UNIVERSITATEA BABEŞ-BOLYAI CLUJ-NAPOCA

FACULTATEA DE MATEMATICǍ ŞI INFORMATICǍ

SPECIALIZAREA INFORMATICĂ-ROMÂNĂ

LUCRARE DE LICENŢĂ

Detecția și eliminarea distorsiunii din înregistrări audio de pe formate analoage

Conducător ştiinţific

Lect. Dr. Sterca Adrian

Absolvent

**Drimba Alexandru**

**2018**

BABEŞ-BOLYAI UNIVERSITY CLUJ-NAPOCA

FACULTY OF MATHEMATICS AND COMPUTER SCIENCE

SPECIALIZATION COMPUTER SCIENCE - ROMANIAN

DIPLOMA THESIS

Distortion detection and removal on audio recordings from analog formats

Supervisor

Lect. Dr. Sterca Adrian

Author

**Drimba Alexandru**

**2018**

Abstract

Contents

1. Introduction. Problem statement and motivation
   1. Introduction. Analog vs digital audio formats
   2. A brief history of audio recording formats.
   3. Mechanical analog storage. Recording and playback
   4. Causes of distortion in mechanical analog formats
   5. Purpose of this work
2. Related work (applications)
   1. Audacity
   2. Nero WaveEditor
3. Basic digital signal processing
   1. Digital audio signal representation
   2. Filters: Finite Impulse Response and Infinite Impulse Response
   3. Frequency domain and Fourier transforms
   4. Filter design and frequency equalization
4. Detecting and correcting distortion
   1. Automated marking of distorted samples using neural networks
   2. Extrapolation and linear prediction
   3. Burg’s method for calculating LP coefficients
   4. Repairing the distorted sample intervals
5. Application
   1. Requirements and specification
   2. Audio Data Sources
      1. File Storage. WAV and AU formats
      2. Caching
      3. Version control on an audio project
   3. Applying effects on data sources.
      1. FIR and IIR Filters. Equalizers.
      2. Linear prediction for sample repair
      3. Finding the distorted regions in an audio recording. Correction
      4. Signal pre- and post processing

6. Conclusions and Future Work

Chapter 1

Introduction. Problem statement and motivation

1.1. Introduction

We all listen to music. Some of us more, some – less, and some can’t live without it! Nowadays, it’s very easy to listen to whatever you want, whenever you want, due to the technological advances made in the last decades. But this wasn’t always the case. Until the late 1800’s, there was no device that would allow sound playback, so music could be listened to only from live performances. With the invention of various sound record and playback devices, like the gramophone, the magnetic tape, and later – the digital formats, sound reproduction soon became accessible to everyone.

Even if many people nowadays consider the analog formats obsolete and rely solely on their electronic devices to listen to audio, those “considered to be extinct” formats are actually regaining their former popularity: new vinyl pressing plants are being opened due to the increasing demand [12] (even though turntable sales stay at a constant rate [13]), and cassette tape sales grew by 74% in 2016 (still, the amount of albums sold on cassettes is very small reported to the total number of albums sold) [14].

The melomaniacs are divided into three main groups: those who stick to the digital formats and don’t want to hear about analog ones, those who consider that analog is by any means superior in quality to digital, and those who recognize the ups and downs of each of the formats. In reality, every storage medium has its advantages and disadvantages, so choosing between them is, in the end, a mostly subjective opinion.

In Table 1.1, we can see various qualities and defects of three main formats: digital formats (such as CDs, online streams, MP3s etc.), vinyl records and audio cassette tapes. One might say that the digital format is clearly the winner, but some still prefer the “touchable” formats for one or another of their unique features, being it the musical content inscribed on it, it’s historical value, the artwork and information on the cover or just for the feeling of owning a palpable collection.

Leaving aside preferences, it is undoubtedly that there are recordings on non-digital formats that haven’t yet made their way off to a more easily to use media. Often, the analog media degrades over time, and in order to get the best out of a recording, restoration of some degree may be required. This is done by making use of **signal processing,** either using specialized electronic devices (analog signal processing), or by editing the sound on a computer (digital signal processing).

|  |  |  |  |
| --- | --- | --- | --- |
|  | Format | | |
| Digital | Vinyl records | Cassette tapes |
| Advantages | Very easy to use and get access to | Artwork and packaging are much more intriguing than other formats' | Small and portable |
| Great sound quality (when using appropriate encoding) | The engraved sound wave is visible; you can see what you're listening to | Cheaper than vinyl records |
| Storage support doesn't easily degrade in time | Pretty good sound quality, relative to its predecessors | Easy to record, duplicate and edit |
| Playback, recording and editing can be easily accomplished | Can fit large amounts of information on the cover | Little wear over time |
| Disadvantages | Large file sizes for quality recordings | Prone to irreversible wear over time | Louder background noise than vinyl |
| Audible compression artifacts (when using lossy encodings to reduce file size) | Needs specialized equipment for playback | Production faults, such as wow and flutter |
| Easy to illegally share copyrighted content | Limited recording time | Limited recording time |
|  | Large and heavy format; storage needs to be made carefully | Limited dynamic range and frequency response |
|  | Costly to produce |  |

Table 1.1: Comparison between today’s most popular audio formats

* 1. A brief history of audio recording formats

History of sound recording begins with Édouard-Léon Scott de Martinville’s phonoauto-graph (Fig. 1.1), an entirely mechanical device that was constructed as an analog to the human ear. It consisted of a funnel-like horn, with a flexible membrane covering the small end of the horn acting as a diaphragm. A lightweight stylus (or needle) was attached to the membrane and traced a line on the moving surface of a lampblack (carbon deposited by the flame of an oil lamp) coated paper. The sound waves in the air were captured and concentrated into the diaphragm by the horn, which would cause a movement in the diaphragm and, subsequently, in the stylus, causing a modulation in the traced line (Fig. 1.2). The phonoautograph recording is called a phonoautogram, but because of the recording medium’s nature, it was impossible to play back. Way later, in 2008, playback was realized with computers, by optically playing high-quality scans of the recordings. [15]



Fig. 1.1: The phonoautograph

Fig. 1.2: Detail of a phonoautogram

Thomas Edison attached a stylus to a sound absorbing diaphragm, and he arranged such that the vibrations picked up by the diaphragm were translated into an up-and-down motion (vertical-cut) in the stylus. He used a cylinder, just like Scott, but replaced the fumed paper with tinfoil. Tinfoil is malleable and can be embossed with little effort, which is an important requirement for recording sound. The stylus would rest on the cylinder, and as it rotated, a mechanism dragged the stylus sideways, creating a spiral. The stylus would press into the tinfoil, and make a groove, a physical impression of the recorded sound. After running the machine while shouting “Marry had a little lamb” into the diaphragm, he ran it again, this time without speaking, and the machine spoke! The hill-and-dale groove in the cylinder moved the stylus, and thus the diaphragm, making it move in the same pattern as when recording.

 Tinfoil was impractically fragile, as the recordings would wear out after a few plays. By changing the recording medium to wax, Alexander Graham Bell’s team solved the durability problem, and it also sounded better. From 1888, Edison started producing phonographs for home use. The actual records are called wax cylinders, and were on the market until 1929, when they died out in favor of other formats. [18]

Fig. 1.3: Edison’s Phonograph

Meanwhile, Emil Berliner was experimenting with another method of recording sound into a wax record. The first change he made was switching from the up-and-down motion of the stylus to a left-right wobble. He used photoengraving to etch the lateral-cut groove into the cylinder, rather than cutting directly into it. He realized that using a flat disc instead would make the engraving a lot easier, by recording into a softer material and using it as mold to create a hard stamper. Berliner’s first discs were released around the same time as Edison’s cylinders were hitting the market.

In the end, Berliner’s disc won against the Edison cylinder by simply being better at everything. They were cheaper, as the duplication process was much easier and faster than the equivalent process for cylinders, which in their early days, were each recorded individually. The disc’s playing time could go up to 5 minutes per side on the 12" record, and could have a song on each of its sides, while the playing time of the cylinder was 2 minutes, eventually going up to 4 minutes with the Edison Amberol Cylinder. Besides that, discs were more durable, as they were made of plastic materials ( like shellac ) instead of wax, easier to store, easier to use, and could fit a lot of information on their center labels [19]. Discs also came in a variety of sizes, the common ones being 7", 10" and 12", and the playing speed was standardized to 78RPM.

With the advance of the electronic technology, electrical microphones, signal amplifiers and electromechanical devices were developed, and they heavily impacted the industry of audio recording. Starting from 1925, phonograph recording masters were cut with electrically powered cutting heads, which increased the quality of the recording by means of frequency response, dynamic range and sound clarity. Playback could also be made electrically, by picking up the stylus’ movement as an electrical signal, amplifying and providing it to a loudspeaker.

A couple of decades later, by exploiting some metals’ property of magnetization, a new audio recording format was introduced: the magnetic recording. If some metals, like ferrite and iron AlNiCo alloy are exposed to a strong magnetic field, they will in turn become slightly magnetic, i.e. retain a part of the magnetic force that was applied on them.

The two main formats to make use of magnetization were the wire recorder and the magnetic tape. The wire recorder used thin steel wire, while the magnetic tape was a long, thin and narrow strip of plastic film, coated with magnetizable material. By passing an electrical signal through a coil, it would generate a fluctuating magnetic field that would be partially stored on the tape/wire. By passing the wire/tape by a similar coil, a small current, almost identical to the original signal, would be induced in the coil. Wire recording lasted from 1946 to 1954, when tape recording already became much simpler, affordable and compact. Magnetic tape came in various formats, sizes and speeds, and was the industry standard for record mastering until the digital era kicked in in the 80’s.



Fig. 1.4: Comparison between disc and cylinder, lateral and vertical-cut

Fig. 1.5: Magnetic storage formats: wire recorder steel wire + magnetic tape reel (left), Compact Audio Cassette (right)

The gramophone disc was still present. It has increased in fidelity over years: the electric recording was introduced; then, in 1948, the vinyl record appeared. By using the more fine grained PVC instead of shellac, the grooves could be more compact and the speed went down to 33⅓RPM or 45RPM. Thus, the new format brought more durability, longer playing times and a great improvement in sound quality. In 1957, the first stereophonic disc was released, by combining vertical-cut with horizontal-cut to provide two moving axis for the reproducing stylus. The vinyl record stayed the most popular format until the rise of the digital formats, starting with the CD.

Storing the audio signal in mechanical or magnetic form introduces elements of confusion because of the storage medium itself. Analog recording devices have a limit in their capability of tracing a line between their input and output, mostly due to the inherent characteristics of the storage medium, like limited signal-to-noise ratio, frequency response, dynamic range and many others. The idea of recording something that is not a direct analog of the desired information, but a descriptive form that avoids the quality traps of ordinary recording is the idea of digital recording.

In the basic system of digital recording, the electrical analog of the sound wave pressure variations being recorded is sampled at fixed intervals. This operation converts the smoothly varying waveform into a series of amplitude values. A number is then assigned to each of these values. This results in an approximation of the original waveform. The first commercial digital format was the CD (Compact Disc), which was designed to be the successor of the vinyl. It stored the audio signal optically, as a sequence of binary numbers, each representing the value of a sample’s amplitude. It replaced the vinyl record and cassette tape by the early 2000’s, just to be rendered obsolete with the advent of internet-based distribution of files.

1.3. Mechanical analog storage. Recording and playback

As we’ve seen in the previous section, mechanical analog storage was the most preferred audio support for the general public until the introduction of the Compact Disc and the subsequent digital formats. Its basic principles haven’t changed much since the development of Berliner’s gramophone disc, but the advances in technological and electric technologies made the recordings increase in fidelity.

The production of a disc is as following: the music is usually recorded on magnetic tape, as it’s a lot easier to edit it. This step is called “mastering”. After mastering is done, the sound is provided to a cutting head, which cuts into a lacquer. After the lacquer hardens, it is electroplated, resulting in a metallic negative stamper. The stamper is then used to press multiple copies of the original lacquer, but the pressed material being made of much durable plastic pellets.



Fig. 1.6: The “discs”. From the largest to the smallest: 12" shellac, 78RPM, max. 5 min/side; 10" vinyl, 33⅓ RPM, max. 20 min/side; 7" vinyl, 45 RPM, typically 4-5 min/side, the most ambitious going up to more than 9 min/side [20]; 5" CD, constant linear speed, max. 80 min

Both cutting and playback styli are made of very hard materials, like sapphire and diamond, in order to limit the wear it receives over time. A worn stylus will also damage the groove it’s playing. The cutting styli are specially shaped, such that the recorded groove follows the provided signal as accurately as possible. The cutting stylus is moved by one or two electromagnets. The movement is lateral for monaural recordings or a combination of vertical and lateral (Fig1.10) – for stereo recordings (i.e. each channel is cut 45° from the vertical axis (Fig 1.13)) .

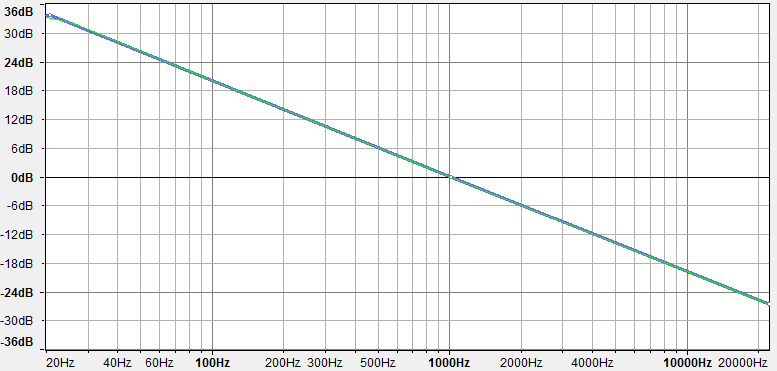
The playback transducers (Fig 1.9), which transform the stylus’ movement into electrical current, are called cartridges. Playback styli shapes are usually less complex, the most common being spherical and elliptical, for they’re much easier and cheaper to produce. More exotic shapes, like Hyper-Elliptical, Shibata, S.A.S and MicroLine exist (Fig 1.16). Their purpose is to track the groove as precisely as possible, resulting in increased sound quality, especially in highly modulated passages. Special shaped styli also wear the grooves much less than conventional styli.

Cartridges are characterized by lots of properties, which summed together show how the cartridge will act at playback time. These specifications include: frequency response, tracking force (the force the stylus puts on the groove), output current, impedance and capacitance, compliance and others. They are also classified by how the electrical current is produced: there are piezoelectric ones (which use piezoelectric materials like Rochelle salt to convert physical movement into electricity), moving magnet (MM, a permanent magnet is moving near fixed coils) and moving coil (MC, coils are moving inside a permanent magnetic field).

Electric cartridges/cutting heads consist of one or more coils and one or more permanent magnets. When the magnetic flux through a coil’s surface changes, Faraday’s law of induction says that the conductor acquires an electromotive force (measured in volts), which is equal to the rate of change of the magnetic flux (Eq. 1.1). Similarly, if a current is passed through a coil, it will generate a magnetic field and move the coil relative to the permanent magnetic field surrounding it.

If we interpret Eq. 1.1 for our case, the induced voltage in a coil is proportional to a coil’s movement relative to the magnets. That means a cartridge’s output voltage is given by the stylus’ speed while tracking a groove. Analog, for a cutter head, the stylus’ displacement (relative to its rest position) is given by the integral over the provided signal’s voltage.

(Eq. 1.1)

Because of this, when cut into the cutting lathe, low frequencies will have much larger amplitudes than high frequencies at the same input signal amplitude. The actual frequency response (groove modulation vs inputted electric signal) is shown in Fig 1.7. To counter this, various equalization curves were used (pre-emphasis before cutting, de-emphasis on playback). Besides reducing the groove’s width (which would permit greater recording times), equalization curves also improve sound quality by amplifying the high frequencies to reduce surface noise and clicks. In 1954, the RIAA equalization became the industry standard for records. It consists of a recording and playback equalization curve which cancel each other out. The playback curve (Fig 1.8) is applied in the phono preamp, a device that boosts the low voltage produced by the cartridge (<10mV) to greater levels (>500mV) so it can be used by an audio receiver such as an amplifier.

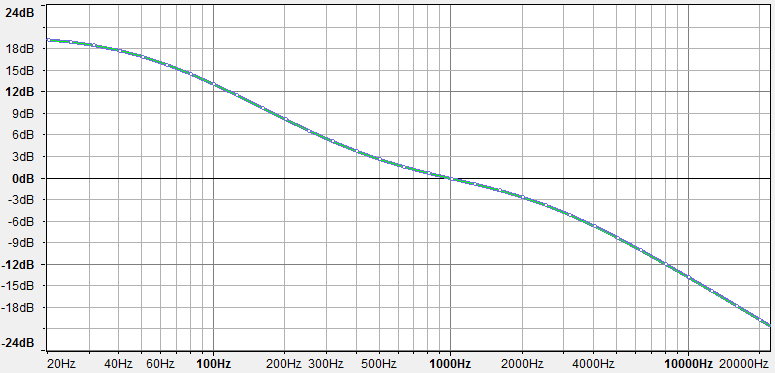
Fig. 1.7: Groove modulation vs inputted electric signal: equivalent frequency response

Fig. 1.8: RIAA playback curve (blue), RIAA recording curve (red)

1.4. Causes of distortion in mechanical analog formats

By using an unadjusted turntable or a poor quality cartridge (worn stylus (yes, even diamonds wear out), big tracking force, low compliance), the record groove wears out (Fig 1.15). Irreversible. This is a big disadvantage of the discs. Also, having the recorded surface always exposed, it is prone to dust and grease accumulation, scratches (Fig 1.12), warps and other permanent damages. These damages will be generally called distortions, i.e. portions that greatly differ from the original state of the media.

The following list contains most of the occurring forms of distortion occurring on vinyl records:

* clicks and pops: when you hear “vinyl record”, you associate it with the sounds of clicks and pops. They are mostly caused by scratches that locally damage the grooves (Fig 1.12). Dust accumulation, electrostatic discharges or manufacturing defects are also causes for them. Clicks have a wide range of periodicities: from a few per record side, up to tens of them every second.
* white noise/hiss: this should not be confused with the inherent surface noise of the recording medium itself. Every play will wear the groove a bit, even if the used stylus is in perfect shape and has an advanced cut. This is because all the tracking force (which varies from 1g up to over 10g for old cartridge types) is concentrated on two very small surfaces: the points the stylus is touching the groove. Big pressures cause great friction amounts, which shears little by little the plastic material the record is made of. If a chipped or worn out stylus is used, the wear it causes to the groove is greatly increased. White noise is usually heard as a continuous sound, lasting from a few hundreds of milliseconds up to whole minutes.
* mistracking: this is where things start to get nasty. In highly modulated passages, when the groove violently wobbles on the two axes, based on the cartridge’s properties, the needle may not be able to accurately track the groove underneath it. This is caused by the large forces it is subjected to, and also the angles it attacks the groove walls with. The stylus will run into the groove wall with greater force than it should, and cause deformations at the place of impact. If things go really bad, the stylus might go up the groove’s wall, losing the contact with the other wall, and start graving a new groove as it digs through the vinyl (Fig 1.11). Important cartridge characteristics when taking mistracking into account are: stylus shape, material and wear, cantilever weight and compliance, tracking force. Also, when approaching the record’s center and the linear velocity decreases, the stylus’ radius in the contact points may become larger than the radius of the groove it touches (Fig 1.14). This is called the pinching effect. The stylus will be pressed into the curve’s innermost wall, causing its deformation, and also a vertical movement in the stylus, as it has nowhere else to go. When occurring, mistracking will cause groove damage even in thousands of points in the time span of a second.

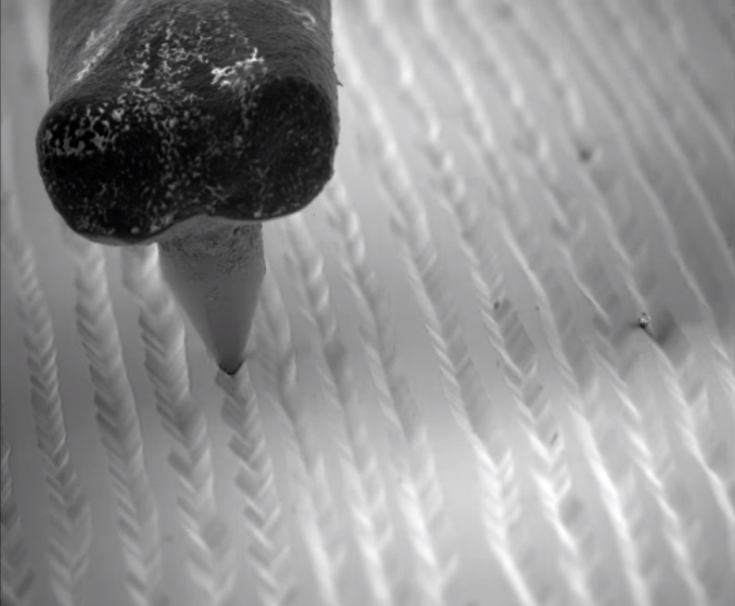


Fig. 1.9: Stylus in a record’s grooves under electron microscope []





Fig. 1.15: Groove wear under microscope [ss]

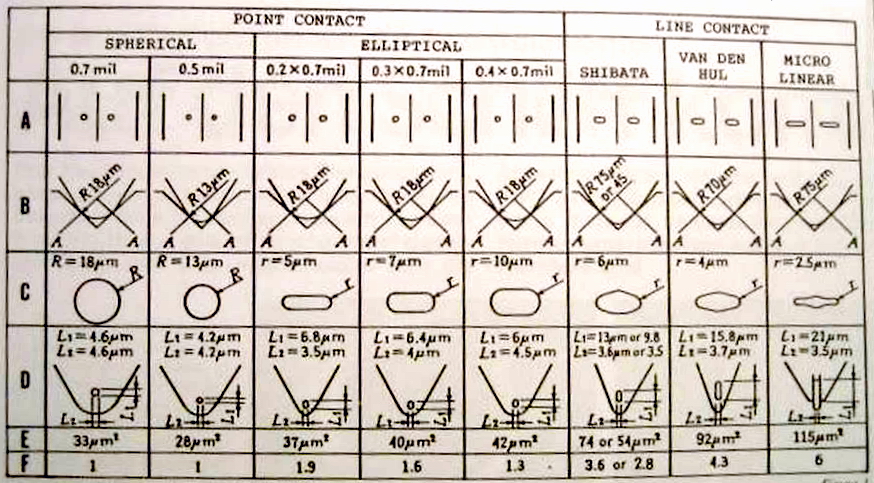
Fig. 1.14: The pinching effect []

Fig. 1.12: Deep scratch causing the well-known pops

Fig. 1.11: Heavily worn mono groove, caused by poor playback equipment

Fig. 1.10: Stereo groove. It moves both in the vertical and horizontal axis

Fig. 1.13: Stylus’ movement in a stereo recording []

Fig. 1.16: Comparison of various styli shapes []

1.5 Purpose of this work

Excessive levels of distortion make the listening unpleasant. If the recording cannot be found anywhere else, either on another recording medium or on a better-shaped record, or it’s simply too difficult to obtain another recording, one would like to recover the original sound or at least bring the current one to an acceptable condition.

The purpose of this work is to create a software application that will automate the process of correcting various distortions occurring at groove level. The targeted distortions will be clicks, pops and some types of mistracking. By using various digital signal processing methods, we want to obtain a sound that resembles as much as possible the original engraved sound.

Chapter 2

Related work (applications)

2.1. Audacity

Audacity is a free, easy-to-use, multi-track audio editor and recorder for Windows, Mac OS X, GNU/Linux and other operating systems [30]. It is open source software written in C and C++. Because of Audacity’s free license, many people have contributed code, bug fixes, documentation and graphics [31]. Audacity provides lots of editing capabilities, from simple effects like amplifying, low- and high-pass filters, equalizers, fade in and fade out, to more complex effects such as speed/pitch change, noise reduction and dynamic compression.

One of the effects is called “Click Repair”, and it seems to promise the user it will solve the problem of clicks appearing on records. The functionality has as inputs two parameters: “Threshold” and “Max Spike Width”, which can be adjusted by the user. Its performance isn’t as impressing as other built-in effects, the results actually being rather poor and often causing more distortion than in the original recording.

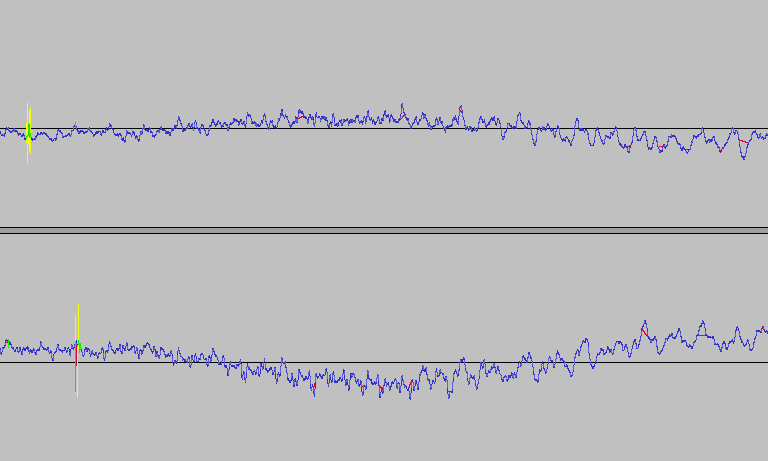
In Fig 2.1, the results of the effect can be seen. With blue – the undamaged parts of the signal; with yellow – the damaged parts; with green – repairs made in damaged regions, with red – repairs made in undamaged regions. It can be seen that even the repairs made in damaged sections are inaccurate, and also there’s lots of “repairs” in undamaged areas. The repairs use linear piecewise interpolation (it traces a straight line from one point to another), which is a very poor choice for audio signal processing purposes.

Fig 2.1: Click Repair effect in Audacity

Audacity also has an effect called “Repair”. It can be used on regions up to 128 samples long and will interpolate the samples from a selected section based on the audio outside the selection. The region length limit is caused by the high computational complexity of the interpolation algorithm, which becomes less accurate and exponentially slower as the selection length increases. Its results are very satisfying and accurately reproduce a bad section based on the surrounding audio. It is understandable why this wasn’t used in the “Click Repair” effect (high computational time), but it’s curious why the developers wouldn’t come up with some less complex but higher quality interpolation technique.

2.2. Nero WaveEditor

Nero 7 Ultra Edition is a software suite that, besides the well known CD and DVD burning facilities, also includes tools for image, audio and video processing. One of those tools is “Nero WaveEditor”, a program for editing and recording audio files. Like Audacity, it provides standard audio processing effects such as equalization, dynamic compression, fading and noise reduction.

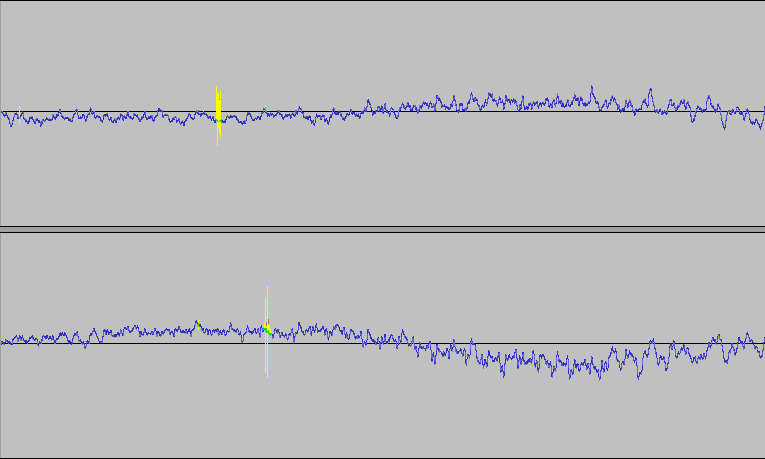
 The discussed effect will be “Declicker”, which also promises to get rid of pops and crackles. This one does its job, and it does it very good! As it can be seen in Fig 2.2, it accurately repairs clicks in not-that-bad records. Yellow represents the damaged sections, green – the reconstructed data, blue – untouched samples. The data reconstruction algorithm is way better than Audacity’s linear interpolation, and the detection of damage is more precise.

Fig. 2.2: Declicker effect in Nero WaveEditor,

applied on the same recording as in Fig. 2.1

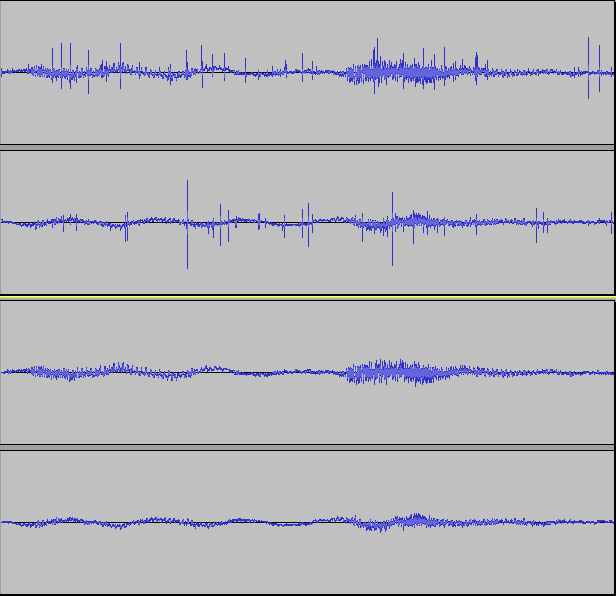
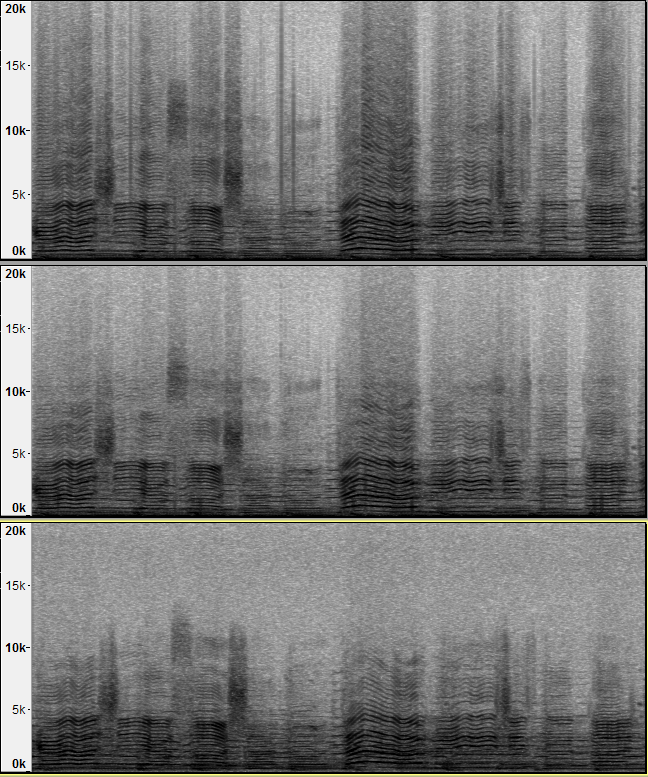
Its capability doesn’t stop here. It can also handle very clicky recordings, like the one in Fig. 2.3. At this level of sound degradation, it’s impossible to exactly reconstruct the recording; some noise will still be there, in a form or another. It’s unrealistic to expect the processed recording to come out as clear as the freshly pressed record was, but the great reduction of distortion does make a huge difference.

Fig. 2.3: Nero WaveEditor’s Declicker applied on a really scratched record.

Top two tracks: before; bottom two tracks: after.

This effect also handles some forms of mistracking pretty good. With high enough sampling rate, it corrects very narrow spaced clicks with outstanding results. This is very similar to the example presented in Fig. 2.3, but with the clicks less intense and way more often. Fig. 2.4 shows a side by side comparison between three versions of the same recording. The top spectrogram is plotted based on the recording of a vinyl record heavily affected by groove wear. The middle spectrogram is obtained from the previous recording after passing it twice through the Declicker. The bottom spectrogram shows the same music as on the vinyl, but this time taken from an audio cassette. Declicker does a nice job here too, but the results it can achieve are limited by the very bad condition of the recording.

Fig. 2.4: Spectral comparison of the same audio: From a worn vinyl, declicker applied on the worn recording, and from a audio cassette.

xxxxxxxxxxx\n

Chapter 3

Basic digital signal processing

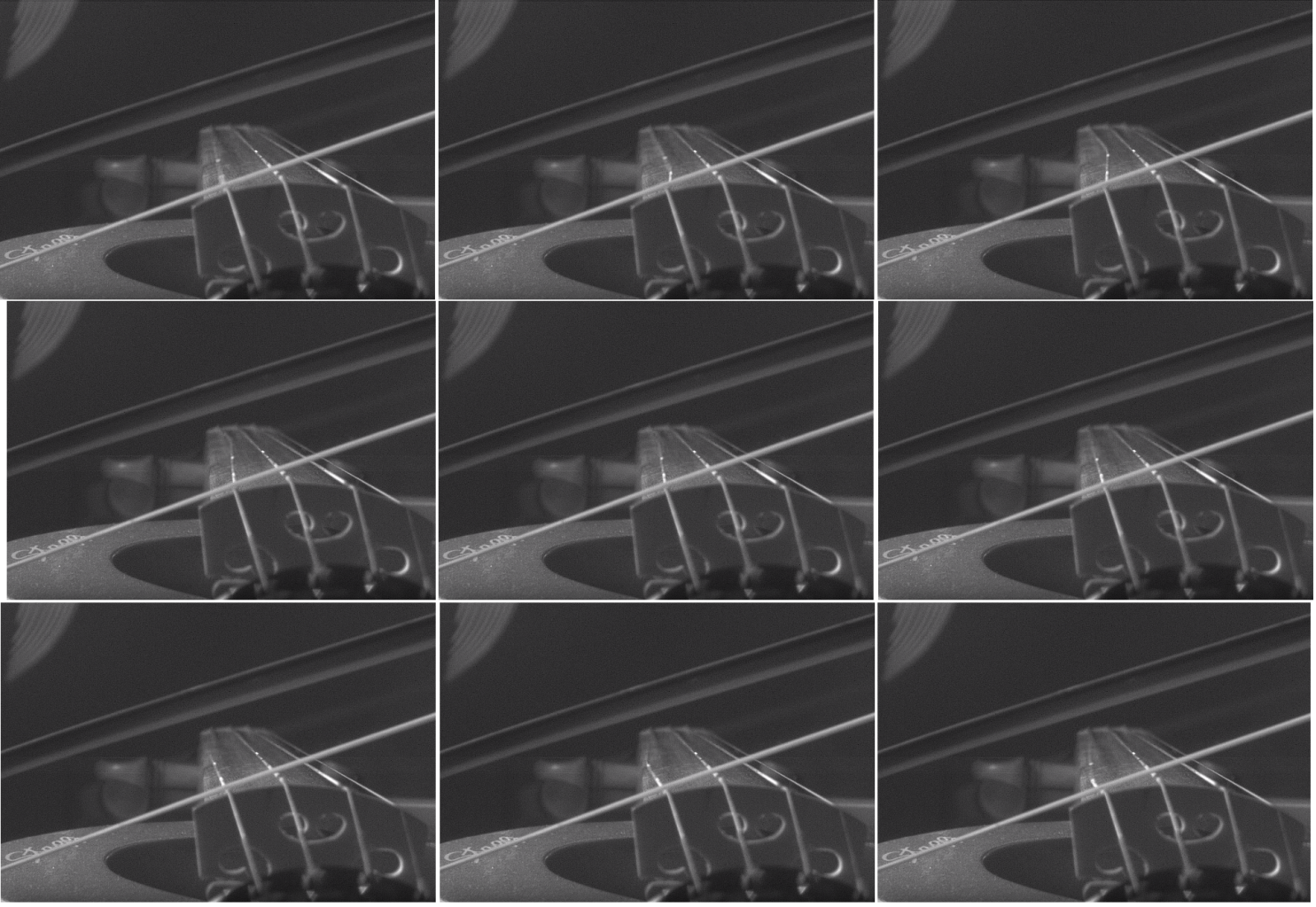
3.1. Digital audio signal representation

Sound is defined as an oscillation in pressure, stress, particle displacement, particle velocity, etc., propagated in a medium with internal forces (e.g., elastic or viscous), or the superposition of such propagated oscillation [1]. This oscillation can be represented as a continuous function that describes the variation in time of the medium’s pressure, allowing us to “see” sounds. (Fig. 3.1)

Sound is transmitted through gases and liquids as longitudinal waves, and through solids both as longitudinal and transverse waves. A transmitting medium is required, so sound cannot travel through vacuum. In a longitudinal wave, the direction of displacement is the same as the direction of propagation, while in a transverse wave, the direction of displacement is perpendicular to the direction of propagation [2]. To better understand the difference between the longitudinal and transversal waves, Fig. 3.1 depicts sound waves in air (longitudinal waves), while Fig. 3.2 presents transverse waves travelling through a metal wire. As seen in Fig. 3.1., a single-frequency sound wave traveling through air will cause a sinusoidal pressure variation in the air. The air motion which accompanies the passage of the sound wave will be back and forth in the direction of the propagation of the sound, a characteristic of longitudinal waves [3]. In Fig. 3.2., the bow’s movement drags the chord, its displacements being transmitted as transverse waves moving along the chord with the chord’s speed of sound.



Fig. 3.1: Sound Waves in Air.

Fig. 3.2. Transverse waves seen in a bowed violin chord. [4]

Audio signals are representation of sound, typically as an electrical voltage (analog) or as discrete numerical values (digital). Conversion from the analog continuous-time signal to the digital discrete-time signal (also called sampling) is made usually with ADCs (Analog to Digital Converters), and with DACs(Digital to Analog Converters) from digital to analog. A **sample** isa signal’s value at a point in time**.** When converting from analog to digital, some of the information is lost because of factors like:

* 1. discretization – the resulted signal is no longer continuous (precisely defined in every point in time), but discrete: the intensity of the analog signal is recorded at fixed time points. The number of equidistant time points in a second is called **sample rate** (or sampling frequency)**.** The highest frequency that can be carried by the signal, called the **Nyquist frequency**, is given by the formula 3.1.;

(3.1)

* 1. storage as finite numbers – as opposed to analog values, the precision of the digital values is finite, so only some of the significant digits can be stored.

Figure 3.3 and Table 3.1 show the process of conversion from an analog signal to a digital one. In Fig. 3.3, a sine wave is sampled 20 times for each cycle. Each sample is then stored as a 8-bit signed integer ( values in [ -128, 127 ] ). Table 3.1 shows the discretization errors raised at the conversion. The digital samples are stored as 8-bit signed integers, with values in [ -128, 127 ], rescaled here to [ -1,1 ) to show the error. Analog samples are in the range [ -1,1 ], where -1 is the smallest possible signal value, and 1 is the maximum. We can see the errors are pretty large for the chosen sample encoding.

By choosing an appropriate sample rate and sample encoding, the lost information (error) can be small enough to be considered negligible.

After getting the sample values from the original analog signal, the samples are then stored as digital numbers in audio files. These sample values can be stored either as uncompressed files, like the WAV and AU formats, which we’ll be discussing about later, or as compressed files (to decrease file size). Compressed file formats can be lossy (the decompressed data is an approximation of the original), such as MP3, or lossless (compression preserves the exact original values ), such as FLAC. Audio files typically contain information about the sampling rate, number of channels and sample encoding (float/integers, signed/unsigned, bit-depth, companding and others).



Fig. 3.3. Conversion from continuous-time to discrete-time.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Sample no. | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| Actual value | 0,00000 | 0,30902 | 0,58779 | 0,80902 | 0,95106 | 1,00000 | 0,95106 | 0,80902 | 0,58779 | 0,30902 | 0,00000 |
| Sampled value | 0,00000 | 0,30469 | 0,58594 | 0,80469 | 0,94531 | 0,99219 | 0,94531 | 0,80469 | 0,58594 | 0,30469 | 0,00000 |
| Error | 0,00% | 0,43% | 0,19% | 0,43% | 0,57% | 0,78% | 0,57% | 0,43% | 0,19% | 0,43% | 0,00% |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Sample no. | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| Actual value | -0,30902 | -0,58779 | -0,80902 | -0,95106 | -1,00000 | -0,95106 | -0,80902 | -0,58779 | -0,30902 | 0,00000 |
| Sampled value | -0,31250 | -0,59375 | -0,81250 | -0,95313 | -1,00000 | -0,95313 | -0,81250 | -0,59375 | -0,31250 | 0,00000 |
| Error | 0,35% | 0,60% | 0,35% | 0,21% | 0,00% | 0,21% | 0,35% | 0,60% | 0,35% | 0,00% |

Table 3.1. Errors at conversion from analog to digital samples.

3.2. Filters: Finite Impulse Response and Infinite Impulse Response

Signal filtering is one of the main applications of signal processing [[5], p.185]. Filters are used for many purposes, but, in audio signal processing, they are mostly used to achieve a desired frequency response, the most basic being low-pass (which only let low frequencies pass), high-pass (same for high frequencies), band-pass and band-stop filters. Low- and high-pass filters are widely used in speaker cabinets to separate the input signal to each of the drivers, depending on drivers’ frequency response. An example for use of band-pass filters is in radio communication, to isolate the required frequency band (representing a radio channel) from the others.

Depending on the considered audio signal, there are two types of filters: analog (electronic) and digital. Electronic filters can range from simple circuits, made just from simple passive components: resistors, inductors and capacitors, to complex ones, including along the usual passive components, active components: transistors, amplifiers, operational amplifiers and others.

The digital signal processing systems use samples of input signals, which constitute series of numbers. The result may be also series of numbers, to be used as output signals [[5], p. 239]. Each sample of the output signal is computed as a weighted sum of the previous few input and output samples. This weighted sum is also known as a convolution. There are two primary types of digital filters: FIR (Finite Impulse Response) and IIR (Infinite Impulse Response). The impulse response of a system is its output when presented with a brief input signal, called an impulse. Applying a filter to a signal will alter each frequency’s magnitude and phase according to the filter’s impulse response. It remains to design a filter, knowing the desired response, which we’ll discuss in the next section.

When using a FIR filter, each output sample is computed from the previous input samples and a fixed set of coefficients, the weights. The weights are associated to input samples based on each sample’s delay (how far it is from the most recent input sample). Input samples, ordered by delay, are stored in a so-called delay line. The resulted sample is calculated by a set of multiply-accumulate operations: each is input sample is multiplied with its weight, and summed to the output sample:

(3.1)

where:

• is the input signal,

• is the output signal,

• is the filter order and the number of taps. A “tap” is simply a coefficient/delay pair [6]; an th-order filter needs N previous input samples and has terms on the right-hand side,

• is the set of the coefficients; is the weight associated to the th input sample.

For an FIR filter, the filter coefficients are, by definition, the impulse response of the filter [6]. The impulse response is finite ( when the input samples become 0, the output will eventually become 0 ) because there is no feedback in the FIR. A lack of feedback guarantees that the impulse response will be finite [7].

The other class of digital filters is IIR filters which, unlike FIR filters, use feedback, i.e. samples already computed by the filter are used in the next iterations. Each output sample is computed from the previous input sample (feedforward), but also from the previous output samples (feedback). Like FIR filters, each sample has an associated weight, based on the sample’s delay, so there are needed two sets of coefficients: feedforward and feedback. The formula is as following:

(3.2)

In many digital signal processing applications, FIR filters are preferred over their IIR counterparts. The main advantages of the FIR filter designs over their IIR equivalents are the following [8]:

1. FIR filters with linear phase response (all frequencies are delayed by the same constant time amount) can easily be designed;
2. They are simple to implement (two nested loops, one for iterating the input samples, one for the coefficients);
3. FIR filters are always stable, i.e. given a bounded input signal, the output signal will also be bounded. If designed wrong, IIR filters can diverge.
4. They have desirable numeric properties. The use of finite-precision arithmetic in IIR filters can cause significant problems due to the use of feedback, but FIR filters can usually be implemented using fewer bits, and the designer has fewer practical problems to solve related to non-ideal arithmetic [7];
5. Excellent design methods are available for various kinds of FIR filters with arbitrary specification [8].

The main disadvantage of FIR filters is that they may require much more computational effort and memory than a comparable IIR counterpart.

Figures 3.4 and 3.5 give an example as how an FIR filter and IIR filter work along the input data:

Fig. 3.4: Application of a digital FIR filter



Fig. 3.5: Application of a digital IIR filter

3.3. Frequency domain and Fourier transforms

As we previously saw, an audio signal is represented as a variation of intensity over time (being it air pressure or electric voltage). The human auditory system picks up the eardrum’s vibrations and transduces them into nerve impulses, which are then perceived in the brain as “sound”. However, the brain doesn’t interpret the sound by its pressure wave, but rather by its frequencies’ amplitudes, phase and pitch. For example, an audible sine wave will be perceived not as the periodic function the sin function looks like, but as a pure tone, with a certain pitch and loudness.

The function that gives the variation of wave’s intensity is called **time-domain.** The function that gives the component frequencies is called **frequency-domain.**

In 1822, Fourier in his work on heat flow made a remarkable assertion that every function f(x) with period 2π can be represented by a trigonometric infinite series of the form given by formulas 3.3, 3.4 and 3.5.

(3.3)

[[9], p.1]

An infinite series of this form is called a Fourier series [[9], p.1]. an and bn are called the Fourier coefficients. Using the Fourier series, one can transform from time-domain to frequency domain and vice-versa. Fig. 3.6 shows the same sound, plotted in time-domain and in frequency-domain. In the time-domain plot, x-axis is time, y-axis is signal’s amplitude. In the frequency-domain plot (spectrogram), x-axis is time (the signal was partitioned into wavelets), y-axis is frequency, and z-axis (color intensity) is amplitude. A note played on flute has the fundamental considerably louder than the harmonics, so the sound wave generated look pretty close to a sine wave. The spectrogram clearly shows the harmonics.



background noise

fundamental (1st harmonic) – 880 Hz

2nd harmonic – 1760 Hz

3rd harmonic – 2640 Hz

4th harmonic – 3520 Hz

5th harmonic – 4400 Hz

Fig. 3.6.: Note A5 (880Hz) played on a flute. Top – plot of the time-domain function. Bottom – plot of the frequency-domain function.

As we’ll do digital signal processing, we’ll not be working with continuous functions, but rather with discrete ones. A DFT (Discrete Fourier Transform) is a Fourier that transforms a discrete number of samples of a time wave and converts them into a frequency spectrum. The spectrum of the signal from Fig. 3.6 can be seen in Fig. 3.7. This time, a single wavelet covering the whole shown signal was input into the Fourier transform. The opposite of DFT is the IDFT (Inverse Discrete Fourier Transform), which transforms the frequency spectrum to a discrete number of samples. The equation for the Discrete Fourier Transform is:

where F(n) is the amplitude at the frequency n, and N is the number of discrete samples taken. Note that:

i.e. each frequency has a cosine and a sine component, each with its own amplitude.

The IDFT formula looks similar to the DFT:



Fig. 3.7. Spectral plot of the same signal from Fig. 3.6. Arrow point to sound’s harmonics.

3.4. Filter design and frequency equalization

We’ll want to use filters, among other uses, to change the output signal’s frequency response, i.e. to apply an equalization curve ( a set of frequency-gain pairs ). In this section, we’ll discuss how to create a FIR filter from a given equalization curve. The problem stands in calculating its coefficients based on the desired frequency and phase response.

For the design of FIR filters, there have been developed several methods:

1. Direct Calculation: In the case of some types of filters, such as high-pass and low-pass filters, their coefficients can be directly calculated from formulas. For example, the ideal low-pass filter follows the function, as shown in Fig. 3.8.
2. Parks-McClellan: The Parks-McClellan method is probably the most widely used FIR filter design method. It is an iteration algorithm that accepts filter specifications in terms of passband and stopband frequencies, passband ripple, and stopband attenuation. The fact that you can directly specify all the important filter parameters is what makes this method so popular [11].
3. Windowing: Using the property that, the DFT of the impulse response gives the filter’s frequency response, we can calculate the coefficients by applying the IDFT on the wanted response. After that, the impulse response can be refined by applying a windowing function.



Fig. 3.8.: The ideal low-pass filter, the sinc function.. With blue, the FIR coefficients following the sinc function. With red, the coefficients after applying a Blackman window.

Chapter 4

Detecting and correcting distortion

4.1. Automated marking of distorted samples using neural networks

As stated previously, the purpose of this work is to create a software application that will automate the process of correcting various distortions in recordings on the analog format of vinyl records. The targeted distortions will be clicks, pops and some types of mistracking. The chosen approach in solving this problem consists of two steps: identifying the damaged portions and reconstructing them.

Identifying the damaged portions means, in this case, creating a set of non-overlapping intervals for a given audio sequence. Each of these intervals (further called “marking”) represents a portion of the signal which is considered to be damaged and needs correction. Finding them is not a deterministic job, as there’s no way to precisely differentiate between what is signal and what is noise. At the end, this whole marking process can be viewed as classifying each sample of the digital audio signal as either “marked”/damaged or “unmarked”/not damaged.

A possible approach for classifying data is using artificial intelligence. The intelligent agent, i.e. the system that will solve this classification problem, will have as input a sequence of samples of odd length, and as output – one of the two possible labels: “marked” or “not marked”. In the input data, the sample in the middle is the one being classified, while the rest of them are used to help decide the output.

Before an intelligent agent can be used, it needs to be trained. A training set, a set of pre-labeled inputs, is used for both the training and validation. Validation is made using the trained system to label already classified inputs in order to calculate the system’s accuracy.

Before a system can be trained, big enough training sets are needed. As the chosen number of inputs is quite big – 129(each central sample has 64 samples to the left and 64 to the right), a good training set must have at least a few tens of thousands of examples. Thus, generating the training sets by manually marking samples is a no-go. A method that proved to be very efficient and accurate was to use monaural records. By using a stereo cartridge to pick up the sound from a mono groove, one would expect the left and right channels to be identical. This isn’t the case when distortion occurs, as it’s almost sure it will affect the channels differently. Using the assumption that every difference big enough between the channels is distortion, large and accurate datasets can be quickly produced. In fact, all the trainings done in the following examples use solely training sets generated this way. Table 4.1 lists the training sets used in this work: the record, its condition, dataset size, type of music. All sets use a 96 000 samples / second sampling rate.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Dataset name | Record label/series | Record condition | | | Dataset size | Music type |
| Scratches | Mistracking | Groove wear |
| Shostakovich | Мелодия НД-02243 track A2 | moderate density, small intensities | no | moderate |  | Strings, Percussion, Brass, Winds |
| Chopin | Международная Книга 33Д-0008830(а) | high density, small intensities | moderate | no |  | Piano |
| Beethoven | The Classics Record Library MAQ 3333 track B4 | moderate density, medium intensities | no | moderate |  | Strings |
| Andries | Electrecord EDE 03765 track B2 | moderate density, small intensities | light | light |  | Guitar, vocals |
| Salesbury | Ryemuse RP 7016 tracks A1, A2 | moderate density, medium intensities | moderate | light |  | Organ |
| Dvorak | Musical Masterpiece Society MMS-121 track B2 | moderate density, small intensities | light | moderate |  | Strings, Percussion, Brass, Winds |
| Enescu | Electrecord ECD 23 track A | small density, small intensities | heavy | no |  | Strings, Percussion, Winds |

Table 4.1: The datasets used in this work for intelligent agent training

There are many intelligent classification methods. Table 4.2 shows a comparison between various classifiers and their precision. The training dataset used for this example is a subset of length 20 000 of the “Shostakovich” dataset. It can be seen that KNN and CART were the most accurate, but they were discarded as viable options due to the very high training times for the relatively small training set.

|  |  |  |
| --- | --- | --- |
| Classifier | Accuracy | Training time (s) |
| Logistic Regression (LR) | 0.579 | 8.62 |
| Linear Discriminant Analysis (LDA) | 0.572 | 2.91 |
| K-Neighbors Classifier (KNN) | 0.921 | 90.25 |
| Decision Tree Classifier (CART) | 0.838 | 45.28 |
| Gaussian Naive Bayes (GNB) | 0.711 | 0.70 |
| C-Support Vector Classification (SVC) | ? | > 300, timeout |

Table 4.2: Performance of various classification methods

Much better performances were obtained by using Artificial Neural Networks. Table 4.3 lists ANN training results on various combinations of datasets, signal source and pre-processing, input sizes and ANN parameters. The possible signal sources are the following:

* groove modulation – the physical movement of the stylus as it follows the groove
* pre RIAA – the current produced by the cartridge, without the RIAA equalization
* direct pick-up – the current produced by the cartridge, passed through the RIAA stage

As for signal pre-processing, a high-pass filter may or not be applied. Its purpose is the removal of the lower frequencies, which are not affected by distortion and thus might be redundant for the intelligent system.

Based on the results presented in Table 4.3, it’s been decided that a good combination of NN configuration and input data would be the following: pre RIAA signal source, no high-pass, ( 129, 64, 32, 1 ) layer configuration, standard scaling, shuffling on training, tolerance = 0.00001.

4.2. Extrapolation and linear prediction

Having marked the bad sections, the remaining step is to reconstruct the samples in those regions.

X

Table 4.3.

Chapter 5

Application

6.1. Requirements and specification

The application must provide the following features:

1. Be able to read audio data from audio files. The required format types are WAV and AU files, which are both uncompressed formats. Complete format support isn’t necessary, but support for Linear Pulse Code Modulation encoding and arbitrary sample rates, channel numbers, byte depths, file lengths is mandatory.
2. Use in-memory cache for the audio data, in order to avoid multiple reads/writes to the disk.
3. Be able to execute basic signal processing operations: applying arbitrary FIR and IIR filters on arbitrary segments of the audio signal.
4. Have versioning on an audio signal processing project: Starting from the original file, each alteration of it shall create another project version. The original file will never be written into; all alteration must be stored in auxiliary files. Undo and redo capabilities are necessary. Each alteration will be depended on the previous ones, so writing operations done in the auxiliary files shall not affect in any way the previous versions.
5. Apply an equalizer, i.e. a signal processor that will change the signal’s frequency response by given criteria. The equalizer can be implemented as a FIR, whose coefficients can be calculated based on the needed frequency response.
6. Provide means of applying linear prediction. The coefficients will be calculated based on known data with Burg’s algorithm.
7. Repair segments of signal, based on the adjacent data, by using a combination of forward and backward linear prediction.
8. Work with “markings”, i.e. set of intervals that mark portions of the signal that is considered to be damaged and needs reconstruction.
9. Generate the set of “markings” for a given interval of audio signal.
10. Use the repair feature on all the “markings” inside a certain interval.
11. Be able to compute Fast Fourier Transform and Inverse Fast Fourier Transform.
12. Generate a FIR from a given frequency response
13. Be able to interpolate 1-D functions
14. Be able to apply windowing functions on arrays
15. Provide an easy to use and intuitive GUI

Specifications:

1. Audio data: for storage (in memory), it will be stored into “Audio Samples Windows”: matrixes of floating-point values; each channel has a row assigned, each row contains the audio samples. Metadata such as the interval the window maps to the project, the number of channels, sample rate must be included.
2. Audio files: for the WAV and AU formats, first, metadata will be read from the header and store in memory. While reading the header, the program must check for any inconsistencies. After that, each read/write operation will randomly access the file, read/write the necessary bytes, convert them to/from floating-point values (samples) and output/input an “Audio Samples Window”.
3. Cache: The purpose of it is to minimize the accesses to the disk. Caching policy will be **Least recently used (LRU).**
4. For FIR and IIR filters, as well as the linear prediction, the coefficients must be floating-point.
5. The program must implement an efficient data-structure for a set of non-overlapping intervals.
6. The application must have a well defined architecture that respects architectural design patterns. It must be easily extensible (adding of new modules), follow coding standards and have a well-developed exception treatment system.
   1. Audio Data Sources

6.2.1 File Storage. WAV and AU formats

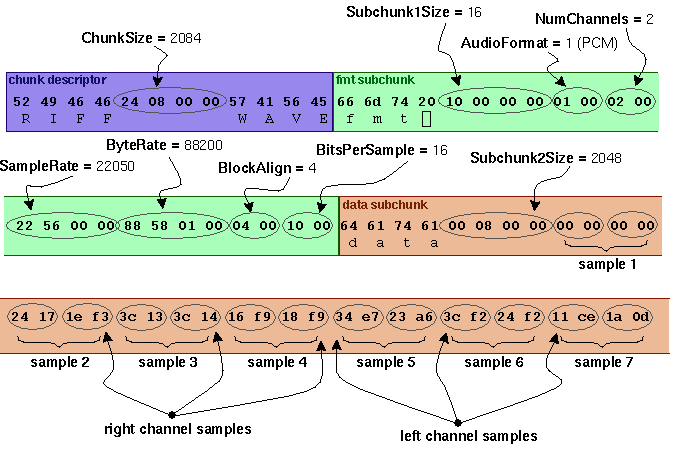
An Audio Data Source (ADS) is an abstract place where audio data can be read from/written to. It also provides means of getting metadata like sample-rate, bit-depth, number of channels and others. An ADS can be stored in memory, in a file or in multiple files. A small continuous fragment of audio, called an Audio Samples Window (ASW), holds a number of samples, divided into channels. Each ASW is aware of the number of channels and samples it stores, as well as where it is located in the project.

ASWs are the basic form of transmitting audio data inside the application. ADSs return ASWs for “get” operations, require ASWs for “put” operations. The low-level signal processing algorithms use number arrays as input/output, but their high-level counterparts, which also take care of things like fragmenting the task into smaller pieces, mainly use ASWs.

The application will work starting from an already existent audio file, and after being done editing, the user would want to save the edited sound to a file on a persistent media. Thus, the application must be capable of reading and writing audio files.

Even if some file formats compress the audio data to take up less space, the processing of audio signal requires that reads/writes are made as efficient as possible, so only uncompressed formats will be used in the editing process. The file types supported by this application are the WAV format and the AU format. Both are uncompressed, LPCM formats, and make use of very simple structures, as seen in Fig. 6.1. The only notable difference between the two formats is that AU has a simpler and non-redundant header.

The process of using a file as ADS is quite simple. First, read the header and keep the metadata in memory. Reading samples is done by reading bytes from a certain position in file, converting them to numerical values (samples), and returning them as arrays. Writing does the exact opposite: reads samples from arrays, converts them to bytes which are then written to the file in the right position. If after writing, the total number of samples is bigger than before, an update of the header may be necessary.

Fig. 6.1: Structure of a WAV file

* + 1. Caching

As it is well known in the industry, read/write operations on disk are significantly slower than in memory. To reduce the number of hard disk access operations, a method of caching audio data in memory is required. As well as allowing faster access to some portions of the project’s audio signal, a cache helps to reduce the number of consecutive small-length reads/writes in neighbor sections of a file by condensing them into a single, larger-length operation.

* + 1. Version control on an audio project

While editing an audio project, multiple modifications can be made one after another, and the user will want to have the possibility of undo and redo actions. Each modification creates a new “version” of the audio project. Audio data can get big enough to become impractical to be kept all in memory. Raw audio data, at CD quality ( 16 bits, 2 channels, 44100 samples/second ) will take up over 600 MB for an hour of recording, so having a copy of the whole project each time a change is made could take up a lot of HDD space.

To avoid unnecessarily using disk space, a scheme for linking each version of the project to its parent was thought. It uses auxiliary files and project-to-file mappings in order to only store the differences between a version and its parent. This is as well an Audio Data Source, and is called “Versioned Audio Data Source”. Each step of the project’s editing process is an “ADS Version”. The algorithm that manages a Versioned ADS goes as following:

* Initialization of a Versioned ADS can be either from scratch (0 samples stored) or from an already existing audio file.
* All the versions present are kept in an ordered list; each version has a parent, excepting the first version.
* There’s a cursor that points to the current version used by the project. Moving the cursor up and down the version hierarchy basically implements the undo-redo system.
* Read/Writes to the Versioned ADS are redirected to the version currently being pointed to. Then, for “get” operations, the version uses its project-to-file mapping to read the required data from files, and for “put” operations – it alters the mapping accordingly.
* A version can have up to one child. When creating a version as a child of another version, all of its siblings are deleted recursively. The new child then copies its parent’s project-to-file mapping.

The mentioned auxiliary files, which store all the audio data modified since the first version, and project-to-file mappings are used at “ADS Version” level. There’s also a component that manages the life-cycle of the auxiliary files. It keeps, for each temporary file, a list of ADS Versions that refer that file. The functionalities it provides are creating a new temporary file and mapping a portion of the project’s audio data to a portion of a file. When a file is no longer referred by any ADS Version, it can be safely deleted.

XXX

Bibliography

[1]. American National Standard on Acoustical Terminology, ANSI/ASA S1.1-2013

[2]. Definitions of longitudinal and transverse waves, <http://www.dictionary.com>

[3]. Sound waves in air, <http://hyperphysics.phy-astr.gsu.edu/hbase/Sound/tralon.html>

[4]. Bowed violin string in slow motion, <https://www.youtube.com/watch?v=6JeyiM0YNo4> (11.04.2018)

[5]. Jose Maria Giron-Sierra: *Digital Signal Processing with Matlab Examples, Volume I,* Springer, ISBN 978-981-10-2534-1

[6]. FIR Filter Design: <http://shodhganga.inflibnet.ac.in/bitstream/10603/24055/10/10_chapter%205.pdf>

[7]. FIR Filter Basics: <http://dspguru.com/dsp/faqs/fir/basics/>

[8]. Finite Impulse Response Filter Design:

<https://www.cs.tut.fi/~ts/Mitra_Kaiser.pdf>

[9]. Tang, Kwong-Tin: Mathematical Methods for Engineers and Scientists 3. Fourier Analysis, Partial Differential Equations and Variational Methods, 2017

[10]. Discrete and Fast Fourier Transforms: <http://www.alwayslearn.com/DFT%20and%20FFT%20Tutorial/DFTandFFT_BasicIdea.html>

[11]. FIR Filter Design: <https://dspguru.com/dsp/faqs/fir/design/>

[12]. Furnace Record Pressing Plant Is Ready To Roll, <https://blog.discogs.com/en/furnace-record-pressing-plant/> (grabbed 2018.04.27)

[13]. <https://qz.com/103785/hipsters-are-buying-vinyl-records-but-they-arent-listening-to-them/>

[14]. <https://www.billboard.com/articles/columns/chart-beat/7662572/us-cassette-album-sales-increase-2016-guardians>

[15]. Phonoautograph: <https://en.wikipedia.org/wiki/Phonautograph>

[16]. Léon Scott de Martinville’s phonoautograph: <http://ginsteve.e-monsite.com/blog/un-dimanche-une-decouverte/9-avril-1860-leon-scott-de-martinville-realise-le-tout-1er-enregistrement-sonore.html>

[17]. Phonoautogram: <https://www.sfgate.com/news/article/Physicists-convert-first-known-sound-recording-3289341.php#photo-2437531>

[18]. Edison's Impression: Laying Sound into a Groove, <https://www.youtube.com/watch?v=0vbyoZDQaIY>

[19]. Emile Berliner's Fix: Flatten the Cylinder to a Disc <https://www.youtube.com/watch?v=w_g4cAXkz80>

[20]. Wolfgang Amadeus Mozart – Eine Kleine Nachtmusik KV 525, 7" 45RPM vinyl, Deutsche Grammophon 30 053 EPL

[21]. John Pfeiffer: *Quality by the numbers*, Liner notes of *RCA Red Seal Digital ARC1-3459 (Stravinsky – The Firebird Suite (1919), Symphony in three movements )*

[22]. Retro Tech: The Wire Recorder: <https://www.youtube.com/watch?v=90ihiTwJPCc&t=71s>

[23]. Vertical-cut (cylinder) vs lateral-cut (Berliner disc): <http://www.soundfountain.com/amb/ttadjust.html>

[24]. Modern record movement axes: <https://www.psaudio.com/pauls-posts/the-great-mystery-of-vinyl/>

[25]. Wear in the loud passages of a mono vinyl record: <http://www.micrographia.com/projec/projapps/viny/viny0200.htm>

[26]. Groove pinching: <https://i.stack.imgur.com/wgerL.png>

[27]. Electron microscope slow-motion video of vinyl LP: <https://www.youtube.com/watch?v=GuCdsyCWmt8>

[28]. Advanced Stylus Shapes: Pics, discussion, patents <https://www.vinylengine.com/turntable_forum/viewtopic.php?f=19&t=22894>

[29]. <https://nationalmaglab.org/education/magnet-academy/watch-play/interactive/electromagnetic-induction>

[30]. <https://sourceforge.net/projects/audacity/>

[31]. <http://manual.audacityteam.org/man/faq_about_audacity.html>

[]. A tutorial on Burg's method, algorithm and recursion, Cedrick Collomb, 2009:

<http://www.emptyloop.com/technotes/A%20tutorial%20on%20Burg's%20method,%20algorithm%20and%20recursion.pdf>

[]. Jose Maria Giron-Sierra: *Digital Signal Processing with Matlab Examples, Volume II,* Springer, ISBN 978-981-10-2537-2

[]. Jose Maria Giron-Sierra: *Digital Signal Processing with Matlab Examples, Volume III,* Springer, ISBN 978-981-10-2540-2

[]. FIR Filter Properties: <http://dspguru.com/dsp/faqs/fir/properties/>

[]. IIR Filter Basics: <http://dspguru.com/dsp/faqs/iir/basics/>