



ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

Fakulta elektrotechnická

Katedra radioelektroniky

Rozpoznání titulu gramofonové desky podle krátké ukázky

Vinyl Record Title Identification Based on Short Sound Sample

Václav Moldan

Bakalářská práce

Studijní program: Komunikace, multimédia a elektronika

Studijní obor: Multimediální technika

Vedoucí práce: Ing. František Rund, Ph.D.

Praha 2017

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Moldan** Jméno: **Václav** Osobní číslo: **435035**
Fakulta/ústav: **Fakulta elektrotechnická**
Zadávající katedra/ústav: **Katedra radioelektroniky**
Studijní program: **Komunikace, multimédia a elektronika**
Studijní obor: **Multimediální technika**

II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

Rozpoznání titulu gramofonové desky podle krátké ukázky

Název bakalářské práce anglicky:

Vinyl Record Title Identification Based on Short Sound Sample

Pokyny pro vypracování:

Seznamte se s problematikou rozpoznávání a identifikace hudebních děl. Implementujte systém pro identifikaci gramofonové desky na základě analýzy krátké zvukové ukázky z této desky s využitím vhodné metody strojového učení. Při návrhu systému zohledněte typické artefakty mechanického záznamu. Otestujte úspěšnost navrženého systému.

Seznam doporučené literatury:

[1] Král, V. Identifikace gramofonové desky podle krátké ukázky. Bakalářská práce FEL ČVUT v Praze, 2016
[2] Lartillot, O.; Toivainen P., A Matlab Toolbox for Musical Feature Extraction from Audio, 10th. Int. Conf. on Digital Audio Effects, Bordeaux, 2007.

Jméno a pracoviště vedoucí(ho) bakalářské práce:

Ing. František Rund Ph.D., katedra radioelektroniky FEL

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) bakalářské práce:

Datum zadání bakalářské práce: **13.02.2017** Termín odevzdání bakalářské práce: **26.05.2017**

Platnost zadání bakalářské práce: **31.08.2018**

Podpis vedoucí(ho) práce

Podpis vedoucí(ho) ústavu/katedry

Podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Student bere na vědomí, že je povinen vypracovat bakalářskou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v bakalářské práci.

Datum převzetí zadání

Podpis studenta

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze dne 23. 5. 2017

Václav Moldan

Poděkování

Chtěl bych poděkovat vedoucímu práce Ing. Františku Rundovi, Ph.D. za cenné rady a pomoc při vypracování této práce. Zároveň bych chtěl poděkovat firmě GZ Media, a.s. za poskytnutí nahrávek pro testovací účely.

Abstrakt

Tato práce se zabývá problematikou rozpoznávání a identifikace zvukové nahrávky. Cílem je seznámení s problematikou a implementace programu v prostředí MATLAB, který bude schopen identifikovat gramofonovou desku podle krátké zvukové ukázky. Nejdůležitější kroky v algoritmu jsou parametrizace a klasifikace, klasifikace se provádí pomocí metody strojového učení K-nearest neighbors (KNN). Práce se zabývá také vlastnostmi gramofonových desek, které mohou mít na identifikaci vliv. Funkčnost realizovaného algoritmu je ověřena na nahrávkách dodaných zadavatelem. Po natrénování na množině všech 269 dostupných masterů nahrávek byl program schopen rozpoznat 92,8 % nahrávek uměle znehodnocených atributy gramofonové desky (posunutí začátku, snížení amplitudy, přidání šumu a ztráta vysokých frekvencí) a zároveň všech 5 dostupných nahrávek pořízených z gramofonové desky.

Klíčová slova: identifikace, audio nahrávka, parametrizace, klasifikace, strojové učení, KNN, akustický otisk, gramofonové desky, MATLAB

Abstract

The main subject of this thesis is audio records recognition and identification. The aim is to introduce this subject and to implement algorithm in MATLAB programming environment, which will be able to identify vinyl disc based on a short audio record sample. The most important parts of the implemented algorithm are parametrization and classification, classification is performed with K-nearest neighbors machine learning algorithm. The thesis also deals with common attributes of vinyl discs, which can affect the process of identification. The functionality of the implemented algorithm is tested on audio records, which were provided for the purpose of this thesis. After being trained with set of 269 available master records, algorithm was able to successfully identify 92,8 % records with artificially simulated attributes of vinyl records (extra silence added before the start of the record, lower level of volume, additional noise and loss of high frequencies) and also all 5 records which were recorded from real vinyl discs.

Key words: identification, audio record, parameters, classification, machine learning, KNN, acoustic fingerprint, vinyl discs, MATLAB

Obsah

Úvod	13
1 Identifikace audio signálu.....	15
1.1 Akustický otisk.....	15
1.2 Parametrizace.....	15
1.2.1 Tempo	15
1.2.2 Mode	16
1.3 Klasifikace	16
1.3.1 KNN.....	16
1.3.2 ANN	16
2 Dostupné technologie pro identifikaci nahrávek.....	19
2.1 Shazam.....	19
2.2 SoundHound	20
2.3 Ekaterina Koshkina: Identifikace obsahu archivních zvukových záznamů	21
2.4 Václav Král: Identifikace gramofonové desky podle krátké zvukové ukázky	21
3 Gramofonové desky	23
3.1 Snížení frekvenčního rozsahu	23
3.2 Přeslechy.....	23
3.3 Šum	23
3.4 Zkreslení.....	25
3.5 Impulsní vady.....	25
4 Implementace algoritmu v prostředí MATLAB.....	27
4.1 Testovací množina nahrávek	27
4.2 Klasifikace	27
4.2.1 Klasifikace pomocí metody KNN	27
4.2.2 Trénování klasifikátoru.....	28
4.3 Parametrizace.....	28
4.3.1 Výběr parametrů	28
4.3.2 Implementace hierarchie parametrů.....	29
4.4 Předzpracování nahrávky	29
4.4.1 Převzorkování.....	29
4.4.2 Segmentace.....	29
4.4.3 Zkrácení nahrávky	30

4.4.4 Převedení nahrávky do libovolné reprezentace signálu (MS, stereo, mono)	33
4.5 Simulace vlastností vinylových desek	34
4.5.1 Přidání šumu, snížení amplitudy	34
4.5.2 Posunutí začátku	34
4.5.3 Úprava frekvenčního rozsahu	34
4.5.4 Další atributy gramofonové desky	38
4.6 Testování algoritmu	39
4.7 Schéma vytvořeného programu, řízení programu funkcí main.....	40
4.8 Výpočetní náročnost.....	42
4.9 Srovnání metod klasifikace.....	42
Závěr	45
Literatura	47
Přílohy.....	49

Úvod

V této práci jsem se zabýval problematikou rozpoznání zvukové nahrávky a i implementací vlastního algoritmu, který bude schopný tuto úlohu řešit.

V první části práce se věnuji teorii identifikace obsahu nahrávky, v druhé části pak implementaci vlastního algoritmu v prostředí MATLAB.

Problematika identifikace zvukových nahrávek je v dnešní době velmi aktuální. Motivací pro návrh tohoto algoritmu je využití v průmyslu při výrobě nosičů ve velkém množství, kdy snadno může nastat případ, kdy je potřeba identifikovat neoznačený nosič. Program, který by umožnil rychle tuto desku identifikovat, by zde byl vítanou pomůckou.

Podobné systémy mají ale uplatnění i pro běžného posluchače, který by rád rychle zjistil název skladby, kterou slyší například z rádia. Převážně proto se stávají populárními i různé další programy (často ve formě mobilních aplikací), které identifikaci umožňují. V teoretické části je nastíněn princip činnosti nejznámějších aplikací. Zároveň teoretická část seznamuje i s dalšími studentskými pracemi na FEL ČVUT, které se zabývaly identifikací audio záznamu a na které tato práce navazuje.

Zabýval jsem se i přizpůsobením algoritmu pro aplikaci na rozpoznávání gramofonových desek, kde bylo nutné vzít v úvahu charakteristické vlastnosti gramofonových desek.

1 Identifikace audio signálu

Nezávisle na tom, zda chceme rozpoznat konkrétní skladbu, žánr této skladby nebo jestli chceme například odlišit, zda se na nahrávce nachází hudba nebo řeč, celý proces identifikace se obvykle odehrává v podobných krocích. Nejprve audiosignál projde přípravou a předzpracováním tak, abychom s ním byli schopni dále pracovat (může se jednat například o odstranění vad, převzorkování nebo zkrácení nahrávky). V dalším kroku je běžné přistoupit k segmentaci, tedy rozdělení nahrávky na kratší časové úseky. Kratší časové úseky jsou výhodné v dalším kroku, kdy přistoupíme k parametrizaci. Parametrizace spočívá ve spočítání určitých parametrů, podle nichž se bude dále řídit klasifikace. A finálním krokem je samotná klasifikace, která může stát na nějakém principu umělé inteligence, buď učení s učitelem, nebo učení bez učitele.

Tato práce čerpá z poznatků z bakalářských prací [1] a [2] vytvořených na FEL ČVUT, které se již problematikou identifikace audio signálu zabývaly. V případě práce [1] šlo o zkoumání toho, zda se na nahrávce nachází hudba, řeč, šum nebo ticho, v případě práce [2] šlo již o identifikaci konkrétní gramofonové desky. Stručné shrnutí a popis obou prací se nachází v kapitole 2.

1.1 Akustický otisk

V odborné literatuře a článcích k tématu identifikace audio nahrávek se často setkáváme s pojmem akustický otisk (acoustic fingerprint). Akustický otisk je obecně shrnutí vlastností nahrávky, kdy by měly být brány v potaz především člověkem vnímatelné vlastnosti audio nahrávky [11, 21]. Tedy například dvě nahrávky stejné skladby, každá v jiném formátu, jedna v .mp3, druhá v .wav, nebo v jakémkoliv jiném formátu, by měly mít stejné akustické otisky [11, 21]. V případě řešení zvoleném v tomto projektu můžeme za akustický otisk považovat sadu parametrů, na základě kterých nahrávku identifikujeme.

1.2 Parametrizace

Parametrizace je zásadní krok v celém procesu identifikace zvukové stopy. Parametrů existuje celá řada, může se jednat buď o jednoduché parametry obecně používané pro popis signálů (např.: root mean square energy nebo zero-crossing rate) nebo parametry sloužící k charakterizaci přímo hudební nahrávky (např.: tempo, key, mode). Pro zjednodušení úlohy je v programu využito již přichystaných funkcí k výpočtu parametrů z MIR Toolboxu [4], který nabízí sadu funkcí pro analýzu audio nahrávek.

Už ve svém předchozím projektu jsem se zabýval zkoumáním úspěšnosti parametrů při identifikaci nahrávek. Na základě dřívějších zkoumání jsem se rozhodl použít v této bakalářské práci především parametry, které se zabývají popisem nahrávky z hudební oblasti, parametry tempo a mode. V následujících odstavcích bude přiblížena podstata a princip výpočtu těchto použitých parametrů.

1.2.1 Tempo

Tempo popisuje nahrávku z hlediska rytmu a rychlosti, je běžné udávat tempo v jednotkách bpm (beats per minute), což udává počet čtvrtových not za minutu [14].

Výpočet v MIR Toolboxu [4] je založen na autokorelační funkci, kdy jsou nalezeny její vrcholy a z nich je poté odhadnuta hodnota tempa v bpm.

1.2.2 Mode

Mode (česky modus, mód) popisuje nahrávku z hlediska tónového uspořádání.

V hudební teorii existuje řada módů (například aiolský, jónský, lydický, dórský nebo mixolydický) [15], každý z nich udává konkrétní uspořádání celých tónů a půltónů v dané tónině.

Výpočet tohoto parametru v MIR Toolboxu [4] je omezen na odlišení molové a durové tóniny. Čím je hodnota parametru větší než 0, tím více durová je očekávaná tónina skladby, čím je menší než 0, tím více molová je očekávaná tónina skladby.

1.3 Klasifikace

Pro klasifikaci audio signálu existují různé přístupy. V této práci jsem zvolil metodu k-nejbližších sousedů (KNN). Další běžně používanou metodou strojového učení je metoda umělé neuronové sítě (ANN).

1.3.1 KNN

KNN [9] je algoritmus strojového učení, jedná se o metodu učení s učitelem, učení tedy probíhá z trénovacích dat. Trénovací data jsou rozdělena do konkrétních tříd, do těch jsou poté při klasifikaci zařazována i testovaná data. Každý prvek trénovací množiny je poté umístěn do některého místa v N-rozměrném prostoru. Následně přichází na řadu samotná klasifikace, kdy je testovaný objekt zařazen do té třídy, do které patří většina jeho nejbližších sousedů.

Při hledání nejbližších sousedů lze použít různé metriky, v práci [1] je použita euklidovská metrika, další obvyklou metrikou bývá Hammingova metrika [9]. Euklidovská metrika je dána vztahem:

$$d(\vec{x}, \vec{y}) = \sqrt{\sum_i (x_i - y_i)^2}$$

kde \vec{x} a \vec{y} jsou vektory o stejném počtu prvků, a prvky vektorů x_i a y_i jsou hodnoty atributů.

1.3.2 ANN

ANN je další metodou strojového učení. Tento model napodobuje svým chováním chování sítě neuronů živé biologické struktury. Přehledný popis principu umělých neuronových sítí je uveden v článku [20].

Algoritmus je založený na propojení velkého množství základních jednotek (umělých neuronů), podobně jako jsou propojeny axony (neuronové výběžky) ve skutečném biologickém mozku. Každý umělý neuron má libovolný počet vstupů, ale pouze jeden výstup. Vazby mezi umělými neurony jsou charakterizovány váhovým koeficientem w_{ji} , kterým je násobena každá vstupní proměnná x_i j-tého neuronu. Tyto násobky se sčítají a na výsledný součet je v neuronu aplikována určitá nelineární přenosová funkce $S(x)$ (tzv. aktivační funkce) [1].

Neuronová síť může mít jednu či více vrstev. U jednovrstvé sítě jsou přímo vstupy napojeny na výstupy. Běžné je použití vícevrstevných sítí s tisíci nebo milióny neuronů. Její výpočetní schopnosti se pak dají připodobit už například k mozku žížaly [17].

Přístupy k učení ANN mohou být různé pro různé varianty algoritmu. Nejpoužívanější metoda trénování algoritmu se nazývá backpropagation [20]. Při trénování tohoto algoritmu je předem známá trénovací množina poslána do neuronové sítě, ve které je stanoven určitý výsledek. Výsledek je dále porovnán s vstupní trénovací množinou, je stanovena chyba a na základě té se upravují váhové koeficienty w_{ji} tak, aby chyba byla minimální.

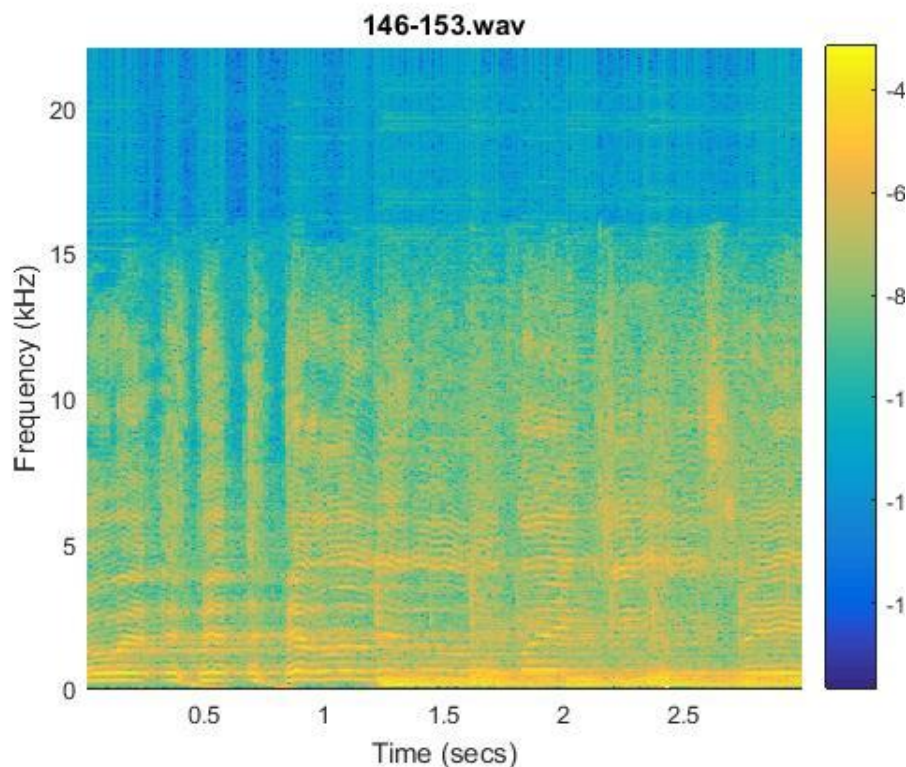
2 Dostupné technologie pro identifikaci nahrávek

Téma identifikace hudebních nahrávek má dnes značné uplatnění a proto se také mnoho komerčních společností a výzkumných projektů zabývá vývojem aplikací, které jsou schopny rozeznat zvukovou nahrávku. Pro ukázkou se v další části zmíním jednak o známých komerčních aplikacích a jednak o již vytvořených studentských pracích [1, 2], na které tato práce částečně navazuje.

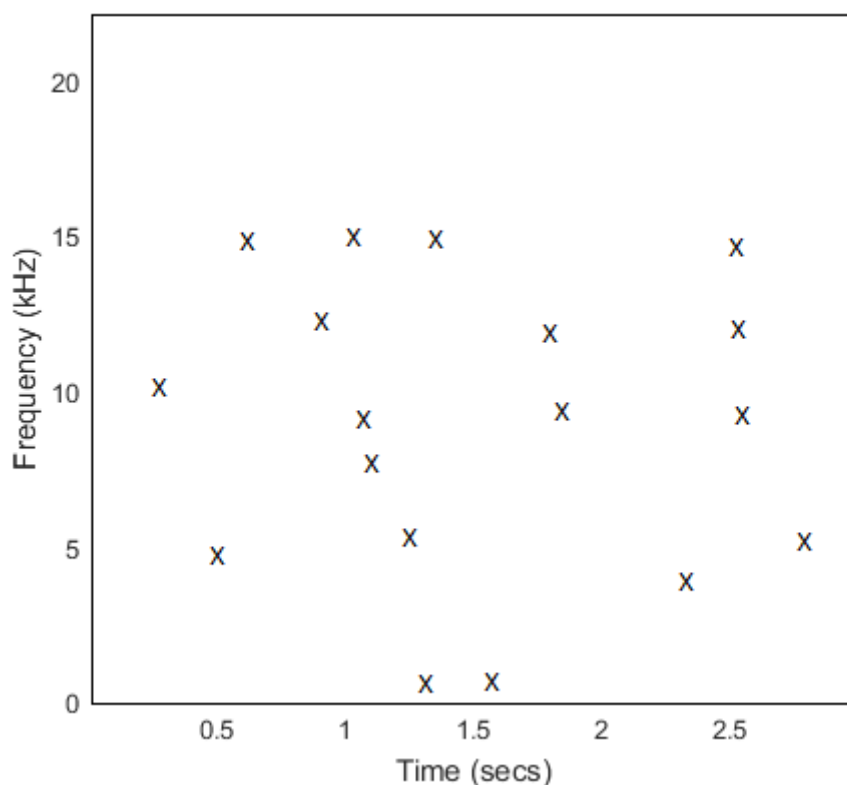
2.1 Shazam

Princip činnosti známé mobilní aplikace *Shazam* je přiblížen v článku [18] bývalého ředitele *Shazamu* Avery Li-Chun Wanga, ze kterého vychází i následující popis.

Protože aplikace byla už zpočátku mířena především na použití u mobilních telefonů, byly při vývoji aplikace *Shazam* experimentálně hledány parametry, které odolají i v přítomnosti šumu a při velkém zkreslení. Jako zásadní parametr byly tak vybrány špičky spektrogramu nahrávky. Jako špička je zvolen bod na spektrogramu (definovaný určitým časem a frekvencí), který má vyšší energii než jeho sousedé. Dojde tak k redukci původně složitého spektrogramu na mapu několika bodů. Zároveň je v tuto chvíli odstraněna informace o amplitudě špičky, výhoda tohoto přístupu je v eliminaci vlivu ekvalizace a frekvenčních úprav signálu. Výsledná „konstelační mapa“ by měla být podobná pro odpovídající si úseky v databázi a na zkoumané nahrávce. Díky šumu přítomnému na nahrávce může dojít k přidání falešných bodů, stejně tak mohou některé špičky být na nahrávce potlačeny a tím může dojít ke ztrátě některých bodů v mapě, proto se vyhledávací algoritmus snaží najít co největší počet odpovídajících si bodů v mapě.



Obr. 1: Příklad spektrogramu nahrávky



Obr. 2: Ilustrační příklad spektrogramu převedeného do konstelační mapy, vytvořený na základě popisu principu činnosti Shazamu v článku [18]

Aby nebylo nutné ukládat v databázi celé diagramy, *Shazam* vyvinul rychlý způsob indexace map. Dvojice bodů v diagramu jsou spárovány, jeden kotevní bod je vždy zvolen a k němu je vybrána cílová oblast bodů, kdy každý z nich je na tento bod napojen a spárován s ním. Je uložena frekvence každého bodu a časový rozdíl mezi nimi, zároveň je připojena informace o časovém posuvu prvního bodu od začátku úseku. Každá tato dvojice utvoří tzv. „hash“.

Tento přístup výrazně sníží objem dat nutný k uložení takových spektrogramů a umožní zvýšit rychlost vyhledávání v databázi.

Při vyhledávání jsou nalezeny shody výše popsaných sejmutých hashů. Odpovídající nahrávka je nalezena tam, kde dojde k nalezení velkého počtu po sobě následujících shod.

Kvůli zaměření na spektrogram nahrávky *Shazam*, na rozdíl od některých jiných softwarů, ale umožňuje rozpoznávat pouze originály nahrávek, neumožňuje například identifikovat píseň pouze podle zahrané nebo zazpívané melodie.

2.2 SoundHound

Další oblíbená mobilní aplikace *SoundHound* používá najednou více metod pro rozpoznání nahrávky.

Na rozdíl od *Shazamu* je pro *SoundHound* zásadní metoda *Query by Humming* [13], která umožňuje uživateli poznat nahrávku i podle zazpívané nebo jinak reprodukováné melodie [6]. Základní princip této metody je poměrně jednoduchý. Signál zachycený mikrofonem je digitalizován a jsou analyzovány jednotlivé tóny a poté jsou převedeny na zapsanou melodii. Na závěr je tento zápis porovnán s melodiemi v databázi a je nalezena nejbližší shoda nebo více nejbližších shod [13].

Následovníkem aplikace *SoundHound* je aplikace *Hound*, která je zajímavým příkladem toho, jak velký má zkoumání problematiky identifikace nahrávek potenciál. Aplikace používá podobné algoritmy, které byly původně určené k rozpoznávání nahrávek, ale v tomto případně už slouží jako virtuální asistent se schopností rozpoznávání řeči. [6]

2.3 Ekaterina Koshkina: Identifikace obsahu archivních zvukových záznamů

Bakalářská práce [1] se zabývala rozpoznáváním, zda se na zkoumané nahrávce nachází hudba, řeč, hluk nebo ticho.

V práci došlo pomocí rozsáhlého testování k výběru kombinace parametrů, která přinášela pro tuto úlohu nejlepší výsledky, vybrány byly parametry Zero-Crossing Rate, Harmonic Change Detection Function, Pulse Clarity, Roll-of a Brightness.

Parametry jsou vždy počítány pro jednotlivé segmenty, na které je zkoumaná nahrávka rozdělena, v této práci je zvolena délka segmentu 2,4 s.

Ke klasifikaci byla v této práci vybrána metoda strojového učení KNN. Klasifikátor musí být pro identifikaci nejdříve natrénovaný, tj. musí být spočítané parametry pro určitou množinu nahrávek, kde u každé je předána informace, zda se na nahrávce nachází hudba, řeč, hluk nebo ticho. Poté jsou při identifikaci spočítány parametry pro zkoumanou nahrávku a je pomocí metody KNN vyhodnoceno, do které ze zmíněných 4 tříd po umístění nahrávky do n-rozměrného prostoru (kde každý rozměr náleží jednomu z použitých parametrů) patří nejvíce jejích sousedů.

Je dosaženo výsledků v rozmezí 70-90 % pro rozpoznání hudby, řeči a ticha. Hluk se v této práci příliš úspěšně identifikovat nepodařilo, úspěšnost jeho detekce byla pouze 22 %.

2.4 Václav Král: Identifikace gramofonové desky podle krátké zvukové ukázky

Další bakalářská práce [2] se již zabývala identifikací konkrétní gramofonové desky.

Autor opět testováním vybral sadu parametrů, která se pro tuto úlohu nejlépe osvědčila, v tomto případě to byla kombinace parametrů low energy, brightness, tempo a mode.

Parametry jsou zde počítány z celého zkoumaného úseku nahrávky, není zde prováděna segmentace nahrávky.

Přístup zvolený k vyhledávání mezi nahrávkami zde autor označuje jako relační databázi. Nejprve zde dojde k vytvoření databáze, tedy opět ke spočítání parametrů a jejich uložení spolu s názvy nahrávek.

Při klasifikaci pak dojde ke spočítání rozdílů mezi parametry spočtenými pro konkrétní nahrávku a mezi parametry spočtenými pro všechny nahrávky v databázi. Na závěr dojde ke spočítání průměru rozdílů hodnot parametrů pro každou nahrávku z databáze, a je vypsáno 5 nejpravděpodobnějších nahrávek.

Pro zpřesnění identifikace je v práci představen jednoduchý způsob pro detekci začátku nahrávky, kdy je začátek detekován tam, kde dojde k překročení prahu 0,25 maximální amplitudy přítomné na nahrávce.

V této práci jsou při testování úspěšnosti algoritmu k původním nahrávkám přidány některé typické atributy gramofonových desek, konkrétně zjednodušená simulace šumu pomocí bílého šumu a zjednodušená simulace přeslechů. Úspěšnost algoritmu je různá pro různé parametry znehodnocení, pohybuje se v rozmezí 82 % - 96 %.

3 Gramofonové desky

Protože cílem této práce je úspěšně identifikovat nahrávku z gramofonové desky, je nutné se kromě vlastností audio signálu zabývat i vlastnostmi gramofonových desek, a to především různými možnými nedostatky vinylové desky, které mohou ovlivnit kvalitu testovaného zvuku.

3.1 Snížení frekvenčního rozsahu

Pro gramofonovou desku je typické, že s rostoucím počtem přehrání dochází k jejímu opotřebení a postupnému snižování frekvenčního rozsahu gramofonové desky. Tento úbytek je znát především na vysokých frekvencích.

Kromě opotřebení je frekvenční charakteristika mechanického záznamu sama o sobě limitována parametry záznamového i čtecího zařízení [5, 10]. Horní mez přenášeného kmitočtu je dána rozměrem snímacího hrotu [10]. Ztráty vzniklé daným poloměrem snímacího hrotu a záznamového nože jsou závislé na poloze na desce [5], větší ztráta bude přítomná blíže středu desky

3.2 Přeslechy

Další příčina znehodnocení desky může být vlivem přeslechů. Na desce je stereofonní zvuk uchovávan v jedné drážce, každá stěna drážky obsahuje jeden kanál. Řezání do desky je prováděno pomocí jednoho záznamového nože, na který jsou připojeny dvě cívky, z nichž každá způsobuje řezání do jedné strany drážky. Při tomto výrobním procesu může snadno dojít k přeslechům mezi těmito kanály, pokud na sebe nejsou stěny drážek kolmé, nebo pokud na sebe obě cívky nejsou kolmé. [2]

Nejlepší hodnota přeslechů se udává na 1 kHz, podle normy ČSN IEC 98 by zde hodnota měla být lepší než 20 dB [7], dále by podle této normy na kmitočtu 6,3 kHz měla být hodnota lepší než 15 dB.

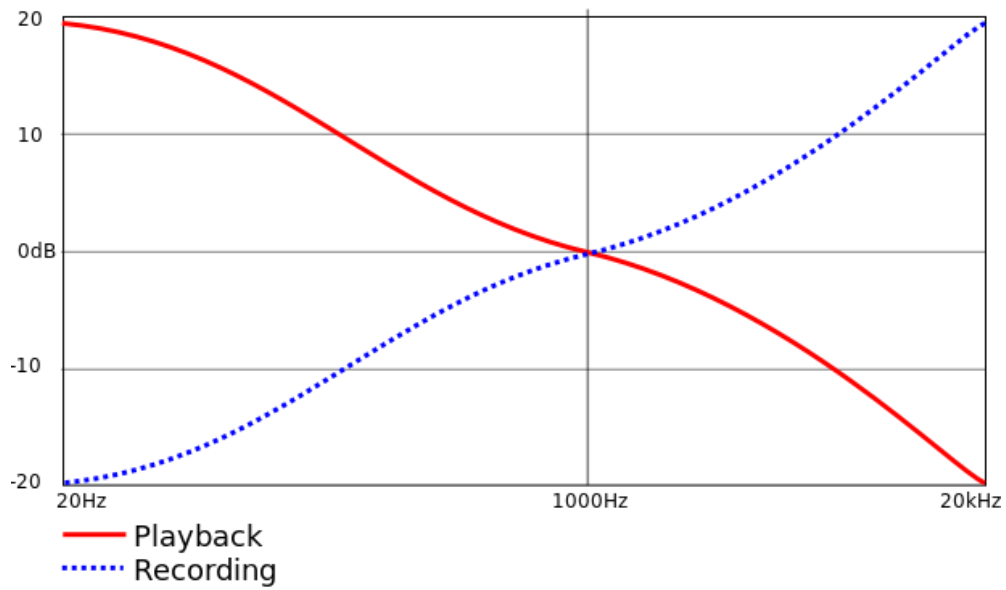
3.3 Šum

Dalším atributem může být slyšitelný šum na desce. Ten může mít mnoho příčin.

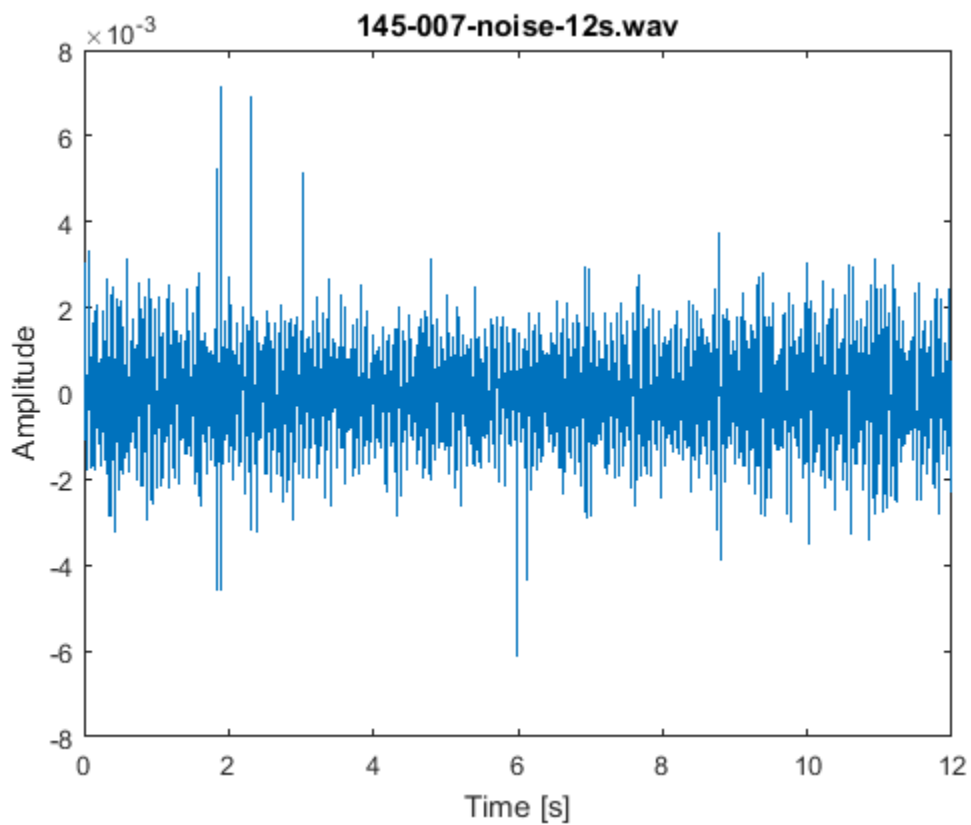
Může vzniknout přímo ve studiu už při nahrávání. V tomto případě se pak šum nachází už na masteru desky a tím pádem nám při rozpoznávání nahrávky nevadí. Podstatnější pro naši úlohu bude šum, který se na desce objeví během její výroby. Vzhledem k tomu, že není technicky možné vyrobit drážky naprosto hladké a materiál je vždy do jisté míry zrnitý, nerovnosti se projeví ve zvuku ve formě šumu [10]. Zároveň dojde k přidání šumu samotným přehrávačem, jednak při snímání jehlou a jednak díky elektrickým obvodům v přehrávači. Moderní přehrávače dosahují hodnot SNR=50 dB. [22]

Pro potlačení šumu se při masteringu desky přistupuje k předekvalizaci záznamu [5]. Protože je šum přítomný především na vysokých kmitočtech, jsou vysoké frekvence před výrobním procesem, ve kterém je očekávatelné přidání šumu, zesíleny. Opačná ekvalizace je poté použita při přehrávání desky, výsledek je tedy nahrávka se stejnou frekvenční charakteristikou, jaká byla u původní nahrávky, a zároveň je tak potlačen šum, který při výrobním procesu vznikl na vysokých kmitočtech.

Tvar křivky předekvalizace byl v roce 1954 standardizován Asociací amerického nahrávacího průmyslu (Recording Industry Association of America, RIAA) [5,19].



Obr. 3: Tvar křivky ekvalizace při záznamu a při přehrávání gramofonové desky, převzato z [19]



Obr. 4: Příklad šumu nalezeného na začátku gramofonové desky

3.4 Zkreslení

Zkreslení v akustice je obecně jev, při kterém dochází k deformaci křivky signálu. Například u různých typů zesilovačů bychom se s ním setkali při přebuzení zesilovače, kdy už není možné ve stejné míře zesílit maxima křivky jako její minima.

V literatuře [10] je popsáno několik typů zkreslení, které mohou nastat u gramofonové desky.

Zásadní je nelineární dotykové zkreslení, které nastává především na vysokých kmitočtech, kdy je poloměr zaoblění vrcholů křivky srovnatelný s poloměrem dotykové kružnice snímacího hrotu, a v důsledku toho není hrot schopen přesně opsat křivku a dojde k její deformaci.

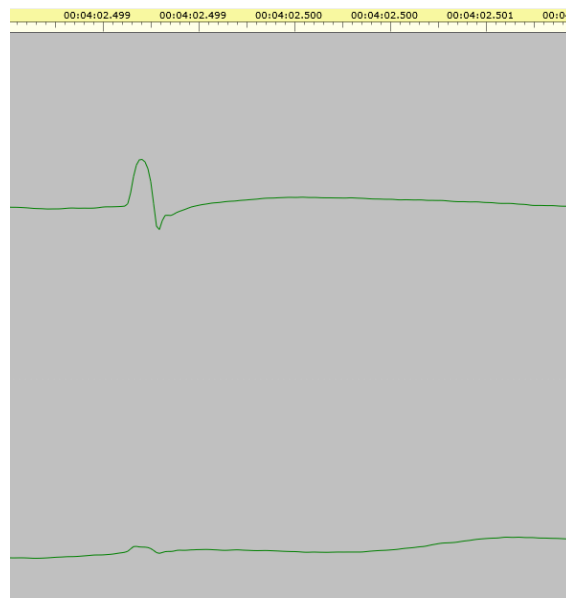
Zkreslení, které tak vzniká, je tedy značně nelineární, závislé na amplitudě a frekvenci signálu. Pro frekvence nižší než 2 kHz jsou hodnoty zanedbatelné, při frekvenci 10 kHz může dosahovat i přes 10 %.

Další typy zkreslení mohou nastat například při vychýlení hrotu jehly z drážky nebo při velkých výchylkách jehly vlivem setrvačné síly.

3.5 Impulsní vady

Velmi typickým atributem gramofonových desek jsou impulsní vady. Impulsní vady vznikají především při výrobě gramofonové desky a projevují se často velmi zřetelným lupáním nebo praskáním. Na rozdíl od ostatních zmíněných atributů impulsní vady neovlivní signál v celé délce, ale pouze v některých jeho částech [3].

Jak může vypadat průběh signálu s impulsní vadou je znázorněno na obrázku.



Obr. 5: Ukázka průběhu signálu obsahujícího impulsní vadu (zde vzniklou v důsledku poškrábané desky) [16]

4 Implementace algoritmu v prostředí MATLAB

V této části se zabývám implementací vlastního programu v prostředí MATLAB, jehož cílem je rozpoznat nahrávku a přiřadit k ní správnou nahrávku z databáze.

Práce se opírá o poznatky a funkce vytvořené již v pracích [1, 2].

Základ pro implementaci funkcí pro klasifikaci pomocí metody KNN je převzat z bakalářské práce [1], proces vyhledávání a identifikace je ale kvůli rozdílnosti obou úloh značně upraven (vizte kapitola 4.2). Z práce [2] jsem se rozhodl použít především funkci pro detekci začátku nahrávky.

4.1 Testovací množina nahrávek

Pro účely testování úspěšnosti algoritmu bylo nutné mít k dispozici dostatečnou databázi nahrávek. Pro tento účel jsem měl k dispozici 264 masterů gramofonových desek (pozn.: pojem master zde znamená digitální master, „originál“ nahrávky uchovaný v digitální podobě, po všech fázích postprodukce, kterými nahrávka prošla ve studiu), poskytnutých firmou GZ Media, a 5 dalších nahrávek, u kterých jsem měl k dispozici jak master desky tak nahrávku pořízenou přímo z gramofonové desky. Nahrávka byla vždy zkrácena v programu na 20 s od detekovaného začátku.

Délka 20 s by jistě pro posluchače byla dostatečná k rozpoznání nahrávky. Zároveň nebrání ani praktickému využití programu, v praxi by nebyl pro uživatele problém přehrát k identifikování právě začátek desky.

4.2 Klasifikace

Celý proces klasifikace je řízen ve funkci *CLASSIFICATION.m*. V té dojde nejdříve k předzpracování nahrávky (kapitola 4.4), poté jsou pro nahrávky spočítány parametry pomocí funkce *computeParameter.m* (kapitola 4.3). Poté dojde k samotné klasifikaci pomocí metody KNN a stanovení výsledku pomocí hierarchie parametrů (kapitola 4.3.2). Na závěr dojde k výpisu výsledků ve funkci *printResults.m* a uložení výsledků do excelu pomocí funkce *resultsToXls.m*.

4.2.1 Klasifikace pomocí metody KNN

Z metod strojového učení jsem se rozhodl v této práci použít metodu KNN. V programu je implementována ve funkci *KNN.m*. Jako základ pro tuto funkci jsem využil již připravenou funkci z bakalářské práce [1].

Na rozdíl od úlohy, ke které byla funkce určena v [1], tedy rozdělení nahrávek do 4 tříd (hudba, řeč, ticho, šum), je úloha identifikace konkrétní nahrávky specifická tím, že každá nahrávka je sama o sobě třídou, máme tedy v našem případě 269 tříd, a pro natrénování klasifikátoru máme od každé třídy k dispozici pouze jednu nahrávku.

V rámci svých předchozích projektů jsem tedy přistoupil k použití metody KNN podobným způsobem jako v [1], při vyhodnocování nahrávky se zkoumaný segment vyhodnocoval na základě dat získaných při trénování všech segmentů (tedy například při identifikaci prvního segmentu nahrávky byla zohledněna data získaná i z jiných než prvních segmentů nahrávek v trénovací množině).

V této práci jsem přístup změnil. Rozhodl jsem se porovnávat vždy pouze segmenty se stejným pořadím na nahrávce (tedy například první s prvním, druhý s druhým atd.). Tento přístup je v naší úloze výhodnější, protože po implementaci funkce pro zkrácení nahrávky by mělo docházet k porovnávání prakticky stejných úseků nahrávky. Zároveň tak snížíme výpočetní náročnost algoritmu.

Pro každý segment je spočítána vzdálenost k segmentům z trénovací množiny a následně dojde ke stanovení k-nejbližších sousedů. V práci [1] je vzdálenost spočítána pomocí euklidovské metriky, u nás tato metrika kvůli vyhodnocování pouze na základě jednoho parametru najednou (vizte kapitola 4.3) přechází v obyčejný rozdíl dvou hodnot parametru. Na závěr je vyhodnoceno, která nahrávka se umístila mezi k-nejbližšími sousedy v nejvíce segmentech, a ta je stanovena jako nejpravděpodobnější.

4.2.2 Trénování klasifikátoru

Pro natrénování klasifikátoru je v programu implementována funkce *TRAIN.m*. Trénování spočívá ve spočítání parametrů pro nahrávky z vybrané složky a uložení hodnot parametrů spolu s názvy nahrávek. Každá nahrávka ještě předtím projde kroky zmíněnými v kapitole 4.4 (převzorkování, převedení do zvolené reprezentace signálu a zkrácení).

Jako trénovací množina je použito všech 269 dostupných masterů nahrávek.

4.3 Parametrizace

Pro spočítání parametrů, na základě kterých bude algoritmus moci dále nahrávky identifikovat, jsem využil parametrů z MIR Toolboxu.

Pro spočítání konkrétního parametru pro všechny segmenty nahrávky je v programu implementována funkce *computeParameter.m*, jejímž prvním vstupním argumentem je matice jednotlivých segmentů nahrávky a druhým řetězec udávající parametr, který chceme spočítat.

4.3.1 Výběr parametrů

Z velkého množství parametrů, které MIR Toolbox nabízí, bylo nutné testováním najít vhodnou kombinaci těchto parametrů, která se hodí k identifikaci gramofonové desky.

Ve svých předchozích projektech na toto téma jsem nejprve testoval pouze úspěšnost identifikování samotných masterů, na které byl algoritmus natrénován. Při testování jsem nejprve došel k nejlepšímu výsledku pro kombinaci parametrů RMS, flux, rolloff a brightness. Následně jsem ale testoval úspěšnost pro nahrávky s mírně upravenými spektrálními vlastnostmi (na nahrávky byla aplikována dolní propust, zkreslení pomocí funkce arcus tangens a bílý šum), čímž jsem zjednodušeně simuloval vlastnosti gramofonových desek. Výsledek byl takový, že i pro nahrávky s takovou úpravou, která byla poslechem prakticky nerozpoznatelná, se výše zmíněné parametry přestaly osvědčovat. Výsledek nebyl nijak zarážející, uvědomíme-li si, že byly převážně založené na spektrálních vlastnostech nahrávky, například brightness i přímo na energii na vysokých kmitočtech, a výše uvedenými simulacemi došlo k úpravě především vysokých kmitočtů. Po dalším testování a vyzkoušení více parametrů se nejvíce osvědčil parametr tempo, relativně dobré úspěšnosti dosáhl i parametr mode.

Odolnost vůči šumu a frekvenční úpravě signálu je u tempa očekávatelná, rytmus v nahrávce je i posluchač schopen vnímat a identifikovat i při velkém znehodnocení nahrávky. Výpočet tempa v MIR Toolboxu s pomocí autokorelační funkce dokáže podobně jako posluchač snadno identifikovat periodicity a špičky v signálu i při přítomnosti velkých frekvenčních a jiných úprav nahrávky.

Jelikož se dá očekávat relativně spolehlivá identifikace nahrávky pomocí tempa, rozhodl jsem se identifikaci pomocí tempa nekombinovat s žádným dalším parametrem. Při samotném použití parametru tempo ale hrozí, že nastane i mnoho případů, kde bude vyhodnoceno podobné tempo pro větší množství nahrávek. Proto jsem se rozhodl pro nasazení hierarchie parametrů, kdy v nejednoznačných případech dojde ke spočítání a vyhodnocení na základě dalšího parametru. Jako další parametr jsem vybral zmíněný parametr mode, který se také dá pokládat za odolný vůči znehodnocení, jelikož úpravy spektra nahrávky ani přidaný šum přímo nepřidají ani neodeberou žádné tóny z nahrávky, na základě kterých je spočítána tonalita nahrávky.

4.3.2 Implementace hierarchie parametrů

Nejprve dojde k vyhodnocení pomocí metody KNN na základě prvního parametru a poté, pokud je to třeba, je spočítán druhý parametr a jeho výsledky jsou také vyhodnoceny pomocí metody KNN. Následně dojde ve funkci *getResults.m* k porovnání obou výsledků a určení konečného výsledku. Ten je určen tak, že z matice výsledků získaných podle druhého parametru jsou vybrány výsledky, které jsou na prvním místě podle prvního parametru. Z takto získané matice výsledků je na závěr vybrána jedna nebo více nahrávek, které měly podle druhého parametru největší pravděpodobnost.

4.4 Předzpracování nahrávky

4.4.1 Převzorkování

Při identifikaci nahrávek je výhodné porovnávat nahrávky se stejným vzorkovacím kmitočtem. Za tímto účelem je do programu zařazena funkce *fsChange.m*, která se zakládá na návrhu funkce z bakalářské práce [1] a která umožní převzorkování nahrávek při trénování a při klasifikování na referenční kmitočet. Jako referenční kmitočet jsem zvolil 48 kHz, který se běžně používá u audio nahrávek.

4.4.2 Segmentace

Pro účely parametrizace je vhodné nahrávky rozdělit na kratší časové úseky, u kterých budeme moci dále počítat parametry. Za tímto účelem je v programu zařazena funkce *segment_signal.m*, která je převzata z bakalářské práce [1].

Rozhodl jsem se použít segmenty poměrně dlouhé, o délce 4,5 s a překryvem 1,5 s.

Důvodem je jednak charakter parametrů, kdy pro oba použité parametry tempo a mode je třeba delší úsek. Tempo je počítáno na základě periodicit a poloh špiček signálu, na krátké nahrávce by bylo obtížné tuto pravidelnost najít. Parametr mode také potřebuje k určení vztahů mezi tóny spíše delší úsek, na krátkém by se nemusel vyskytovat dostatečný počet tónů pro získání hodnověrných výsledků.

Zároveň delší úseky pomohou částečně zanedbat případné malé posunutí mezi testovanou nahrávkou a nahrávkou z trénovací množiny.

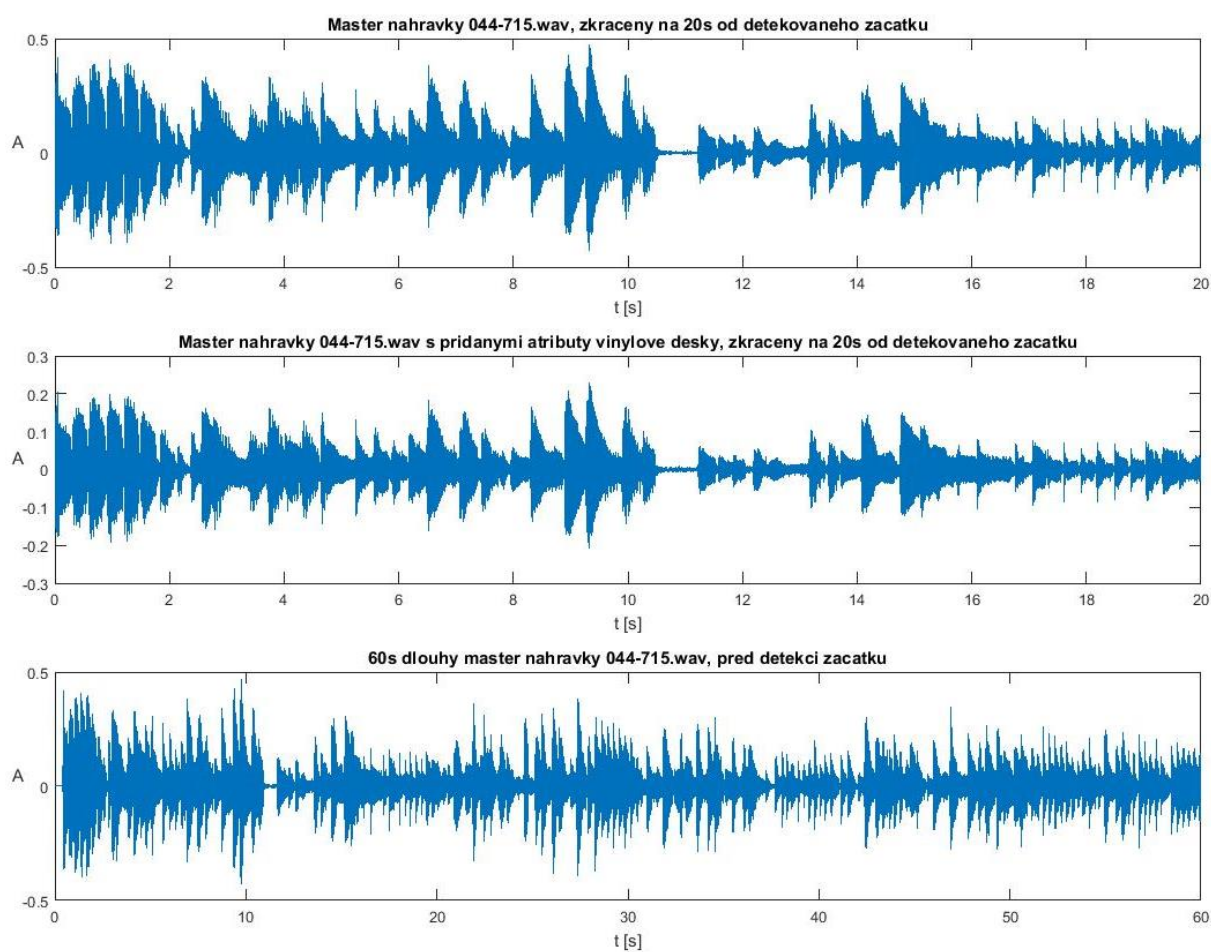
4.4.3 Zkrácení nahrávky

Pokud chceme algoritmus využít k identifikaci reálné gramofonové desky, jedním z problémů, se kterým je nutné se vypořádat, je vzájemné posunutí testované nahrávky pořízené z gramofonové desky a masteru této nahrávky.

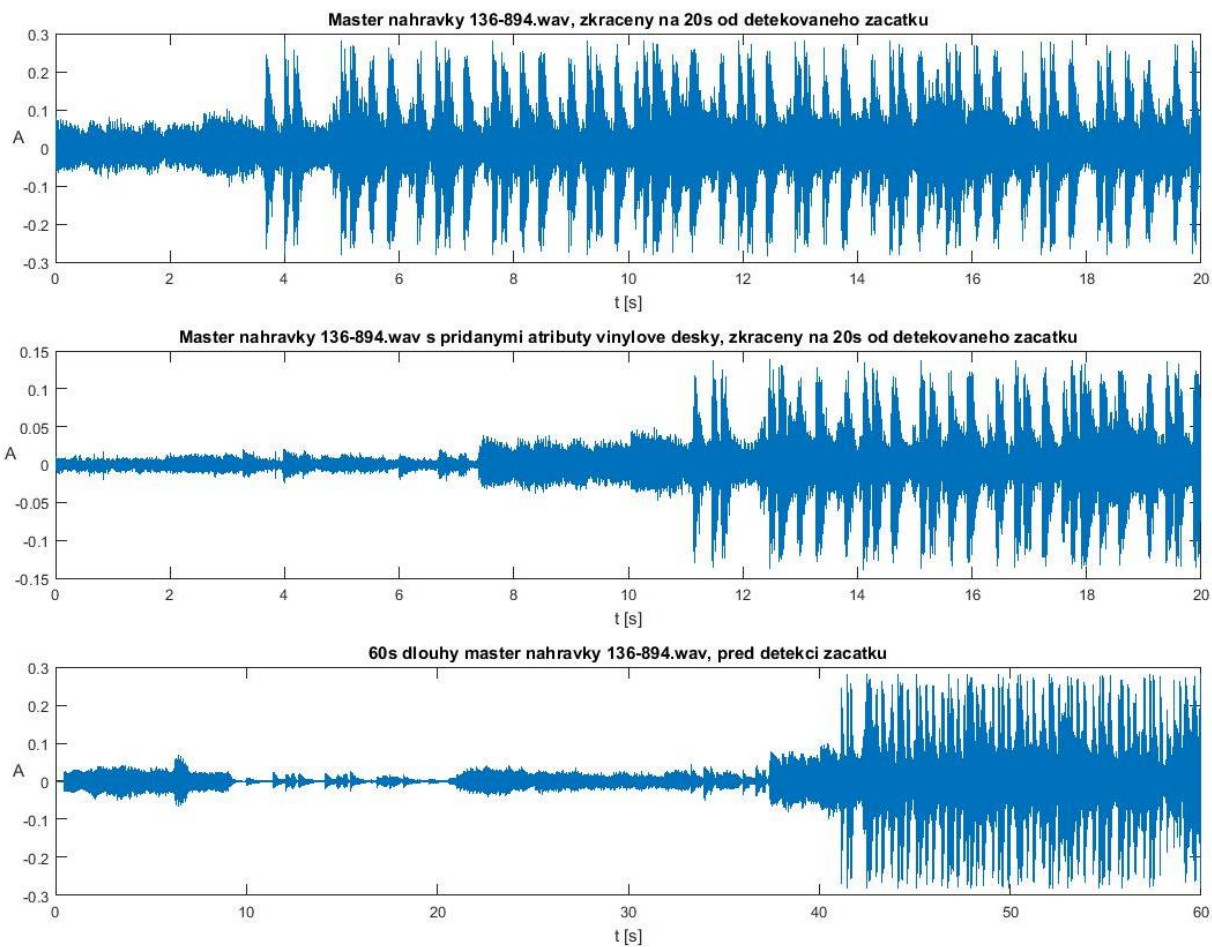
Bylo tedy nutné do programu zařadit funkci pro nalezení začátku a zkrácení nejen testovaných nahrávek, ale i masterů z trénovací množiny. Implementovaná funkce *findStartCut.m* využívá návrhu, který je popsán v bakalářské práci [2]. Nahrávka je nejprve zkrácena na 40 s od začátku, a začátek skladby je poté detekován tam, kde hodnota nahrávky přesáhne daný práh. V bakalářské práci [2] je použit práh 0,25 hodnoty maximální amplitudy přítomné na předzkrácené 40 s nahrávce.

Při vyzkoušení této hodnoty prahu jsem narazil na problém, kdy v ojedinělých případech pro větší hodnoty vzájemného posunu nahrávky z testovací a trénovací množiny byla nalezena výrazně jiná hodnota maximální amplitudy, a tím pádem i došlo k detekci začátku na jiném místě. Rozhodl jsem se tedy hodnotu prahu snížit na 0,15 maximální nalezené amplitudy, která by měla přinést lepší detekci začátku i v případě většího rozdílu mezi maximální amplitudou a začátkem hudby. Zároveň je to hodnota, která by měla být ještě stále výrazně větší než hodnota šumu na nahrávce.

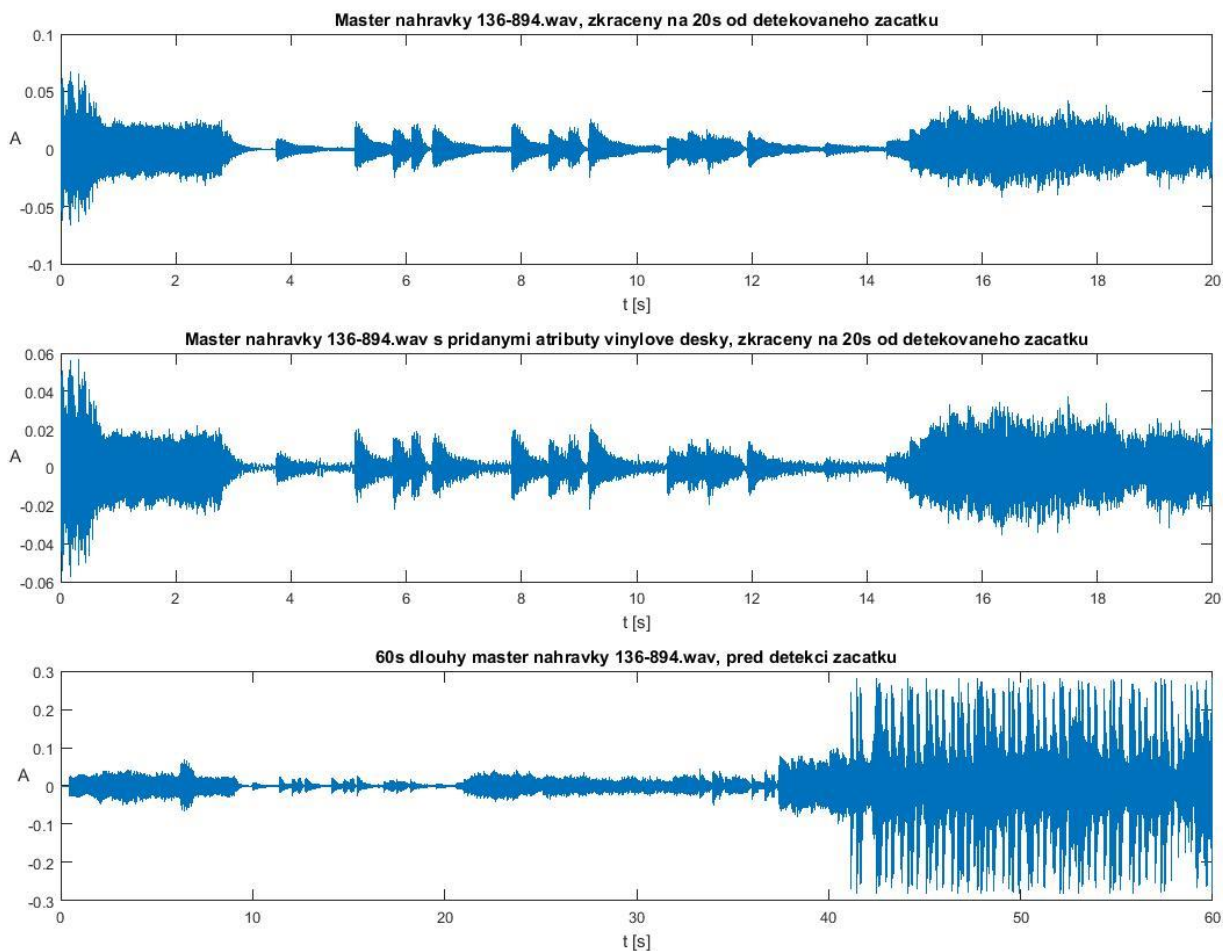
Nicméně ani po využití této funkcionality nemůžeme očekávat naprosto přesnou shodu úseků, ale pro identifikace je důležité, aby posunutí bylo výrazně menší než je délka segmentů, čehož se touto funkcí téměř vždy podařilo dosáhnout.



Obr. 6: Příklad správného zkrácení masteru i posunutě simulace vinylové nahrávky na stejný úsek, případ většiny nahrávek.



Obr. 7: Ojedinělý příklad zkrácení nahrávky na jiný úsek pro posunutou nahrávku se simulací vlastností vinylové desky a pro originální master, zde pro použitý práh 0,25 maximální amplitudy, z průběhu originální nahrávky je zřejmé, že se jedná z hlediska nalezení začátku o problematickou nahrávku.



Obr. 8: Příklad zkrácení nahrávky na stejný úsek pro posunutou nahrávku se simulací vinylových vlastností i pro originální master, zde pro použitý práh 0,15 maximální amplitudy.

4.4.4 Převedení nahrávky do libovolné reprezentace signálu (MS, stereo, mono)

Zvukovou stopu, jak na zkoumané gramofonové desce, tak na digitálních masterech, lze očekávat stereofonní.

Pro účely rozpoznání desky se nabízí možnost využít i jiných možných reprezentací signálu.

Jednou možností je využít nahrávku pouze monofonní a sečíst oba kanály a poté pouze normovat výsledek.

Další možností je využít tvar mid-side (MS) [8], kdy se v kanálu mid nachází součet L+R a v kanálu side rozdíl L-R. To by nám případně mohlo umožnit i kanály váhovat, což by sice bylo technicky možné i u pravého a levého kanálu, ale nemělo by to prakticky žádný význam.

Pro tento účel je v programu napsaná funkce *soundRep.m*. Jejím vstupem je matice reprezentující stereo zvuk, a zároveň řetězec, který předává informaci o tom, do jakého tvaru chceme zvuk převádět. Výstupem funkce je zvuk v libovolném tvaru, buď stereo (v tom případě je pouze vstupní

signál rovný výstupu), mono (kdy dojde k sečtení levé a pravé složky a normování výsledku) nebo ve tvaru MS.

Z důvodu testování identifikace, jak při využití jednoho tak i dvou kanálů, jsou i ostatní funkce v programu napsány tak, aby byly schopny pracovat s oběma možnostmi.

4.5 Simulace vlastností vinylových desek

Pro účely testování se ukázalo jako praktické mít k dispozici co největší počet dostupných nahrávek pořízených z reálných gramofonových desek. Protože jsem měl k dispozici jen 5 takových nahrávek, rozhodl jsem se pokusit se co nejvěrněji simulovat vlastnosti gramofonové desky a přidat je do masterů nahrávek.

Při porovnání masterů a nahrávek z vinylů bylo patrné několik zásadních rozdílů. V první řadě u nahrávek z vinylů byla obvykle nižší amplituda než u masteru, o několik vteřin byl posunutý začátek nahrávky a zároveň byl na vinylové nahrávce přítomný šum.

4.5.1 Přidání šumu, snížení amplitudy

Pro přidání šumu do nahrávek a zároveň snížení amplitudy jsem vytvořil funkci *addNoise.m*.

Při úvaze, jak nejlépe docílit simulování šumu desky, jsem se rozhodl využít šum nahraný z vinylových desek, který byl patrný na začátku desky. Z 5 vinylů jsem vybral 3, na kterých se na začátku nacházel dostatečně dlouhý úsek obsahující pouze šum (u jedné desky 3 s, u další 9 s a u poslední 12 s). Ve funkci *addNoise.m* dojde k náhodnému vybrání jednoho z těchto šumů, ve for cyklu dojde k nabalení potřebného počtu úryvků toho šumu za sebou a přičtení k masteru desky. Zároveň funkce sníží amplitudu původního masteru na hodnotu, která byla přítomna na vinylové nahrávce, ze které je pořízený šum. Tím docílíme přibližného zachování poměru signálu a šumu SNR.

Ač by bylo možné samostatně simulovat impulsní vady, nepovažuji to za nutné, neboť šum převzatý z gramofonových desek v sobě již několik impulsních vad obsahoval.

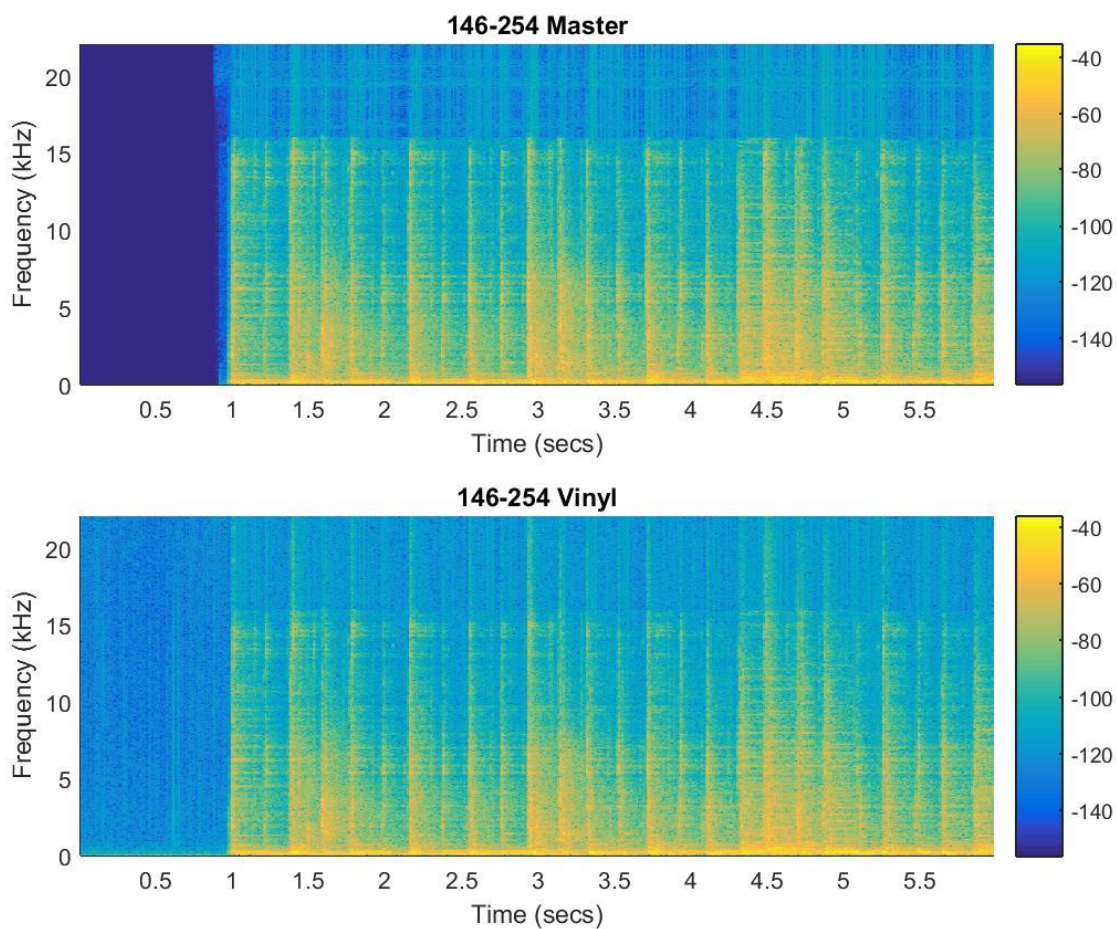
4.5.2 Posunutí začátku

Pro posunutí začátku je implementována funkce *shiftStart.m*, která umožní posunutí začátku (tedy přidání ticha před začátek samotné nahrávky) o zvolený čas a zároveň zkrácení výsledné nahrávky na požadovanou délku. Protože i na nahrávkách pořízených z vinylů byla doba posunutí až 10 s, rozhodl jsem se začátek všech nahrávek také posunout o 10 s.

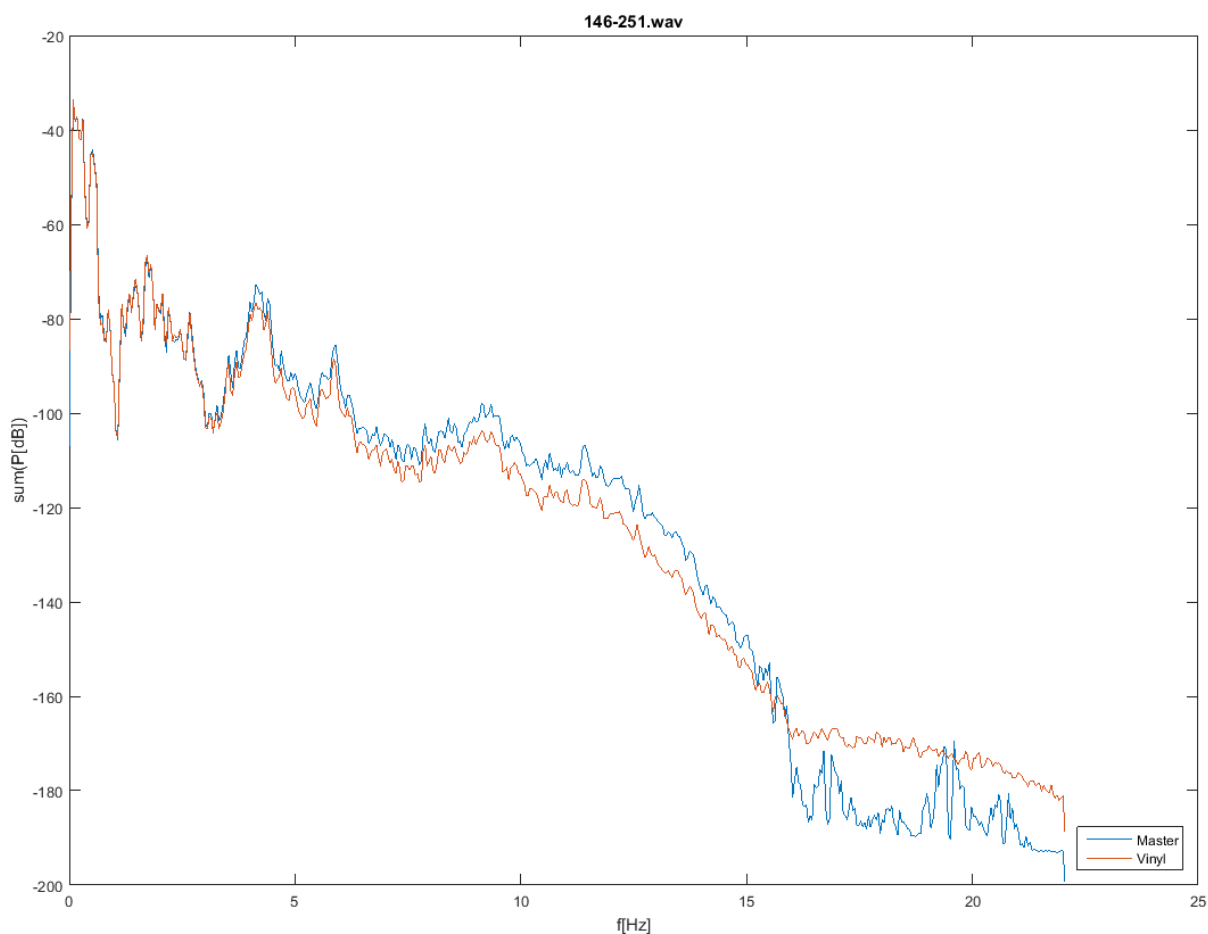
4.5.3 Úprava frekvenčního rozsahu

Další vlastností vinylových desek, kterou je vhodné simulovat, je zúžení frekvenčního pásma a ztráta především vysokých frekvencí.

Abych se přesvědčil, jak velká ztráta kmitočtů je přítomna u nahrávek z dostupných gramofonových desek, v Matlabu jsem vytvořil a poté porovnal spektrogramy nahrávek. Pro lepší možnost porovnání jsem poté provedl součet hodnot výkonů pro jednotlivé frekvence přes celou délku zkoumaného úseku nahrávky a výsledek vynesl do grafu.



Obr. 9: Srovnání spektrogramu masteru nahrávky a nahrávky z gramofonové desky (pozn.: pro názornost není zahrnuto vzájemné posunutí nahrávek a nahrávky jsou zkráceny na 6 s včetně 1 s obsahující ticho na začátku nahrávky)



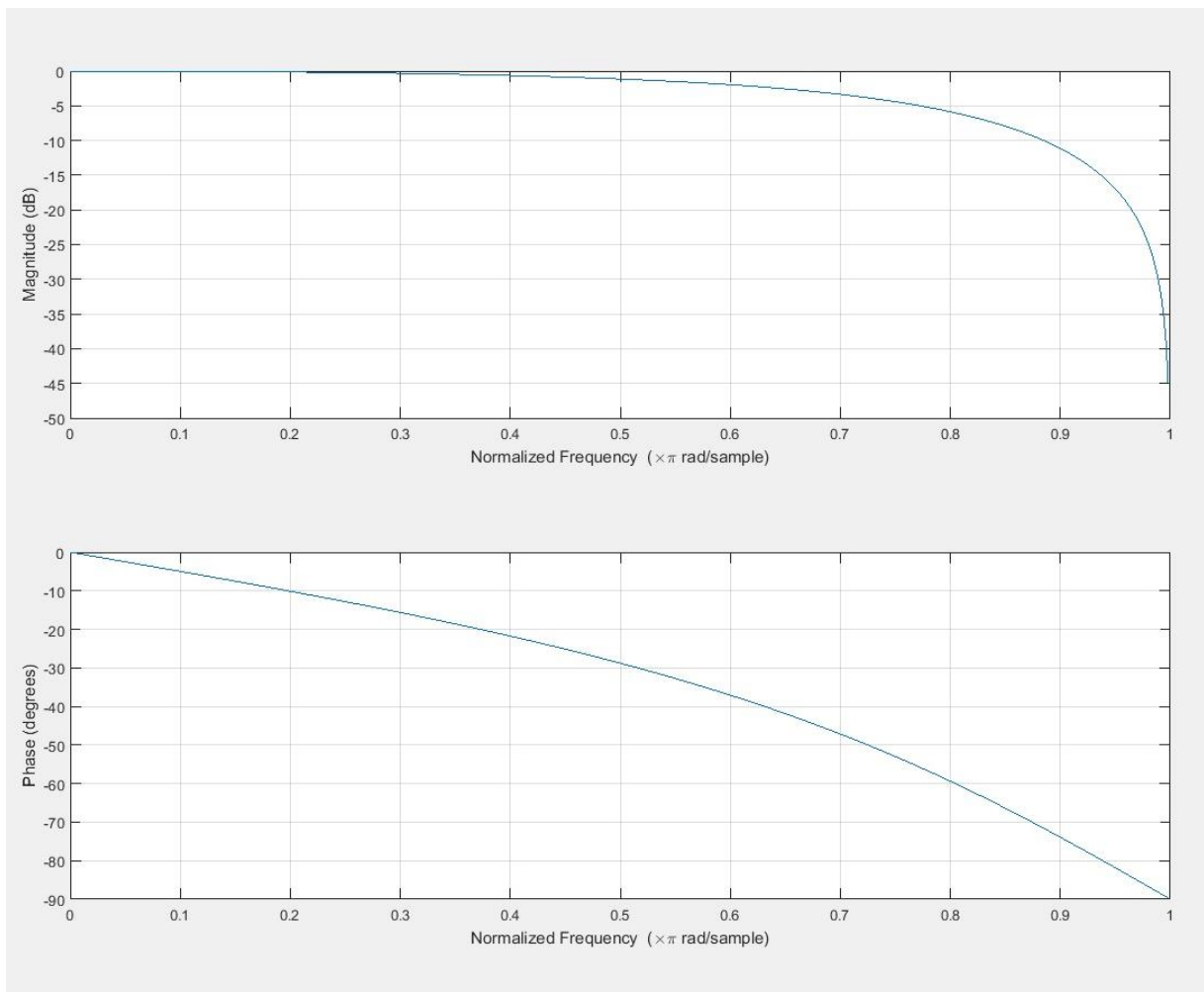
Obr. 10: Srovnání frekvenčních charakteristik masteru a vinylové desky, grafy vznikly součtem výkonů nahrávky v celém zkoumaném úseku

Z grafů lze pozorovat, že s rostoucí frekvencí se rozdíl mezi výkonem masteru a vinylu zvětšuje, to je v souladu s očekávatelným úbytkem výšek. Nad frekvencí 15 kHz je pozorovatelné, že na vinylové desce je původní spektrum z masteru nahrávky zamaskované šumem.

Dá se ale říci, že omezení kmitočtového pásma zde není nijak výrazné. To je i v souladu s tím, že u nahrávek nebyl ani při poslechu znát žádný výrazný úbytek vysokých frekvencí. Pravděpodobně se také jednalo o nové nahrávky, u kterých zatím nastal jen malý počet přehrání. Navíc se také vždy jedná o začátek desky, kde se dají očekávat lepší parametry a menší ztráta kmitočtů.

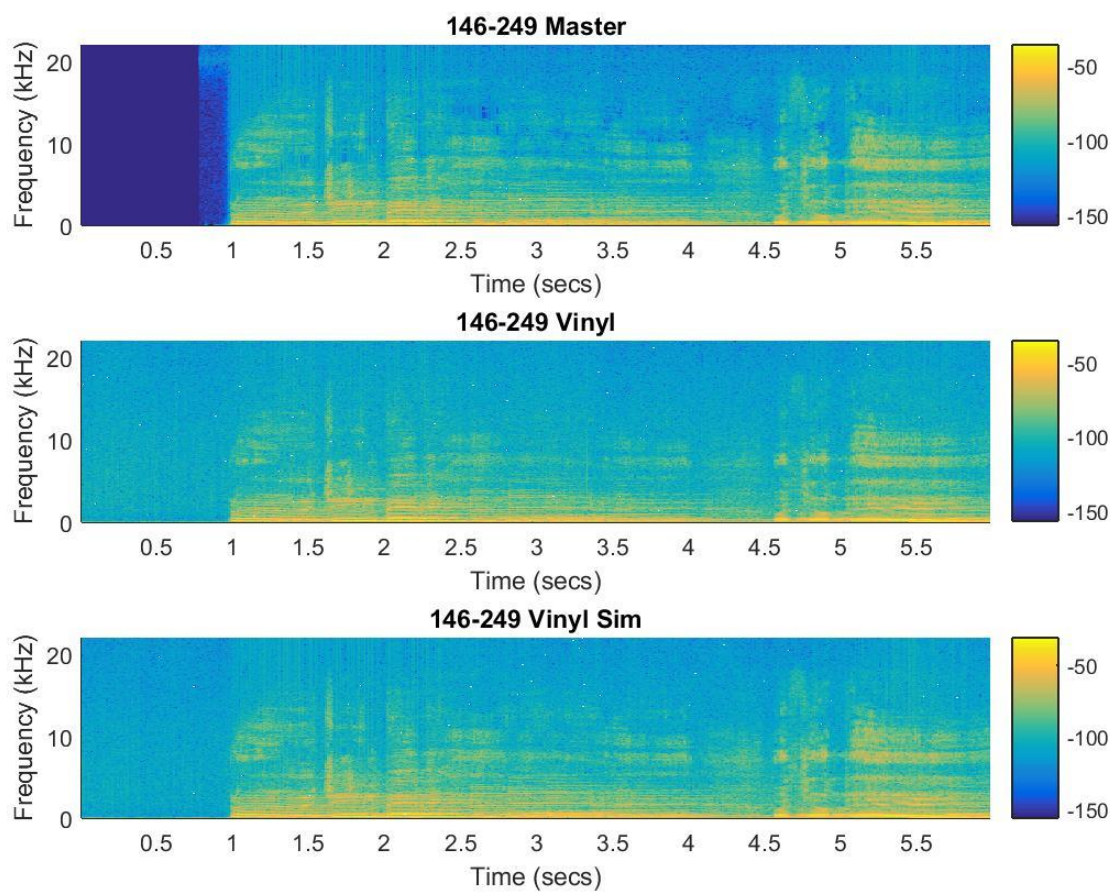
I přesto je pro simulaci nahrávky z vinylové desky pro tento účel vhodné přistoupit k úpravě spektra, a zajistit tak, že klasifikátor se bude muset potýkat s nahrávkami, jejichž spektrum se zcela neshoduje se spektrem nahrávky z trénovací množiny, což je u porovnávání masteru a vinylu desky očekávatelné.

Pro tento účel jsem implementoval jednoduchou funkci *lowPass.m*, ve které byla realizována dolní propust pomocí Butterworthova filtru. Funkce umožňuje pomocí parametru *cutOff* nastavit hranici, od které bude docházet k útlumu vysokých frekvencí. Tuto hranici jsem zvolil jako 15 kHz.



Obr. 11: Frekvenční odezva použitého filtru

Po implementování zmíněných znehodnocení nahrávky jsem opět pomocí spektrogramu porovnal spektra masteru, vinylové nahrávky a simulované vinylové nahrávky. Výsledek se dal označit jako pro tento účel dostatečný, spektrogram masteru s přidáním atributy gramofonové desky skutečně připomíná spektrogram reálné vinylové nahrávky, stejně tak při poslechu simulovaná nahrávka připomíná nahrávku nahranou z gramofonové desky.



Obr. 12: Srovnání spektrogramu masteru nahrávky, nahrávky z gramofonové desky a masteru s přidánými vlastnostmi gramofonové desky (pozn.: pro názornost není zahrnuto vzájemné posunutí nahrávek a nahrávky jsou zkráceny na 6 s včetně 1 s obsahující ticho na začátku nahrávky)

4.5.4 Další atributy gramofonové desky

Dalšími atributy popsanými v teoretické části jsou zkreslení a přeslechy mezi kanály. Tyto atributy jsem rozhodl nesimulovat.

U zkreslení je důvodem to, že zkreslení u gramofonové desky je nelineární a bylo by obtížné najít tvar, který by se blížil reálným vlastnostem desky. Ve svém důsledku by zkreslení zapříčinilo přidání vyšších harmonických složek, které původně na nahrávce přítomné nebyly. Jelikož nahrávka již prošla především na vysokých kmitočtech značnou úpravou díky přidání šumu a díky aplikované dolní propusti, nepovažuji přidání dalšího atributu, jehož simulace by stejně musela být značně zjednodušená a pravděpodobně by tak nahrávku nepřiblížila nahrávce z reálné desky, za nutné.

Zjednodušená implementace přeslechů by byla snadno realizovatelná při zanedbání jejich frekvenční závislosti a závislosti na poloze na desce, v práci [2] již byla taková simulace implementována. Jelikož jsem ale v testech nejprve pro jednoduchost pracoval pouze s mono složkou signálu, neměla by simulace přeslechů smysl. V práci [2] již bylo zjištěno, že vliv přeslechů na identifikaci je minimální a ke stejnému závěru jsem došel ve svém předchozím projektu na toto téma.

4.6 Testování algoritmu

Pro testování úspěšnosti identifikace, nebo případně i pro identifikování většího množství nahrávek najednou, umožňuje program uživateli vybrat složku se soubory určenými k identifikování, provést identifikaci všech přítomných nahrávek a uložit získané výsledky do excelu. Zároveň dojde v excelu automaticky k vyhodnocení, ve kterých případech byla identifikace úspěšná. Zápisu do excelu je umožněno pomocí vytvořené funkce *resultsToXls.m*.

Při otestování programu na 264 masterech s přidávanými atributy vinylové desky se podařilo dosáhnout úspěšnosti 88,6 % správně rozpoznaných nahrávek, této hodnoty se podařilo dosáhnout pro $k=4$. Pro jiné hodnoty k byla úspěšnost nepatrně nižší, proto jsem se rozhodl ve výsledném programu použít $k=4$. Výsledky testování různých hodnot k jsou uvedeny v tabulce 1.

Tabulka 1: Vliv hodnoty k na úspěšnost identifikace (pozn.: číslo v závorce udává počet správně rozpoznaných nahrávek z 264 testovaných)

k	Úspěšnost identifikace
3	86,7 % (229 nahrávek)
4	88,6 % (234 nahrávek)
5	87,8 % (232 nahrávek)
6	88,2 % (233 nahrávek)
7	88,2 % (233 nahrávek)

Zároveň byl program otestován pro všech 5 dostupných nahrávek pořízených přímo z gramofonové desky. Z těchto 5 se programu podařilo rozeznat všechny.

Pro zjednodušení úlohy a na základě výsledků z mého předchozího projektu jsem při výše uvedených testech pracoval se signálem v monofonní reprezentaci. Nakonec jsem ale otestoval, jaká by byla úspěšnost pro stereo a mid-side reprezentaci.

Pro testování úspěšností pro jiné tvary signálu je nutné znovu provést natrénování programu, pro zajištění dobrých výsledků si musí vždy odpovídat reprezentace použitá při trénování a reprezentace použitá při identifikaci.

Výsledky testování různých reprezentací signálu jsou uvedeny v tabulce 2. Výsledky byly na použité množině stejné jak pro stereo tak pro mono reprezentaci signálu. Pro signál ve tvaru mid-side se úspěšnost zvedla na 92,8 %. Při rozpoznávání nahrávek z vinylových desek se opět podařilo rozpoznat všech 5 (vždy navíc byla identifikace jednoznačná, tj. byla nabídnuta pouze jedna nahrávka s největší pravděpodobností). Rozhodl jsem se tedy ve výsledném programu použít právě tuto reprezentaci signálu.

Pro srovnání jsem program vyzkoušel i na množině samotných neupravených masterů, kde byl program schopen rozpoznat všech 264 nahrávek.

Tabulka 2: Vliv zvolené reprezentace signálu na úspěšnost identifikace (pozn.: číslo v závorce udává počet správně rozpoznaných nahrávek z 264 testovaných)

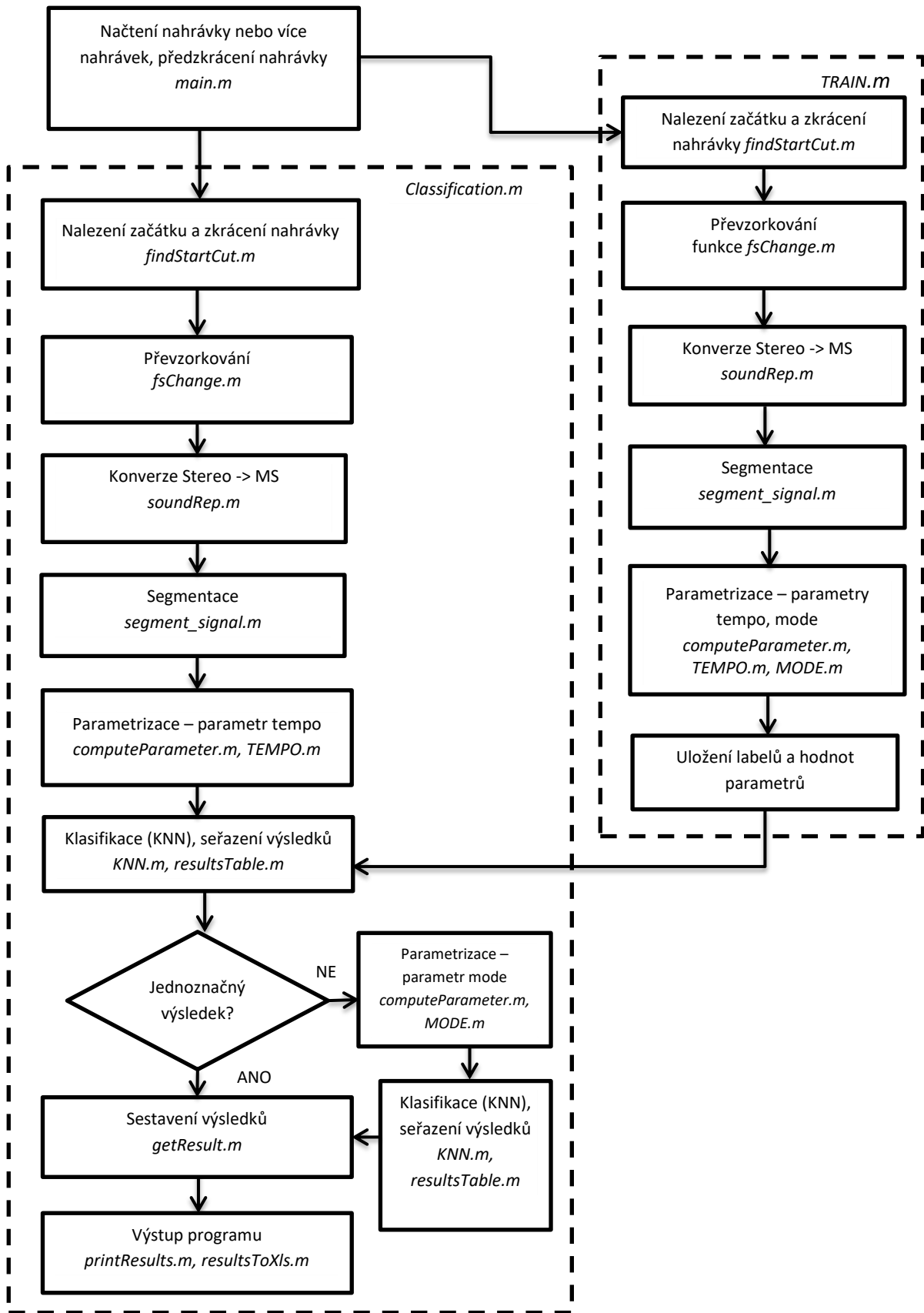
Reprezentace signálu	Úspěšnost identifikace
Mono	88,6 % (234 nahrávek)
Stereo	88,6 % (234 nahrávek)
MS	92,8 % (245 nahrávek)

Rozdíl ale není nijak veliký a může být částečně dílem náhody, že na zvolené testovací množině získala reprezentace mid-side a k=4 nejlepší výsledky. Nedá se pro reprezentaci mid-side a pro hodnotu k=4 ani najít žádné jasné zdůvodnění, proč by v tomto případě mohla být zásadně výhodnější než další dvě reprezentace a jiné hodnoty k. Proto bych v případné navazující práci i nadále považoval za vhodné pokračovat v testování všech tří reprezentací a i jiných hodnot k.

4.7 Schéma vytvořeného programu, řízení programu funkcí main

Program je spuštěn pomocí hlavní funkce *main.m*, ve které si po jejím spuštění uživatel může zvolit, jestli chce přistoupit k natrénování programu na trénovací množině z vybrané složky nahrávek, identifikovat vybranou nahrávku ze souboru, testovat úspěšnost identifikace celé složky nahrávek, nebo zaznamenat a identifikovat nahrávku z externího zdroje (například gramofonu připojeného k PC). Okamžitě po načtení nahrávky vždy následuje její zkrácení na 60 s, čímž zabráníme přetížení programu, které by mohlo nastat při práci s příliš dlouhou nahrávkou.

Po ukončení identifikace MATLAB vypíše výsledek, tedy název nahrávky, o kterou se pravděpodobně jedná. Pokud dojde k nalezení stejné pravděpodobnosti u většího počtu nahrávek, MATLAB vypíše všechny nahrávky s touto pravděpodobností. Zároveň dojde k zápisu výsledku do excelu do souboru *results.xls*.



Obr. 13: Schéma vytvořeného programu

4.8 Výpočetní náročnost

Doba nutná pro identifikaci jedné vinylové nahrávky ze souboru je přibližně 10 s. Doba je tak tedy poměrně dlouhá. Při detailnější analýze se ukázalo, že nejdéle trvá samotný výpočet tempa v MIR Toolboxu, který se pohybuje pro 20 s nahrávku mezi 5 a 8 s. Další zásadní bod je načtení souboru, které trvalo v rozmezí 0,3 s až 3 s (3 s nastaly pro načtení nahrávky z gramofonové desky o délce 18 minut). Zbylé operace zaberou ve srovnání s těmito zmíněnými zanedbatelné časy.

Tyto časy byly naměřeny na PC s dvoujádrovým procesorem a taktovací frekvencí 2,7 GHz.

Čas nutný pro načtení souboru snadno eliminujeme, pokud přistoupíme ke zkrácení nahrávky už před samotnou identifikací v programu (načtení například 18 minutové nahrávky do programu je zbytečné, protože ihned po načtení dojde ke zkrácení nahrávky.)

Čas nutný k výpočtu tempa je dán složitým výpočtem nutným k získání tohoto parametru. Pro snížení tohoto času by bylo nejspíše nutné neprovádět pro výpočet tempa segmentaci a vypočítat tempo rovnou z celého úseku, tím by se potřebná doba výrazně zmenšila, došlo by ale ke ztrátě informace o případném vývoji tempa v celém úseku. Především pokud se jedná o začátek nahrávky, který může obsahovat například pozvolný začátek a zrychlování skladby, může být tato informace velmi cenná. Bylo by tedy pravděpodobně nutné najít vhodný kompromis mezi dobou, která poskytuje dostatečnou rozlišovací schopnost a dobou, která je přijatelnou dobou výpočtu.

Zároveň by bylo možné na základě již napsané funkce pro tempo v MIR Toolboxu implementovat vlastní podobnou funkci a pokusit se ji optimalizovat pro účely tohoto programu.

4.9 Srovnání metod klasifikace

V této práci jsem využil metodu strojového učení KNN. Bakalářská práce [2], která se zabývala podobnou problematikou, místo metody strojového učení používala principu relační databáze. Při srovnání obou metod ale zjistíme, že v našem případě, díky jeho specifičnosti, kdy každé nahrávce náleží vlastní třída, není v metodách velký rozdíl. Pro oba přístupy se nejprve provede natrénování (tedy spočítání parametrů), poté na základě vzdálenosti hodnoty parametrů získaných pro zkoumanou nahrávku od hodnot získaných z trénování se najde nejbližší nahrávka. Způsob vyhledávání se mírně odlišuje, je dán ale především tím, že v práci [2] nebyla použita segmentace a parametry se počítaly z celého úseku nahrávky.

Možné řešení by byla i metoda ANN, ale ačkoliv se její použití doporučuje právě při velkém počtu tříd [1], pro náš případ by nejspíše nepřinesla žádné výhody. Kvůli stejnému množství nahrávek jako tříd, by opět fungovala podobně jako obě již vyzkoušené metody.

Aby nějaká z metod strojového učení (KNN nebo ANN) měla větší přínos a došlo k plnému využití jejího potenciálu, přicházelo by v úvahu při trénování trénovat klasifikátor nejen na masterech, ale na nahrávkách, které prošly simulací vlastností gramofonových desek. Tedy při trénování bychom pro každou nahrávku postupně prováděli přidání atributů, pokaždé s mírně odlišnými hodnotami (tedy například různé odstupy signálu šumu, různé typy šumu, různé parametry dolní propusti).

Tuto metodu by však mělo smysl vyzkoušet ve chvíli, kdy bychom měli k dispozici dostatečné množství skutečných gramofonových nahrávek. Zároveň by byla výpočetně náročnější a není jisté, že by přinesla lepší výsledky.

Závěr

V této bakalářské práci jsem se zabýval metodami pro identifikaci obsahu gramofonových desek. V teoretické části jsou nastíněny odlišné přístupy k této problematice, které používají další softwary zabývající se identifikací nahrávky podle zvukové ukázky.

Zvolená klasifikační metoda byla v této práci metoda strojového učení s učitelem KNN. Metodu se podařilo v algoritmu implementovat úspěšně a dosáhnout s ní dobrých výsledků, zároveň se ale ukázalo, že při této úloze není zcela využito celý její potenciál a dalo by se uvažovat i o využití jednodušších přístupů (například relační databáze použité v práci [2]).

Jako parametry byly zvolené již implementované parametry z MIR Toolboxu. Zvolil jsem parametry z hudební oblasti tempo a mode. Výhoda těchto parametrů je, že na rozdíl od parametrů popisujících energii signálu a barvu tónu nedojde k přílišné změně jejich hodnoty ani po upravení vlastností nahrávky a simulaci vlastností gramofonové desky.

Při testování úspěšnosti programu se podařilo dosáhnout úspěšnosti 92,8 % při identifikaci masterů s přidávanými simulovanými vlastnostmi gramofonových nahrávek a zároveň se podařilo správně rozpoznat všech 5 reálných vinylových nahrávek.

Vzhledem k tomu, že program v obou případech musel rozpoznávat nahrávky, které měly od původních značně odlišné vlastnosti, dá se výsledek testu označit za dobrý, i když pro praktické využití by jistě bylo nutné úspěšnost ještě zvýšit a otestovat program na více reálných nahrávkách.

Ve srovnání s bakalářskou prací [2], kde byla úspěšnost pro různá nastavení atributů gramofonové desky v rozmezí 82-96 % (82 % pro přidání bílého šumu s odstupem 50 dB a 96 % pro přidání bílého šumu s odstupem 70 dB) zjistíme, že se v mé práci s dosaženou úspěšností 92,8 % tyto výsledky sice nepodařilo překonat, ale úloha v mé práci byla komplikovanější díky zahrnutí více atributů desky zároveň a jejich věrnější simulaci. Zároveň v mé práci již bylo testování prováděno na nahrávkách z reálných gramofonových desek, kdy byl program po natrénování pouze na masterech schopen rozeznat všech 5 nahrávek z gramofonových desek.

Nevýhoda vytvořeného programu je velká výpočetní náročnost, kdy pro identifikaci jedné nahrávky program potřebuje přibližně 10 s. Tato doba je dána především použitím parametru tempo a jeho počítáním pro všechny segmenty úseku nahrávky. V kapitole 4.8 jsou nastíněny možné postupy pro snížení této náročnosti, které by bylo možné dále zkusit využít. Bakalářská práce [2] dosahuje výrazně vyšší rychlosti, kdy doba nutná pro identifikaci jedné nahrávky je přibližně 2,8 s. Vyšší rychlosti je dosaženo především proto, že výpočetně náročný parametr tempo je zde počítán pro celý úsek nahrávky, a není počítán pro více segmentů.

Jelikož v mé práci bylo nastavení programu optimalizované na základě testování na reálných gramofonových deskách, domnívám se, že by při využití na rozsáhlejší množině reálných desek přineslo lepší výsledky než program v bakalářské práci [2]. Nicméně metody použité v této práci i v práci [2] se ukázaly jako funkční pro rozpoznání nahrávky z gramofonové desky a při otestování na větším počtu nahrávek pořízených z gramofonových desek by bylo možné jejich nastavení dále upravit tak, aby dosáhly co největší úspěšnosti.

Literatura

- [1] Koshkina, E., Identifikace obsahu archivních zvukových záznamů, FEL ČVUT, 2015
- [2] Král, V., Identifikace gramofonové desky podle krátké zvukové ukázky, FEL ČVUT, 2016
- [3] Bolek, M., Detekce vad u mechanického záznamu zvuku, FEL ČVUT, 2016
- [4] Lartillot, O., MIRtoolbox 1.5 User's Manual, Finnish Centre of Excellence in Interdisciplinary Music Research, 2013
- [5] Eargle, J., Handbook of Recording Engineering, 1986
- [6] Wikipedia, „SoundHound“, [cit. 5. 2. 2017], [online], dostupné z:
<https://en.wikipedia.org/wiki/SoundHound>
- [7] Převod analogových nahrávek, „Formáty mechanického záznamu zvuku“, [cit. 5. 2. 2017], [online], dostupné z:
<http://www.vhs-na-dvd.eu/gramofony/formaty>
- [8] Pro Audio Files, „The Basics of Mid-Side Processing“, [cit. 5. 2. 2017], [online], dostupné z:
<https://theproaudiofiles.com/mid-side-processing/>
- [9] Wikipedia, „K-nearest neighbors algorithm“, [cit. 4. 2. 2017], [online], dostupné z:
https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm
- [10] Dušek, K., Záznam a reprodukce zvuku. 5 vydání. Praha: Vydavatelství ČVUT, 1982
- [11] Wikipedia, „Acoustic fingerprint“, [cit. 3. 2. 2017], [online], dostupné z:
https://en.wikipedia.org/wiki/Acoustic_fingerprint
- [12] Sound Matters, „Vinyl Record Inner-Groove Distortion“, [cit. 8. 2. 2017], [online], dostupné z:
<http://www.soundmattersblog.com/vinyl-record-inner-groove-distortion-simple-explanation/>
- [13] Ghias A., Logan J., Chamberlin D., Smith B. C. „Query By Humming – Musical Information Retrieval in an Audio Database“, ACM Multimedia 1995, [cit. 14. 2. 2017], [online], dostupné z:
<http://www.cs.cornell.edu/zeno/papers/humming/humming.html>
- [14] Wikipedia, „Tempo“, [cit. 11. 5. 2017], [online], dostupné z:
<https://cs.wikipedia.org/wiki/Tempo>
- [15] Wikipedia, „Modus“, [cit. 11. 5. 2017], [online], dostupné z:
[https://cs.wikipedia.org/wiki/Modus_\(hudba\)](https://cs.wikipedia.org/wiki/Modus_(hudba))
- [16] Zvukové ukázky vad mechanického záznamu, GZmedia a.s. 2016
- [17] Wikipedia, „Artificial neural network“, [cit. 11. 5. 2017], [online], dostupné z:
https://en.wikipedia.org/wiki/Artificial_neural_network
- [18] Wang A. Li-Chun, An Industrial-Strength Audio Search Algorithm, 2003, [online], dostupné z:
<https://www.ee.columbia.edu/~dpwe/papers/Wang03-shazam.pdf>
- [19] Wikipedia, „RIAA equalization“, [cit. 11. 5. 2017], [online], dostupné z:
https://en.wikipedia.org/wiki/RIAA_equalization

[20] Gershenson C., „Artificial Neural Networks for Beginners“, [cit. 18. 5. 2017], [online], dostupné z: <https://arxiv.org/abs/cs/0308031>

[21] Cano P., Batlle E., Kalker T, Haitsma J., „A Review of Algorithms for Audio Fingerprinting“, In International Workshop on Multimedia Signal Processing, US Virgin Islands, Prosinec 2002, [cit. 18. 5. 2017], [online], dostupné z: <http://mtg.upf.edu/files/publications/MMSP-2002-pcano.pdf>

[22] „Gettin' Serious About Vinyl: Wow, Flutter, and S/N Ratio“, 120studio.com, 2015, [cit. 18. 5. 2017], [online], dostupné z: <http://120studio.com/audio/wowflutter.htm>

Přílohy

Na přiloženém CD se nacházejí následující adresáře:

- xls_results
- Spectrograms
- MATLAB

Ve složce *xls_results* se nachází soubory formátu *.xls*, ve kterých jsou uloženy výsledky testování popsaných v této práci.

Ve složce *Spectrograms* jsou uloženy obrázky porovnávající spektrogramy masterů, nahrávek pořízených z gramofonových desek a masterů s přidanými atributy gramofonové desky a grafy vzniklé součtem výkonů ve spektrogramu podél časové osy (vizte kapitola 4.5.3).

Ve složce MATLAB se nachází všechny funkce a skripty využitě ve vytvořeném programu. Zároveň je zde k dispozici stručný návod k práci s programem, soubor *train.mat* obsahující data získaná z trénovací množiny, a složka *noise*, která obsahuje nahrávky šumů nutné k simulaci atributů vinylové desky.

Seznam skriptů a funkcí v příloze:

- *addNoise.m*: přidání šumu k nahrávce a snížení její amplitudy
- *CLASSIFICATION.m*: řízení procesu klasifikace, funkce vznikla úpravou skriptu *CLASSIFICATION_main.m* z práce [1]
- *computeParameter.m*: umožní spočítání vybraného parametru pro všechny segmenty nahrávky
- *findStartCut.m*: nalezení začátku nahrávky a její zkrácení, založeno na algoritmu popsaném v práci [2]
- *fsChange.m*: převzorkování nahrávky, funkce je převzata z bakalářské práce [1], kde se vyskytuje jako funkce *prevzorkovani.m*
- *getResults.m*: sestavení matice výsledků, případné aplikování hierarchie na výsledky získané ze dvou parametrů, využita a pro použití v této práci upravena část kódu ze skriptu *CLASSIFICATION_main.m* z práce [1]
- *KNN.m*: klasifikace pomocí metody strojového učení KNN, převzato z bakalářské práce [1] a upraveno pro využití v této úloze
- *lowPass.m*: dolní propust implementovaná pomocí Butterworthova filtru
- *main.m*: řízení celého programu, dává uživateli možnost přehledného ovládání programu
- *MODE.m*: výpočet parametru mode, převzato z práce [1], ve funkci je volána funkce *mirmode* z MIR Toolboxu [4]
- *nr2Ltr.m*: převedení čísla sloupce na jeho označení v excelu, pomocná funkce pro funkci *resultsToXls.m*
- *printResults.m*: umožňuje výpis výsledků klasifikace
- *recordSound.m*: umožňuje nahrání zvuku z externího zdroje

- *resultsTable.m*: seřazení výsledků získaných klasifikací KNN, využito část kódu ze skriptu *CLASSIFICATION_main.m* z práce [1]
- *resultsToXls.m*: zápis výsledků do excelu
- *segment_signal.m*: segmentace signálu, převzato z bakalářské práce [1]
- *shiftStart.m*: posunutí začátku nahrávky
- *soundCut.m*: zkrácení nahrávky
- *soundRep.m*: převedení signálu do libovolné reprezentace signálu
- *TEMPO.m*: výpočet parametru tempo, převzato z práce [1], ve funkci je volána funkce *mirtempo* z MIR Toolboxu [4]
- *TRAIN.m*: trénování klasifikátoru, funkce vznikla úpravou skriptu *TRAIN_main.m* z práce [1]
- *vinylSim.m*: řízení přidání atributů vinylové desky