VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ

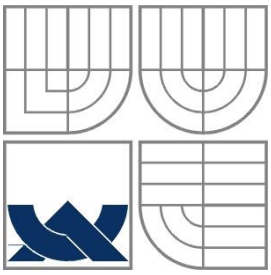
FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

VÝPOČET TEMPA ŘEČI

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

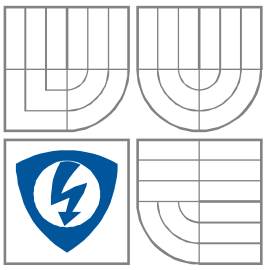
AUTOR PRÁCE ZOLTÁN GALÁŽ
AUTHOR

BRNO 2011



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

DEPARTMENT OF TELECOMMUNICATION

VÝPOČET TEMPA ŘEČI

CALCULATION OF SPEECH RATE

BAKALÁŘSKÁ PRÁCE

SEMESTRAL THESIS

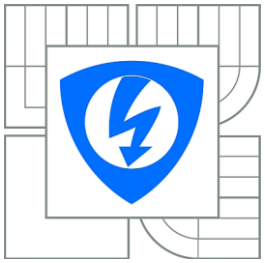
AUTOR PRÁCE — ZOLTÁN GALÁŽ

AUTHOR

VEDOUCÍ PRÁCE — Ing. JIŘÍ MEKYSKA

SUPERVISOR

BRNO, 2011



VYSOKÉ UCENÍ TECHNICKÉ V BRNE

Fakulta elektrotechniky a komunikačních technologií

Ústav telekomunikací

Bakalářská práce

bakalářský studijní obor
Teleinformatika

Student: Zoltán Galáž ***ID:*** 121019
Ročník: 3 ***Akademický rok:*** 2010/2011

NÁZEV TÉMATU:

Výpočet tempa řeči

POKYNY PRO VYPRACOVÁNÍ:

Cílem bakalářské práce je rozbor aktuální problematiky stanovení tempa řeči a následný návrh a implementace systému, který by tempo vypočítal. Systém bude využívat segmentální příznaky LPC, MFCC, PLP a k detekci bude použita vektorová kvantizace založena na algoritmu K-means.

DOPORUCENÁ LITERATURA:

[1] PSUTKA, Josef, et al. Mluvíme s pocítacem cesky. Praha: Academia, 2006. 752 s. ISBN 80-200-1309-1.
[2] HUANG, Xuedong; ACERO, Alex; HON, Hsiao-Wuen. Spoken Language Processing a Guide to Theory, Algorithm and System Development. Upper Saddle River: Prentice Hall PTR, 2001. 980 s. ISBN 0-13-022616-5.
[3] SMÉKAL, Zdenek. Císlicové zpracování signálu. Skripta FEKT VUT vBrne, 2010.

Termín zadání: 7.2.2011 ***Termín odevzdání:*** 2.6.2011

Vedoucí práce: Ing. Jiří Mekyska.

prof. Ing. Kamil Vrba, CSc.
Předseda oborové rady

UPOZORNENÍ:

ABSTRAKT

V rámci bakalárskej práce boli naštudované metódy spracovania reči za účelom výpočtu tempa rečového signálu. Ďalej bola na základe získaných poznatkov zostavená štruktúra programu. Program bol vytvorený v prostredí MATLAB a kvôli zjednodušeniu práce s programom bol implementovaný do grafického užívateľského rozhrania. Samotný výpočet tempa reči prebieha v niekoľkých fázach. V prvej fáze je rečový signál spracovaný v bloku predspracovania signálu a následne segmentovaný na rámce dĺžky 25 ms. Z každého rámca sa pomocou metódy krátkodobej analýzy pre zvolené parametre vypočíta tzv. matica príznakov. Z matice príznakov sa ďalej v bloku vektorovej kvantizácie priraďujú rámce jednotlivým centridom. Zmena centroidu vyjadruje zmenu hlásky. Tým sa zistí počet hlások v danom rečovom signály a následne sa z tohto počtu a známej dĺžky trvania nahrávky určí samotné tempo reči. V programe bol implementovaný algoritmus Fuzzy K-means a na ošetrenie testovania iba aktívnej reči bol do programu zaradený tzv. detektor rečovej aktivity. K programu bol vytvorený stručný návod k obsluhe a v práci boli rozobraté výsledky testovania algoritmu Fuzzy K-means ako aj celkového testovania programu Hromadné testovanie, ktorý testoval všetky nahrávky databáze TIMIT. Výsledky testov boli prehľadne spracované a vynesené do grafov daných závislostí.

KLÚČOVÉ SLOVÁ

Rečový signál, tempo, predspracovanie, segmentácia, krátkodobá analýza, príznaky, vektorová kvantizácia, centroid, K-means, Fuzzy K-means, detektor rečovej aktivity.

ABSTRACT

The submitted bachelor work deals with some methods of speech processing. The purpose of the work is to calculate the rate of speech signal. Also, based on the obtained knowledge, there was designed structure of the program. The program was created in MATLAB development environment and in order to make the program user-friendly, it was implemented into the graphic user interface. The speech rate calculation involves several steps. First, the speech signal is processed in a pre-processing block followed by its segmentation in fractions as long as 25 ms. Then, by means of the short term analysis method, the so called feature matrix is calculated. In next step these features are assigned to the calculated centroids in block of vector quantization. The change of the centroid reflects the change of phoneme. This is to show the number of the phonemes in the particular speech signal. The speech rate itself is based on this number devided by the known length of input speech signal. There, in created program, was implemented the Fuzzy K-means algorithm and for testing only the active speech, there was also the voice activity detector included. The brief program manual was also attached and there was discussed results of testing the Fuzzy K-means algorithm and results of the final testing of the mass testing program, that tested the whole TIMIT database. The testing results were clearly processed and got into graphs.

KEYWORDS

Speech signal, rate, pre-processing, segmentation, short-time analysis, features, vector quantization, centroid, K-means, Fuzzy K-means, voice activity detector.

GALÁŽ, Z. *Výpočet tempa reči*. Brno: Vysoké učení technické v Brně. Fakulta elektrotechniky a komunikačních technologií, 2011. 24 s. Vedoucí práce Ing. Jiří Mekyska.

Prehlásenie

Prehlasujem, že svoju bakalársku prácu na téma Výpočet tempa reči som vypracoval samostatne pod vedením vedúceho bakalárskej práce a s použitím odbornej literatúry a ďalších informačných zdrojov, ktoré sú všetky citované v práci a uvedené v zozname literatúry na konci práce.

Ako autor uvedenej semestrálnej práce ďalej prehlasujem, že v súvislosti s vytvorením tejto práce som neporušil autorské práva tretích osôb, predovšetkým som nedovoleným spôsobom nezasiahol do cudzích autorských práv osobných a som si plne vedomí následkov porušenia ustanovenia § 11 a nasledujúcich autorského zákona č. 121/2000 Sb., vrátane možných trestnoprávnych dôsledkov vyplývajúcich z ustanovenia§ 152 trestného zákona č. 140/1961 Sb.

V Brne dňa ....

Podpis autora

Poďakovanie

Rád by som poďakoval vedúcemu mojej semestrálnej práce, pánovi inžinierovi Jiřímu Mekyskovi, za veľmi kvalitné vedenie práce, potrebnú pomoc a poskytnutie dôležitých materiálov potrebných pre realizáciu tejto bakalárskej práce.

V Brne dňa .....

.....

Podpis autora

Obsah

Úvod ... 8
1 Rečový signál ... 9
1.1 Vytváranie reči ... 9
1.1.1 Dýchacie ústrojenstvo ... 10
1.1.2 Hlasové ústrojenstvo ... 10
1.1.3 Artikulačné ústrojenstvo ... 10
1.2 Rečové jednotky ... 11
1.2.1 Akustická úroveň ... 11
1.2.2 Fonetická úroveň ... 14
1.2.3 Fonologická úroveň ... 15
1.3 Suprasegmentálne rysy (prozódia) ... 15
1.3.1 Tempo reči ... 16
2 Analýza reči ... 16
2.1 Predspracovanie rečového signálu ... 16
2.1.1 Preemfáza (zdôraznenie vyšších frekvencií) ... 17
2.1.2 Ustredenie (odstránenie jednosmernej zložky) ... 18
2.2 Homomorfná transformácia rečového signálu ... 20
2.2.1 Realizácia kepstrálnej transformácie ... 21
2.2.2 Reálne kepstrum ... 22
2.2.3 Mel-frekvenčné kepsrálne koeficienty (MFCC) ... 23
2.3 Lineárna prediktívna analýza ... 24
2.3.1 Dopredná lineárna prediktívna analýza ... 24
2.3.2 LPC-cepstrum ... 26
2.3.3 Perceptívne lineárne predikčné koeficienty ... 27
2.4 Vektorová kvantizácia ... 29
2.4.1 K-means algoritmus ... 30
2.4.2 Fuzzy k-means algoritmus ... 30
3 Návrh systému ... 31
3.1 Grafický návrh ... 31
3.2 Popis systému ... 31
4 Praktická realizácia ... 33
4.1 Popis užívateľského rozhrania ... 33
4.2 Hromadné testovanie ... 35
5 Výsledky práce ... 36
5.1 Vytvorené skripty a funkcie ... 36
5.2 Výsledky hromadného testovania ... 38
5.2.1 Testovanie počtu centroidov algoritmu K-means ... 38
5.2.2 Testovanie algoritmu Fuzzy K-means ... 40
5.2.2 Priebeh testovania ... 41
5.3 Zhrnutie dosiahnutých výsledkov ... 44
5.4 Spustenie testovania ... 45
6 Záver ... 46
Literatúra ... 47
Zoznam použitých skratiek a symbolov ... 48
Obsah priloženého DVD ... 51

Úvod

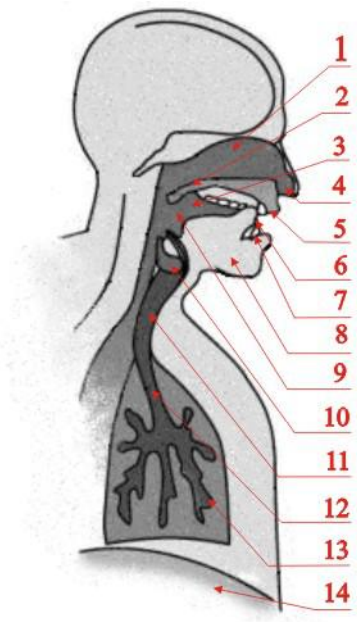
V tejto práci je snaha navrhnúť systém výpočtu tempa reči, otestovať príznaky krátkodobej analýzy reči ako MFCC, PLP atď. a ich následné spracovanie vo forme vektorovej kvantizácie. Vstupný rečový signál najprv prejde VAD systémom, ktorý z neho odstráni nežiaduce pauzy. Ďalej bude spracovávaný v blokoch predspracovania signálu, tj. ustredenie, preemfáza a následne bude segmentovaný na menšie časti o veľkosti 20-30 ms. Toto rozsegmentovanie signálu je potrebné pre nestacionaritu, ktorú má rečový signál. Predpokladáme totiž, že na dostatočne malom úseku bude signál približne stacionárny a bude naňho možné použiť niektorú z metód krátkodobej analýzy. Cieľom krátkodobej analýzy bude extrakcia príznakov. Podľa použitej metódy sa bude jednať jednak o mel-frekvenčné kepstrálne koeficienty (kepstrálna transformácia) alebo o príznaky lineárnej prediktívnej analýzy ako LPC, LPCC, PLP. Vypočítané príznaky budú ďalej podrobené vektorovej kvantizácii. Úlohou vektorovej kvantizácie bude vytvoriť kódovú knihu, podľa ktorej budú vektory príznakov nahradzované centroidmi. Centroid je vektor s najmenšou geometrickou vzdialenosťou od ostatných vektorov danom zhluku vektorov, ktorý vznikne pri vytváraní kódovej knihy. Vektorová kvantizácia totiž predpokladá, že jeden foném (hláska) odpovedá vždy vektorom príznakov v jednom zhluku, teda je možné tento zhluk nahradiť jedným vektorom, a to práve takzvaným centroidom. Výsledná postupnosť centroidov bude ďalej systémom analyzovaná a jednotlivé centroidy budú porovnávane a ak nastane nerovnosť, systém detekuje zmenu centroidu a tým aj zmenu hlásky. Na realizáciu vektorovej kvantizácie bude použitý algoritmus k-means. Pri vektorovej kvantizácii je potrebné určiť počiatočné polohy centroidov. Túto úlohu bude systém riešit postupným zväčšovaním počtu centroidov. Ďalej bude testovaný algoritmus fuzzy k-means, na zlepšenie detekcie hraníc hlások, teda na minimalizovanie chyby vzniknutej v dôsledku plozív. Plozivy sú hlásky, ktoré sa skladajú ako keby z dvoch úsekov, z úseku šumu a oblasti zákmitu. V práci bude teda snaha dosiahnuť deteciu plozív ako jednej hlásky. Po detekcii hraníc hlások systém vypočíta výsledné tempo reči.

1 Rečový signál

Reč ako taká je jedným z najstarších komunikačných prostriedkov na našej planéte. Toto dorozumievanie sa pomocou premyslených slov a fráz tvoriacich jeden komplexný celok je vlastná iba ľudským bytostiam. Pre reč na rozdiel od sluchu, ktorý je pre dorozumievanie sa pomocou akustického signálu akým reč je tiež nevyhnutná, sa ale nevyvinul zvláštny separovaný orgán. Vytvára sa v takzvanom rečovom ústrojenstve (Vocal Tract) ktoré sa nachádza v našom tele a zahŕňa dýchacie orgány ako aj ústnu dutinu plus jej časti.

1.1 Vytváranie reči

Reč je akustický signál, ktorý sa šíri hmotným (elastickým) prostredím a je vytváraný v rečových orgánoch. Tie sa skladajú z dýchacieho, hlasového a artikulačného ústrojenstva a dohromady tvoria tzv. vokálny (hlasový) trakt (viď Obr1. 1).



Obr1. 1: Vokálny (hlasový) trakt. 1. Nosová dutina, 2. Mäkké podnebie, 3. Ústna dutina, 4. Nozdry, 5. Pery, 6. Jazyk, 7. Zuby. 8. Čeľusť, 9. Hltan, 10. Hrtan, 11. Ezofágus, 12. Priedušnica. 13. Pľúca, 14. Bránica

1.1.1 Dýchacie ústrojenstvo

Je umiestnené v hrudnom koši a je niečo ako zdroj energie pre rečové vyjadrovanie. Pri nadýchnutí sa do pľúc dostane vzduch. Pri výdychu zatlačí bránica na pľúca a vydychovaný vzduch putuje dýchacím ústrojenstvom cez hlasové a artikulačné ústrojenstvo.

1.1.2 Hlasové ústrojenstvo

Je časť rečového ústrojenstva v ktorom vzniká hlas, ktorý je ďalej domodelovaný do podoby výslednej reči. Je uložené v hrtane, ktorý je spolu s pľúcami spojený bránicou. Samotný hlas vzniká v hlasivkách uložených v hrtane tak, že vydychovaný prúď vzduchu smeruje k hrtanu, resp. hlasivkám, kde sa nachádza hlasivková štrbina, ktorá je počas mlčania otvorená (kľudové postavenie) a človek môže dýchať, no počas tvorby hlasu sa uzavrie a hlasivky sa nachádzajú v tzv. fonačnom postavení. Prúdom vzduchu narážajúcim na hlasivky vzniká kvázi periodické kmitanie membrány hlasiviek, ktoré vytvára zvukové pulzy a nazýva sa základný hlasivkový tón F_0. Jeho frekvencia sa označuje ako základná (fundamentálna) hlasová frekvencia a odpovedá výške hlasu akú poslucháč v danej reči vníma [8].

Pri tvorbe hlasu sa využíva oboch postavení hlasiviek. Pri fonačnom postavení vznikajú znelé zvuky ako napríklad samohlásky a pri kľudovom postavení vznikajú neznelé zvuky. Tie sú charakteristické tým, že neobsahujú základný hlasivkový tón a vznikajú až nasledovnou modifikáciou v nadhrtanových dutinách. Ďalej obsahujú viac či menej šumu, ktorý je základom mnohých spoluhlások ako sú napr. sykavky (s, š...).

1.1.3 Artikulačné ústrojenstvo

Sa skladá z nadhrtanových dutín a z artikulačných orgánov. Dutiny sa na tvorbe reči zapájajú pasívne a artikulačné orgány, teda hlavne jazyk, mäkké podlažie a pery sa podieľajú aktívne, teda menia svoju veľkosť, poprípade postavenie pri tvorbe rôznych zvukov reči. V nadhrtanových dutinách sa pri prechode základného hlasivkového tónu prejaví rezonancia, ktorú je možné vyjadriť ako kaskádne zapojenie dvojpólových rezonátorov. To podnieti rozloženie akustickej energie okolo určitých frekvencií. Nazývajú sa formantové frekvencie F_1, F_2, F_3. Dochádza teda k zvýšeniu energie určitých frekvenčných pásiem, ku vzniku tzv. formantov. Ak sa do tohto procesu zapojí aj nosná dutina, potlačí naopak niektoré (antiformantové) frekvencie. Výsledný zvuk je potom daný superponovanou hodnotou zvuku z ústnej a nosnej dutiny. Naopak ak artikulačným ústrojenstvom základný hlasivkový tón neprechádza, teda jedná sa o kľudové postavenie hlasiviek, atikulátory svojou priestorovou modifikáciou (zmenou veľkosti, priechodnosti atď.) vytvárajú vydychovanému vzduchu prekážky. Prechodom vydychovaného vzduchu cez rôzne prekážky vznikajú rôzne druhy šumu. Šum tvorí základ všetkých spoluhlások. Behom reči sa postavenie všetkých artikulátorov plynule mení a vzniká nespočetne mnoho konfigurácií hlasového traktu, ktoré majú za následok vznik všetkých hlások v hovorenom jazyku. Podrobnejšie informácie napríklad v [7], [8].

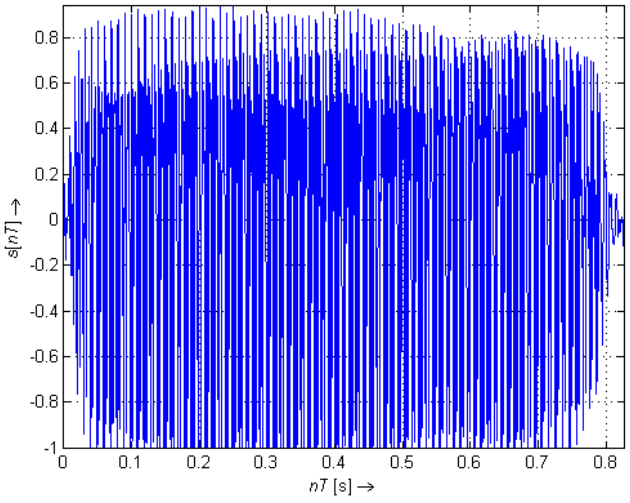
1.2 Rečové jednotky

Pre segmentálny popis reči je charakteristické to, že na reč sa pozeráme ako na postupnosť jednotlivých akustických segmentov, tj. hlások a ich vzťahu k abstraktnej jednotke zvanej foném. Reč ako hovorenú podobu jazyka skúma veda zvaná lingvistika. Pozerá sa na ňu z viacerých hľadísk (úrovní), akustického, fonetického, fonologického, morfologického atď.

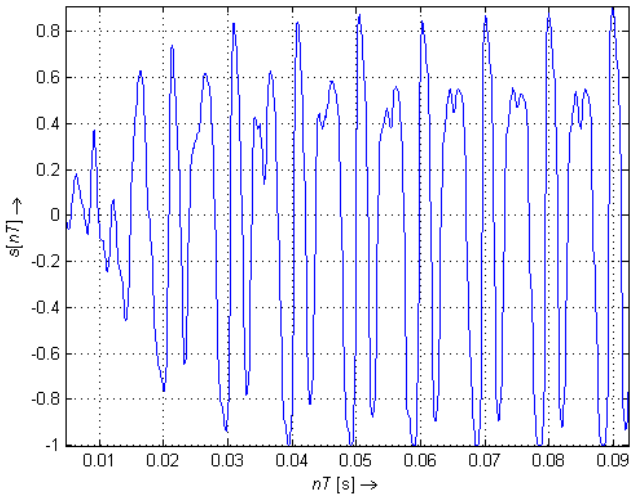
1.2.1 Akustická úroveň

Na akustickej úrovni sa lingvistika zaoberá fundamentálnou frekvenciou hlasiviek, intenzite reči a spektrálnemu rozloženiu jej energie. Spektrogram slúži k zobrazeniu krátkodobého modulového spektra v čase. Reč má široké, pestré spektrum a preto sa na jeho spektrálnu analýzu používajú banky filtrov, ktoré daný spektrogram vytvárajú. Rozlišujeme úzkopásmové a širokopásmové spektrogramy. U znelých zvukov prevláda rozloženie energie v oblasti nízkych kmitočtov (viď Obr. 1. 1), tj. v oblasti prvých troch formantových frekvencií. Naopak u neznelých zvukov zase prevládajú vysokofrekvenčné zložky (viď Obr. 1. 2). Viac informácií na [7], [8], [10], [1].

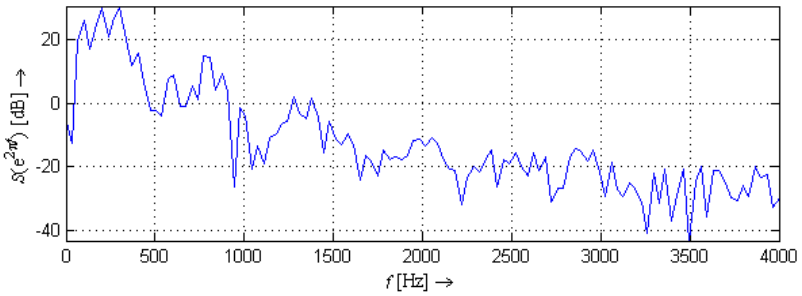
a)



b)

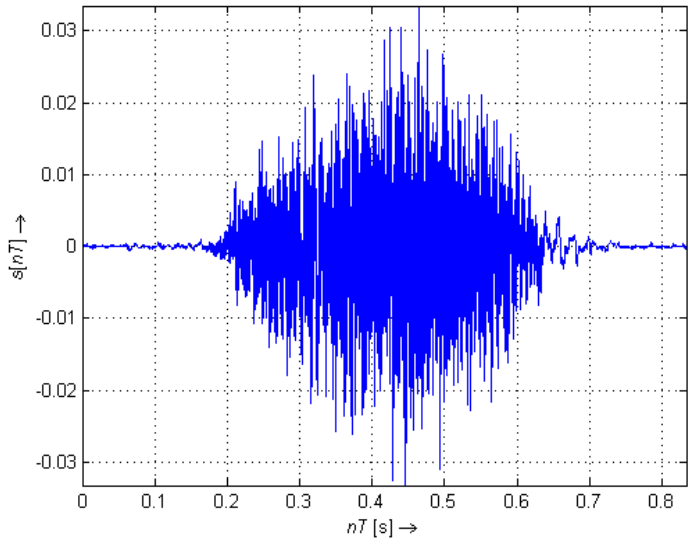


c)

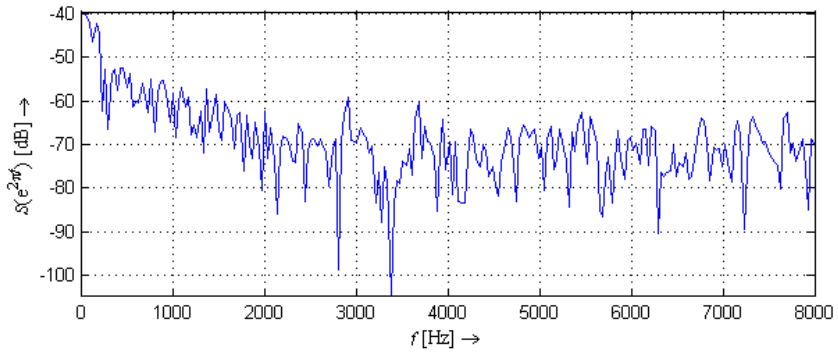


Obr. 1. 1: Časová a spektrálna charakteristika znelého segmentu rečového signálu: a) časový priebeh hlásky u ($\omega = 2\pi f_{vz}$, $f_{vz} = 8000\,\mathrm{Hz}$); b) zozoomovaný časový priebeh (je vidno kvázy-periodicitu kmitania hlasiviek vo fonačnom postavení); c) modulové spektrum hlásky e.

a)

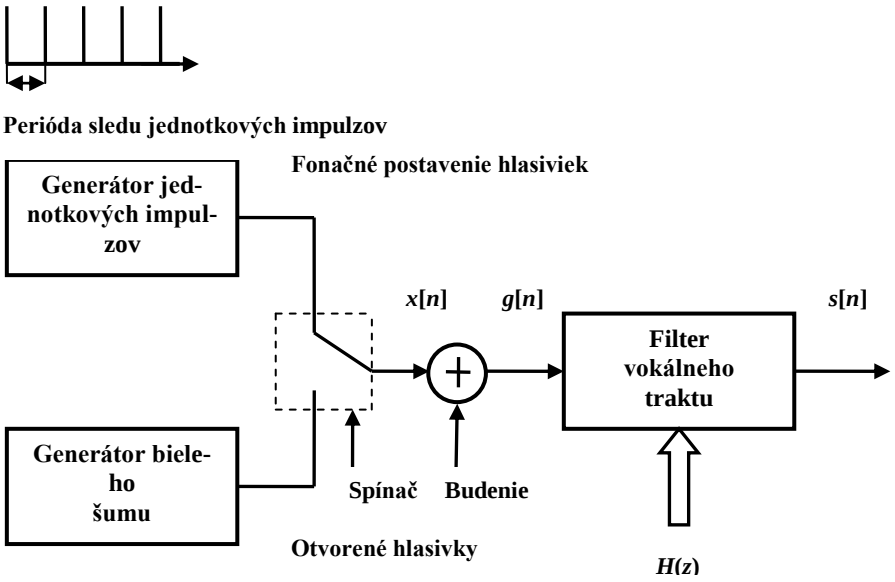


b)



Obr. 1. 2: Časová a spektrálna charakteristika neznelého segmentu rečového signálu: a) časový priebeh neznelej hlásky s ($\omega = 2\pi f_{vz}$, $f_{vz} = 16000$ Hz); b) modulové spektrum hlásky s.

Väčšina metód analýzy rečového signálu vychádza so zjednodušeného modelu generovania reči (viď Obr. 1. 3).



Obr. 1. 3: Model generovania reči.

Je založený na fakte, že pri fonačnom postavení hlasiviek je signál vstupujúci do artikulačného ústrojenstva tvorený (pre zjednodušenie) jednotkovými impulzmi o perióde T_0 a pri otvorených hlasivkách je to šum, ktorý je v tomto modelu reprezentovaný bielym šumom. Biely šum má približne konštantnú hustotu výkonu v celom frekvenčnom pásme. Z toho vyplýva, že signál g_0 bude tvorený buď sledom jednotkových impulzov alebo bielym šumom. Tento signál je potom ďalej modulovaný vokálnym traktom, ktorý upraví jeho spektrum. Vokálny

trakt je pri spracovávaní krátkych segmentov reči možné pokladať za časovo nepremenný lineárny systém, ktorého výsledný signál bude daný konvolúciou impulzovej odozvy $h[n]$ a vstupného signálu podľa rovnicou [12]:

$$s[n] = g[n] * h[n] = \sum_{m=-\infty}^{\infty} g[m]h[n-m]. \tag{1.1}$$

Táto rovnica sa volá diskrétna lineárna konvolucia. Konvoúcia v časovej oblasti odpovedá násobeniu vo frekvenčnej, teda aj vzorec (1.1) sa transformuje na súčet spektra budiaceho signálu a frekvenčnej charakteristiky vokálneho traktu. Vzťah medzi impulzovou odozvou lineárneho časovo nepremenného systému (LTI) a jeho frekvenčnou charakteristikou je vyjadrený pomocou vzťahov [12]:

$$H(\mathrm{e}^{\mathrm{j}\omega}) = \sum_{n=-\infty}^{\infty} h[n] \cdot \mathrm{e}^{\mathrm{j}\omega n}. \tag{1.2}$$

$$h[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} H(\mathrm{e}^{\mathrm{j}\omega}) \cdot \mathrm{e}^{\mathrm{j}\omega} \mathrm{d}\omega. \tag{1.3}$$

Zo signálu $s[n]$ v časovej oblasti sa do kmitočtovej dostávame pomocou Fourierovej transformácii diskrétneho času (DTFT), ktorá je periodická s periódou 2π. V spracovaní signálov a najmä rečového signálu sa pre tento prechod používa tzv. Z-transformácia, definovaná [11]:

$$F(z) = \sum_{n=-\infty}^{\infty} f[n] \cdot z^{-n},$$

kde $f[n]$ je vstupný signál (pre náš prípad $g[n]$). Viac o Z-transformácii v [10], [11], [12], [13].

1.2.2 Fonetická úroveň

Oborom skúmania je činnosť artikulačného ústrojenstva hlasového traktu pri tvorbe reči. Fonetika teda študuje komplexne vznik reči od vzniku vydychovaného prúdu vzduchu až po jeho konečnú úpravu v artikulátoroch, o charakter výsledných zvukov a tiež o to ako bude poslucháč vnímať práve počutý zvuk. Základnou rečovou jednotkou na fonetickej úrovni je hláska, zložením hlások vznikajú morfémy a kombináciou morfém sa tvoria slová. Hlásky sú súbor foneticky podobných zvukov vydaných človekom. Z toho vyplýva, že na reč sa môžeme pozerať ako na postupnosť hlások, ktoré na seba nadväzujú [8], [2], [6]. Hlásky delíme na samohlásky (vokály) a spoluhlásky (konsonanty).

Samohlásky: [a], [e], [i], [o], [u], [á], [é], [í], [ó], [ú] sú čisté tóny, pri ktorých vydychovaný vzduch neprekonáva žiadnu prekážku, teda nevzniká šum. Úprava sa deje až polohou jazyka, pootvorením úst atď. Spojením dvoch samohlások vznikajú dvojhlásky.

Spoluhlásky: sa ešte ďalej delia podľa miesta artikulácie na záverové (okluzívy), výbuchové (explozivy), úžinové (konstriktívy), trené (frikatívy), polozáverové (semiokluzívy), polotrené (afrikáty) a spôsobu artikulácie na – obojperné (bilabiálne) a perozubné (labiodentálne), ďasnové (alveolárne), hrtanové (laryngálne) atď.

1.2.3 Fonologická úroveň

Na fonologickej úrovni sa skúmajú jednotlivé zvuky z hľadiska systémovej stavby jazyka. Zaujíma sa teda o postavenie, funkciu a jednotlivé vzťahy medzi týmito zvukmi v rámci jazyka. Fonológia je niečo ako prechodová veda k vyššej lingvistike.

Jej základnou jednotkou je jeden foném, ktorý je definovaný ako najmenšia lingvistická jednotka schopná rozlišovať významové jednotky reči. Je to abstraktná jednotka, ktorá nepredstavuje nejaký konkrétny zvuk, ale lingvistickú povahu danej konfigurácie hlasového traktu. Až artikuláciou istého fonému vzniká zvuk. Každý foném má teda svoju vlastnú konfiguráciu hlasového traktu a prechodom od jednej konfigurácie k druhej v dôsledku zmeny fonému, môže byť nasledujúca konfigurácia ovplyvnená predchádzajúcou. Nastáva tzv. koartikulácia. Následkom týchto variácií môže byt jeden foném realizovaný viacerými zvukmi. Zvuky s podobnými fonetickými vlastnosťami sa nazývajú hlásky, z toho teda vyplýva, že foném môže byť realizovaný viacerými hláskami, ktoré budeme súhrnne nazývať alofóny, teda varianty fonému. Podrobnejšie informácie sú na [2], [6].

1.3 Suprasegmentálne rysy (prozódia)

Pod termínom prozódia sú myslené také vlastnosti akustického rečového signálu, ktoré súvisia hlavne s fundamentálnou frekvenciou hlasového traktu, intenzitou a časovaním. Zmena základnej hlasivkovej frekvencie sa prejaví v zmene melódie reči a zmena intenzity a časovania zase v zmene tempa. Suprasegmentálny popis sa teda nezaoberá hláskami alebo fonémami ale väčšími časťami reči (slabikami, slovami). Suprasegmentálne javy sú popisované na troch úrovniach, akustickej, percepčnej a lingvistickej. Základné akustické parametre charakterizujúce prozódiu sú fundamentálna frekvencia (základná frekvencia hlasového traktu), ďalej intenzita (amplitúda, resp. energia) a doba trvania rečových segmentov (hlások alebo fonémov). Percepčná úroveň popisuje prozodické charakteristiky tak ako ich vníma priemerný poslucháč. Akustické parametre sa tu volajú výška hlasu (melódia), hlasitosť a dĺžka segmentov reči. Z hľadiska lingvistiky sa jedná o tón, intonáciu a prízvuk [7], [8]. Ďalší suprasegmentálny jav, ktorý vychádza zo zmeny časovania a rýchlosti hovorenej reči je tempo reči. Keďže tempo je ústrednou témou tejto práce povenujeme sa mu podrobnejšie.

1.3.1 Tempo reči

Je suprasegmentálny, časovo modulovaný jav, ktorý je daný rýchlosťou práce artikulátorov a je daný množstvom slabík (alebo slov) vypovedaných za časovú jednotku (najčastejšie sekundu). Viaže sa teda na celkovú rýchlosť reči. Jednotlivé jazyky majú vlastné tempo, vtedy hovoríme o tzv. národnom tempe a existuje aj individuálny návyk tempa reči. Tempo ako také závisí od mnohých činiteľov – napr. od obsahu a formy prejavu, od jeho vnútornej stavby (napr. dramatické úseky sa môže artikulovať rýchlym rečovým tempom, zdôrazňované časti sa môžu vyslovovať pomalším tempom apod.) alebo od dôležitosti informácie, ktorú reč sprostredkúva (informácia ktorá je dôležitá sa väčšinou hovorí pomalšie a zreteľnejšie). Zmenu rýchlosti produkcie reči, teda tempa spôsobuje preťahovanie alebo naopak skracovanie jednotlivých hlások. V reči sa takto upravujú najmä samohlásky ale občas aj spoluhlásky. Pri modulácii súvislej reči však nejde o zmeny dĺžky jednotlivých slabík, ale o zmeny vyšších celkov ako napr. slabík. Tempo reči sa dá merať pomocou rôznych algoritmov, ktoré najprv súvislú reč rozdelia na jednotlivé segmenty (hlásky) a potom vypočítajú ich počet za daný čas. Väčšina však stroskotá na tzv. plozivách, sú to hlásky ako napr. [p, b] atď. , ktoré vznikajú tak, že prúdu vzduchu z pľúc sa do cesty postavia uzavreté hlasivky. Tie sa potom otvoria a všetok tento vzduch naraz putuje vokálnym traktom. Z toho vyplýva, že časový priebeh plozív sa skladá z oblasti šumu, charakterizujúcej uzavreté hlasivky a nárazovému zákmitu. A v tom je problém s detekciou plozív. Pri segmentácii sa môže stať, že daný algoritmus bude plozivu brať ako dve hlásky. Jednu šumovú a zákmit.

2 Analýza reči

2.1 Predspracovanie rečového signálu

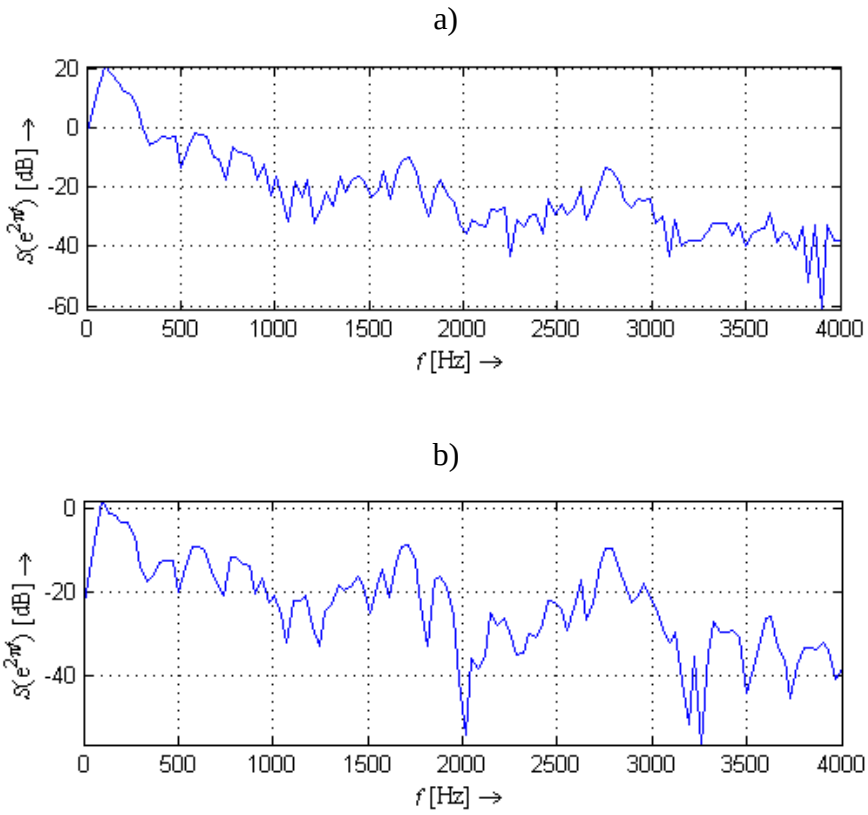
Ľudská reč je veľmi rozmanitá a časovo premenná. Záleží aj na individuálnej reprezentácii jednotlivými rečníkmi. Dnešné spracovanie reči prebieha už takmer výlučne číslicovo, preto je potrebné tento vstupný rečový signál pred prevodom na číslicový signál a jeho samotným spracovaním ešte upraviť. Táto úprava väčšinou prebieha pomocou filtrácie vstupného signálu. Filtrácia vstupného signálu je operácia, ktorá má za cieľ upraviť vstupný signál tak, aby zdôraznila vybrané časti frekvenčného spektra signálu, zlepšila odstup signálu od šumu a pod. čím sa jeho následné spracovanie stáva ľahším a efektívnejším. Bližšie informácie sa nachádzajú napr. v [1], [9]. Pre výpočet tempa reči je ešte pred samotným predspracovaním, ustredením a filtráciou, vhodné odstrániť prípadné pauzy v reči. Na tento účel sa používa tzv. VAD (Voice Activity Detector), ktorý je založený na energii signálu. Sofistikovanejšie VAD systémy sú už v dnešnej dobe založené na periodicite signálu [3].

2.1.1 Preemfáza (zdôraznenie vyšších frekvencií)

Jednou zo základných úprav rečového signálu pred ďalším spracovaním je číslicová filtrácia tzv. preemfázovým filtrom typu hornej priepusti (HP), ktorý zdôrazňuje vyššie frekvenčné zložky rečového signálu približne o 20 dB na dekádu, čím sa vyrovná jeho prirodzený útlm, spôsobený vyžarovacou charakteristikou úst a zároveň sa zdôraznia percepčne najvýznamnejšie zložky spektra reči. Z toho vyplýva, že preemfáza vyrovnáva kmitočtovú charakteristiku vstupného signálu. Ako už bolo spomenuté preemfázový filter najčastejšie realizujeme filtrom typu horná priepusť (HP) s konečnou impulznou odozvou s prenosovou funkciou, kde a_{pre} je koeficient filtru (väčšinou prvého rádu) [12]:

$$H_{pre}(z) = 1 - a_{pre} z^{-1}. \tag{2.1}$$

Príklad signálu bez a po preemfáze je na (Obr. 2. 1):



Obr. 2. 1: Modulové spektrum signálu hlásky e: a) bez preemfáze; b) s preemfázou.

2.1.2 Ustredenie (odstránenie jednosmernej zložky)

Rečový signál môže obsahovať jednosmernú zložku (vplyv prenosového kanálu, zvukovej karty a pod.), ktorá nenesie žiadnu informáciu. Naopak pri určovaní niektorých charakteristík rečového signálu ako je napr. počet prechodov nulou, energia signálu atď. môže spôsobiť chybné určenie hodnôt, preto je vhodné jednosmernú zložku odčítaním odstrániť [1].

$s'[n] = s[n] - \mu_s$, kde μ_s je potrebné odhadnúť. (2.2)

Strednú hodnotu môžeme počítať dvoma spôsobmi:

1. Ako tzv. off-line, ktorá sa počíta priemerovaním vstupného signálu:

$$s'[n] = \frac{1}{N}\sum_{n=0}^{N-1} s[n]. . \quad (2.3)$$

2. A ako strednú hodnotu on-line. Kedy nemáme k dispozícii celý signál alebo naopak máme ale je príliš dlhý a neustále s časom pribúda. Vtedy sa stredná hodnota počíta rekurzívne pomocou vzorca:

$s'[n] = \gamma s'[n-1] + (1-\gamma)s[n]$, kde γ sa blýži k 1. (2.4)

2.1.3 Segmentácia rečového signálu

Rečový signál je ako taký nestacionárny (dá sa povedať náhodný). Na výpočet parametrov však potrebujeme potrebujeme aby bol signál stacionárny. Preto sa za týmto účelom rečový signál segmentuje na menšie časti (rámce, segmenty, mikrosegmenty apod.), na ktorých sa daný signál už chová približne stacionárne. Artikulačné orgány vykazujú zotrvačnosť, v jednej polohe zotrvajú určitý čas (10 – 30ms), podľa toho sa volí aj dĺžka segmentu, resp. mikrosegmentu okolo 20 ms, tj. (160 až 200 vzorkou pre vzorkovaciu frekvenciu 8000 Hz), aby bol daný segment dostatočne krátky a teda stacionárny a zároveň dostatočne dlhý aby postihol kvázi periodicitu rečového signálu, a aby sme mohli príznaky určiť s požadovanou presnosťou. Segmentáciu rozdeľujeme na segmentáciu s prekrývaním rámcov a na segmentáciu bez prekrývania rámcov [1].

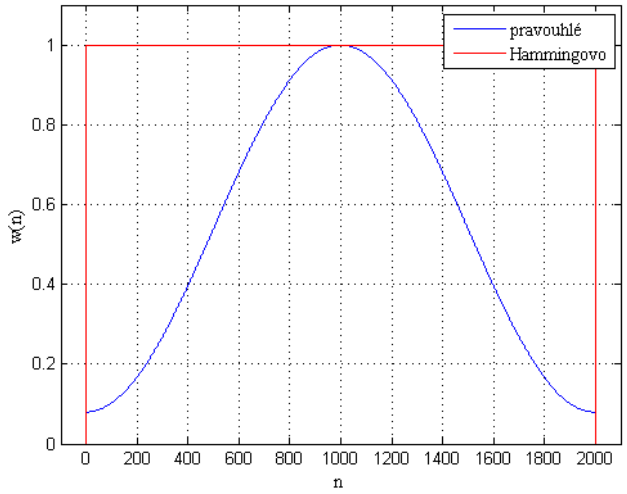
Vhodnejšia z hľadiska priebehu počítaných príznakov je segmentácia s prekrývaním. Čím menšie bude prekrytie jednotlivých rámcov, tým bude väčšia rýchlosť posunu signálu v čase a tým menšie budú aj nároky na pamäť. Ale na druhej strane sa zase hodnota príznakov môže od jedného rámca k druhému skokovo meniť. Veľké prekrytie rámcov zabezpečí hladký priebeh parametrov (príznakov) a menšie nároky na pamäť, ale výsledné príznaky môžu byť rámec od rámca príliš podobné, čo nie je dobré z hľadiska detekcie daných parametrov a ich rozpoznávania. Počet rámcov (N_{ram}) je možné vypočítať podľa vzťahu:

$$N_{\text{ram}} = 1 + \left[\frac{N - I_{\text{ram}}}{s_{\text{ram}}} \right]. \tag{2.5}$$

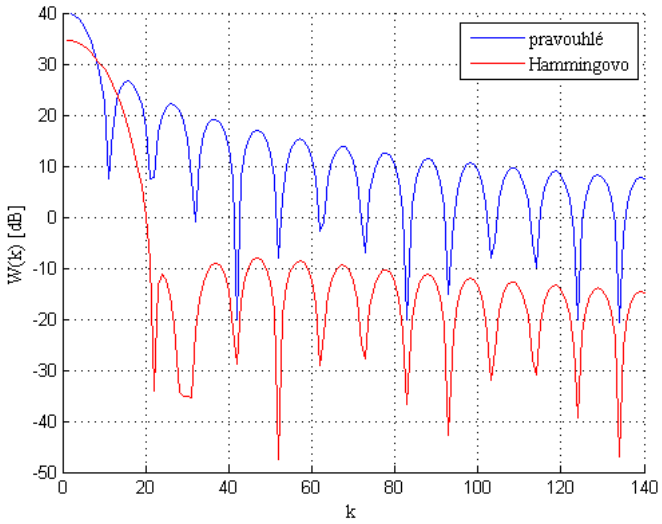
Pri segmentácii bez prekrývania, teda $p_{\text{ram}} = 0$ sa počet rámcov vypočíta podľa vzťahu:

$$N_{\text{ram}} \left[\frac{N}{I_{\text{ram}}} \right]. \tag{2.6}$$

Segmentácia, resp. tvorba rámcov sa realizuje tzv. okienkovou funkciou. V číslicovom spracovaní signálov sa používa viacero typov okien, ako napr. pravouhlé, Hammingovo, Hannovo, Blackmanovo, Barlettovo a iné. Príklad na Hammingovo a pravouhlé okno v časovej oblasti je na (Obr. 2. 2) a na ich modulové kmitočtové charakteristiky na (Obr. 2. 3).



Obr. 2. 2: Pravouhlé a Hammingovo okienko v časovej oblasti.



Obr. 2. 3: Pravouhlé a Hammingovo okno v kmitočtovej oblasti (modulová kmitočtová charakteristika).

Pre pravouhlé okno platí vzťah [10]:

$$w[n] = 1 \quad \text{pre } 0 \le n \le I_{ram} - 1. \tag{2.7}$$

Kde $w[n]$ je hodnota vzorku signálu po násobení oknom. Inde je $w[n]$ nulové. Pre Hammingovo okno platí [10]:

$$w[n] = 0{,}54 - 0{,}46\cos\left(\frac{2\pi n}{I_{ram} - 1}\right) \quad \text{pre } 0 \le n \le I_{ram} - 1. \tag{2.8}$$

Inde je $w[n]$ taktiež nulové.

Rámec vzniká násobením signálu a okienkovej funkcie v časovej oblasti. Keďže násobenie v časovej oblasti odpovedá kovolúcii v kmitočtovej, zmení sa aj spektrum signálu v rámci. Z obrázkov je vidno, že Hammingovo okno má širší hlavný lalok, z toho vyplýva, že má horšie kmitočtové rozlíšenie a širšie prechodové pásmo ako pravouhlé okno. Naopak potlačenie postranných lalokov má omnoho väčšie ako pravouhlé okno, takže má lepšie vlastnosti v nepriepustnom pásme a do spektra vysekávaného signálu sa nedostanú takmer žiadne spektrálne zložky z vedľajších okien. Z týchto dôvodov sa pri spracovaní rečového signálu používa skôr používa Hammingovo ako pravouhlé okno.

2.2 Homomorfná transformácia rečového signálu

Rečový signál je možné transformovať a modelovať ako súčet budiaceho signálu a odozvy na vokálny trakt. K tomuto účelu sa používa homomorfná transformácia (HMF) rečového signálu. Jednou zo základných oblastí použitia HMF je dekonvolúcia rečového signálu, ktorej cieľom je oddelenie charakteristík budenia od charakteristík hlasového traktu. HMF sa pre jej efektívnosť postupne stala jedným z hlavných nástrojov spektrálnej analýzy rečového signálu, pomocou ktorého vieme získať súbor parametrov, reprezentujúcich spektrálnu obálku. Tieto parametre sa nazývajú kepstrálne koeficienty a sú výsledkom tzv. kepstrálnej transformácie (KT). Základnou požadovanou vlastnosťou KT je schopnosť transformovať konvolúciu na všeobecnú operáciu "#", umožňujúcu oddelenie signálových zložiek lineárnym filtrom (LF).
Najväčšie uplatnenie v oblasti spracovania rečového signálu má logaritmická kepstrálna transformácia definovaná [5]:

$$\hat{x}[m] = \frac{1}{2\pi j}\oint_c \ln[X(z)]z^{m-1}\mathrm{d}z \quad \text{pre } -\infty < m < \infty, \tag{2.9}$$

kde

$$X(z) = \sum_{n=-\infty}^{\infty} x(n) z^{-n} \tag{2.10}$$

a $\hat{x}[m]$ je tzv. komplexné kepstrum, ktoré je obrazom vstupného signálu v transformovanej tzv. kvefrenčnej oblasti. Komplexné kepstrum je obojstranná neohraničená postupnosť, z toho teda vyplýva, že sa chová ako impulzná odozva nekauzálnej sústavy s prenosovou funkciou $\hat{X}(z) = \ln[X(z)]$. Ak vstupný signál je signálom s minimálnou fázou, zodpovedajúce komplexné kepstrum je kauzálne, teda póly a nulové body ležia vo vnútri jednotkovej kružnice. Inverznú kepstrálnu transformáciu definujeme vzťahom [5]:

$$x(n) = \frac{1}{2\pi \mathrm{j}} \int_c \exp[\hat{X}(z)] z^{n-1} \mathrm{d}z \quad \text{pre} \quad -\infty < \mathrm{n} < \infty, \tag{2.11}$$

kde

$$\hat{X}(z) = \sum_{m=-\infty}^{\infty} \hat{x}(m) z^{-m} \,. \tag{2.12}$$

Komplexné kepstrum najčastejšie realizujeme pomocou diskrétnej Fourierovej transformácie (DFT) tak, že do vzťahu (2.3) dosadíme $z = e^{\mathrm{j}\Omega}$, kde $\Omega = 2\pi f / f_{vz}$. Spätnú kepstrálnu transformáciu realizujeme inverznou diskrétnou Fourierovou transformáciou (IDFT). Myšlienka homomorfnej dekonvolúcie rečového signálu vychádza zo zjednodušeného modelu generovania reči na základe ktorého môžeme rečový signál vyjadriť pomocou konvolúcie budiaceho signálu $g[n]$ a impulzovej odozvy lineárnej sústavy $h[n]$, reprezentujúcej hlasový trakt (1.1), ktorý sa do kepstrálnej oblasti transformuje na súčet:

$$\hat{s}(m) = \hat{g}(m) + \hat{h}(m), \tag{2.13}$$

kde $\hat{g}(m)$ je komplexné kepstrum budiaceho signálu a $\hat{h}(m)$ je komplexné kepstrum impulzovej odpovede hlasového traktu.

2.2.1 Realizácia kepstrálnej transformácie

Najprv je potrebné urobiť DFT konvolučnej zmesi signálu budiaceho traktu a impulznej odpovede vokálneho traktu. DFT transformuje konvolúciu na súčin spektier tak, ako je to v rovnici (2.15).

$$\mathrm{DFT}\{s[n]\} = \mathrm{DFT}\{g[n] * h[n]\}. \tag{2.14}$$

Keďže konvolúcii v časovej oblasti odpovedá násobenie v kmitočtovej, tak sa aj vzťah (2.14) transformuje do podoby:

$$S[k] = G[k] \cdot H[k]. \tag{2.15}$$

V ďalšom kroku sa súčin zlogaritmuje: :

$$\ln(S[k]) = \ln(H[k] \cdot G[k]) = \ln(H[k]) + \ln(G[k]). \tag{2.16}$$

Takto transformovaný signál do kepstrálnej oblasti je už súčtom. Teraz je už možné signály od seba oddeliť filtráciou a takto upravený signál môžeme inverznými matematickými úpravami vrátiť z transformácie späť do časovej oblasti, čiže najprv spravíme IDFT a takisto ako to platilo predtým pre konvolúciu, operácia súčtu sa aj v obraze rovnala súčtu a na odlogaritmovanie použijeme funkcie exponentu.

2.2.2 Reálne kepstrum

Komplexné kepstrum sa skladá z reálnej zložky, tzv. reálneho kepstra a imaginárnej zložky, tzv. fázového kepstra. V spracovaní reči sa komplexné kepstrum väčšinou nepoužíva. Nahradzuje sa reálnym kepstrom, pretože jeho impulzná odozva sa dá rozdeliť do dvoch častí. Po určitú frekvenciu ide o odozvu na signál budenia a od tejto frekvencie ďalej sa už jedná o odozvu na vokálny trakt. Na impulznej odozve na vokálny trakt môžeme pozorovať lokálne extrémy (zákmity), ktoré odpovedajú formantovým frekvenciám. Reálne kepstrum je párnou, reálnou (modulom) časťou komplexného kepstra a môžeme ho vypočítať pomocou IDFT funkcie $\ln\left|X(e^{j\Omega})\right|$. A je zároveň komplexným kepstrom pre signály „s nulovou fázou“ $\arg[X(e^{j\Omega})] = 0$. Reálne kepstrum má tvar [5]:

$$\hat{x}_c(m) = \frac{1}{2\pi}\int_{-\pi}^{\pi} \log\left|X(e^{j\Omega})\right| e^{j\Omega} d\Omega. \tag{2.17}$$

Z toho vyplýva, že komplexné kepstrum je súčtom reálneho a fázového kepstra:

$$\hat{x}[m] = \hat{x}_c[m] + \hat{x}_{phase}[m], \tag{2.18}$$

kde $\hat{x}_{phase}[m]$ je fázové kepstrum pôvodného signálu.

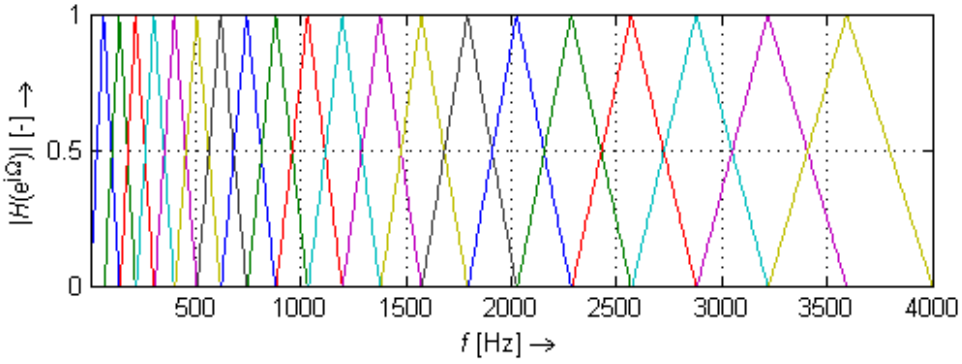
2.2.3 Mel-frekvenčné kepsrálne koeficienty (MFCC)

Kepstrálny model (konkrétne kepstrálne koeficienty) neodpovedá vlastnostiam ľudského sluchu, pretože užitím DFT sa frekvenčné rozlíšenie rozloží po celom spektre rovnomerne. Ľudské ucho ale nemá rovnakú rozlišovaciu schopnosť na všetkých frekvenciách v oblasti počuteľných zvukov. Preto sa zavádzajú tzv. Mel-frekvenčné kepstrálne koeficienty (MFCC), ktoré upravia spektrum tak, aby lepšie odpovedalo nelinearite ľudského sluchu. Táto úprava prebieha tak, že na frekvenčnej ose najprv rozmiestnime nelineárne (konkrétne trojuholníkové) filtre a odmeriame energiu na výstupe každého. Pri výpočte kepstra túto energiu potom použijeme namiesto DFT. Ak by sme chceli docieliť nelinearitu o ktorú ide, museli by sme filtre na frekvenčnej ose nerovnomerne. Preto sa frekvenčná osa transformuje na Melovskú použitím transformačného vzťahu [10]:

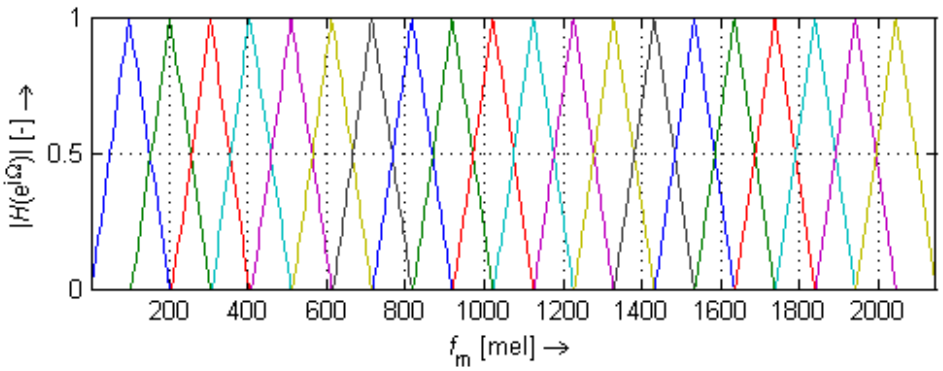
$$f_{Mel} = 2059 \log_{10}\left(1 + \frac{f_{Hz}}{700}\right). \tag{2.19}$$

Použitím tohto vzťahu môžeme teda nelineárne trojuholníkové filtre rozložiť rovnomerne na melovskej ose a na frekvenčnej ose sú rozložené nerovnomerne. Príklad banky trojuholníkových filtrov je na (Obr. 2. 4).

a)



b)



Obr. 2. 4: Banka trojuholníkových filtrov používaná pri výpočte MFCC: a) frekvenčná škála ($\Omega = 2\pi f$, $f_{vz} = 8000\,\text{Hz}, N_{\text{FFT}} = 2048,\ p = 20$); b) melovská škála ($\Omega = 2\pi f_m$).

Frekvenčná charakteristika každého použitého trojuholníkového filtra je rovná [8]:

$$H_i(\mathrm{e}^{\mathrm{j}2\pi f}) = \begin{cases} \dfrac{f - b_{i-1}}{b_i b_{i-1}} & \text{pre } b_{i-1} \le f < b_i \\ \dfrac{f - b_{i+1}}{b_i - b_{i+1}} & \text{pre } b_i \le f < b_{i+1} \\ 0 & \text{pre ostatné prípady} \end{cases}, \tag{2.20}$$

kde $b_{m,i}$ je stredná frekvencia daného filtra v Melovskej škále a je daná vzorcom:

$$b_{m,i} = b_{m,i-1} + \Delta_m, \text{pre} i = 1,2,..., p, \text{ a} \tag{2.21}$$

$$\Delta_m = \frac{B_m}{p+1}, \tag{2.22}$$

kde p je číslo filtra v banke filtrov a B_m je šírka prenosového pásma filtra. Mel-frekvenčné kepstrálne koeficienty počítame tak, že sa najprv pomocou DFT získa výkonové spektrum signálu, ktoré vo frekvenčnej oblasti násobíme bankou trojuholníkových filtrov. Potom sú vzorky každého filtra sčítane a vložené do vektoru o p vzoriek. Vektor nakoniec zlogaritmujeme a namiesto IDFT použijeme diskrétnu kosínusovú transformáciu (DCT). Vďaka použitiu DCT sú koeficienty MFCC menej korelované. Viac informácii k tejto problematike na [7], [8].

2.3 Lineárna prediktívna analýza

Lineárnu prediktívnu analýzu LPC (Linear Predictive Analysis) poznáme dvojakého druhu. Doprednú LPC a spätnú LPC. V tejto kapitole sa zameriame na doprednú lineárnu predikciu. Spätná je svojimi operáciami inverzná k doprednej LPC.

2.3.1 Dopredná lineárna prediktívna analýza

Dopredná lineárna prediktívna analýza (Forward Linear Prediction) patrí medzi najefektívnejšie metódy spracovania akustického (v našom prípade rečového) signálu, využívajúcu krátkodobej analýzy založenej na segmentácii spracovávaného signálu. Princíp LPC je založený na

predpoklade, že $n-$tú vzorku signálu je možné vypočítať ako lineárnu kombináciu predchádzajúcich p vzoriek. LPC sa tak ako väčšina metód spracovania reči taktiež opiera o zjednodušený model generovania reči. Pri prechode signálu vokálnym traktom sa vplyvom rezonančných frekvencií, nazývaných formantové frekvencie, zosilnia niektoré spektrálne zložky. Koeficienty odvodené z LPC (PARCOR) sa využívajú napríklad v mobilnej komunikácii, pretože pri tejto komunikácii sa prenosovým kanálom prenáša rečový signál kódovaný metódou pulznej kódovej modulácie PCM, ktorý neobsahuje základný hlasivkový tón F_0 a rýchlosť takto prenášaných dát je príliš vysoká a treba ju komprimáciou znížiť. Preto sa používa kodér LPC. Najpoužívanejšie LPC kodéry sú [12]:

1. CELP (Code Excited Linear Prediction)
2. ACELP (Algebraic CELP)

LPC ako už bolo spomenuté dokáže odhadnúť veľkosť nasledujúcej vzorky na základe kombinácie niekoľko predchádzajúcich pomocou vzťahu [12] a [8] ostatné vzorce pochádzajú taktiež z tejto literatúry:

$$\bar{s}[n] = -\sum_{i=1}^{p} a_p[i]s[n-i], \tag{2.23}$$

kde a_p je koeficient LPC a p je rád predikcie. Chyba lineárnej predikcie je daná ako rozdiel skutočnej a vypočítanej hodnoty:

$$e[n] = \sum_{i=0}^{p} a_p[i]s[n-i], \text{ kde } a_p[0] = 1. \tag{2.24}$$

Ak za $e[n]$ budeme pokladať výstup filtra, potom je možné vo frekvenčnej oblasti, pomocou Z-transformácie vyjadriť prenosovú funkciu takéhoto systému ako:

$$H(z) = \frac{1}{A(z)}, \text{ kde} \tag{2.25}$$

$$A(z) = 1 + \sum_{i=1}^{p} a_p[i]z^{-i}. \tag{2.26}$$

$A(z)$ je tzv. analyzujúci filter a $H(z)$ je prenosová funkcia syntetizačného filtra. Po prechode analyzujúcim filtrom (na strane mikrofónu) chyba predikcie približne odpovedá budeniu. Naopak na strane sluchátka (príjemcu) sa na syntetizačný filter dostáva toto budenie a pomocou LPC koeficientov sa spätne syntetizuje v reč. Chybu predikcie je potrebné vyjadriť v normovanom tvare. Preto sa zavádza tzv. normalizovaná chyba predikcie daná vzťahom:

$$E = \sum_n e^2[n]. \tag{2.27}$$

Túto chybu sa snažíme minimalizovať. Proces minimalizácie je podrobne popísaný v [5], [7], [8]. Výsledok vedie k sústave lineárnych rovníc:

$$\Gamma_p = a_p \cdot \gamma_p. \tag{2.28}$$

Z týchto tzv. normálnych rovníc počítame predikčné koeficienty a_p. Korelačné koeficienty γ_p odhadujeme na rámcoch o dĺžke N vzoriek. Existujú dve metódy líšiace sa v náhľade na signál vo vnútri rámca [8]:

1. kovariačná metóda: sa nepoužíva, pretože vedie k nestabilite filtra.
2. korelačná metóda sa používa v spracovaní reči, kvôli tomu, že počítáme so zjednodušenými lineárnymi rovnicami a táto metóda ponecháva filter stabilným, pretože vždy počítame s rovnaným počtom vzoriek.

Pri použití korelačnej metódy sú koeficienty γ_p vložené do matice, ktorá je symetrická a na diagonálach má rovnaké hodnoty. Takáto matica sa nazýva Toeplitzova. Na túto sústavu lineárnych rovníc po je možné aplikovať Levinsonov-Durbinov algoritmus. Viac o Levinsonovom-Durbinovom algoritme na [7], [8], [12].

2.3.2 LPC-cepstrum

Sa požíva najmä k energetickej analýze segmentovanej reči, pomocou LPC kepstrálnych koeficientov (LPCC). Spektrálnu hustotu stacionárneho náhodného procesu akým segmentovaná reč je možné odhadnúť pomocou vzťahu [8], ostatné vzorce sú tiež z tejto literatúry:

$$\hat{G}_{\mathrm{LPC}}(f) = \left| \frac{G}{A(z)} \right|^2_{\mathrm{e}^R}, \text{ kde } R = \mathrm{e}^{\mathrm{j}2\pi f/\mathrm{f}_{\mathrm{vz}}}. \tag{2.29}$$

$A(z)$ je polynóm rádu p. Nultý LPCC nesie informáciu o energii daného rámca:

$$c_0 = \ln(G)^2, \text{ kde } c_p \text{ je LPCC rádu } p. \tag{2.30}$$

Ďalšie LPCC sa počítajú pomocou rekurzívnych vzťahov:

$$c(n) = -a_n - \frac{1}{n}\sum_{k=1}^{n-1} k c_k a_{n-k} \quad \text{pre} 1 \leq n \leq p, \tag{2.31}$$

$$c(n) = -\frac{1}{n}\sum_{k=1}^{n-1} k c_k a_{n-k} \quad \text{pre } n \succ p. \tag{2.32}$$

Výhodou LPC kepstrálnych koeficientov je to, že sú menej korelované ako LPC a_p koeficienty. Používajú sa najmä pri realizácii rozpoznávačov reči.

2.3.3 Perceptívne lineárne predikčné koeficienty

Spektrálne parametre rečového signálu, získané jeho spektrálnou analýzou by mali čo možno najlepšie aproximovať charakteristiky reči tak, ako ich vníma ľudské ucho. Lineárne predikčné koeficienty dokonale však túto požiadavku nesplňujú. Jednou z metód získania percepčne významných charakteristík rečového signálu je jeho spektrálna analýza na nelineárnej frekvenčnej stupnici (napr. mel-kepstrálne koeficienty). Transformáciou lineárnej frekvenčnej stupnice na melovskú sa síce zohľadňuje spôsob percepcie výšky tónu, nezohľadňujú sa však ostatné vlastnosti sluchového orgánu človeka, ako je frekvenčná závislosť vnímania hlasitosti, maskovací jav a ďalšie. Preto Hynek Heřmansky publikoval novú metódu. Nazval ju percepčnou lineárnou predikciou (Perceptual Linear Predictive Analysis). Koeficienty získané perceptívnou LPC sa nazývajú perceptívne lineárne predikčné koeficienty PLP, ktoré už zohľadňujú vyššie spomenuté zákonitosti vnímania rečového signálu sluchom človeka. Perceptívnu LPC môžeme rozdeliť do niekoľkých krokov. V prvom kroku sa najprv aktuálny mikrosegment v časovej oblasti násobí Hammingovým oknom a pomocou DFT sa pretransformuje do frekvenčnej oblasti, kde sa vypočíta jeho výkonové spektrum podľa vzťahu [8]:

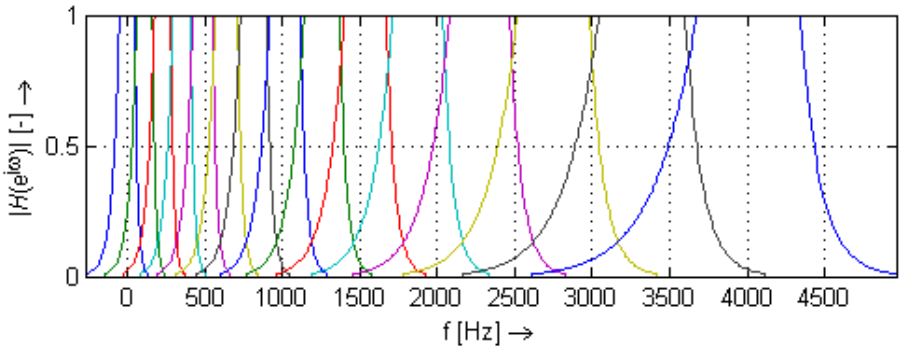
$$P(\Omega) = |S(\Omega)|^2 = \mathrm{Re}[S(\Omega)]^2 + \mathrm{Im}[S(\Omega)]^2, \text{ kde } \Omega = \mathrm{e}^{\mathrm{j}\omega}. \tag{2.33}$$

V ďalšom kroku sa lineárna frekvenčná stupnica spektra $P(\Omega)$ pretransformuje na nelineárnu Barkovu stupnicu podľa:

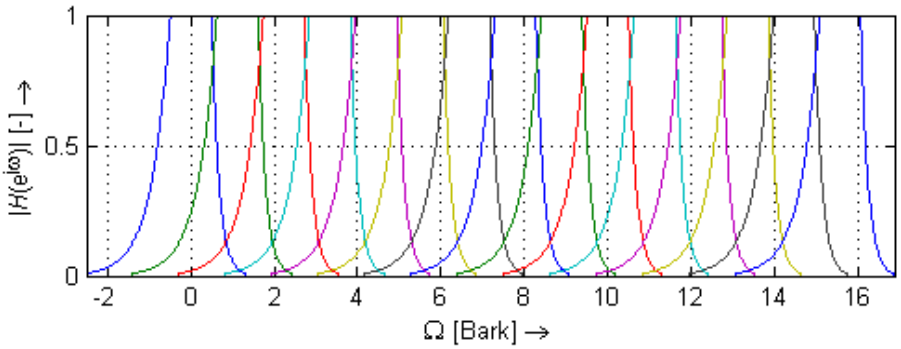
$$\Omega_B = 6\log\left\{\frac{\Omega}{1200\pi} + \left[\left(\frac{\Omega}{1200\pi}\right)^2 + 1\right]^{\frac{1}{2}}\right\}[Bark]. \tag{2.34}$$

Príklad nelineárnej Barkovej škály (stupnice) je na (Obr. 2. 5).

a)



b)

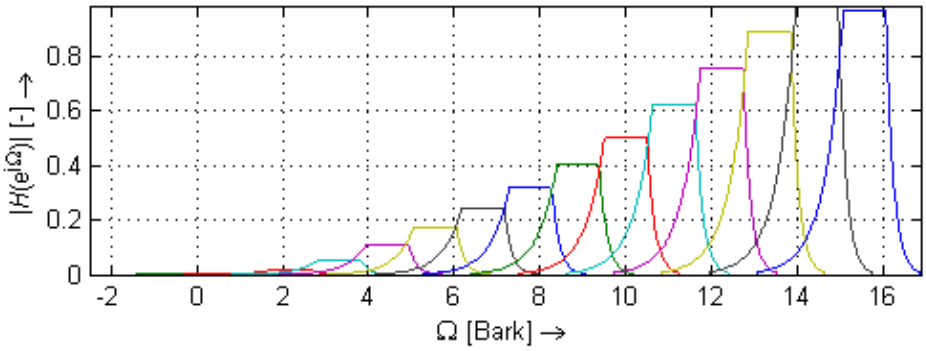


Obr. 2. 5: Rozloženie banky filtrov používane pri výpočte LPCC: a) frekvenčná škála ($\omega = 2\pi f$, $f_{vz} = 8000\,\text{Hz}, N_{\text{FFT}} = 2048,\ p = 15$); b) barkova škála ($\omega = \Omega(\omega)$).

Takto upravené spektrum je podobné konvolúcii s výkonovým spektrom $\Psi(\Omega_B)$ banky filtrov (pásmových priepustí s charakteristickou šírkou pásma). $\Psi(\Omega_B)$ spektrum aproximuje tvar maskovacej krivky ľudského ucha. $\Psi(\Omega_B)$ je daná vzťahom [8]:

$$\Psi(\Omega_B) = \begin{cases} 0 & \text{pre } \Omega_B < -1.3 \\ 10^{2.5(\Omega_B+0.5)} & \text{pre } -1.3 \le \Omega_B \le -0.5 \\ 1 & \text{pre } -0.5 < \Omega_B < 0.5 \\ 10^{-1.0(\Omega_B-0.5)} & \text{pre } 0.5 \le \Omega_B \le 2.5 \\ 0 & \text{pre } \Omega_B \succ 2.5 \end{cases}. \tag{2.35}$$

Banka filtrov upravená podľa vzťahu (2.35) je na (Obr. 2. 6).



Obr. 2. 6: Banka filtrov po konvolúcii s funkciou aproximujúcou krivky rovnakej hlasitosti.

Výkonovému spektru konvolúcie Ω_B a $\Psi(\Omega_B)$ upravíme amplitúdu podľa kriviek rovnakej hlasitosti použitím vzťahu:

$$\mathrm{H}(\Omega_B)^* = \mathrm{H}(\Omega_B)^{0.33}. \tag{2.36}$$

$\mathrm{H}(\Omega_B)$ je spektrum signálu vynásobené bankou filtrov v Barkovej škále. Po upravení amplitúdy prevedieme spektrum späť do časovej oblasti pomocou IDFFT a pokračujeme ďalej ako pri koeficientoch LPC.

2.4 Vektorová kvantizácia

Kvantizácia je proces, pri ktorom sa jedna hodnota aproximuje inou hodnotou z konečného počtu hodnôt. Kvantizáciu rozdeľujeme na:

1. Skalárnu kvantizáciu: ak kvantujeme iba jednu veličinu (parameter).
2. Vektorovú kvantizáciu: ak kvantujeme vektory (poprípade bloky vektorov, čiže matice).

Vektorová kvantizácia sa veľmi často používa pri spracovaní príznakov (ktoré sú usporiadané do vektorov) popisujúcich jednotlivé mikrosegmenty rečového signálu. Ak postupnosť vektorov príznakov pre jednotlivé mikrosegmenty spojíme vznikne matica príznakov s počtom riadkov odpovedajúcim rádu koeficientov použitej metódy na získanie daného/daných príznakov a počtom stĺpcov odpovedajúcim počtu mikrosegmentov. Pri vektorovej kvantizácii ide o to priradiť vektor príznakov $X = [x_1, x_2, ..., x_L]$ (poprípade viacero vektorov) jednému vektoru z kódovej knihy $Y = [y_1, y_2, ..., y_L]$ podľa vzťahu [8]:

$$y_l^* = \arg_{l=1,...,L} \min d(x_k, \mathrm{y}_1) \ . \tag{2.37}$$

Tento vektor sa nazýva centroid a je pre neho charakteristické že má od každého vektoru v danom zhluku vektorov minimálnu vzdialenosť $d(x_k, y_1)$. Ak napr. do dvojdimenzionálnej sústavy xy vynesieme na x-ovú os hodnoty formantovej frekvencie F_1 a na y-ovú os hodnoty formantovej frekvencie F_2 zistíme, že dané zhluky vektorov odpovedajú niektorým hláskam vysloveným rôznymi rečníkmi, teda úlohou kódovej knihy je aby si tzv. natrénovala tieto vektory a z nich určila centroidy. Na trénovanie kódovej knihy je viacero postupov. Najpresnejší je asi ten, že kvantizér najprv spracuje celý rečový signál segment po segmente a z celej matice príznakov si natrénuje zhluky vektorov, z ktorých vypočíta centroidy. K tomuto účelu boli vyvinuté rôzne algoritmy. Medzi najpoužívanejšie patria algoritmus k-means, LBG (Linde-Buzo-Gray) alebo fuzzy k-means [1]. Po prebehnutí trénovacieho procesu sa dané vektory príznakov priraďujú k jednotlivým centroidom z kódovej knihy.

2.4.1 K-means algoritmus

K-means je iteračný algoritmus, ktorý vytvára zhluky vektorov (Groups) a z nich potom vypočíta jednotlivé centroidy, tj. geometrické stredy. Ako už bolo spomenuté centroid, keďže je geometrickým stredom, musí mať od všetkých vektorov jedného zhluku minimálnu vzdialenosť danú vzťahom:

$$d(x_k, y_l) = \sqrt{\sum_{k=1}^{p} |x_k - y_l|^2}. \tag{2.38}$$

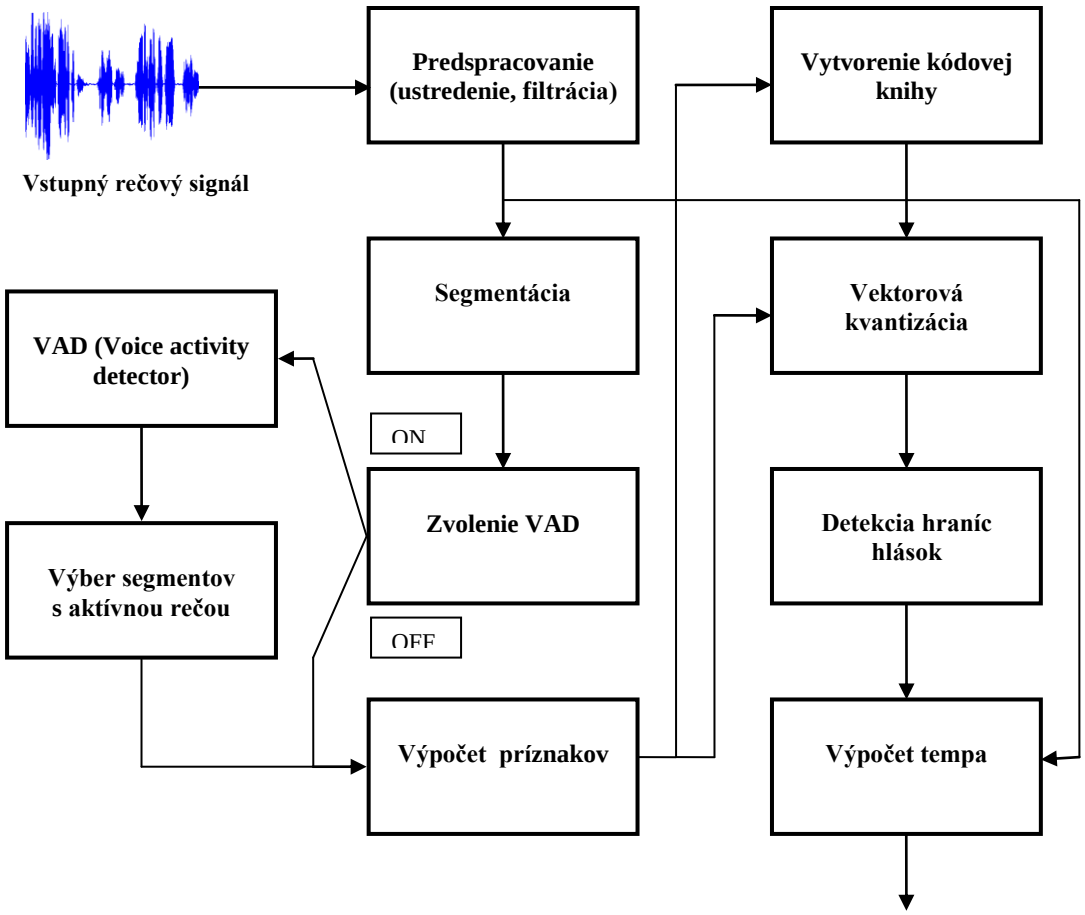
Potom je tento centroid rovný aritmetickému priemeru vektorov. Po natrénovaní kódovej knihy ju k-means prehľadáva a priradzuje jednotlivé vektory príznakov daným centroidom. Každá hláska má svoj centroid a program, ktorý počíta tempo pomocou príznakov vektorovo kvantovaných algoritmom k-means pomocou prechodov priradzovaných vektorov z jedného centroidu na druhý vie rozlíšiť jednotlivé hlásky a teda následne vypočítať tempo. Tu však hrozí chyba. Ako už bolo spomenuté plozivy ako také môžu byť brané ako dve hlásky. Jedna šumová a druhá tzv. explozíva. Tomuto problému čelí algoritmus Fuzzy k-means.

2.4.2 Fuzzy k-means algoritmus

Tento algoritmus na rozdiel od k-means nepriradí vektor danému centroidu priamo, ale pred samotným pridelením prebieha overenie podmienky náležitosti vektoru k tomuto centroidu. Podmienka spočíva v tom, že algoritmus overí s akou pravdepodobnosťou vektor náleží centroidu a ak táto pravdepodobnosť klesne pod určitú hraničnú hodnotu, priradenie neprebehne a tento vektor je priradení centroidu predchádzajúceho vektoru.

3 Návrh systému

3.1 Grafický návrh



Obr. 3.1. : Návrh systému výpočtu tempa reči.

3.2 Popis systému

Systém, ktorý som navrhol sa skladá z desiatich blokov:

1. V prvom bloku predspracovania sa od signálu odčíta jeho stredná hodnota, ktorá neprenáša žiadnu užitočnú informáciu, ktorú by program využíval. Následne sa signál filtruje preemfázovým filtrom, ktorý zdôrazní vysokofrekvenčné zložky, ktoré obsahujú dôležité informácie o danom rečovom signály a v dôsledku prenosu hmotným prostredím bývajú potlačené.

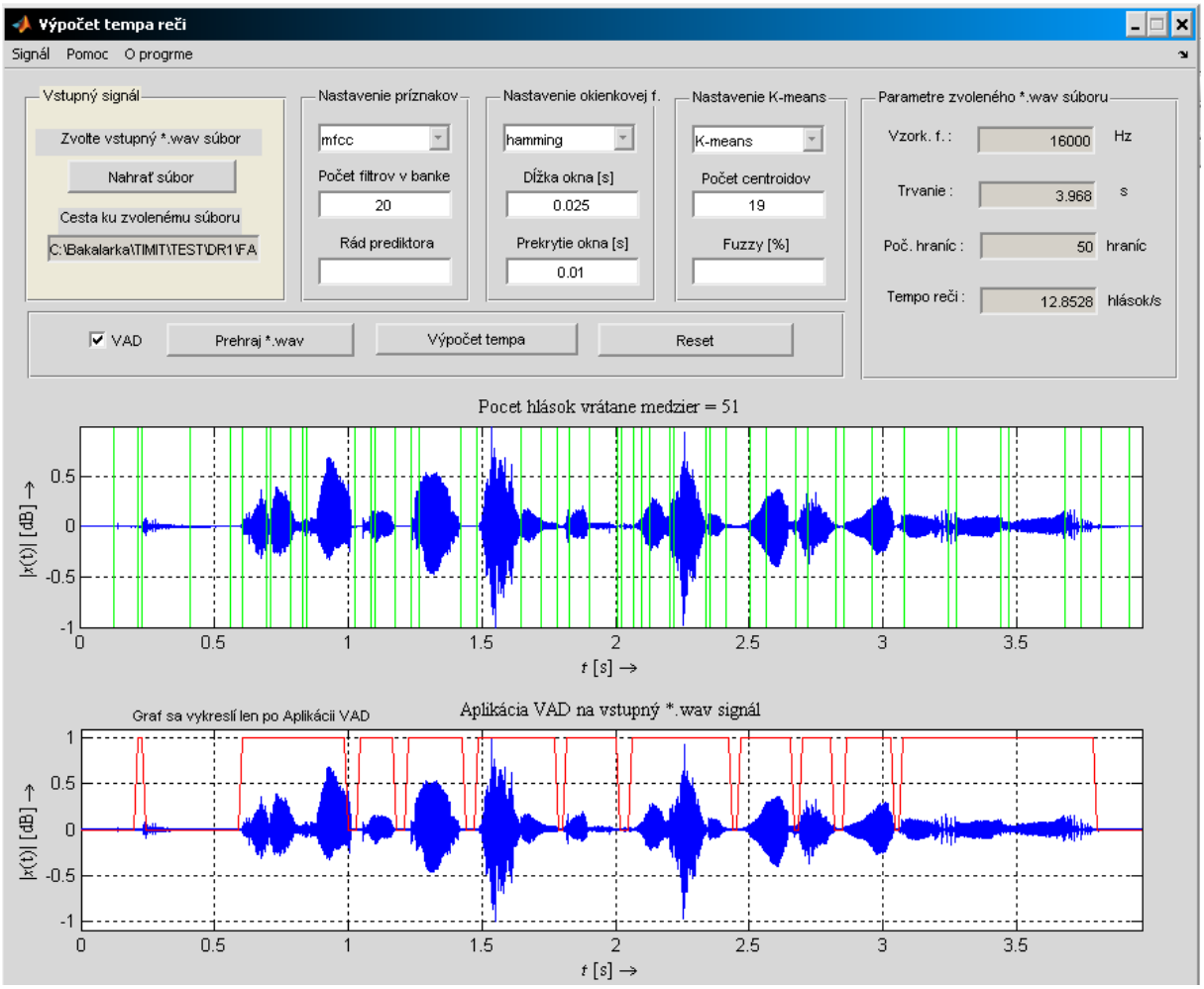
2. V druhom bloku programu sa vykonáva segmentácia rečového signálu na menšie časti, tzv. rámce (segmenty). Program to vykonáva z toho dôvodu, že rečový signál bude spracovávaný krátkodobou analýzou reči, pretože rečový signál je ako taký nestacionárny a táto krátkodobá analýza práve predpokladá, že na dostatočne malých úsekoch (najčastejšie 20 – 30 ms) sa dá považovať za kvázy-stacionárny. Na testovanie programu boli použité segmenty s dĺžkou 25 ms a na ich vytváranie bolo použité Hammingovo okno.
3. V treťom bloku programu sa užívateľ rozhoduje, či bude počítať tempo aj spolu s pauzami, alebo zaradí do výpočtu aj tzv. detektor rečovej aktivity (Voice activity detector), čím by program na základe harmonicity daných segmentov vypustil segmenty v ktorých sa nenachádza reč a tempo by sa ďalej počítalo iba z aktívnej reči.
4. V štvrtom bloku VAD (je aktívny len ak je zvolený tento detektor) pomocou programu praatcon.exe program určí, ktoré segmenty obsahujú pauzy a ktoré nie
5. V piatom bloku (ktorý je taktiež aktívny len pri zvolenom VAD) program vyberie z matice segmentov iba segmenty, ktoré obsahujú aktívnu reč a tie sú ďalej spracovávané pomocou metód krátkodobej analýzy reči.
6. V šiestom bloku Výpočet príznakov sa podľa zvolených parametrov vypočítajú koeficienty (príznaky) pre celú maticu segmentov. Výsledkom je matica príznakov. Príznaky ktoré program počíta sú MFCC, LPC, LPCC, PLP.
7. V siedmom bloku Vytvorenie kódovej knihy sa z matice príznakov „natrénuje“ kódová kniha. To znamená že pre zvolený počet centroidov sa pomocou algoritmu K-means vypočítajú jednotlivé centroidy, teda miesta z najmenšou geometrickou vzdialenosťou ku všetkým bodom z daného zhluku vektorov.
8. V ôsmom bloku Vektorovej kvantizácie program podľa zadanej metódy priraďuje koeficienty krátkodobej analýzy reči jednotlivým centroidom. Pri zvolení algoritmu K-means toto priradenie robí algoritmus K-means sám na základe euklidovskej vzdialenosti. Pri zvolení Fuzzy K-means je možnosť nastaviť pravdepodobnosť pre distribučnú funkciu všetkých vzdialeností daných príznakov k centroidom pod ktorou ak sa táto vzdialenosť nachádza, koeficient bude priradený k tomuto centroidu a ak je táto vzdialenosť väčšia, tak bude priradený centroidu predchádzajúceho príznaku. Touto podmienkou program teoreticky eliminuje chybnú detekciu plozív, tj. hlások, ktoré by sa inak javili ako dve hlásky (jedna šumová, resp. spoluhláska a druhá ako explozíva (samohláska)). Vďaka tejto podmienke program plozivu detekuje ako jednu hlásku. Z toho vyplýva zlepšenie detekcie a zmenšenie chybysamotného výpočtu tempa reči.
9. V deviatom bloku Detekcie hraníc hlások program na základe vektora centroidov (každý segment má priradený jeden centroid) určujú hranice hlások. Program to robí tak, že ak nastane zmena centroidu, nastala aj zmena hlásky. V programe sú ošetrené centroidy, ktoré by sa v skupine vyskytovali samostatne, tj. ak sa v skupine centroidov, napr. skupine troch centroidov, vyskytne jeden iný centroid, je pomocou mediánovej filtrácie vyhladený. Táto úprava je založená na predpoklade, že žiadna hláska netrvá kratšie ako 20 ms. Počet hlások sa potom rovná 1 + počet hraníc detekovaných programom.
10. V desiatom bloku Výpočet tempa program vezme dĺžku trvania spracovávaného rečového signálu (ak bolo zvolené VAD, tak až po prechode týmto blokom) a tento počet hlasok vypočítaný v predchádzajúcom bloku podelí danou dĺžkou trvania. Tým program vypočíta výsledné tempo reči pre zvolený rečový signál.

4 Praktická realizácia

Na spracovanie rečového signálu bol použitý software MATLAB (MATrix LABoratory) verzia 7.9.0 (R2009b), ktorý sa používa na vedecko-technické výpočty, modelovanie, simuláciu a analýzu dát, spracovanie a úpravu signálov atď. Na testovanie vytvoreného programu bola použitá databáza TIMIT.

4.1 Popis užívateľského rozhrania

Program *Výpočet tempa reči* bol vytvorený pre automatickú segmentáciu rečového signálu na hlásky pomocou krátkodobej analýzy reči a vektorovej kvantizácie pre zvolené parametre a následný výpočet tempa pre zvolený rečový signál databáze TIMIT. Program bol implementovaný do grafického užívateľského rozhrania.



Obr. 4.2. : Vytvorené grafické užívateľské rozhranie programu Výpočet tempa reči.

Grafické užívateľské rozhranie sa skladá z niekoľkých častí (podrobnejší popis je uvedený v prílohe v stručnom manuáli k programu). :

1. Poľa vstupných parametrov :
   - Panel pre načítanie vstupného *.wav signálu (rečovej nahrávky z databáze TI-MIT). Po načítaní signálu program do editovateľného poľa vypíše absolútnu cestu k vybranému súboru, v prvom grafe sa zobrazí časový priebeh tohto signálu a v poli parametrov zvoleného *.wav súboru sa zobrazí dĺžka trvania signálu v sekundách a jeho vzorkovacia frekvencia.
   - Panel nastavenia príznakov krátkodobej analýzy reči. Je v ňom možné nastaviť druh koeficientov, s ktorými bude program počítať (MFCC, LPC, LPCC, PLP), počet filtrov v banke a rád prediktora.
   - Panel s nastavením okienkovej funkcie (váhovacieho okna) použitej pri segmentácii. Je tu možné nastaviť štyri typy okienok (Hammingovo, pravouhlé, Bartletovo a Hannovo) a k nim prislúchajúcu dĺžku okna a prekrytie.
   - Panel nastavenia Vektorovej kvantizácie, v ktorom je možnosť zvoliť algoritmus vektorovej kvantizácie (K-means, Fuzzy K-means). Počet centroidov, s ktorými bude program počítať a pri Fuzzy K-means aj parameter „Fuzzy“, tj. percentuálnu hodnotu popísanú v kapitole 3.2 *Popis systému*.

2. Poľa zobrazenia výstupných parametrov, v ktorom ako už bolo spomenuté sa zobrazí dĺžka trvania signálu a jeho vzorkovacia frekvencia po jeho načítaní do programu. Ďalej sú tu ešte dve editovateľné polia, do ktorých sa po výpočte tempa zobrazí počet detekovaných hraníc a výsledné tempo reči pre zvolený signál.

3. Poľa funkčných tlačítok :
   - Zaškrtávacie tlačítko VAD, ktoré ak je zaškrtnuté program počíta tempo len aktívnej reči, tj. pauzy sú z programu vypustené.
   - Tlačítko Prehraj *.wav. Po stlačení program prehrá načítaný rečový signál.
   - Tlačítko Výpočet tempa, ktoré vypočíta tempo daného vstupného signálu pre nastavené parametre.
   - Tlačítko Reset. Slúži na vymazanie načítaného signálu a všetkých vypočítaných parametrov.

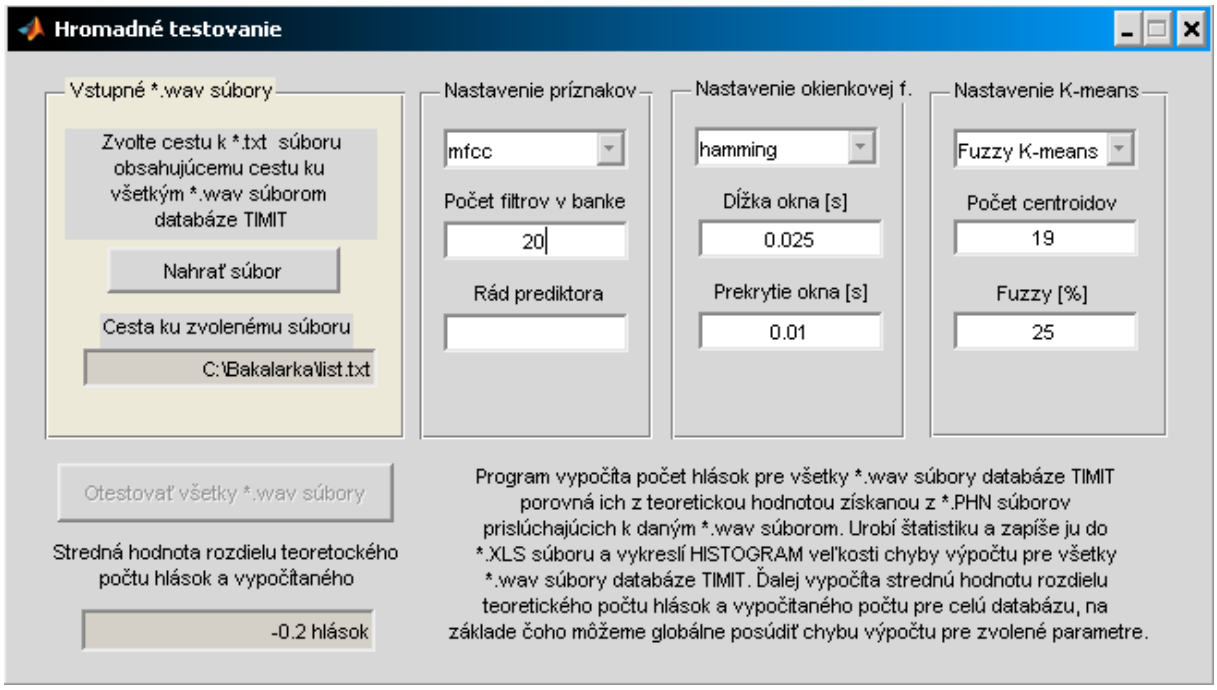
4. Dvoch grafov. Do prvého grafu sa po načítaní vstupného signálu vykreslí jeho časový priebeh. Po výpočte tempa sa dokreslia dané detekované hranice. Druhý graf je aktívny len pri zvolenom VAD a vykreslí (vyznačí) do časového priebehu signálu oblasti s aktívnou rečou.

4.2 Hromadné testovanie

Program bol testovaný na nahrávkach databáze TIMIT. V tejto databáze sa nachádza okolo 5000 rečových nahrávok. Testovanie bolo dôležité kvôli zisteniu a nastaveniu ideálnych parametrov výpočtu v snahe o minimalizáciu chyby výpočtu počtu hlások pre jednotlivé nahrávky a tým aj samotného tempa.

Kvôli zjednodušeniu práce a zefektívneniu testovania bol vytvorený program *Hromadné testovanie*, ktorý otestuje všetky nahrávky danej databázy pre zvolené parametre. Testuje konkrétne počet hlások v jednotlivých nahrávkach a porovnáva to s počtom hlások uvádzaných v súboroch *.PHN, ktoré sú k dispozícii v databáze pre každý *.wav súbor. Program testuje počet hlások a nie tempo reči pre danú nahrávku, kvôli zjednodušeniu a zefektívneniu testovania. Z počtu hlások zisteného pre dané nastavenie je možné jednoduchým výpočtom pri znalosti dĺžky trvania nahrávky toto tempo vypočítať. Pre účel testovania bol vytvorený skript *get_phn_info.m*, ktorý vygeneruje *.MAT súbor *test_info.mat*, ktorý obsahuje štruktúry (pre každú nahrávku jednu),v ktorých sa nachádza absolútna cesta k danej nahrávke a bunku, v ktorej v prvom stĺpci sa nachádzajú počiatočné vzorky jednotlivých hlások, v druhom stĺpci sa nachádzajú koncové vzorky a v treťom aká hláska prislúcha danému riadku. Počet riadkov odpovedá počtu hlások danej nahrávky. V skripte sú vypustené fonetické znaky odpovedajúce pauzám. Program načíta tento *.MAT súbor a pomocou absolútnych ciest z textového súboru *list.txt* načíta všetky rečové nahrávky a vypočíta počet hlások pre všetky tieto nahrávky. Vytvorí ďalší *.MAT súbor *stat.mat*, ktorý bude obsahovať štruktúry pre všetky nahrávky. V každej štruktúre bude absolútna cesta k danej nahrávke, typ, tj. text zložený z nastavených parametrov. Napr. mfcc fuzzy kmeans 25, znamená, že program počítal koeficienty mfcc, ktoré spracoval pomocou Fuzzy K-means s 25% nastavením. Ďalej obsahuje bunku, v ktorej sa v prvom stĺpci nachádza počet hlások nahrávky zistený z *.PHN súboru, v druhom stĺpci je počet hlások vypočítaný programom, v treťom stĺpci je chyba výpočtu v percentách a vo štvrtom stĺpci je rozdiel medzi počtom hlások z *.PHN súboru a vypočítaným počtom hlások. Táto hodnota je pre zistenie ideálnych parametrov pre výpočet kľúčová, pretože ak vypočítame strednú hodnotu týchto odchýlok, vypočítame o koľko hlások sa program pre dané parametre pomýlil. Program následne použije tento *.MAT súbor a a vytvorí výsledný s finálnou štatistikou testovania pre nastavené parametre *final_stat.mat.* Tento súbor obsahuje bunku s troma stĺpcami. V druhom stĺpci bude počet nahrávok, v ktorých bola chyba výpočtu menšia ako 5%, 10%, 15% ... 90%. V treťom stĺpci je koľko percent zo 100 tento počet nahrávok, v ktorých bola chyba výpočtu menšia ako hranica nastavená v tomto riadku, je tento počet. Program po skončení do editovateľného poľa vypíše strednú hodnotu odchýlky spomínanú vyššie a vykreslí histogram zobrazujúci pri koľkých nahrávkach bola aká chyba výpočtu.

Ovládanie programu je veľmi jednoduché. Funguje tak isto ako program Výpočet tempa reči. Nastavenie parametrov je rovnaké. Jediný rozdiel je v tom, že ako vstup sa do programu zadáva textový súbor obsahujúci cesty ku jednotlivým rečovým súborom databáze TIMIT.



Obr. 4.3. : Grafické užívateľské rozhranie programu Hromadné testovanie

5 Výsledky práce

5.1 Vytvorené skripty a funkcie

V bakalárskej práci boli vytvorené dva programy implementované do grafického užívateľského rozhrania. Tieto programy volajú funkcie a skripty, ktoré vykonávajú úkony potrebné pre daný proces a zjednodušujú štruktúru hlavného programu. Výpis všetkých funkcií a skriptov vytvorených pre túto bakalársku prácu :

- **Bark2hz** je funkcia, ktorá prevádza tzv. Barkovu škálu (stupnicu) na škálu v Hertzoch. Využíva sa pri výpočte PLP koeficientov.
- **Barkbanka** je funkcia vytvárajúca banku filtrov v Barkovej stupnici. Taktiež sa využíva pri výpočte koeficientov PLP.
- **Gen_file_list** je spustiteľný *.bat súbor, ktorý prehľadá vyžiadanú oblasť a vypíše zoznam všetkých *.wav súborov. Použité pri testovaní programom Hromadné testovanie na získanie textového dokumentu *list.txt* obsahujúceho absolútnu cestu ku všetkým nahrávkam databáze TIMIT.
- **Gen_timit_info** je skript, ktorého úlohou je načítať textový dokument *list.txt* a pomocou ciest k daným nahrávkam otvoriť *.PHN súbory pre dané nahrávky a pre každú nahrávku pomocou funkcie **get_phn_info** vytvoriť bunku kde v prvom stĺpci

budú jednotlivé fonémy (hlásky) nachádzajúce sa v danej nahrávke, v druhom stĺpci budú počiatočné vzorky a v treťom koncové vzorky danej hlásky. Tohto sa taktiež využije pri testovaní, kde sa vezme počet riadkov v prvom stĺpci a tento počet odpovedá počtu hlások danej nahrávky. Funkcia **get_phn_info** je nastavená tak, aby fonémy označujúce pauzy vynechávali. Z toho vyplýva, že výsledný *.MAT súbor vytvorený skriptom **gen_timit_info** obsahuje počet hlások pre každú nahrávku bez prestávok v reči (páuz).

- **Get_stat** je funkcia používaná pri testovaní. Funkcia si najprv načíta *.MAT súbor *test_info.mat* vytvorený skriptom **gen_file_list.** Ďalej načíta textový dokument *list.txt* pre získanie absolútnych ciest ku všetkým nahrávkam databáze TIMIT. Pre zvolené parametre vypočíta počet hlások pre každú nahrávku a vytvorí *.MAT súbor *stat.mat,* ktorý bude pre každú nahrávku obsahovať štruktúru v ktorej bude cesta k danej nahrávke, text popisujúci zvolené parametre a bunku. V bunke bude počet hlások z *.PHN súboru, vypočítaný počet hlások a ich rozdiel. Tento rozdiel je použitý pri hromadnom testovaní. Stredná hodnota tohto parametra pre všetky nahrávky dáva informáciu o koľko hlások bol daný výpočet nepresný.
- **Get_final_stat** je skript vytvárajúci konečnú štatistiku daného testovacieho cyklu. Skript si najprv načíta *.MAT súbor *stat.mat* vytvoreného funkciou **get_stat** a potom vytvorí *.MAT súbor, ktorý bude obsahovať text popisujúci parametre testovania a bunku. V bunke bude počet nahrávok, ktorých chyba testovania bola pod 5%, 10% atď. Tento údaj bude v druhom stĺpci. V treťom stĺpci bude informácia o tom koľko percent zo 100% je tento počet nahrávok.
- **Get_wav_info** je funkcia používaná funkciou **get_stat**. Jej úlohou je vytvorenie bunky pre každú nahrávku databáze TIMIT.
- **Gui_bakalarka** je program implementovaný do grafického užívateľského rozhrania. Program bol rozobratý vyššie v kapitole *4.1 Popis užívateľského rozhrania*.
- **Gui_final_stat** je taktiež program v grafickom užívateľskom rozhraní vykonávajúci hromadné testovanie databáze TIMIT pre zvolené parametre. Program bol popísaný v kapitole *4.2 Hromadné testovanie*.
- **Hnr_vad** je funkcia vykonávajúca detekciu rečovej aktivity. Využíva konzolovú aplikáciu **praatcon.exe** a používa skript **praat_harmonicity.praat**. Funguje na základe tzv. Harmonicity.
- **Hz2bark** je funkcia použitá pri prevode stupnice v Hertzoch na tzv. Barkovu stupnicu. Používa sa pri výpočte PLP koeficientov.
- **Hz2mel** je funkcia, ktorá prevádza stupnicu v Hertzoch na tzv. Melovskú stupnicu. Využité pri výpočte MFCC koeficientov.
- **Lpc_coef** je funkcia počítajúca Lineárne predikčné koeficienty. Využíva „MATLABOVSKÚ“ funkciu *lpc*.
- **Lpc2cep** je funkcia, ktorá prevádza LPC koeficienty na Kepstrálne LPC (LPCC) koeficienty.
- **Mel2hz** je funkcia prevádzajúca stupnicu v Meloch na stupnicu v Hertzoch. Použité opäť pri koeficientoch MFCC.
- **Melbanka** je funkcia, ktorá vytvorí banku filtrov v Melovskej stupnici.
- **Mfcc** je funkcia počítajúca Mel-frekvenčné kepstrálne koeficienty.

- **Plp** je funkcia počítajúca Perceptívne lineárne predikčné koeficienty.
- **Praat_harmonicity** je funkcia, ktorá podľa zadaných parametrov zostaví konzolové volanie pre aplikáciu *praatcon.exe*.
- **Preemfaza** je funkcia vykonávajúca preemfázovú filtráciu signálu. Používa sa pre každú z použitých metód výpočtu príznakov.
- **Readsph** je funkcia, ktorá načítava vstupný *.wav signál databáze TIMIT. Zdroj : http://www.ee.ic.ac.uk/hp/staff/dmb/voicebox/doc/voicebox/readsph.html
- **Segmentacia** je funkcia segmentujúca signál na rámce (segmenty) zvolenej dĺžky okna a prekrytia a samozrejme typu oknovej funkcie.
- **Ustredenie** je skript odstraňujúci jednosmernú zložku signálu.

5.2 Výsledky hromadného testovania

Hromadné testovanie bolo vykonávané programom *Hromadné testovanie*. Testovala sa celá databáza TIMIT. Pri testovaní sa zisťoval ideálny počet centroidov, pri ktorom by algoritmus vektorovej kvantizácie vykazoval najmenšiu chybovosť. Ďalej sa testoval algoritmus Fuzzy K-means, ktorý bol vytvorený pre túto úlohu. Konkrétne sa testovala hodnota „Fuzzy [%]“, ktorá odpovedala nastaveniu prahu distribučnej funkcie pre všetky vzdialenosti hodnôt vypočítaných koeficientov k daným centroidom. Táto hodnota určovala tzv. prah určenia, tj. ak sa vzdialenosť priradená ako najmenšia k danému centroidu nachádzala pod týmto prahom, vzdialenosť bola priradená tomuto centroidu. Ak bola vzdialenosť väčšia ako tento prah bol tejto hodnote priradený centroid predchádzajúceho segmentu. Z toho vyplýva, že pre každý segment bol vybraný práve jeden centroid a o priraďovaní týchto centroidov rozhodovala práve táto hodnota nastavená parametrom „Fuzzy“.

Ďalšie hodnoty, ktoré bolo možné nastavovať boli pri hromadnom testovaní prednastavené na hodnoty :

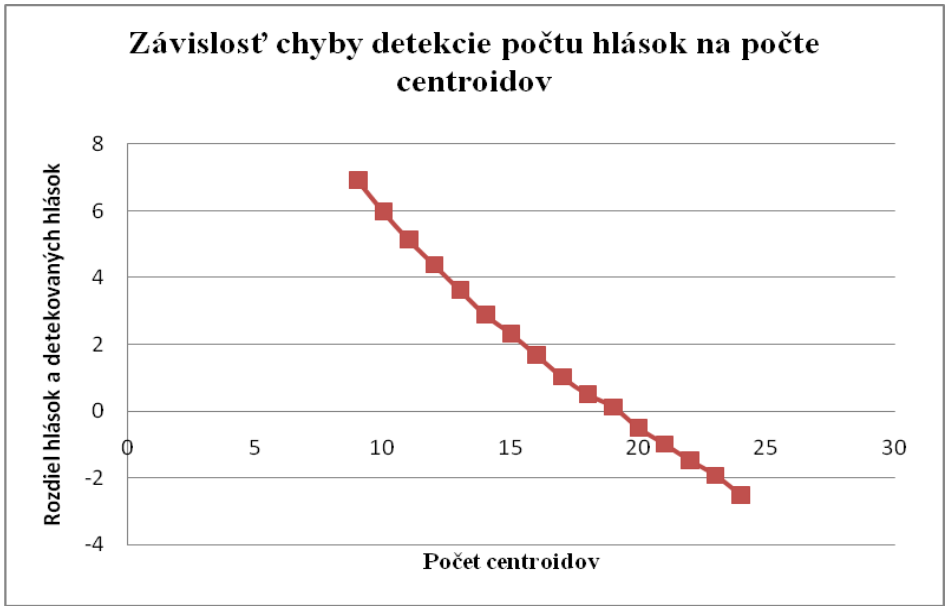
- Počet filtrov v banke = 20.
- Rád prediktora (iba pri prediktívnej analýze) = 14.
- Typ okienkovej funkcie (váhovacieho okna) = Hammingovo okno.
- Dĺžka váhovacieho okna = 25ms.
- Prekrytie váhovacích okien = 10ms.

5.2.1 Testovanie počtu centroidov algoritmu K-means

Testovanie spočívalo v nájdení najvyhovujúcejšieho počtu centroidov pre algoritmus vektorovej kvantizácie K-means, pri ktorom by bol rozdiel hlások z *.PHN súboru daných nahrávok (odpovedajúci skutočnému počtu hlások) a počtu hlások vypočítaných programom čo najmenší.

Tab. 5.1. : Tabuľka závislosti rozdielu skutočného počtu hlások v nahrávkach databáze TIMIT a vypočítaného počtu hlások pre koeficienty MFCC na počte centroidov zvolených pre algoritmus K-means.

Počet centroidov	Chyba [hlások]
9	6,9172
10	5,977
11	5,1283
12	4,3797
13	3,6261
14	2,8808
15	2,3176
16	1,6755
17	1,0234
18	0,5
19	0,1265
20	-0,50744
21	-0,97985
22	-1,474
23	-1,9178
24	-2,5167



Obr. 5.1. : Závislosť rozdielu skutočného počtu hlások v nahrávkach databáze TIMIT a vypočítaného počtu hlások pre koeficienty MFCC na počte centroidov zvolených pre algoritmus K-means.

Z tabuľky aj z grafu je vidno, že najmenšej chyby sa algoritmus dopustil pri nastavení 19 centroidov. Pri prekročení tejto hodnoty už algoritmus detekoval viac hlások ako ich v daných

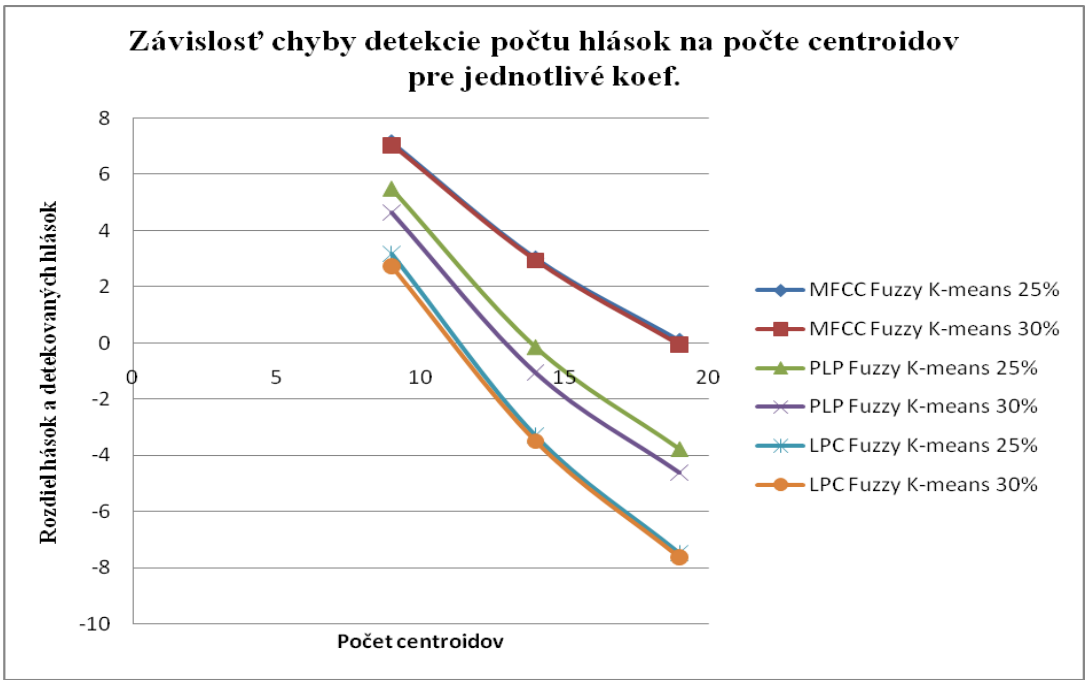
nahrávkach naozaj bolo. Ďalším zvyšovaním počtu centroidov by teda hodnota rozdielu rástla smerom do záporných hodnôt. Táto hodnota počtu centroidov samozrejme nie je definitívna. Pre iné koeficienty krátkodobej analýzy reči by sa ideálny počet centroidov mohol líšiť.

5.2.2 Testovanie algoritmu Fuzzy K-means

Pri testovaní algoritmu Fuzzy K-means sa testovala závislosť rozdiel skutočného počtu hlások v nahrávkach databáze TIMIT a vypočítaného počtu hlások pre jednotlivé koeficienty krátkodobej analýzy reči na nastavení parametra „Fuzzy“, ktorého funkcia bola popísaná v kapitole *5.2 Výsledky hromadného testovania* a taktiež závislosť tohto rozdielu hlások na zvolenom počte centroidov pre algoritmus Fuzzy K-means.

Tab. 5.2. : Tabuľka závislosti rozdielu skutočného počtu hlások a vypočítaného počtu hlások pre jednotlivé koeficienty krátkodobej analýzy reči na nastavení parametra „Fuzzy“ a závislosti tohto rozdielu hlások na zvolenom počte centroidov pre algoritmus Fuzzy K-means.

Koeficienty	Kvantizácia	Počet centroidov	Perc. h. fuzzy	Chyba [hlások]
MFCC	Fuzzy K-means	9	25	7,1543
MFCC	Fuzzy K-means	9	30	7,066
MFCC	Fuzzy K-means	14	25	3,0181
MFCC	Fuzzy K-means	14	30	2,9544
MFCC	Fuzzy K-means	19	25	0,082396
MFCC	Fuzzy K-means	19	30	-0,047913
PLP	Fuzzy K-means	9	25	5,4991
PLP	Fuzzy K-means	9	30	4,6468
PLP	Fuzzy K-means	14	25	-0,14356
PLP	Fuzzy K-means	14	30	-1,0639
PLP	Fuzzy K-means	19	25	-3,7728
PLP	Fuzzy K-means	19	30	-4,6247
LPC	Fuzzy K-means	9	25	3,1837
LPC	Fuzzy K-means	9	30	2,7361
LPC	Fuzzy K-means	14	25	-3,29
LPC	Fuzzy K-means	14	30	-3,4944
LPC	Fuzzy K-means	19	25	-7,4702
LPC	Fuzzy K-means	19	30	-7,6328
LPCC	Fuzzy K-means	9	25	7,1096
LPCC	Fuzzy K-means	9	30	7,0408
LPCC	Fuzzy K-means	14	25	3,0503
LPCC	Fuzzy K-means	14	30	3,0762
LPCC	Fuzzy K-means	19	25	0,063521
LPCC	Fuzzy K-means	19	30	0,13321



Obr. 5.2. : Závislosť rozdielu skutočného počtu hlások a vypočítaného počtu hlások pre jednotlivé koeficienty krátkodobej analýzy reči na nastavení parametra „Fuzzy“ a závislosť tohto rozdielu hlások na zvolenom počte centroidov pre algoritmus Fuzzy K-means.

Koeficienty LPCC neboli vynesené do grafu, pretože ich hodnoty boli takmer rovnaké ako hodnoty získané pri testovaní koeficientov MFCC. Z testov vyplýva, že nastavenie ideálneho počtu centroidov je potrebné pre každý príznak krátkodobej analýzy urobiť zvlášť. Z grafu je vidno, že napr. ideálny počet centroidov pre koeficienty MFCC zistený v kapitole *5.2.1 Testovanie počtu centroidov algoritmu K-means* (19 centroidov) je napr. pri analýze LPC už príliš veľký. Odhadom by sa dalo povedať, že pre analýzu LPC by ideálny počet centroidov ležal niekde v rozmedzí 10 až 13 centroidov. Ďalej z grafu vidno, že pre analýzu PLP je 14 centroidov, ktoré boli testované takmer ideálna hodnota vykazujúca chybu menšiu ako jedna hláska.

Parameter „Fuzzy“, ako je vidno z grafu, spôsobuje to, že pri jeho nastavení na väčšiu hodnotu program detekuje viac hlások. Pri určitej hraničnej hodnote by sa z neho stal algoritmus K-means, pretože by priradil každú minimálnu vzdialenosť. Z toho vyplýva, že nastavenie tohto parametra má veľký vplyv na presnosť detekcie hraníc hlások, teda aj samotného tempa reči.

5.2.2 Priebeh testovania

Príklad Hromadného testovania pre koeficienty MFCC, Fuzzy K-means a parameter „Fuzzy“ = 25%. Najprv program vypočítal počet hlások pre všetkých nahrávok databáze TIMIT pre zvolené parametre a uložil ich do *.MAT súboru. Potom Z tohto súboru vypočítal výslednú štatistiku testovacieho cyklu a vykreslil Histogram počtu *.wav súborov pre jednotlivé chyby výpočtu.

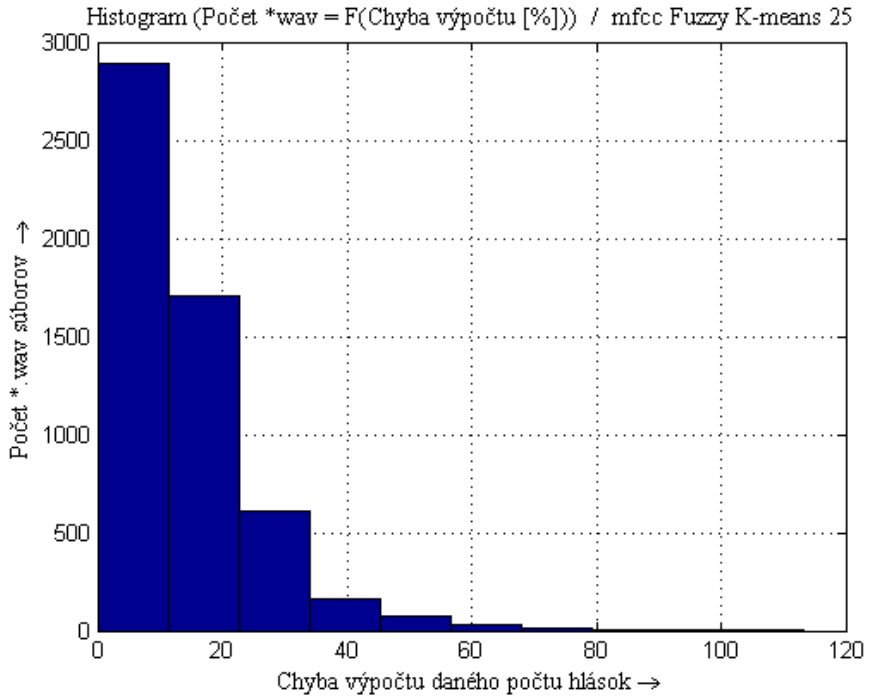
Tab. 5.3. : Tabuľka zobrazujúca vytvorenie štatistiky pre prvých 29 nahrávok databáze TIMIT testovanej programom pre výpočet koeficientov MFCC. Na vektorovú kvantizáciu bol použitý algoritmus Fuzzy K-means. Nastavenie parametra „Fuzzy“ bolo 25%.

n.	Počet hlások	Vypočítaný počet hlások	Chyba výpočtu	Rozdiel [hlások]
1.	38	43	13,15789474	-5
2.	31	40	29,0323	-9
3.	56	49	12,5	7
4.	29	32	10,3448	-3
5.	37	35	5,4054	2
6.	36	34	5,5556	2
7.	30	25	16,6667	5
8.	35	37	5,7143	-2
9.	38	43	13,1579	-5
10.	33	24	27,2727	9
11.	37	42	13,5135	-5
12.	30	40	33,3333	-10
13.	32	40	25	-8
14.	29	46	58,6208	-17
15.	68	65	4,4118	3
16.	28	27	3,5714	1
17.	41	49	19,5112	-8
18.	31	40	29,0302	-9
19.	23	34	47,8251	-9
20.	27	33	22,2222	-5
21.	39	48	23,0679	-9
22.	35	36	2,8571	-1
23.	65	62	4,6154	3
24.	31	36	16,129	-5
25.	41	50	21,9512	´-9
26.	28	29	3,5714	-1
27.	26	24	7,6924	2
28.	27	37	37,037	-10
29.	45	44	2,2222	1

Z tabuľky je vidno, že program počíta daný počet hlások s určitou nepresnosťou, ktorej veľkosť vyjadruje jednak ako chybu výpočtu (v percentách) a v počte hlások o ktoré sa program pomýlil. V ďalšej tabuľke bude uvedený príklad testovania pre koeficienty MFCC, algoritmus Fuzy K-means a „Fuzzy“ = 25%. Bude uvedený aj Histogram vyjadrujúci počet *.Wav súborov (nahrávok), pre ktoré vznikla daná chyba výpočtu.

Tab. 5.4. : Tabuľka vyjadrujúca konečnú štatistiku testovacieho cyklu pre koeficienty MFCC, algoritmus Fuzzy K-means a „Fuzzy“ = 25%. Testovanie sa vykonávalo pre všetky nahrávky databáze TIMIT.

Chyba výpočtu	Počet nahrávok	Percent (zo 100)
< 5%	1473	26,73321234
< 10%	2673	48,51179673
< 15%	3702	67,18693285
< 20%	4370	79,31034483
< 25%	4830	87,65880218
< 30%	5082	92,2323049
< 35%	5247	95,22686025
< 40%	5331	96,75136116
< 45%	5386	97,74954628
< 50%	5431	98,56624319
< 55%	5449	98,89292196
< 60%	5466	99,20145191
< 65%	5479	99,43738657
< 70%	5489	99,61887477
< 75%	5495	99,7277677
< 80%	5502	99,85480944
< 85%	5503	99,87295826
< 90%	5505	99,9092559



Obr. 5.3. : Histogram testovacieho cyklu pre koeficienty MFCC, algoritmus Fuzzy K-means a „Fuzzy“ = 25%

Z tabuľky je vidno, že program pre zadané parametre dosiahol chybovosť výpočtu menšiu ako 20% vo vyše 79% nahrávok, čo je vidno aj v grafe (Histograme).

5.3 Zhrnutie dosiahnutých výsledkov

Bol vytvorený program *Výpočet tempa reči*, ktorý vypočíta tempo reči pre zvolenú nahrávku databáze TIMIT. Konkrétny príklad výpočtu tempa pomocou tohto programu sa nachádza v prílohe v dokumente *help.pdf*. Je to dokument slúžiaci ako stručný návod k programu, v ktorom sú vysvetlené funkcie jeho jednotlivých častí a ich použitie.

Ďalej bol vytvorený program Hromadné testovanie, ktorý testuje celú databázu TIMIT pre zvolené parametre. Testuje konkrétne počet hlások v daných nahrávkach a porovnáva ich so skutočným počtom hlások. Testovanie počtu hlások namiesto tempa bolo vykonávané kvôli väčšej prehľadnosti testovania. Z uvedeného počtu hlások a dĺžky trvania danej nahrávky je možné zistiť tempo reči. Tento výpočet vykonáva program *Výpočet tempa reči* pre jeden daný testovaný *.wav súbor.

Cieľom práce bolo navrhnúť program pre výpočet tempa reči a otestovať jednotlivé parametre krátkodobej analýzy reči a ďalšie parametre, ktoré je možné v programe meniť. Niektoré parametre boli zvolené empiricky ako napr. typ váhovacej funkcie, dĺžka okna a prekrytie. Ďalej bol prednastavený počet filtrov v banke a rád prediktora s ohľadom na čo najpresnejší výsledok a zároveň akceptovateľné zaťaženie výpočtu. Testovanie sa vykonávalo najmä pre tzv. vektorovú kvantizáciu, kde bol najprv testovaný počet centroidov na úspešnosť výpočtu, resp. minimalizáciu chyby výpočtu počtu hlások. Bolo zistené, že pre rôzne príznaky sa tento počet líši. Pre koeficienty MFCC a LPCC je tento počet 19 centroidov. Rovnaká hodnota nasvedčuje ich povahe kepstrálnych koeficientov. Pre koeficienty LPC sa táto hodnota nachádza niekde medzi 10 až 13 centroidmi a pre koeficienty PLP je ideálny počet centroidov 14. Testovanie dokazuje, že pre dlhšie slová až vety je potrebný väčší počet centroidov. Vyplýva to z úvahy, že dlhšie slovo alebo veta obsahuje väčší počet rôznych hlások a keďže zmena hlásky je reprezentovaná prechodom skupiny centroidov na inú skupinu, aj počet centroidov by mal rásť s počtom hlások. Počtom hlások sa v tejto úvahe nemyslí celkový počet hlások vo vete ale rôznosť hlások vo vete (napr. slovo *mama* obsahuje len dve hlásky ale v sove sú použité dva krát, teda celkovo je počet hlások v slove 4. V tejto úvahe by nás zaujímal prvý údaj). Testoval sa ďalej algoritmus Fuzzy K-means, konkrétne jeho parameter „Fuzzy“. Testovanie prebehlo pre dve hodnoty 25 a 30 percent. Menšie hodnoty by boli už moc nízke a nastavovali by prah pre priradenie centroidov tak nízko, že by algoritmus nepriraďoval niektoré centroidy aj keď by mal. Vyššie hodnoty by zas posunuli prah na príliš benevolentnú hodnotu, čo sa prejavilo už aj pri nastavení na 30%. Je vidno, že čím vyššie by sme nastavili hodnotu „Fuzzy“, tým viac hlások by program detekoval. Pri nastavení určitej prahovej hodnoty by tento prah už nebol podstatný pretože každá minimálna vzdialenosť by bola pod týmto prahom.

5.4 Spustenie testovania

Pre správne spustenie a funkčnosť vytvorených programov pri prvom spustení alebo pri prenesení programu alebo databáze na iné miesto na disku je treba dodržať niekoľko zásad :

- Všetky skripty, funkcie atď. , teda celý obsah priečinku ZDROJOVÉ KÓDY, ktorý sa nachádza na priloženom DVD je potrebné mať v rovnakom priečinku ako databázu TIMIT, teda oba priečinky (zložky) ZDROJOVÉ KÓDY a TIMIT by mali byť v rovnakej zložke.

- Pred spustením testovania je pomocou programu. Gen_file_list.bat potrebné vygenerovať nový textový list.txt súbor, v ktorom budú aktuálne absolútne cesty k testovaným nahrávkam databáze.

- Po vytvorení tohto dokunentu je potrebné pomocou skriptu get_timit_info vytvoriť nový *.MAT súbor test_info.mat.

6 Záver

Pre výpočet tempa reči bol v prostredí MATLAB vytvorený program *Výpočet tempa reči,* ktorý bol vytvorený na základe štruktúry vytvorenej v kapitole 3.1 *Grafický návrh* Program sa skladá z viacerých funkčných bokov, ktorých funkcia je popísaná v kapitole 3.2 *Popis systému.* Program bol kvôli zjednodušeniu práce implementovaný do grafického užívateľského rozhrania. K programu bol vytvorený stručný návod, ktorý je priložený v prílohe.

Ďalej bol vytvorený program *Hromadné testovanie*, ktorý slúži na testovanie všetkých nahrávok databáze TIMIT. Tento program bol taktiež implementovaný do grafického užívateľského rozhrania. Testovanie programom Hromadné testovanie bolo potrebné kvôli zisteniu najlepšej kombinácie vstupných parametrov (sú rovnaké ako v programe *Výpočet tempa reči*), pri ktorých by testovanie vykazovalo najmenšiu chybovosť detekcie hraníc hlások, teda v konečnom dôsledku aj samotného výpočtu tempa reči. Testované boli všetky nahrávky databáze TIMIT, teda v jednom testovacom cykle sa na danú kombináciu vstupných parametrov testovalo vyše 5000 rečových nahrávok. Ďalším prínosom testovania bolo nastavenie týchto „ideálnych" hodnôt parametrov ako prednastavené hodnoty, teda ak užívateľ nezadá veľkosť daného parametra program použije túto prednastavenú hodnotu.

Pri testovaní sa niektoré hodnoty, resp. ich hodnota určili empiricky, teda použila sa hodnota, ktorá je známa ako hodnota, pri ktorej je chybovosť výpočtu minimálna. Príkladom môže byť napr. použitie Hammingovho okna pri segmentácii reči na rámce ako aj ich samotná dĺžka 25 ms. Testovalo sa nastavenie vektorovej kvantizácie, resp. počet centroidov a ideálne nastavenie parametra „Fuzzy" pre jednotlivé príznaky krátkodobej analýzy reči. Testovanie počtu centroidov prebiehalo pre príznaky MFCC v rozmedzí od 9 do 24 centroidov. Testy ukázali na fakt, že pri testovaní dlhších nahrávok aké obsahuje databáza TIMIT, teda tetovanie slovných spojení až viet je pre minimalizáciu chyby potrebný väčší počet centroidov. Pre tento test ako ideálna hodnota vyšla hodnota 19 centroidov. Z testov teda vyplýva aj fakt, že počet centroidov by teoreticky mohol odpovedať počtu rôznych hlások v nahrávke. Táto úvaha sa opiera o fakt, že počet hlások v danom rečovom signály program počítal ako počet zmien skupín centroidov plus jedna. Testovanie vektorovej kvantizácie Fuzzy K-means overovalo zmenu presnosti detekcie na nastavení parametra „Fuzzy", ktorého funkcia bola v texte vysvetlená viac krát. Testy ukázali, že nastavenie tohto parametra na nižšiu hodnotu zmenší počet detekovaných hraníc, pretože sa testuje daná minimálna vzdialenosť, ktorá sa má priradiť centroidu, resp. je porovnávaná s hranicou nastavenou podľa veľkosti paramentra „Fuzzy". Čím menšia je hodnota „Fuzzy", tým menšia je aj veľkosť hranice. Ak je táto hodnota nastavená príliš vysoko nastane situácia, že táto hranica bude strácať význam, pretože väčšina minimálnych vzdialeností bude menšia ako hranica. Testovali sa dve hodnoty hranice 25% a 30%. Ukázalo sa, že hodnota 30% už bola mierne benevolentná. Výsledky testovania boli vynesené do grafov.

Ak by vytvorené programy mali byť používané v praxi, bolo by potreba vykonať podrobnejšie a rozsiahlejšie testy aby sa chyba testovania zmenšila na úplné minimum. Ďalej by bolo potrebné upraviť zobrazenie detekovaných hraníc v programe *Výpočet tempa reči* pri použití VAD, pretože program vykresľuje hranice, ktoré sú počítané iba pre aktívnu reč, vo „waveforme", teda vykreslení signálu v čase, pre nahrávku, ktorá tieto pauzy odstránené nemá.

Literatúra

[1] ČERNOCKÝ, J. *Zpracování řečových signálů*. Ústav počítačové grafiky a multimédií, FIT, VUT Brno, 2006.

[2] *Fonetika* [online]. Dostupné z http://www.phil.muni.cz/jazyk/krcmova/fon/ucebnitext/5.htm.

.

[3] HAUTAMÄKI, V. et al. *Improving Speaker Verification by Periodicity Based Voice Activity Detection*. In: Proc. 12th Internat. Conf. on Speech and Computer (SPECOM 2007), Moskva, Rusko, 2007. s 645–650.

[4] HOLČÍK, J. *Analýza a klasifikace signálů*. Brno: VUT, 1992, ISBN 80-214-0450-7.

[5] JUHÁR, J. *Spracovanie signálov v systémoch automatického rozpoznávania reči*. Habilitačná práca, Katedra elektrotechniky a multimediálnych telekomunikácií, FEI, Technická univerzita v Košiciach, 1999.

[6] KRČMOVÁ, M. *Fonetika*. Elektronické texty. Masarykova Univerzita, Brno, 2003 http://is.muni.cz/do/1499/el/estud/ff/js07/fonetika/materialy/index.html

[7] PSUTKA, J. *Komunikace s počítačem mluvenou řečí*. 1. vydání. Praha: Academia, 1995. 287 s. ISBN 80-200-0203-0.

[8] PSUTKA, J. et. al. *Mluvíme s počítačem česky*. 1. vydání. Praha: Academia, 2006. 752 s. ISBN 80-200-1309-0

[9] SIGMUND, M. *Analýza řečových signálů*. Brno: VUT, 2000, ISBN 80-214-1783-8

[10] SMÉKAL, Z. *Číslicové zpracování řeči* (MZPR). Elektronická skripta pro magisterská studia, Ústav telekomunikací, FEEC, VUT Brno, 2009.

[11] SMÉKAL, Z., a SYSEL P. *Číslicové filtry* (BCIF). Elektronická skripta pro bakalářská studia, Ústav telekomunikací, FEEC, VUT Brno, 2004.

[12] SMÉKAL, Z. *Číslicové zpracování signálů* (MCSI). Elektronická skripta pro magisterská studia, Ústav telekomunikací, FEEC, VUT Brno, 2009.

[13] SMÉKAL, Z., a Šebesta V. *Signály a soustavy* (BASS). Elektronická skripta pro bakalářská studia, Ústav telekomunikací], FEEC, VUT Brno, 2010.

Zoznam použitých skratiek a symbolov

DCT	diskrétna kosínova transformácia
DFT	diskrétna Fourierova transformácia
DTFT	Fourierova transformácia diskrétneho času
FFT	rýchla Fourierova transformácia
HMF	homomorfná transformácia
HP	filter typu horná priepusť
IDFFT	spätná rýchla Fourierova transformácia
IDFT	inverzná diskrétna Fourierova transformácia
KT	kepstrálna transformácia
LBG	Linde-Buzo-Gray algoritmus
LF	lineárny filter
LPC	lineárna predikcia
LPCC	lineárne predikčné kepstrálne koeficienty
LTI	lineárny časovo nepremenný systém
MFCC	mel-frekvenčné kepstrálne koeficienty
PCM	pulzná kódová modulácia
PLP	perceptívna lineárna predikcia
VAD	voice activity detector

a_p	koeficient LPC
a_{pre}	koeficient preemfázového filtra
$A(z)$	prenosová funkcia analyzujúceho filtra
B_m	šírka prenosového pásma filtra
$b_{m,i}$	stredná frekvencia filtra v Melovskej škále
$c(n)$	n-tý LPCC koeficient
E	normalizovaná chyba predikcie
$e[n]$	chyba predikcie
e	Eulerovo číslo $\left(\mathrm{e} = \sum_{n=0}^{\infty} \frac{1}{n!} = 2.71828...\right)$

F_0 základný hlasivkový tón [Hz]

F_i formantové frekvencie [Hz]

$F(z)$ obraz signálu $f[n]$ podľa Z-transformácie

$f[n]$ vstupný systém

f_{Hz} frekvencia v Hertzoch

f_{Mel} frekvencia v Meloch

f_{vz} vzorkovacia frekvencia

$\hat{G}_{\mathrm{LPC}}(f)$ spektrálna hustota rečového signálu (jeho segmentu)

g_0 budiaci signál

$\hat{g}(m)$ komplexné kepstrum budiaceho signálu

$h[n]$ impulzná odozva

$\hat{h}(m)$ komplexné kepstrum impulzovej odpovede hlasového traktu

$H(\mathrm{e}^{\mathrm{j}\omega})$ frekvenčná charakteristika

$H(z)$ prenosová funkcia syntetizačného filtra

$H_{pre}(z)$ prenosová funkcia preemfázového filtra

j imaginárna jednotka

I_{ram} dĺžka rámca

ln prirodzený logaritmus

$\ln\left|X(e^{\mathrm{j}\Omega})\right|$ modulové kepstrum

N dĺžka segmentu

N_{FFT} dĺžka FFT (počet vzorkov)

N_{ram} počet rámcov v jednom segmente

p rád predikcie

p_{ram} dĺžka prekrytia rámcov

$P(\Omega)$ výkonové spektrum rečového signálu (jeho segmentu)

$s[n]$ rečový signál

$s'[n]$ signál po odstredení

$\bar{s}[n]$ vzorka odhadnutá pomocou doprednej LPC

s_{ram} dĺžka neprekrytého úseku rámca

$\hat{s}(m)$ komplexné kepstrum výstupného signálu

T_0 perióda jednotkových impulzov

$w[n]$ hodnota vzorku signálu po násobení oknom

X vektor príznakov

$X(z)$ obraz signálu $x[n]$ podľa Z-transformácie

$\hat{X}(z)$ prirodzený logaritmus zo signálu $X(z)$

$x(n)$ signál po použití inverznej kepstrálnej transformácie na signál $\hat{X}(z)$

$\hat{x}[m]$ komplexné kepstrum

$\hat{x}_c[m]$ reálne kepstrum

$\hat{x}_{phase}[m]$ fázové kepstrum

Y vektor z kódovej knihy

ω uhlová frekvencia [$\mathrm{rad} \cdot \mathrm{s}^{-1}$]

Ω_B transformačný člen [Bark]

$\Psi(z)$ prenosová funkcia filtru typu pásmová priepusť z banky filtrov

$\Psi(\Omega_B)$ výkonové spektrum banky filtrov aprox. maskovacie krivky ľudského sluchu

μ_s odhadnutá stredná hodnota

π Ludolphovo číslo $\left(\pi = 4\sum_{n=0}^{\infty} \frac{(-1)^n}{2n+1} = 3.14159... \right)$

γ_p korelačné koeficienty

Obsah priloženého DVD

Priložený disk DVD obsahuje niekoľko priečinkov :

- Priečinok **DOKUMENTY** obsahuje *.pdf súbory.
  - Bakalárska práca Zoltán Galáž.pdf
  - Manuál.pdf
  - O programe.pdf

- Priečinok **TESTOVANIE** obsahuje zložky, v ktorých v každej sa nachádzajú *.MAT súbory s dátami z daného testovacieho cyklu pre nastavené parametre. Parametre sú jasné z názvu zložky. Ďalej sa v zložke nachádza *.png súbor, ktorý je Histogramom počtu testovaných nahrávok z databáze TIMIT, pre ktoré sa vyskytla daná chyba testovania. Obsahuje ešte textový súbor so strednou hodnotou rozdielu počtu hlások z *.PHN súborov a vypočítaného počtu hlások daného testovacieho cyklu pre nastavené parametre testovania.

- Priečinok **TIMIT**. Obsahuje databázu TIMIT, ktorá slúžila ako testovacia databáza pre vytvorené programy.

- Priečinok **ZDROJOVÉ KÓDY**. Jeho obsahom budú všetky *.m a *.fig súbory, tj. Všetky použité skripty a funkcie v bakalárskej práci. V tomto priečinku sa nachádzajú programy *gui_bakalarka.m,* čo odpovedá programu Výpočet tempa reči a *gui_final_stat.m,* čo odpovedá programu Hromadné testovanie. Priečinok ďalej obsahuje textový dokument *list.txt*, spustiteľný súbor *gen_file_list.bat,* program *praatcon.exe,* súbor *praat_harmonicity.praat* a *.MAT súbor *test_info.mat*, ktorých funkcia bola vysvetlená v kapitole 4.2 *Hromadné testovanie* a 5.1 *Vytvorené skripty a funkcie*.