Automatic Detection of Audio Defects in Personal Music Collections

Ignasi Adell Arteaga

MASTER THESIS UPF / 2016

Master in Sound and Music Computing

Master thesis supervisor:

Perfecto Herrera Boyer

Department of Information and Communication Technologies

Universitat Pompeu Fabra, Barcelona

UNIVERSITAT
POMPEU FABRA

To my parents and my grandparents, specially my grandfather Manolo, who passed away and could nott see me finishing this master and my graduation. I dedicate it to him with all my love.

Acknowledgements

I would like to thank you all my master colleagues for the help received during the last two years, and for the good environment we created during the course.

I would specially thank you my thesis' supervisor, Perfecto Herrera, for the guidance during this work and helping me to deal with the limitation of time. He really transmitted his knowledge in the classes and during the realization of the thesis.

And the last but not least, I would like to thank you Xavier Serra and all the SMC/MTG members for allowing me to study this master and help me to extend my knowledge about audio, music and research. It has been a long 2-year journey where I have learn many aspects about the current technology already developed and being currently developed for Music Information Retrieval purposes.

Summary

Personal digital music collections have been growing in the last decade due to audio formats like MP3, WAVE or FLAC. They usually come from diverse sources and some may not be always reliable. They may have clicks, gaps, bad equalization, clipping, noise or many other kinds of alterations.

Vast amounts of important audio material, from historic recordings to relatively recent recordings on analogue or even primitive digital media, were re-released on the new digital media formats. Digital audio restoration has had an increasing application to sound recordings from the Internet, however, there is still a very large amount of music collectors with lots of music in legacy formats such as vinyl or cassette that with the advent of sharing culture through Internet are making available many digitized audio files that may not have passed through a proper audio restoration.

The quality assessment of audio signal has made a lot of improvements since the digital signal processing (DSP) techniques appeared and started to be used for sound restoration purposes. Subjective assessment of audio quality has been used for a long time, but its time-consuming and external human and technical influences (such as listener’s expertise, sensitivity or the evaluation equipment) have lead to objective approaches, such as the PEAQ for wideband audio signals and PESQ for speech signals in ITU standard regulations. Some of these issues have been already addressed and they even have commercial implementations. However, there is still room for research for some others.

In this work, current taxonomy of known audio defects is reviewed according to the state of the art methods, highlighting the characteristics of each type and the solutions (if any) for their detection and correction. Afterwards, the vinyl technology is analyzed due to its error-prone nature. That is why the defects related to digitizing vinyl media are chosen for research here: the lack of RIAA filtering and the altered playback speed.

Later, the mechanisms for detection are exposed. Those mechanisms are based on the psychoacoustic model developed by Zwicker (that is, the use of bark-band decomposition of the spectrum) and state-of-the-art machine learning techniques. Their implementation is defined based on preliminary data obtained from a reduce dataset of 200 instances split in 10 different genres.

The resulting algorithms are evaluated under an extended defect-controlled dataset of 2000 and 800 files respectively. Two different machine-learning techniques are used, a decision-tree (C4.5) and Support Vector Machines (SMO).

The accuracy is discussed for both of them against the global dataset and per genres subsets (in the case of the the lack of RIAA filtering) using the 10-Fold cross-validation method. Finally their doability for the problem under test is analyzed and further improvements are suggested.

Index

1. Introduction

Personal digital music collections have been growing in the last decade due to audio formats like MP3[1], AAC [9], WAVE[2][3] and FLAC[4]. Nowadays, Internet is the main source of music content for the majority of consumers. They can easily find their favourite music and download it from many different sources[5][6]. However, some of the obtained audio files might not be very reliable. They can have many different kinds of alterations: glitches, gaps, bad equalization, clipping, noise, incompleteness, time-stretching, they might have been transcoded from lossy formats, or they might have bit level errors due to being transferred through Internet. In addition, Compact Disk media (CD) [1] may yield reading errors due to mechanical and surface problems while reading the data stored in them, which may lead to having faulty encoded files. Also, when compressing digital audio files the perceived quality of compressed audio depends on the dynamic range of the encoded track [22].

These alterations or errors present in audio files are known in the Music Information Retrieval field as *audio defects*. An audio defect is a consequence of the degradation of an audio source. It will be considered as any undesirable modification to the audio signal [33], which can appear due to many different causes:

- **Digitization**: while transferring legacy media (vinyl noise, warped vinyl, CD reader errors, CD scratches, cassette degradation).
- **Transcoding** among formats: failures in the encoding algorithms.
- **Mastering** mistakes: Stereo-imaging alteration, low quality equipment.
- **Faulty recordings** due to noisy environments, faulty recording equipment.
- **Compression** errors for broadcasting/downloading: lossy/lossless compression issues.
- **Transmission** errors while transferring files: data loss, improper transmission equipment.
- **Filtering** issues: undesired audio effects, signal cancellations.

The application of digital signal processing (DSP) to problems in audio has been an area of growing importance since the pioneering DSP work of the 1960s. In the 1980s, DSP micro-chips became sufficiently powerful to perform the complex processing operations required for sound restoration. This led to the first commercially available restoration systems, with companies such as CEDAR Audio Ltd[7] or Sonic Solutions[8] selling dedicated systems to recording studios, broadcasting companies, media archives and film studios. Vast amounts of important audio material, from historic recordings to relatively recent recordings on analogue or even digital media, were noise-reduced and re-released on the new digital media formats. Digital audio restoration has therefore an increasing application to sound recordings from the Internet, and high-quality defect reducers are a standard part of any computer or HIFI system. However, there is still a very large amount of music collectors with many much music in legacy formats such as vinyl or cassette that are making available many digitized audio files that may not have

[1] http://www.iis.fraunhofer.de/en/ff/amm/prod/audiocodec/audiocodecs/mp3.html
[2] http://soundfile.sapp.org/doc/WaveFormat/
[3] http://www.digitalpreservation.gov/formats/fdd/fdd000001.shtml
[4] https://xiph.org/flac/
[5] http://whymusicmatters.com/find-music
[6] http://www.digitaltrends.com/music/best-free-and-legal-music-download-sites/
[7] http://www.cedaraudio.com/products/products.shtml
[8] https://en.wikipedia.org/wiki/Sonic_Solutions

passed through a proper audio restoration process, leading to unreliable quality files circulating on the Web.

1.1. Audio quality

Many applications in the music industry rely on high quality standards, mainly when offering products or services to a huge number of costumers. Internet stores or subscription platforms are just some examples of important stakeholders that deserve a defect-free corpus of music files to be used for their business purposes.

Digitization techniques, in charge of developing digital preservation strategies, add more complexity to quality requirements. Normally, master files (initial acquisitions from the very first source, i.e. the recording studios) are of higher quality and therefore claim more storage space. Quality assurance (QA) is an essential tool for estimating the loss of information during the process of digitization, that must focus on maintaining the quality after migration or curation actions, and to verify that the migrated collection matches to those quality standards.

In order to ensure a certain level of audio quality in a large music collection, an automated procedure to check its status is needed. Here is where Music Information Retrieval researchers play an important role to develop mechanisms and systems that will be able to detect any possible issue in the audio signal so that they can be corrected or at least detected and therefore increase the value of our music collections, either by correcting those files or marking them as defective. Currently, many of the commonly applied quality metrics do not provide an accurate interpretation of information loss and distortions; on the other hand, quality assessment of digitized collections is in many parts an unresolved matter. In addition, many of the quality assurance processes have been relying mostly on subjective testing (as explained in Chapter 2), consequently involving many other personal or external variables that may affect the final verdict on a given digitized item. As exposed by *Schindler and Huber-Mörk in [20]*, perceptual quality metrics should play an essential part in quality assurance of either initial digitization, migration or curation[9] workflows.

Assessing the quality of audio objects or collections lacks of reference objects to calculate relative quality estimations. Non-Reference quality assessment (also known as blind- or non-intrusive quality assessment) tries to define objective estimates describing distortions of audible stimuli that correlate to the often-called subjective *mean opinion scores (MOS)*[10]. Applying the described user-evaluated attributes to blind-quality estimates would provide an invaluable objective measure for assessing the quality of digitized objects.

[9] Digital curation is the selection, preservation, maintenance, collection and archiving of digital assets. Digital curation establishes, maintains and adds value to repositories of digital data for present and future use. Successful digital curation will mitigate digital obsolescence, keeping the information accessible to users indefinitely.

[10] http://www.ntt.co.jp/qos/qoe/eng/technology/sound/03_1.html

1.2. Motivation

As can be seen in various reports from the recording industry like Nielsen[11], IFPI[12] and RIAA[13], vinyl sales have raised quite a lot in the last decade. An evolution of it can be seen in Figure 1.

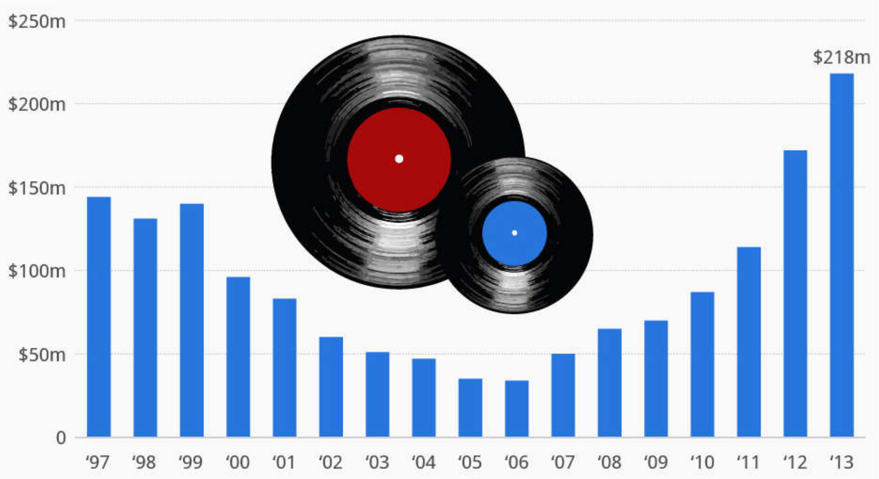


Figure 1. IFPI's worldwide vinyl sales evolution from 1997 to 2013.

These results actually mean that this music format is still taken into account by the customers and much of the audio data being shared across Internet may have its source in this technology. Although vinyl quality improved in the last 3 decades with formats like 180-220gram records[14], the promising high-definition vinyl[15] and new pressing methods[16], this medium has some already known issues[17]: noise (due to its reduced dynamic range, 60-80 dB versus 90-96 dB for CD recordings), tracking distortion or rotor fluctuations in the playback equipment are some examples. Thus, they can cause that digitized files are not good enough in terms of quality. One can find some audiophile and non-audiophile blogs[18][19][20][21] where those problems are exposed and solutions are proposed among vinyl consumers. However, there are still some issues unaddressed that particularly relate to vinyl. In next section, that problematic is reviewed among many other audio defects one can find in their personal music collections.

The aim of thesis is therefore to extend the quality assessment on specific audio issues focusing on defects still unaddressed. As it is exposed in the next chapter, these issues are two: the lack of RIAA filtering and the altered playback speed, both related to vinyl recordings. The RIAA filtering is a pre-processing needed to be done prior to recording

[11] http://www.homescanprivacy.com/us/en/insights/news/2015/thanks-to-strong-sales-vinyl-albums-are-off-and-spinning.html
[12] http://www.ifpi.org/news/IFPI-publishes-global-vinyl-market-details
[13] http://www.riaa.com/vinyl-still-rocks/
[14] http://blog.vinylgourmet.com/2015/10/180-gram-vinyl-what-are-the-benefits-heavyweight-vinyl-records-explained.html
[15] http://www.digitalmusicnews.com/2016/03/15/high-definition-vinyl-will-soon-become-a-reality/
[16] http://dancingastronaut.com/2016/02/record-sales-vinyl-lead-new-pressing-technology/
[17] http://www.recordingmag.com/resources/resourceDetail/113.html
[18] https://hydrogenaud.io/index.php/board,70.0.html
[19] http://www.diyaudio.com/forums/analogue-source/
[20] https://www.facebook.com/vinylproblems/
[21] http://yabb.jriver.com/interact/index.php?PHPSESSID=3lffm8qa33uuh4ivio1v9jctk2&topic=73005.0

the audio to the physical vinyl format, so that it matches the conditions of the medium. This pre-process is then reversed when playing back the vinyl record in order to recover the original sound. The other issue under test is the detection of an altered playback speed. The idea behind this is to detect if the vinyl record was played at the nominal speed or not while it was being digitized. As it is explained in the next chapter, turntables have a pitch control (speed control), so the rotation motor can run faster or slower. If this control is changed (not set to 0) when digitizing the vinyl, the resulting audio file will not be the same as the original. Therefore, the purpose of the algorithm presented here in the following chapters is to detect this problem.

2. State of the art

In this chapter some of the research already done in the field is reviewed, and current status of it is exposed. First, recent research on quality evaluation and later audio defect types are explained along with related work on each of them.

2.1. Quality evaluation in digital music collections

Although there's no consistent definition for quality yet, for certain restricted circumstances, it has an understood meaning for audio [6]. Audio quality can be defined as a subjective metric that describes music audio content, as an assessment of the accuracy, enjoyability, intelligibility of audio[22] or a perceptual reaction to the sound of a product that reflects the listener's reaction to how acceptable the sound of that product is[23]. As music is performed, recorded, and then perceived by users in a sequential manner, audio quality can also be assessed from different perspectives along this process. Many aspects are involved to assess audio quality, including musical aspects, environmental and equipment-related aspects for the recording conditions, and pleasantness or appealing aspects for the end-user perception of the recording.

Although it is obviously easy for humans to judge the quality of a digital artefact, it is a challenging task to describe objective measures that can be used to automate quality inspection of digital collections. Thus, quality evaluation can be considered from two different scopes: the subjective assessment and the objective assessment.

First attempts in audio quality assessment were performed through subjective approaches. That means creating a group of listeners and building a set of questions for them to answer. Some examples can be found in the works done by Herrero [25], Repp [53] and Zimmerman, Levitin and Guastavino [8], where researchers elaborate different kinds of questionnaires and surveys are performed on a significant amount of listeners. As an example, Zimmerman et al [8] perform a subjective evaluation experiment on different listeners in order to determine the perceptual effects of different MP3 compressions. They investigate if listeners prefer CD quality to mp3 files at various bitrates (from 96kbps to 320kbps), and if this preference is affected by the musical genre. Their conclusion is that listeners significantly preferred CD quality to MP3 files up to 192kbps for all musical genres. In addition, it was observed a significant effect of expertise (sound engineers vs. musicians) and musical genres (electric vs. acoustic). The results indicate that MP3 compression introduces audible artefacts, and that listeners' sensitivity to them varies depending on their musical expertise. Specifically, trained listeners can discriminate, report and significantly prefer CD quality over MP3 compressed files for bitrates ranging from 96 to 192kbps. Obviously, the conclusions drawn by those works can be extrapolated to almost any subjective attempt on audio evaluation, where external aspects such as the listener's musical training, listener's expertise, or the used audio equipment have to be considered as variables in the experiment, although not belonging to the nature of the recordings under evaluation.

[22] https://en.wikipedia.org/wiki/Sound_quality

[23] http://www.salford.ac.uk/computing-science-engineering/research/acoustics/psychoacoustics/sound-quality-making-products-sound-better/sound-quality-testing/defining-sound-quality

There are also some standards on how to proceed in the subjective assessment of audio signals, mainly published regulations from the ITU[24] (International Telecommunications Union), such as the ITU-R BS.1284-1 [10], the ITU-R BS.1534-1 [11] or the ITU-R BS.1116 [12]. BS.1284 presents "General methods for the subjective assessment of sound quality", where the experimental design of listening tests is exposed, plus the test methodology concerning test procedures and grading scales is defined. BS.1534 specifies the "Method for the subjective assessment of intermediate quality level of coding systems" and BS.1116 is used for the evaluation of high quality audio systems having small impairments. The ITU recommendations for the subjective assessment of audio quality are very efficient, but difficult to implement due to the aforementioned aspects such as the need of a large set of listeners or the proper setup for the surveys to be performed.

The objective evaluation of the quality of digital audio collections requires identifying the amount of lost information. Due to semantic discrepancies [24] between low-level audio descriptors (for example Mean Squared Error, MSE) and perceived quality degradation (unnoticeable, annoying...), similarity measures estimating the perceived quality audio files are normally required. Further analysis of the content (often requiring concrete domain knowledge) might be necessary. In addition, a severity scale needs to be defined to classify corruptions occurred after recording, migration, transmission or reading. Existing quality metrics have to be evaluated on how they align to such severity definitions and if they are robust under certain levels.

So far, most of the work reported on reference-less quality estimation relies on the use of prior knowledge of quality degrading processes. Such processes have to be identified and analyzed in order to define proper models that can be used to formulate adequate non-reference quality estimates for digital collections. However, music is lacking proper definitions of quality issues and degradation models, as well as attempts towards comprehensive objective perceptual quality metrics. Previous research on objective assessment of sound quality started in the early 1990s. ITU released some regulations which cover various assessment methods for both audio quality (Perceptual Evaluation of Audio Quality, *PEAQ* [21]) and speech quality (Perceptual Evaluation of Speech Quality, *PESQ* [25]). These approaches generally compare the quality of the sound signals processed/affected by a test system (a multimedia device, a codec, or a telecommunication network...) with a reference signal in order to evaluate or improve the performance of the system.

Particularly for the audio domain, *PEAQ* model compares a signal that has been processed in some way with the corresponding original signal (see Figure 2). Concurrent frames of the original and processed signal are transformed into a time-frequency representation by the psychoacoustic model of hearing developed by Zwicker and Fastl [18], using Discrete Fourier Transform and filter banks. Then a task-specific model of auditory cognition reduces these data to a number of model output variables (MOV[25]), and finally, those scalar values are mapped to the desired quality measurement.

[24] http://www.itu.int/en/Pages/default.aspx

[25] https://en.wikipedia.org/wiki/Model_output_statistics

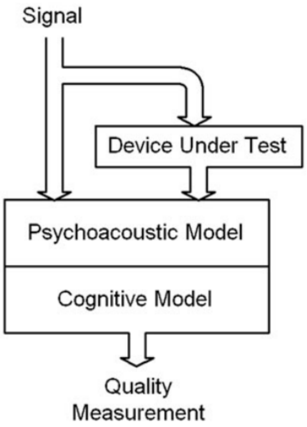


Figure 2. PEAQ's high-level description model

Important information for yielding the quality measurement is derived from the differences between the frequency and pitch domain representations of the reference and test signals. In the frequency domain, the spectral bandwidths of both signals are measured and the harmonic structure in the error is determined. In the pitch domain, error measures are derived from the excitation envelope modulations, the excitation magnitudes, and the excitation derived from the error signal calculated in the frequency domain.

In addition to the aforementioned approaches for audio quality measurement, one can find other relevant research with regards to the attempt of combining subjective and objective approaches for audio evaluation. Some of them are based on complementing the subjective assessment with improvements in the tests and their evaluation, and others are trying to implement solutions for automatic and objective quality assurance methods for audio signals.

Wilson and Fazenda [6] evaluate the quality of recordings from two different perspectives: the objective part and the subjective part in order to create a quality-prediction model. A set of objective measures are defined in order to characterize the audio signals, and they are compared among a set of different qualities. The set of objective measures are low-level and high-level features extracted from the audio such as Roll-off, Crest Factor, Low-frequency energy, tempo and emotion. On the other hand, a subjective test is run on a group of 24 listeners with different musical training and skills who are asked to answer about their experience after listening to the set of music samples. The dataset consists of 55 music samples from rock and pop genres with different audio qualities. The hypotheses under tests are mainly three: if there are noticeable differences in quality between samples, if listener training has an influence on perception of quality and if quality is related to one or more objective signal parameters. The results show that dynamics, distortions, tempo and spectral features are correlated with the perceived audio quality. However, it is shown that additional subjective testing is required to increase confidence in their findings. Aspects as increasing the group of listeners or the dataset and extending the set of features would enhance and make more reliable the results obtained.

Kim, Sung, Lee and Park [7] propose an objective method for sound quality evaluation. Due to the inner inaccurate and time/place dependency properties of the subjective listening test, they discuss the implementation of a new objective system using the subjective evaluation items as a database for it. The database is built based on evaluation items as Tonal Balance, Clarity, Spatial and Ambience. These items are obtained through a correlogram of physically measured audio features such as Spectral Deviation, Peakness, and level differences/ripple/time decay of bass, mid and treble frequencies. Afterwards they verify its performance by analyzing the correlation between the subjective factors and the objective ones. The results show that the proposed system is able to offer results equally accurate as the results obtained from subjective listening tests (with a correlation of around 0.93). Therefore, the objective evaluation system seems to have the ability to discriminate quality in a way that could be similar to listener's criteria.

Zhonghua Li et al. [23] propose the first automatic, non-reference audio quality assessment framework to improve live music video online search. They first construct two annotated datasets of live music recordings, from four genres of music (rock, pop, electronic, and country) that tend to have more live recordings of concerts. The first dataset contains 500 human-annotated pieces, and the second contains 2400 synthetic pieces generated by adding noise effects to clean recordings. The noise effects are: amplitude compression and amplification, band-pass filtering, white noise addition and crowd noise addition. Different audio features sets such as low-level acoustic features, MFCCs, and psychoacoustic features are extracted from all recordings. Then, the assessment task is formulated as a ranking problem and they try to solve it using a learning-based scheme called Learning-to-rank (LTR) approach against the feature sets. They later validate the effectiveness of the framework by performing both objective and subjective evaluations. As a result, it has been observed that most video search engines do not take audio quality into consideration when retrieving and ranking results.

2.2. Audio defects and their detection

One can find many different problems that may appear in an audio file. Their characteristics and related detection works are reviewed in the following paragraphs.

a) Clicks and pops

Clicks or pops are sudden, short peaks in the audio file that can result from a variety of causes, including mechanical defects in analogue recordings (caused by a small scratch in the record). They yield unwanted transient signals that generate noises appearing in the audio along with the original signal. Clicks are perceived as a variety of defects ranging from isolated 'tick' noises to the characteristic 'crackle' associated with 78rpm disc recordings. Godsill and Rayner [28] from CEDAR Audio Ltd present different mechanisms for finding them based on modelling the distinguishing features of audio signals and abrupt discontinuities in the waveform (i.e. clicks). They also propose some mechanisms to remove the detected click and interpolating the missing samples in audio, which are applied to the commercial software of the company [3]. In addition, Laney [5] proposes another method based on Wavelet fingerprinting: a tool to recognize errors in the signal by comparing a correct audio image matrix against the matrix of the defect. By taking the continuous wavelet transform of various recordings, a two-dimensional binary display is created from the audio data. The detection algorithm involves a numerical evaluation of the similarity between the known flaw and a given fingerprint.

b) Hum

Hum is a sound associated with alternating current at the frequency of the mains electricity. The fundamental frequency of this sound is usually 50Hz or 60Hz, depending on the local power-line frequency. The sound often has heavy harmonic content above 50–60 Hz. Because of the presence of mains current in mains-powered audio equipment as well as ubiquitous AC electromagnetic fields from nearby appliances and wiring, 50/60 Hz electrical noise can get into audio systems, yielding an undesired low-frequency noise that can be heard in an audio signal. This also creates strong 2nd and 3rd harmonics at 100/120Hz and 150/180Hz. The basic detection and correction method is as the one explained in [49], a high-quality band-stop filter to remove the specific frequency components.

c) Phase issues

Phase issues can result from a distorted or inaccurate stereo image, caused by poor microphone placement or other similar issues. When converting a stereo file to mono, the presence of phase issues can cause the left and right channels to cancel each other out partially or completely. A perfect audio component will maintain the phase coherence of a signal over the full range of frequencies. If some waveforms are “out of phase”, or delayed with respect to one another, there will be some cancellation in the resulting audio, causing the comb filter effect[26], where the spectral components of the signal get altered due to the constructive and destructive interferences. This often produces what is described as a “hollow” sound. How much cancellation, and which frequencies it occurs at depends on the waveforms involved, and how far out of phase they are (two identical waveforms, 180 degrees out of phase, will cancel completely). The human ear is largely insensitive to phase distortion, though it is very sensitive to relative phase relationships within heard sounds. An out-of-phase source has normally missing parts in low and low-mid frequencies. It may also lack a proper spot in the stereo field, and seem to constantly move around without reason. In the most drastic cases, the stereo field will appear to become immensely wide, almost enveloping the listener from behind the ears, but have a huge gap in the centre. Lipshitz, Pocock and Vanderkooy [47] perform a series of experiments to survey the audible consequences of phase nonlinearities in the audio chain. Those experiments are conducted using a flexible system of all-pass networks constructed for this purpose, focusing on the audible effects of mid-range phase distortions. Another good explanation of the consequences of this issue can be found in Mike Senior’s article in *Sound On Sound* magazine[27].

[26] https://en.wikipedia.org/wiki/Comb_filter

[27] http://www.soundonsound.com/techniques/phase-demystified

d) DC offset

DC offset is a shift in the audio, causing the positive and negative parts of the signal not to average to zero and it can limit the dynamic range of an audio file. The cause is normally a fixed voltage offset somewhere in the audio chain before the analog signal is digitized. The voltage may be directly caused by a faulty or low-quality soundcard, or may come from some other device that is attached to the card. It can also occur with some microphones, during the A/D conversion by feedback loops within delay units. Low-frequency distortion may not be audible in the initial recording, but if the waveform is re-sampled to a compressed or lossy digital format like MP3 those distortions may become audible. DC offset can cause audible clicks where audio sections are cut and pasted together, and can cause a click on playback at the start and end of the track, even without editing. DC offset will become worse if the recording is amplified. An example of detection can be seen in [29], where Krochmal, Hamel and Whitecar develop a system based on a method that detects a DC offset in an audio signal provided by an audio processing unit to an audio power amplifier, wherein the audio amplifier provides a clip detect signal back to the audio processing unit. The method begins by sampling the clip detect signal to determine if the clip detect is active. A power level of the audio amplifier is noticed if the clip detect is active, and the power level is compared to a threshold. DC offset can be reduced by a one-pole one-zero high-pass filter. Having the entire waveform, the mean amplitude can be subtracted from each sample and remove the offset.

e) Clipping

If a signal is passed through an electronic device which cannot accommodate its maximum voltage or current requirements, the waveform of the signal can become clipped, containing a large amount of harmonic distortion. The result of it sounds very rough and harsh. In [30], Adler et al. present a novel sparse representation based approach for the restoration of clipped audio signals. The clipped signal is decomposed into overlapping frames and the de-clipping problem is formulated as an inverse problem, per audio frame. The clipping is solved by a constrained matching pursuit algorithm that exploits the sign pattern of the clipped samples and their maximal absolute value.

f) Silence

Silence detects the parts of the audio file at which the audio signal falls to zero, since the signal is perceived as discontinuous. Muhlbauer [4] designs an algorithm using only low-level signal processing methods without any training as in traditional machine learning approaches. The silence detection algorithm finds silent parts in the audio, trying to distinguish between intentional pauses and real defects. The envelope of the signal is estimated and the filtered audio data is thresholded to obtain a vector of silent parts in the signal.

g) Gaps

A gap is a missing part of the audio signal that creates a jump in it, yielding a discontinuity in the waveform. The cause of it may be a scratch, a cut in the recording or digitization error where samples cannot be read and get discarded, resulting in an incomplete file. Kauppinen and Kauppinen [51] develop an algorithm for the correction of gaps of up to several thousand samples in an audio signal. The reconstruction is based on a novel method for time-domain discrete signal extrapolation. The missing or disturbed portion of the audio signal is replaced by a weighted average of signals extrapolated from the areas preceding and following the disturbed part. Impulsive-type errors usually distort the underlying signal irreversibly, and the damaged signal portion does not contain any information of the original signal. In the proposed method the damaged signal samples are not used in computing the replacing samples. The reconstruction method is applied in practice to correct scratches from signals digitized from badly damaged vinyl recordings.

h) Aliasing

Aliasing [30] appears when a very low sample rate is used. It actually occurs when the frequency being sampled is higher than one-half the sample rate (called the Nyquist Frequency by the sampling theorem) and no low-pass filtering at that frequency has been performed. If too low sampling rate is used, the signal can impersonate another signal at lower frequency. Aliasing is well-understood but often overlooked in the coding process. Even the use of some anti-aliasing filters may not prevent aliasing, since a poor design may permit some high frequency components. And since many signals are sampled at very close to Nyquist, design of suitable anti-aliasing filters is difficult. Aliasing introduces additional problems when used in conjunction with compression. It results in the quantization noise introduced into a specific subband creating additional noise at different frequency locations. Thus, frequency components that have negligible effect on audio quality become non-negligible when they are aliased down into frequencies that are more audible. Although there are many ways that aliasing problems can be avoided, it is not guaranteed that all popular audio coders will have implemented these methods. Even if the sample rate of the audio file would offer it, the signal in that file might not utilize the full bandwidth. This could happen when vintage recordings are digitized or the file format uses insufficient bandwidth. Muhlbauer [4] addresses this problem by implementing an algorithm that analyzes the audio and compares it to a configurable, normalized bandwidth. The overall bandwidth is estimated to be the frequency where the cumulative spectral power reaches 80% of the overall spectral power.

i) Lossy compression

Irreversible compression (also known as lossy compression) uses inexact approximations and partial data discarding to represent the content. Most lossy compression reduces perceptual redundancy by first identifying perceptually “irrelevant” sounds, that is, sounds that are very hard to hear. However, some artefacts are created when low bitrates are used: loss of bandwidth (MP3s and AACs show the effects of a brick-wall filter on the upper frequencies, removing high-frequency content above about 16kHz), pre-echoes or birdies (explained in the next section). An extensive explanation is also written in Ian Corbett's article in Sound On Sound Magazine[28]. Liu et al. [32] model the frequently induced audible artefacts and analyze the problematic encoder modules. The severity of the artefacts is investigated through both the subjective and objective measures. In addition, Zhou, Jinglei, et al. [34] develop a method for telling whether the WAV audio has been compressed with the audio encoders (MP3, AAC or OGG) by using the statistical features of phase difference. The proposed method can effectively detect whether the given WAV audio is original or not, and furthermore, it can identify the type of the codec.

j) Birdies

A birdie [30] is a false or phantom signal that appears in the signal due to the aforementioned lossy compression. For low bit rates, slight variations of the masked threshold from frame to frame leads to very different bit assignments. As a result, groups of spectral coefficients may appear and disappear, resulting in the appearance of spurious audio objects. They usually sound like unmodulated carriers (signals with "dead air"). Occasionally they are modulated by clicks, humming sounds, or audible tones. Conventional approaches to overcome the birdie artefact involve use of Low Pass Filters to reduce the amount of signal to quantize. However, they do not eliminate the birdie artefact if the effect is seen in the in-band components. Prakash, Vinod, et al. [36] propose a new algorithm that modifies the bit allocation strategy such that the critical bands are preserved, while still maintaining the perceptual distortion criteria.

k) Pre-echoes

Pre-echoes [30] occur when a signal with a sharp attack begins near the end of a transform block immediately following a region of low energy. When a transient occurs, the perceptual model of the codec will allocate only a few bits to each of the quantizers because a transient signal will spread out over many subbands. It results in unmasked distortion throughout the low-energy region preceding in time the signal attack. One hears only the echo preceding the transient, not the one following because this latter is masked by the transient, that is, a sound is heard before it actually occurs. It is most noticeable in impulsive sounds. Iwai and Lim [37] examine the factors which contribute to a pre-echo, and discuss the method of pre-echo detection and reduction implemented on the MIT Audio Coder (MIT-AC) real-time system. The MIT-AC uses an adaptive window selection algorithm to switch between long and short transform windows. Long windows offer higher coding gains and greater frequency selectivity, while short windows reduce the length of a pre-echo. Due to temporal masking effects, pre-echoes which are sufficiently reduced in duration become inaudible to the human ear. Thus, by switching between long and short windows, the MIT-AC is able to maintain high coding gain while reducing the pre-echo effect.

[28] http://www.soundonsound.com/techniques/what-data-compression-does-your-music

l) Noise bursts

Noise bursts are short-time noisy sounds overlapped with original audio signal. The reason for those defects might be as diverse as MP3 frame errors or other transmission or coding errors. These defects show high energy and almost random distribution across the full bandwidth. Muhlbauer [4] creates a method that, in a frame-wise manner, the audio data is scanned for regions with high energy and almost equally spread spectrum, which are considered to be noise. To detect the spread spectrum the mean value of the spectrum for a frame X of N audio samples is calculated. Another work, by Benjamin and Gannon [31], examine the distortions in the audio at the point of conversion back to the analogue domain, where the effect of digital-to-analogue converters (DACs) introduces errors in the process.

m) Dynamic range compression

A compressed dynamic range (or a low SNR) means a reduction of the energy difference between softer and louder parts of the signal (reduction of the ratio of maximum to minimum loudness in a given audio signal). Several parameters may be involved, including attack, release, delay and slope. Audio compression reduces loud sounds above a certain threshold while leaving quiet sounds unaffected. Fielder [50] examines the criterion yielded by the peak sound levels of music performances combined with the audibility of noise in sound reproduction circumstances for noise-free reproduction of music. The limitations due to microphones, analogue-to-digital conversion (ADC), digital audio storage, low-bitrate coders, DAC conversion, and loudspeakers are reviewed, so that the necessary dynamic range is determined for the most demanding circumstances.

n) Added noise

Added noise to the original signal can result in a very low Signal-to-Noise ratio (SNR). It is referred as random noise and it is normally caused by an undesired background noise while recording or by digitization errors while converting among formats (due to inconsistencies in a low-quality audio converter). There are also some other particular cases of noise, such hum or hiss, that are explained later in this chapter as they follow specific patterns of behaviour and affectation to the signal. Laney's approach [5] tries to identify this defect by using its aforementioned Wavelet fingerprinting technique.

o) Altered stereo image

The stereo image of an audio signal is the perceived spatial locations of the sound source(s), both laterally and in depth. An image is good if the location of the performers can be clearly perceived, and bad if the location of the performers is difficult to locate. When there is a defect in it, (mainly for signals with uncorrelated transient information in each channel, like an applauding audience) the signal may seem to disappear from different locations at different times. In Sound On Sound article[29] by Hugh Robjohns

[29] http://www.soundonsound.com/techniques/processing-stereo-audio-files

the different procedures for creating a correct stereo image plus the creation of diverse effects are reviewed. A patent by Paul F. Bruney [35] presents an audio stereo image recovery system using a set of speakers in conjunction with a conventional stereo system and an auditory interface unit, which provides improved high fidelity playback. The system includes for example, two front high fidelity speakers in the normal stereophonic position, and a second pair of high fidelity speakers placed on an axis defined by the ears of the listener. The amplifier feeds the front speakers directly. The interface unit attenuates the left and right channel signals according to the inter-channel signal level differential, and distributes them in proper proportion to the second pair of speakers to create full stereophonic realism with enhanced depth perception and positioning of the original sound sources. Carlos Avedano describes in [56] a frequency domain framework for source identification, separation and manipulation in stereo music recordings. Based on a simplified model of the stereo mix, a similarity measure between the Short-Time Fourier Transforms (STFT) of the input signals is used to identify time-frequency regions occupied by each source based on the panning coefficient assigned to it during the mix. Individual sources are identified and manipulated by clustering the time-frequency components with a given panning coefficient. After modification, an inverse STFT is used to synthesize a time-domain processed signal. He first describes a cross-channel metric, known as the panning index, that identifies the different sources based on their panning coefficients in the mix. Given the behavior of the panning index error, an adaptive mapping or window function to separate and/or manipulate the individual sources in the mix is proposed. This method is then applied to several problems such as source enhancement and re-panning.

p) Lack of RIAA filtering

Most of the amplitude of a recorded signal comes from its low frequencies. Because of these high amplitudes, if the signal was directly transcribed to fit the constraints of the vinyl format, it would reduce the signal-to-noise ratio unacceptably at many frequencies. To avoid this, the transcribed audio is previously equalized so that low frequencies are attenuated and high frequencies are boosted. On playback, this process is reversed, so that low frequencies are boosted and high frequencies are attenuated. This is known as the RIAA filtering. When the recording RIAA filter is missing in the reproduction and conversion processes (no pre-emphasis) two main things are noticeable: the bass frequencies, with their long wavelengths, are so big and loud that they cause the groove to make really large squiggles. The second thing is that records are noisy. When the playback RIAA filter is missing (pre-amplifier), the low frequencies are attenuated and high frequencies are boosted, yielding and audio signal with mostly scratchy noise and clicks. By the time this thesis has been written, no methods were found to detect the lack of it, so that is why it was addressed in this work. A more detailed explanation of the defect and the detection mechanism can be found in the next chapter.

q) Crosstalk

Crosstalk is the introduction of noise (from another signal channel) caused by ground currents, stray inductance or capacitance between components or lines. It reduces, sometimes noticeably, the separation between channels (e.g., in a stereo or a multichannel audio system). The perceptual effect is the signal bleeding or leaking from one channel to another. Renals et al. [38] focus their detection experiments in four types: local speech, crosstalk plus local speech, crosstalk alone and silence. They describe two experiments related to the automatic classification of audio into these classes. The first experiment attempts to optimize a set of well-known acoustic features for its use with a Gaussian Mixture Model (GMM) classifier. The second experiment used these features to train an ergodic[30] Hidden Markov Model (HMM) classifier. Tests performed on a large corpus of recorded meetings show classification accuracies of up to 96%.

r) Speed-up and time-stretch

When the audio has suffered speed-up or time-stretch alterations [39], the signal is re-sampled at a specified sampling rate but returned using the original sampling rate, which results in a speed-up (or slow-down). This yields a sound file of a given length sped up or slowed down so it will play in a shorter or longer period of time. Making the file longer increases its duration and reduces its tempo and pitch, whereas making the file shorter reduces duration and increases tempo and pitch. This effect can be easily created with the pitch control of a turntable, where the rotation motor can be set to run faster or slower. If this control is changed (not set to 0, typically within a range from -8% to +8%), the tonal components and the tempo of the record are decreased or increased. In latest turntable models (and also in professional CD players), the pitch can be maintaned so that only the tempo is altered (therefore creating the time-stretch effect). In addition, this problem can be caused just by a miscalibration in the turntable nominal rotation speed or also in magnetic tapes incorrectly calibrated when digitizing the recording. By the time this thesis has been written, no methods were found to detect this modification, so that is why it was addressed in this work. An explanation of the defect and the detection mechanism can be found in the next chapter.

s) Wow

Wow is similar to speed-up mechanism, but the re-sampling frequency is time-dependent: it oscillates around the original sampling rate at a specified frequency and amplitude. It mainly happens in audio files converted from magnetic tape recordings, where the playback rotor may not run at a constant speed. The frequency oscillated overtime, as can happen when there’s non-constant speed in record players or tape machines. These are pitch variation defects which may be caused by eccentricities in the playback system or motor speed fluctuations. The effect is a very disturbing modulation of all frequency components. Godsill and Rayner [28] present a novel technique for restoration of musical material degraded by wow and other related pitch variation defects. An initial frequency tracking stage extracts frequency tracks for all significant tonal components of the music. This is followed by an estimation procedure which identifies pitch variations which are common to all components, under the assumption of smooth pitch variation with time. Restoration is then performed by non-uniform re-sampling of the distorted signal. Czyzewski and Maziewski [40] [41] have also worked

[30] Relating to or denoting systems or processes with the property that, given sufficient time, they include or impinge on all points in a given space and can be represented statistically by a reasonably large selection of points.

on mechanisms to reduce the wow effect. In [40] they provide a short overview of the concepts that establish methods based on the tone tracking, on the spectral analysis of audio components, and on non-uniform resampling. In [41] they examine the capacity of non-uniform sampling rate conversion techniques, involving different interpolation methods. Those techniques are: linear interpolation, four polynomial-based interpolation methods and the windowed-sinc based method [14]. The performance of an artificially distorted audio signal, restored using non-uniform resampling, is evaluated on the basis of standard audio defect measurement criteria and compared for all of the aforementioned interpolation methods. The chosen defect descriptors are: total harmonic distortion, total harmonic distortion plus noise and signal to noise ratio.

t) Rumble

Rumble is a low frequency (normally below 50Hz) noise contributed by the turntable of an analogue playback system. It is caused by imperfect bearings, uneven motor windings, vibrations in driving bands or room vibrations that are transmitted by the turntable mounting to the phono cartridge. Bauer [44] analyzes the nature of turntable rumble, with the object of evolving a method for rating the performance quality of turntables with respect to this defect. Spectral distribution of rumble for a typical turntable is shown, identifying the effect of resonances of the tone-arm, and the audibility characteristics of rumble are reviewed, relating it to the equal loudness contour characteristics of the ear and the typical noise in a room. Dolby [42] presents a noise reduction system which is suitable for high-quality audio recording or transmission channels. A special signal component, derived from four band-splitting filters and low-level compressors, is combined with the incoming signal during recording or sending. During reproduction, the additional component is removed in a complementary way and noises acquired in the channel are attenuated in the process. In addition, Donald Knight has patented a hardware rumble eliminator [43] for electric phonographs or in wave-signal receivers.

u) Hiss

Hiss is random additive background noise form of degradation common to all analogue measurement, storage and recording systems. It is present on analogue magnetic tape recordings caused by the size of the magnetic particles used to make the tape. In the case of audio signals, the noise, which is generally perceived as ‘hiss’ by the listener, will be composed of electrical circuit noise, irregularities in the storage medium and ambient noise from the recording environment. Godsill and Rayner [28] expose different methods for hiss reduction, mostly based upon a frequency domain attenuation principle but also using a model-based setting (that is, using machine-learning approaches). Deng, Bao and Liang [45] propose a method based on Modified Discrete Cosine Transform (MDCT): the human auditory model and the parametric soft-thresholding are introduced to the proposed method. A modified median absolute deviation is first adopted to avoid overestimate of noise levels. Next, the Modified Discrete Fourier Transform (MDFT) is constructed using the pre-enhanced MDCT coefficients so the masking threshold and parameters are calculated in MDFT domain. Finally, a parametric soft-thresholding method is employed to attenuate the noise significantly and keep more high-frequency information. Czyzewski [46] implements learning algorithms for the elimination of strong hiss found in old records and of impulse noise affecting transmitted audio signals.

2.3. Commercial implementations

Some of the aforementioned defects are already treated by commercial software applications.

CEDAR's commercial audio restoration applications have different corrective solutions for defects such as clicks and pops[31] or hiss[32]. Other audio and video editors like Audacity[33] or Final Cut[34] already have some plug-ins for audio restoration that can remove issues such as background noise, clicks, pops, hum, hiss, silences, clipping and some other kinds of stationary noise. There are also noise reducers developed by Waves and Izotope[35] that can be used while processing the audio in the digital audio workstation (DAW). Waves provides a package of plug-ins called *Restoration*[36], containing applications for the removal of clicks, hum, DC offset and hiss among other types of noise. It has also stereo imaging processors such as S1 or PS22[37]. Izotope provides an audio editor suite called RX with a set of plugins for audio recovery, where defects such as clicks, pops, breathing, clipping or different equalizations can be corrected. It also has time-stretch and pitch-shifting functionalities. Other options are also other suites such as Bias Soundsoap[38] or the Sony Oxford Restoration Tools called Sonnox Restore[39].

Universal Audio and FXSound have some tools for the stereo image recovery, called *K-Stereo Ambience Recovery*[40] and *DFX Audio Enhancer*[41] respectively. They both have regeneration mechanisms for the the ambience and stereo depth.

Celemony is another company in the audio restoration field that released *Melodyne*[42], a tool capable to apply corrections on the melody and tempo of the song. It detects the tempo, musical scale and the tuning and lets us to correct melody and tempo deviations (such as Wow or time-stretch) using a graphical interface, through its different algorithms: *Melodic* and *Polyphonic*, *Percussive* and *Universal*. Another important system from this company is *Capstan*[43], a Wow and Flutter corrector. The *Capstan* algorithm is able to recognize small amounts of wow and flutter and speed variations within the musical material (tape, wax, vinyl...). It also allows detailed editing so that the corrective curve can be drawn manually. The detection of notes and their deviations is based on the patented *DNA Direct Note Access* technology included also in Melodyne.

[31] http://www.cedaraudio.com/products/duo/declickle.shtml
[32] http://www.cedaraudio.com/products/cedarforpyramix64/cfp64autodehiss.shtml
[33] http://www.audacityteam.org/
[34] https://documentation.apple.com/en/finalcutpro/usermanual/index.html#chapter=59%26section=6%26tasks=true
[35] https://www.izotope.com/en/products/repair-and-edit/rx.html
[36] http://www.waves.com/bundles/restoration
[37] http://www.waves.com/plugins/stereo-imaging
[38] http://www.soundness-llc.com/products/soundsoap5/
[39] https://www.sonnox.com/bundles/sonnox-restore
[40] http://www.uaudio.com/store/mastering/k-stereo-ambience-recovery.html
[41] https://www.fxsound.com/dfx/features.php
[42] http://www.celemony.com/en/melodyne/what-can-melodyne-do
[43] http://www.celemony.com/en/capstan

Below there’s a summary table (Table 1) with some of the aforementioned commercial implementations:

	Audacity	CEDAR	Final Cut	Celemony (Melodyne, Capstan)	Universal Audio	FX Sound	Sony Oxford Restoration Tools	Bias Soundsoap	Waves	Izotope
Clicks, pops	X	X	X				X	X	X	X
Hum	X		X				X	X	X	X
DC offset	X						X	X	X	
Clipping	X						X	X		X
Silence	X									
Pre-echoes		X								
Dynamic range compression	X	X	X						X	X
Added noise	X						X	X	X	X
Altered stereo image					X	X			X	
Speed-up and time-stretch	X			X						
Wow				X						
Rumble	X									
Hiss	X	X					X	X	X	X

Table 1. Commercial software implementations for audio restoration.

2.4. Research goal

As can be seen from the aforementioned works at the time this thesis has been written, there is still room for research on the automatic detection of audio defects. Some of the reviewed audio issues do not have a proper mechanism for their analysis and therefore for their elimination. Many of them are directly related to human auditory perception. They don't just refer to errors in the audio signal; they have intrinsic characteristics of the psychoacoustic behaviour of the human hearing. Issues such as faulty stereo image, narrow dynamics or time-stretching are examples of them. Listening tests are very reliable but also very expensive, time-consuming and sometimes impractical. Because of that, as exposed in section 2.1, recommendations for objective measurement of sound quality have been ultimately proposed. Although the objective methods standardized by ITU (PEAQ for wideband audio signals and PESQ for speech signals) try to imitate the way human listeners perceive sounds using psychoacoustic and cognitive models, they appear not to be suitable for some audio defects, such as gaps, silence, noise bursts, phase issues or the defects addressed in this thesis. That is why current audio restoration and quality analysis systems deserve enhancements in the detection mechanisms in order to be able to find other audio signal issues (such as time-stretch, bad stereo image, undesired filtering).

Therefore, the research goal in this thesis is to investigate two different audio issues not widely addressed apparently in the field and propose mechanisms to find them. First, the characteristics for each defect are analyzed (signal-wise and in the perceptual domain), and then a detection method is proposed for each one. These methods are based on audio signal processing techniques and state-of-the-art machine-learning techniques, which are explained in the following section. And second, a piece of software is built using current digital signal processing and data mining tools. The details of the design are explained in chapters 4 and 5.

3. Problems focused in this thesis: defective RIAA filtering and playback speed changes

In order to understand the defects under study, this section reviews the basic concepts behind the recording and playback technologies of the vinyl music format. First, the process of creating a vinyl record is exposed, and later the necessary equipment to play back such format is reviewed. As will be seen, these both aspects present some drawbacks that need mechanisms of detection. One refers to the pre-process needed in the audio prior to be stored in the medium (the so-called RIAA equalization), and the other refers to the problems when digitizing the audio from the vinyl record to digital formats (altered playback speed).

Since there were no such detection mechanisms at the time this thesis was written, algorithms for this purpose are presented in the next chapters.

3.1. The vinyl technology

a) Creation and characteristics of a vinyl recording

A vinyl gramophone or phonograph record consists of a disc of polyvinyl chloride plastic, engraved on both sides with a single concentric spiral groove in which a sapphire or diamond needle is intended to run, normally from the outside edge towards the centre (see Figure 4).

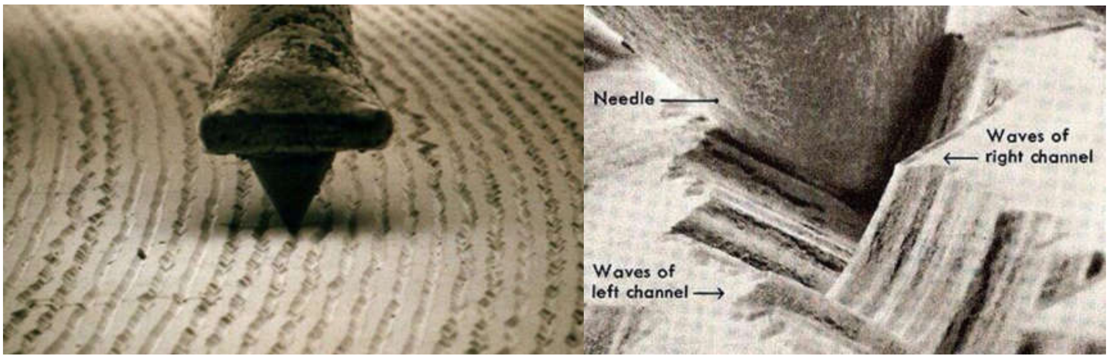


Figure 4. Vinyl grooves under 1000x magnification microscope.

Vinyl record standards follow the guidelines of the RIAA (the Record Industry Association of America). One can find different sizes: normally 12inch, 10inch and 7inch are available. The inch dimensions are not actual record diameters, but a trade name. The record diameters are commonly 30 cm, 25 cm and 17.5 cm in most countries. 12 inch records are often associated with a play speed of 33 1/3 rotations per minute. 7 inch records, on the other hand, are often referred to as 45s because they are most commonly played at a speed of 45 rotations per minute.

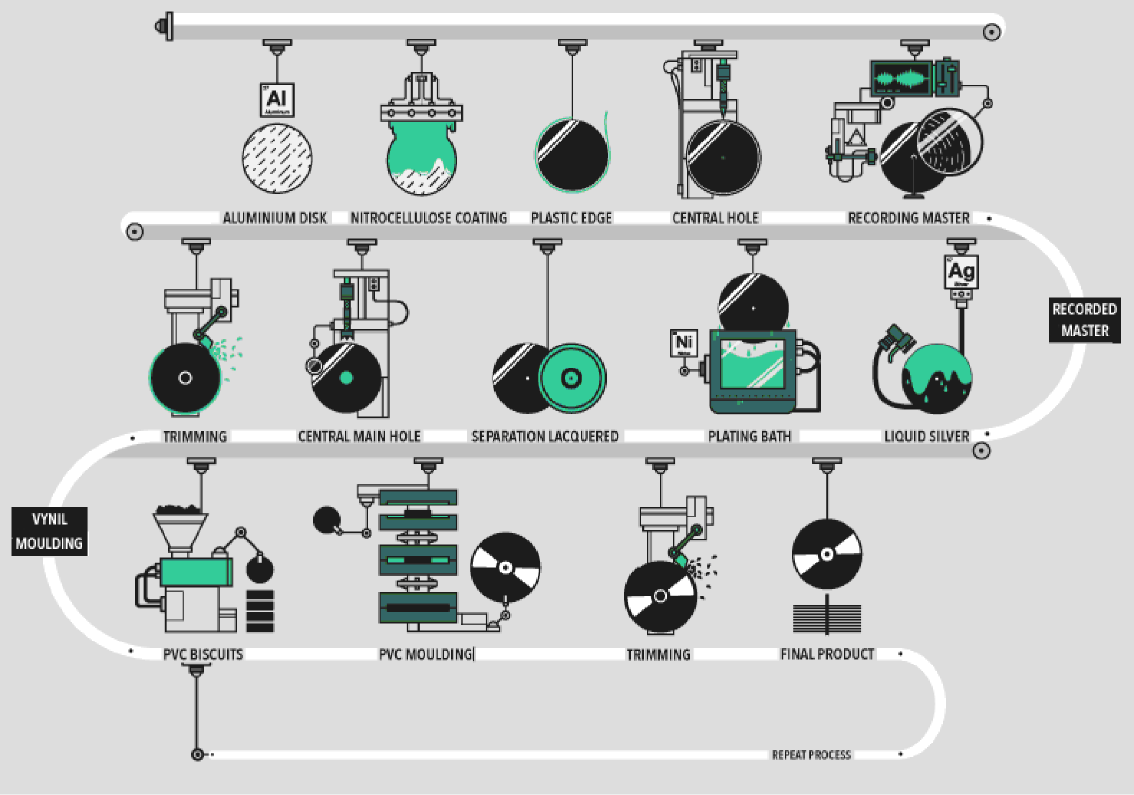


Figure 5. Vinyl production process.

The sound quality and durability of vinyl records is highly dependent on the quality of the vinyl used. Most vinyl records are pressed on recycled vinyl. Formats like 180-gram or 220-gram tend to resist the deformation caused by normal play better than regular vinyl.

b) The playback equipment

In order to play a vinyl record, a turntable is needed in the audio chain, as seen in Figure 6. A complete turntable is comprised of three main different parts: a device for physically turning records, a tone-arm to hold the cartridge, and a cartridge to produce the signal. The cartridge houses the needle, which is the tiny part that actually comes into contact with the record and traces the groove.

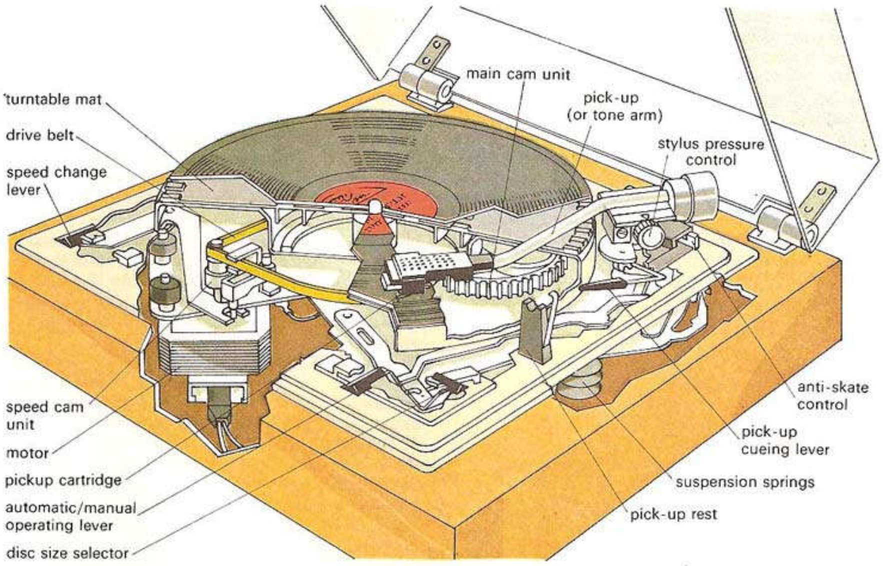


Figure 6. Turntable parts.

A turntable is essentially a mechanical device and its output is a tiny signal of the order of a few mV. The signal is generated by the motion of the stylus as it traces the groove. The mechanism has tiny coils of wire within the cartridge moving relative to a magnetic field as the stylus moves from side to side whilst tracing the record's groove. This voltage needs to be amplified to raise its level to something comparable with CD players etc, so that a typical domestic system amplifier can handle it.

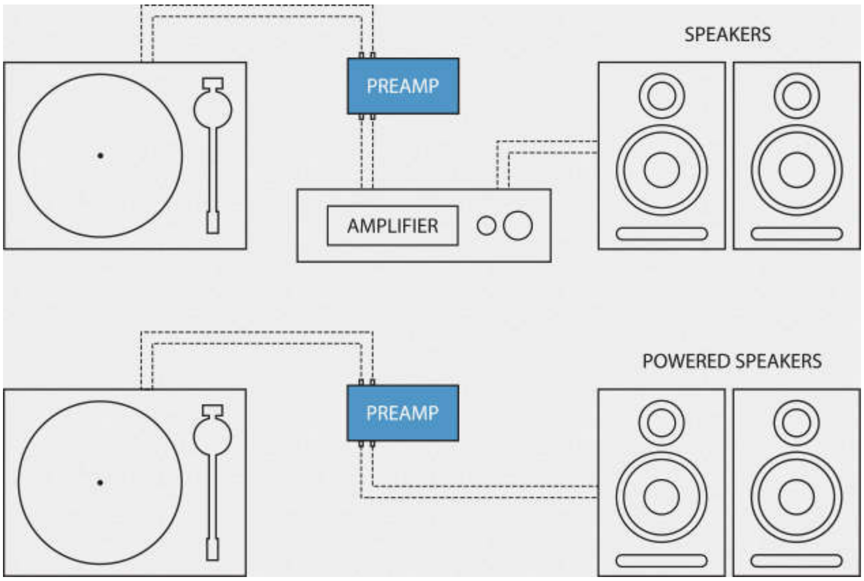


Figure 7. Vinyl playback chain

As can be seen in Figure 7, a turntable with tone-arm and cartridge is required, plus a preamp which is then fed into an amplifier which drives a pair of loudspeakers. Before CD players were invented, and up until the mid 1990s, it was normal for amplifiers to have a preamplifier built-in to them, with one input on the amplifier labelled *phono*. The built-in preamplifier was often compatible with moving magnet cartridges only. The reason behind having a preamplifier in the chain is due to particular characteristics of the vinyl technology that require a pre-process in the vinyl recording, called RIAA equalization. This process is explained in the next section.

Nowadays a lot of loudspeakers which do not appear to require an amplifier have become available. In actual fact, they don not require an external amplifier because the amplifier is built into one of the speakers. Built-in preamplifiers are not usually included with these amplifiers/speakers and an external preamplifier will be required for playing back vinyl.

3.2. Lack of RIAA filtering – The RIAA curve

Most of the amplitude of a recorded signal comes from its low frequencies. Due to their high amplitudes, if they were directly transcribed to fit the constraints of the vinyl format, they would reduce the signal-to-noise ration unacceptably at many frequencies. Bass frequencies would take up too much space on the record (which would reduce available playing time) and treble frequencies would take up so little space that surface noise would attenuate them. Therefore, the transcribed audio is equalized so that low frequencies are attenuated and high frequencies are boosted. On playback, this process is reversed, so that low frequencies are boosted and high frequencies are attenuated. This bass cutting and treble boosting is known as pre-equalisation or pre-emphasis, and on playback the opposite must be done to restore the correct balance, i.e. the bass must be boosted and treble cut.

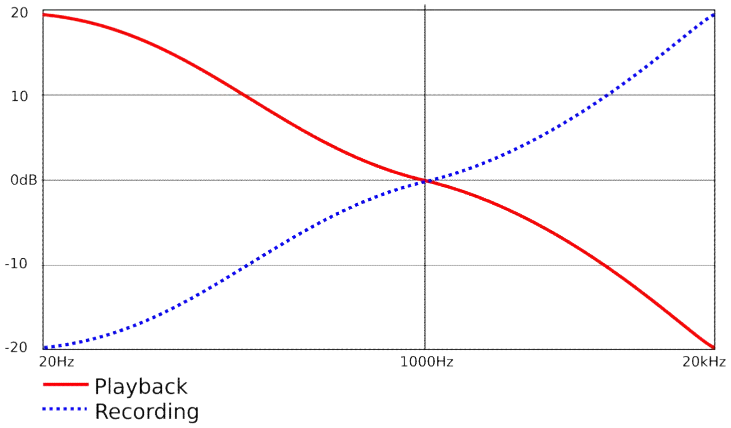


Figure 8. RIAA filtering curves for playback and recording

RIAA set a standard in 1954 for the precise amount of low frequencies cut and high frequencies boost to be applied when records are made, and the converse boost/cut required when records are played back. There were different standards of cut/boost before 1954, each requiring amplifiers with different playback characteristics to achieve accurate reproduction, but the RIAA specification became universally adopted and allowed all record manufacturers and amplifier manufacturers to work with a common standard. The particular equalization curve used is known as the RIAA curve, and its shape can be seen in Figure 8. All modern records are cut to the RIAA standard. The line input connections, designed for tape machines and CD players do not have the

RIAA curve. Every *phono* pre-amplifier must have this playback equalization built into it.

The curve was designed to allow the record pressing plant (and mastering facility) to pre-emphasize higher frequencies, which evened out the size of the grooves making high quality records much easier to manufacture. The curve acts as a sort of equalizer, attenuates low frequencies and amplifies high frequencies (relative to a 1 kHz reference point) in order to achieve the maximum dynamic range for a lateral cut vinyl disc. The grooves in a stereo phonograph disc are cut by a chisel shaped cutting stylus driven by two vibrating systems arranged at right angles to each other. The cutting stylus vibrates mechanically from side to side in accordance with the signal impressed on the cutter. The resultant movement of the groove back and forth about its centre is known as groove modulation. The amplitude of this modulation cannot exceed a fixed amount or cutover occurs. Cutover, or overmodulation, describes the breaking through the wall of one groove into the wall of the previous groove. Since low frequencies cause wide undulations in the groove, they must be attenuated to prevent overmodulation. At the other end of the audio spectrum, high frequencies must be amplified to overcome the granular nature of the disc surface acting as a noise generator, thus improving signal-to-noise ratio.

When the recording RIAA filter is missing (no pre-emphasis) two main aspects are noticeable: the bass frequencies, with their long wavelengths, are so big and loud that they cause the groove to make really large squiggles. And secondly, the records are noisy. If the playback RIAA filtering is missing (no preamplifier) the recording sounds with very low loudness overtime, a very “weak” sound.

3.3. Playback speed in vinyl recordings - The pitch control

A variable speed pitch control is a mechanism on an audio device such as a turntable, tape recorder, or CD player that allows to deviate from a nominal speed (see Figure 9). Analog pitch controls vary the voltage being used by the playback device, whereas digital controls use digital signal processing to change the playback speed or pitch. A typical DJ deck allows the pitch to be increased or reduced by up to 8%. In the turntable mechanism, it is achieved by increasing or reducing the speed at which the platter rotates. Because the pitch of a sound is directly related to its frequency, lowering the frequency will also lower the pitch. This is what pitch control does: it slows down (or speeds up) the platter rotation by a certain amount. This reduces (or increases) the number of times the record will spin past the needle in one second and therefore the pitch will drop (or rise). In addition, the tempo (measured in beats per minute, or BPM) of the track also drops.

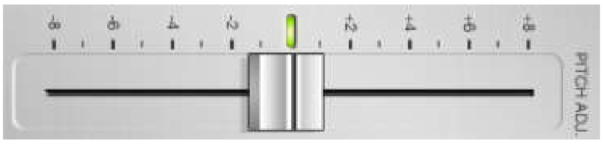


Figure 9. Detail of the pitch control of a turntable

When digitizing recordings from those legacy formats, it is important to be sure the speed the recording is played at is the nominal one. Otherwise, the digitized audio file will not be as equal as the original one: the playback speed will get altered. This problem can appear due to a different position of the pitch control when digitizing, but it can also be caused by problem in the rotor of the turntable, so that nominal speed is not reached or overpassed due to a low-quality or faulty equipment.

The purpose of this thesis is to detect if an audio recording was digitized in such abnormal conditions, either due to the pitch control not being at nominal value (0%), or due to a defect on the turntable's rotor mechanism.

4. Method

4.1. Materials

a) The datasets

a) Dataset for RIAA detection

The dataset used for testing and in the final implementation consists on 1000 files split in 10 different music genres (100 files each): *Classical*, *Electronic-Dance*, *Experimental*, *Jazz*, *Lounge-Downtempo*, *Metal-Industrial*, *Pop-Rock*, *Rap-Dubstep*, *Reggae-Ska*, *Soul-Funk*. This selection of music genres tries to cover a large number of music spectrum.

For each original file, the corresponding "RIAA-less" counterpart has been created, by filtering the inverse response of the RIAA (that is, filtering with the recording curve as seen in Figure 10):

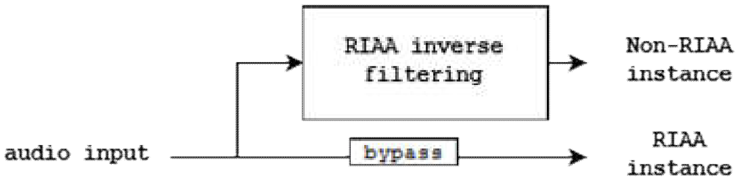


Figure 10. RIAA and Non-RIAA instances creation.

In total, the number of instances is 2000: 1000 with correct RIAA and 1000 without the RIAA filtering applied.

b) Dataset for altered playback speed.

The dataset in this case consists on 100 files split in 8 different music genres: *Classical*, *Electronic-Dance*, *Jazz*, *Lounge-Downtempo*, *Pop-Rock*, *Rap-Dubstep*, *Reggae-Ska* and *Soul-Funk*.

For each original file, the corresponding altered counterparts are created (as shown in Figure 11), by modifying the speed at the different percentages (pitch ranges) explained before: +/-1%, +/-2%, +/-5% and +/-8%. As a result, a total the of 800 instances is created and different speeds.

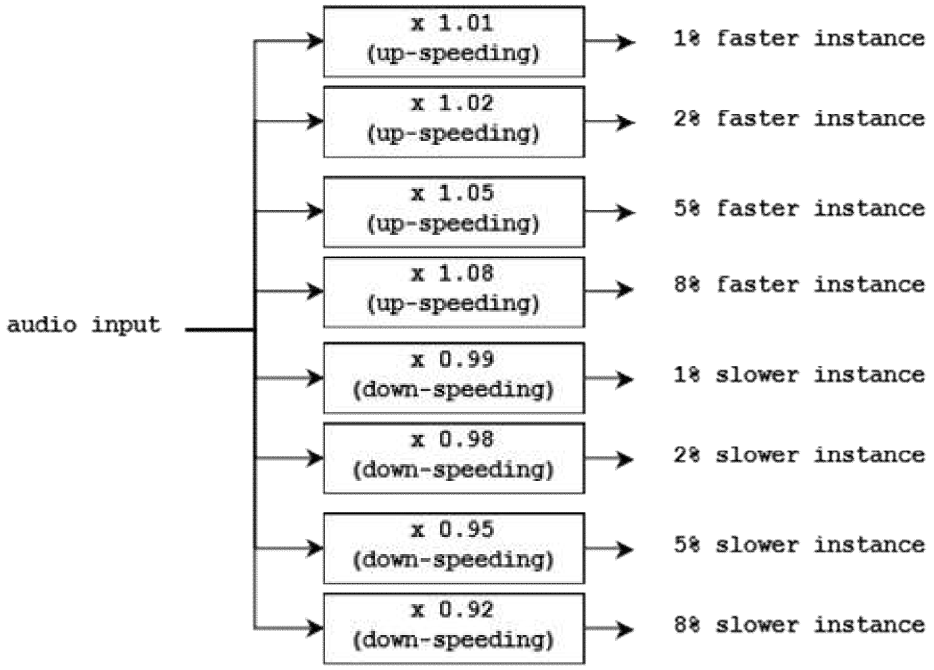


Figure 11. Different altered instances creation from the original input

b) Software tools

a) MA-Toolbox

The Music Analysis Toolbox (MA-Toolbox) [13] is a collection of functions for MATLAB[44] created by Elias Pampalk. It contains functions to analyze and compute similarities on audio. The implemented measures focus on aspects related to timbre and periodicities in the signal. In this case, the function *ma_sone()* has been used in order to get the bark spectrum of each file under test. The bark spectrum (bark scale) is explained in the next section.

b) *WEKA* and Python-weka-wrapper

WEKA or "Waikato Environment for Knowledge Analysis" [3] is a collection of machine-learning algorithms for data mining tasks developed by the Wakaito University from New Zealand. It is open source software issued under the GNU General Public License. is published under the GNU General Public License. It is implemented in Java, and the algorithms can be applied either directly to a dataset or called from Java code. Weka contains tools for data pre-processing, classification, regression, clustering, association rules, and visualization.

WEKA uses a proprietary file format called Attribute-Relation File Format (ARFF)[45], which is used to store all information needed for a dataset and the data mining tasks to be performed with it. It is structured in two main parts: the header (containing the name

[44] http://uk.mathworks.com/products/matlab/index.html

[45] http://weka.wikispaces.com/ARFF

of the dataset the attribute types and the available classes of data) and the data of each instance following the rule in the example below:

```
@RELATION riaa
@ATTRIBUTE ratio_0_9 NUMERIC
@ATTRIBUTE ratio_1_9 NUMERIC
@ATTRIBUTE ratio_2_9 NUMERIC
@ATTRIBUTE ratio_21_9 NUMERIC
@ATTRIBUTE ratio_22_9 NUMERIC
@ATTRIBUTE ratio_23_9 NUMERIC
@ATTRIBUTE ratio_4_0 NUMERIC
@ATTRIBUTE ratio_5_0 NUMERIC
@ATTRIBUTE ratio_6_0 NUMERIC
@ATTRIBUTE class {riaa_ok, riaa_ko}
@DATA
0.46483529, 0.66772020, 0.80625808, 0.66896194, 0.55944031, 0.40802291,
1.91796827, 1.98463261, 2.09811139, riaa_ok
0.64432651, 0.87849170, 1.02301598, 0.58938521, 0.46214759, 0.31966868,
1.78075182, 1.80180609, 1.77807617, riaa_ok
```

This file is created when extracting the parameters from the signal and later used to train the system, as explained in the procedure of the algorithm in section 4.2.

In order to use WEKA from python code, the Python-weka-wrapper[46] has been used. The *python-weka-wrapper* package makes it easy to run Weka algorithms and filters from within Python, as it offers access to Weka API using wrappers around JNI calls by a bridge from Java code. This wrapper has been included in the classification process between for both detection algorithm, explained in the next section.

c) FFMPEG

FFmpeg[47] is a free software project that produces libraries and programs for managing multimedia data, originally developed by Fabrice Bellard. FFmpeg includes *libavcodec*, an audio/video codec library, *libavformat*, an audio/video container multiplexing and demultiplexing library, and the ffmpeg command line program for transcoding multimedia files. FFmpeg is published under the GNU General Public License, and can be used under most operating systems, including Linux, Mac OS X, Microsoft Windows, Android or iOS.

This tool has been used to filter the original files with the recording (RIAA inverse) filtering in batch mode in order to create the dataset.

d) SoX

SoX[48] stands for Sound-eXchange and is a cross-platform (Windows, Linux, MacOS X, etc.) command line utility that can convert various formats of computer audio files in to other formats. It can also apply various effects to these sound files, and, as an added

[46] http://pythonhosted.org/python-weka-wrapper/

[47] http://ffmpeg.org/

[48] http://sox.sourceforge.net/sox.pdf

bonus, SoX can play and record audio files on most platforms. In this case, SoX has been used to prepare the instances with different playback speeds.

4.2. Detection algorithms

The algorithms to detect the aforementioned defects base their design in two main concepts: the bark decomposition of the signal spectrum and the use of machine-learning algorithms such as Decision Trees, Support Vector Machines and Dynamic Time Warping. They are explained in the following sections so the whole procedure can be easily understood.

a) Bark bands - The bark scale

The Bark scale is a psychoacoustical scale proposed by Eberhard Zwicker [54] to model the frequency resolution of the human cochlea (that is, the bandwidth of the auditory filters). It is named after Heinrich Barkhausen who proposed the first subjective measurements of loudness. It is a frequency scale on which equal distances correspond with perceptually equal distances by the human auditory system. Above about 500Hz this scale is more or less equal to a logarithmic frequency axis, and below 500Hz it becomes more and more linear.

The Bark scale is calculated using the formula below, that lets changing from frequency scale to barks scale:

$$Bark = 13\arctan(0.00076 f) + 3.5\arctan((f / 7500)2)$$

The scale ranges from 1 to 24 and corresponds to the first 24 critical bands of hearing, that is, the regions in basilar membrane where there is a distinction in sound amplitude (see Table 2 below).

Band number	**Central Frequency (Hz)**	**Cut-off Frequency (Hz)**	**Bandwidth (Hz)**
		20	
1	60	100	80
2	150	200	100
3	250	300	100
4	350	400	100
5	450	510	110
6	570	630	120
7	700	770	140
8	840	920	150
9	1000	1080	160
10	1170	1270	190
11	1370	1480	210
12	1600	1720	240
13	1850	2000	280
14	2150	2320	320

15	2500	2700	380
16	2900	3150	450
17	3400	3700	550
18	4000	4400	700
19	4800	5300	900
20	5800	6400	1100
21	7000	7700	1300
22	8500	9500	1800
23	10500	12000	2500
24	13500	15500	3500

Table 2. Bark scale

These bands have been directly measured in experiments on the threshold for complex sounds, on masking, on the perception of phase, and on the loudness of complex sounds. In all these cases the critical band seems to play an important role. The critical bands have a certain width, but their position on the frequency scale is not fixed, it can be changed continuously, perhaps by the ear itself. Therefore, the important attribute of the Bark scale is the width of the critical band at any given frequency, not the exact values of the edges or centres of any band.

The Bark scale can therefore be a compact representation of the audio spectrum when calculating the spectrum's energy for each band. This principle is used in the RIAA detection algorithm as it is described later.

b) Machine-learning methods: C4.5 decision tree, Support Vector Machines and Dynamic Time Warping

Machine-learning [26] comprises the techniques that let performing data mining, that is, the process of discovering patterns in data [19]. Data mining is defined as the extraction of meaningful information previously unknown and potentially useful from the data. This allows the analysis and the obtention of knowledge from it, so that a model of learning can be created. Later, this model can be applied to other data and predict or classify it. One can find many types of techniques, so that they can be applied depending on the nature of the data and according to the task to be performed: prediction, classification, clustering, etc.

Normally, the process of data mining takes the following steps:

1- Selection of variables: independent (attributes to be analyzed) and dependent (the value to be predicted or the class to be assigned.
2- Data pre-processing (optional) in case data needs some arrangement such us removal of null, missing or inconsistent values.
3- Knowledge model creation from training data.
4- Knowledge extraction from observed data patterns.
5- Model evaluation.

Given the type of data used in this thesis, a decision tree and support vector machines algorithms have been chosen, as they appear to yield great results on classification purposes [19]. Their fundamentals are explained in the next sections.

a) C4.5 Decision tree

A decision tree, as its name indicates it, is an algorithm where each branch is a decision rule depending on only one attribute: if an attribute a has value X then follow branch T (see Figure 12). Once the tree is built, an instance is classified by starting with the rule at the root, until a leaf is reached. Leaves are marked with the class, into which the instance is classified. C4.5 is a particular implementation of this type of algorithms, developed by Ross Quinlan [15].

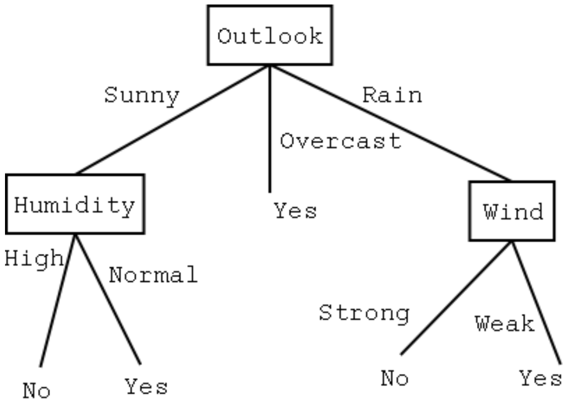


Figure 12. Example of a Decision Tree
Leaves are created when the node has different possible values. If the no more decisions are needed to assign the class, the branch ends with a leaf containing the class.

The tree is built recursively (starting at the root): if all remaining instances are of the same class c, the current node becomes a leaf, otherwise the current node is expanded by choosing the attribute for which information gain [2] is maximal and a subtree is created for each of its possible values.

b) Support Vector Machines (SVM)

A Support Vector Machine (SVM) is a two-class learning algorithm that performs linear separation between the instances and it can be extended to n classes [17]. The main idea behind the SVM is that the linear separation task is not done on the n attribute values directly, but after transforming the n-dimensional attribute space into a so-called feature-space of higher dimensionality. Depending on the mapping function (known as kernel function), many problems that are not linearly separable in the attribute space get linearly separable in feature space).

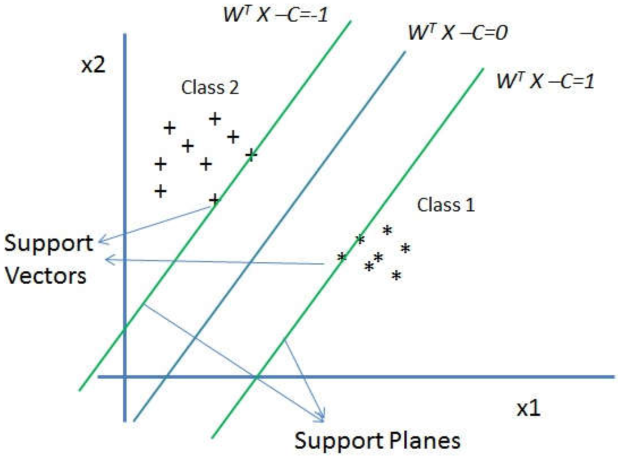


Figure 13. Example of a Support Vector Machine to separate 2 classes

The optimal decision boundary is the hyperplane which separates the two classes and that has maximal distance to the closest data points from each class (this hyperplane is called *maximal margin hyperplane*, as can be seen in Figure 13). Once the maximal margin hyperplane is computed, its position (and hence the computed model) depends only on the points that are closest to it. These points are called *support vectors* as they can be seen as the most informative data points.

c) Dynamic Time Warping

Dynamic time warping (DTW) is an algorithm for measuring similarity between two temporal sequences which may vary in speed. In general, DTW is a method that calculates an optimal match between two given sequences (e.g. time series) with certain restrictions [48] [55]. The sequences are "warped" non-linearly in the time dimension to determine a measure of their similarity independent of certain non-linear variations in the time dimension, yielding a distance quantity between two given sequences, as seen in Figure 14:

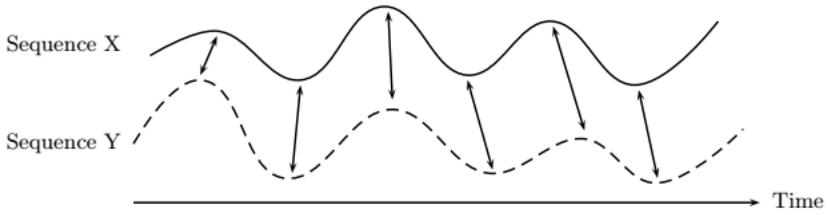


Figure 14. Time alignment of two time-dependent sequences.
Aligned points are indicated by the arrows.

The optimal path (called the *cost matrix*) of these alignments is calculated and as a result a global DTW distance is obtained (the total cost).

This sequence alignment method is often used in time series classification, and it used in this thesis for calculating de distances between the reference file and the modified instances in the altered playback speed algorithm, explained later in this chapter.

c) Procedure for RIAA detection algorithm

The algorithm to detect if an audio file has been correctly converted using the RIAA pre-emphasis can be seen in the Figure 15 below:

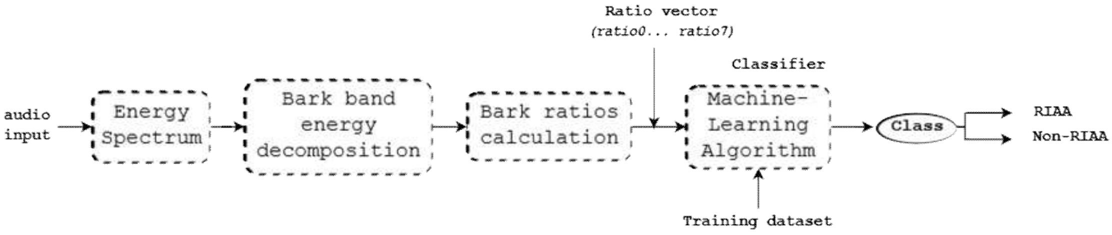


Figure 15. Block diagram of the RIAA detection algorithm

First, the bark-band decomposition spectrum is obtained from the audio file under test. In order to do that, the energy spectrum is calculated along the frames, and the mean value for all the frequencies is obtained (as seen in figure 16):

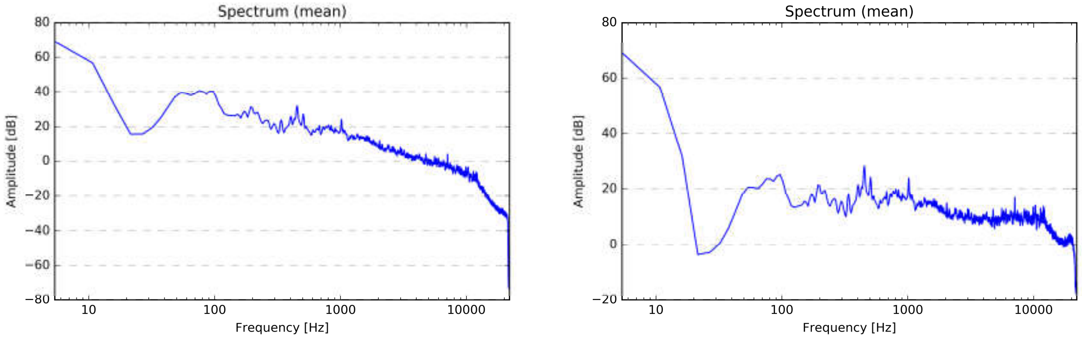
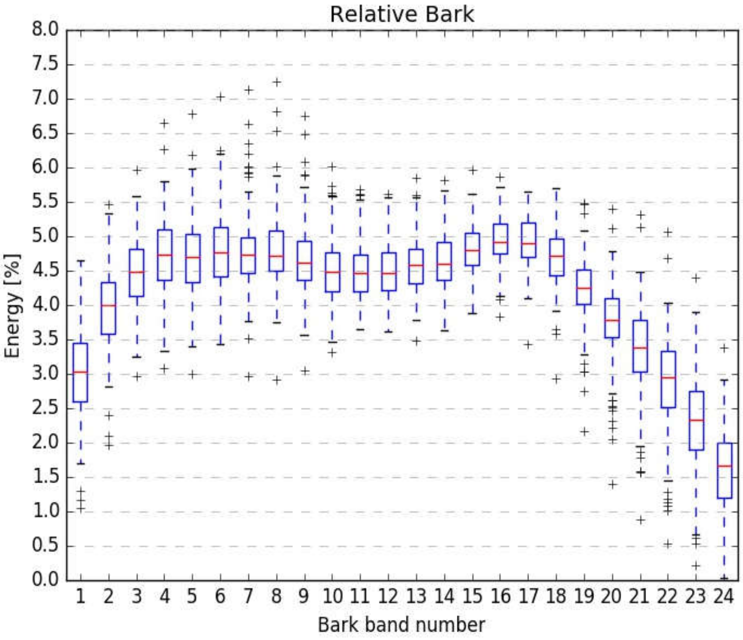


Figure 16. Example of correct RIAA spectrum (left) and incorrect RIAA spectrum (right)

As can be seen above, the boosting of HF component is clearly noticed: the upper half of the spectrum has shifted upwards (more amplitude) and the lower has been shifted downwards (less amplitude).

After that, the mean spectrum is decomposed in the 24 critical bands in the bark scale and the energy percentage is calculated, as shown in Figure 17:



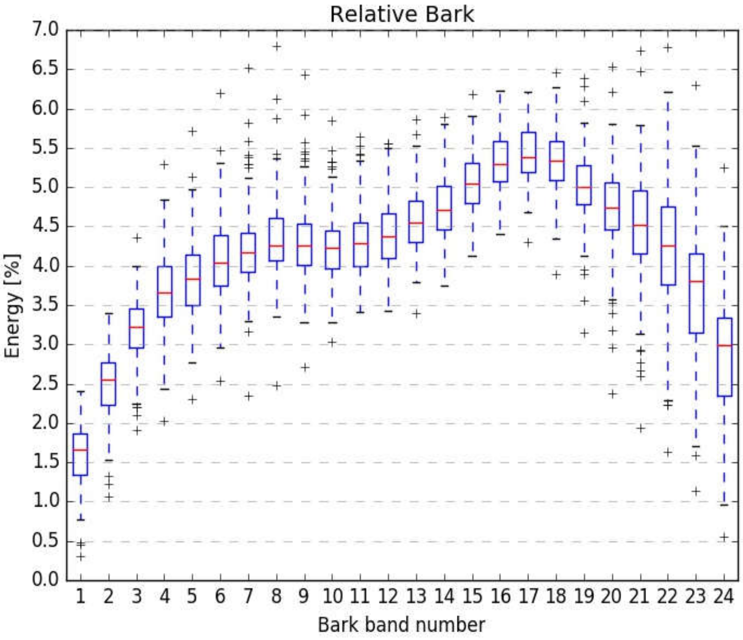


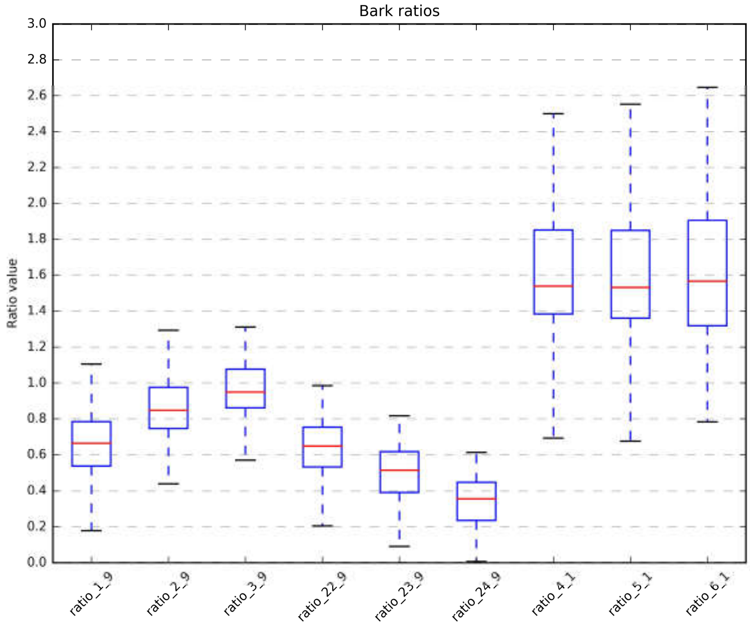
Figure 17. Correct RIAA spectrum (up), Incorrect RIAA spectrum (down) using boxplots for a subset of 100 files. This method allows to see the energy percentage distribution per each bark band for all analyzed instances.

According to Figure 17 , the difference between a correct RIAA processing and the lack of it is clearly noticeable. However, the relation between low-frequency and mid-frequency components on one hand, and the relation between mid-frequency and high-frequency components on the other, is indeed more relevant to see the spectrum behaviour. For that, a vector of bark ratios is calculated.

Bark ratios are obtained taking into account the aforementioned relations between low and high frequencies against the mid ones (where no attenuation is performed by the RIAA filtering, so the original and the RIAA-less signal are the same). Obtaining such relations can provide an idea of the attenuation suffered in the most altered frequencies for the RIAA curve.

The ratios calculated are 9 in total: ratio_1_9, ratio_2_9, ratio_3_9, ratio_22_9, ratio_23_9, ratio_24_9, ratio_1_4, ratio_1_5, ratio_1_6. That is, for example in ratio 1_9, the relation of energy for band 1 (around 100Hz) and the band 9 (energy around 1000Hz) is obtained. If the ratio has a big variation when calculated from RIAA-less signal to the RIAA signal, it could be said that low-frequencies have suffered an alteration (big ratio value).

The values of those ratios for the given dataset are shown in Figure 18:



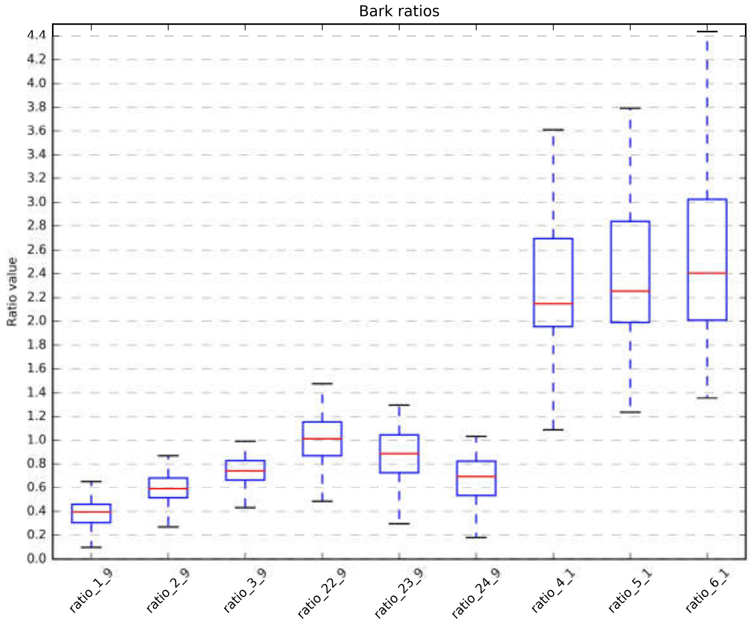


Figure 18. Boxplot of bark ratios for a subset of correct RIAA instances (up) and incorrect RIAA instances (down). The interval of values for each of the ratios is shown in a box, the red line being the mean value and the blue square containing the majority of the instances.

These ratios are really useful for the RIAA problem under study, since the spectrum is altered in an almost linear manner, as can be seen by the RIAA curves (in Figure 8). Therefore, the ratios between boundaries in the spectrum and the bark band number 9 could be highly discriminant (where the spectrum is not altered as the curve is not filtering: it crosses 0dB at 1000Hz, and bark band number 9 contains the information for that frequency interval). In addition, the ratios 4-1, 5-1 and 6-1 are calculated due to being bands where instruments usually have their fundamental frequencies, and they are attenuated when RIAA filtering is not applied.

	Interval		**Central**	
	RIAA_OK	**RIAA_KO**	**RIAA_OK**	**RIAA_KO**
ratio_1_9	(0.55, 0.80)	(0.30, 0.45)	0.65	0.40
ratio_2_9	(0.75, 1.00)	(0.50, 0.70)	0.85	0.60
ratio_3_9	(0.85, 1.10)	(0.65, 0.85)	0.95	0.75
ratio_22_9	(0.55, 0.75)	(0.85, 1.15)	0.65	1.00
ratio_23_9	(0.40, 0.60)	(0.70, 1.05)	0.50	0.90
ratio_24_9	(0.25, 0.45)	(0.55, 0.825)	0.35	0.70
ratio_4_1	(1.40, 1.85)	(1.9, 2.65)	1.5	2.1
ratio_5_1	(1.40, 1.85)	(2.0, 2,75)	1.5	2.2
ratio_6_1	(1.30, 1.90)	(2.0, 3.0)	1.6	2.4

Table 3. Bark ratios for Non-RIAA and RIAA instances from boxplots of figure 15.
Interval bark ratio values from boxplots for both RIAA-KO and RIAA-OK are shown.
Also the central value for each ratio is displayed for the whole set of instances.

As can be seen from Figure 17 and 18 and Table 3 above, it seems to be quite clear where thresholds can be set. However, the use of a machine-learning algorithm lets the thresholds to be set by the data and not in an absolute manner. Given an unknown instance, its resulting ratios vector is then classified using a machine-learning algorithm, trained by the aforementioned dataset, so that the class (*RIAA_OK*, *RIAA_KO*) is yielded as output.

d) Procedure for altered playback speed detection algorithm

The proposed algorithm to determine if a given audio file has suffered an alteration of the playback speed can be seen in the Figure 19 below:

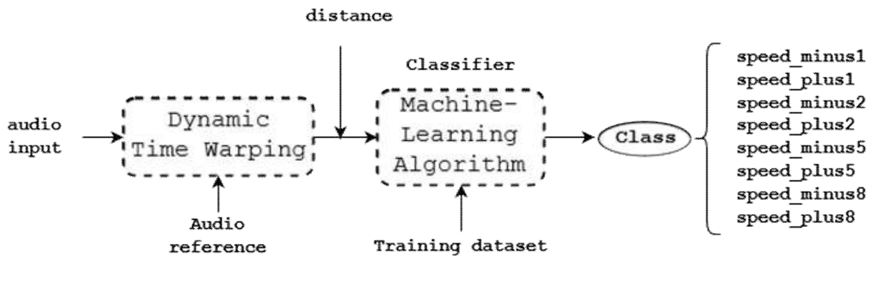


Figure 19. Block diagram of the altered playback speed detection algorithm

Given the audio file under test, the distance against the reference file is performed (that is, the same file but at nominal speed), using Dynamic Time Warping (DTW) technique. As a result, a distance value is obtained. Afterwards the instance under test is classified according to the distance value using a machine-learning algorithm, that lets to assign a class and therefore determine the amount of speed variation the audio file has suffered. If the distance is 0, it means the instance is at nominal speed, otherwise it is classified into one of the following classes: *speed_minus1*, *speed_plus1*, *speed_minus2*, *speed_plus2*, *speed_minus5*, *speed_plus5*, *speed_minus8*, *speed_plus8*.

In Figure 20 it can be seen how the distance values move depending on the amount of alteration (that is, the percentage of up-speed or down-speed):

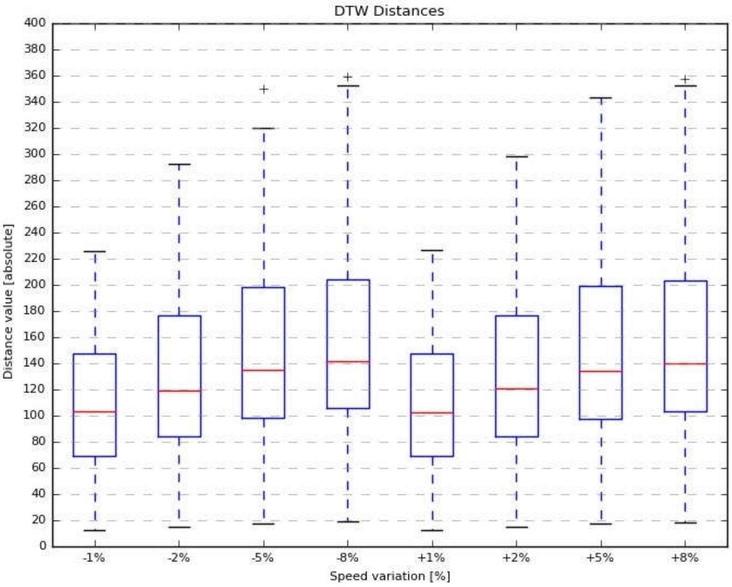


Figure 20. Boxplot of absolute distance values for different speed variations.
The interval of values for each of the speeds are shown in a box, the red line being the mean value and the blue square containing the majority of the instances

The log value of the distance is also considered in order to see if there is any trend in the increment of speed following such logarithmic[49] behaviour. As seen in Figure 20, the amount of distance increment between speeds seems to decrease as the speed alteration gets higher. Using the log of can may let us separate the distances more so that classification process could be easier for the algorithm. However, according to Figure 21 below, it doesn’t seem to be the case, as it follows the same trend as using absolute distances:

[49] https://en.wikipedia.org/wiki/Logarithm

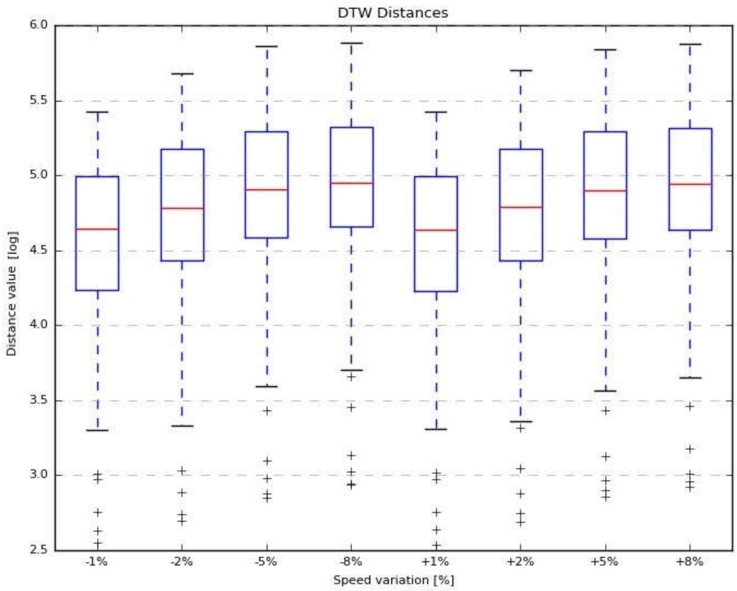


Figure 21. Boxplot of logarithmic distance values for different speed variations. The interval of values for each of the speeds are shown in a box, the red line being the mean value and the blue square containing the majority of the instances

From the Table 4 below, the comparison for absolute and logarithmic distance values is exposed:

	Interval		**Central**	
	Absolute value	**Logarithmic value**	**Absolute value**	**Logarithmic value**
-1%	(70, 148)	(4.25, 5.00)	105	4.65
-2%	(85, 178)	(4.4, 5.15)	120	4.75
-5%	(100, 200)	(4.6, 5.25)	135	4.8
-8%	(108, 205)	(4.7, 5.3)	140	4.9
+1%	(70, 148)	(4.25, 5.00)	105	4.65
+2%	(85, 178)	(4.4, 5.15)	120	4.75
+5%	(100, 200)	(4.6, 5.25)	135	4.8
+8%	(105, 205)	(4.7, 5.3)	140	4.9

Table 4. Absolute and logarithmic distance values from boxplots of figures 20 and 21. Interval distance values from boxplots for all different percentages of speed alteration are shown. Also the central value for each distance is displayed for the whole set of instances.

According to Table 4, at first sight it does not seem to be any difference in the interval or central values when comparing the same speed percentages, as they are clearly symetric either for the up-speed and down-speed counterparts (same amount of variation when decreasing or increasing the same percentage of playback speed).

Looking at the central values of distance, the increment of distance does not seem to follow any regular pattern either when using absolute or logarithmic. In absolute values, when doubling from 1% to 2%, the increment is 15, the same as when increasing from 2% to 5% (that is, more than the double), and later, when going from 5% to 8%, the increment gets reduced by 3 (5). For logarithmic values, the sequence is 1 from 1% to 2%, 0.5 from 2% to 5%, and 1 again from 5% to 8%.

e) Evaluation

a) 10-Fold Cross-validation

N-Fold cross-validation is a method to evaluate the algorithm under a large dataset. In order to know the accuracy of it, the algorithm is tested against a large amount of files to see how it would behave in real data. Although it is impossible to test it for all existing data, using a large dataset can give a proper idea about the suitability of the system for real-world situations.

The idea behind this method is the following: The dataset is split randomly in n parts of the same size. Every iteration, one of the parts is used for testing, and the other n-1 parts are used for training. Therefore every part for testing is different in each iteration. The results for every round are collected and weighted against the whole dataset, so that an estimated global accuracy value is obtained.

The standard mode for this method is the so-called stratified 10-fold cross-validation, that is: the dataset is split in 10 parts of equal proportion from the whole dataset. As can be seen in Figure 22, 1 part is used for testing and the remaining 9 are used for training. As a result, the training process is performed 10 times and the 10 estimated accuracies are weighted into a global one.

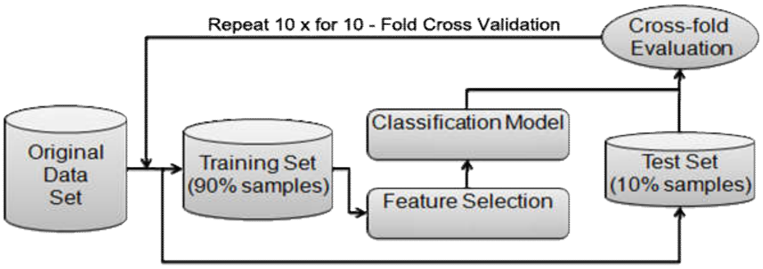


Figure 22. 10-Fold cross-validation procedure

Numerous experiments on different machine-learning algorithms and different datasets show that 10 is usually the value that fits best for Music Information Retrieval purposes [16] [19].

b) Precision, recall and F-Measure

The accuracy for each algorithm is given by four different measures: the Accuracy, the Precision, the Recall, and the F-Measure defined by the formulas below:

$$\text{Precision} = \frac{tp}{tp + fp} \qquad \text{Recall} = \frac{tp}{tp + fn}$$

$$F = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \qquad \text{Accuracy} = \frac{tp + tn}{tp + tn + fp + fn}$$

Where:
- tp are *True Positives*: correctly identified.
- fp are *False Positives*: incorrectly identified.
- tn are *True Negatives*: correctly rejected.
- fn are *False Negatives*: incorrectly rejected.

These measures are standards for performance evaluation in machine-learning algorithms field [27].

5. Results

5.1. Results for RIAA detection

The algorithm was tested using the default parameter values for each classifier, except the C4.5, where different values of M (number of instances to create a leaf) have been used in order to see the effect of this parameter in the accuracy of the results. The value of this parameter is important as it will determine the complexity of the final classifier (the compactness of the tree [19]), and can improve predictive accuracy by the reduction of overfitting[50].

Below, the different accuracy values are provided given by the different values of M in C4.5:

M=2	M=5	M=10	M=11	M=12
94.22%	94.37%	94.82%	94,67%	94,42%

Table 5. Correctly classified instances by C4.5 depending on the number of instances per leaf (M).

As seen in Table 5, the value M=10 seems to reach the best results in the global dataset, since lower and higher values than this decrease the accuracy of the model.

The mechanism of the decision tree for the given data can be understood by looking at the schema created by the algorithm (see Figure 23). In the example below, the tree is built using M=10 into the global dataset (all 2000 instances):

```
ratio_1_9 <= 0.560713
|   ratio_22_9 <= 0.619036
|   |   ratio_2_9 <= 0.533008: riaa_ko (118.0/16.0)
|   |   ratio_2_9 > 0.533008
|   |   |   ratio_1_9 <= 0.469211
|   |   |   |   ratio_23_9 <= 0.265462: riaa_ok (82.0/10.0)
|   |   |   |   ratio_23_9 > 0.265462: riaa_ko (44.0/7.0)
|   |   |   ratio_1_9 > 0.469211: riaa_ok (106.0/12.0)
|   ratio_22_9 > 0.619036: riaa_ko (781.0/13.0)
ratio_1_9 > 0.560713
|   ratio_23_9 <= 0.673384: riaa_ok (791.0/12.0)
|   ratio_23_9 > 0.673384
|   |   ratio_1_9 <= 0.681671: riaa_ko (44.0/1.0)
|   |   ratio_1_9 > 0.681671
|   |   |   ratio_4_1 <= 1.3434: riaa_ok (13.0/3.0)
|   |   |   ratio_4_1 > 1.3434: riaa_ko (11.0/3.0)
```

Figure 23. Output yielded by WEKA framework.

[50] Overfitting occurs when a model is too complex (such as having too many parameters relative to the number of observations). The model has poor performance and as it overreacts to minor fluctuations in the data.

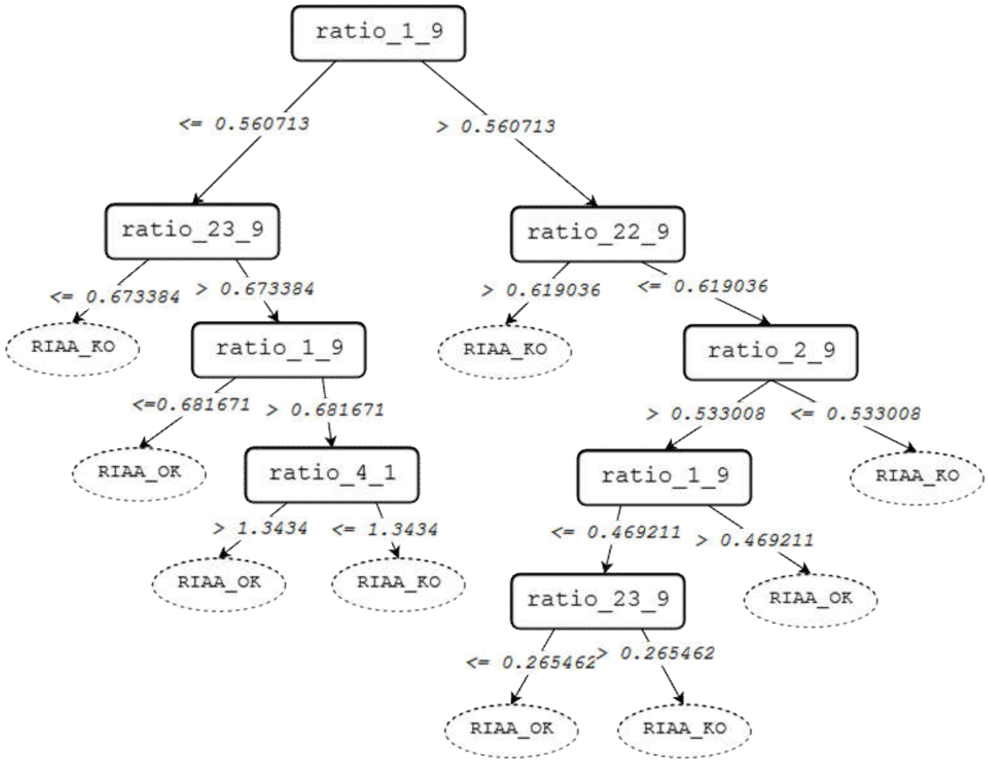


Figure 24. Design of the decision tree by C4.5 for M=10 in the global dataset (2000 instances).

According to it, not all ratios are necessary to find the best model to classify the given data (that is, their information gain is not as relevant for the training data as the others so no new node is created). Only 9 leaves and 17 nodes are necessary.

It can be observed that the values for those ratios are evaluated (see Figure 24), and a new branch or a class is assigned. For example, if ratio_1_9 has a value greater than 0.5607, then ratio 23_9 is evaluated (left part of the tree). On the other hand, if ratio_1_9 is less or equal to 0.5607, another branch is created (right side of the tree) and ratio_22_9. When a ratio is evaluated and no new branch is necessary to be created (that is, the value for that attribute directly relates to the class), then the class is yielded and the branch is finished. An example of this can be seen for ratio_2_9, where values lower or equal to 0.533 directly belong to the class *RIAA-KO*, whereas values greater than 0.533 create a new sub-branch with other attributes to evaluate (ratio_1_9).

It can also been seen that the same attribute can be evaluated several times with other values if that means the information provided by it gives some meaning for the model (such as ratio_1_9).

The accuracy for each genre has been also considered for the study as the results can give relevant information about the suitability of the algorithm for different kinds of music. Since there is an huge variety of music, with their particular spectral characteristics (due to the instruments involved, the combination of frequencies and the effects into the signal), it has been considered a relevant focus in the experiments.

The results for both learning mechanisms are exposed in Table 6 below:

	C4.5 (M=10)	SVM
Classical	83,67%	89,28%
Electronic-Dance	83,50%	96,50%
Experimental	79,79%	83,33%
Jazz	92,42%	97,47%
Lounge-Downtempo	92,93%	96,46%
Metal-Industrial	96,42%	98,46%
Pop-Rock	93,43%	100%
Rap-Dubstep	96,46%	98,98%
Reggae-Ska	96%	98%
Soul-Funk	93%	99%
GLOBAL	94.82%	95.02%

Table 6. Accuracies for global dataset (all genres) and per genre.

For the global dataset, SVM (95.02%) has better results than C4.5 (94,82%). C4.5 yields the best results for *Rap-Dubstep* (96,46%) and *Reggae-Ska* (96%), and the worst for *Experimental* (79,79%), *Electronic-Dance* (83,5%) and *Classical* (83,67%). SVM yields the best results for *Pop-Rock* (100%), *Rap-Dubstep* (98,99%) and *Soul-Funk* (99%), whereas the worst results are for *Experimental* (83,33%) and *Classical* (89,28%). In overall terms and for both classifiers, the genres *Experimental* and *Classical* yield the worst results, whereas the best ones depend on the classifier used. However, *Metal-Industrial* and *Rap-Dubstep* yield similar results for both classifiers. In addition, some differences are spotted: *Pop-Rock* and *Soul-Funk* and far better classified by SVM than by C4.5.

The overall Precision, Recall and F-Measure are given in Table 7 below:

	SVM		C4.5 (M=10)	
	RIAA-OK	*RIAA-KO*	*RIAA-OK*	*RIAA-KO*
Precision	0.935	0.968	0.942	0.954
Recall	0.969	0.933	0.955	0.942
F-Measure	0.952	0.950	0.949	0.948

Table 7. Overall Precision, Recall and F-Measure for both classifiers per class.

If we look at the confusion matrices, where detailed information about the correctly and incorrectly classified instances is shown (see Table 8), we can see the different accuracies for each class:

SVM

	riaa_ok	riaa_ko
riaa_ok	964	31
riaa_ko	67	928

C4.5 (M = 10)

	riaa_ok	riaa_ko
riaa_ok	950	45
riaa_ko	58	937

Table 8. Confusion matrices per SVM and C4.5.

The accuracies per genre are all displayed in the annex at the end of the document.

C4.5 and SVM seem to have better results for *RIAA-OK* (950/964 correct respectively) than for *RIAA-KO* (937/928 correct respectively). *RIAA-KO* seems to be more difficult to detect than *RIAA-OK*. Per genre, the same pattern seems to happen: SVM also outperforms C4.5 for all genres, and *RIAA-OK* instances seem to be also easier to detect than the *RIAA-KO* instances.

5.2. Results for altered playback speed detection

The algorithm was tested using the same classifiers as the RIAA filtering detection and using the default parameter values for each of them.

Two main different accuracies are calculated: one for the 8 classes of altered speed, and another for a binary classification (*up_speed* and *down_speed*). For this latter one, all the “minus”classes are grouped into *down_speed* class, and all the “plus” classes are grouped into *up_speed*. The idea in this case is to see if the accuracy improves or decreases when the knowledge model is simplified to only 2 classes. In addtion, per each experiment, both absolute distances and logarithmic distances are used to test the system. The results can be seen in Table 9 below:

	C4.5	**SVM**
8 classes – absolute distance	12.74%	13.92%
8 classes – logarithmic distance	12.74 %	12.86%
2 classes – absolute distance	50%	49.64 %
2 classes – logarithmic distance	50%	47.14 %

Table 9. Accuracies for each experiment using both machine-learning algorithms

For the 8-classes dataset, the results show a very low accuracy (less than 14%) for both algorithms. However, SVM seems to have slightly better accuracy (13,92% against 12,74% in the best case). On the other hand, the 2-classes dataset yields far better results (although not high enough for the algorithm to be considered useful for that purpose). In this case, C4.5 seems to perform better than SVM (50% against 49.64% in the best case). In addition, it can be seen that SVM absolute distance values yield slightly better results (13.92% and 49.64%) than logarithmic (12.86% and 47.14%).

Due to similarity of the results between absolute and logarithmic approaches plus the lower performance of the logarithmic one, this section will be focused on the analysis of the absolute distances experiment. However, results from logarithmic distances tests can be found in the annex at the end of the document.

The Precision, Recall and F-Measure are given in the Table 10 below for the case of absolute values:

	SVM		C4.5	
	8-classes	*2-classes*	*8-classes*	*2-classes*
Precision	0.125	0.496	0.107	0.250
Recall	0.139	0.496	0.127	0.500
F-Measure	0.113	0.496	0.105	0.333

Table 10. Average Precision, Recall and F-Measure for both classifiers per each experiment.

Looking at the confusion matrices we can see the correctly and incorrectly classified instances with the different accuracies for each class:

	SVM ABSOLUTE DISTANCES (8-CLASSES)							
	speed_minus1	**speed_plus1**	**speed_minus2**	**speed_plus2**	**speed_minus5**	**speed_plus5**	**speed_minus8**	**speed_plus8**
speed_minus1	46	30	3	8	3	5	8	2
speed_plus1	47	29	4	6	5	10	4	0
speed_minus2	41	24	5	4	7	9	12	3
speed_plus2	41	21	5	7	5	11	14	1
speed_minus5	39	14	8	7	5	9	18	5
speed_plus5	36	19	8	6	6	4	21	5
speed_minus8	33	18	7	11	9	3	18	6
speed_plus8	36	17	6	8	4	8	23	3

Table 11. Confusion matrices per SVM in 8-class dataset.

	C4.5 ABSOLUTE DISTANCES (8-CLASSES)							
	speed_minus1	**speed_plus1**	**speed_minus2**	**speed_plus2**	**speed_minus5**	**speed_plus5**	**speed_minus8**	**speed_plus8**
speed_minus1	23	36	2	2	2	1	19	20
speed_plus1	33	26	5	2	3	3	15	18
speed_minus2	25	22	1	15	4	1	20	17
speed_plus2	21	24	14	5	3	1	22	15
speed_minus5	10	22	4	6	4	13	26	20
speed_plus5	13	19	8	4	8	1	31	21
speed_minus8	15	16	4	4	4	7	28	27
speed_plus8	9	23	3	4	4	7	36	19

Table 12. Confusion matrices per C4.5 in 8-class dataset.

It can be observed in Table 11 and Table 12 that the classes getting the most instances are the extremes: minus1/plus1 and minus8/plus8 for both algorithms, that is, the classes that yield either the lowest values of distance or the highest values of distance, so that the classifier can separate the data in an easier manner. The other classes are in the area that overlaps among the classes (see Figure 25 below):

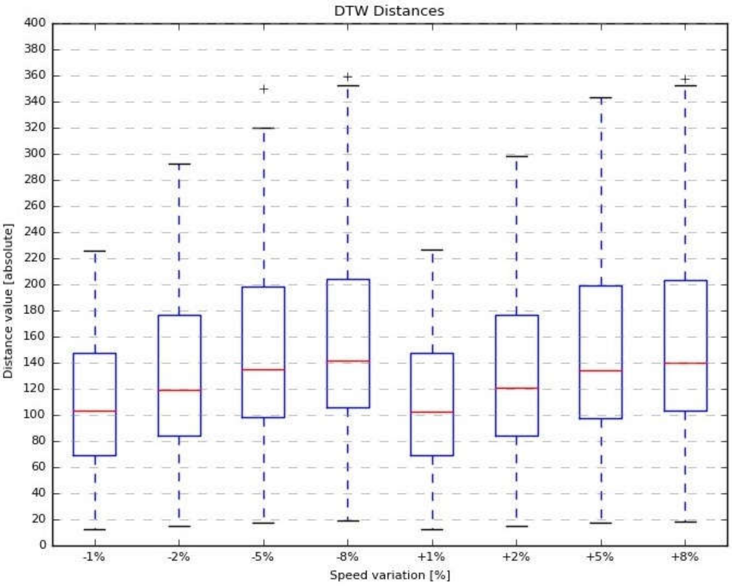


Figure 25. Boxplot of absolute distance values for different speed variations.
The interval of values for each of the speeds are shown in a box, the red line being the mean value and the blue square containing the majority of the instances

For the 2-class experiment (see Table 13), it seems clear for SVM that there is no class easier to classify than the other, since as previously commented, the distance values are simmetric (almost equal for each counterpart, i.e minus1 and plus1). In the case of C4.5 all instances are classified as down_speed, given an unreliable 50% of accuracy.

SVM

	down_speed	up_speed
down_speed	202	218
up_speed	205	215

C4.5

	down_speed	up_speed
down_speed	420	0
up_speed	420	0

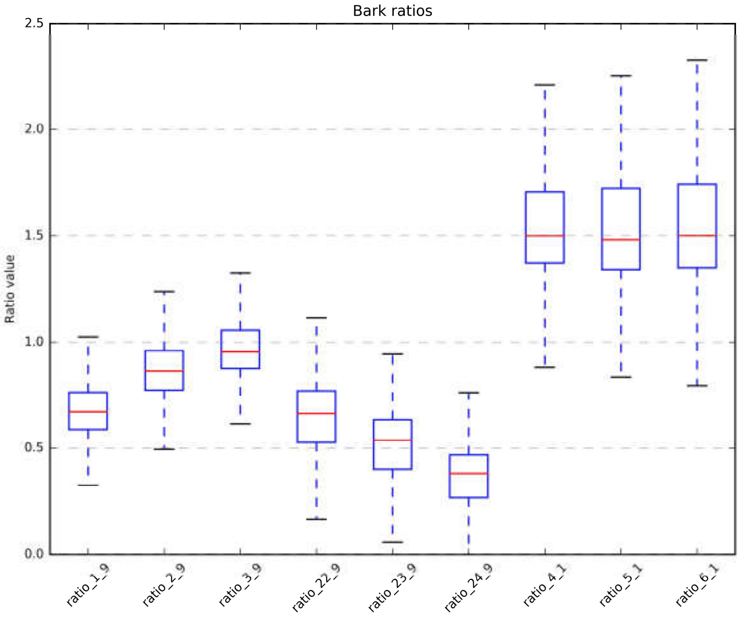
Table 13. Confusion matrices per SVM and C4.5 for the 2-class dataset

6. Discussion

6.1. RIAA detection

Results show a great accuracy on the RIAA filtering detection, as bark ratios yield a proper representation of the importance of the frequency components within the spectrum. This representation is easily differentiable among audio files where the RIAA has not been applied and audio files correctly converted from legacy formats like vinyl. The global results show an overall accuracy of around 95% for the case of Support Vector Machines which leads us to think that it is a good model to detect such type of defect. The differences among classifiers show that SVM machines are more suitable for the given data and this particular binary classification.

As can be seen below, it is clear where the ratio value move for *RIAA-OK* and *RIAA-KO* instances in the global dataset, as shown in Figure 26:



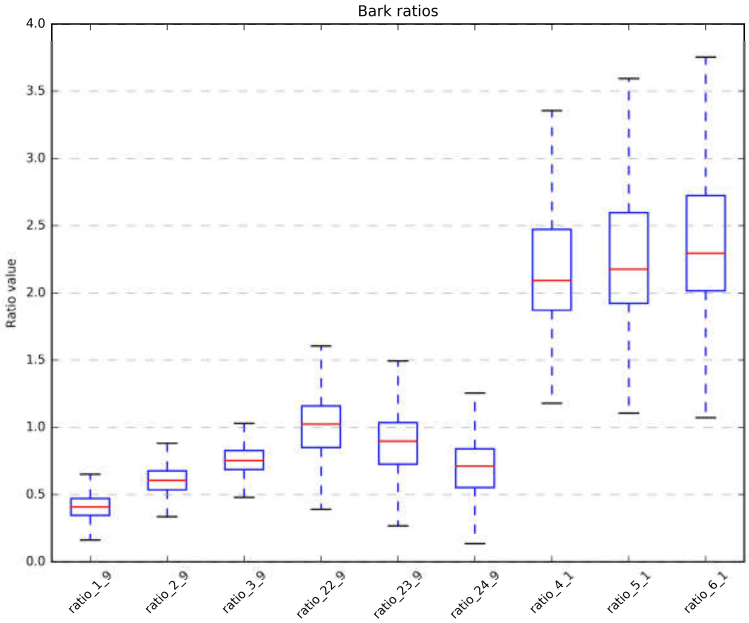
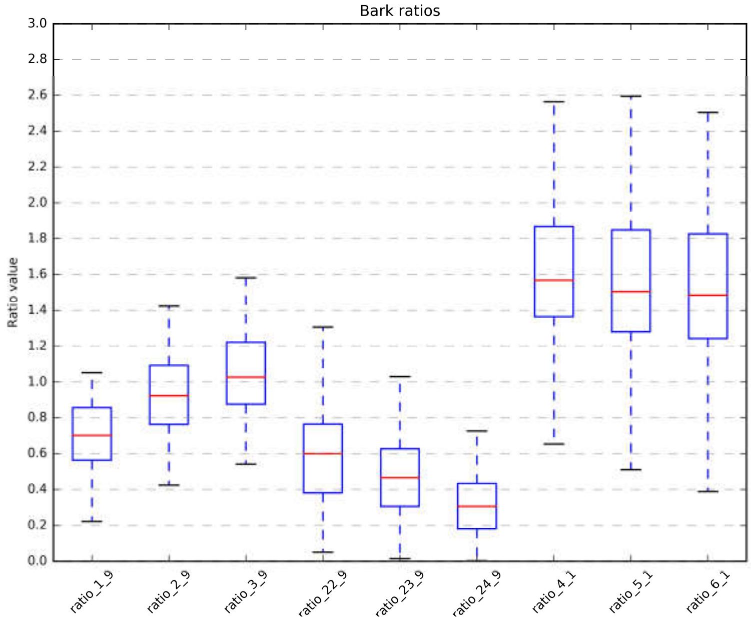


Figure 26. Boxplots for the global dataset of RIAA-OK (up) and RIAA-KO (down).

However, if we look into the per genre results, we can clearly see that this performance decreases for some genres, such as *Classical* or *Experimental*. On the other hand, genres such as *Metal-Industrial*, *Rap-Dubstep* or *Reggae-Ska* perform really well.

If we check the ratios distribution for *Experimental*, we see that they comprise a larger interval of values (Figure 27):



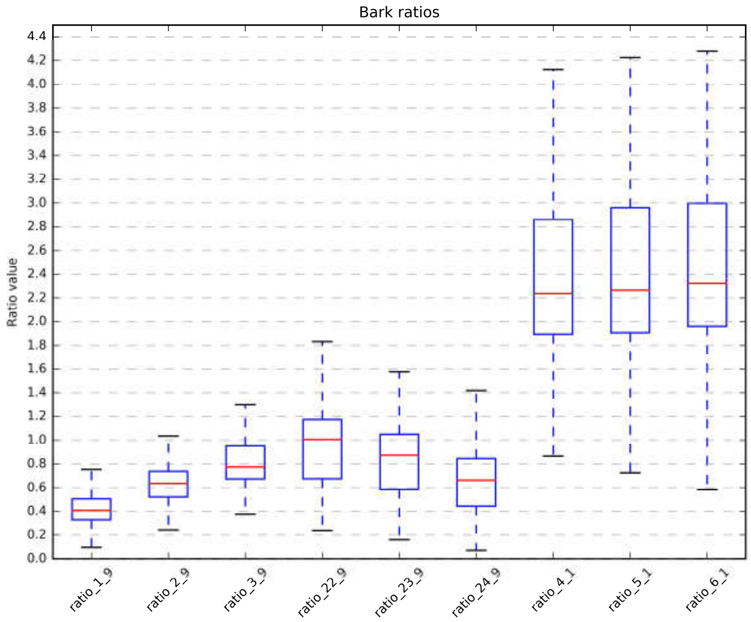
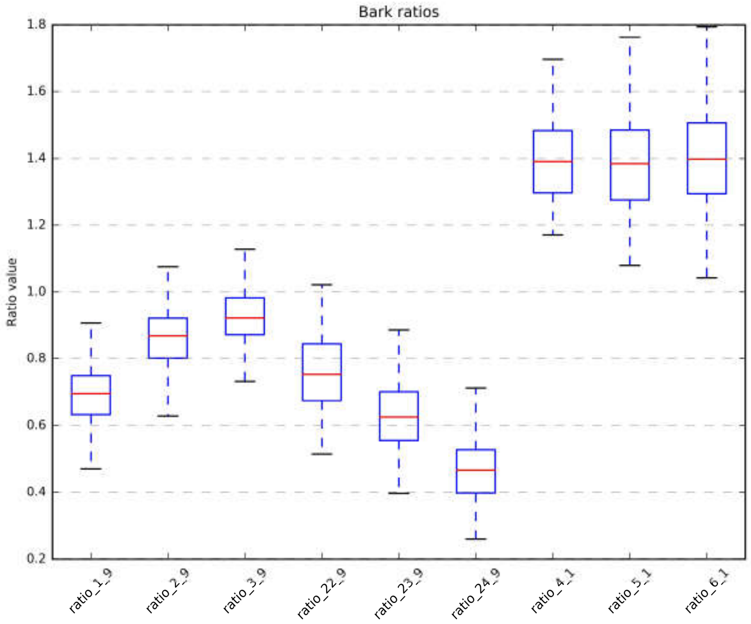


Figure 27. Boxplots for Experimental genre of RIAA-OK (up) and RIAA-KO (down)

Many of the ratios move within a large interval (for example, as seen in Figure 27, ratio_2-9 moves from 0.4 to 1.4 for *RIAA-OK* and from 0.2 to 1.0 for *RIAA-KO*). This behaviour is not observed for the higher-accuracy genres, where the interval of values for this ratio is narrower (Figure 28):



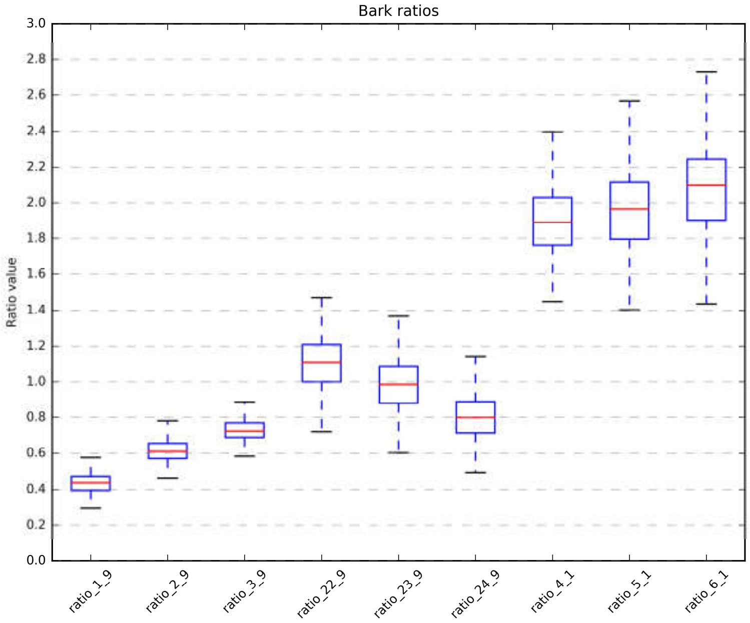
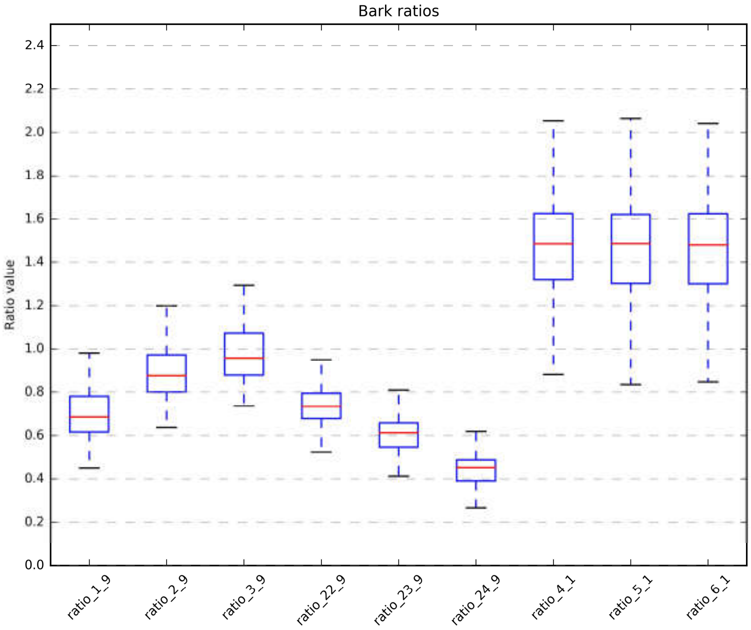


Figure 28. Boxplots for Metal-Industrial genre of RIAA-OK (up) and RIAA-KO (down)



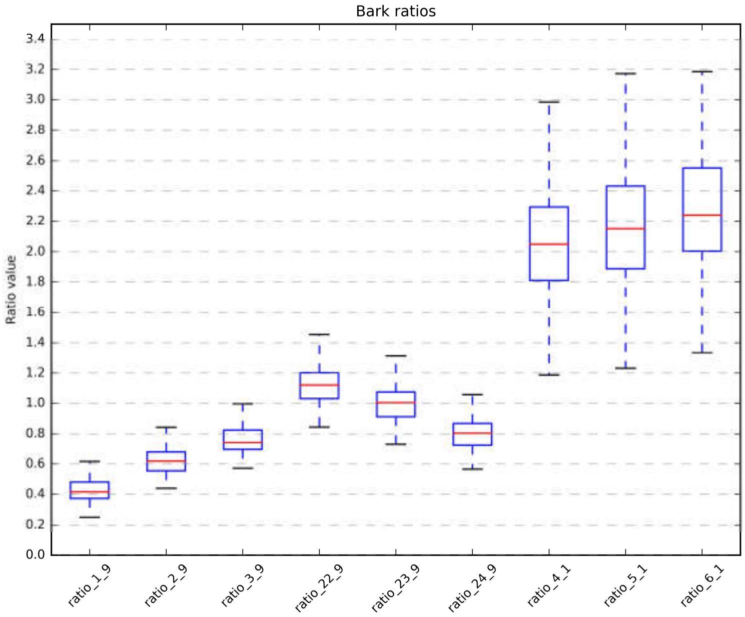
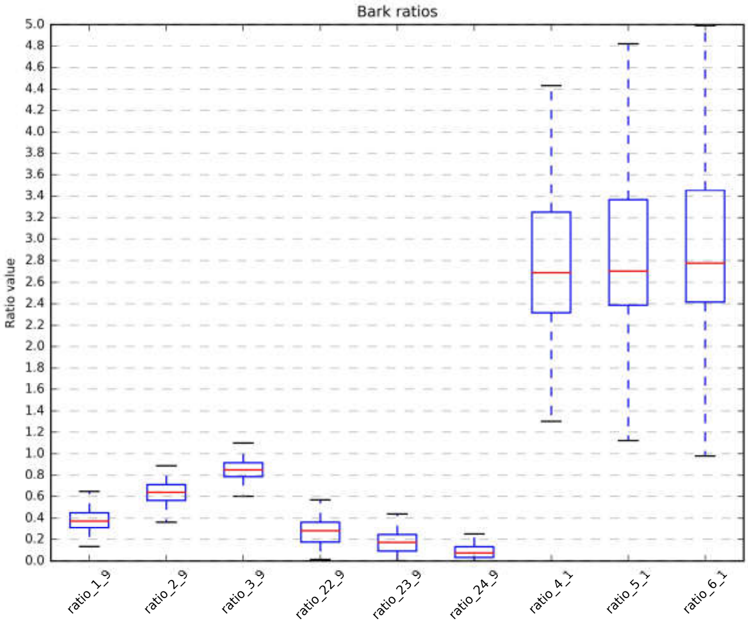


Figure 29. Boxplots for Rap-Dubstep genre of RIAA-OK (up) and RIAA-KO (down)

For *Metal-Industrial* and *Rap-Dubstep*, ratio_2_9 shows similar values and within a smaller interval than Experimental (as seen in figure 21 and figure 22: they move from 0.6 to 1.1 or 1.2 for *RIAA-OK* and from 0.4 to 0.8 *for RIAA-KO*).

Similar behaviour can be observed in *Classical* music, however the variability is reduced, as the boxes are very narrow compared to the other genres (Figure 30):



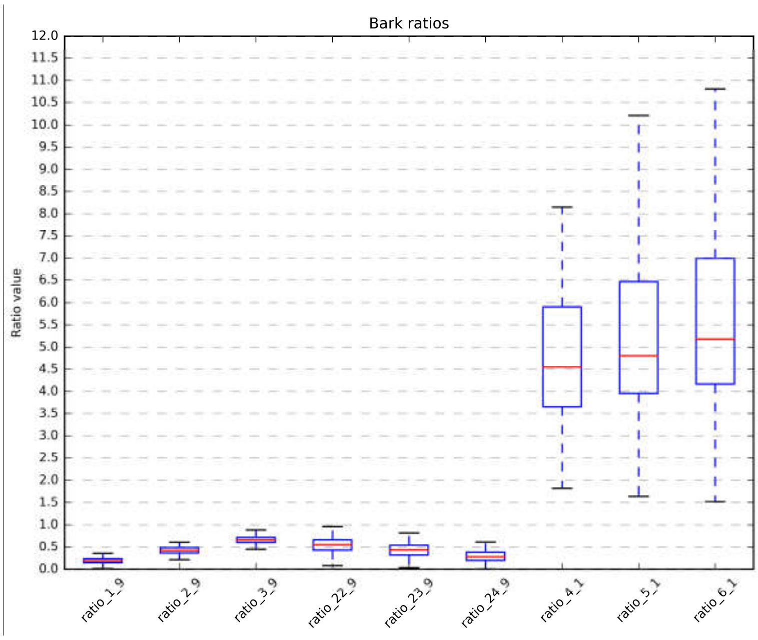


Figure 30. Boxplots for Classical genre of RIAA-OK (up) and RIAA-KO (down)

This means that some instances yield values out of the average range (called outliers, as shown in Figure 31 and figure 32), that is, the balances for the bark bands do not follow the average path due to the variability of music belonging to those genres. *Metal-Industrial* or *Rap-Dubstep* instances usually follow similar patterns of musicality, whereas genres such as *Experimental* or *Classical* music have wider musical patterns. Therefore, when setting the threshold values for the ratios, many examples of *Experimental*/*Classical* may not fall in the set interval (as previously seen in the C4.5 tree for example), and therefore they will be incorrectly classified by the algorithm.

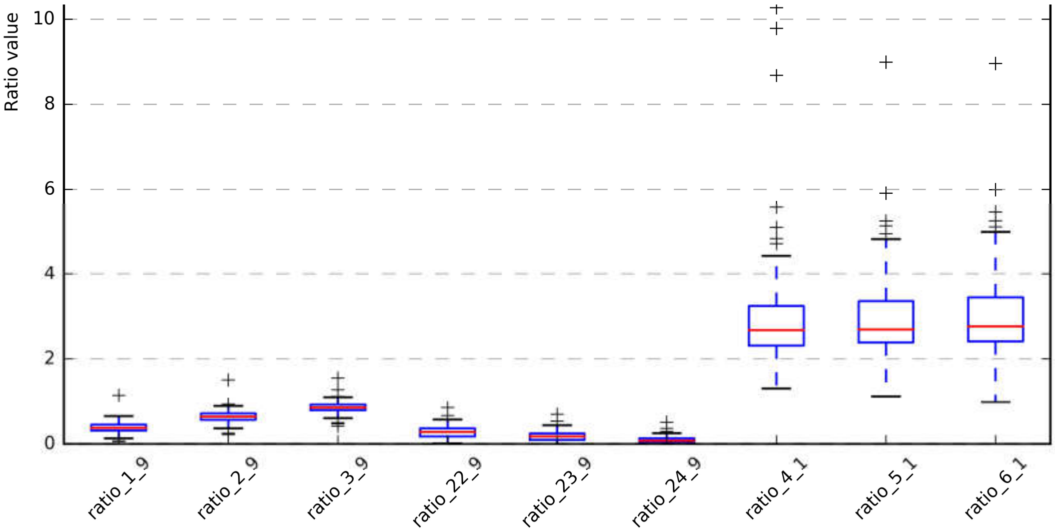


Figure 31. Detail of the Boxplots for Classical genre of RIAA-OK showing outliers.

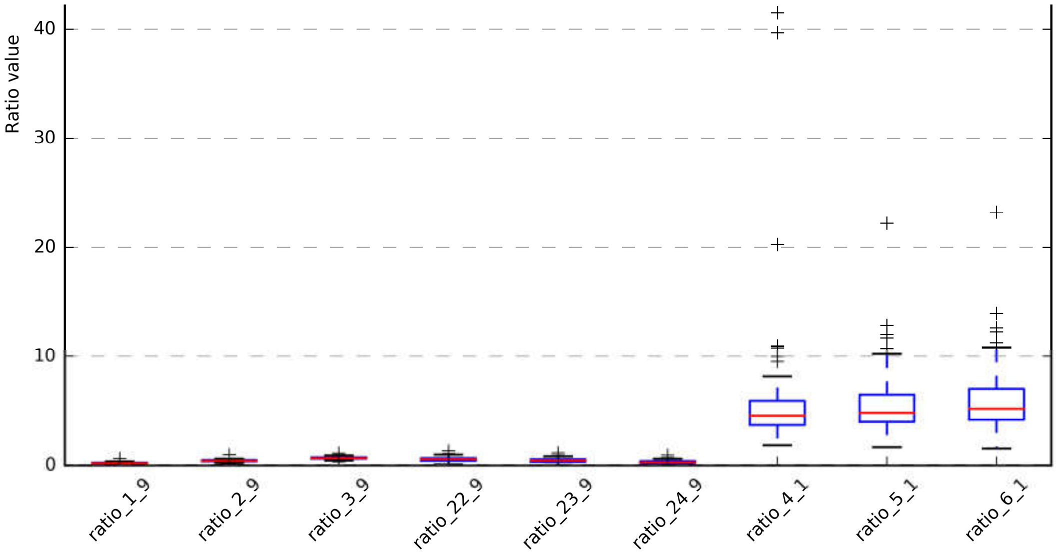


Figure 32. Detail of the Boxplots for Classical genre of RIAA-KO showing outliers.

Those outliers, due to their spectral components, do not get that much altered when the RIAA filtering is not applied. *Experimental* music instances usually have most of the energy in the extremes of the spectrum (wider range of frequencies), so that attenuation has not as relevant effect as in other genres. In addition, since *Classical* music has lower amount of energy in low frequencies than other genres and not much energy in the very-high frequencies due to the instruments involved, in some cases, the attenuation at those frequencies is not as relevant as in other genres such as *Metal-Industrial*, *Rap-Dubstep* or *Reggae-Ska*, where there is prominence of the LF components (and loudness) is higher. Then, if the RIAA filtering is missing, those frequencies get highly attenuated.

The piece *Gata* by Jeff Mills (Figure 33) is a good example of the algorithm not working properly for *Experimental* music, due to the invariance of the spectrum. The *RIAA-KO* and the *RIAA-OK* spectrums have almost the same shape, so the ratios will be similar:

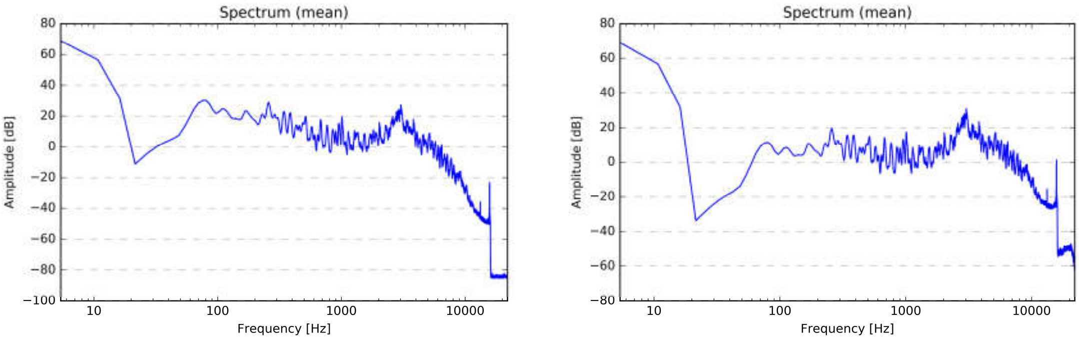


Figure 33. Spectrum of Jeff Mills' Gata for RIAA-OK (left) and RIAA-KO (right)

On the other hand, *Links 234* from Rammstein is a *Metal* example where the alteration of the spectrum is clearly noticeable. It can be seen how the spectrum shape gets flatter when no RIAA filtering is applied (Figure 34):

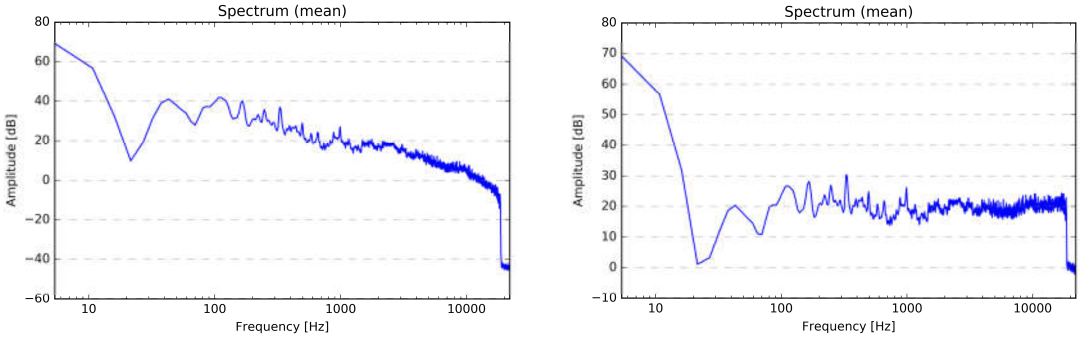


Figure 34. Spectrum of Rammstein's Links 234 for RIAA-OK (left) and RIAA-KO (right).

An example of Classical music brought by Stravinsky, *Concerto for Piano and Winds* can be useful to understand the lower-accuracy results for this genre:

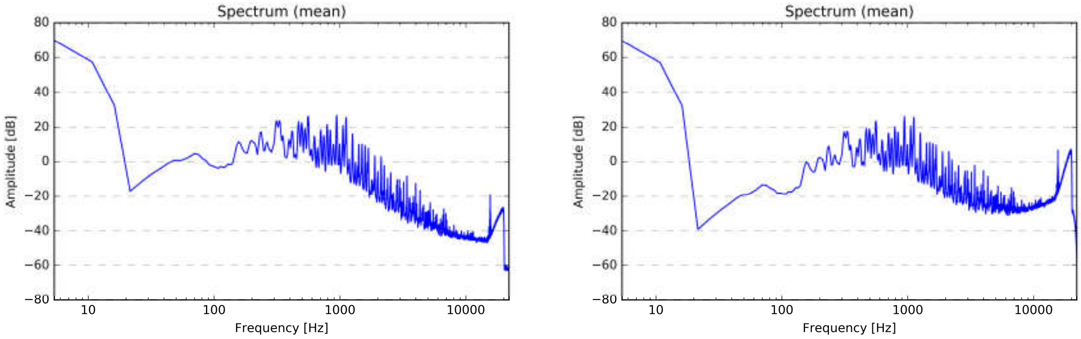


Figure 35. Spectrum of Igor Stravinslky's Concerto for Piano and Winds for RIAA-OK (left) and RIAA-KO (right).

As seen in Figure 35, the shape of the spectrum remains almost the same, yielding similar results for the ratios.

According to the aforecommented results this method seems to work very well for genres where the musical patterns are more similar such as *Rap-Dubstep*, *Metal-Industrial* or *Jazz-Swing*, whereas other musically wider genres such as *Classical* or Experimental will have more difficulties in the detection part.

6.2. Altered playback speed detection

As can be seen from previously exposed results, the algorithm does not yield a proper performance for the given problem.

In one hand, for the 8-classes experiment, accuracy percentage is extremely low (less than 14% in the best case). On the other, for the case of the 2-class experiment, the maximum reached accuracy is 50% (in the case of C4.5). However, it has been demonstrated in the previous section that it not reliable, since all instances are classified to the same class. For the case of Support Vector Machines, the accuracy is a bit lower, but at least the clasification is spread between both classes.

All the aforementioned results are mainly caused by the simmetries between positive and negative counterparts, since they yield almost the exact value of distance:

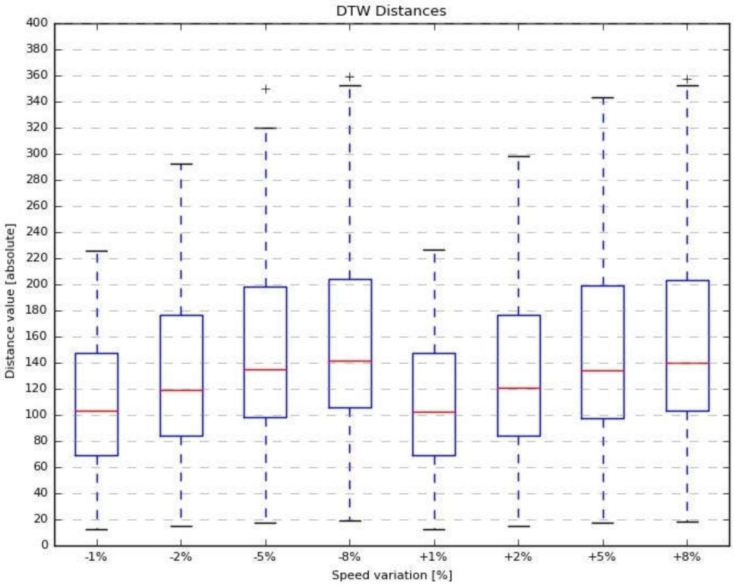


Figure 36. Boxplot of absolute distance values for different speed variations.
The interval of values for each of the speeds are shown in a box, the red line being the mean value and the blue square containing the majority of the instances

As seen in Figure 36, for example, +1% and -1% yield distances within the same interval, and it happens for all the altered speed cases: they are clearly symetric either for the up-speed and down-speed counterparts (same amount of variation when decreasing or increasing the same percentage of playback speed). In addition, the intervals of values overlap in most of the cases (boxes move within the same range of values) and this phenomenon makes it really difficult for the classifier to establish a proper threshold in order to separate the classes, and therefore many instances are wrongly classified.

This is seen even clearer with the 2-class example, as both classes (*up_speed* and *down_speed*) move within almost the same interval of values, and classifiers like C4.5 directly wrongly classify all the instance to one class (as seen in Table 14 below):

C4.5

	down_speed	up_speed
down_speed	420	0
up_speed	420	0

Table 14. Confusion matrices per C4.5 for the 2-class dataset

All this leads us to think that using Dynamic Time Warping (at least, calculating distances by just comparing sample by sample between two audio signals) does not show a proper resolution for the problem under study.

7. Conclusions

In this work, current taxonomy of known audio defects is reviewed according to the state of the art methods, highlighting the characteristics of each type and the solutions (if any) for their detection and correction. Afterwards, the vinyl technology is analyzed due to its error-prone nature. That is why the defects related to digitizing vinyl media are chosen for research here: the lack of RIAA filtering and the altered playback speed.

Later, the mechanisms for detection are exposed. Those mechanisms are based on the psychoacoustic model developed by Zwicker (that is, the use of bark-band decomposition of the spectrum) and state-of-the-art machine learning techniques.

Results show a great accuracy on the RIAA filtering detection, as bark ratios yield a proper representation of the importance of the frequency components within the spectrum. This representation is easily differentiable among audio files where the RIAA filtering has not been applied and audio files correctly converted from legacy formats like vinyl. The global results show an overall accuracy of around 95% in the best case. However, if we look into the per genre results, we can clearly see that this performance decreases for some genres, such as *Classical* or *Experimental*. On the other hand, genres such as *Metal-Industrial*, *Rap-Dubstep* or *Reggae-Ska* perform really well. For genres such as *Classical* or *Experimental*, some instances yield values out of the average range (outliers), as the balances for the bark bands do not follow the average path due to the variability of music belonging to those genres and the distribution of their frequencial components. *Metal-Industrial* or *Rap-Dubstep* instances usually follow similar patterns of musicality, whereas genres such as *Experimental* or *Classical* music have wider musical patterns. Therefore, when setting the threshold values for the ratios, many examples of *Experimental* and *Classical* do not fall in the set interval and therefore they are incorrectly classified by the algorithm. Those outliers, due to their spectral components, do not get that much altered when the RIAA filtering is not applied. *Experimental* music instances usually have most of the energy in the extremes of the spectrum (wider range of frequencies), so that attenuation has not as relevant effect as in other genres. In addition, since *Classical* music has lower amount of energy in low frequencies than other genres and not much energy in the very-high frequencies due to the instruments involved, in some cases, the attenuation at those frequencies is not as relevant as in other genres such as *Metal-Industrial*, *Rap-Dubstep* or *Reggae-Ska*, where there is prominence of the low-frequency components (and loudness) is higher. Then, if the RIAA filtering is missing, those frequencies are highly attenuated.

Some improvements could be considered in order to raise the accuracy for those genres. The inclusion of other bark ratios (comparing other bark bands) could help and avoid the limitation the lower energy in the extremes of the spectrum (in the case of *Classical*) or the inverse phenomenon for *Experimental* (where main spectral energy appears at very-high or very-low frequencies).

With regards to altered playback speed detection, the algorithm does not yield a proper performance for the given problem. In one hand, for the 8-classes experiment, accuracy percentage is extremely low (less than 14% in the best case). On the other, for the case of the 2-class experiment, the maximum reached accuracy is 50% (in the case of C4.5), however, it has been seen in the previous section that this result is not reliable, since all instances are classified to the same class. All the aforementioned results are mainly caused by the simmetries between positive and negative counterparts, since they yield almost the exact value of distance: they are clearly symetric either for the up-speed and down-speed counterparts (same amount of variation when decreasing or increasing the same percentage of playback speed). In addition, the intervals of values overlap in most of the cases (boxes move within the same range of values) and this phenomenon makes it really difficult for the classifier to establish a proper threshold in order to separate the classes, and therefore many instances are wrongly classified. This leads us to consider that calculating distances by just comparing sample by sample between two audio signals does not resolve the problem under study.

In order to dramatically improve the performance of this algorithm, other approaches should be considered. Extracting descriptors for the tonal components of the signal could be performed [57], since altering speed also alters the pitch of the signal. Therefore, comparing the values of an altered file against the nominal reference may improve the classification task. Also, some other parameters such us the duration or tempo estimation [58] (increasing or decreasing speed implies increasion or decreasing the tempo) could be extracted and used by the classifier when comparing to the reference audio file.

All code for both algorithms can be found in GitHub:
https://github.com/ignasi42/defect_detector

REFERENCES

1. Pohlmann, Kenneth C. The Compact Disc Handbook. Middleton, Wisconsin: A-R Editions, 1992.
2. Shannon, Claude E. (1948). "A Mathematical Theory of Communication". Bell System Technical Journal 27 (3): 379–423.
3. Ian H. Witten, Eibe Frank, Len Trigg, Mark Hall, Geoffrey Holmes, and Sally Jo Cunningham. Weka: Practical Machine Learning Tools and Techniques with Java Implementations. Department of Computer Science, University of Waikato, New Zealand, 1999.
4. Rudolf Mühlbauer. Automatic Audio Defect Detection. Bachelorarbeit. Bachelor of Science im Bachelorstudium Informatik, Jonhannes Kepler Universität Linz, Juni 2010.
5. Ryan Laney. Automatic Detection of Flaws in Recorded Music Using Wavelet Fingerprinting. Diss. College of William & Mary, 2011.
6. A.Wilson and B.M. Fazenda, Perception & evaluation of audio quality in music production in Proc. of the 16th Int. Conference on Digital Audio Effects (DAFx-13), Maynooth, Ireland, 2013, pp. 1-6.
7. Kim, J., Lee, J.H., Park, S. and Sung, H.Y., 2005, May. Development of Objective Sound Quality Evaluation Method Based on Subjective Sound Quality Evaluation. In *Audio Engineering Society Convention 118*. Audio Engineering Society.
8. Pras, A., Zimmerman, R., Levitin, D. and Guastavino, C., 2009, October. Subjective evaluation of mp3 compression for different musical genres. In Audio Engineering Society Convention 127. Audio Engineering Society.
9. MPEG. MPEG–2 advanced audio coding, AAC. International Standard IS 13818–7, ISO/IEC JTC1/SC29 WG11, 1997.
10. ITU-R BS. 1284-1: EN-General methods for the subjective assessment of sound quality. Technical report, International Telecommunication Union (ITU), Geneva, Switzerland; 2003.
11. Rec IT. BS. 1534-1". Method for the subjective assessment of intermediate quality level of coding systems. 2003..
12. ITU-R BS.1116-1, Methods for the subjective assessment of small impairments in audio systems including multichannel sound systems. ITU Radiocommunication Assembly (1997).
13. E. Pampalk. A matlab toolbox to compute music similarity from audio. In Proceedings of the Fifth International Confer- ence on Music Information Retrieval (ISMIR'04), Barcelona, Spain, October 10-14 2004.
14. Smith, Steven W. "The scientist and engineer's guide to digital signal processing." (1997).
15. J.R. Quinlan. C4.5: Programs for Machine Learning. Morgan Kaufmann, San Mateo, CA, 1993.
16. Tim Pohle. Extraction of Audio Descriptors and Their Evaluation in Music Classification Tasks. Diplomarbeit. OFAI, DFKI, Technische Universität Kaiserslautern, 2005.
17. J. Platt: Machines using Sequential Minimal Optimization. In B. Schoelkopf and C. Burges and A. Smola, editors, Advances in Kernel Methods - Support Vector Learning, 1998.
18. E. Zwicker and H. Fastl, Psychoacoustics, Facts and Models (Springer, Berlin, Heidelberg, 1990).

19. Ian H.Witten and Eibe Frank (2005) ”Data Mining: Practical machine learning tools and techniques”, 2nd Edition, Morgan Kaufmann, San Francisco, 2005.
20. Schindler A, Huber-Mörk R. Towards Objective Quality Assessment in Digital Collections, 2013.
21. R. ITU, “Method for objective measurements of perceived audio quality,” in ITU-R Recommendation BS.1387, (International Telecommunications Union, Geneva), 1998.
22. E. Ruzanski, “Effects of mp3 encoding on the sounds of music,” Potentials, IEEE, vol. 25, no. 2, pp. 43–45, 2006.
23. Li, Z., Wang, J.C., Cai, J., Duan, Z., Wang, H.M. and Wang, Y., 2013, October. Non-reference audio quality assessment for online live music recordings. In Proceedings of the 21st ACM international conference on Multimedia (pp. 63-72). ACM.
24. Wiggins, G.A., 2009, December. Semantic gap?? Schemantic schmap!! Methodological considerations in the scientific study of music. In Multimedia, 2009. ISM'09. 11th IEEE International Symposium on (pp. 477-482). IEEE.
25. Herrero C. Subjective and objective assessment of sound quality: Solutions and applications. In Proc. CIARM Conf 2005 (pp. 1-20).
26. T. M. Mitchell, Machine Learning, McGraw-Hill, 1997.
27. Kent, A., Berry, M. M., Luehrs, F. U., & Perry, J. W. (1955). Machine literature searching VIII. Operational criteria for designing information retrieval systems. *American documentation*, *6*(2), 93-101.
28. Godsill, S.J. & Rayner, P.J.W., 1998. Digital Audio Restoration - a statistical model based approach, pp.143–146
29. Krochmal, Andrew C., Gregory R. Hamel, and John E. Whitecar. "Method of detecting a DC offset in an automotive audio system." U.S. Patent No. 6,577,737. 10 Jun. 2003.
30. Reiss, J. & Sandler, M., 2004. Audio Issues In MIR Evaluation. Proceedings of the 5th International Conference on Music Information Retrieval ISMIR, pp.28–33.
31. Benjamin, E. and Gannon, B., 1998, September. Theoretical and Audible effects of jitter on Digital Audio Quality. In *Audio Engineering Society Convention 105*. Audio Engineering Society.
32. Liu, C.M., Hsu, H.W. & Lee, W.C., 2008. Compression artifacts in perceptual audio coding. *IEEE Transactions on Audio, Speech and Language Processing*, 16(4), pp.681–695.
33. Godsill, S., Rayner, P. and Cappé, O., 2002. Digital audio restoration. In Applications of digital signal processing to audio and acoustics (pp. 133-194). Springer US.
34. Zhou, Jinglei, et al. "Detecting Fake-Quality WAV Audio Based on Phase Differences." *International Workshop on Digital Watermarking*. Springer International Publishing, 2014.
35. Bruney, Paul F. "Audio image recovery system." U.S. Patent 4,204,092, issued May 20, 1980.
36. Prakash, Vinod, et al. "Removal of Birdie Artifact in Perceptual Audio Coders." *Audio Engineering Society Convention 116*. Audio Engineering Society, 2004.
37. Iwai, K.K. & Lim, J.S., 1994. Pre-Echo Detection & Reduction, (1991).
38. Renals, S., Wrigley, S. & Brown, G., 2003. Speech and crosstalk in multi-channel audio. , pp.1–24.
39. Driedger, J. & Müller, M., 2016. A Review of Time-Scale Modification of Music Signals. *Applied Sciences*, 6(2), p.57.

40. Czyzewski, A. and Maziewski, P., 2007, September. Some techniques for wow effect reduction. In 2007 IEEE International Conference on Image Processing (Vol. 4, pp. IV-29).
41. Czyzewski, A. and Maziewski, P., 2006. Wow defect reduction based on interpolation techniques, 54(4).
42. Dolby R. An audio noise reduction system. Journal of the Audio Engineering Society. 1967 Oct 1;15(4):383-8.
43. https://www.google.com/patents/US2845490.
44. Bauer, Benjamin B. "On the Measurement of Rumble in Phonograph Reproduction." Journal of the Audio Engineering Society 15.2 (1967): 143-146.
45. F. Deng, C. c. Bao, B. y. Xia and Y. Liang, "A novel hiss noise reduction method for audio signals based on MDCT," Wireless Communications and Signal Processing (WCSP), 2011 International Conference on, Nanjing, 2011.
46. Czyzewski, A. "Implementation of the Rough-Set Method for the Removal of Hiss." Journal of the Audio Engineering Society 45.11 (1997): 931-943.
47. Lipshitz SP, Pocock M, Vanderkooy J. On the audibility of midrange phase distortion in audio systems. Journal of the Audio Engineering Society. 1982 Sep 1;30(9):580-95.
48. M Muller. Information Retrieval for Music and Motion. Springer, 2007.
49. Electronics Projects Vol. 17, Volume 17, EFY Enterprises Pvt Ltd, 2009.
50. Fielder, L. D. (1995). Dynamic-range issues in the modern digital audio environment. *Journal of the Audio Engineering Society*, *43*(5), 322-339.
51. Kauppinen I, Kauppinen J. Reconstruction method for missing or damaged long portions in audio signal. Journal of the Audio Engineering Society. 2002 Jul 15;50(7/8):594-602.
52. Martinez, M. H. (2007). Evaluation of Audio Compression Artifacts, 47(1), 12–16.
53. Repp, R. (2006). Recording Quality Ratings by Music Professionals. International Computer Music Conference, 468–474.
54. Zwicker, E. (1961), Subdivision of the audible frequency range into critical bands, The Journal of the Acoustical Society of America, Volume 33, Issue 2, pp. 248-248 (1961).
55. Keogh E J, M J Pazzani, M J. Derivative Dynamic Time Warping. In First SIAM International Conference on Data Mining, 2001.
56. Avedano, C. Frequency-domain source identification and manipulation in stereo mixes for enhancement, suppression and re-panning applications. InApplications of Signal Processing to Audio and Acoustics, 2003 IEEE Workshop on. 2003 Oct 19 (pp. 55-58). IEEE.
57. Gómez E. Tonal description of music audio signals. Department of Information and Communication Technologies. 2006.
58. Peeters G, Flocon-Cholet J. Perceptual tempo estimation using GMM-regression. InProceedings of the second international ACM workshop on Music information retrieval with user-centered and multimodal strategies 2012 Nov 2 (pp. 45-50). ACM.

ANNEX1 – GENRE CONFUSION MATRICES FOR RIAA DETECTION

SVM

	riaa_ok	riaa_ko
riaa_ok	99	1
riaa_ko	6	94

C4.5 (M = 10)

	riaa_ok	riaa_ko
riaa_ok	88	12
riaa_ko	9	91

Table A. Confusion matrix per in Electronic-Dance genre.

SVM

	riaa_ok	riaa_ko
riaa_ok	97	2
riaa_ko	5	94

C4.5 (M = 10)

	riaa_ok	riaa_ko
riaa_ok	86	13
riaa_ko	1	98

Table B. Confusion matrix per in Lounge-Downtempo genre.

SVM

	riaa_ok	riaa_ko
riaa_ok	83	16
riaa_ko	17	82

C4.5 (M = 10)

	riaa_ok	riaa_ko
riaa_ok	72	27
riaa_ko	13	86

Table C. Confusion matrix per in Experimental genre.

SVM

	riaa_ok	riaa_ko
riaa_ok	98	1
riaa_ko	4	95

C4.5 (M = 10)

	riaa_ok	riaa_ko
riaa_ok	92	7
riaa_ko	8	91

Table D. Confusion matrix per in Jazz-Funk genre.

SVM

	riaa_ok	riaa_ko
riaa_ok	97	1
riaa_ko	2	96

C4.5 (M = 10)

	riaa_ok	riaa_ko
riaa_ok	93	5
riaa_ko	2	96

Table E. Confusion matrix per in Metal-Industrial genre.

SVM

	riaa_ok	riaa_ko
riaa_ok	89	9
riaa_ko	12	86

C4.5 (M = 10)

	riaa_ok	riaa_ko
riaa_ok	82	16
riaa_ko	16	82

Table F. Confusion matrix per in Classical genre.

SVM

	riaa_ok	riaa_ko
riaa_ok	99	0
riaa_ko	0	99

C4.5 (M = 10)

	riaa_ok	riaa_ko
riaa_ok	93	6
riaa_ko	7	92

Table G. Confusion matrix per in Pop-Rock genre.

SVM

	riaa_ok	riaa_ko
riaa_ok		
riaa_ko		

C4.5 (M = 10)

	riaa_ok	riaa_ko
riaa_ok	97	2
riaa_ko	5	94

Table H. Confusion matrix per in Rap-Dubstep genre.

SVM

	riaa_ok	riaa_ko
riaa_ok	99	1
riaa_ko	3	97

C4.5 (M = 10)

	riaa_ok	riaa_ko
riaa_ok	98	2
riaa_ko	6	94

Table I. Confusion matrix per in Reggae-Ska genre.

SVM

	riaa_ok	riaa_ko
riaa_ok	100	0
riaa_ko	2	98

C4.5 (M = 10)

	riaa_ok	riaa_ko
riaa_ok	95	5
riaa_ko	9	91

Table J. Confusion matrix per in Soul-Funk genre.

ANNEX 2 – LOG DISTANCES CONFUSION MATRICES FOR PLAYBACK SPEED DETECTION

	SVM LOGARITHMIC DISTANCES (8-CLASSES)							
	speed_ minus1	speed_ plus1	speed_ minus2	speed_ plus2	speed_ minus5	speed_ plus5	speed_ minus8	speed_ plus8
speed_minus1	15	18	7	2	13	9	22	19
speed_plus1	15	20	4	2	14	10	27	13
speed_minus2	11	15	3	0	14	13	30	19
speed_plus2	16	14	2	1	15	8	34	15
speed_minus5	12	13	2	0	7	9	38	24
speed_plus5	12	12	2	0	5	8	38	28
speed_minus8	9	9	4	0	7	13	37	26
speed_plus8	9	11	3	0	8	9	48	17

Table A. Confusion matrix per in 8-class dataset using logarithmic distances.

	C4.5 LOGARITHMIC DISTANCES (8-CLASSES)							
	speed_ minus1	speed_ plus1	speed_ minus2	speed_ plus2	speed_ minus5	speed_ plus5	speed_ minus8	speed_ plus8
speed_minus1	23	36	2	2	2	1	19	20
speed_plus1	33	26	5	2	3	3	15	18
speed_minus2	25	22	1	15	4	1	20	17
speed_plus2	21	24	14	5	3	1	22	15
speed_minus5	10	22	4	6	4	13	26	20
speed_plus5	13	19	8	4	8	1	31	21
speed_minus8	15	16	4	4	4	7	28	27
speed_plus8	9	23	3	4	4	7	36	19

Table B. Confusion matrix per in 8-class dataset using logarithmic distances.

SVM

	down_speed	up_speed
down_speed	213	207
up_speed	237	187

C4.5

	down_speed	up_speed
down_speed	420	0
up_speed	420	0

Table C. Confusion matrix per in 2-class dataset using logarithmic distances.