Západočeská univerzita v Plzni
Fakulta aplikovaných věd
Katedra kybernetiky

BAKALÁŘSKÁ PRÁCE

Plzeň,2017

Viktor März

Prohlášení

Předkládám tímto k posouzení a obhajobě bakalářskou práci zpracovanou na závěr studia na Fakultě aplikovaných věd Západočeské univerzity v Plzni.

Prohlašuji, že jsem bakalářskou práci vypracoval samostatně a výhradně s použitím odborné literatury a pramenů, jejichž úplný seznam je její součástí.

V Plzni dne 18. května 2017

……

Poděkování

Rád bych poděkoval vedoucímu této bakalářské práce Ing. Mgr. Josefu Psutkovi, Ph.D. za jeho podporu a čas, který mi věnoval při řešení této práce.

Anotace

Tato bakalářská práce se zabývá detekcí hudby v mixovaném signálu (hudba/řeč). Její teoretická část je zaměřena na základní přístupy zpracování signálu. Jsou zde popsány základní metody zpracování signálu jak v časové tak i ve frekvenční oblasti (krátkodobá energie signálu, krátkodobý průchod nulou, krátkodobé spektrum, MFCC). Použitelnost těchto metod je pak ověřena na jednoduché úloze detekce znělky v přenosech z PSPČR.

Klíčová slova

Detekce hudby, MFCC, K-means algoritmus, Zero-crossing, Root Mean Square, rychlá fourierova transformace

Annotation

This bachelor thesis deal with problem of music detection in mixed signa (music/speech). The teoretical part is focused on basic approaches of signal processing. There are described methods of signal processing in time and frequency domains (Short-time signal energy, short-time zero-crossing, short-time spectrum, MFCC). The applicability of these methods is then verified by a simple task of detecting the sound in recording from House of Deputies of the Czech Republic.

Key words

Music detection, MFCC, K-means algorithm, Zero-crossing, Root Mean Square, fast Fourier transformation

Obsah

Úvod

Základním a nejpřirozenějším způsobem přenosu informace mezi lidmi je komunikace prostřednictvím mluvené řeči. Proto se nelze divit, že při současném pokroku výpočetní techniky usilují vědci po celém světě, aby se i počítač mohl stát plnohodnotním partnerem člověka v mluveném dialogu. Tento cíl, by mohl být člověku velmi prospěšný proto, že takový způsob komunikace může člověku velmi často usnadnit život. Například automatickým titulkováním mluvené řeči pro neslyšící.

V rozpoznávání mluvené řeči, která je zcela automatický proces, je několik překážek, které zabraňují bezproblémovému rozpoznávání řeči. Například hlas každého člověka je unikátní, v pozadí řečníka může být nějaký hluk, který přerušuje řečníka a tak dále. Jednou z těhle překážek je i hudba v řeči. Hudební vložky v signálu zabraňují správnému fungování automatického titulkování.

Během mého studia jsem absolvoval mnoho předmětů, které se zabývali zpracováním signálu a dat. Velmi mne zaujalo tohle téma, protože rozdíly mezi hudbou a řečí jsou velmi výrazné, například energie hudby je zcela odlišná, než energie řeči.

Tato bakalářská práce se zabývá základním zpracováním zvukového signálu a prostudováním základních metod detekce hudby v mixovaném signálu, tj. hudba/řeč. Jako modelovou úlohu, se zvolily zvukové nahrávky z jednání Poslanecké sněmovny Parlamentu České republiky, protože v těhle zvukových signálech lze jasně odlišit mluvené slovo od hudby.

Dále se tato práce zabývá základnímy metodami zpracování signálu a možnými algoritmy na detekci hudby. Vybrané algoritmy budou zrealizovány a vyzkoušeny na testovacích datech. Získané výsledky budou poté vyhodnoceny..

1 Teoretický úvod

Lidé jsou schopni rozlišit hudbu a řeč po celou dobu své existence bez nějaké vědomé snahy. Rozpoznávají hlasy při telefonním hovoru, poznají rozdíl mezi zvonkem na dveřích a vyzváněním telefonu.

V mnoha aplikacích existuje velký zájem o segmentaci a klasifikaci zvukových signálů. První kategorií může být klasifikace do 2 tříd, hudba a vše ostatní. Další kategorií může být klasifikace do tříd ticho, hudba a řeč. V současnosti pracujeme pouze s první kategorií, s klasifikací do hudby a vše ostatní.

V minulosti bylo navrženo mnoho algoritmů na segmentaci zvukového signálu. Některé z nich jsou prozkoumány v téhle práci.

1.1 Popis zvuku a jeho charakteristika

Zvuk je fyzikální veličina, která přenáší nějakou zprávu vzuchem, kapalinou nebo pevnými částicemi. Zpráva může obsahovat určité množství informace. Zvuk může být reprezentován popisem závislosti jednoho parametru, závislá proměnná, na parametru jiném,nezávislá proměnná. Nejčastěji je jako nezávislá proměnná čas.

2 Zpracování signálu

Zvukový signál se dá zpracovávat v několika oblastech. Nejzákladnější jsou časová a frekvenční. V časové oblasti, lze studovat jak se zvukový signál mění v čase. Ve frekvenční oblasti, lze zkoumat, jak se mění frekvenční vlastnosti zvukového signálu

2.1 Zpracování v časové oblasti

Většinu metod krátkodobé analýzy v časové oblati lze vyjádřit vztahem

$$Q_n = \sum_{k=-\infty}^{\infty} \tau\big(s(k)\big) w(n-k), \qquad (2.1)$$

Kde Q_n je krátkodobá charakteristika, *s(k)* značí vzorek audiosignálu získany Pulzně kódová modulace (PCM) v čase *k*. PCM je modulační metoda převodu analogového zvukového signálu na signál digitální. *T(.)* vyjadřuje příslušnou tranformační funkci a *w(n)* je váhová posloupnost, neboli takzvané okénko, kterým se vybírají, respektivě váží vzorky *s(k)*. Úkolem okénka je vybrat příslušné vzorky ze signálu a těm přidělit jim určitou váhu. Nejčastější se v metodách zpracování v časové oblasti používají dva druhy okének, pravoúhlé okénko a Hammingovo okénko.

U pravoúhlého okénka je stejná váha na všechny okénkém vybrané vzorky. Pravoúhlé okénko je definováno vztahem

$$w(n) = \begin{cases} 1 & pro\ \ 0 \leq n\ \leq L-1 \\ 0 & pro\ ostatní\ n \end{cases} \tag{2.2}$$

kde *L* je počet vzorků vybraných okénkem. Velmi často je snahou potlačit vzorky na krajích okénka. Při potlačování těchto vzorků je vhodné využít Hammingovo okénko, které je definováno vztahem:

$$w(n) = \begin{cases} 0{,}54 - 0{,}46\cos(\frac{2\pi n}{L-1}) & pro\ \ 0 \leq n\ \leq L-1 \\ 0 & pro\ ostatní\ n \end{cases} \tag{2.3}$$

2.1.1 Metoda míry průchodu nulou

Míru průchodu signálu nulou (angl. The zero-crossing rate) lze chápat jako jednoduchou charakteristiku popisující spektrální vlastnosti signálu. Míru průchodu nuly je rychlost změny znaménka podél signálu, to znamená rychlost, při které se signál mění z pozitvního na negativní nebo naopak. Tato funkce bývá velmi využívána jak při rozpoznávání řeči, tak při získávání hudebních informací.

Definice míry průchodu nulou je následující:

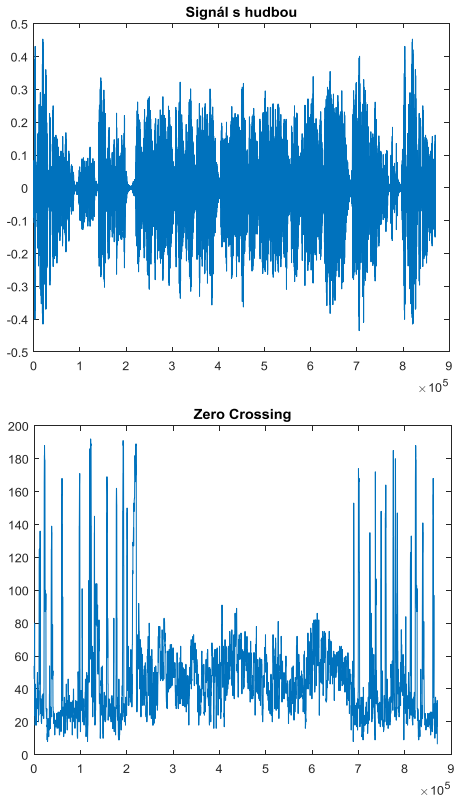
$$Z_n = \sum_{k=-\infty}^{\infty} \left| sgn\big(s(k)\big) - sgn\big(s(k-1)\big) \right| w(n-k) \tag{2.4}$$

kde

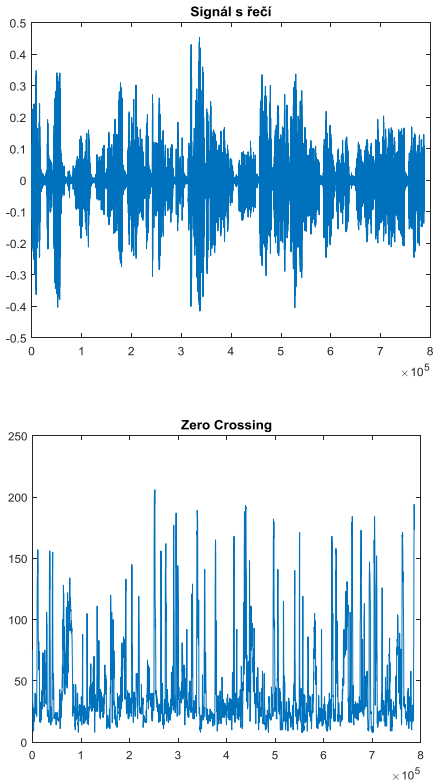
$$sgn[s(k)] = \begin{cases} 1 & pro\ \ s(k)\ \geq 0 \\ -1 & pro\ s(k)\ <\ 0 \end{cases} \tag{2.5}$$

a *w(n)* je pravoúhlé okénko.

Na obrázku 2.1a je vidět signál s hudbou a odpovídající průběh funkce středního průchodu nulou. Na obrázku 2.1b pak stejná informace pouze pro signál řečový.



Obr. 2.1a signál s řečí



Obr.2.1b signál s hudbou

2.1.2 Metoda krátkodobé energie

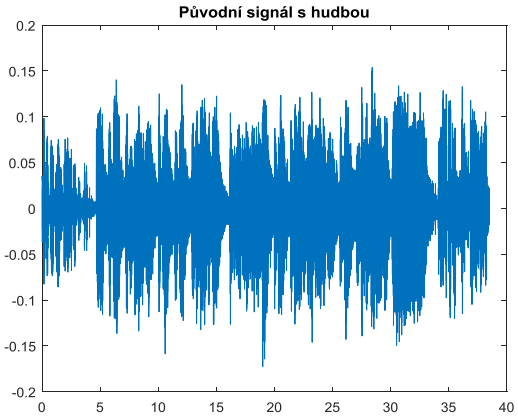
Hodnoty krátkodobé energie mohou být využívány například při automatickém oddělování segmentů hluku od segmentů ticha. Tuhle metodu lze například využít i při odělování znělých a neznělých částí signálu. Hodnoty funkce krátkodobé energie se kdysi využívaly též jako příznaky v jednoduchých klasifikátorech slov.

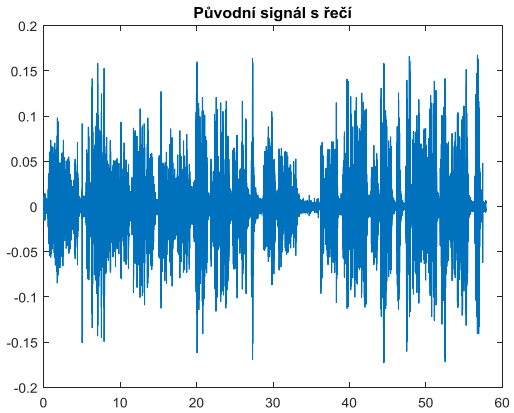
Funkce krátkodobé energie signálu lze definovath vztahem

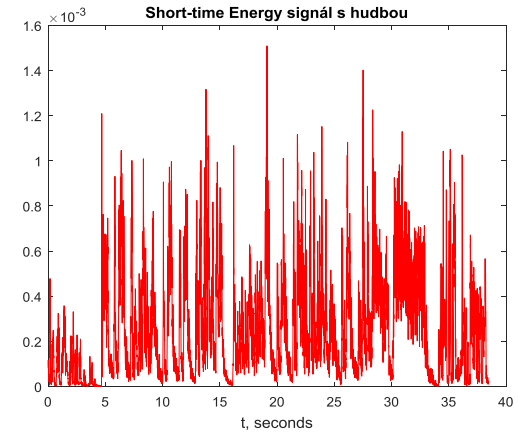
$$E_n = \sum_{n=1}^{N} [s(k)w(n-k)]^2 \qquad (2.6)$$

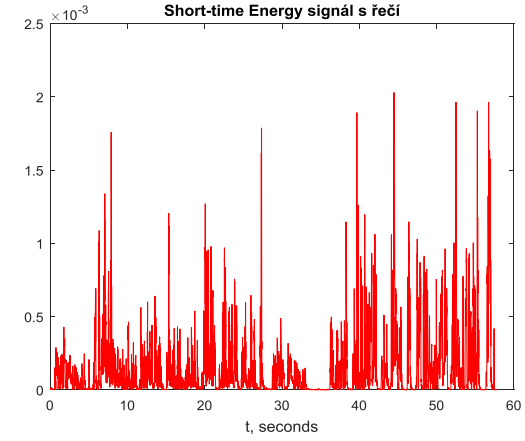
Kde *s(k)* je vzorek signálu v čase k, *w(n)* reprezentuje příslušný typ okénka.

Na obrázcích 2.2a a 2.2b lze vidět původní signál a průběh funkce krátkodobé energie. Na obrázku 2.2a lze vidět signál s hudbou a na obrázku 2.2.b signál jen s řečí.









Obr. 2.2a Signál s hudbou

Obr. 2.2b Signál s řečí

2.2 Zpracování ve frekvenční oblasti

Mluvená řeč může být ve frekvenční oblasti reprezentována jako kompozice spektrální obálky charakterizující vlastnosti mluvené řeči. Jak se mění buzení a původ signálu, vytvářejí se rozdílné zvuky a mění se i spektrum signálu. Podobně jako u metod ve zpracování v časové oblasti , tak i ve frekvenční oblasti lze pracovat s představou toho, že řečový signál je v krátkémm časovém intervalu přibližně stacionární, a proto je dobré mluvit v tomto případě o krátkodobé spektrální analýze..

2.2.1 Fourierova transformace

Fourierova transformace je integrální transformace převádějící signál mezi časově a frekvenčně závislým vyjádřením pomocí harmonických signálů, funkcí komplexní exponenciáli, tj. sinus a cosinus. Fourierova transformace slouží pro převod signálů z časové oblasti do oblasti frekvenční. Signál může být jak ve spojitém, tak i v diskrétním čase.

Krátkodobá Fourierova transformace je základním přístupem pro frekvenční analýzu. Krátkodobá Fourierova transformace je definována vztahem

$$S(\omega, n) = \sum_{k=-\infty}^{\infty} s(k)h(n-k)e^{-j\omega k} \tag{2.7}$$

Kde *h(n)* je blíže nespecifikované okénko, které vybírá pouze pro zpracování určený úsek signálu. Ze vztahu je zřejmé, že takto vyjádřená Fourierův obraz je funkcí jak spojitě proměnné frekvence *ω*, tak i diskrétně proměnného času *n* a odpovídá v podstatě konvoluci okénka *h(n)* a vzorku *s(n)* modulovaného $e^{-j\omega k}$. Lze tedy napsat:

$$S(\omega, n) = \sum_{k=-\infty}^{\infty} \left[s(n)e^{-j\omega n}\right] * h(n) \tag{2.8}$$

kde operace * značí konvoluci.

Funkce $S(\omega, n)$ je obecně komplexní funkce

$$S(\omega, n) = a(\omega, n) - jb(\omega, n) \tag{2.9}$$

Je vemi důležité uvést ještě často používaný vztah pro výpočet amplitudy.

$$|S(\omega, n)| = \sqrt{a^2(\omega, n) + b^2(\omega, n)} \tag{2.10}$$

Alternativní zápis $S(\omega, n)$ získáme úpravou základního vztahu:

$$S(\omega, n) = e^{-j\omega n} \sum_{k=-\infty}^{\infty} \mathrm{s(k)h(n-k)}e^{j\omega(n-k)} \tag{2.11}$$

Nebo

$$S(\omega, n) = e^{-j\omega n} \left\{s(n) * [h(n)e^{j\omega n}]\right\} \tag{2.12}$$

Předpokládejme, že zafixujeme čas *n*. Funkce $S(\omega, n)$ pak představuje obyčejnou Fourierovu transformaci posloupnosti *s(k)h(n-k)*, $-\infty<k<\infty$. Ovšem jestliže zafixujeme frekvenci *ω*, pak

Fourierův obraz $S(\omega, n)$ je funkcí času *n*, a protože je vyjádřen konvolucí, lze na něj pohlížet jako na výstup z lineárního filtru.

V případě, kdy se zafixuje čas *n*, je signál zpracováván nejčastěji krátkodobou diskrétní Fourierovou transformací, jsou čas i frekvence diskrétní. Získané koeficienty se využívají dále hlavně ve spektrálních analyzátorech řeči, nebo řečových syntezátorech.

V druhém případě, kdy se zafixuje frekvence ω, lze na Fourierovu analýzu pohlížet jako na proces lineární filtrace a zkoumat spektrální vlastnosti signálu.

2.2.2 Rychlá Fourierova transformace

Rychlá Fourierova transformace je způsob výpočtu diskrétní Fourierovi transformace, kterým získáme stejné výsledky ale mnohem rychleji.

V klasické podobě lze provádět pro signály, u nichž bylo sejmuto 2^n vzorků. V současné době existují už i sofistikovanější algoritmy, které umožňují provést rychlou transformaci pro libovolný počet vzorků.

Rychlá Fourierova transformace je definována vztahem

$$F_n = \sum_{k=0}^{N-1} f_n e^{-j\frac{2\pi}{N}kn}, \qquad n = 1,2,\ldots\ldots,N-1 \tag{2.12}$$

kde $\frac{2\pi}{N}$ je frekvence ω.

2.3 Parametrizace signálu

Pod pojmem parametrizace si lze představit převod zvukové nahrávky například ve formátu .wav na posloupnost vektorů parametrů. Po parametrizaci dostaneme pro každý mikrosegment velikosti přibližne 30ms (velikost odvozena od stacionarity signálu), vektor parametrů (obvykle s dimenzí 13). Pod pojmem mikrosegment si lze tedy představit časový úsek o velikosti několika desítek milisekund.

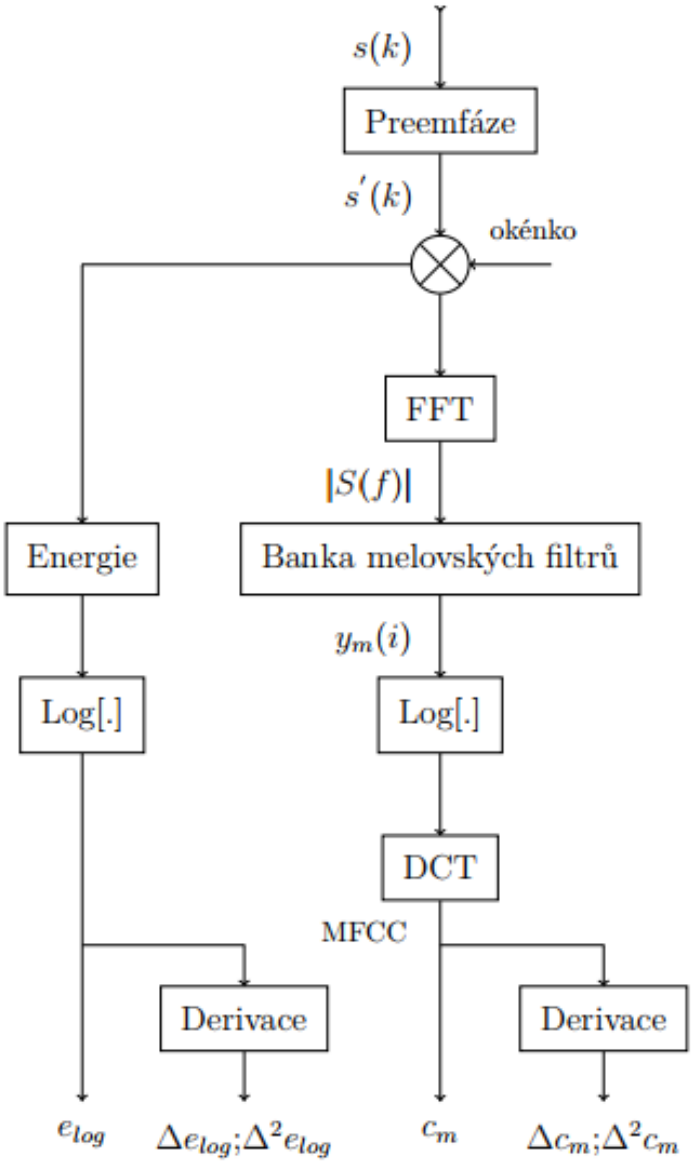
2.3.1 Melovské frekvenční kepstrální koeficienty MFCC

Tato metoda patří do oblasti Homomorfní analýzy, která je ze skupiny postupů nelineárního zpracování signálů využívajících principu superpozice. Tyto postupy se zaměřují na analýzu signálů, vznikajících konvolucí dvou nebo více složek. Což odpovídá řečovému signálu, jelikož jeho vytváření se dá vnímat i jako konvoluce budící funkce a impulsní odezvy hlasového ústrojí. Kde budící funkce má podobu periodického sledu pulsů u znělých hlásek a náhodného šumu u neznělých.

Metoda MFCC využívá zpracování řečového signálu jak v časové tak i v frekvenční oblasti. Popisuje při tom spektrální vlastnosti zmíněného signálu, konkrétně krátkodobé komplexní kepstrum. Snaží se při tom o respektování poznatků o citlivosti lidského ucha při vnímání zvukového vlnění na různých frekvencích.

Rozšířením této metody o kalkulaci delta a akceleračních koeficientů, získáme aditivní informaci o dynamickém průběhu řeči. Cílem metody Melovských frekvenčních kepstrálních koeficientů je určit parametry řečového systému pomocí homomorfní filtrace. Výsledkem analýzy pro každý mikrosegment je pak vektor čísel, které popisují danou část řečového signálu.

Obrázek 2.3 představuje schéma algoritmu výpočtu MFCC s delta a akceleračními koeficienty



Obr. 2.3 schéma algoritmu výpočtu MFCC

Proces určení melovských kepstrálních koeficientů lze popsat následujícím postupem.

Na vstup systému jsou přivádění vzorky řečového signálu *s(k)*, dále je provedena preemfáze signálu a na mikrosegmenty signálu (obvykle délky 10 až 30ms) je aplikováno nejčastěji Hammingovo okénko. Přesná časová délka okénka se volí rovna mocnině 2 vzhledem k následujícímu zpracování rychlou Fourierovou transformací (FFT). Přitom je doporučováno posouvat okénko o časový úsek 10ms, to znamená výsledná parametrizace řečového signálu je vyčíslována 100krát za sekundu.

V dalším kroku zpracování se pomocí FFT vypočte amplitudové spektrum |*S(f)*| analyzovaného signálu. Nejdůležitější část celého zpracování je melovská filtrace. Tento

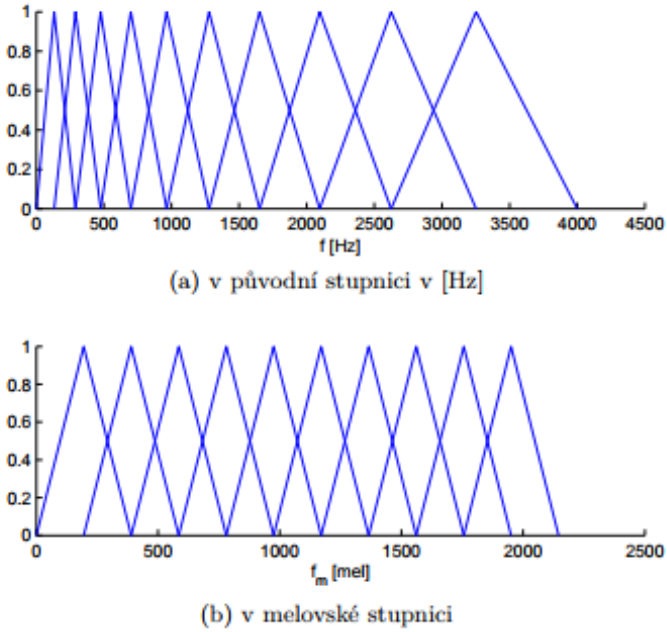
algoritmus je realizován bankou trojúhelníkových pásmových filtrů s rovnoměrným rozložením středních frekvencí jednotlivých troúhelníkových filtrů podél frekvenční osy s měřítkem v melovské škále.

Trojúhelníkové filtry jsou standartně rozloženy přes celé frekvenční pásmo od nuly až do Nyqistovy frekvence. Na obrázku 2.4 lze vidět příklad takové banky filtrů, v níž každý filtr má trojúhelníkovou pásmovou frekvenční odezvu a vzdálenosti filtrů a rovněž jejich šířka pásma jsou určeny konstantním intervalem v melovské frekvenční škále, přitom pro střední frekvence jednotlivých filtrů $b_{m,i}$ platí v melovské škále vztah

$$b_{m.i} = b_{m.i-1} + \Delta_m, \tag{2.13}$$

Kde $b_{m.i}$ = 0 mel, i=1,2,...,M^*,a Δ_m=B_{mw}/(M^*+1),

Odezvy jednotlivých filtrů mají v melovské frekvenční škále tvar rovnoramenných trojúhelníků a jsou rovnoběžně rozloženy ve frekvenci (Obr.2.4)



Obr. 2.4 Odezvy melovských filtrů

Pro výpočet odezev filtrů se ovšem musí přepočítat všechny koeficienty FFT do melovské frekvenční škály. Spíše se používá alternativí postup, kde je vyjádření trojúhelníkových filtrů ve frekvenční škále s měřítkem v herzích při současném využití původních koeficentů získaných FFT. Další postup pak spočívá v přepočtu všech středních frekvencí $b_{m,i}$ i=1,...,M*+1, s využitím inverzního vztahu (2.14) na střední frekvence vyjádřené v jednotce [Hz], poté lze odezvy filtrů vyjádřit vztahem (2.15)

$$f_m = 2595 log_{10}(1 + \frac{f}{100}), \tag{2.14}$$

Kde *f*[Hz] je frekvence v lineární škále a f_m[mel] je odpovídající frekvence v nelineární melovské škále.

$$y_m(i) = \sum_{f=b_{i-1}}^{b_{i+1}} |S(f)| u(f,i), \qquad i=1,2,....M^* \tag{2.15}$$

kde frekvence *f* jsou vybírány ze souboru frekvencí využívaných při výpočtu FFT a *u(f,i)* je vyjádření trojúhelníkovéhu filtru, který lze popsat vztahem (2.16)

$$u(f,i) = \begin{cases} \frac{1}{b_i - b_{i-1}}(f - b_{-1}) & \text{pro } b_{i-1} \leq \mathrm{f} \leq b_i \\ \frac{1}{b_i - b_{i+1}}(f - b_{+1}) & \text{pro } b_i \leq \mathrm{f} \leq b_{i+1} \\ 0 & \text{pro ostatní případy} \end{cases} \tag{2.16}$$

Další krok je výpočet logaritmu výstupů $y_m(i)$ jednotlivých filtrů. Jako posledním krokem při výpočtu melovských kepstrálních koeficientů je provedení zpětné diskrétní Fourierovy transformace (IDFT). Vzhledem k tomu, že výkonové spektrum je reálné a symetrické, bude se IDFT redukovat na diskrétní kosinovu transformaci

$$c_m(j) = \sum_{i=1}^{M^*} log y_m(i) \cos\left(\frac{\pi j}{M^*}(i - 0{,}5)\right), \qquad \text{pro } j = 0,1,\dots,M, \tag{2.17}$$

Kde *M** je počet pásem melovského pásmového filtru a *M* je počet melovských kepstrálních koeficientů.

2.4 Shluková analýza

Shluková analýza se zabývá metodami a algoritmy, pomocí kterých klasifikuje objekty s podobnými vlastnostmi do shluku, tak aby si objekty náležící do stejného shluku byly podobnější než objekty z ostatních shluků. Shluk je skupina objektů, které jsou si navzájem podobné a rozdílné od objektu do této skupiny nepatřících.

Shlukovací algoritmy mají široké využití. Například k analýze sociálních sítí, kde lze rozdělit komunitu uvnitř velkých skupin lidí. Dále lze shlukovou analýzu použít ke zpracování obrazu, kde se rozděluje digitální obraz na určité oblasti kvůli detekci hran, nebo rozpoznání objektů.

Agloritmy shlukové analýzy můžeme rozdělit na hierarchické a nehierarchické.

Hierarchické metody využívají dříve nalezených shluků a vytvoří z nich shluky nové. Průnikem každých dvou podmnožin při hierarchickém shlukování je buť prázdná množina, nebo jedna z původních.

Na druhou stranu divizní algoritmy berou vstupní množinu objektů jako celek a ten pak dělí. V každém kroku dělí shluk na dva nové, které nejlépe splňují dané kritérium rozkladu.

2.4.1 K-means algoritmus

K-means je často používaný algoritmus nehierarchické shlukové analýzy. Předpokládá se, že shlukované objekty lze chápat jako body v nějakém euklidovském prostoru a že počet shluků *k* je předem dán. Případně je možné vyzkoušet různá *k* a výsledky poté porovnat. Každý shluk je definován svým centroidem, což je bod ve stejném prostoru jako shlukovaný objekt. Objekty se zařazují do toho shluku, jehož centroidu jsou nejblíže.

K výpočtu jednotlivých vzdáleností mezi centroidy a objektem se používá Euklidovská vzdálenost jejíž výpočet je následující:

$$D_E = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \tag{2.18}$$

Kde první bod má souřadnice *[x_1,x_2]* a druhý bod má souřadnice *[y_1,y_2].*

2.4.1.1 Princip algoritmu K-means

Označí se *$T_i(k)$* množina vektorů *i*-tého shluku v k-tém kroku algoritmu, *$v_i(k)$* centroid i-tého shluku v k-tém kroku, *$J_i(k)$* dílčí hodnota kritéria i-tého shluku v *k*-tém kroku a *$n_i(k)$* počet vektorů x ve shluku *T_i* v k-tém kroku algoritmu. Výstupem algoritmu je *L* centroidů v_i, které reprezentují navržený vektorový kvantizér. Vektory, které náleží k jednotlivým shlukům T_j, společně s aplikovanou mírou zkreslení jsou podkladem pro vymezení oblastí X_j pro *j*=1,...,*L*.

Vlastní algoritmus má poté následující postup:

1) Vybere se *L* počátečních centroidů *v_1*(1*)*, *v_2*(1), v_3*(1),..., *v_L*(1)

2) V k-tém iterativním kroku se rozdělí vektory trénovací množiiny *T* do *L* shluků *$T_1(k)$, $T_2(k)$,...*, *$T_L(k)$* podle vztahu:

$$x \in T_j(k),\ jestliže \quad d\left(x, v_j(k)\right) < \quad d\left(x, v_i(k)\right) \tag{2.19}$$

pro všechna *i,j*=1,....,*L* a *i≠j*. Tento vztah se postupně aplikuje na všechny vektory trénovací množiny *T*.

3) Z výsledků z bodu 2 se vypočítá pro každý shluk nový centroid *$v_i(k+1)(j=1,...,L)$* tak, aby suma měr zkreslení, zde jsou to standartně kvadráty vzdálenosti (Euklidovská vzdálenost), všech vektorů v *$T_j(k)$* vzhledem k novému centroidu byla minimální. Centroid *$v_j(k+1)$*, který minimalizuje kritérium

$$J_j(k+1) = \sum_{x \in T_j(k)} d^2(x, v_j(k+1),\ \ j = 1, \dots, L \tag{2.20}$$

lze určit z následujícího vztahu

$$v_j(k+1) = \frac{1}{n_n(k)} \sum_{x \in T_j(k)} x\,,\ \ j = 1, \dots, L \tag{2.21}$$

4) Jestliže $v_j(k+1)=v_j(k)$ pro všechna j=1,...,L, nebo jestliže pokles celkového zkreslení $J(k)$, kde

$$J(k) = \sum_{i=1}^{L} J_i(k) \tag{2.22}$$

je v k-té iteraci ve vztahu $J(k-1)$ pod předem definovaným prahem, se ukončí činnost algoritmu. V opačném případě se pokračuje krokem 2.

2.5 Hodnocení algoritmů rozpoznávání hudby

Cílem hodnocení je poskytnou přehled veličin pro vyhodnocení úspěšnosti dektektoru hudby ve zvukovém signálu.

2.5.1 Detekční schopnost

Jedná se o poměr mezi počtem úspěšně nalezených vzorků s hudbou N_{OK} a počtem všech vzorků s hudbou ve zvukovém signálu $N_{MU/all}$ definuje detekční schopnost (detection rate-DR), někdy také nazývané jako úspěšnost detekce.

$$DR[\%] = \frac{N_{OK}}{N_{MU/all}} * 100 \tag{2.23}$$

2.5.2 Falešné poplachy

Falešné poplachy, neboli FA, můžeme posuzovat různými způsoby. Vzhledem k celkovému počtu všech vzorků ve zvukovém signálu N_{signal}

$$FA^{MUSIC}[\%/signal] = \frac{N_{FA}}{N_{signal}} * 100 \tag{2.24}$$

Kde N_{FA} je celkový počet falešných poplachů ve zvukovém signálu. V podstatě ekvivalentní je místo počtu všech vzorků v testu použít celkovou délku zvukového signálu DUR_{test} v hodinách

$$FA^{TIME}[\%/hod] = \frac{N_{FA}}{DUR_{test}} * 100 \tag{2.25}$$

Další možností je provést vyhodnocení vůči počtu všech hudebních vzorků ve zvukovém signálu $N_{MU/all}$

$$FA^{MU}[\%/MUSIC] = \frac{N_{FA}}{N_{MU/all}} * 100 \quad (2.25)$$

Pro tuhle práci se jako kritérium *FA* zvolila metoda, kde se *FA* posuzují vzhledem k celkovému počtu vzorků v signálu.

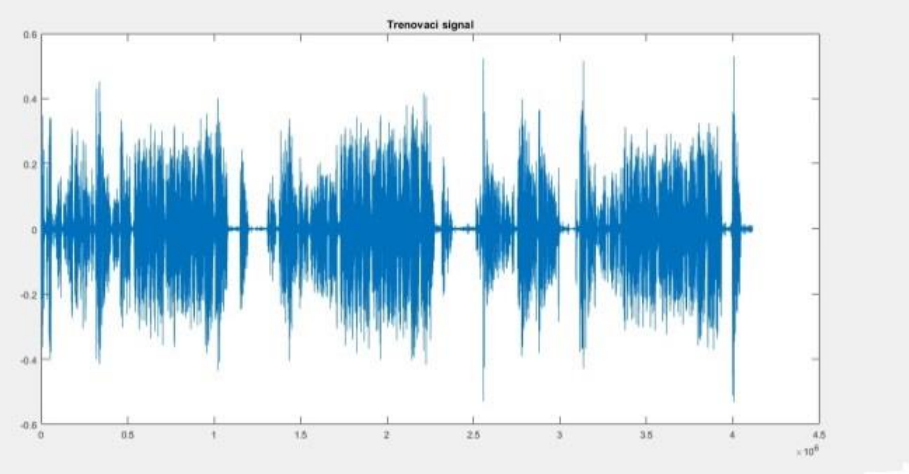
3 Praktická část

Při automatickém titulkování je velký problém s hudbou. Program na automatické titulkování neví co má psát když začne hrát nějaká znělka nebo písnička. Proto je dobré zaměřit se na algoritmy, které detekují hudební část a určí, kdy program nemá titulkovat.

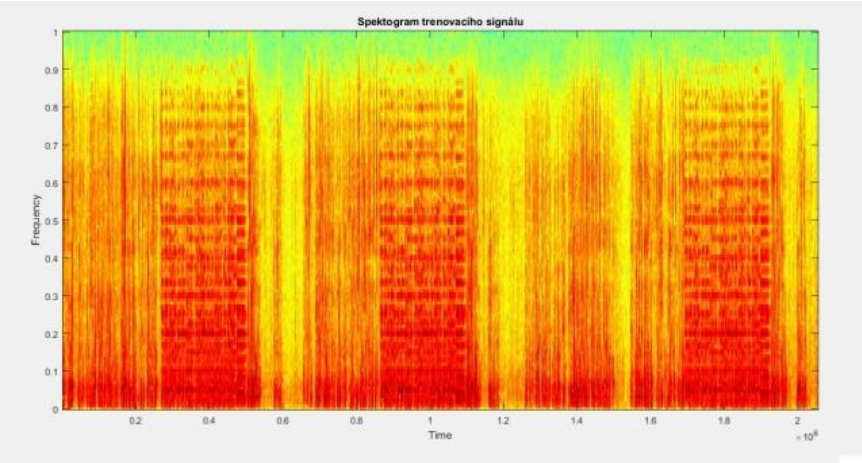
3.1 Trénovací a testovací signály

Jako trénovací i testovací signál se vybral zvukový záznam z jednání z Poslanecké sněmovny Parlamentu České republiky.

Jak lze vidět z obrázku, na první pohled nelze přesně určit kde je hudba. Ovšem, když se podíváme na spektogram, tam už lze z grafu vyčíst, kde by mohla hudba být. V trénovacím signálu se nacházejí 3 hudební vložky. Celý signál má délku 4minuty a 17 sekund.

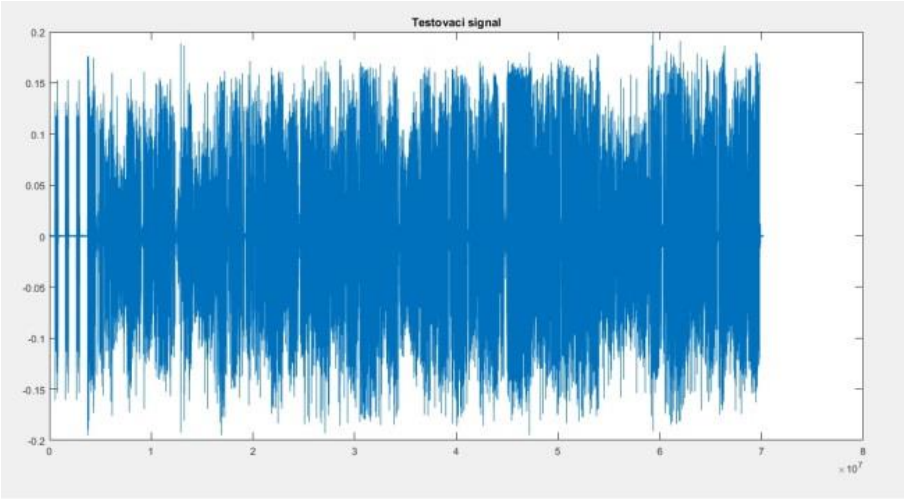


Obr. 3.1 Trénovací signál

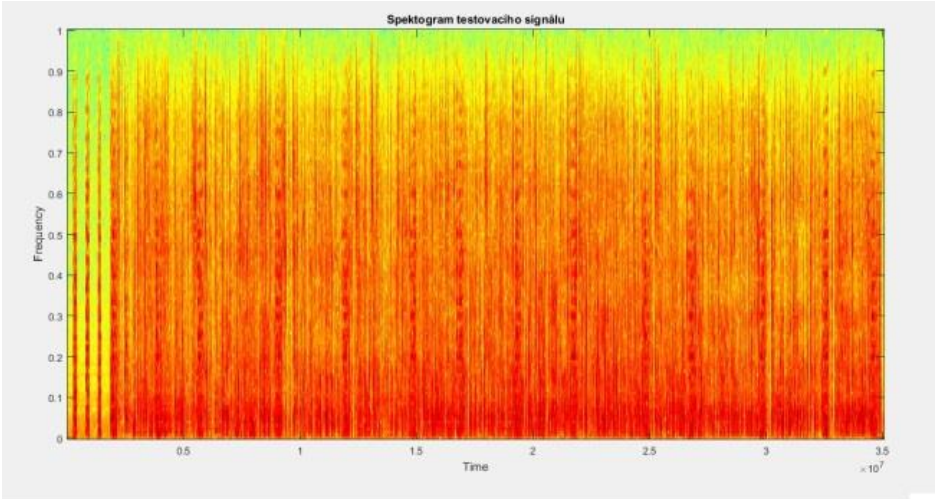


Obr. 3.2 Energie spektra trénovacího signálu

Testovací signál je rovněž nahrávka z jednání z Poslanecké sněmovny Parlamentu České republiky. Signál je dlouhý 1hodinu 13minut a 6sekund, obsahuje 17 hudebních vložek.



Obr. 3.3 Testovací signál



Obr. 3.4 Energie spektra testovacího signálu

3.2 Zpracování v časové oblasti

3.2.1 Metoda míry průchodu nulou

Při této mětodě se vycházelo ze základního vztahu

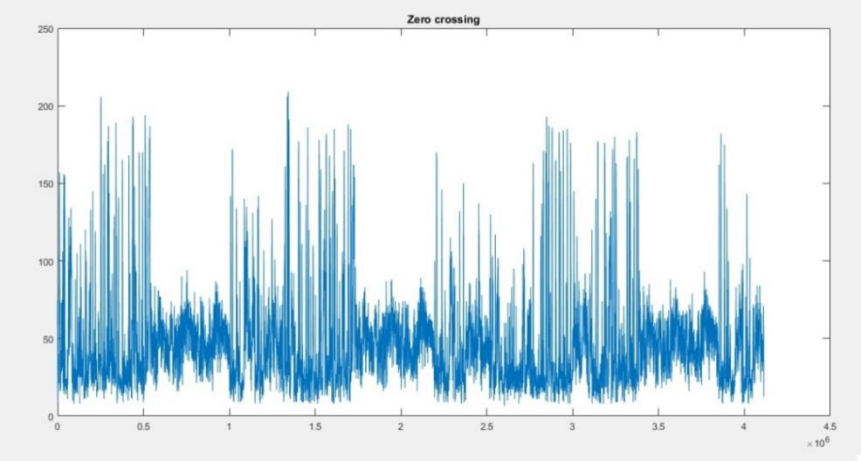
$$Z_n = \sum_{k=-\infty}^{\infty} \left| sgn\big(s(k)\big) - sgn\big(s(k-1)\big) \right| w(n-k) \qquad (3.2)$$

kde

$$sgn[s(k)] = \begin{cases} 1 & pro\ \ s(k) \geq 0 \\ -1 & pro\ s(k) < 0 \end{cases}$$

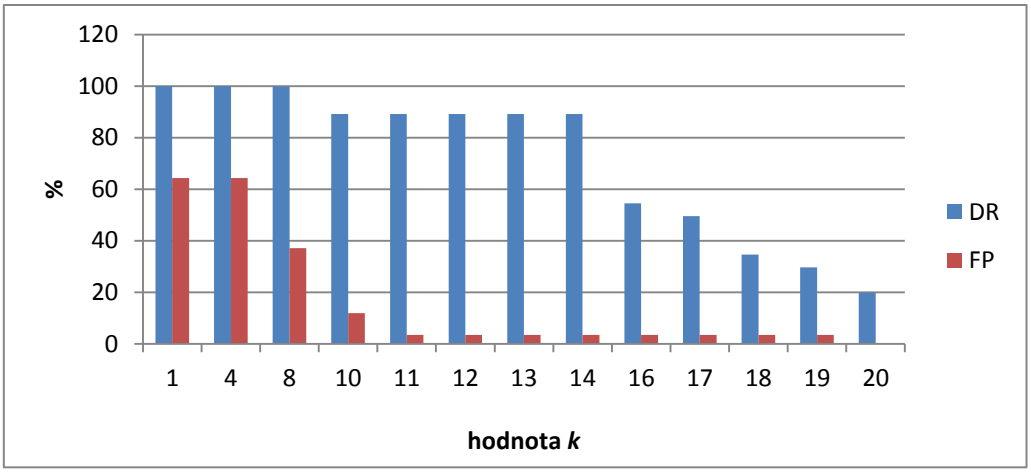
a *w(n)* je pravoúhlé okénko.

Předpoklad pro tuhle metodu je, že řeč je více rozdílná než hudba. Jak je vidět na následujícím grafu, hudba má opravdu jinou dynamiku.



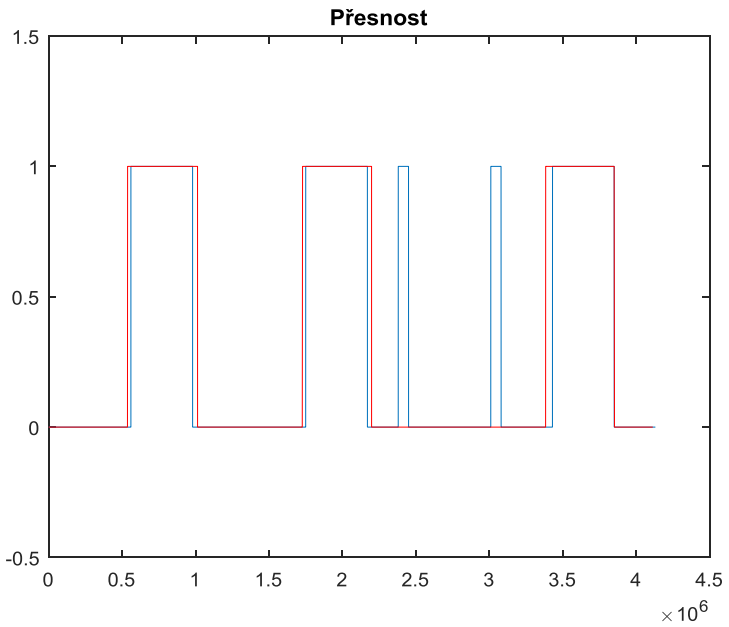
Obr. 3.5 Signál po průchodu nulou

Pro tenhle algoritmus se použilo několik druhů nastavení algoritmu, kdy se zkoušela nastavovat mez *k*, pod kterou neklesne dynamika hudby a hledalo se nejlepší možné řešení. (Obr. 3.6)



Obr. 3.6 Graf závislosti DP a FP na zvolenou mez

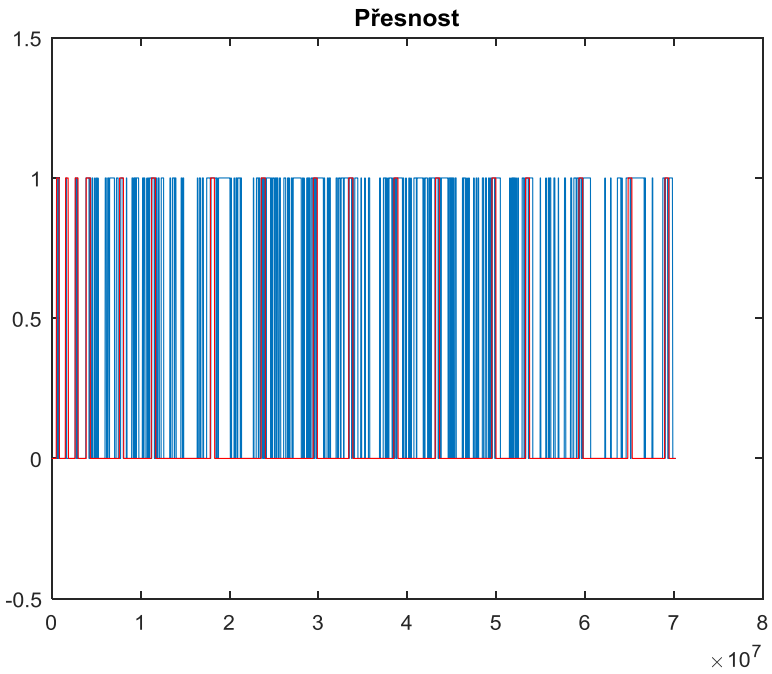
Jako nejlepší možné řešení v poměru co nejlepší hodnoty DR a co nejmenší hodnoty FP se zvolila mez *k* =14. Při tomto nastavení byla hodnota DR=89,10 a FP=3,40.



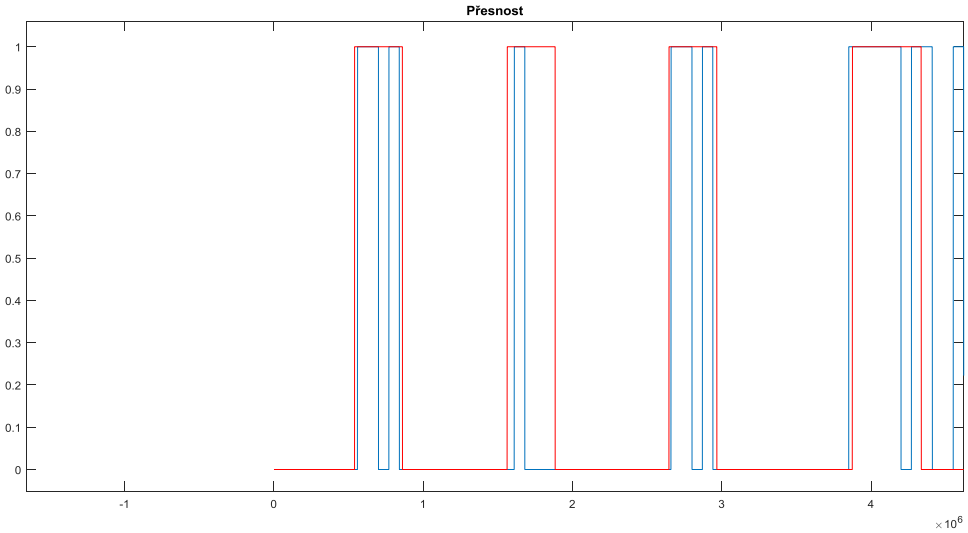
Obr. 3.8 Přesnost ZC na trénovacím signálu

Když se tenhle algoritmus s kritériem *k* = 14 pustil na testovací signál úspěšnost byla 69,57% s 37,04% falešných poplachů.

Na obrázku 3.9 lze vidět červeně hudbu v signálu a modře detekovanou hudbu algoritmu. Jelikož je zvukový signál dlouhý, na obrázku 3.10 je přiblíženo na začátek zvukového signálu, kde se nacházejí čtyři hudební znělky.

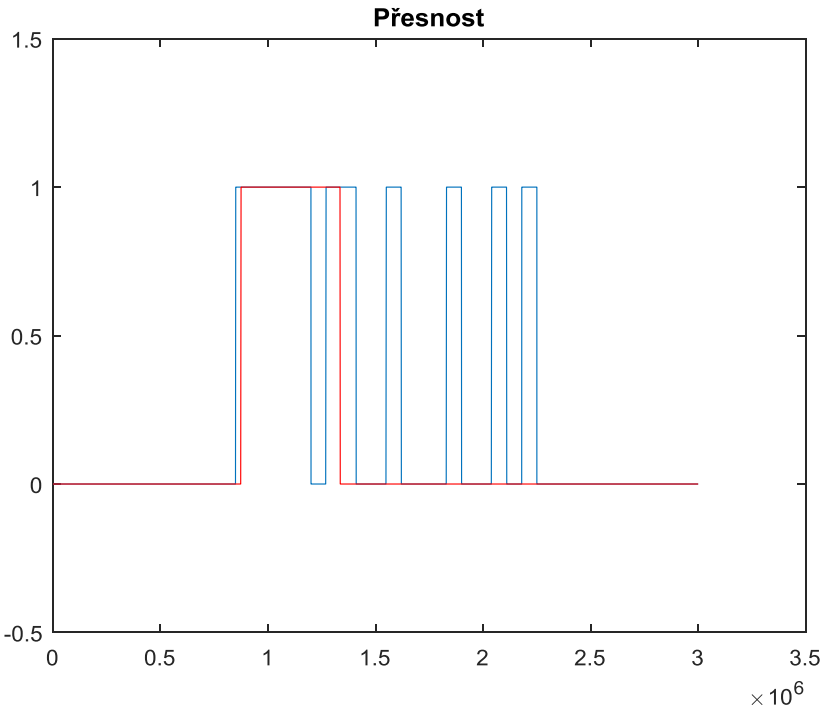


Obr. 3.9 Přesnost ZC na testovacím signálu



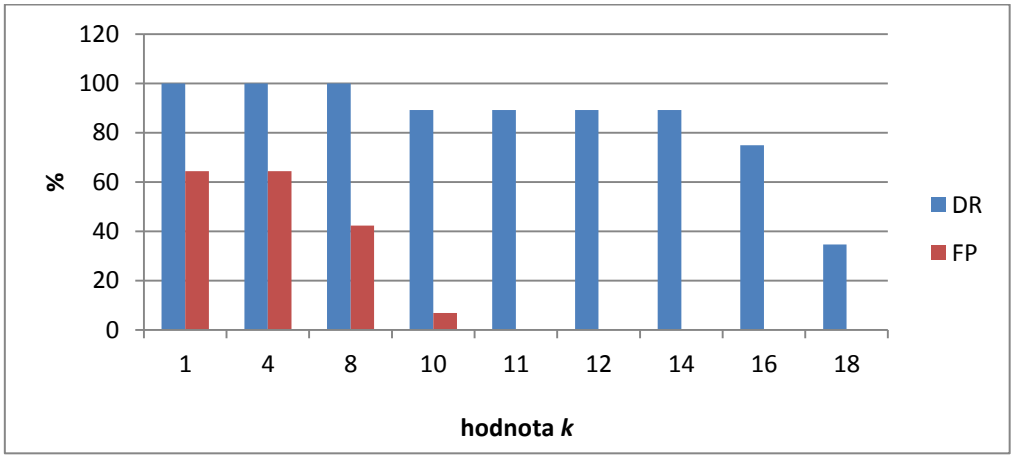
Obr. 3.10 Přesnost ZC na testovacím signálu

Ovšem jak lze vidět na obrázku 3.11 tahle metoda generuje spousta falešných poplachů. Proto by bylo dobré navrhnout jednoduchý okénkový filtr, který by filtroval rozpoznanou hudbu. Je jasné, že žádná hudební vložka nebude kratší než jedna vteřina. Proto jestli se v jedné vteřině zvukového signálu vyskytne méně zvukových vzorků než hudebních, lze o nich říci, že jsou to falešné poplachy.



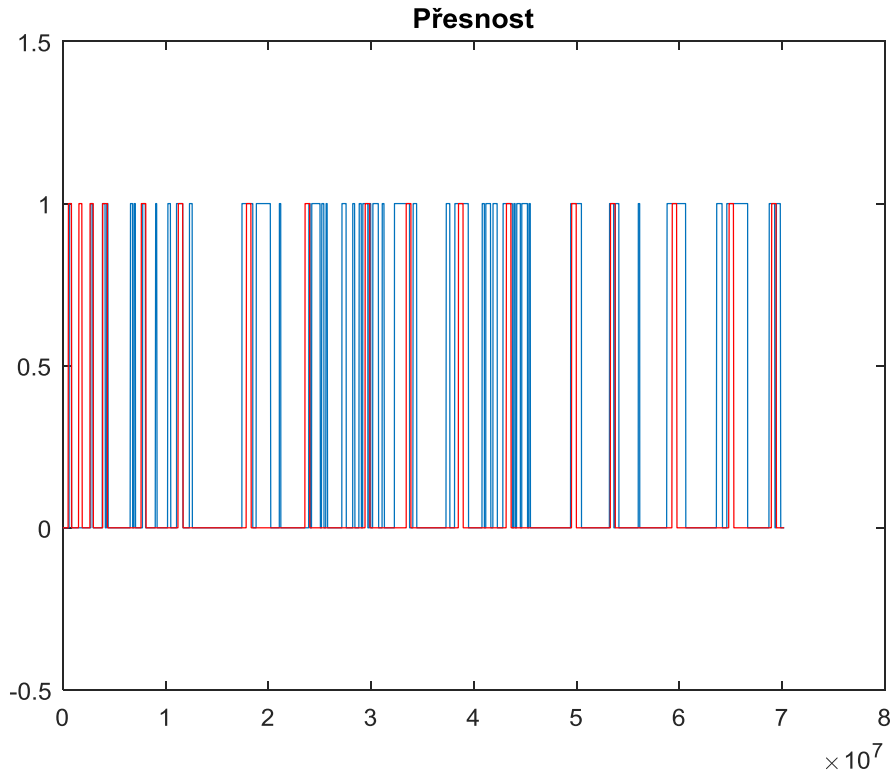
Obr. 3.11 přiblížení obrázku 3.10

Po zrealizování tohoto jednoduché filtru se zkoušel algoritmus na trénovací sadě dat. Po vyzkoušení různého nastavená konstanty rozhodování *k* se dospělo k názoru, že rozhodovací konstanta zůstane neměnná na hodnotě *k*=14, kde úspěšnost detekce byla stejná jako bez filtru, tj DP= 89.1, ovšem falešné poplachy zmizeli zcela úplně.



Obr. 3.12 Graf závislosti DP a FP na zvolenou mez

Při takovém nastavení algoritmu došlo na testovacích datech ke zvýšení úspěšnoti na hodnotu 75,82%, a hodnota falešných poplachů klesla o 13,23% na hodnotu 24,81%.



Obr. 3.13 Přesnost ZC na testovacím signálu

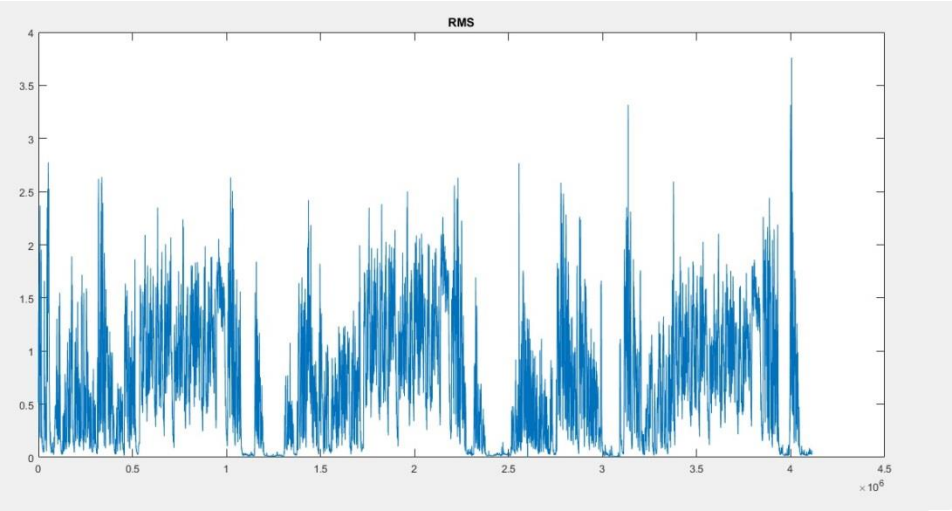
3.2.2 Metoda krátkodobé energie

Metoda krátkodobé energie je definována vztahem

$$E_n = \sum_{n=1}^{N} [s(k)w(n-k)]^2 \tag{3.1}$$

Kde *s(k)* je vzorek signálu v čase *k, w(n)* reprezentuje příslušný typ okénka.

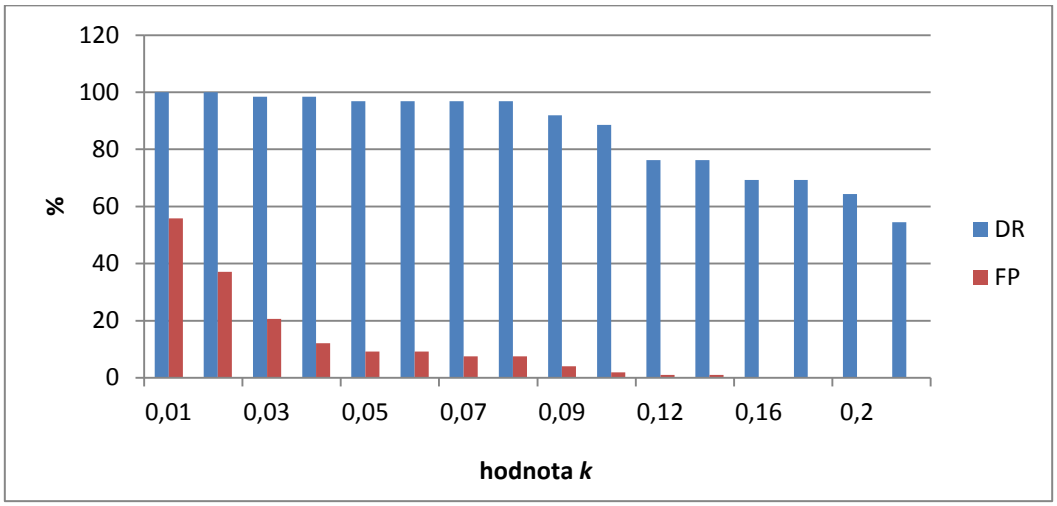
Zpracování signálu vypadalo následovně (obr. 3.13)



Obr. 3.13 RMS signálu

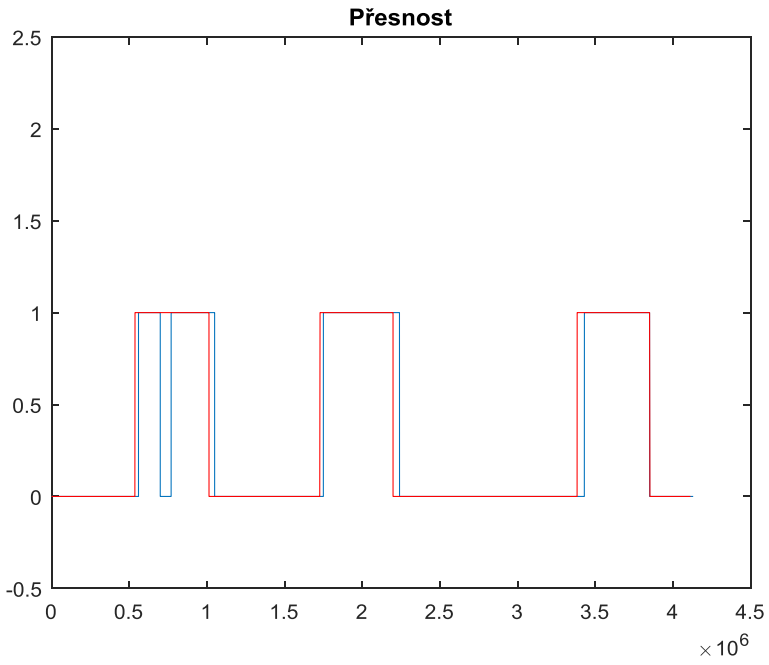
Jak lze vidět, hudba se dá z grafu zase krásně rozeznat.

Pro tohle zpracování se požilo několik druhů nastavení algoritmu, kdy se zkoušela nastavovat mez *k* energie, pod kterou neklesne hudba a hledalo se nejlepší možné řešení. (Obr. 3.14)



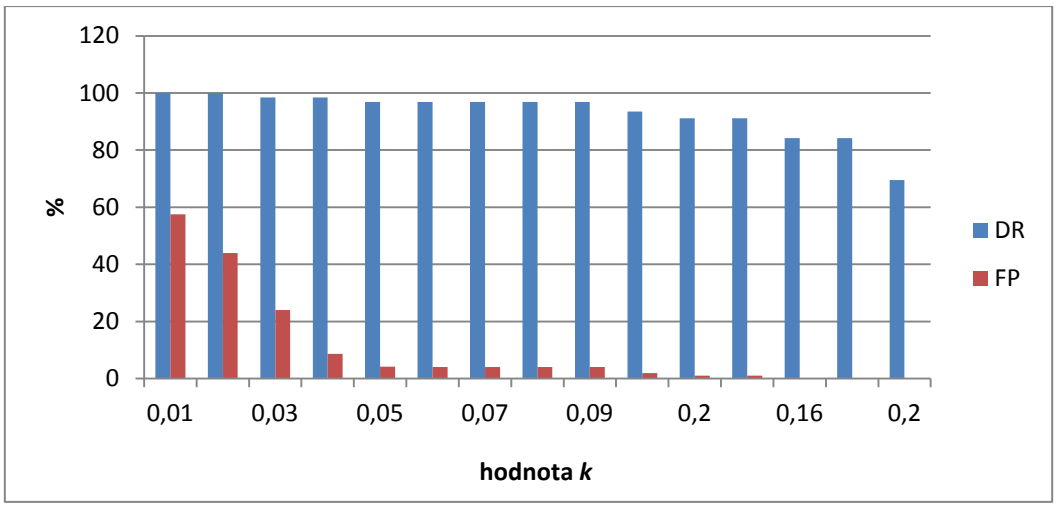
Obr. 3.14 Graf závislosti DP a FP na zvolenou mez

Jako nejlepší možné řešení v poměru co nejlepší hodnoty DR a co nejmenší hodnoty FP se zvolila mez *k* =0.1. Při tomto nastavení byla hodnota DR=88,53 a FP=1,89.

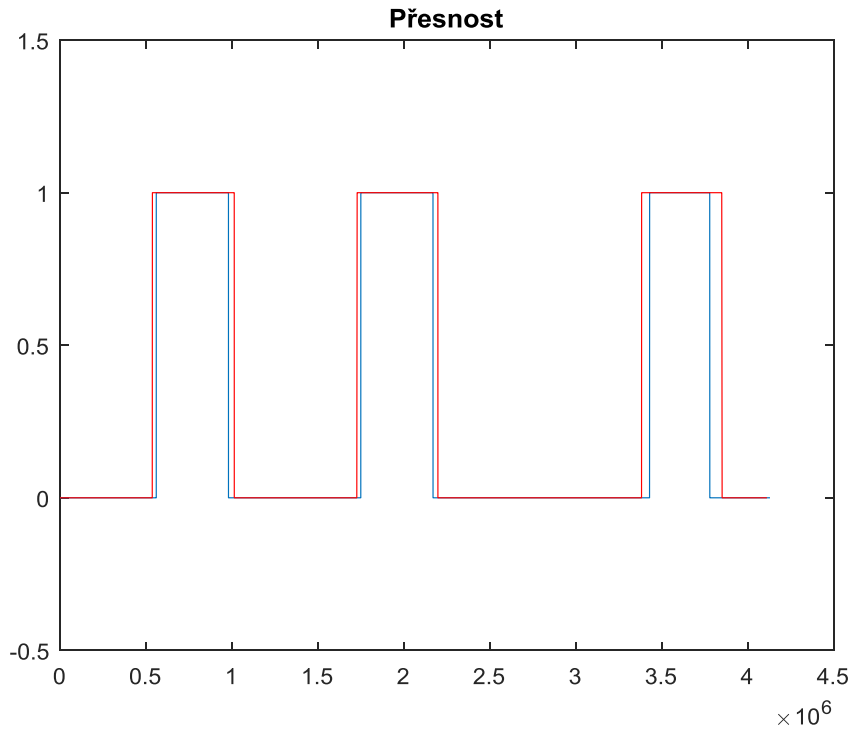


Obr. 3.15 Přesnost RMS na trénovacím signálu

Ovšem i na téhle metodě by bylo vhodné vyzkoušet, jestli se výsledky na trénovacích datech zlepší, pokud se použije stejný okénkový filtr jako v předešlém algoritmu. Výsledky lze vidět na obrázku 3.17. Falešné poplachy zcela zmizely na hodnotě k=0.16, přičemž hodnota úspěšnosti je na 84% (Obr. 3.16)

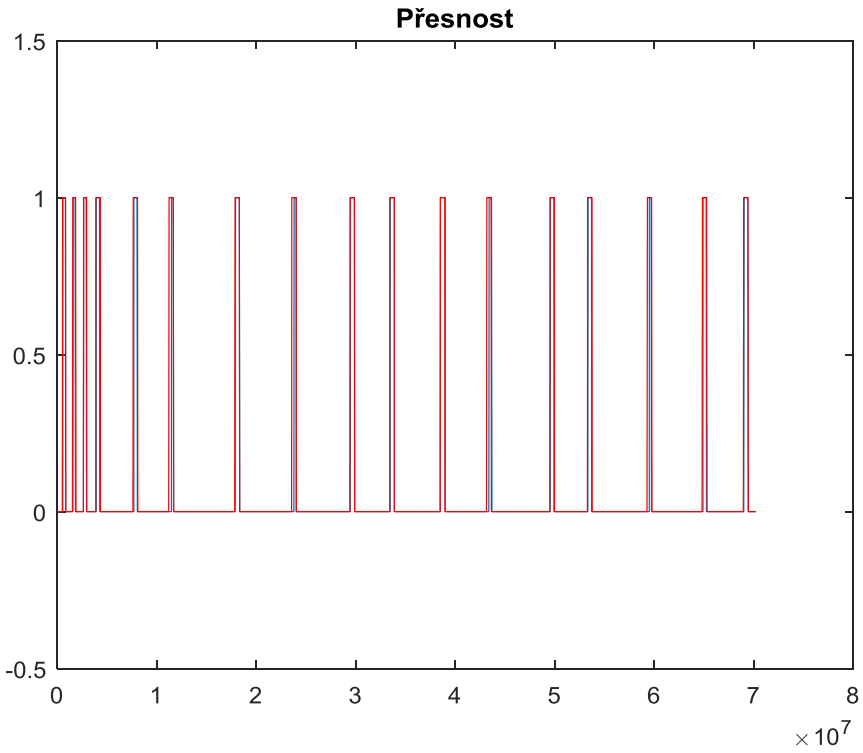


Obr. 3.16 Graf závislosti DP a FP na zvolenou mez

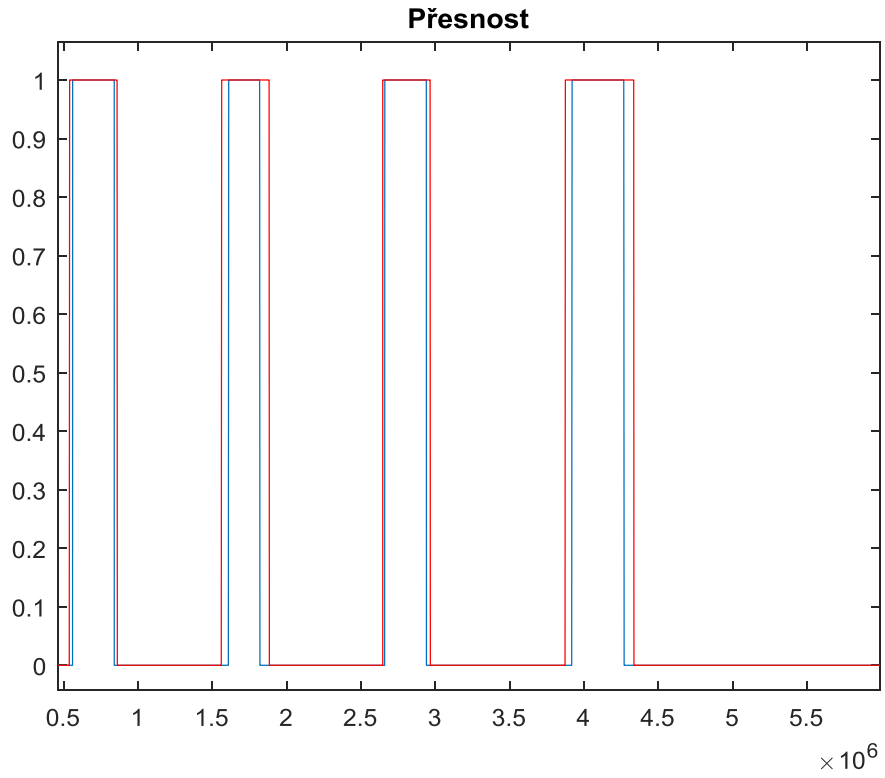


Obr. 3.17 Přesnost RMS na trénovacím signálu

Když se tenhle algoritmus s kritériem 0.16 pustil na testovací signál úspěšnost byla 74,03% s 0,16% falešných poplachů. Na obrázku 3.18 lze vidět červeně hudbu v signálu a modře detekovanou hudbu algoritmu. Jelikož je zvukový signál dlouhý, na obrázku 3.19 je přiblíženo na začátek zvukového signálu, kde se nacházejí čtyři hudební znělky.



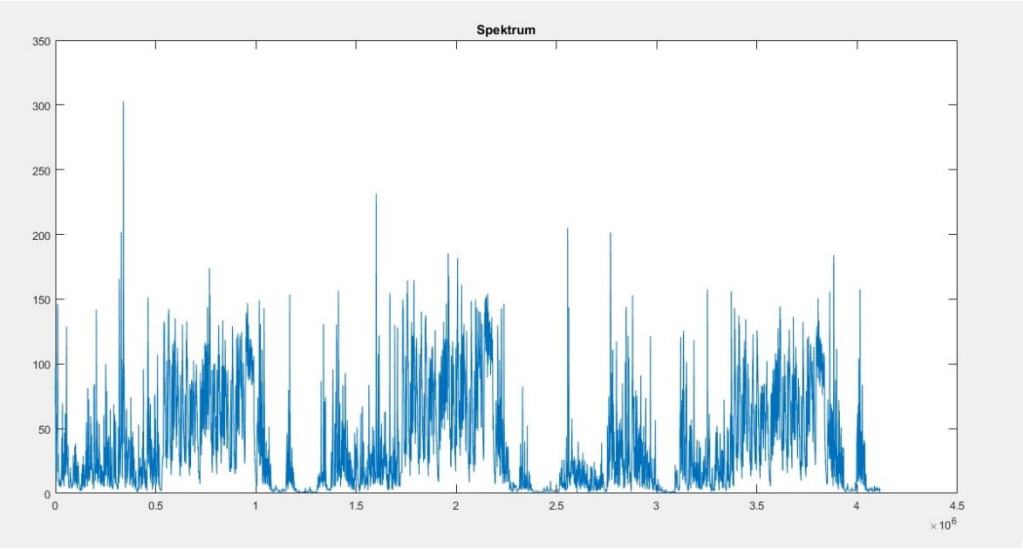
Obr. 3.18 Přesnost RMS na testovacím signálu



Obr. 3.19 Přesnost RMS na testovacím signálu

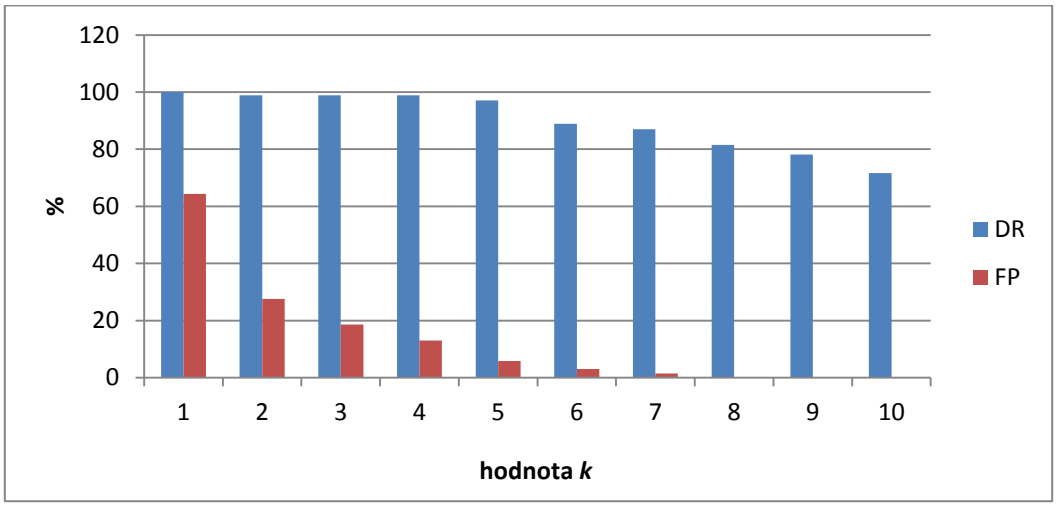
3.3 Zpracování ve frekvenční oblasti

Na zpracování signálu se použila rychlá Fourierova transformace, kde se vychází ze vztahu (2.12). Po zpracování signálu lze vidět, kde se v signálu nachází hudba. Z grafu lze vidět, že energie hudby je výrazně vyšší než energie řeči. Hudba dlouhodobě neklesne pod určitou hodnotu.



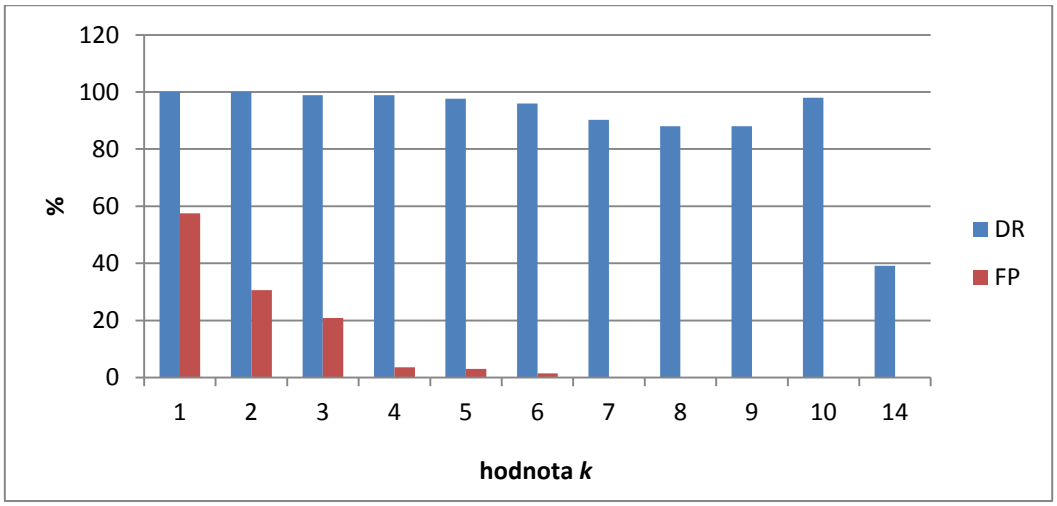
Obr. 3.20 Energie trénovacího signálu.

Pro tenhle algoritmus se použilo několik druhů nastavení algoritmu, kdy se zkoušela nastavovat mez *k*, pod kterou neklesne dynamika hudby a hledalo se nejlepší možné řešení. (Obr. 3.21) Jako nejlepší nastevní se jeví mez nastavena na *k*=7 s úspěšností 87,02% s 1,45% falešnými poplachy.

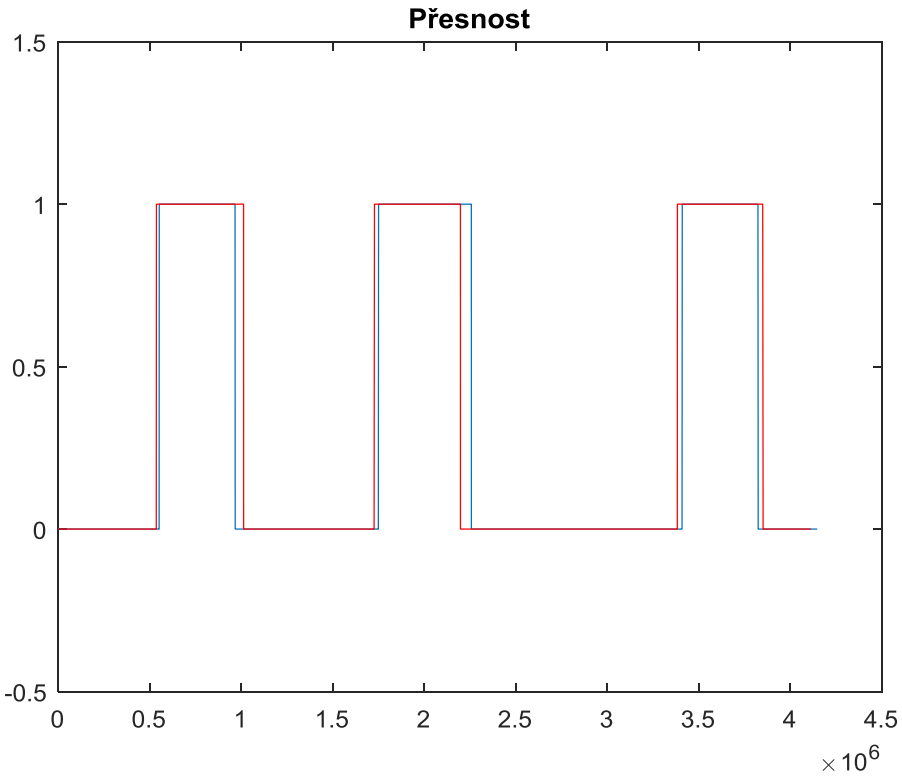


Obr. 3.21 Graf závislosti DP a FP na zvolenou mez

Ovšem i na téhle metodě by bylo vhodné vyzkoušet, jestli se výsledky na trénovacích datech zlepší, pokud se použije stejný okénkový filtr jako v předešlých algoritmech. Výsledky lze vidět na obrázku 3.22. Falešné poplachy zcela zmizely na hodnotě *k*=7, přičemž hodnota úspěšnosti je na 90,28% (Obr. 3.16)

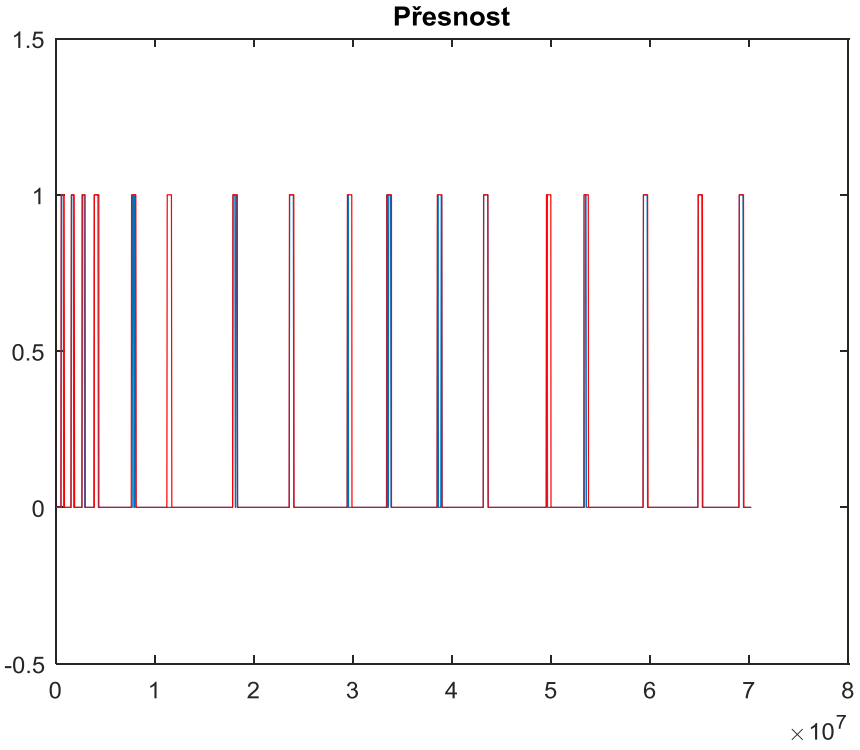


Obr. 3.22 Graf závislosti DP a FP na zvolenou mez



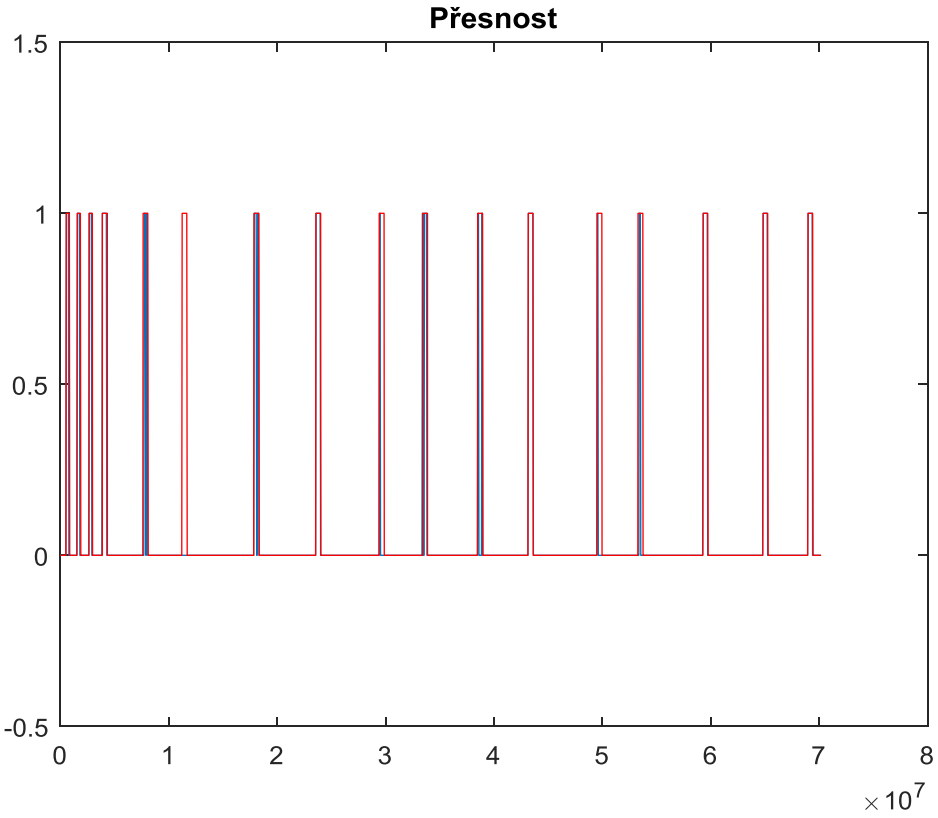
Obr. 3.23 Přesnost zpracování ve frekvenční oblasti na trénovacím signálu

Když se tenhle algoritmus s kritériem k = 7 pustil na testovací signál úspěšnost byla 62,05% s 0,08% falešných poplachů.



Obr. 3.24 Přesnost zpracování ve frekvenční oblasti na testovacím signálu

Na obrázku 3.24 lze vidět červeně hudbu v signálu a modře detekovanou hudbu algoritmu. Jelikož je zvukový signál dlouhý, na obrázku 3.25 je přiblíženo na začátek zvukového signálu, kde se nacházejí čtyři hudební znělky.



Obr. 3.25 Přesnost zpracování ve frekvenční oblasti na testovacím signálu

3.4 Parametrizace signálu

Signál se zparametrizoval pomocí programu ErisParam.exe, který zparametrizoval signál po 10ms na na parametry o dimenzi 13 pomocí metody MFCC (Kepstrální analýza řečového signálu).

Poté se signál zpracoval algoritmem K-means.

3.5 Zpracování K-means algoritmem

Jako první nápad se nabízelo, že zvukový signál by se dal rozdělit do 3 shluků:

- Hudba
- Řeč
- Ticho

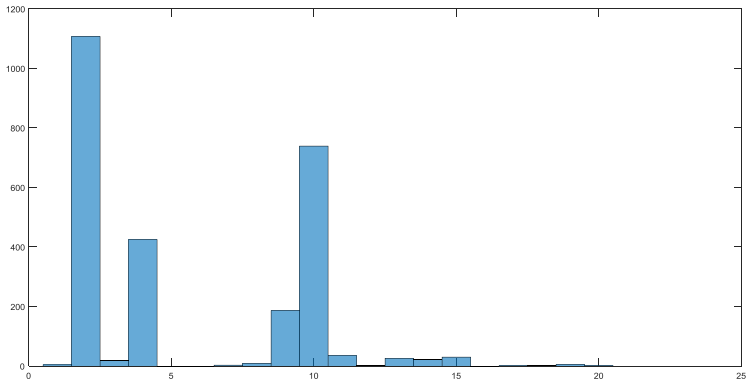
První pokusy nedopadly podle představ. Prvotní myšlenk byla, že algoritmus rozdělí parametry do 3 tříd (hudba, řeč a ticho). Algoritmus se trénoval bez učitele, avšak po natrénování se muselo ověřit do jaké třídy se natrénovaly parametry, které obsahovaly hudbu. Úspěšnost natrénování byla 3,46% a falešných poplachů bylo 5.83%.

Je velmi pravděpodobné, že řeč má mnohem vetší spektrum parametrů než hudba, protože pokaždé je jiný řečník nebo třeba jiná výška hlasu. Ovšem budeme muset využít trénování s učitelem.

Jako další pokus, bylo zkusit rozdělit parametry například na 9 shluků, z nichž 1 shluk byla hudba a zbytek řeč a ticho. Ovšem ani tohle není úplně ideální. I zde se po natrénování muselo ověřit, který shluk obsahoval hudbu.Trénovací signál se natrénoval s přesnosti 85,02% s 9,59% falešných poplachů

Z předešlých výsledků, lze vidět, že k-means by mohl fungovat, ovšem chce to ještě větší počet shluků. Některé parametry řeči si jsou hodně podobné s parametry, které obsahují hudbu.

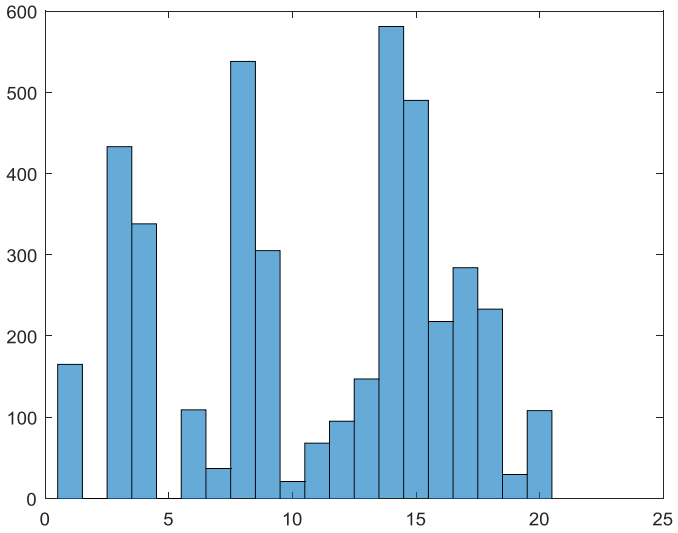
Je možné, že jedna třída na hudbu stačit nebude. Proto zkusíme algoritmus pustit na 20 shluků. Po natrénování se můžeme podívat kam se natrénovala hudba.



Obr 3.26 Histogram shluků hudby

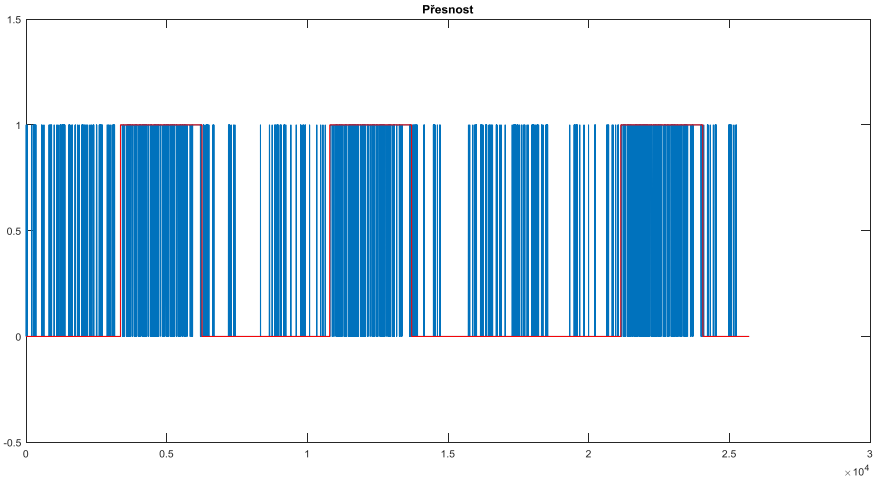
Z obrázku 3.26 lze vidět, že hudba se nejvíce nachází ve shlucích 2,4,9 a 10.

Na obrázku 3.27 můžeme vidět, kam se natrénovala řeč. Řeč se nachází skoro ve všech shlucích, kromě 2,4 a 10.



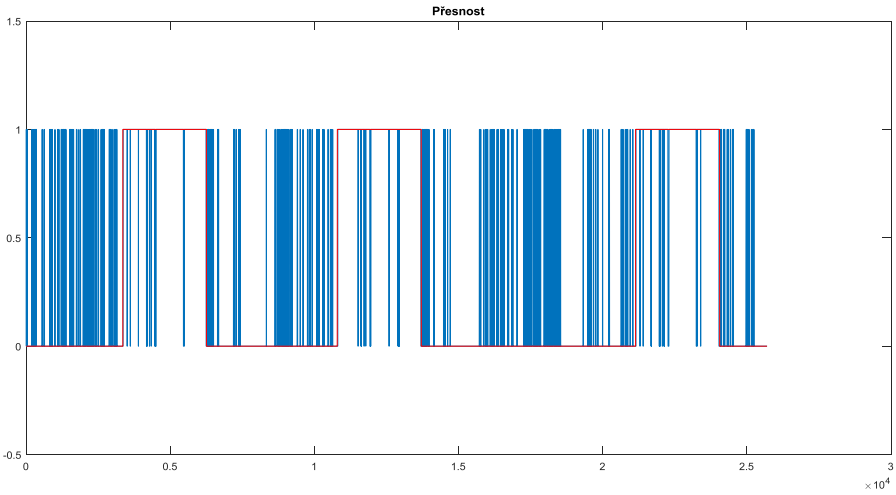
Obr. 3.27 Histogram shluků řeči

Algoritmus se pustil nejdříve s rozdělením hudby do shluků 2,4 a 10, kde se natrénovaly jednotlivé centroidy. Úspěšnost byla velice slibná přesněji 75,11% s 3,93% falešných poplachů.



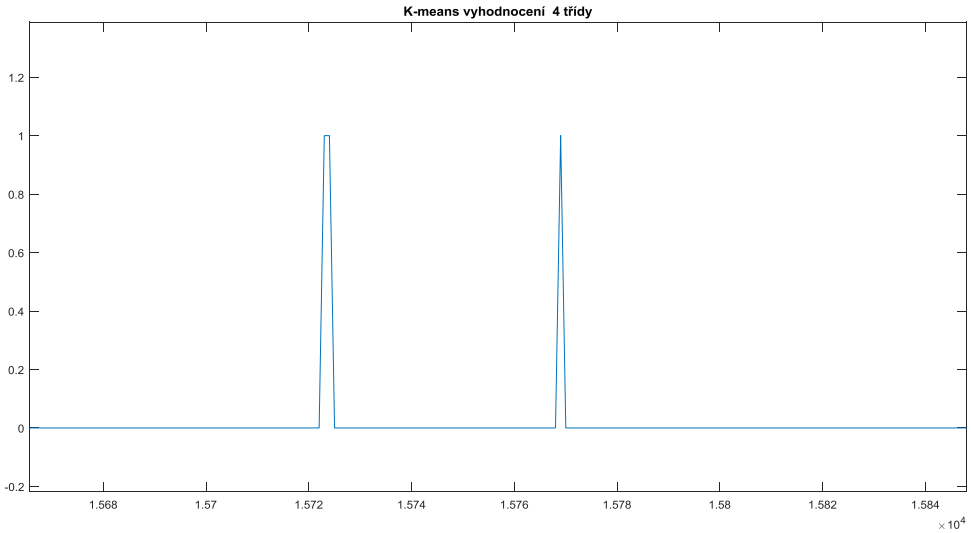
Obr. 3.28 K-means algoritmus pro 3 shluky hudby

Jako další krok se pustil tento algoritmus s rozdělením hudby do 4 shluků. Konkrétně do shluků 2,4,9 a 10. Úspěšnost byla velmi dobrá přesněji 96,72% s 6,41% falešných poplachů.



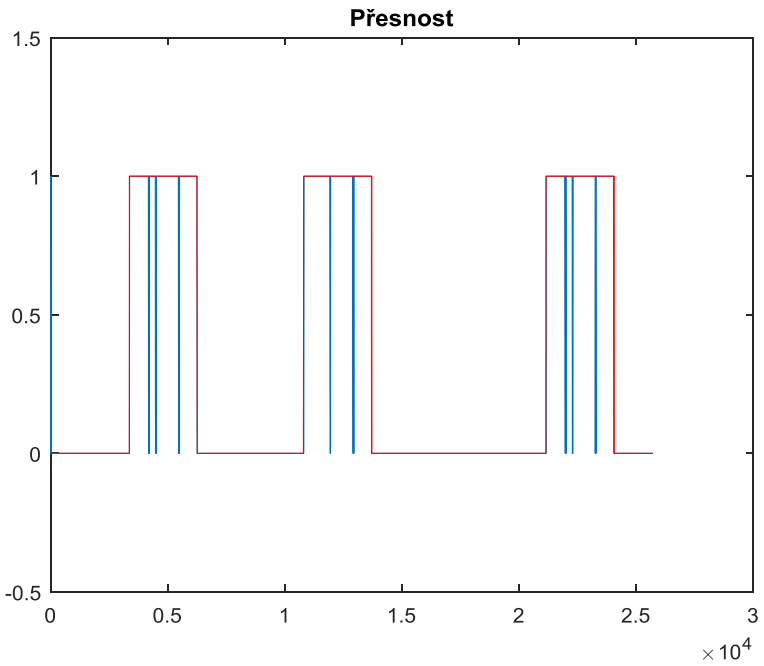
Obr. 3.29 K-means algoritmus pro 4 shluky hudby

Jak lze vidět na obrázku 3.29, falešných poplachů je tam stále dost. Jelikož falešné poplachy mohou být způsobeny i tím, že se jeden parametr, který je podobný hudbě, ale hudba to není, zařadí do shluku, který určuje hudbu. Na obrázku 3.30 lze takové zařazení vidět. I proto na zmenšeném grafu to vypadá, že algoritmus vyhodnotí mnohem více falešných poplachů. Z tohoto důvodu by bylo dobré zkusit udělat jednoduchý okýnkový filtr, který by nám přefiltroval parametry vyhodnocené jako hudba, ovšem v jeho okolí, například jedné vteřiny, žádné další hudební parametry nejsou. Tím by mohl poklesnout počet falešných poplachů.



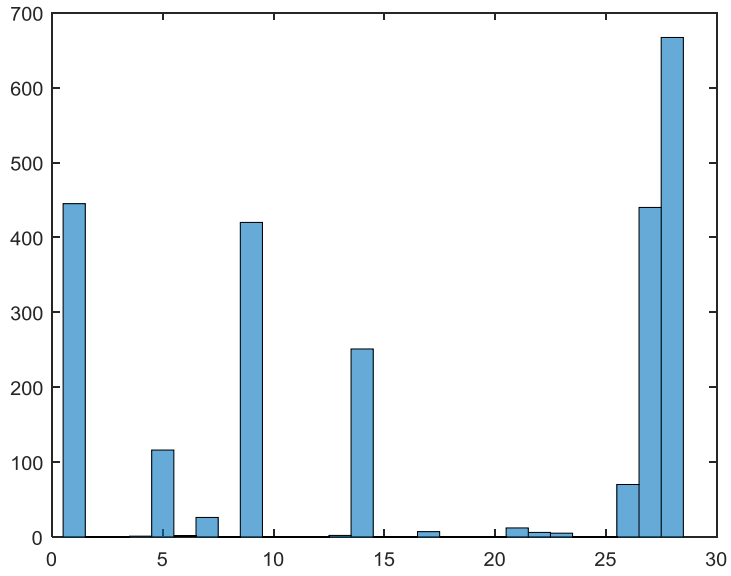
Obr. 3.30 Detail falešného poplachu

Po přidání takového jednoduchého filtru je úspěšnost 97,02% s 0,03% falešných poplachů. Na obrázku 3.31 lze vidět, že falešné poplchy nejsou skoro žádné.

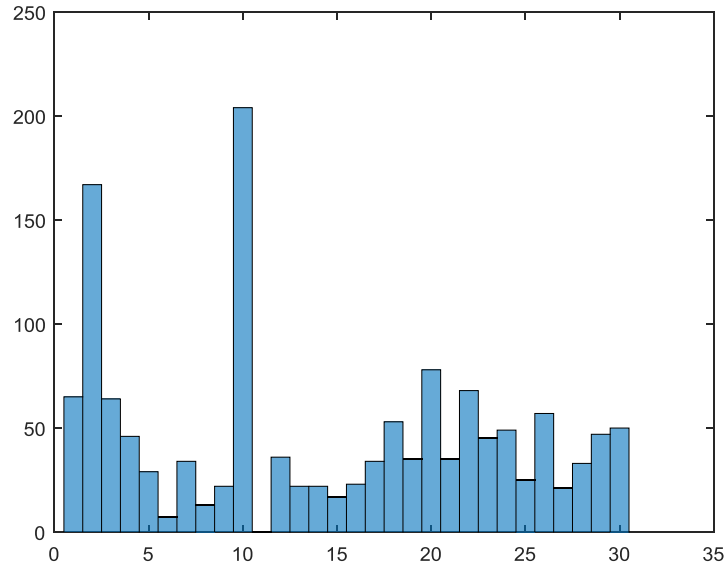


Obr. 3.31 K-means algoritmus pro 4 shluky hudby s okénkovým filtrem

I když tento algoritmus pracuje velmi dobře, je dobré vyzkoušet, co se stane, když zkusíme pustit algoritmus k-means s více shluky, například s 30. Jak lze vidět z obrázku 3.32a a z brázku 3.32.b, hudba už se natrénovává do mnoha shluků a začíná se zařazovat do stejných shluků jako je řeč. Proto se jako nejlepší verze zvolila metoda s 20 shluky, kde výsledky jsou více než uspokojivé.

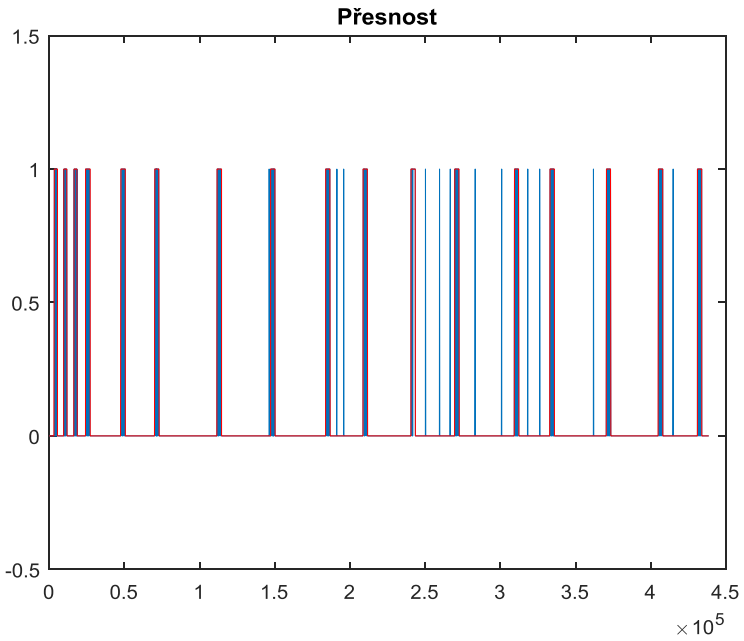


Obr. 3.32a K-means algoritmus pro 30 shluků-hudba



Obr. 3.32b K-means algoritmus pro 30 shluků – řeč

Jako nejlepší algoritmus na testovací sadu se vybral k-means, který natrénoval parametry, pomocí centroidů z trénvací sady do 20 shluků a z toho 4shluky byla hudba. Tyto výsledky se následně projeli jednoduchým okýnkovým filtrem. Úspěšnost na testovací sadě je velmi dobrá přesněji 91,72% s 0,05% falešných poplachů.



Obr. 3.32K-means algoritmus pro 4 shluky hudby s okénkovým filtrem

testovací sada

3.6 Vyhodnocení jednotlivých metod

Na základě předešlého testování lze zhodnotit jednotlivé metody rozpoznávání hudby. U každé metody se zkusily různá nastavení a to nastavení, které mělo nejlepší výsledky na trénovacím souboru se použilo na testovací.

Na trénovacím souboru nejlépe pracovaly metody K-means s 20 shluky a 4třídy, do kterých se zařazovala hudba a metoda RMS. Ostatní metody taky dokázaly rozpoznat hudbu velmi dobře.

Na testovacím souboru si nejlépe vedla metoa založena na shlukování pomocí algoritmu k-means, která rozdělovala parametry do 20 shluků. Dosáhla Úpěšnosti 91,72% s velmi malým počtem falešných poplachů. Naopak metoda založena na detekci počtu průchodu sigálu nulou zaznamenala velmi vysoký počet falešných poplachů.

Metoda	Trénovací sada		Testovací sada	
	DR	FP	DR	FP
RMS	84,0	0	74,03	0,16
ZC	89,1	0	75,82	24,81
Spektrální analýza	90,28	0	62,05	0,08
K-means	97,02	0,03	91,73	0,05

Jako nejlepší algoritmus na detekci huby se jeví metoda k-means, která využívá 20 shluků a hudbu zařazuje do 4 tříd. Jedna třída pak je ticho a zbylých 15 tříd je řeč.

4 Závěr

Úkolem této bakalářské práce bylo prostudovat základní metody detekce hudby v řečovém signálu (hudba/řeč). V teoretickém úvodu se práce zaměřuje na základní metody zpracování signálu v časové a frekvenční oblasti. V oblasti časového zpracování byla detailně analyzována metoda průchodu nulou a metoda krátkodobé energie. V oblasti frekvenčního zpracování byl popsán převod signálu z času do spektra metodou DFT. Dále byla popsána metoda MFCC, která je hojně využívaná při zpracování řečového signálu pro účely rozpoznávání. Na závěr teoretického úvodu byla popsána teorie slukování s důrazem na unsupervised slukování metodou K-means.

Všechny teoreticky popsané metody bylo třeba optimálně nastavit, což bylo provedeno na trénovacím souboru. Ten obsahoval pouze tři hudební ložky a měl délku necelých 5 minut. Ovšem i na trénovacích datech bylo zjištěno, že analyzované algoritmy generují nezanedbatelné množství falešných poplachů. Velké množství těhto poplachů bylo způsobeno tím, že i velmi krátký vzorek, který může mít podobné vlastnosti jako hudba, například nějaké zvolání v davu, bylo rozpoznáno jako hudba. Proto byl navrhnut jednoduchý filtr, který když našel takto ojedinělý vzorek označený jako hudba a v okolí jedné vteřiny nebyly žádné další hudební vzorky, tak jej označil jako řeč.

Nejnadějnějšími metodami se jevily metoda založená na zkoumání krátkodobé energie a metoda shlukování vektorů parametrů MFCC algoritmem k-means. Při spuštění algoritmu na testovací soubor, všechny metody pracovaly velmi slibně, až na metodu založenou na počtu průchudou nulou (ZC). Ta měla velmi vysoký výskyt falešných poplachů. Počet falešných poplachů dosahoval hranice až 25%. Metody založené na krátkodobé energii a naspektru dosahovaly hodnot falešných poplachů kolem 0,1%. K této hodnotě se dopracovala i metoda založená na shlukování algoritmem k-means. Ovšem detekční schopnost (DR) k-means algoritmu dosahovala až k hranici 91% (úspěšnost detekce hudby) přičemž ostatní algoritmy se pohybovaly pouze kolem 70%.

Literatura

[1] Psutka, J. and Müller, L. and Matoušek, J. and Radová, V.: Mluvíme s počítačem česky, Academia, Praha, 2006.

[2] Wu Chou, Liang Gu : Robust Singing Detection in Speech/Music Discriminator Design,

[3] Sonnleitner R., Niedermayer B., Widmer G., Schlüter J.: A Simple and effective spectral feature for speech detection in mixed audio signals, 2012