency peak map (LPM) was used.

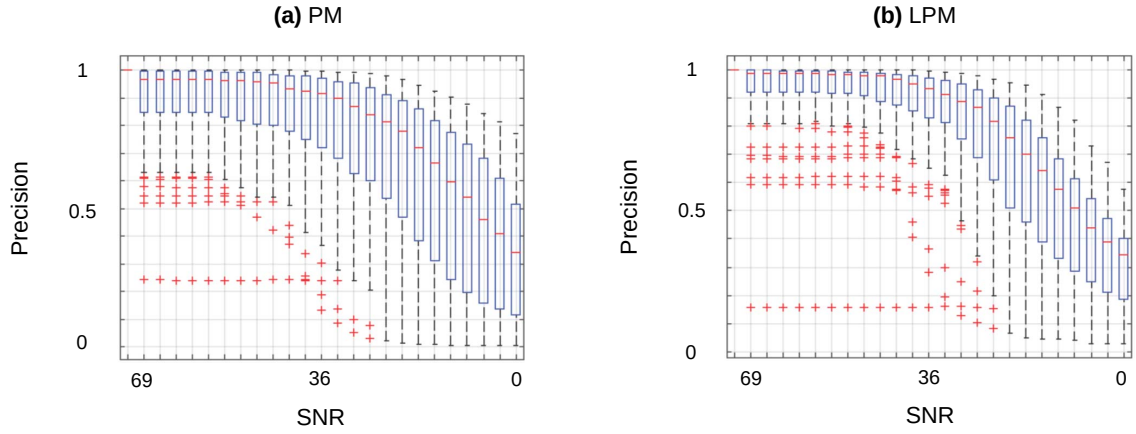

Figure 4.21: Precision with respect of SNR (dB). The audio sound *restaurant08.wav* (included in the audio degradation tool box [13]) was added to the query as described in Figure 3.9. Horizontal axes span the SNR range of 69 dB to 0 dB. The first value (on the left side) is always one, because the track corresponds to the replica of the query. Boxes indicate the intequartiles where the red line inside each box is the median. In (a), the magnitude spectrogram peak map (PM) was used and in (b) the log-frequency peak map (LPM) was used.

results show that the information added by the noise represents a large amount of peaks in the track. Once the SNR decreases, the region of possible values expands considerably in both cases (a) and (b). This is due to the fact that this external sound has strong (high amplitude) peaks even with high SNR, which decreases the precision. In other words, loops that have a small amount of peaks will obtain a low precision value; in Figure 4.21, they correspond to red crosses below boxes.

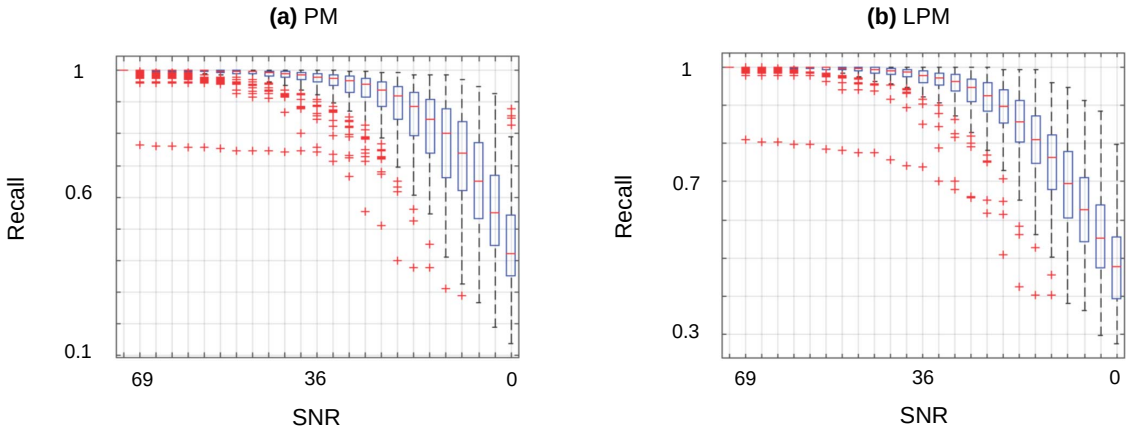

Figure 4.22: Recall with respect of SNR (dB). The audio sound *restaurant08.wav* (included in the audio degradation tool box [13]) was added to the query as described in Figure 3.9. Horizontal axes span the SNR range of 69 dB to 0 dB. The first value (on the left side) is always one, because the track corresponds to the replica of the query. Boxes indicate the intequartiles where the red line inside each box is the median. In (a), the magnitude spectrogram peak map (PM) was used and in (b) the log-frequency peak map (LPM) was used.

In the case of recall (Figure 4.22), in both (a) and (b), loops are resistant to the combination up to SNR of 41 dB. For SNR of 0 dB, the recall in (a) shows a median of 0.43 and a range of 0.35 to 0.55 for the interquartiles. The recall in (b) shows a median of 0.48% and range of 0.39 to 0.56 for the interquartiles. In this experiment, loops can be recognized under this sound combination with at least 13% of the peaks in the query. Loops which are easily distorted in the pub environment sound can be seen as red crosses in high SNR (low intensity of the external sound).

The F-measure in Figure 4.23 starts to decrease when SNR is 51 dB for PMs (a) and 46 dB for LPMs (b). In both cases (a) and (b), from SNR 18 dB to 0 dB, the typical values expand a wide range which decrease toward lower values with minima cases nearly close to zero. When SNR is 0 dB, the median is 0.35 for PMs and 0.38 for LPMs. LPMs have a better performance since the median is higher and the boxes are smaller than in the case of PM. This is due to the fact that logarithmic scale in LPMs emphasize components and compact certain frequency bands (related to the human auditory model). This property helps to compact noisy peaks which decrease precision (less noisy peaks in the track) and increase recall (less distortion). Figure 4.24 shows the retrieval representation of a track produced by the loop 12 (guitar of the genre hip-hop) and the external sound with a SNR of 21 dB. As we can see, the peaks from the log-frequency spectrogram are more spread along low frequency components and the noisy peaks represent a smaller amount of peaks in the track when they are compared to peaks in the PM (a).

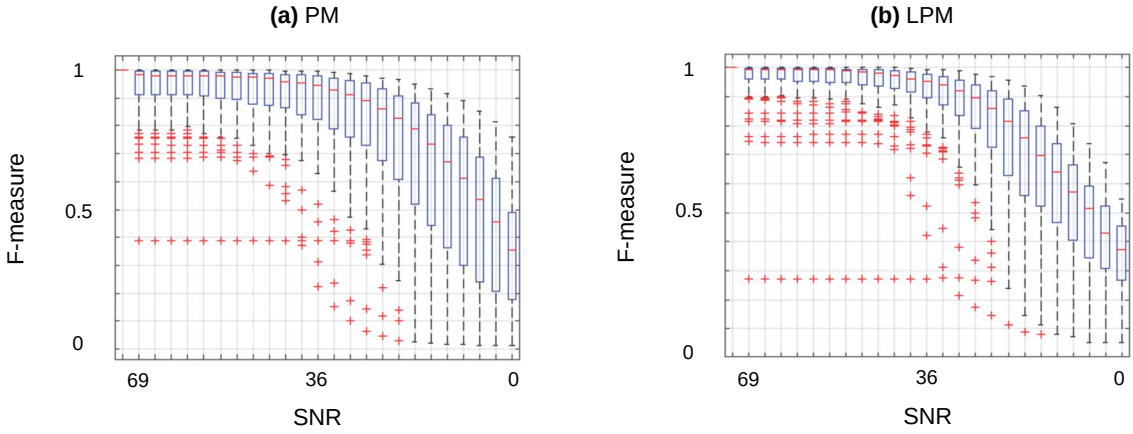

Figure 4.23: F-measure with respect of SNR (dB). The audio sound *restaurant08.wav* (included in the audio degradation tool box [13]) was added to the query as described in Figure 3.9. Horizontal axes span the SNR range of 69 dB to 0 dB. The first value (on the left side) is always one, because the track corresponds to the replica of the query. Boxes indicate the intequartiles where the red line inside each box is the median. In (a), the magnitude spectrogram peak map (PM) was used and in (b) the log-frequency peak map (LPM) was used.

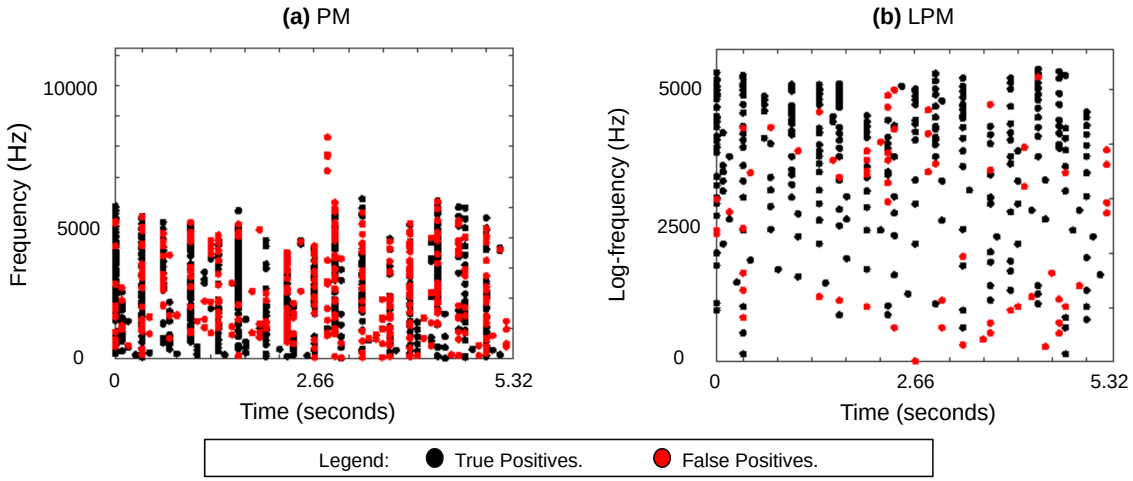

Figure 4.24: Retrieval representation of a track. The track consist of a pub environment added to the query as described in Figure 3.9. The query is loop 12 (guitar) of the genre hip-hop (see Table A.4). The SNR is equal to 21 dB. Horizontal axes span the duration range of 0 to 5.32 seconds (2 bars). Black points denote the matched peaks (true positives). Red points indicate peaks in the track but not in the query (false positives).

4.4.1.3 Vinyl Noise

Figures 4.25, 4.26, and 4.27 show the retrieval information precision, recall, F-measure respectively. Magnitude spectrogram peak maps (PM) were used in (a) whilst log-frequency peak maps (LPM) were used in (b).

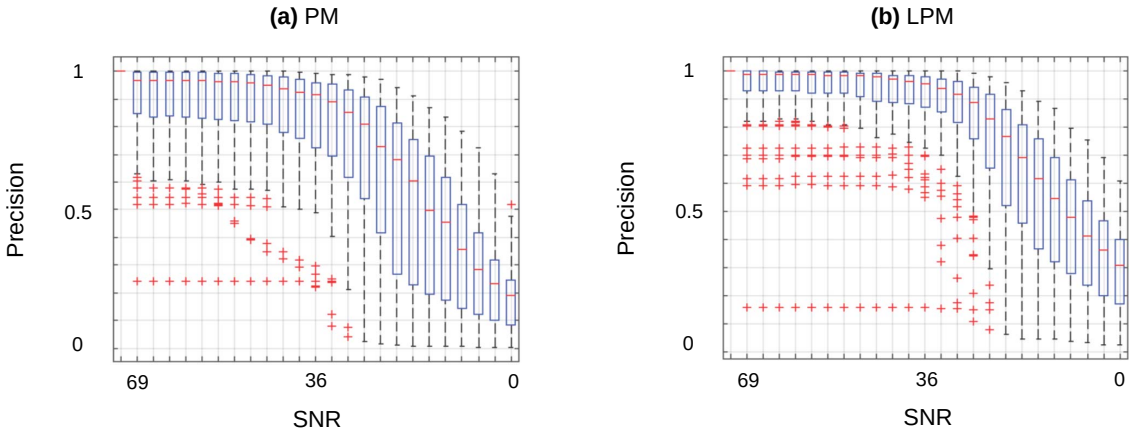

Figure 4.25: Precision with respect of SNR (dB). The audio sound *old-dusty-vinyl-recording.wav* (included in the audio degradation tool box [13]) was added to the query as described in Figure 3.9. Horizontal axes span the SNR range of 69 dB to 0 dB. The first value (on the left side) is always one, because the track corresponds to the replica of the query. Boxes indicate the intequartiles where the red line inside each box is the median. In (a), the magnitude spectrogram peak map (PM) was used and in (b) the log-frequency peak map (LPM) was used.

Precision results in Figure 4.25 indicate the presence of noisy peaks in the track for high SNRs. A strong decrease starts from SNR of 51 dB for PMs and from SNR of 45 dB for LPMs. The F-measure starts to decrease with a high rate when SNR is 51 dB for PMs and 42 dB for LPMs. In the case of peak maps (a), when SNR is equal to 0 dB, the median is 0.20 and results can vary from 0 to 0.42. In the case of LPMs, when SNR is equal to 0 dB, the median is 0.31 and results can vary from 0.01 to 0.60. Loops which do not have a representative amount of peaks can be seen as red crosses below boxes in high SNR (low intensity of the external sound).

Figure 4.26 shows the recall results. Loops that are sensitive to this external sound can be seen as red crosses below boxes. In most cases, the vinyl noise starts affecting the peaks in the query from a SNR of 49 dB for PMs and LPMs. The variance in the recall is smaller and the drops are slower for LPMs. For SNR $= 0$ dB, most of the loops can be identified with at least 15% of the peaks in the query in both (a) and (b).

As a result of such precision and recall, the F-measure in Figure 4.27 shows that in both cases (a) and (b), from SNR 21 dB to 0 dB, the typical values expand a wide range which decrease toward lower values with minimum cases close to zero. LPMs have a better performance since the median is higher and the variances in the F-measure are smaller than in the case of PM; in most cases, matched peaks are relevant in both query and track. Figure 4.28 shows a retrieval representation of a track produced by adding the external sound to the loop 3 (bass) of the genre hip-hop with a SNR of 21 dB. The retrieval representation of the track and query is shown in (a) and (b) respectively. The vinyl sound has a representative number of peaks, however, they do not affect severely the peaks in the query.

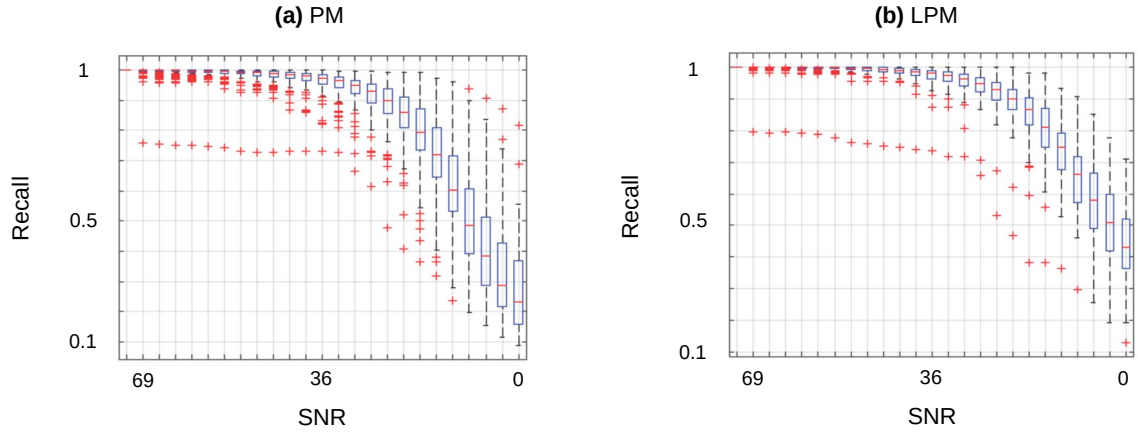

Figure 4.26: Recall with respect of SNR (dB). The audio sound *old-dusty-vinyl-recording.wav* (included in the audio degradation tool box [13]) was added to the query as described in Figure 3.9. Horizontal axes span the SNR range of 69 dB to 0 dB. The first value (on the left side) is always one, because the track corresponds to the replica of the query. Boxes indicate the intequartiles where the red line inside each box is the median. In (a), the magnitude spectrogram peak map (PM) was used and in (b) the log-frequency peak map (LPM) was used.

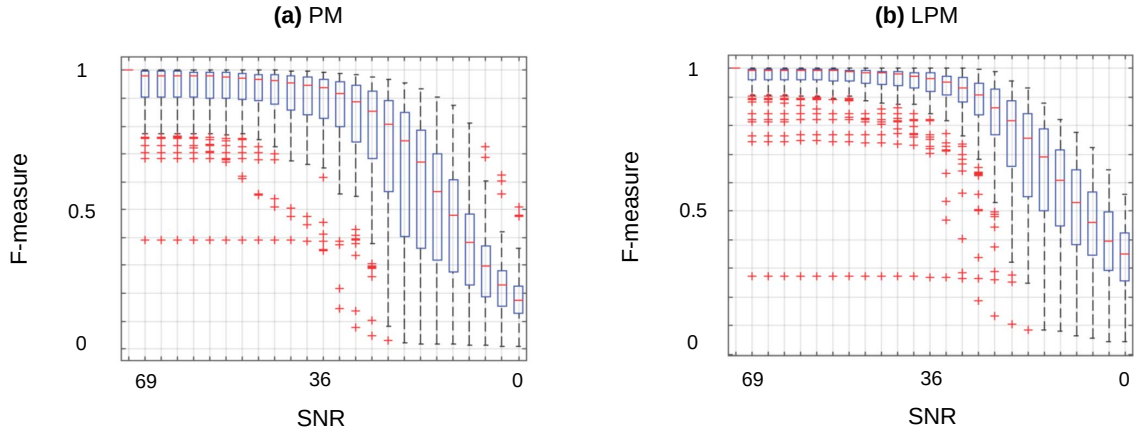

Figure 4.27: F-measure with respect of SNR (dB). The audio sound *old-dusty-vinyl-recording.wav* (included in the audio degradation tool box [13]) was added to the query as described in Figure 3.9. Horizontal axes span the SNR range of 69 dB to 0 dB. The first value (on the left side) is always one, because the track corresponds to the replica of the query. Boxes indicate the intequartiles where the red line inside each box is the median. In (a), the magnitude spectrogram peak map (PM) was used and in (b) the log-frequency peak map (LPM) was used.

4.5 Time Shift Differences within a STFT window frame

As said in Section 3.5, this experiment consists of comparing different shifted versions of a loop. Figure 4.29 shows the F-measure results when we try to match peaks in the query to peaks in the track at the exact time-frequency position. The STFT is increasingly different to those

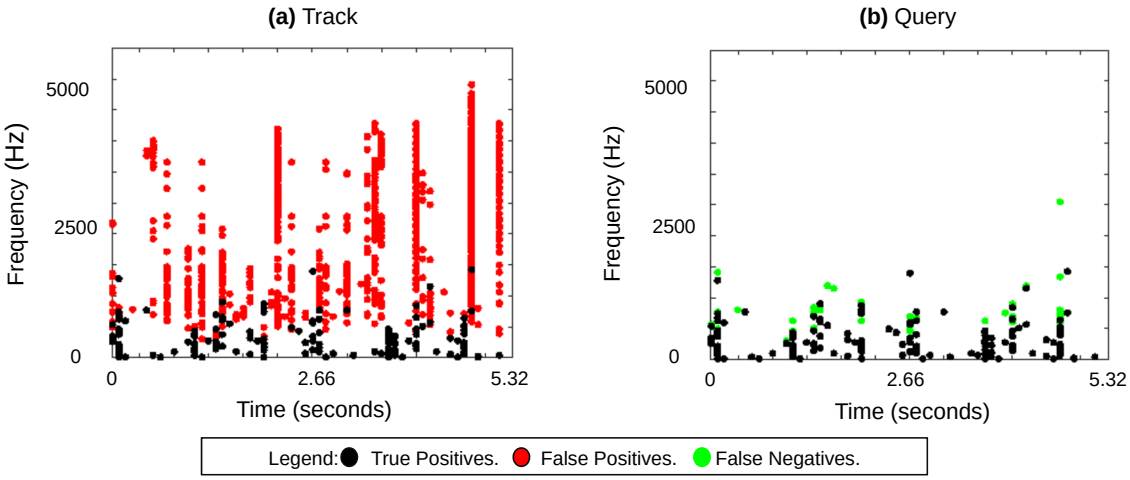

Figure 4.28: Retrieval representation of a track and a query. The track consist of vinyl noise sound added to the query as described in Figure 3.9. The query is loop 3 (bass) of the genre hip-hop (see Table A.4). The SNR is equal to 21 dB. Horizontal axes span the duration range of 0 to 2 bars. Black points denote the matched peaks (true positives). Red points indicate peaks in the track but not in the query (false positives). Green points the peaks in the query but not in the track (false negatives)

computed with shift zero until the *hop* size is reached. In both (a) and (b), the difference is high for values close to zero. In the case of shift 1984 (just before the hop size), the F-measure is close to zero; the median is 0.01.

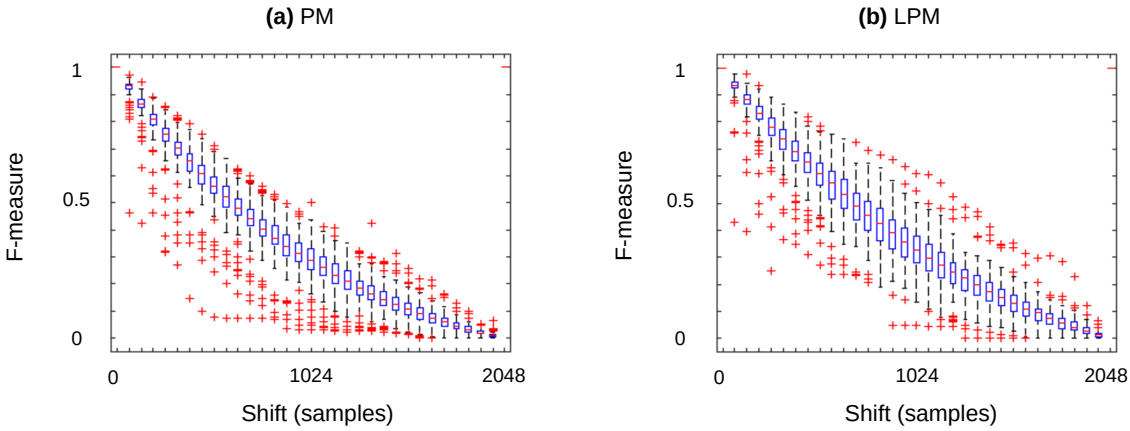

Figure 4.29: F-measure with respect of shift differences (in samples). Horizontal axes span the SNR range of 0 to 1024 samples. The first value (on the left side) is always one, because the track corresponds to the replica of the query (shift 0). Boxes indicate the intequartiles where the red line inside each box is the median. In (a), the magnitude spectrogram peak map (PM) was used and in (b) the log-frequency peak map (LPM) was used. True positives are defined as peaks that matched to peaks in the track at the exact time-frequency position.

Figure 4.30 shows the F-measure results when we try to match peaks in the query to peaks in

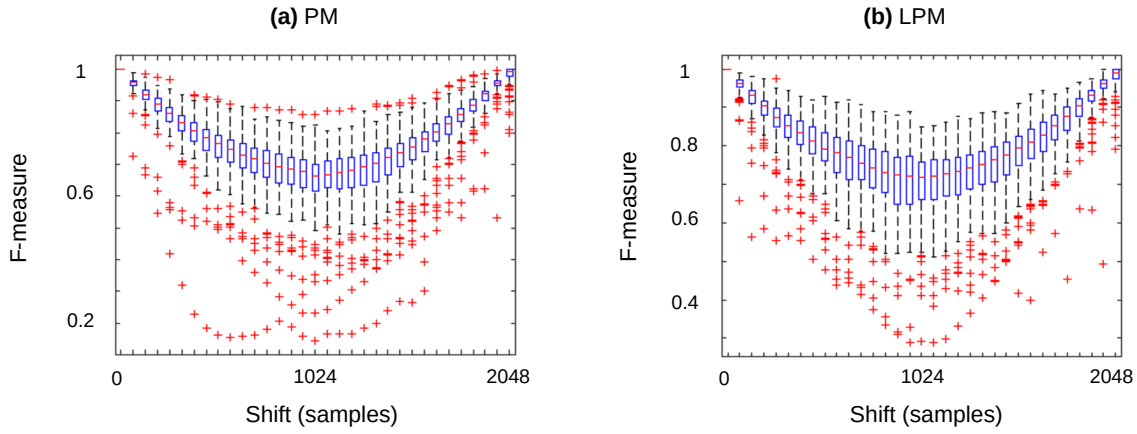

Figure 4.30: F-measure with respect of shift differences (in samples). Horizontal axes span the SNR range of 0 to 1024 samples. The first value (on the left side) is always one, because the track corresponds to the replica of the query (shift 0). Boxes indicate the intequartiles where the red line inside each box is the median. In (a), the magnitude spectrogram peak map (PM) was used and in (b) the log-frequency peak map (LPM) was used. True positives are defined as peaks that matched to peaks in the track at the exact time-frequency position, one time frame on the left, one time frame on the right, one frequency index up, or one frequency index down.

the track at the exact time-frequency position, one time frame to the left, one time frame to the right, one frequency index up, and one frequency index down. As we can see, if we search for matches not only at the exact time-frame position but also at neighbor time-frequency indexes (Figure 4.30), the similarities between shifted versions increase. This is due to the fact that we consider some shift variance in the matching process. In both (a) and (b), the decrease reaches a median value close to 0.7 (more than 50% of matched peaks in both query and track) for shift 1024.

As we can notice, shift differences affect the position of peaks because different information is captured by the windows in the STFT. Figure 4.31 shows the retrieval representation of loop 9 (Fx) of the genre techno (see, Table A.5) where the track is the loop shifted by 1024 samples. In (a), the matching process is done by comparing peaks at the exact time-frequency position. In (b) the matching process is done by considering shift differences of peaks. More matched peaks (true positives) are captured for the case of matching peaks with shift differences; there are more true positives than false negatives.

In summary, in this chapter, we used a collection of 111 audio samples (loops) and described the feature extraction configuration used. Then, we applied experiments such as loop combinations, audio degradation, and shift differences within STFT window frames in order to study the behavior of features. In each experiment we identify parameters and components in the feature extraction procedure that have an important influence on the results obtained. Now, in the following chapter, we discuss the conclusions based on these results and analysis. In addition, we

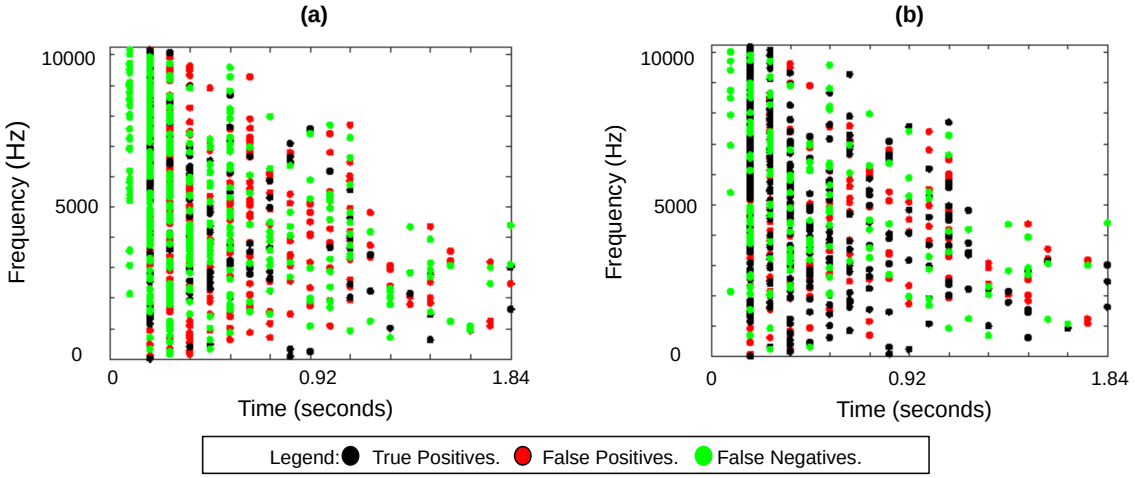

Figure 4.31: Retrieval representation the comparison between shift versions of a query. The query is the loop 9 (Fx) of the genre techno (see, Table A.5). The track consist of the query shifted by 1024 samples. Horizontal axes span the duration range of 0 to 1.84 seconds (1 bar). Black points denote the matched peaks (true positives). Red points indicate peaks in the track but not in the query (false positives). Green points the peaks in the query but not in the track (false negatives).

mention recommendations and ideas for future works.

Chapter 5

Conclusions

In this thesis, we studied the parameters and components of a fingerprint-based sample identification system in electronic music. The fingerprinting technique is based on peak maps, where maximum values (above an amplitude of 1) in the spectrogram are chosen as peaks yielding to a peak map representation. We considered the electronic music scenario in which musical patterns (short fragments of an musical piece) are repeatedly triggered and may appear superimposed with other sound sources. We used a collection of 111 EM music loops organized into 6 genres: dance, deep house, dubstep, hip-hop, techno, and trap. We compared loops with each other and we matched peaks of a loop query into a track produced by complex combinations, audio degradation, or time shift differences. The evaluation of the feature performance was made by a variation of the definition of precision, recall, and F-measure.

Based on the results and analysis in Chapter 4, we demonstrated by using audio sample retrieval matrices that loops can contain information that is also included in other loops. Furthermore, we demonstrated that specific combinations of loops can yield to information that is also contained in other loops which are not present in corresponding mixture.

In complex combinations, the relevance of a loop in a track (in terms of number of peaks) depends of the number of peaks in the loop query. In the case of loops which have a significant amount of peaks, their presence in a mixture may tend to slowly decrease (easy audio recognition) while in the case of a loop with a small amount of peaks, their presence in a combination may sharply decrease (hard audio recognition). Furthermore, information produced by audio combinations can severely affect the recognition of loops which contain a small amount of representative peaks since small changes can represent high variation of information in such loops. In addition, matched peaks strongly decrease once an audio sample is added and it slowly decrease for combinations of 5 to 10 loops.

Amplitude limitations in the fingerprinting process such as exponential decay and minimum

amplitude of a peak can help to avoid noisy or irrelevant components of an audio signal. However, there are loops whose peaks values are close to the amplitude thresholds. Their peaks in combination with other sounds may be distorted and/or may represent a small amount of peaks in the mixture, thus, the recognition of such loops can be negatively affected.

Magnitude spectrogram peak maps (PMs) have a stronger decrease in complex combination when comparing to log-frequency peak maps (LPMs). The logarithmic scale helps to emphasize certain frequency components (related to the human auditory system), where distortions are less powerful and there are less noisy peaks.

Time-shift differences within STFT window frames produce changes into the peak maps of an audio signal. In the case of matching peaks at the exact time-frequency index, the STFT of an audio signal is increasingly different to the original audio (audio without time shift) reaching a minimum value of matched peaks close to zero just before the *hop* size. In the case of matching peaks at the exact time-frequency index and neighbor time-frequency indexes, the similarities slowly decrease where a minimum is reached at the shift of half of the hop size, however, this minimum in most cases indicate less than 50% of different peaks.

The behavior of the features in addition to the identified parameters and components of a fingerprint-based sample identification system can lead to interesting approaching in music structure analysis not only in electronic music but also in other genres in which sample identification is used. We recommend to do experiments of volume changes proposed in Section 3.4.2.2 using a realistic larger data-set. In our initial experiments, we showed that peaks are not chosen due to the amplitude limitations. This limitations can affect particular loops which are sensitive to amplitude variations and/or with a small amount of peaks, thus, recognition can be hard in cases of having tracks or queries with low volume. Experiments on a realistic larger data-set can help to identify such types of loops and to obtain a better understanding of the behavior of the features under this scenario. Furthermore, we recommend to elaborate experiments of audio matching with shift-queries or with different query versions. In our initial experiments (Section 3.6), we showed that using at least 2 time-shifted queries, the matching procedure can discriminate irrelevant values and obtain better matching performance. Therefore, experiments on a realistic larger data-set can help to conclude to general behaviors of the matching performance and identify particular cases.

Experiments with adding effects to a larger-scale data set may lead to interesting conclusions about the behavior of the features in electronic music. In our initial experiments (Section 3.4.2.1), we showed that reverberation effects and delay can affect the recognition of an audio sample. These audio effects add information in a track that can reduce the relevance of a loop in terms of the peaks, and/or can make some peaks to not appear in the peak map of the track (less matched peaks).

Appendix A

Data-Set Description

The following tables give a general description of each loop of the data-set used in Chapter 4. A total of 111 loops were colleted from the data-base of "MAGIX Music Maker Premium" software [12]. Each table shows a list of loops corresponding to the electronic music genres: *dance*, *deep house*, *dubstep*, *hip-hop*, *techno* and *trap*.

Audio file name	Instrumental family	BPM	F_s	Size (samples)	Duration (seconds)	Bar
Patrick Bass A 1.ogg	Bass	130	44100	162830	3.692	2
Patrick Beat A.ogg	Drums	130	44100	162830	3.692	2
Lindstroem Fx 1.ogg	Fx	130	44100	325662	7.385	4
Straight On 1.ogg	Guitar	130	44100	651324	14.769	8
Maya Piano 1.ogg	Keys	130	44100	162832	3.692	2
Pierre Pad 1.ogg	Pad	130	44100	325662	7.385	4
Percusser A.ogg	Percussion	130	44100	325662	7.385	4
Mats Synth B 1.ogg	Sequence	130	44100	162832	3.692	2
El Acid A 1.ogg	Synth	130	44100	162832	3.692	2
Be There 1.ogg	Vocals	130	44100	651324	14.769	8

Table A.1: Audio loops from the genre *dance*.

Audio file name	Instrumental family	BPM	F_s	Size (samples)	Duration (seconds)	Bar
Jupiter Bass 1.ogg	Bass	120	44100	176400	4.000	2
Wild Trumpet 1.ogg	Brass	120	44100	352800	8.000	4
Basic Beat A.ogg	Drums	120	44100	176400	4.000	2
Free Beat A.ogg	Drums	120	44100	176400	4.000	2
Jupiter Beat A.ogg	Drums	120	44100	176400	4.000	2
Polar Beat E.ogg	Drums	120	44100	176400	4.000	2
Polar Scape 1.ogg	Fx	120	44100	176400	4.000	2
Old Fairytale 1.ogg	Pad	120	44100	176400	4.000	2
Basic Congas.ogg	Percussion	120	44100	176400	4.000	2
Moonchild Clave.ogg	Percussion	120	44100	88200	2.000	1
Origin Perc 1.ogg	Percussion	120	44100	352800	8.000	4
Jupiter Sequence 1.ogg	Sequence	120	44100	176400	4.000	2
Jupiter Rhodes 1.ogg	Special	120	44100	176400	4.000	2
Polar Lead 1.ogg	Special	120	44100	176400	4.000	2
Solar Chord 1.ogg	Special	120	44100	176400	4.000	2
Free Orchester 1.ogg	Strings	120	44100	352800	8.000	4
Warm Sine 1.ogg	Synth	120	44100	176400	4.000	2
Warm Voice A 1.ogg	Vocals	120	44100	176400	4.000	2
Wild Vox 1.ogg	Vocals	120	44100	176400	4.000	2

Table A.2: Audio loops from the genre *deep house*.

Audio file name	Instrumental family	BPM	F_s	Size (samples)	Duration (seconds)	Bar
KomplexWobble 1.ogg	Bass	135	44100	313600	7.111	4
Sub Bass 1.ogg	Bass	135	44100	313600	7.111	4
Clean Set A.ogg	Drums	135	44100	627200	14.222	8
Cutwater A.ogg	Drums	135	44100	627200	14.222	8
TimeKeepsTicking A.ogg	Drums	135	44100	627200	14.222	8
Clean UpLifter.ogg	Fx	135	44100	313600	7.111	4
CoinFX.ogg	Fx	135	44100	313600	7.111	4
Noise Uplifter FX.ogg	Fx	135	44100	313600	7.111	4
TalkingPiano 1.ogg	Keys	135	44100	313600	7.111	4
Aprocalyptic Break 1.ogg	Lead	135	44100	627200	14.222	8
MelodyLead Saw 1.ogg	Lead	135	44100	313600	7.111	4
MysticDgtlChoir 1.ogg	Pad	135	44100	313600	7.111	4
VectorFM Pad 1.ogg	Pad	135	44100	313600	7.111	4
CosmosTwoArp 1.ogg	Sequence	135	44100	313600	7.111	4
DarkAnlgPowerArp 1.ogg	Sequence	135	44100	313600	7.111	4
OldSpaceArp 1.ogg	Sequence	135	44100	313600	7.111	4
Impact Wobble 1.ogg	Synth	135	44100	313600	7.111	4
PartyDropTri 1.ogg	Synth	135	44100	313600	7.111	4
YahyahWobble 1.ogg	Synth	135	44100	313600	7.111	4

Table A.3: Audio loops from the genre *dubstep*.

Audio file name	Instrumental family	BPM	F_s	Size (samples)	Duration (seconds)	Bar
Dizzy 1.ogg	Bass	90	44100	235200	5.333	2
Jungle_Time 1.ogg	Bass	90	44100	235200	5.333	2
Tip_Bass 1.ogg	Bass	90	44100	235200	5.333	2
French_Horn 1.ogg	Brass	90	44100	235200	5.333	2
Big_Bang A.ogg	Drums	90	44100	235200	5.333	2
Double_Kit A.ogg	Drums	90	44100	470400	10.667	4
Finger_Snap A.ogg	Drums	90	44100	235200	5.333	2
Hisser A.ogg	Drums	90	44100	235200	5.333	2
Play_Kit A.ogg	Drums	90	44100	235200	5.333	2
Effect K.ogg	Fx	90	44100	235200	5.333	2
Scratch A.ogg	Fx	90	44100	235200	5.333	2
Hot_Pick 1.ogg	Guitar	90	44100	235202	5.333	2
Strange_Sound 1.ogg	Keys	90	44100	235200	5.333	2
5th_Pad 1.ogg	Pad	90	44100	235200	5.333	2
Delayed 1.ogg	Sequence	90	44100	235200	5.333	2
Cello 1.ogg	Strings	90	44100	235200	5.333	2
Tip_Cello 1.ogg	Strings	90	44100	235200	5.333	2
Wow_Stab 1.ogg	Synth	90	44100	235200	5.333	2
Attraction 1.ogg	Vocals	90	44100	470402	10.667	4
Adlib_Yeah B.ogg	Vocals Raps	90	44100	117602	2.667	1

Table A.4: Audio loops from the genre *hip-hop*.

Audio file name	Instrumental family	BPM	F_s	Size (samples)	Duration (seconds)	Bar
Astral Bass 1.ogg	Bass	130	44100	162832	3.692	2
Dark Bass 1.ogg	Bass	130	44100	162832	3.692	2
Haunted Bass 1.ogg	Bass	130	44100	81416	1.846	1
Broken Beat A.ogg	Drums	130	44100	162830	3.692	2
Broken Beat B.ogg	Drums	130	44100	162830	3.692	2
Dark Beat A.ogg	Drums	130	44100	162832	3.692	2
Dark Beat B.ogg	Drums	130	44100	162832	3.692	2
Space Beat A.ogg	Drums	130	44100	162832	3.692	2
Space Sharp Riser 3.ogg	Fx	130	44100	325662	7.385	4
Space Organ 1.ogg	Keys	130	44100	325662	7.385	4
Haunted Bells 1.ogg	Mallet	130	44100	162832	3.692	2
Haunted Pad 1.ogg	Pad	130	44100	162830	3.692	2
Dark Groove With Sidechain.ogg	Percussion	130	44100	162832	3.692	2
Lunar Hihats.ogg	Percussion	130	44100	81416	1.846	1
Magnetic Rythm.ogg	Percussion	130	44100	162832	3.692	2
Space Clicker.ogg	Percussion	130	44100	162832	3.692	2
Astral Sequence 1.ogg	Sequence	130	44100	162832	3.692	2
Broken Cycle 1.ogg	Sequence	130	44100	162832	3.692	2
Future Toms 1.ogg	Sequence	130	44100	81416	1.846	1
Haunted Groove 1.ogg	Sequence	130	44100	81416	1.846	1
Haunted Phaser 1.ogg	Sequence	130	44100	162832	3.692	2
Obscure Dancer 1.ogg	Sequence	130	44100	81416	1.846	1
Space Ping 1.ogg	Sequence	130	44100	162832	3.692	2

Table A.5: Audio loops from the genre *techno*.

Audio file name	Instrumental family	BPM	F_s	Size (samples)	Duration (seconds)	Bar
Klong Oxi A 1.ogg	Bass	75	44100	282240	6.400	2
Plucky Deep 1.ogg	Bass	75	44100	282240	6.400	2
Yai Duo 1.ogg	Bass	75	44100	282240	6.400	2
Plucky Tro En 1.ogg	Brass	75	44100	564480	12.800	4
Cheeze Combi C.ogg	Drums	75	44100	282240	6.400	2
Cheeze Combi E.ogg	Drums	75	44100	282240	6.400	2
Final Chapter A.ogg	Drums	75	44100	282240	6.400	2
Yai Lizard D.ogg	Drums	75	44100	282240	6.400	2
Cheeze Down A.ogg	Fx	75	44100	282240	6.400	2
Cheeze Single 1.ogg	Fx	75	44100	282240	6.400	2
Freaky UpUp 1.ogg	Fx	75	44100	564480	12.800	4
Scream Deep 1.ogg	Pad	75	44100	282240	6.400	2
Scream Step 1.ogg	Sequence	75	44100	141120	3.200	1
Yai Spoken 1.ogg	Sequence	75	44100	282240	6.400	2
Freaky Uni 1.ogg	Synth	75	44100	141120	3.200	1
Dip Fx A.ogg	Vocals	75	44100	282240	6.400	2
Dip Fx B.ogg	Vocals	75	44100	564480	12.800	4
Dip Squad Hook A.ogg	Vocals	75	44100	564480	12.800	4
Dip Squad Verse 1 A.ogg	Vocals	75	44100	564480	12.800	4
Faded Fx A.ogg	Vocals	75	44100	564480	12.800	4

Table A.6: Audio loops from the genre *trap*.

Bibliography

[1] Ableton AG, *Ableton live*. https://www.ableton.com/, Retrieved August 2016, 2016.

[2] Attack Magazine. http://www.attackmagazine.com/, Retrieved August 2016, 2016.

[3] M. J. Butler, *Unlocking the Groove: Rhythm, Meter, and Musical Design in Electronic Dance Music*, Profiles in popular music, Indiana University Press, 2006.

[4] Dolmetsch Organisation. http://www.dolmetsch.com/musictheorydefs.htm, Retrieved September 2016, 2016.

[5] D. P. Ellis and G. E. Poliner, *Identifying 'cover songs' with chroma features and dynamic programming beat tracking*, in Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), vol. 4, Honolulu, Hawaii, USA, 2007.

[6] Freesound. http://freesound.org/, Retrieved June 2016, 2016.

[7] J. Haitsma and T. Kalker, *A highly robust audio fingerprinting system*, in Proceedings of the International Society for Music Information Retrieval Conference (ISMIR), Paris, France, 2002, pp. 107–115.

[8] J. A. Hockman, M. E. P. Davies, and I. Fujinaga, *Computational strategies for breakbeat classification and resequencing in Hardcore, Jungle and Drum & Bass*, in Proceedings of the International Conference on Digital Audio Effects (DAFx), Trondheim, Norway, December 2015.

[9] International Audio Laboratories, *Dataset*. https://www.audiolabs-erlangen.de/resources/MIR/2016-ISMIR-EMLoop, Retrieved June 2016, 2016.

[10] F. Kurth and M. Müller, *Efficient index-based audio matching*, IEEE Transactions on Audio, Speech, and Language Processing, 16 (2008), pp. 382–395.

[11] P. López-Serrano, C. Dittmar, J. Driedger, and M. Müller, *Towards modeling and decomposing loop-based electronic music*, in Proceedings of the International Conference on Music Information Retrieval (ISMIR), New York, USA, 2016, pp. 502–508.

[12] MAGIX COMPUTER PRODUCTS INT'L CORP., *Magix Music Maker Premium*. https://www.magix.com,RetrievedJuly2016, 2016.

[13] M. Mauch and S. Ewert, *The audio degradation toolbox and its application to robustness evaluation*, in Proceedings of the 14th International Society for Music Information Retrieval Conference (ISMIR 2013), Curitiba, Brazil, 2013.

[14] M. Müller, *Fundamentals of Music Processing – Audio, Analysis, Algorithms, Applications*, Springer Verlag, 2015.

[15] A. V. Oppenheim, A. S. Willsky, and H. Nawab, *Signals and Systems*, Prentice Hall, 1996.

[16] Propellerhead Software, *Reason*. `https://www.propellerheads.se/`, Retrieved August 2016, 2016.

[17] B. Rocha, N. Bogaards, and A. Honingh, *Segmentation and timbre similarity in electronic dance music*, in Proceedings of the Sound and Music Computing Conference (SMC), Stockholm, Sweden, 2013, pp. 754–761.

[18] R. Snoman, *Dance Music Manual: Tools, Toys, and Techniques*, Taylor & Francis, 2013.

[19] J. Van Balen, J. Serrà, and M. Haro, *Sample Identification in Hip Hop Music*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013, pp. 301–312.

[20] A. Wang, *An industrial strength audio search algorithm*, in Proceedings of the International Society for Music Information Retrieval Conference (ISMIR), Baltimore, Maryland, USA, 2003, pp. 7–13.