SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 1154

METODE RAZLIKOVANJA GOVORA I GLAZBE U DIGITALNIM ZVUČNIM ZAPISIMA

Ivan Križanić

Zagreb, lipanj 2015

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA
ODBOR ZA DIPLOMSKI RAD PROFILA

Zagreb, 6. ožujka 2015.

DIPLOMSKI ZADATAK br. 1154

Pristupnik: **Ivan Križanić (0036456477)**
Studij: Elektrotehnika i informacijska tehnologija
Profil: Elektroničko i računalno inženjerstvo

Zadatak: **Metode razlikovanja govora i glazbe u digitalnim zvučnim zapisima**

Opis zadatka:

U okviru diplomskog rada potrebno je proučiti metode automatske identifikacije i odvajanja govora od glazbe unutar zadane snimke. Na temelju javno dostupnih anotiranih snimki potrebno je provesti testiranje postojećih algoritama i ustvrditi njihovu točnost i preciznost. Govor i glazba imaju različita svojstva u vremenskoj, spektralnoj i kepstralnoj domeni. U okviru rada potrebno je identificirati pogodne značajke govora i glazbe u sve tri domene, koje se mogu koristiti u svrhu diskriminacije govora i glazbe. Usporediti najčešće korištene statističke modele za klasifikaciju, (npr. model s Gaussovim mješavinama) s drugim modelima. Diskutirati metode određivanja početka i kraja riječi, u ovisnosti o žanru glazbe i vrsti izgovora. Odabrane algoritme implementirati u Matlabu, a one koje je moguće prevest u C programski jezik i prilagoditi izvedbi na DSP procesoru.

Zadatak uručen pristupniku: 13. ožujka 2015.
Rok za predaju rada: 30. lipnja 2015.

Mentor:
Prof. dr. sc. Davor Petrinović

Djelovođa:
Izv. prof. dr. sc. Dražen Jurišić

Predsjednik odbora za diplomski rad profila:
Prof. dr. sc. Mladen Vučić

Sadržaj

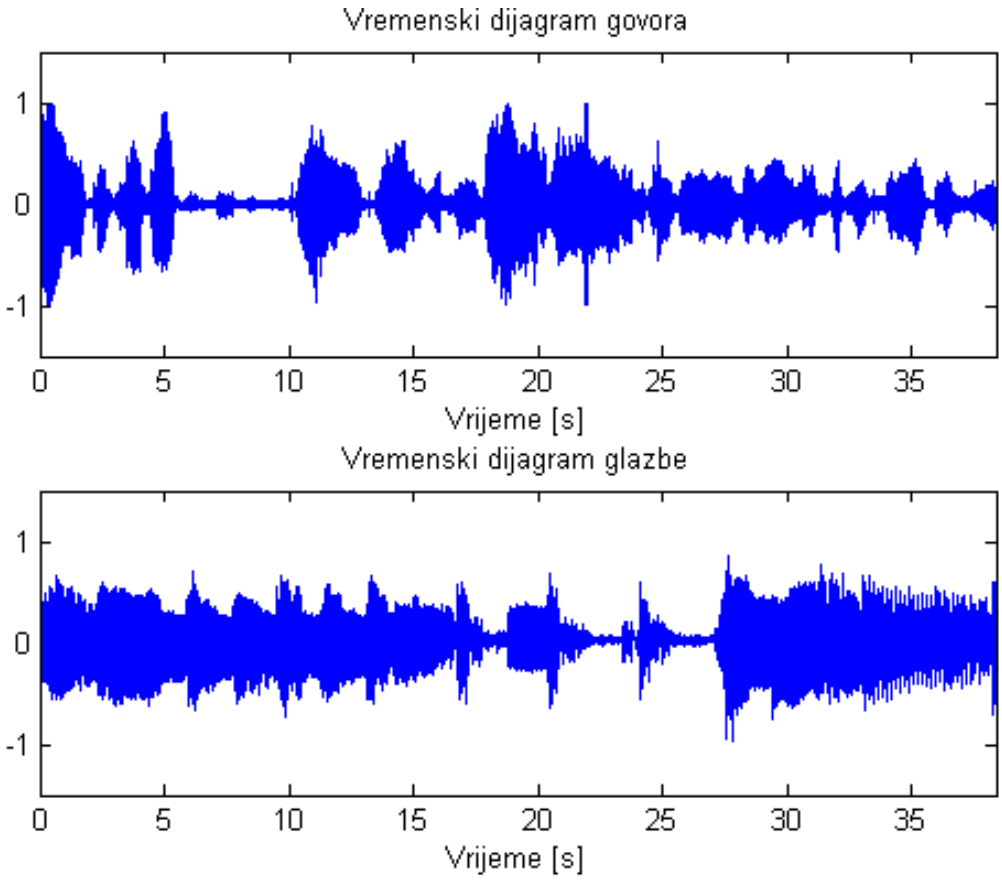
Uvod

Ubrzanim napretkom računala strojno razlikovanje govora i glazbe te klasifikacija žanrova glazbe postali su široko područje interesa unutar digitalne obradbe signala. Riječ je o jednom od najvažnijih problema za odabir prikladne reprezentacije signala i optimalno kodiranje podataka s obzirom na to što oni predstavljaju i kakva je njihova namjena.

Tijekom godina brojne su se metode koristile za više ili manje uspješnu klasifikaciju. Cilj je ovog rada približiti načine na koji se može obavljati strojna klasifikacija kroz značajke zvuka i klasifikatore, ali i predstaviti neke gotove javno dostupne metode.

1. Motivacija

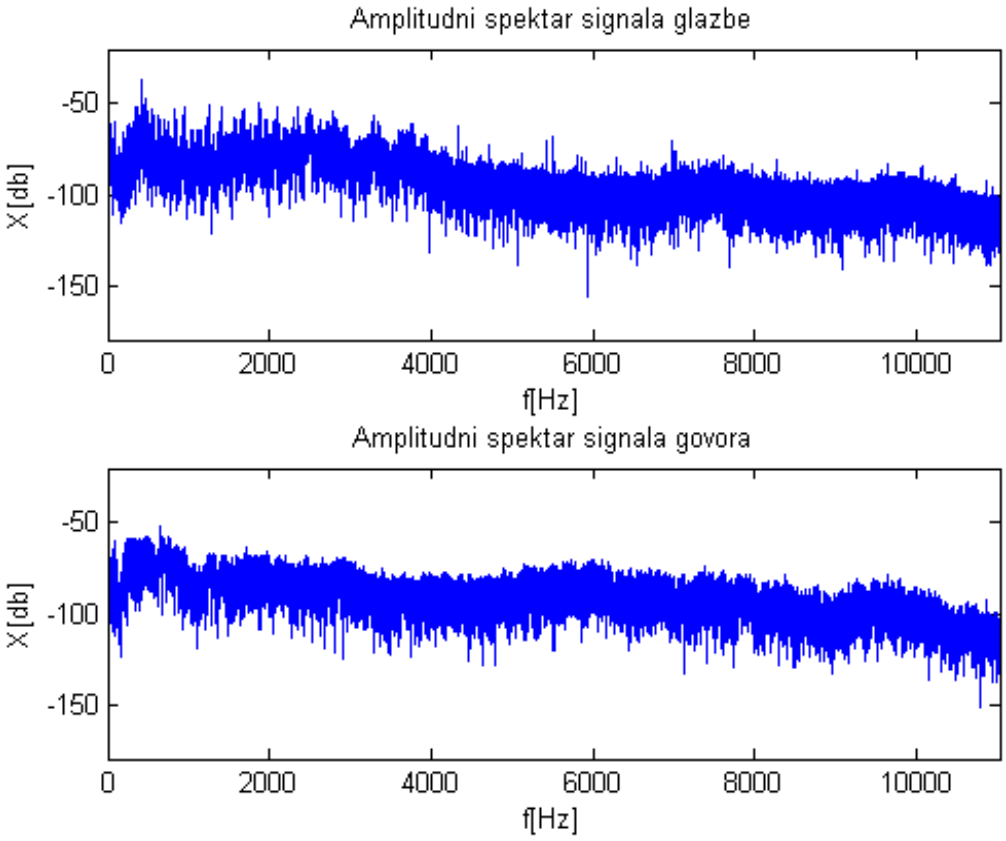
Ljudi intuitivno razlikuju govor i glazbu, no računalima je taj zadatak puno teži. I govor i glazba se sastoje od niza diskretnih vremenskih uzoraka te je iz vremenskih dijagrama na slici br 1. na prvi pogled vidljiv problem.



SLIKA 1. VREMENSKI DIJAGRAM GOVORA I GLAZBE

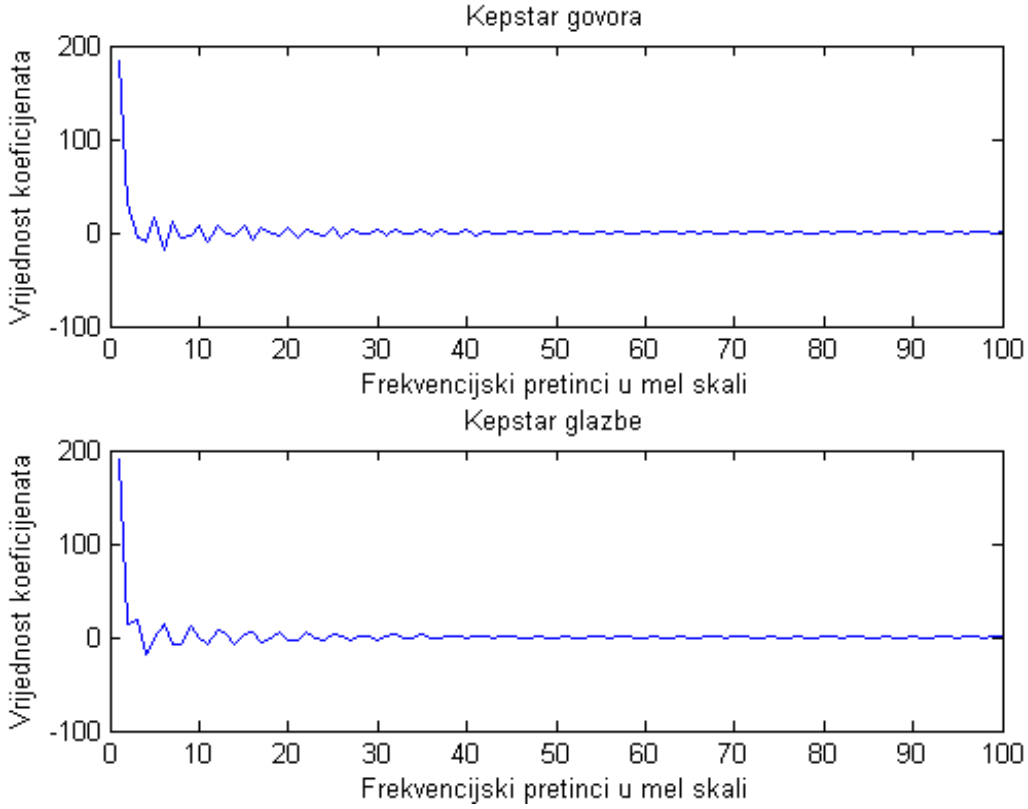
Naime, iz ovakvog prikaza ni čovjek ne može jednostavno odrediti je li prikazan vremenski dijagram govora ili glazbe. Jedina karakteristika kojom se vizualno može „prepoznati“ govor su česte pauze, čak ni ta karakteristika ponekad nije dovoljna, zbog toga što se pauze javljaju i u glazbi.

S obzirom da iz vremenskih dijagrama ne možemo s dovoljnom točnošću odrediti je li riječ o govoru ili glazbi, postavlja se pitanje možemo li razlikovati govor i glazbu iz njihovih spektara ili kepstara. Sljedeća slika prikazuje spektre govora i glazbe.



SLIKA 2. SPEKTAR GOVORA I GLAZBE

Kao što se vidi sa slike br. 2., iz grafičkih prikaza spektara je u potpunosti nemoguće odrediti je li riječ o govoru ili glazbi, zbog toga što spektri imaju identičan oblik.



SLIKA 3. KEPSTAR GOVORA I GLAZBE

Slika br. 3. pokazuje da se ni iz kepstara govora i glazbe ne može pouzdano odrediti o kojoj je vrsti signala riječ. Time se postavlja cilj i svrha ovog diplomskog rada.

Rad će, motiviran činjenicom da iz jednostavnih prikaza zvučnog signala nije moguće odrediti o kojoj je vrsti signala riječ, analizirati glavne parametre koji se mogu koristiti u svrhu klasifikacije zvučnih signala, te metode kojima se zvučni signali mogu klasificirati. Kao primjer će se prikazati nekoliko javno dostupnih metoda.

2. Povijesni pregled

Interes za razlikovanje govora i glazbe stariji je i od modernih računala. U svom radu *Categories and boundaries in speech and music*[1]. Cutting i Rosner još su 1974. godine proučavali način na koji ljudi percipiraju govor i glazbu. Njihov je rad temeljen na pretpostavci da ljudi lakše razlikuju kategorije (govor i glazbu) nego što mogu razlikovati članove svake od kategorija.

Eksperimenti su se sastojali od kontinuirane promjene jednog fizikalnog parametra kroz vrijeme. Primjerice, kontinuiranom promjenom nagiba drugog formanta dobivena su tri različita sloga, [ba], [ga] i [da]. Ono što je zanimljivo napomenuti je da je promjena u dojmu tih slogova diskontinuirana. Odnosno, ljudi su u svakom trenu čuli isključivo jedan od slogova, a nikad „nešto između".

Eksperiment je proširen na generiranje zvukova sličnih sviranju violine gudalom (*vibrato*) i prstima (*pizzicato*). Mijenjanjem brzine porasta signala dvije vrste sviranja prelaze iz jednog u drugo. I ovdje se može primijetiti da je ljudski doživljaj kvantan, odnosno, ljudi nisu ni u jednom trenutku doživili „nešto između" već uvijek ili *vibrato* ili *pizzicato*.

Osim odraslih ljudi, i novorođenčad pokazuje da razlikuje govor i glazbu. Eksperiment izveden 1982. godine u sklopu rada *Development of infant ear asymmetries for speech and music*[2] pokazao je da na fizičkoj razini postoji razlika u načinu na koji novorođenčad tumači zvukove. Naime, tri skupine novorođenčadi, starosti dva, tri i četiri mjeseca podvrgnuto je mjerenju broja otkucaja srca prije i nakon zvučnih podražaja. Zvučni podražaji su umjetno generirani na sličan način kao i u prethodnom eksperimentu. Govorni podražaji podrazumijevaju troformantne slogove [ba], [pa], [da], [ta], [ga] i [ka], a glazbeni notu veliko A, frekvencije 440 Hz, sintetiziranu različitim instrumentima.

Rezultati su pokazali da desno uho lakše prepoznaje govor, a lijevo uho ton glazbe. S obzirom na to da ušima upravljaju obrnute polutke mozga, može se zaključiti da lijeva polutka služi za obrađivanje govora, a desna za obrađivanje glazbe. U sve su tri dobne skupine dobiveni vrlo slični rezultati, iako se može

primjetiti da se u novorođenčadi starosti dva mjeseca još razvija fonetski rječnik pa je slabije prepoznavanje govora u oba uha.

Razlikovanje govora od glazbe bilo je predmet brojnih psiholoških i medicinskih istraživanja tijekom posljednje trećine prošlog stoljeća, a s vremenom i napretkom procesne moći računala postavilo se pitanje može li, i u kojoj mjeri, računalo razlikovati govor od glazbe.

Prvi rad koji se bavio strojnim razlikovanjem govora od glazbe je *Real-time discrimination of Broadcast Speech/Music*[3] iz 1996. godine. Cilj ovog rada bio je stvoriti sustav koji pretražuje FM radio kanale te odabire one na kojima se u danom trenutku emitira glazba. Za to je bilo potrebno razviti sustav koji u stvarno vremenu prepoznaje glazbu. Sustav se temelji na razlikama govora i glazbe u vremenskoj domeni, zato da se izbjegne izračun Fourierove transformacije.

Korišteni algoritam oslanja se prije svega na normalizirani broj prolaza signala kroz ništicu (*Zero Crossing Rate*, ZCR). ZCR je vrlo snažna mjera za izuzetno jednostavno određivanje bezvučnog od zvučnog govora, zato što je energija bezvučnih suglasnika vrlo malena. Bezvučni suglasnici nastaju prolaskom struje zraka kroz suženje u govornom traktu pri čemu je vrijednost signala vrlo niska te dolazi do većeg broja prolaza kroz ništicu. Govor se sastoji od stalne izmjene zvučnih i bezvučnih samoglasnika što se očituje u naglim promjenama ZCR-a kroz vrijeme. Glazba, s druge strane, ne pokazuje značajne promjene ZCR-a kroz vrijeme zbog toga što se uglavnom sastoji od tonova slične glasnoće i trajanja.

Kombinirajući podatak o ZCR i snazi signala na blokovima duljine 2.4 sekunde dobivena je točnost od 98.4% na podacima koji su korišteni za izračun značajki i parametara klasifikacije.

Ovaj je rad poslužio kao inspiracija za najznačajniji znanstveni rad u ovom području, *Construction and Evaluation of a Robust Multifeature Speech/Music Discriminator*[4], iz 1997. godine.

Riječ je o računalnom sustavu sposobnom u stvarnom vremenu razlikovati govorne i glazbene signale. Iz signala se izračunava trinaest značajki po čijim se svojstvima mjere razlike između govora i glazbe. Važno je napomenuti da svaki od radova koji se naknadno bavio razlikovanjem govora i glazbe koristi neke od značajki navedenih u radu i koristi ovaj rad kao jedno od glavnih polazišta u daljnjem istraživanju.

Sve se značajke izračunavaju na intervalu od jedne sekunde, a većina njih je izražena kao očekivanje parametra. Pet značajki su relativno nepromjenljive za glazbu, a razlikuju se značajno kod zvučnog i bezvučnog govora pa se u tim slučajevima koristi i varijanca tih parametara. Osim velikog broja značajki, ovaj rad je testirao četiri različite metode klasifikacije da bi se utvrdilo kojom se ostvaruje najmanji broj pogrešno klasificiranih segmenata. S obzirom da je riječ o vrlo značajnom znanstvenom radu, bit će detaljnije obrađen u zasebnim poglavljima.

Pokazalo se da se velik broj značajki zvučnih zapisa može koristiti u svrhu klasificiranja govora i glazbe, ali uz upitnu korisnost. Usporedbom korisnosti značajki bavi se rad *A Comparison of Features for Speech, Music Discrimination*[5]. U sklopu rada uspoređena su četiri parametra i njihove derivacije, ukupno 8 značajki zvučnih signala. Riječ je o kepstralnim koeficijentima i promjeni kepstralnih koeficijanata, amplitudi i promjeni amplitude kroz pet uzastopnih okvira, broju prolaza kroz ništicu unutar okvira od 10 ms i promjeni tog broja kroz pet uzastopnih okvira te visina tona i promjena visine tona. Pritom su se broj prolaza kroz ništicu i njegova promjena pokazali kao uvjerljivo najlošiji parametri. Zasebno pokazuju vjerojatnost pogreške oko 20%, dok je klasifikacija kombiniranjem oba parametra otprilike podjednako točna kao i korištenjem svih preostalim parametara zasebno. Preostala tri para parametara pokazuju podjednaku točnost, no važno je napomenuti da se za njihov izračun troši puno više procesorskog vremena nego za izračuna broja prolaza kroz ništicu.

3. Zvučni zapisi

Za potrebe analize metoda razlikovanja govora i glazbe bilo je potrebno pronaći javno dostupnu bazu podataka koja sadrži zasebne snimke govora i glazbe. U sklopu ovog rada koristit će se baza podataka generirana pomoću programskog okvira *Marsyas*[6]. Radi se o 128 poluminutnih zvučnih zapisa u .wav formatu, od kojih polovica predstavlja glazbu, a polovica govor. Svi su zapisi jednokanalni, 16-bitni i otipkani frekvencijom 22050 Hz. S obzirom da je riječ o prilično velikoj količini podataka, oko 300 MB, nije moguće koristiti čitavu bazu podataka. Stoga sam napisao Matlab skriptu koja generira novu datoteku koja sadrži po petnaest slučajno odabranih zapisa govora i glazbe slučajnim redoslijedom.

Zbog toga što svi zapisi imaju različitu energiju, pri čitanju i stvaranju nove datoteke s odabranim zapisima obavlja se normalizacija po amplitudi. Skripta traži maksimalnu vrijednost unutar svakog zapisa te dijeli čitav zapis tom vrijednošću. Nakon što je obavljena normalizacija svi zapisi sadrže vrijednosti između -1.0 i +1.0 uključivo. S obzirom da vrijednost +1.0 ne postoji u frakcionalnom zapisu s fiksnom decimalnom točkom, ona se pri pisanju u datoteku zamjenjuje s najbližom manjom vrijednošću, $+1.0 - 2^{-(n-1)}$, gdje je n broj bitova preciznosti zapisa.

U slučaju kad je to nužno za izračun nekog od parametara, koriste se snimke u kojima je obavljena i normalizacija po snazi, na takav način da svi poluminutni zapisi imaju jednaku snagu.

4. Značajke važne za razlikovanje govora i glazbe

Construction and Evaluation of a Robust Multifeature Speech/Music Discriminator[4] definira trinaest značajki zvučnih signala. To su:

1. Modulacijska energija na 4 Hz
2. Postotak okvira niske energije
3. Frekvencija većinske spektralne snage
4. Varijanca frekvencije većinske spektralne snage
5. Spektralni centroid
6. Varijanca spektralnog centroida
7. Spektralni tok
8. Varijanca spektralnog toka
9. Broj prolaza kroz ništicu
10. Varijanca broja prolaza kroz ništicu
11. Modul razlike spektra i rekonstruiranog spektra iz kepstra
12. Varijanca modula razlike spektra i rekonstruiranog spektra iz kepstra
13. Pulsna metrika

U nastavku će se posebno obraditi važnije značajke zvučnih signala.

4.1 Modulacijska energija na 4 Hz

Govor ima karakteristični šiljak modulacijske energije na 4 Hz, koji je kod glazbe manje uočljiv. Slaney i Scheirer koriste dio MFCC algoritma da bi zvučni signal pretvorili u četrdeset perceptualnih kanala frekvencija i izračunavaju energiju svakog kanala. Pojasno-propusnim filtrom drugog reda izvlače frekvenciju 4 Hz i izračunavaju energiju kvadriranjem i usrednjivanjem rezultata. Energija se normalizira ukupnom energijom vremenskog okvira koji se promatra i sumira kroz sve kanale. Kao što je već rečeno, zbog prirode govora, on u pravilu ima veću modulacijsku energiju od glazbe.

Da bi se detaljnije obradilo izračunavanje modulacijske energije na 4 Hz u nastavku će se pobliže opisati pojam kepstra, njegove uloge u obradi signala i MFCC algoritam.

Kepstar je rezultat inverzne Fourierove transformacije logaritma spektra polaznog signala.

$$\hat{y}(t) = \mathcal{F}^{-1}(\log(\mathcal{F}(y(t)))) \tag{1}$$

Pritom kepstar ne predstavlja signal u vremenskoj domeni, iako je riječ o rezultatu inverzne Fourierove transformacije spektra, nego signal u takozvanoj kepstralnoj domeni. Posljedica logaritmiranja spektra može se uočiti ako promotrimo ulazni signal kao konvoluciju dva signala.

$$\begin{aligned}
\hat{y}(t) &= \mathcal{F}^{-1}\left[\log\left(\mathcal{F}(y_1(t) * y_2(t))\right)\right] \\
&= \mathcal{F}^{-1}[\log(\mathcal{F}(y_1(t))] \times \mathcal{F}^{-1}[\log(\mathcal{F}(y_2(t))] \\
&= \mathcal{F}^{-1}\left[\log[\mathcal{F}(y_1(t))] + \log[\mathcal{F}(y_2(t))]\right] \\
&= \mathcal{F}^{-1}[\log[\mathcal{F}(y_1(t))]] + \mathcal{F}^{-1}[\log[\mathcal{F}(y_2(t))]] \\
&= \hat{y}_1(t) + \hat{y}_2(t)
\end{aligned} \tag{2}$$

Konvolucija u vremenskoj domeni predstavlja umnožak spektara u spektralnoj domeni. Logaritam ima svojstvo da pretvara logaritam umnožaka u zbroj logaritama. Zbog linearnosti Fourierove transformacije i inverzne Fourierove transformacije u kepstralnoj, kvazivremenskoj domeni je iz konvolucije dobivena linearna kombinacija signala. Odnosno, kepstar ima svojstvo razdvajanja konvoluiranog signala na njegove komponente.

Ovisno o vrsti logaritma i spektra kepstri se dijele na kompleksne, realne, kepstre snage i fazne kepstre. Kompleksni kepstar koristi kompleksnu logaritamsku funkciju i zbog toga zadržava podatak o fazi signala. Realni spektar se dobiva logaritmiranjem amplitudne karakteristike spektra pa se u kepstru gubi informacija o faznom pomaku signala, odnosno, linearna kombinacija signala u kepstralnoj domeni centrira se oko ništice. Kepstar snage dobiva se logaritmiranjem spektra snage, a fazni kepstar izvlačenjem faze iz kompleksnog kepstra.

U digitalnoj se obradbi zvuka vrlo često koriste *Mel-frequency cepstral coefficients* (MFCC), odnosno, mel-frekvencijski kepstralni koeficijenti. Riječ je o

reprezentaciji kratkovremenskog spektra zvuka, a temelji se na Fourierovoj transformaciji, logaritamskom spektru snage i nelinearnoj mel skali.

Algoritam za izračun MFCC se sastoji od šest koraka[7]:

1. Segmentacija zvučnog signala u kratke okvire, trajanja oko 25 ms
2. Izračun spektra snage ulaznog signala diskretnom Fourierovom transformacijom
3. Mapiranje frekvencija iz linearne skale u Hz u logaritamsku mel skalu
4. Logaritmiranje spektra snage u točkama odabranim u prethodnom koraku
5. Izračun diskretne kosinusne transformacije logaritmiranog spektra snage
6. MFCC su definirani kao amplitude dobivenog kepstra, pri čemu su najznačajniji prvih 2-13 koeficijenata

S obzirom da se zvučni signal stalno mijenja, potrebno ga je promatrati u kratkim vremenskim intervalima u kojima se ne mijenja značajno, odnosno, u intervalima u kojima je statistički stacionaran. Zbog toga se u praksi signal segmentira u okvire trajanja 20-40 ms. Okviri kraći od 20 ms ne sadrže dovoljno uzoraka da bi se mogao dovoljno detaljno procijeniti spektar, a okviri duži od 40 ms nemaju svojstvo statističke stacionarnosti. S obzirom da su snimke koje se koriste u sklopu ovog diplomskog rada očitane frekvencijom od 22050 Hz, to znači da je svaki okvir dugačak 551 uzorak.

Svi se okviri preklapaju, tako da idući okvir počinje 10 ms nakon prethodnog. To se radi zato što su rubovi okvira prigušeni Hannovim ili Hammingovim otvorom. Zbog toga pomak od 40% okvira omogućava da i ti dijelovi signala sudjeluju u izračunu.

Idući je korak izračun spektra snage. Motivacija za to je činjenica da je ljudska pužnica nelinearan organ. Različiti dijelovi pužnice vibriraju ovisno o snazi i frekvenciji zvuka koji je došao na bubnjić uha. Sljedeći izraz opisuje izračun spektra snage za svaki od okvira.

$$P_i(k) = \frac{1}{N}\left|\sum_{n=1}^{N} s_i(n)h(n)e^{-j2\pi kn/N}\right|^2 \qquad 1 \le k \le K \qquad (3)$$

Pritom $s_i(n)$ ulazni signal, odnosno, njegov i-ti okvir, $h(n)$ Hammingov ili Hannov otvor, K broj točaka u kojima se izračunava diskretna Fourierova transformacija, a N duljina svakog okvira.

Ovisno o mjestu na kojem pužnica zavibrira, dlačice unutar pužnice podraže odgovarajući živčani završetak i obavijeste mozak o kojoj je frekvenciji riječ. S obzirom na konačno malen razmak između dlačica, pužnica ne može razlikovati vrlo bliske frekvencije. Ova je pojava puno izraženija na visokim frekvencijama. Zbog toga linearna frekvencijska skala nije prikladna za spektar zvuka, već se koristi posebna logaritamska skala, takozvana mel skala.

Linearna i mel skala vezane su sljedećim izrazima:

$$m[mel] = 2595 \log\left(1 + \frac{f[Hz]}{700}\right) = 1127\, ln\left(1 + \frac{f[Hz]}{700}\right) \qquad (4)$$

$$f[Hz] = 700\left(10^{m[mel]/2595} - 1\right) = 700\left(e^{m[mel]/1127} - 1\right) \qquad (5)$$

Iz frekvencijskog intervala na kojem se želi promatrati kepstar izračunava se 20-40 ekvidistantnih frekvencija (u pravilu 26) u mel skali te se one preračunavaju u frekvencije u hercima. Na taj se način dobiva veći broj točaka na nižim frekvencijama, a manji broj točaka na višim frekvencijama, sukladno načinu na koji pužnica detektira zvukove. Oko svake se frekvencije uskim trokutastim filtrom izdvaja spektar snage na toj frekvenciji. Nakon logaritmiranja odgovarajućih energija dobiva se 26 logaritamskih energija. Nad tih 26 energija se obavlja diskretna kosinusna transformacija i rezultat je 26 kepstralnih koeficijenata.

Osim izravnih kepstralnih koeficijenata, u obradi zvuka se često koriste delta i delta-delta koeficijenti. Riječ je o diferencijalnim i akceleracijskim koeficijentima dobivenim iz MFCC koeficijenata. Opisani kepstralni koeficijenti predstavljaju statičke karakteristike zvučnog signala, a delta i delta-delta koeficijenti se

koriste za opis dinamičkih karakteristika signala. Naime, za izračun delta koeficijenata se koriste razlike kepstralnih koeficijenata različitih okvira, najčešće okvira čija je udaljenost veća od dva, tako da ne postoji preklapanja među okvirima. Pritom vrijedi sljedeća formula:

$$d_t = \frac{\sum_{n=1}^{N} n(c_{t+n} - c_{t-n})}{2\sum_{n=1}^{N} n^2} \quad (6)$$

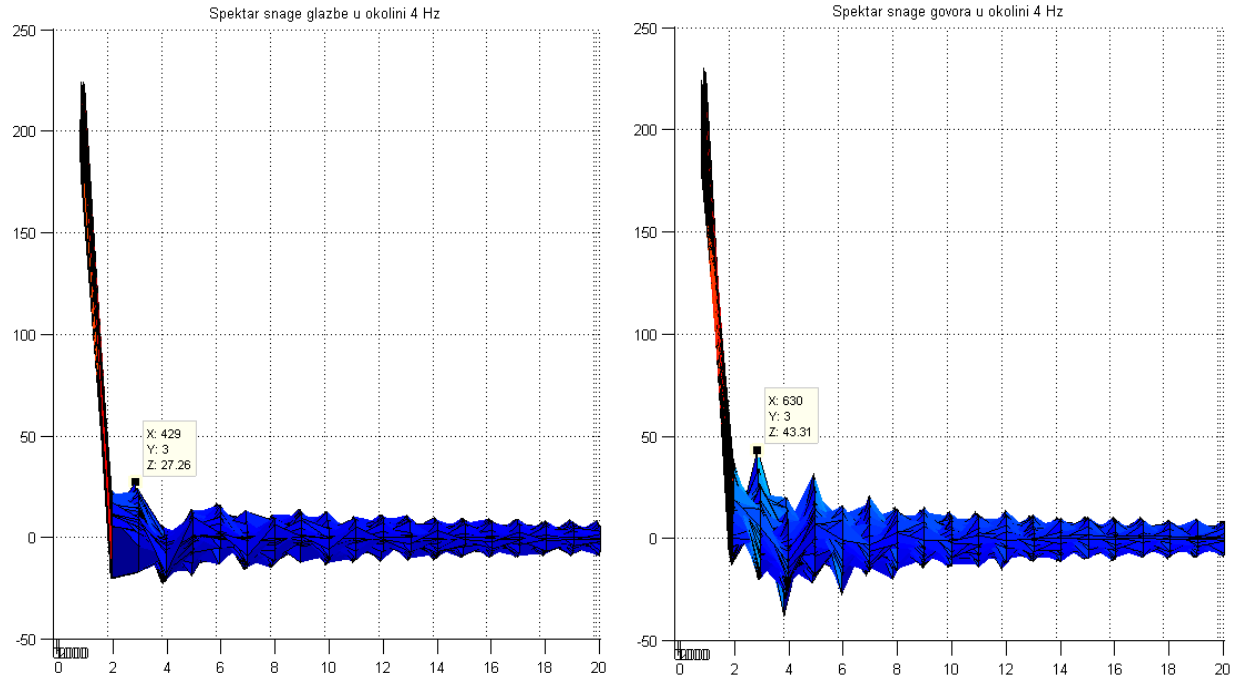
Pritom je d_t delta koeficijent u trenutku t, N razmak između okvira koje se promatra, a c_t kepstralni koeficijent u trenutku t. Delta-delta koeficijenti se dobivaju ako se umjesto kepstralnih koeficijenata u izrazu koriste delta koeficijenti.

MFCC algoritam se lako može prilagoditi tako da obavlja procjenu modulacijske energije na 4 Hz. Naime, uz dobro odabrane donju graničnu frekvencijsku, gornju graničnu frekvenciju i broj ekvidistantnih mel frekvencija moguće je postaviti jednu od frekvencija spektra snage u točno 4 Hz. Na ovaj način su Slaney i Scheirer koristili MFCC algoritam za dobivanje modulacijske energije u 4 Hz.

U sklopu ovog rada korišten je javno dostupan MFCC algoritam[8], uz izmijenjene parametre analize, i *Marsyas* zvučni zapisi.

Algoritam koristi prozore širine 25 ms, uz pomak od 10 ms između prozora te ima promjenljiv broj kepstralnih koeficijenata koje izračunava, odnosno, promjenljiv broj frekvencija za koje se izračunavaju kepstralni koeficijenti te ima promjenljivo frekvencijsko područje. Početni algoritam izračunava spektar snage signala u 512 točaka i time ne osigurava dovoljnu frekvencijsku rezoluciju da bi se mogla promatrati snaga na frekvenciji od 4 Hz, već je rezolucija oko 10 Hz. Povećanje broja točaka u kojima se izračunava spektar snage omogućava smanjivanje frekvencijske rezolucije na 2-3 Hz, što je dovoljno za analizu snage na 4 Hz. Daljnje povećanje broja točaka onemogućava izvršavanje algoritma jer zahtjeva više memorije nego što računalo može izdvojiti.

Uz modificirani broj točaka brze Fourierove transformacije i dobro odabran broj frekvencijskih pretinaca dobivaju se sljedeći grafički prikazi.



SLIKA 2. MODULACIJSKA ENERGIJA NA 4 HZ

Može se primijetiti da treći frekvencijski pretinac ima veći iznos za govor u odnosu na treći pretinac kod glazbe. Naime, u tom se pretincu nalazi frekvencija od 4 Hz. Ona se nije mogla pomaknuti u neki drugi pretinac zbog ograničenja računala koje obavlja simulaciju.

Valja napomenuti da graf predstavlja „spljošteni“ trodimenzionalni graf u kojem je treća, spljoštena os, vrijeme, odnosno trajanje snimke.

Bez obzira na to koji dio snimke je korišten, u svakoj od simulacija je uvijek dobiven puno veći iznos spektralne snage na 4 Hz kod govornih signala u odnosu na glazbene signale te se može reći da je spektralna energija na 4 Hz vrlo dobar parametar za diskriminaciju.

4.2 Postotak okvira niske energije

Slaney i Scheirer[4] kažu da je postotak okvira koji sadrže manje od 50% prosječne energije unutar okolnih jednu sekundu zapisa dobar pokazatelj da li je riječ o govoru ili glazbi. Naime, govor sadrži velik broj pauza, djelomice zbog prirode govora, odnosno, pauza između riječi i rečenica, a djelomice zbog

fonetskih razloga, odnosno, načina na koji se određeni skupovi fonema izgovaraju. Te se pauze, kao i bezvučni suglasnici, manifestiraju kao dijelovi signala koji nose vrlo malu energiju, s obzirom da se radi o vrijednostima signala bliskima ništici.

Glavni problem ovog parametra je što ne može razlikovati govor od tišine, budući da iz niske energije signala tijekom jedne sekunde ne možemo zaključiti je li riječ o tišini ili bezvučnom govoru. Zbog toga se ovaj parametar ne može samostalno koristiti, već se najčešće kombinira s brojem prolaza kroz ništicu (ZCR) na jednosekundnom intervalu. Tijekom tišine nema prolaza kroz ništicu ili ih ima jako malo, dok bezvučni govor karakterizira značajno veći broj prolaza kroz ništicu. Na temelju ova dva parametra moguće je razlikovati govor od glazbe i bezvučni govor od tišine.

Za analizu ovog parametra korišten je MIRtoolbox[9]. Riječ je o nizu Matlab funkcija koje služe za izračun različitih značajki zvučnih zapisa, a u ovom slučaju se koristi funkcija mirlowenergy.m koja služi za izračun postotka okvira unutar zadanog zvučnog zapisa čija je energija manja od zadanog praga.

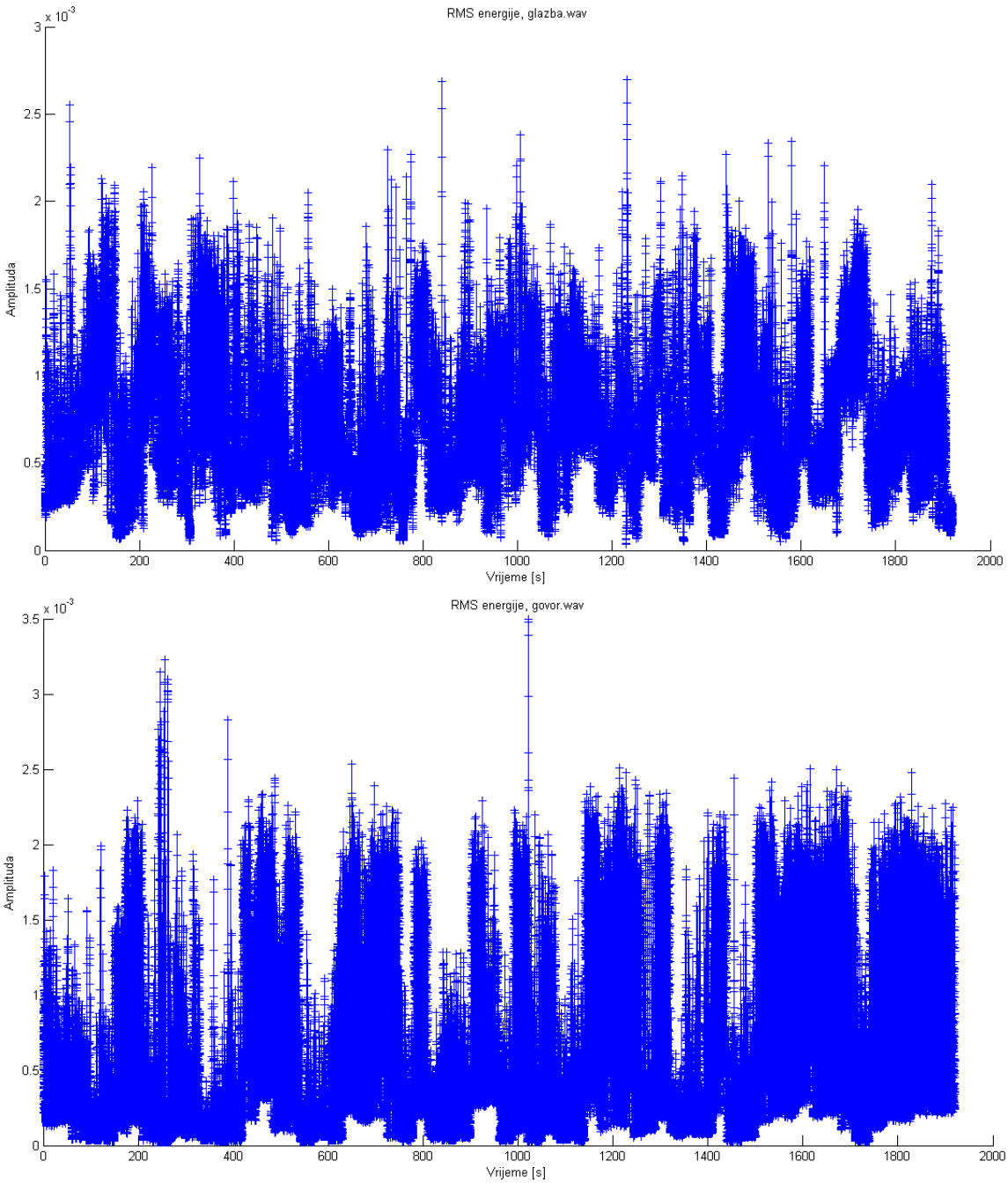
Funkciji se daje naziv datoteke i željeni prag, a moguće je dati i dodatne parametre, ali se u ovom slučaju oni neće koristiti. Za svaki se okvir zvučnog zapisa izračunava snaga unutar okolne jedne sekunde te se analizira da li je prosječna energija zapisa manja od polovine prosječne energije okoline. Snaga se izračunava prema uobičajenom izrazu za snagu diskretnog signala:

$$P = \frac{1}{N}\sum_{n=1}^{N} x^2[n] \qquad (7)$$

Pritom je N duljina na kojoj se izračunava snaga, a $x[n]$ signal za koji se izračunava snaga.

Analizom na našim zvučnim zapisima dobiveni su sljedeći rezultati. 29.076% okvira koji odgovaraju govoru ima prosječnu energiju manju od polovine energije okoline, dok samo 2.9011% okvira glazbe ima prosječnu energiju

manju od polovine okolne energije. Time se jasno pokazuje da postoji velika korelacija između postotka okvira niske energije i vrste zvučnih zapisa.



SLIKA 3. USPOREDBA ENERGIJA SIGNALA

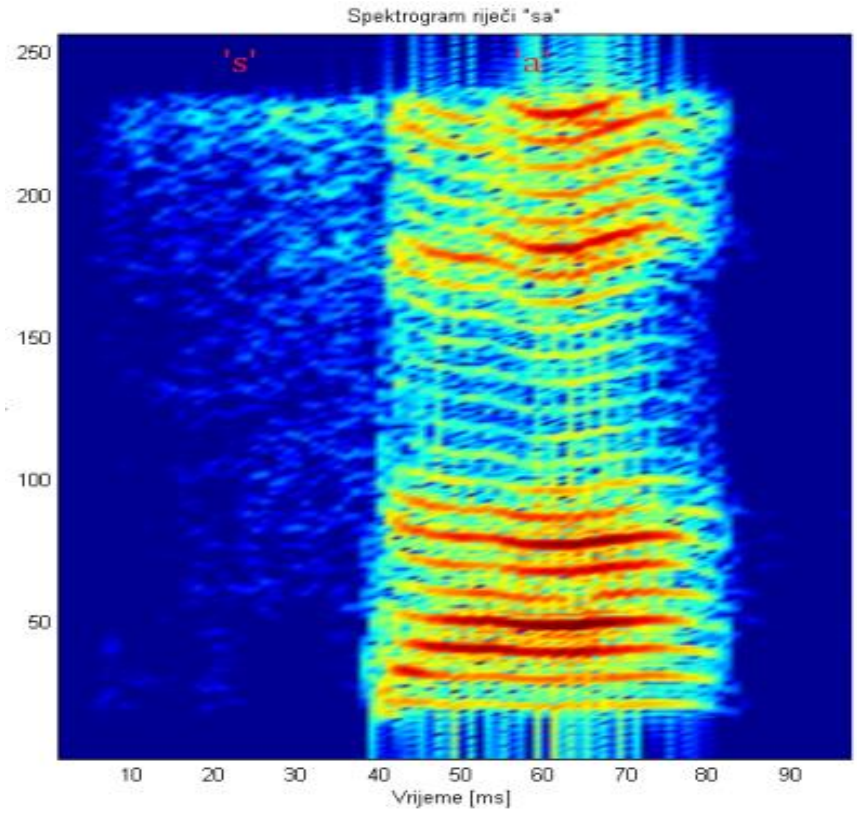
Osim postotka okvira niske energije, funkcija također generira vremenski dijagram energije signala. Oba grafa, jedan za glazbu, drugi za govor su

prikazani na slici br. 5.. Iz grafova se vrlo jasno vidi da je kod govora puno veća zastupljenost nižih energija.

4.3 Frekvencija većinske spektralne snage

Frekvencija većinske spektralne snage je definirana kao N-ti percentil distribucije spektra snage, gdje je N u pravilu 85 ili 95. Riječ je o frekvenciji ispod koje je sadržano N% spektra snage. Ova se mjera koristi za razlikovanje zvučnog od bezvučnog govora. Naime, većina energije zvučnog govora, kao i glazbe, se nalazi na nižim frekvencijama. Za razliku od toga, bezvučni govor ima malenu energiju i nema značajnih komponenti na nižim frekvencijama nego je njegova snaga podjednako raspoređena na nižim i višim frekvencijama. Zbog toga se 95% snage zvučnog govora i glazbe koncentrira na nižim frekvencijama, dok je potrebno proširiti frekvencijsko područje da bi se obuhvatilo 95% snage bezvučnog govora.

Zvučni govor nastaje prolaskom zračne struje iz pluća kroz glasnice dok one titraju. Za razliku od toga, pri nastanku bezvučnog govora glasnice ne titraju.

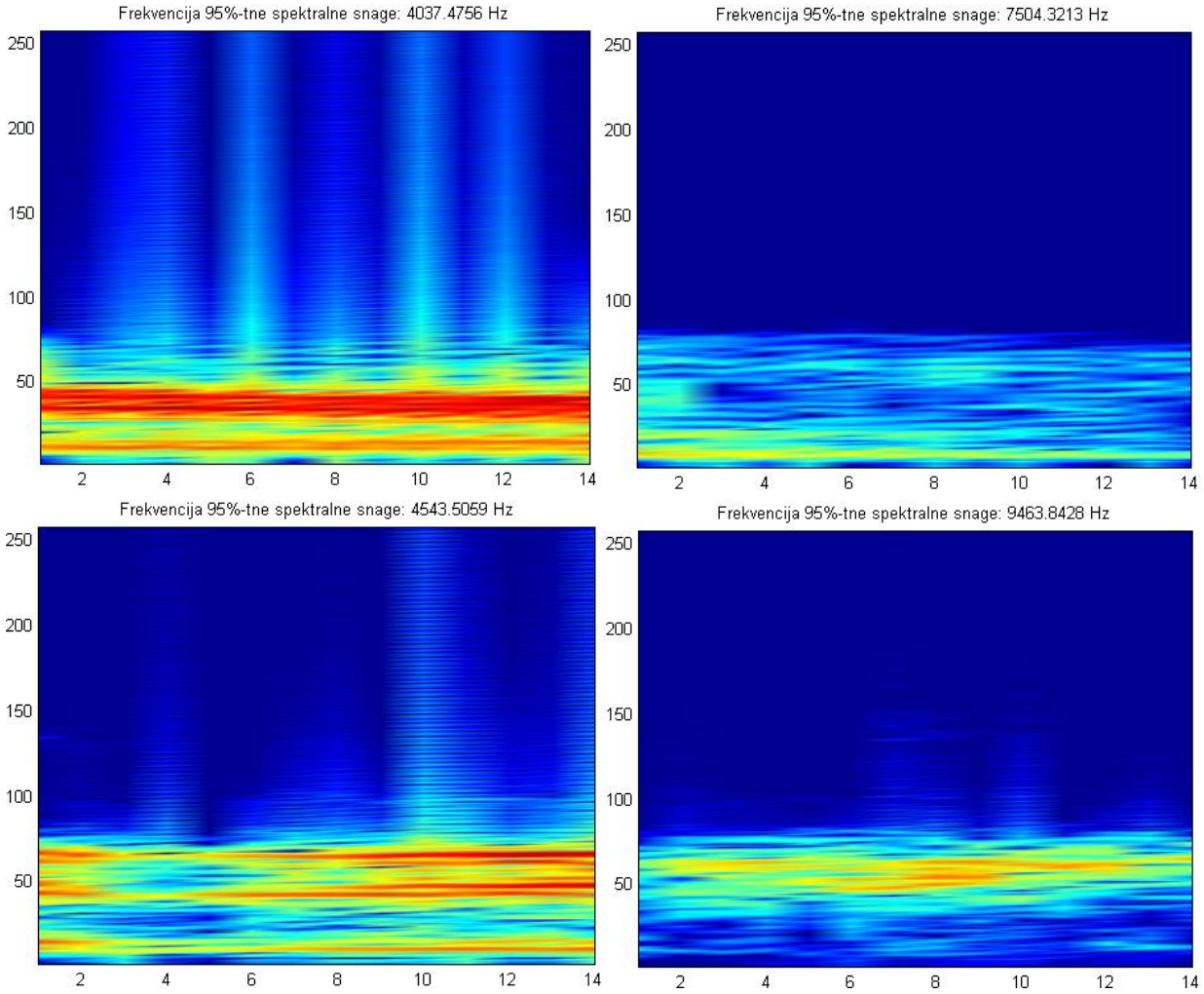


SLIKA 4. SPEKTROGRAM RIJEČI 'SA'

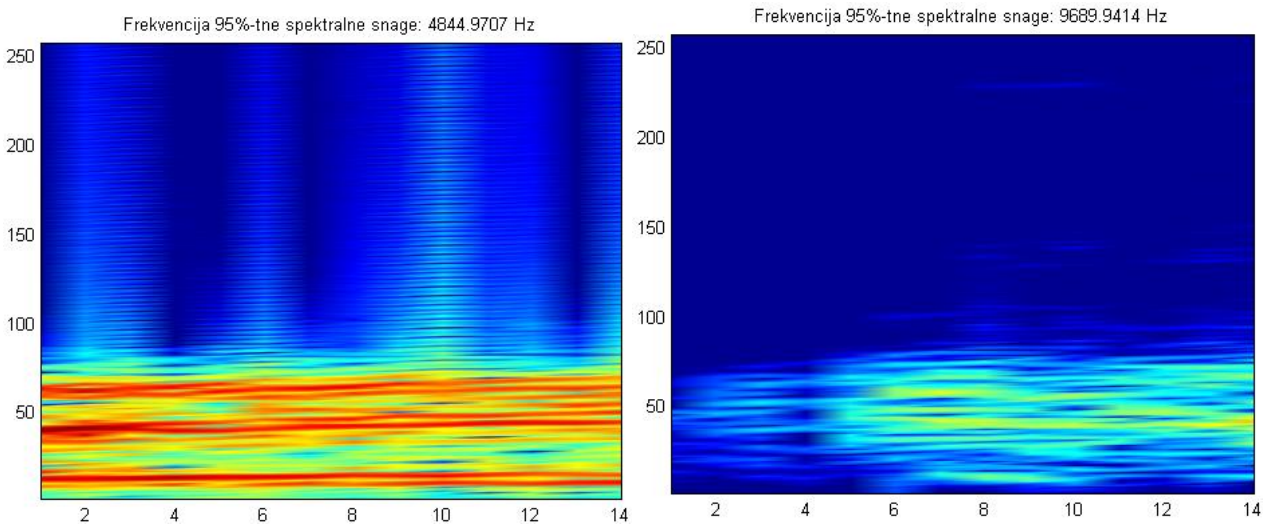
Vibracije glasnica unose značajan dio energije zvučnim glasovima i to se očituje u spektrogramu. U hrvatskom su jeziku bezvučni glasovi: p, t, k, č, ć, s, š, f, c, h.

Slika na prethodnoj stranici prikazuje spektrogram riječi 'sa', u kojem se jasno vidi razlika u snazi između zvučnog 'a' i bezvučnog 's'.

Da bi se mogla provesti analiza povezanosti zvučnosti i bezvučnosti s frekvencijom većinske spektralne snage koriste se govorni zvučni zapisi koji su podijeljeni u segmente trajanja 60 ms. Za te se segmente iscrtava spektrogram kao najjednostavniji način utvrđivanja zvučnosti te se korištenjem MIRtoolboxa i njegove funkcije mirrolloff.m izračunava frekvencija 95%-tne spektralne snage.



SLIKA 5. USPOREDBA 95%-TNIH SPEKTRALNIH SNAGA ZA ZVUČNE (LIJEVO) I BEZVUČNE (DESNO) GLASOVE



SLIKA 6. USPOREDBA 95%-TNIH SPEKTRALNIH SNAGA ZA ZVUČNE (LIJEVO) I BEZVUČNE (DESNO) GLASOVE

Slike 7. i 8. prikazuju da postoji korelacija između zvučnosti glasova i frekvencija većinske spektralne snage. Svi zvučni segmenti signala sadrže većinu snage do frekvencije 4.5 kHz, dok se snaga bezvučnih segmenata nalazi i na puno većim frekvencijama, dosežući i do 10 kHz.

Glavni problem ovog parametra je nemogućnost razlikovanja pauza u govoru od bezvučnih suglasnika. Pauze u govoru vrlo često zauzimaju čitav segment, a s obzirom da u njima skoro pa ne postoji nikakva snaga, 95% snage šuma se protegne na vrlo visoke frekvencije. Ova se pojava djelomice može primijetiti na grafu posljednjeg bezvučnog segmenta, gdje se tek na otprilike trećini segmenta javlja glas.

4.4 Spektralni centroid

Spektralni centroid je mjera koja se koristi u digitalnoj obradi signala za opisivanje spektara. Ona pokazuje gdje se nalazi „centar mase“ spektra. Perceptualno se ova značajka očituje kao boja tona.[10] Pod izrazom spektralni centroid se podrazumijeva frekvencija na kojoj se nalazi centroid, no ponekad se uz frekvenciju spektralnog centroida izračunava i težinski prosjek amplitude unutar zadanog frekvencijskog pojasa, gdje su težine frekvencije unutar pojasa. Pritom se razlikuju termini amplitudni spektralni centroid i frekvencijski spektralni centroid. Frekvencijski spektralnog centroida se izračunava kao težinska sredina frekvencija prisutnih u spektru signala pri čemu je težina svake

frekvencije iznos amplitudne karakteristike spektra na toj frekvenciji. Izraz na spektralni centroid prikazan je sljedećom jednadžbom.

$$Frekvencijski\ Centroid = \frac{\sum_{k=0}^{N-1} f[k]X[k]}{\sum_{k=0}^{N-1} X[k]} \quad (8)$$

Pritom $X[k]$ predstavlja amplitudu spektra na indeksu n, a $f[k]$ frekvenciju koja odgovara tom indeksu.

Alternativno, amplitudni centroid se izračunava prema izrazu:

$$Amplitudni\ Centroid = \frac{\sum_{k=0}^{N-1} f[k]X[k]}{\sum_{k=0}^{N-1} f[k]} \quad (9)$$

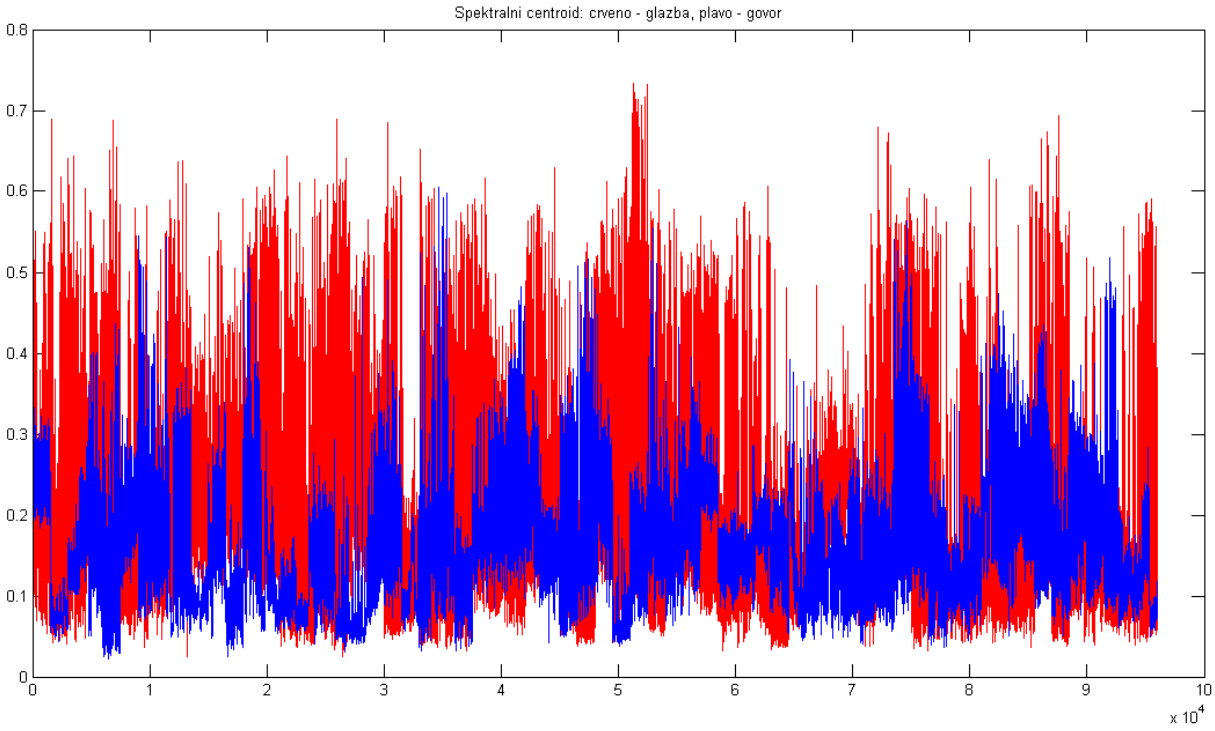
U daljnjem će se tekstu pod spektralni centroid podrazumijevati frekvencijski spektralni centroid zato što je on od puno većeg značaja za razlikovanje govora i glazbe.

S obzirom da je spektralni centroid približni „centar mase“ svakog frekvencijskog pojasa obuhvaćenog Fourierovom transformacijom on se u kontekstu govora može koristiti za aproksimaciju lokacije formanata, zbog toga što se formanti očituju kao vrhovi u okolnom pojasu frekvencija.

Slaney i Scheirer[4] definiraju spektralni centroid kao točku balansa spektra snage. Glazba sadrži razne udaraljke koje unose visokofrekvencijske zvukove te, inducirajući visokofrekvencijski šum, podižu balans spektra na više frekvencije. Ovaj je parametar testiran koristeći javno dostupan Matlab kod.[11] Snimke govora i glazbe segmentirane su na odsječke trajanja 20 ms tako da se osigura stacionarnost signala. Uz tako odabrane snimke za govor je dobivena prosječna frekvencija centroida 3415 Hz, dok je za glazbu dobiveno 4193 Hz. Odnosno, u prosjeku se spektar glazbe proteže na više frekvencije od spektra govora.

Samo 25.2% segmenata govora ima veći spektralni centroid od prosječnog centroida glazbe, dok 57.8% segmenata glazbe ima veći spektralni centroid od prosječnog centroida govora. Samo četiri segmenta govora imaju veći spektralni

centroida od trostukog centroida glazbe dok 2.6% segmenata govora ima veći spektralni centroid od prosječnog centroida govora.



SLIKA 7. SPEKTRALNI CENTROID GOVORA I GLAZBE

Na slici br. 9. se vrlo jasno vidi da je spektralni centroid glazbe uvjerljivo dominira i da je u značajnom broju slučajeva puno veći od centroida govora.

4.5 Spektralni tok

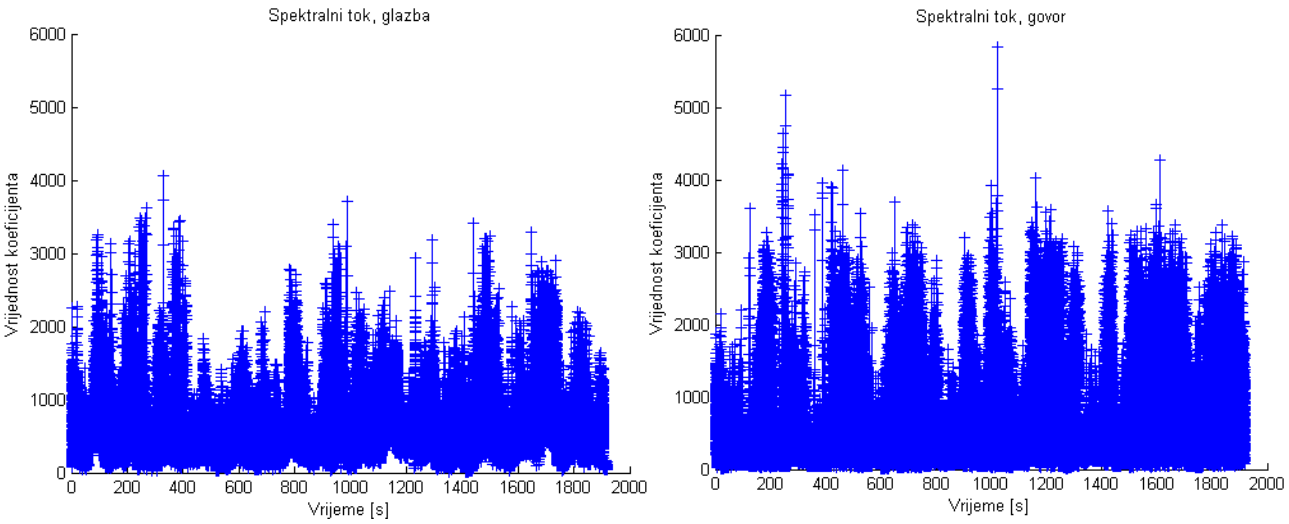
Spektralni tok je mjera koja pokazuje koliko se brzo mijenja spektar snage signala. Izračunava se uspoređujući spektre snage susjednih vremenskih okvira. Pritom se koristi euklidska udaljenost dvaju normaliziranih spektara. Ovako izračunat spektralni tok ne ovisi o snazi signala, zato što su spektri snage normalizirani, ni o faznom odnosu signala, zato što se uspoređuje samo amplitudni dio spektra.

$$Spektralni\ tok = \| |X_n| - |X_{n+1}| \| = \sqrt{\sum_{k=1}^{N} (|X_n| - |X_{n+1}|)^2}_k \qquad (10)$$

Glazba ima veću stopu promjene od govora te dolazi do puno većih promjena između susjednih okvira u odnosu na govor. Zbog harmonijskog kontinuiteta glazbe, ona tipično ima veći, ali konstantniji spektralni tok. S druge strane, govor se sastoji od stalnih prijelaza između samoglasnika i suglasnika. Zbog tih prijelaza govor nema konstantan spektralni tok, već on stalno varira.

Slaney i Scheirer[4] zbog toga zaključuju da glazba ima veći spektralni tok od govora, ali govor ima veću varijancu spektralnog toka od glazbe.

Unatoč brojnim pokušajima, ovaj parametar pokazuje upravo suprotno ponašanje prilikom testiranja na snimkama koje su korištenje u sklopu ovog diplomskog rada. Parametar je testiran koristeći dvije različite metode. Prva metoda je upotrebom MIRtoolboxove funkcije mirflux.m, a druga je metoda javno dostupa u sklopu matlab paketa *A method for silence removal and segmentation of speech signals*[11]. Datoteke koje sadrže govor i glazbu su normalizirane po snazi te su dobiveni sljedeći rezultati.



SLIKA 8. SPEKTRALNI TOK GOVORA I GLAZBE

Koristeći obje analize spektralni tok govora se pokazao značajno veći od spektralnog toka glazbe. Prosječna vrijednost spektralnog toka govora je dvostruko veća od spektralnog toka glazbe.

Usprkos tome što se sama vrijednost spektralnog toka ponaša neočekivano, njegova varijanca odgovara očekivanjima. Odnosno, varijanca spektralnog toka

govora kroz čitavu snimku iznosi 13.9 i značajno je veća od varijance vezane za glazbu, koja iznosi 3.2.

Točan razlog zbog kojeg se spektralni tok ponaša neočekivano na našim snimkama nije jasan, s obzirom da su za testiranje korištene dvije različite metode i da su oba signala normalizirana po snazi.

4.6 Prosječan broj prolaza kroz ništicu

Prosječan broj prolaza kroz ništicu definira se kao stopa promjene predznaka signala u vremenskoj domeni unutar određenog vremenskog intervala. On je određen brojem prolaza kroz ništicu i duljinom intervala na kojem se promatra ovaj parametar. Sljedeći izraz prikazuje formulu za izračun prosječnog broja prolaza kroz ništicu (ZCR, Zero Crossing Rate).

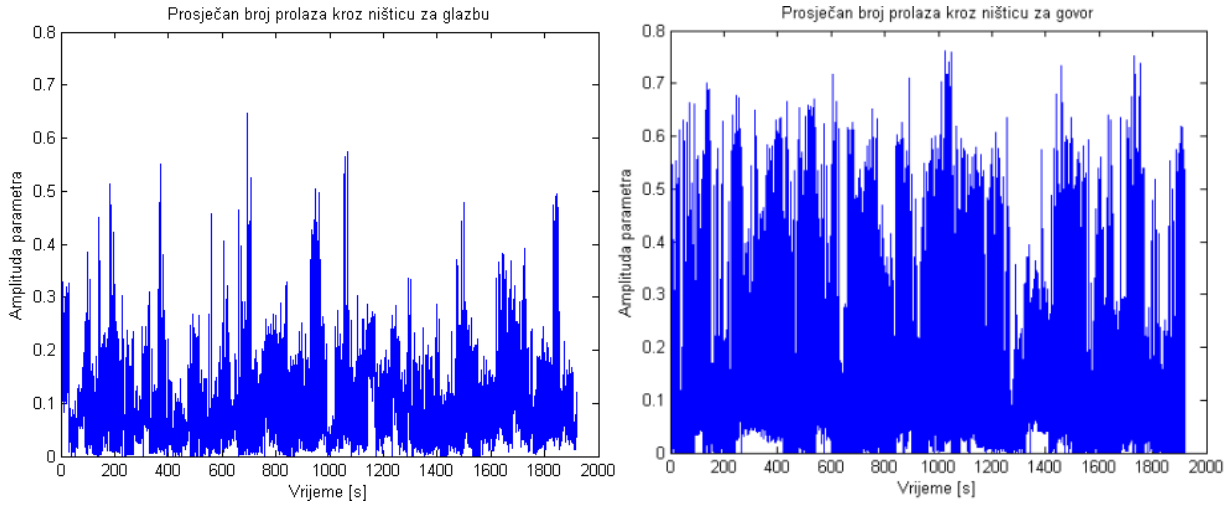
$$ZCR = \frac{1}{T-1}\sum_{t=1}^{T-1} Z\{s_t s_{t-1} < 0\} \qquad (11)$$

Pri tome je vrijednost operatora Z jedan ako je uvjet zadovoljen, a nula ako uvjet nije zadovoljen.

Neki algoritmi ne broje sve prolaze kroz ništicu nego samo rastuće ili padajuće te dobivaju ukupni broj prolaza množenjem s dva, s obzirom da za svaki rastući prolaz nužno mora postojati padajući prolaz.

Za jednokanalne signale ovaj parametar vrlo dobro odgovara dominantnoj frekvenciji signala te se zbog toga može koristiti u primitivnim algoritmima za detekciju visine tona. Prosječan broj prolaza kroz ništicu se također može koristiti u detekciji glasovne aktivnosti da bi se odredilo je li govorni segment zvučni ili bezvučni.

Da bi se testirao ovaj parametar zvučni je signal podijeljen na segmente trajanja 20 ms. Time se osigurava stacionarost signala. Za svaki od segmenata je potom izračunata prosječna vrijednost broja prolaza kroz ništicu te je izračunat prosječan broj prolaza kroz ništicu za čitav govorni i čitav glazbeni signal. Time su dobiveni sljedeći rezultati.



SLIKA 9. PROSJEČAN BROJ PROLAZA KROZ NIŠTICU ZA GOVOR I GLAZBU

S dobivenih se slika vidi da je prosječan broj prolaza kroz ništicu za govor značajno veći. S obzirom da je ovo jedna od najjednostavnijih značajki za izračunati, a da se njome može postići relativno velika točnost, ona se vrlo često koristi samostalno ili u kombinaciji s drugim značajkama koje se vrlo lako izračunavaju. Prosječna vrijednost broja prolaza za govor je također veća te iznosi 0.109 naspram prosječne vrijednosti za glazbu koja iznosi 0.0873.

Rezultati ovog testiranja poklapaju se očekivanim rezultatima. Naime, govor sadrži velik broj pauza, djelomice zbog razmaka između riječi, a djelomice zbog načina na koji izgovaramo određene skupove glasova. Tijekom tih pauza u govoru vrijednost signala je bliska ništici pa dominantna komponenta signala postaje šum. S obzirom na prirodu bijelog šuma, on češće mijenja predznak nego što to čini glazba. Zbog toga je vrijednost ZCR govora puno veći nego ZCR glazbe.

4.7 Modul razlike spektra i rekonstruiranog spektra iz kepstra

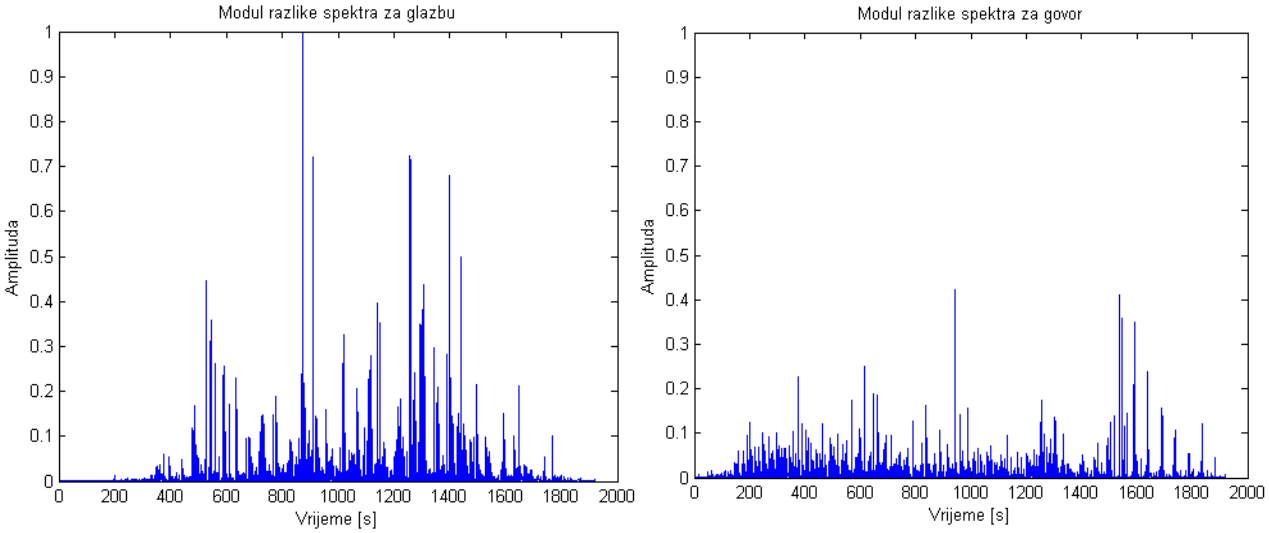
Modul razlike spektra i rekonstruiranog spektra iz kepstra (CRRM, *Cepstrum Resynthesis Residual Magnitude*) dobiva se korištenjem MFCC metode. MFCC metoda je objašnjena u jednom od prethodnih poglavlja. Tom se metodom izračunava spektar i kepstar snage u mel skali. CRRM se izračunava prema sljedećem izrazu.

$$CRRM = \sqrt{\sum_{k}(X[k] - Y[k])^2} \qquad (12)$$

Pritom $X[k]$ predstavlja amplitudu spektra u mel skali dok $Y[k]$ predstavlja usrednjene vrijednosti kepstralnih koeficijenata pomičnim prosjekom okolnih tri koeficijenta. Važno je napomenuti da i spektralni i kepstralni koeficijenti moraju biti izračunati u jednakom broju frekvencijskih pretinaca u mel skali.

Ovaj parametar nam govori koliko dobro možemo aproksimirati spektar iz kepstra te se pokazuje da je spektar bezvučnog govora moguće vrlo dobro aproksimirati iz kepstra. Vrijednosti ovog parametra puno su veće za glazbu i zvučni govor.

Zbog toga što ovaj parametar ne razlikuje govor u cjelosti od glazbe, teško ga je testirati u kontekstu snimki koje su korištene u sklopu ovog diplomskog rada. Međutim, govor kao cjelina, zbog svojih bezvučnih dijelova, pokazuje nešto manju vrijednost ovog parametra. To se može vidjeti na sljedećoj slici.



SLIKA 10. MODUL RAZLIKE SPEKTRA I REKONSTRUIRANOG SPEKTRA IZ KEPSTRA ZA GOVOR I GLAZBU

Ovaj se parametar često koristi u kombinaciji s drugim parametrima za određivanje zvučnosti signala.

4.8 Usporedba značajki

Slaney i Scheirer[4] su uspoređivali sve korištene značajke u temelju dva parametra, zauzeće procesorskog vremena i prosječna pogreška tijekom klasificiranja. Pritom su dobiveni sljedeći rezultati:

TABLICA 1. USPOREDBA ZNAČAJKI

Značajka	Interval	Zauzeće procesora	Pogreška
Modulacijska energija na 4 Hz	1 s	18%	12±1.7%
Postotak okvira niske energije	1 s	2%	14±3.6%
Frekvencija većinske spektralne snage	1 okvir	17%	46±2.9%
Varijanca frekvencije većinske spektralne snage	1 s	17%	20±6.4%
Spektralni centroid	1 okvir	17%	39±8.0%
Varijanca spektralnog centroida	1 s	17%	14±3.7%
Spektralni tok	1 okvir	17%	39±1.1%
Varijanca spektralnog toka	1 s	17%	5.9±1.9%
Broj prolaza kroz ništicu	1 okvir	0%	38±4.6%

Varijanca broja prolaza kroz ništicu	1 s	3%	18±4.8%
Modul razlike spektra i rekonstruiranog spektra iz kepstra	1 okvir	46%	37±7.5%
Varijanca modula razlike spektra i rekonstruiranog spektra iz kepstra	1 s	47%	22±5.7%
Pulsna metrika	5 s	38%	18±2.9%

Ono što se može lako uočiti jest da je za izračun broj prolaza kroz ništicu i okvira niske energije potrebno vrlo malo resursa procesora. Pritom broj prolaza kroz ništicu pokazuje podjednaku točnost kao i spektralni tok ili spektralni centroid, koji uključuju puno složeniji račun, a postotak okvira niske energije ostvaruje vrlo visoku točnost u odnosu na brojne druge parametre.

Pokazuje se da svaki parametar zasebno nema veliku točnost, jer se vjerojatnosti pogreške kreću od 15 do 40%. Međutim, nakon što je objavljen ovaj rad nastali su brojni drugi znanstveni radovi koji su kombinirali neke od ovdje opisanih radova da bi postigli optimalnu točnost trošeći minimalne resurse.

5. Klasifikacijske metode

Slaney i Scheirer[4] u svom radu testiraju četiri različite metode klasifikacije zvučnih signala iz izračunatih parametara. To su:

1. Višedimenzionalni Gaussov maximum a posteriori (MAP) estimator
2. Klasifikator modelom Gaussovih mješavina
3. Prostorno particioniranje temeljeno na k-d stablima
4. Dubinsko klasificiranje metodom najbližih susjeda

U nastavku će se redom objasniti ove metode klasikacije.

5.1 Višedimenzionalni Gaussov maximum a posteriori (MAP) estimator

Maximum a posteriori estimator je estimator točaka koji se temelji na uvjetnoj vjerojatnosti. Riječ je o estimatoru vrlo sličnom često korištenom estimatoru točaka, *maximum likelihood* (ML) estimatoru.

U sklopu analize parametara zvuka potrebno je postaviti određene pretpostavke radi jednostavnosti modela. Prilikom definiranja MAP modela za parametre zvuka pretpostavit će se da je riječ o nezavisnim varijablama, iako je velika većina parametara međusobno zavisno, i da je svaka varijabla distribuirana po Gaussovoj razdiobi. Tako definirani ulazni podaci omogućavaju definiranje MAP modela estimatora u trinaest dimenzija, jednom za svaki parametar.

Uz zadani diskretni niz podataka $\boldsymbol{X} = (x_1, \dots, x_n)$ i slučajnu varijablu θ definira se zajednička vjerojatnost, $p(\boldsymbol{X}, \theta)$. Kolmogorovljeva definicija povezuje zajedničku i uvjetnu vjerojatnost:

$$p(\boldsymbol{X}, \theta) = \frac{p(\theta|\boldsymbol{X})}{p(\boldsymbol{X})} \tag{13}$$

Općenito vrijedi Bayesov teorem za uvjetnu vjerojatnost:

$$p(\theta|\boldsymbol{X}) = \frac{p(\boldsymbol{X}|\theta)p(\theta)}{p(\boldsymbol{X})} \tag{14}$$

Cilj MAP estimatora [12] je odabrati idealnu srednju vrijednost parametra θ da bi se uz tako odabran θ mogla pouzdano estimirati slučajna varijabla X. Pritom MAP estimator definira tu vrijednost izrazom:

$$\theta_{MAP} = argmax_{\theta}\, p(\theta|\boldsymbol{X}) \tag{15}$$

Logaritmiranjem i korištenjem Bayesovog teorema nad izrazom za θ_{MAP} dobiva se sljedeće:

$$\begin{aligned} \theta_{MAP} &= argmax_{\theta}\, p(\theta|\boldsymbol{X}) \\ &= argmax_{\theta}\, ln[p(\theta|\boldsymbol{X})] \\ &= argmax_{\theta} ln\, \frac{p(\boldsymbol{X}|\theta)p(\theta)}{p(\boldsymbol{X})} \end{aligned} \tag{16}$$

Naime, logaritmiranje je u ovom slučaju dozvoljeno, zbog toga što je logaritamska funkcija monotona te ne mijenja lokaciju maksimuma, već samo vrijednost maksimuma. U potrazi za optimalnim θ vrijednost $p(\boldsymbol{X})$ se može zanemariti jer je konstantna u odnosu na θ, čime dobivamo:

$$\begin{aligned} \theta_{MAP} &= argmax_{\theta}\, ln[p(\theta|\boldsymbol{X})p(\theta)] = \\ &= argmax_{\theta}\{\ln[p(\boldsymbol{X}|\theta)] + \ln[p(\theta)]\} \\ &= argmax_{\theta}\, \{\ln[p(\boldsymbol{X}|\theta)]\} + \, argmax_{\theta}\, \{\ln[p(\theta)]\} \end{aligned} \tag{17}$$

U slučaju da je parametar θ distribuiran uniformnom distribucijom, izraz za θ_{MAP} tada prelazi u izraz istovjetan ML (*maximum likelihood*) estimatoru.

$$\theta_{MAP} = argmax_{\theta}\, p(\theta|\boldsymbol{X}) = argmax_{\theta}\, p(\boldsymbol{X}|\theta) = \theta_{ML} \tag{18}$$

U ovom je slučaju MAP estimator prikazan za samo jedan slučajni proces. Naravno, pri analizi zvuka se koristi veći broj parametara. Idućim će se primjerom[13] ilustrirati MAP estimator za dvije slučajne varijable, $\boldsymbol{X_1}$ i $\boldsymbol{X_2}$, definirane na sljedeći način:

$$\begin{aligned} \boldsymbol{X_1} &= \theta + \omega_1, \qquad p(\boldsymbol{X_1}|\theta) \sim G(\theta, \sigma_1^2) \\ \boldsymbol{X_2} &= \theta + \omega_2, \qquad p(\boldsymbol{X_2}|\theta) \sim G(\theta, \sigma_2^2) \end{aligned} \tag{19}$$

Može se definirati zajednička vjerojatnost $p(\boldsymbol{X_1}, \boldsymbol{X_2}|\theta) = p(\boldsymbol{X_1}|\theta) * p(\boldsymbol{X_2}|\theta)$

Uz pretpostavku da se i parametar θ distribuira po Gaussovoj razdiobi može se napisati sljedeći izraz, koji je analogan izrazu za jedan parametar $\boldsymbol{X}$.

$$p(\theta|\boldsymbol{X_1}, \boldsymbol{X_2}) = p(\boldsymbol{X_1}, \boldsymbol{X_2}|\theta) \tag{20}$$

Parametar θ se u ovom slučaju odabire prema sljedećem izrazu:

$$\theta_{MAP} = argmax_\theta\{p(\theta|\boldsymbol{X_1}, \boldsymbol{X_2})\} \tag{21}$$

Logaritmiranjem i korištenjem Bayesovog teorema dobiva se sljedeće:

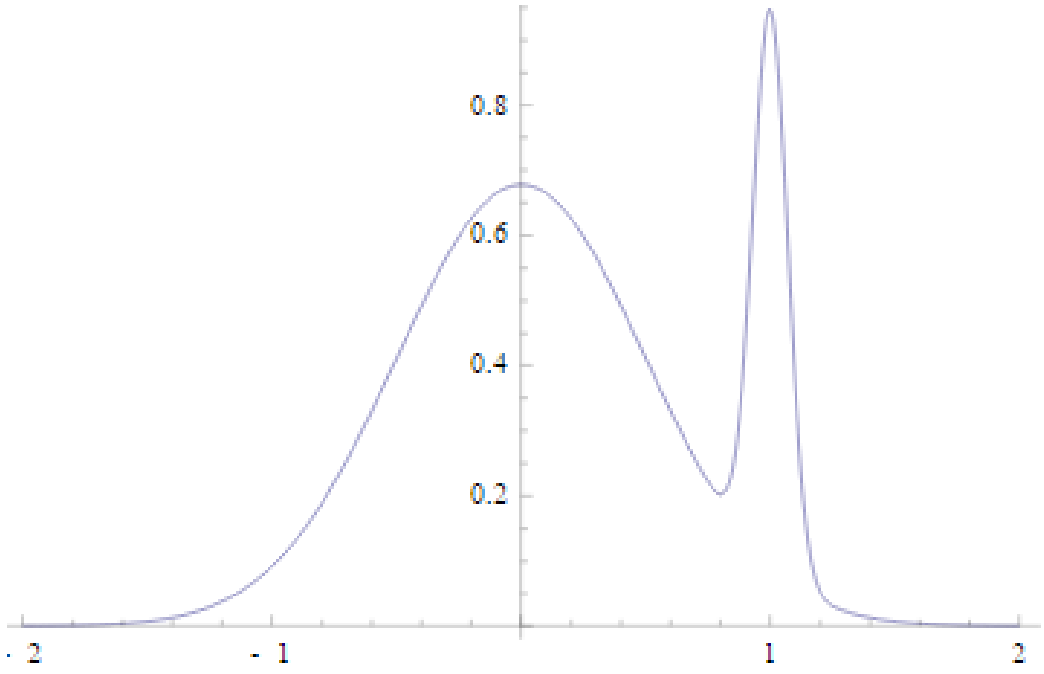
$$\begin{aligned} \theta_{MAP} &= argmax_\theta log \frac{p(\boldsymbol{X_1}, \boldsymbol{X_2}|\theta)p(\theta)}{p(\boldsymbol{X_1}, \boldsymbol{X_2})} \\ &= argmax_\theta\{\log[p(\boldsymbol{X_1}, \boldsymbol{X_2}|\theta)p(\theta)]\} \\ &= argmax_\theta\{\log[p(\boldsymbol{X_1}, \boldsymbol{X_2}|\theta)] + \log[p(\theta)]\} \end{aligned} \tag{22}$$

Kao i u prethodnom slučaju, zajednička vjerojatnost varijabli $\boldsymbol{X_1}$ i $\boldsymbol{X_2}$ ne utječe na izračun jer ne ovisi o θ, a logaritmiranje je monotono pa ne utječe na lokaciju maksimuma nego samo njegov iznos.

MAP estimator ima svoje prednosti i svoje mane. Glavna prednost mu je jednostavna izračunljivost i lako interpretiranje podataka. Uz broj točaka signala koji teži beskonačnosti ovaj estimator asimptotski teži prema ML estimatoru.

Glavni nedostatak MAP estimatora je to što estimira točke, odnosno, nema reprezentacije nesigurnosti parametra θ u estimaciji. Također, ponekad se radi o bimodalnim distribucijama vjerojatnosti, gdje je jedna točka značajno viša od ostatka, dok se velika većina vjerojatnosti smjestila dalje od te točke. Zbog toga taj maksimum netočno reprezentira gustoću vjerojatnosti kao cjelinu. Osim

toga, MAP estimator je varijantan tijekom reparametrizacije, što znači da se lokacija idealnog θ mijenja tijekom reparametrizacije.



SLIKA 11. PRIMJER BIMODALNE DISTRIBUCIJE VJEROJATNOSTI

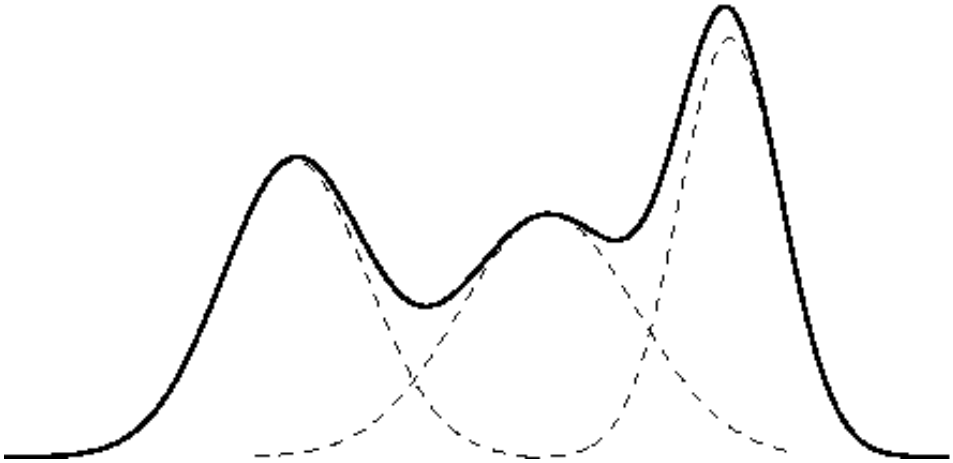
Slika br. 13. prikazuje distribuciju čiji maksimum se nalazi na mjestu koje ne odgovara mjestu gdje se nalazi najveći postotak distribucije, što je u ovom slučaju ništica, nego je riječ o izdvojenom šiljku na vrijednosti jedan.

5.2 Klasifikator modelom Gaussovih mješavina

Općenito, model mješavina[14] je vjerojatnosni model koji služi za reprezentiranje subpopulacija unutar čitavog skupa podataka na takav način da unutar subpopulacija nije potrebno razlikovati svaki zasebni podatak, već samo subpopulacije. Gustoća vjerojatnosti mješavine definira se izrazom:

$$p(\boldsymbol{X}) = \sum_{k=1}^{K} \pi_k \, p(\boldsymbol{X}|\boldsymbol{\theta_k}) \tag{23}$$

Pritom $p(\boldsymbol{X})$ predstavlja gustoću vjerojatnosti mješavine, π_k svaku od komponenti mješavine, a $p(\boldsymbol{X}|\boldsymbol{\theta_k})$ gustoću vjerojatnosti svake od komponenti.



SLIKA 12. MODEL GAUSSOVIH MJEŠAVINA

Kao što se vidi na slici br 14., svaka komponenta mješavine ima vidljiv utjecaj na ukupnu gustoću vjerojatnosti.

Općeniti model mješavina se sastoji od sljedećih komponenti.

- N slučajnih varijabli, od kojih svaka odgovara jednom od promatranih slučajnih procesa, gdje svaka od varijabli pripada istoj razdiobi, ali s različitim parametrima
- N odgovarajućih slučajnih latentnih varijabli, koje određuju identitet komponenti unutar mješavine, svaka od kojih je distribuirana prema K-dimenzionalnoj kategoričkoj razdiobi
- Set K težinskih koeficijenata, koji predstavljaju vrijednosti vjerojatnosti svake od mješavina, a čija je suma jednaka jedinici
- Set K parametara, svaki od kojih specificira parametar ili više parametara odgovarajuće komponente mješavine

Parametri se mogu matematički predstaviti izrazima:

$K - broj\ komponenti\ mješavine$

$N - broj\ promatranih\ slučajnih\ varijabli$

$\theta_{i=1\ldots K} - parametar\ distribucije$

$\varphi_{i=1\ldots K} - težinski\ koeficijenti\ svake\ mješavine$

$\Phi - K - dimenzionalni\ vektor\ sastavljen\ od\ vrijednosti\ \varphi_{i=1\ldots K} - \sum K = 1$

$z_{i=1\ldots N} - komponente\ svake\ promatrane\ slučajne\ varijable$

$x_{i=1\ldots N} - promatrane\ slučajne\ varijable$

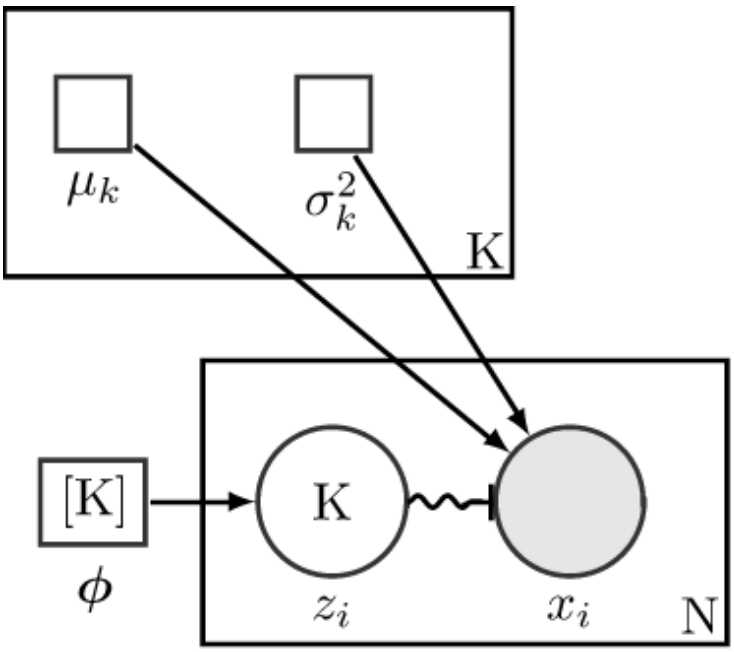
U slučaju Gaussovih mješavina dodaju dodatnih parametri:

$\mu_{i=1\ldots K} = očekivanje\ i - te\ komponente$

$\sigma^2_{i=1\ldots K} = varijanca\ i - te\ komponente$

$z_{i=1\ldots N} \sim Kategorički(\Phi)$

$x_{i=1\ldots N} \sim \boldsymbol{\mathcal{G}}(\mu_{z_i}, \sigma^2_{z_i})$



SLIKA 13. BLOK DIJAGRAM GAUSSOVIH MJEŠAVINA

Kvadrati na slici br. 15. predstavljaju fiksne parametre, očekivanje i varijancu za svaku komponentu mješavine, a kružnice predstavljaju slučajne varijable, x_i i z_i. Pritom ispunjeni krug označava poznate vrijednosti, odnosno vrijednost poznatih slučajnih varijabli, podataka. Vektor [K] označava k-dimenzionalni vektor težina svih mješavina.

U kontekstu analize zvukova, model Gaussovih mješavina modelira svaku od klasa podataka kao jednu komponentu mješavine. To se modeliranje obavlja algoritmom maksimiziranja očekivanja. Riječ je o metodi koja se često koristi za određivanje komponenti mješavina koje imaju *a priori* poznat broj komponenti.

Koristi se maximum likelihood estimator za svaku od mješavina, da bi se odredio optimalni model.

5.3 Prostorno particioniranje temeljeno na k-d stablima

Poseban slučaj binarnih stabala predstavljaju takozvana k-d (skraćenica za k-dimenzionalna) stabla[15]. Riječ je o strukturi podataka koja služi za organiziranje točaka unutar k-dimenzionalnog prostora. Ova je struktura podataka korisna za pretraživanje složenih višedimenzionalnih polja podataka. S obzirom da je zapravo riječ o binarnom stablu, za njega je određena složenost za operacije koje obavljaju nad stablom. U najgorem slučaju se ostvaruje složenost $O(n)$ za umetanje i brisanje čvorova, kao i za pretraživanje, dok se u prosjeku za sve tri operacije postiže složenost $O(logn)$.

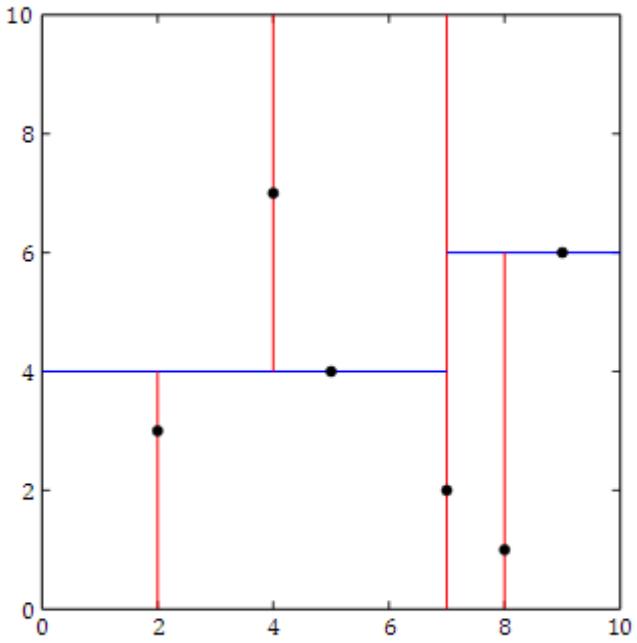
K-d stablo je binarno stablo u kojem je svaki čvor k-dimenzionalna točka. To znači da svaki čvor koji nije list implicitno dijeli hiperplohu na dva dijela, dva polu-prostora. Točke koje se nalaze u lijevom dijelu prostora se nalaze u lijevom dijelu stabla, a one koje se nalaze u desnom dijelu u desnom dijelu stabla. Matematički gledano, točke koje se nalaze lijevo od čvora koji je podijelio hiperplohu na dva dijela imaju vrijednost u odgovarajućoj dimenziji koja je manja od vrijednosti koju čvor ima. Analogno vrijedi za točke koje se nalaze u desnoj polovici, njihova vrijednost u odgovarajućoj dimenziji je veća od vrijednosti čvora koji je podijelio hiperplohu.

Postoje različite metode za konstruiranje stabla, zbog toga što postoje različiti načini da se odaberu osi po kojima se dijele hiperplohe. Kanonska metoda se vodi time da je rezultat balansirano stablo, gdje je svaki list podjednako udaljen od korijena stabla. Ovakva stabla, međutim, nisu optimalna za svaku primjenu. Kanonska metoda vodi se sljedećim pravilima:

- Silaskom niz stablom, ciklički se izmjenjuju osi kojima čvorovi prepolavljaju prostor. Primjerice, u trodimenzionalnom prostoru, korijen bi definirao x-os, njegova djeca y-os, njegovi unuci z-os, njegovi praunuci ponovno x-os itd.

- Od svih točaka koje se umeću u podstablo uzima se ona koja se nalazi najbliže srednjoj vrijednost svih točaka u odgovarajućoj dimenziji koja je odabrana prema prvoj natuknici

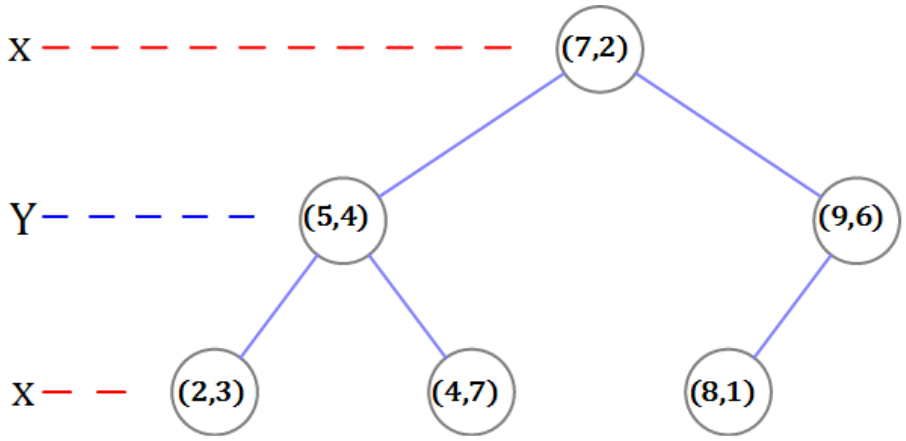
U slučaju da se ne uzima točka koja je najbliža srednjoj vrijednosti, stablo neće biti balansirano.



SLIKA 14. BALANSIRANO 2-D STABLO

Na slici br. 16. je prikazano stablo u 2-d prostoru sa šest točaka. U prvom koraku je cilj izabrati točku koja će podijeliti prostor po x-osi na dvije cjeline u kojima ima podjednak broj točaka. Vrijednosti točaka po x-osi su 2, 4, 5, 7, 8 i 9 te je njihov median šest. Algoritam je u prvom koraku odabrao točku (7,2) kao korijen stabla. U idućem se koraku prostor dijeli po y-osi. Unutar lijeve polovice se nalaze točke čije su vrijednost po y-osi 3, 4 i 7. Njihov je median četiri te je točka (5,4) izabrana kao korijen lijevog podstabla. Preostale dvije točke unutar lijevog podstabla se nalaze same unutar svojih polovica te one svaka dijele svoju polovicu. Unutar desnog podstabla postupak se analogno izvršava te je točka (9,6) izabrana da podijeli desnu polovicu, nakon čega se točka (8,1) nalazi sama unutar svoje polovice.

To se stablo može prikazati na sljedeći način:



SLIKA 15. K-D STABLO

Slika br. 17. prikazuje ono što je prethodno rečeno, prva je podjela napravljena po x-osi, zatim po y-osi i potom ponovno po x-osi. Iz stabla se vidi da je balansirano jer svi listovi imaju jednaku udaljenost do korijena.

U kontekstu obrade zvuka i klasifikacije zvučnih signala, k-d stabla se koriste za reprezentaciju parametara signala unutar višedimenzionalnog prostora te pretraživanje prostora u potrazi za područjima u kojima se nalazi velik broj točaka. Sustav se trenira nad setom podataka te se pronađeni skupovi točaka koriste u daljnjoj klasifikaciji zvukova.

5.4 Dubinsko klasificiranje metodom najbližih susjeda

Dubinsko klasificiranje metodom najbližih susjeda se naziva još i algoritam k-najbližih susjeda (*k-Nearest Neighbors*, k-NN)[16]. Ovaj se algoritam dijeli na dva načina, za regresiju i klasifikaciju. U oba se slučaja ulazni podaci sastoje od k najbližih podataka u određenom prostoru značajki. Izlazni podatak se, međutim, razlikuje.

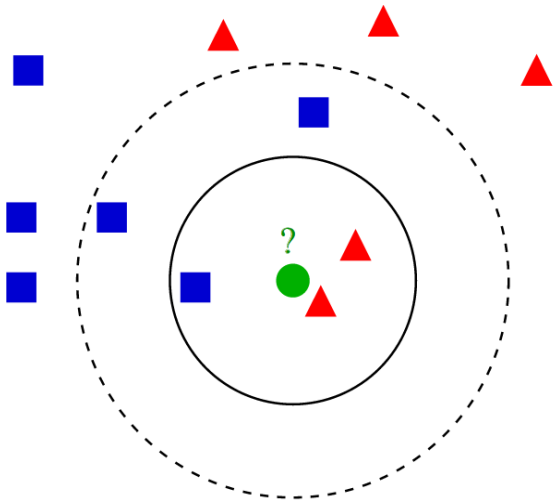
- Tijekom k-NN klasifikacije izlazni podatak je pripadanje kategoriji. Objekt se klasificira na temelju svojih k-najbližih susjeda u klasu koja je najčešća među tim susjedima. U ovom je slučaju k malen pozitivan broj. U slučaju da je $k = 1$ svaki se objekt klasificira na temelju jednog najbližeg susjeda.

- Tijekom k-NN regresije izlazni podatak je vrijednost objekta. Riječ je o srednjoj vrijednosti određenog parametra za k najbližih susjeda tog objekta.

k-NN algoritam je primjer lijenog učenja. Riječ je o porodici algoritama koji ne obavljaju eksplicitnu generalizaciju nekog pravila ili zadaće nego uspoređuju svaki novi problem s problemima koji su riješeni tijekom treniranja, a pohranjeni su u memoriji.

Svi susjedni objekti ne sudjeluju jednako u algoritmu, već se množe se težinama, gdje je težina recipročna vrijednost udaljenosti od polazišnog objekta, $1/d$. Susjedi su uvijek odabiru među objektima za koje se poznaje klasa (tijekom klasifikacije) i vrijednost objekta (tijekom regresije). Algoritam se ne trenira, nego značajke polazišnih objekata predstavljaju „trenirani“ dio algoritma.

Kao što je rečeno, svaki od objekata ima svoju klasu i svoju vrijednost. Vrijednost objekta predstavlja vektor svih značajki koje su bitne za klasifikaciju te je u praktički svim slučajevima nužno višedimenzionalan s obzirom da je klasifikacija objekata koristeći jednu značajku trivijalna. Pri izračunu udaljenosti koristi se euklidska metrika.



SLIKA 16. PROBLEM KLASIFIKACIJE OBJEKTA

Glavni problem ovog algoritma je što je osjetljiv na lokalnu strukturu podataka, odnosno, klasa objekata koja je češća u nekoj okolini ima tendenciju prevladati

nad svim ostalim klasama samo zbog toga što je na početku veći broj objekata bio klasificiran tom klasom. Ovaj se problem popravlja množenjem težinskim faktorima.

Slika br. 18. prikazuje drugi problem ovog algoritma, odabir idealnog parametra k. U slučaju da je $k = 3$, zeleni će objekt sa slike biti klasificiran kao crveni trokut jer su mu najbliži susjedi dva crvena trokuta i jedan plavi kvadrat. No, u slučaju da je $k = 5$, u okolinu zelenog objekta ulaze i dva plava kvadrata koji u ovom slučaju prevagnu i objekt se klasificira kao plavi kvadrat.

Općenito govoreći, idealni k ovisi o podacima koji se koriste. Velik k smanjuje utjecaj šuma na klasifikaciju, ali su zbog njega granice među klasama manje jasne. U praksi se idealna vrijednost k traži heurističkim metodama. Višedimenzionalni objekti značajno usporavaju izvođenje algoritma tako da se zbog toga prije k-NN algoritma uvijek obavlja izvlačenje najznačajnijih značajki iz višedimenzionalnih podataka.

U kontekstu klasifikacije zvukova, k-NN algoritam u pravilu ima dvije ili tri klase, govor, glazbu i tišinu, a u slučaju Slaneyja i Scheirera, objekti su 13-dimenzionalni.

5.5 Usporedba klasifikacijskih metoda

Slaney i Scheirer[4] su u svom radu testirali uspješnost klasifikacije s četiri prethodno opisane metode te se pokazuje da je u prosjeku podjednaka točnost u svim slučajevima. Međutim, pokazuje se da maximum a posteriori estimator i model Gaussovih mješavina puno bolje klasificiraju govor u odnosu na glazbu, dok se u slučaju metode k-dimenzionalnih stabala i k-najbližih susjeda radi o podjednakoj točnosti i za govor i za glazbu. Točnosti su prikazane u tablici na idućoj stranici.

TABLICA 2. USPOREDBA TOČNOSTI METODA

	Pogreška klasifikacije govora	Pogreška klasifikacije glazbe	Ukupna pogreška
Maximum a posteriori	2.1±1.2%	9.9±5.4%	6.0±2.6%
Model Gaussovih mješavina	3.3±1.3%	8.4±6.8%	5.8±2.5%
kNN uz k=1	4.3±1.7%	6.6±6.4%	5.5±3.6%
kNN uz k=1	4.2±1.9%	6.5±6.1%	5.4±3.5%
k-d stablo	5.4±1.1%	6.1±2.8%	5.8±1.6%

6. Klasifikator temeljen na efektivnoj vrijednosti signala i broju prolaza kroz ništicu

Klasifikator koji se temelji na samo dva parametra, efektivnoj vrijednosti signala i broju prolaza kroz ništicu, opisan je u radu *A Speech/Music Discriminator Based on RMS and Zero-Crossings*[17]. U razvoju ovog klasifikatora glavni zahtjevi su postavljeni na brzinu izvođenja algoritma i mogućnost obavljanja operacija u stvarnom vremenu. Upravo se zbog ovih ograničenja na algoritam koristi značajno reduciran set značajki u odnosu na značajke koje koriste ostali često korišteni klasifikatori. Kao što je prije rečeno, koriste se dvije značajke zvukova koje je najjednostavnije izračunati, a to su broj prolaza kroz ništicu i srednja vrijednost signala.

Klasifikacija se odvija u dvije faze. U prvoj fazi algoritam segmentira zadani zvučni zapis na temelju razdiobe efektivne vrijednosti signala i dobiva segmente za koje je potrebno utvrditi da li se radi o govoru ili glazbi. Algoritam koji obavlja segmentiranje zapisa je dizajniran na takav način da uvijek napravi više segmenata nego što je zaista potrebno. S obzirom da se čitav jedan segment uvijek klasificira u samo jednu kategoriju zapisa, ovakva presegmentacija služi tome da se kratki segmenti ne izgube u dužim segmentima i pogrešno klasificiraju.

Nakon što je signal uspješno segmentiran, sve se značajke računaju za intervale duljine 20 ms te je time određena preciznost ove metode. Rezultat izvođenja programa su indeksi početka i kraja svakog segmenta i klasa u koju je svrstan svaki segment.

Izvorni rad se hvali točnošću segmentacije u 97% slučajeva i ispravnom klasifikacijom segmenata u 95% slučajeva.

Klasifikacija se sastoji od čitanja zvučnog zapisa, izračuna značajki, segmentacije zapisa i klasifikacije svakog segmenta.

6.1 Čitanje zapisa i izračun značajki

Ovaj je klasifikator namijenjen za analizu monokanalnih zapisa, ali se može koristiti i za dvokanalne zapise. Radi izračunavanja značajki, prije čitanja snimke obavlja se čitanje prvih nekoliko uzoraka da bi se odredilo jer li zapis dvokanalni. Ako je zapis dvokanalni, koristit će se aritmetička sredina oba kanala za izračun značajki zvučnog zapisa.

S obzirom da klasifikator ponekad mora obrađivati jako velike zapise, čitanje i izračun značajki obavlja se u petlji koja čita interval duljine 1s, tako da ukupna duljina datoteke koju treba obraditi ne opterećuje radnu memoriju. Pročitana jedna sekunda se zatim dijeli na pedeset segmenata duljine 20 ms. Za svaki interval duljine 20 ms izračunava se efektivna vrijednost signala (*RMS*), broj prolaza signala kroz ništicu (*ZC*) i umnožak broja prolaza kroz ništicu i efektivne vrijednosti signala (*RMS*∗*ZC*). Izvođenjem petlje dobivaju se vrijednosti *RMS*, *ZC* i *RMS*∗*ZC* za čitav zvučni zapis u preciznosti od 20 ms. Osim tih značajki, za svaku sekundu signala izračunavaju se i parametri *a* i *b* koji opisuju generaliziranu χ^2 razdiobu vrijednosti *RMS*-a. Oni su definirani preko očekivanja i varijance *RMS*-a prema idućim izrazima.

$$a = \frac{\mu^2}{\sigma^2} - 1 \qquad\qquad b = \frac{\sigma^2}{\mu} \tag{24}$$

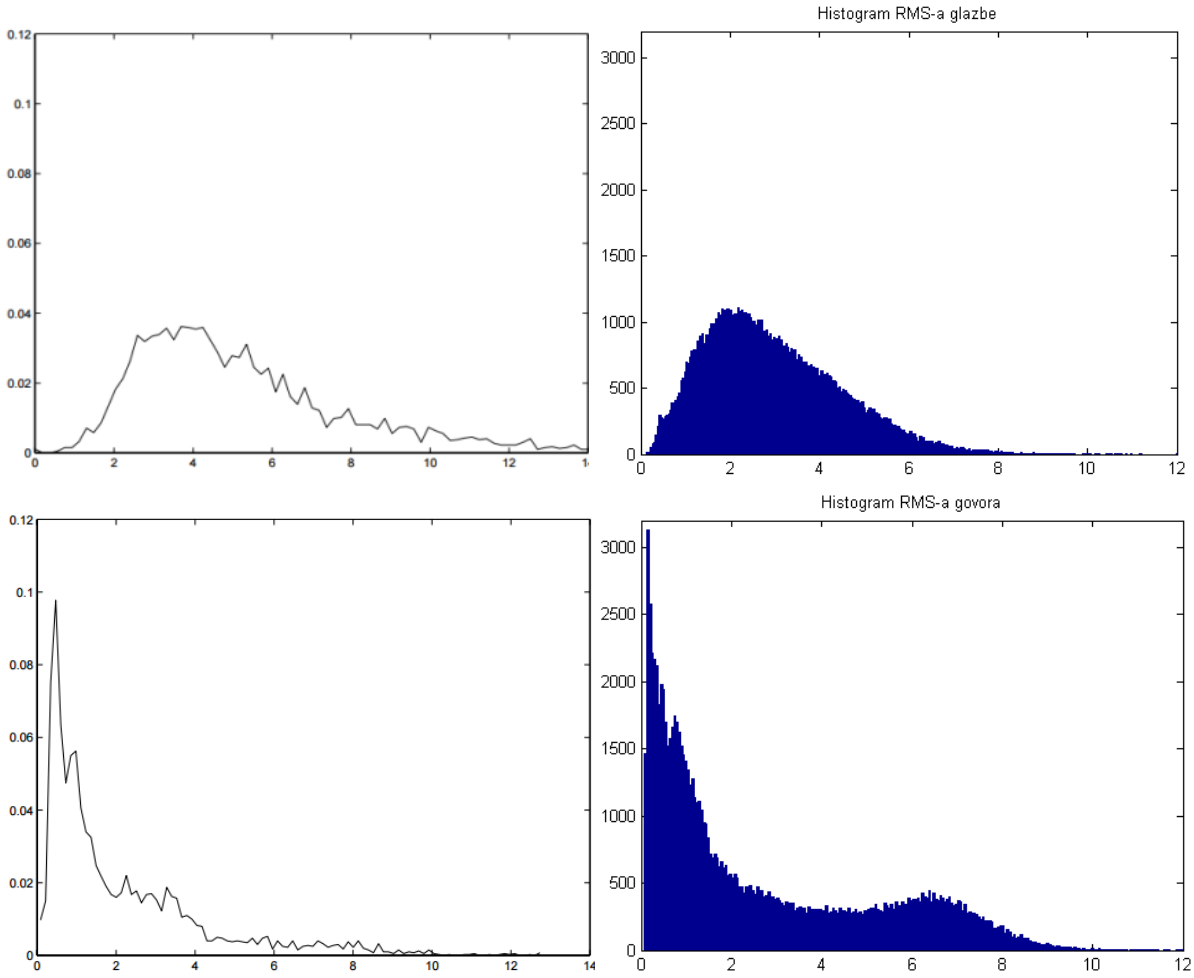
Uz tako definirane parametre *a* i *b*, funkcija gustoće vjerojatnosti opisana je sljedećim izrazom.

$$p(x) = \frac{x^a e^{-bx}}{b^{a+1}\Gamma(a+1)}, x \geq 0 \tag{25}$$

Govor i glazba imaju različitu razdiobu efektivne vrijednosti signala i zbog toga različite vrijednosti parametara distribucije. Upravo na toj razlici počiva ova metoda klasificiranja govora i glazbe. Razlika se može uočiti na histogramima za govor i glazbu koji se dobivaju iz odgovarajućih prethodno izračunatih *RMS*-ova.

Slika br. 19. prikazuje usporedbu histograma koji su dobiveni u okviru ovog diplomskog rada, posebno za svih 64 snimke govora i svih 64 snimke glazbe, s

histogramima koji su dobiveni u izvornom radu. Može se primijetiti da su kod snimki govora puno zastupljenije niže amplitude, dok su kod glazbe zastupljenije nešto više amplitude.

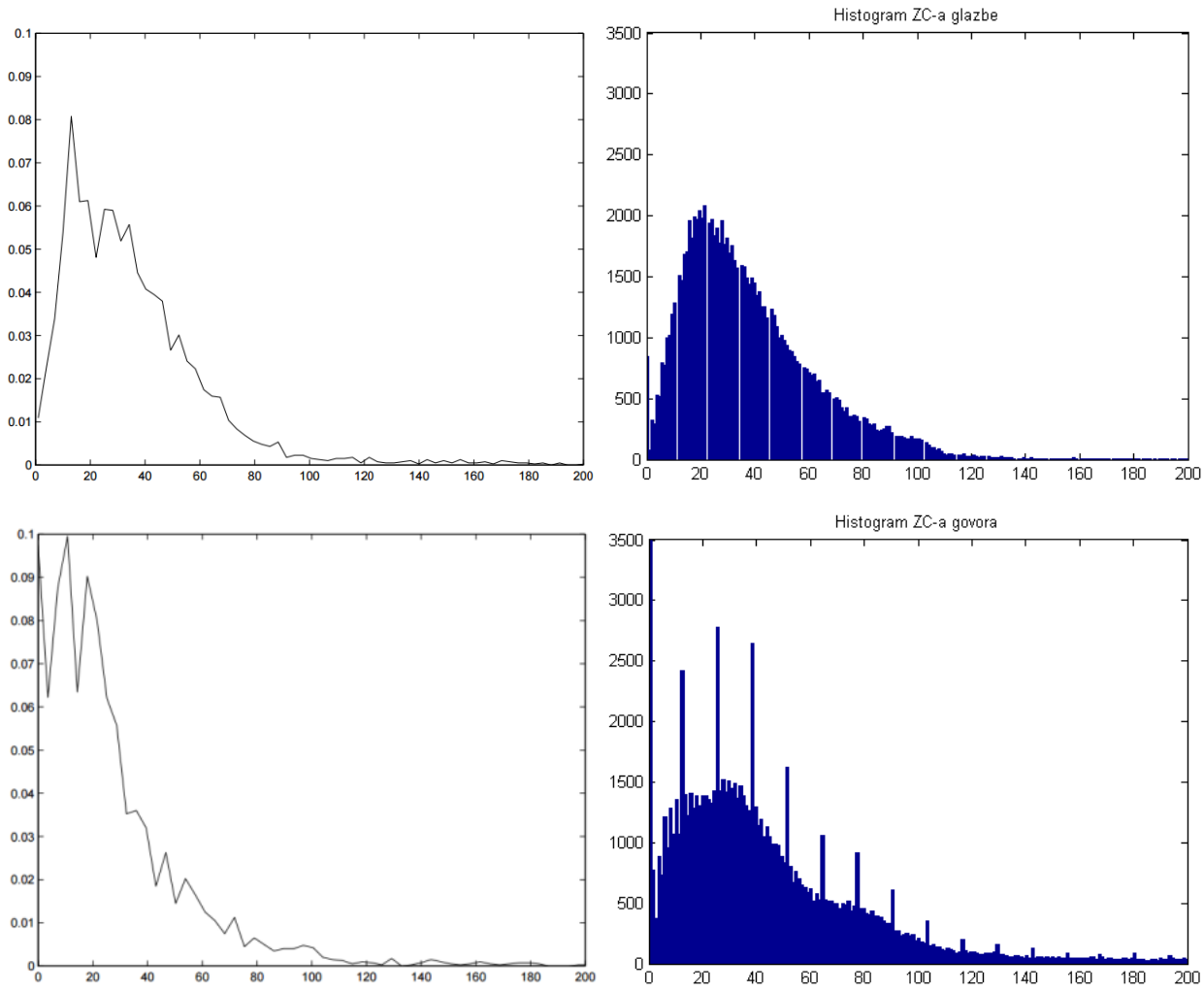


SLIKA 17. USPOREDBA DOBIVENIH HISTOGRAMA RMS-A ZA GOVOR I GLAZBU

Originalni su grafovi dobiveni koristeći posve drugačiji korpus glazbe i govora u odnosu na onaj koji se koristi u ovom diplomskom radu. Iz grafova se vidi da postoji vrlo velika sličnost odgovarajućih histograma. Stoga se može zaključiti da postoji vrlo velika korelacija između razdiobe efektivne vrijednosti signala i vrste zvučnog zapisa.

Druga korištena značajka je broj prolaza kroz ništicu (ZC) zato što se empirijski pokazalo da je nezavisna od RMS-a koristeći Blomquistovu mjeru i količinu zajedničke informacije. Pokazalo se da je nezavisnost snažnija u slučaju glazbe

jer se kod govora pojavljaju kratke pauze u kojima se koreliraju prolazi kroz ništicu i nizak RMS. Na slici br. 2 se vidi usporedba dobivenih histograma vezanih za broj prolaza kroz ništicu.



SLIKA 18. USPOREDBA DOBIVENIH HISTOGRAMA PROLAZA KROZ NIŠTICU ZA GOVOR I GLAZBU

I u ovom se slučaju vidi sličnost dobivenih razdiobi u odnosu na one dobivene u izvornom radu, iako ne u takvoj mjeri u kojoj je bilo vidljivo kod histograma RMS-a.

Nakon što su izračunate navedene značajke, algoritam može započeti sa segmentacijom zadanog zapisa.

6.2 Segmentiranje zapisa

Da bi se skratio i olakšao izračun, za segmentaciju se koriste isključivo vrijednosti RMS-a. Algoritam za segmentiranje se izvodi u dvije faze. U prvoj fazi algoritam izdvaja sekunde u kojima je došlo do predviđenog *'prijelaza'* iz govora u glazbu ili obrnuto. U drugoj fazi se unutar svake sekunde u kojoj je pronađen prijelaz traži u kojih se 20 ms prijelaz dogodio. Na ovaj način se štedi na vremenu koje je potrebno da se program izvede jer se detaljna analiza obavlja samo u sekundama u kojima se stvarno dogodio prijelaz. Preciznost od 20 ms je nužna i dovoljna jer su značajke ljudskog glasa stacionarne unutar intervala od 20 do 25 ms.

Zbog stalnih kratkotrajnih pauza u govoru, varijanca RMS-a je značajno veća od srednje vrijednosti RMS-a, dok je u slučaju glazbe varijanca RMS-a puno manja. Najtočniji način određivanja o kojoj se distribuciji radi bio bi računanje udaljenosti dobivene distribucije od idealiziranih distribucija za govor i glazbu. Međutim, taj bi izračun trošio previše resursa i vremena pa se iz idealiziranih distribucija izračunavaju idealizirane vrijednosti parametara a i b koji su prethodno definirani izrazima (24) i izračunati za svaku sekundu zadanog zvučnog zapisa.

Prethodno je pokazano da se koristeći parametre a i b može rekonstruirati funkcija gustoće vjerojatnosti $p(x)$. Za procjenu udaljenosti između okvira koristi se Bhattacharyyina udaljenost, koja ja definirana sljedećim izrazom.

$$\rho(p_1, p_2) = \int \sqrt{p_1(x)p_2(x)}dx \tag{26}$$

Uz tako definiranu mjeru sličnosti dvaju razdioba i χ^2 razdiobu dobiva se sljedeći izraz.

$$\rho(p_1, p_2) = \frac{\Gamma\left(\frac{a_1 + a_2}{2} + 1\right)}{\sqrt{\Gamma(a_1 + 1)\Gamma(a_2 + 1)}} * \left(\frac{2}{b_1 + b_2}\right)^{\frac{a_1+a_2}{2}+1} * b_1^{\frac{a_2+1}{2}} b_2^{\frac{a_1+1}{2}} \tag{27}$$

Izraz (27) pokazuje da su za izračun sličnosti bilo koja dva okvira potrebni samo odgovarajući *a* i *b* parametri, koji su pak dobiveni iz očekivanja μ i varijance σ^2 RMS-a svakog okvira.

Algoritam za segmentaciju traži onaj okvir duljine 20 ms, kod kojeg se prethodni okvir i idući okvir razlikuju značajno, odnosno, ako se dogodila promjena unutar okvira na indeksu i, onda se okviri na indeksima $i-1$ i $i+1$moraju razlikovati. Trenutačna promjena između okvira na indeksima i i $i+1$ se ne detektira ovim algoritmom. Zanimljivo je napomenuti da se u jednom trenutku računa sličnost susjednih okvira, ali se ona ne koristi u daljnjem računu i odmah je prebrisana vrijednošću sličnosti okvira na indeksima $i-1$ i $i+1$.

Što je veća sličnost među okvirima, to je manja vjerojatnost da je došlo do promjene među njima. Zbog toga se definira mjera vjerojatnosti promjene, $D(i)$. Pritom vrijedi izraz:

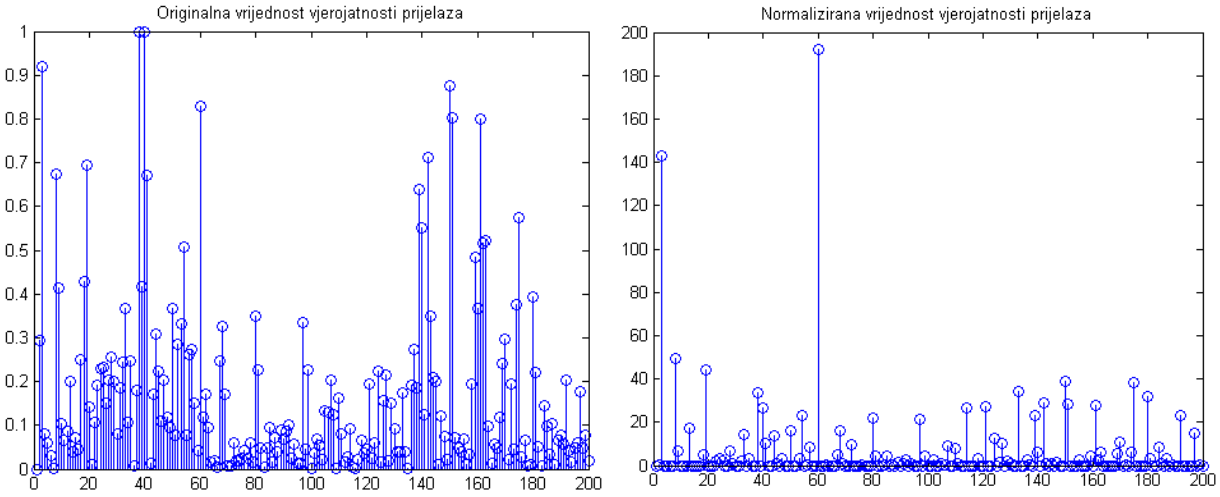
$$D(i) = 1 - \rho(p_{i-1}, p_{i+1}) \tag{28}$$

Ovaj je sustav dizajniran da detektira sve značajne promjene između govora, glazbe i tišine te će se ta promjena očitovati kroz vrijednost parametra $D(i)$. Lokalni maksimumi vrijednosti $D(i)$ detektiraju se empirijski dobivenim pragom.

Da bi se ispravno obavila detekcija promjene potrebno je normalizirati okvire u okolini okvira koji se promatra. Kodni odsječak koji obavlja normalizaciju promatra dva prethodna i dva iduća okvira u odnosu na trenutačni te određuje srednju i maksimalnu vjerojatnost promjene. Zatim se vrijednost vjerojatnosti promjene normalizira prema sljedećem izrazu.

$$D_n(i) = \frac{D(i)V(i)}{D_M(i)} \tag{29}$$

Pritom je $V(i)$ pozitivna udaljenost $D(i)$ do srednje vjerojatnosti promjene lokalnih pet okvira, dok je $D_M(i)$ maksimalna vrijednost vjerojatnosti promjene unutar istih pet okvira.



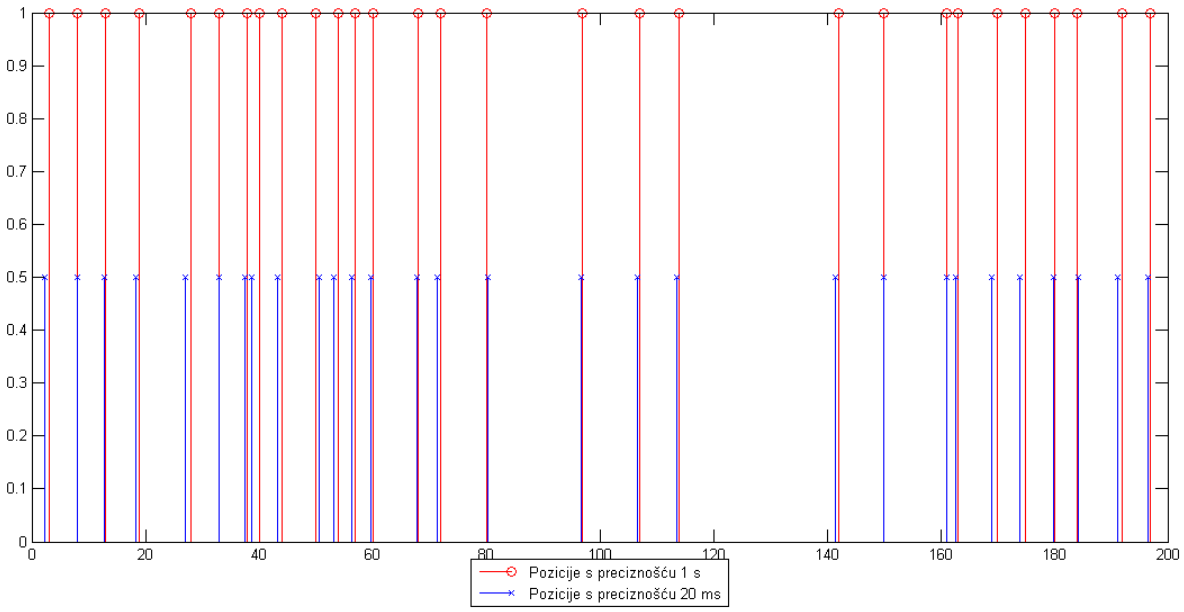
SLIKA 19. USPOREDBA ORIGINALNE I NORMALIZIRANE VRIJEDNOSTI VJEROJATNOSTI PRIJELAZA

Na slici br. 21. vidi se usporedba nenormalizirane i normalizirane vrijednosti vjerojatnosti prijelaza za niz od 200 okvira. Mogu se primijetiti dva šiljka na indeksima 38 i 40. Zbog toga što algoritam za normalizaciju promatra okolne vrijednosti vjerojatnosti promjene među okvirima, ovi šiljci nestaju nakon normalizacije. Šiljak na indeksu 60 oko sebe nema nijedan šiljak koji koji mu je blizak po vrijednosti pa je on prisutan i nakon normalizacije.

Bitno je napomenuti da ako program ne pronađe nijednu promjenu unutar zadanog zvučnog zapisa, završit će izvršavanje i neće biti određeno je li u zapisu govor i glazba. Nužan je barem jedan prijelaz za uspješno izvršavanje programa, makar taj prijelaz bio iz govora ponovno u govor ili iz glazbe ponovno u glazbu.

Iz dobivenih normaliziranih vjerojatnosti prijelaza označavaju se sve sekunde u kojima je došlo do prijelaza te započinje druga faza segmentacije. Unutar svake sekunde za koju je određeno da se u njoj dogodio prijelaz program obavlja identičan algoritam, ali ovaj put utvrđuje u kojih se točno 20 ms dogodio prijelaz.

Primjerice, za zapis duljine 1920 sekundi u prvoj fazi segmentacije nastaje vektor duljine 262 indeksa sekundi u kojima je detektiran prijelaz. Na taj se način u drugoj fazi samo 13.65% zapisa detaljno analizira jer se 1658 sekundi zapisa ne promatra s preciznošću od 20 ms. Ova ušteda omogućava višestruko brže izvođenje programa.



SLIKA 20. FAZE SEGMENTACIJSKOG ALGORITMA

Izvođenjem druge faze segmentacije dobivaju se precizni indeksi promjene unutar zapisa i točno vrijeme promjene. Na slici br. 22. se vidi usporedba vremena dobivenih u prvoj fazi algoritma i ispravljenih vremena dobivenih u drugoj fazi algoritma za segmentaciju.

Ovime je završena segmentacija zadanog zvučnog zapisa i započinje klasifikacija svakog od segmenata.

6.3 Parametri klasifikacijskog algoritma

Klasifikacija započinje izračunavanjem stvarnih značajki signala iz osnovnih karakteristika, broja prolaza kroz ništicu i efektivne vrijednosti signala. Prvi parametar je normalizirana varijanca RMS-a, definirana izrazom:

$$\sigma^2{}_{var} = \frac{\sigma^2}{\mu^2} \qquad (30)$$

Odnosno, varijanca se normalizira dijeljenjem s kvadratom srednje vrijednosti. Drugi parametar je vjerojatnost prolaza kroz ništicu zato što se primijetilo da je broj prolaza kroz ništicu puno veći u govorenih signala nego u glazbenih.

Treći parametar je zajednička mjera RMS-a i ZC-a koja se izračunava iz umnoška $RMS * ZC$. Predstavimo li taj umnožak oznakom A, izračunava se parametar C_Z:

$$C = \frac{\sum_{i=1}^{N} A(i)}{2A_x - A_n - A_m}, \quad C_Z = e^{-C} \tag{31}$$

Pritom A_x prestavlja maksimalnu vrijednost umnoška, A_n srednju vrijednost umnoška, a A_m srednju vrijednost umnoška. Normalizacija izrazom iz nazivnika obavlja se zato što nazivnik poprima veliku vrijednost u snimaka govora. To se događa zato što snimke govora sadrže puno više prolaza kroz ništicu pa su za njih vrijednosti A_n i A_m jako malene.

Četvrti parametar se naziva frekvencija intervala tišine i koristi se za razlikovanje glazbe od govora zato što je puno veći kod govora nego kod glazbe. Ovaj se parametar u praksi koristi za određivanje frekvencije slogova, ali u ovom slučaju je vrlo koristan jer poprima vrlo niske vrijednosti za glazbu.

Peti parametar je vezan za maksimalnu frekvenciju na kojoj se pojavljuju značajne komponente signala. Govorni signal je spektralno ograničen na frekvencije do oko 3500 Hz, dok se u glazbi pojavljuju i puno više frekvencije.

6.4 Klasifikacijski algoritam

Klasifikacijom se zadani segmenti dijele u tri kategorije. Prva je glazba, druga je govor, a treća je tišina i predstavlja dijelove snimke koji se ne mogu klasificirani ni kao govor ni kao glazba jer su ili pretihi ili se zaista radi o tišini.

Kao procjena amplitude signala koristi se težinski zbroj srednje i centralne vrijednosti RMS-a.

$$E = 0.7A_m + \frac{0.3}{N}\sum_{i=1}^{N} A(i) \tag{32}$$

Uz tako definiranu amplitudu, vjerojatnost da određeni segment prestavlja tišinu računa se izrazom:

$$p_{tišina} = \frac{6e^{-E^2}}{2 * 0.6 * 0.6} \tag{33}$$

S obzirom da se vrijednost RMS-a i govora i glazbe može opisati χ^2 razdiobom, klasifikator prvo provjerava sličnost razdbiobe RMS-ova u zadanom segmentu s prethodno definiranim empirijski dobivenim razdiobama govora i glazbe. Te su vjerojatnosti definirane sljedećim izrazima:

$$p_{glazba} = \frac{x^{a_{glazba}} * e^{\frac{-x}{b_{glazba}}}}{\Gamma\left(a_{glazba} + 1\right) * {b_{glazba}}^{a_{glazba}+1}} \tag{34}$$

$$p_{govor} = \frac{x^{a_{govor}} * e^{\frac{-x}{b_{govor}}}}{\Gamma\left(a_{govor} + 1\right) * {b_{govor}}^{a_{govor}+1}} \tag{35}$$

Pritom su a_{govor}, b_{govor}, a_{glazba} i b_{glazba} empirijski dobiveni idealizirani parametri χ^2 razdiobe za govor i glazbu, a x je normalizirana varijanca RMS-a na čitavom segmentu koji se klasificira.

Daljnji princip klasifikacije temelji se na modificiranju vrijednosti p_{glazba}, p_{govor}i $p_{tišina}$ u ovisnosti o parametrima klasifikacijskog algoritma.

Primjerice, govor karakterizira velika frekvencija intervala tišine. Ako je ta frekvencija manja od 0.6, a segment dovoljne duljine (u kodu definirano kao 2.5 s), onda je zasigurno riječ o glazbi te se postavlja vjerojatnost p_{glazba} u vrlo visokih 10, što osigurava da se taj segment označi kao glazba. Ovaj test uspješno klasificira oko 50% svih segmenata koji sadrže glazbu.

Empirijski određenim pragovima na isti se način podešavaju vrijednosti vjerojatnosti s obzirom na vjerojatnost prolaza kroz ništicu, maksimalnu frekvenciju i parametar C_Z.

S obzirom da čak i u ovom, najjednostavnijem, slučaju, algoritam zahtjeva korištenje korjenovanje i logaritmiranja, nije pogodan za prilagodbu za korištenje na DSP procesoru.

6.5 Rezultati klasifikacije

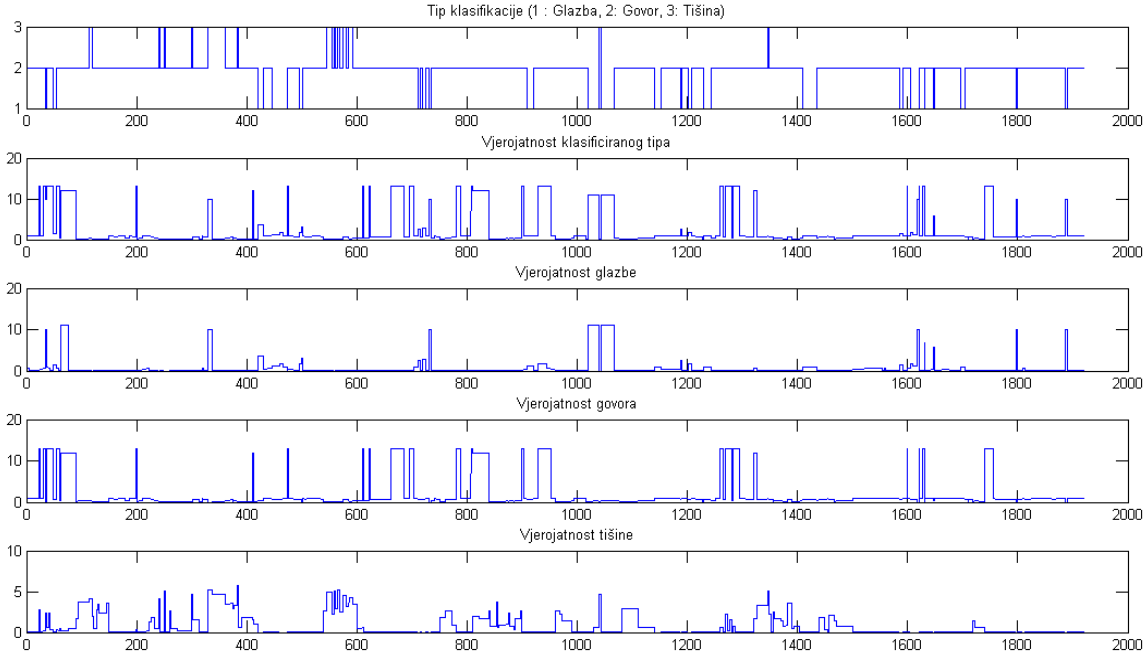
Koristeći navedeni algoritam na korpusu korištenom u sklopu ovog diplomskog rada dobiveni su sljedeći rezultati. Na snimkama govora postignuta je točna identifikacija 84.5% vremena, odnosno, točno je identificirano 82.66% analiziranih segmenata. Za glazbu je postignuta točna identifikacija 90.73% vremena, odnosno, 87.45%-tna uspješnost identifikacije segmenata.

S obzirom da se analiza preko pola sata glazbenih zapisa nije odredila nijedan segment koji predstavlja tišinu, moguće je smatrati segmente tišine kao segmente govora. U tom se slučaju točno identificiralo 88.31% segmenata, odnosno, 88.45% vremena.

6.5.1 Rezultati klasifikacije govora

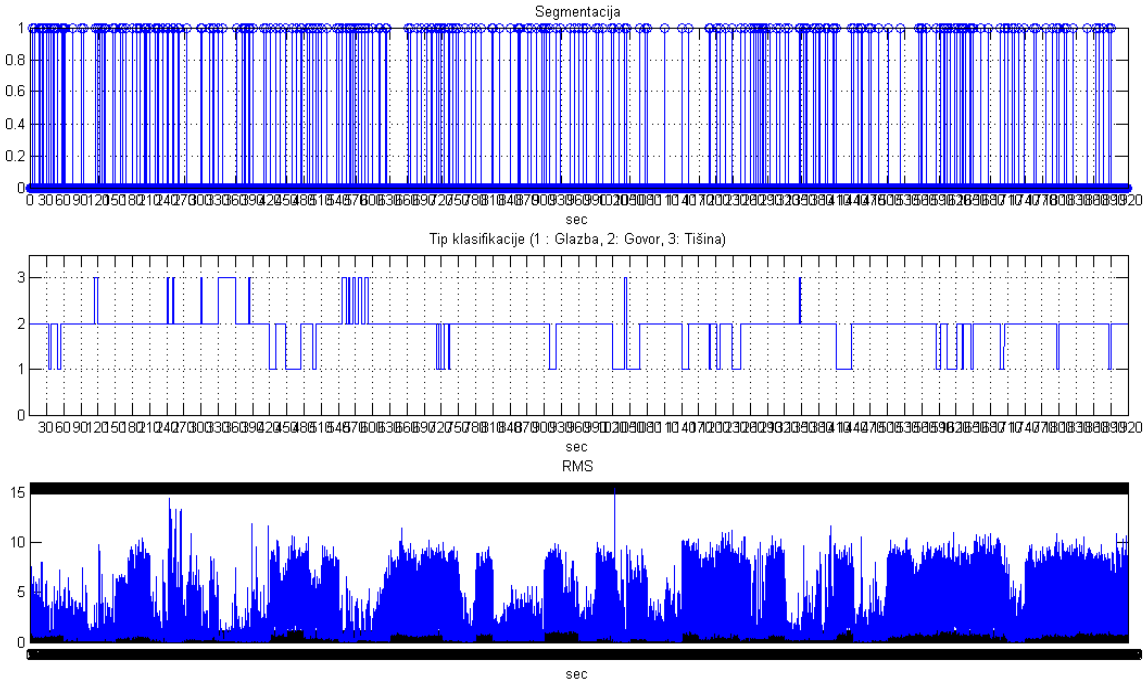
Segmentacijom 64 poluminutne snimke govora dobiva se 248 segmenata. Snimke su isključivo dijelovi radio zapisa na engleskom i njemačkom jeziku. Pritom se velika većina zapisa u potpunosti ispravno klasificira. Algoritam pogrešno klasificira govor kao glazbu u slučaju ljudskog smijeha, puštanja pozivatelja u eter, vrlo glasnih pozadinskih zvukova i u vrlo specifičnoj situaciji kad spiker imitira dječji glas, ali najviše pogrešnih klasifikacija događa se u slučaju ženskih govornika. Vrlo često se isti ženski govornik klasificira naizmjence kao govor i kao glazba, bez konkretnog razloga za različitu klasifikaciju jer je riječ o promjeni tijekom izgovora jedne riječi ili rečenice.

Slika br. 23. prikazuje rezultat procesa klasifikacije 32 minute govora. Prvi graf prikazuje da je velika većina zapisa označena kategorijom 2, govorom, dok postoje mali dijelovi koji su označeni kao tišina ili glazba. Drugi graf za svaki trenutak prikazuje onu vjerojatnost koja je u tom segmentu zapisa nadvladala preostale vjerojatnosti. Preostali grafovi prikazuju vjerojatnosti za govor, glazbu i tišinu i svakom trenutku.



SLIKA 22. VJEROJATNOSTI POJEDINIH TIPOVA NAKON KLASIFIKACIJSKOG ALGORITMA ZA GOVOR

Slika br. 24. prikazuje rezultat segmentacije i klasifikacije govora. Osim toga, prikazuje i vrijednost srednjeg RMS-a u svakom trenutku snimke. Može se primijetiti da vrlo nizak RMS gotovo uvijek uzrokuje da se određeni segment klasificira kao tišina.

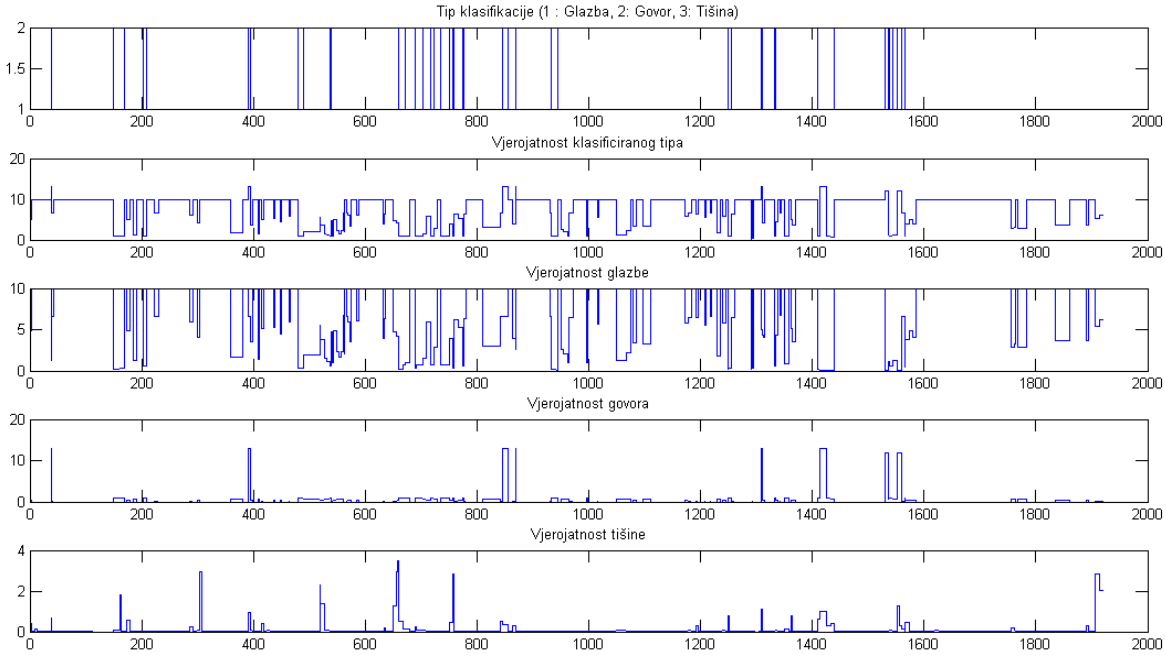


SLIKA 21. KONAČNI REZULTATI DVAJU FAZA ALGORITMA ZA GOVOR

6.5.2 Rezultati klasifikacije glazbe

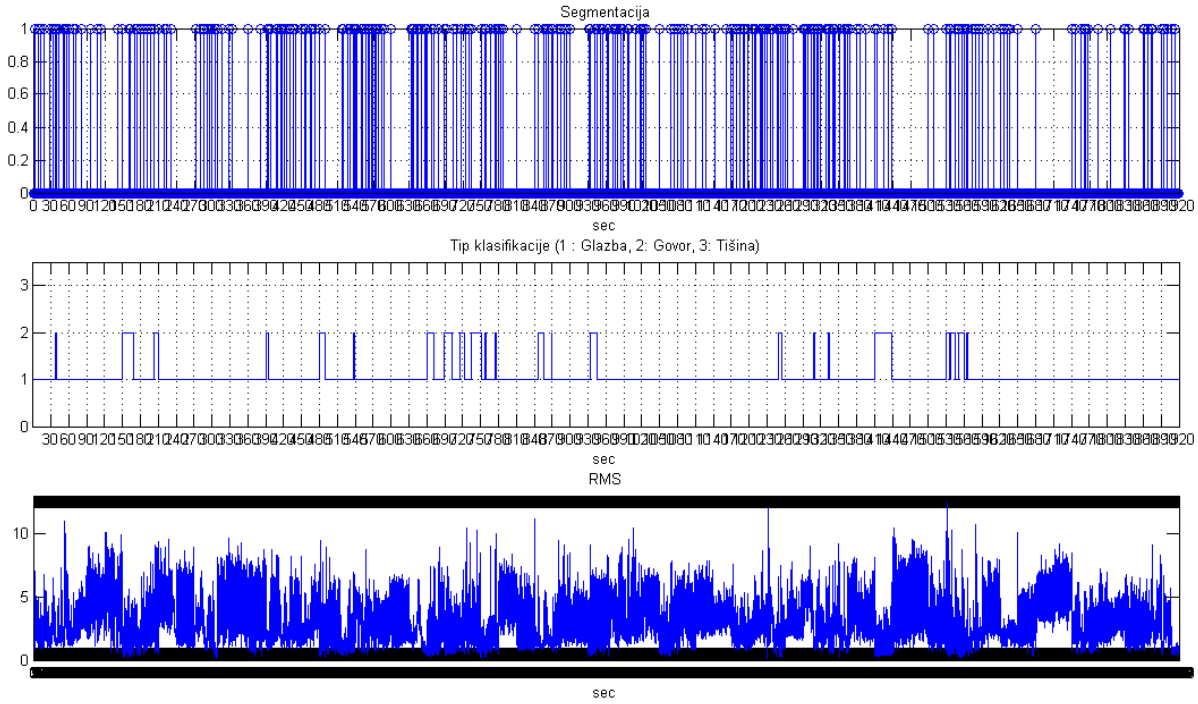
I glazbeni se korpus sastoji od 64 poluminutne snimke, ali je u ovom slučaju segmentacijom dobiveno 263 segmenta. S obzirom da je korpus glazbenih zapisa vrlo raznolik, zanimljivo je promotriti da li žanr glazbe utječe na klasifikaciju, posebice zato što dio zapisa uključuje vokalne izvedbe. Vokalne izvedbe škotske narodne glazbe, funka, rocka, bluesa, hip hopa, opere i techno glazbe su se u potpunosti ispravno klasificirale kao glazba. Unatoč očekivanjima, instrumentalni jazz i klasična glazba su najčešće pogrešno klasificirani kao govor. Glazbala koja izvode jazz tijekom sviranja proizvode specifičan šum koji računalo može zamijeniti za govor. Kod klasične glazbe u pravilu nije jasno zašto dolazi do pogrešne klasifikacije, zato što se ona događa na snimkama različitih instrumenata. Sakralna glazba, u kojoj se ljudski glas ističe, naizmjence se svakih par sekundi označava kao i govor i glazba.

Slika br. 25 prikazuje rezultat procesa klasifikacije 32 minute glazbe. Pritom je velika većina zapisa označena kao glazba, manji dio je označen kao govor, a kao što je rečeno, nijedan segment nije označen kao tišina.. Grafovi vjerojatnosti su analogni onima prethodno prikazanim za govor.



SLIKA 25. VJEROJATNOSTI POJEDINIH TIPOVA NAKON KLASIFIKACIJSKOG ALGORITMA ZA GLAZBU

Slika br. 26 prikazuje rezultat segmentacije i klasifikacije glazbe.



SLIKA 26. KONAČNI REZULTATI DVAJU FAZA ALGORITMA ZA GOVOR

Može se primijetiti da RMS nigdje ne poprima jako malene vrijednosti zbog kojih bi se zapis označio kao tišina, kao i puno manji broj pogrešno označenih zapisa.

7. Klasifikator temeljen na učenju i metodi k-NN

Biblioteka Matlab Audio Analysis opisana je u sklopu knjige *Introduction to Audio Analysis: A MATLAB Approach*[18]. Peto poglavlje te knjige bavi se klasifikacijom zvukova. Matlab odsječci vezani uz ovu knjigu javno su dostupni.[19]

7.1 Princip rada klasifikatora

Ova se biblioteka može koristiti za bilo koju vrstu klasifikacije zvukova, zbog toga što klasifikacija započinje treniranjem klasifikatora. Klasifikator se temelji na metodi učenja. Naime, prije nego što je moguće odrediti je li u nekom zapisu riječ o govoru ili o glazbi, potrebno je odrediti značajke govora i glazbe iz zapisa za koje se pouzdano zna da sadrže govor, odnosno glazbu. Za to se koristi k-NN klasifikator koji je prethodno obrađen u sklopu ovog rada. Funkcija kNN_model_add_class() koristi se zasebno za dodavanje po jedne klase u model kojim se želi obavljati klasifikacija. Za svaku se klasu definira direktorij koji sadrži datoteke koje se žele uvrstiti u tu klasu, ime klase, niz statistika i podataka o prozorima nad kojima se obavlja analiza.

U biblioteci je dostupno pet različitih modela koji su prethodno istrenirani. U svakom su slučaju definirane jednake širine prozora za analizu. Kratkotrajni prozori su trajanja 40ms bez preklapanja među prozorima, dok su veći prozori širine 2s s 50%-tnim preklapanjem. Za svaku od klasa iz zvučnih su zapisa izdvojeno šest statistika, srednja vrijednost, median vrijednosti, standardna devijacija, devijacija srednje vrijednost, minimum i maksimum.

Dostupni modeli su:

- model8, napravljen iz segmenata iz filmskih zapisa, koji se sastoji od osam klasa: glazba, govor, pucnjevi, vrisci, borbe, ostalo1, ostalo2 i ostalo3,
- model4, koji se sastoji od četiri klase: glazba, muški govor, ženski govor i tišina,
- modelMusicGenres3, koji se sastoji od tri klase: klasična glazba, jazz i elektronička glazba,

- modelSM, koji se sastoji od dvije klase: govor i glazba,
- modelSpeech, koji se sastoji od dvije klase: govor i negovor.

Naravno, s obzirom da je moguće generirati model za bilo koju vrstu klasifikatora, moguće je s testirati modele s brojnim drugim kategorijama.

Jednom kad je odabran klasifikacijski model, može se započeti s klasifikacijom. Na početku je potrebno normalizirati vektore koji sadrže zasebne značajke. To se čini tako da se za svaku značajku srednja vrijednost postavlja u ništicu, a standardna devijacija u jedinicu.

$$\tilde{v}_i(j) = \frac{v_i(j) - \mu_i(j)}{\sigma(j)}, \quad i = 1 \dots M, j = 1 \dots L \tag{36}$$

Pritom je M broj uzoraka koji su korišteni tijekom treniranja, L dimenzija prostora značajki, $\mu_i(j)$ srednja vrijednost j-te značajke, a $\sigma(j)$ standardna devijacija te značajke.

Značajke se izvlače iz signala funkcijom stFeatureExtraction.m koja obavlja ekstrakciju značajki nad kratkim intervalima, odnosno 40 ms. Za svaki se okvir izračunava vektor značajki duljine 35. Značajke su redom:

- Broj prolaza kroz ništicu
- Energija signala
- Entropija signala
- Spektralni centroid i njegova varijanca
- Spektralna entropija
- Spektralni tok
- Frekvencija većinske spektralne snage
- 13 kepstralnih koeficijenata
- Harmonijski omjer
- Osnovna frekvencija signala
- 12 koeficijenata Chroma vektora

Prilikom klasifikacije se ne obavlja segmentacija. Naime, funkcija FileClassification.m je zamišljena na način da klasificira čitav zvučni zapis u jedinstvenu kategoriju na temelju njegovih karakteristika. Za potrebe klasifikacije manjih dijelova signala postoji funkcija mtFileClassification.m koja dijeli zadanu snimku u intervale fiksnog trajanja među kojima nema preklapanja te svaki interval zasebno klasificira. To je trajanje definirano modelom kojim se želi izvršiti klasifikacija. U slučaju predefiniranih modela dostupnih u sklopu ove biblioteke, moguće je obavljati klasifikaciju u intervalima minimalne duljine od jedne sekunde.

Funkcija mtFileClassification funkcionira na način da prvo učita model koji se koristi, a zatim izračuna značajke signala. Iz signala se prvo izvlače značajke na intervalima duljine 40 ms te se potom preračunavaju u značajke za duže intervale, duljine 1s. Nakon što su značajke izračunate može započeti klasifikacija. Za nju se koristi funkcija classifyKNN_D_Multi.m koja koristi uzorke signala na danom intervalu, značajke signala na tom intervalu i konstantnu udaljenost k između objekata da metodom najbližih susjeda klasificira dani segment signala.

Za svaku se značajku signala redom izračunava udaljenost jednog segmenta u odnosu na preostale segmente. Prolaskom kroz, po udaljenosti najbliže, manje segmente traži se koja je kategorija signala najbrojnija u okolini te se u ovisnosti o većini uzoraka u segmentu klasificira čitav segment u jednu od kategorija modela.

7.2 Rezultati klasifikacije

Dostupni modeli ovog klasifikatora testirani su na *Marsyas* zvučnim zapisima. Jedini parametar pri analizi je k, odnosno, broj najbližih susjeda s kojima se uspoređuje svaki objekt. Iterativno se odredilo da se najbolji rezultati postižu uz $k = 3$ te je taj k korišten u svim modelima.

modelSM sadrži klase: govor i glazba te je testiran datotekama koje sadrže isključivo govor i isključivo glazbu. Dobiveni su rezultati prikazani u tablici br. 3.

Pokazuje se da je klasifikator obavio vrlo dobar posao prilikom klasifikacije govora jer je postignuta točnost od 92.13%, no čak i uz idealan k nije postignuta zadovoljavajuća točnost prilikom klasifikacije glazbe. Glavni uzrok toga je činjenica da dio zvučnih zapisa glazbe nisu isključivo instrumentalni nego sadrže vokalni dio koji se može krivo protumačiti kao govor. S druge strane, zvučni se zapisi analiziraju na intervalima duljine jedne sekunde, što onemogućava potpunu preciznost koja bi bila moguća na kraćim zapisima.

TABLICA 3. TOČNOST MODELSM KLASIFIKATORA

	Govor	Glazba
Klasificirano kao govor	92.13%	22.77%
Klasificirano kao glazba	7.87%	77.23%

Idući testirani model je modelSpeech koji također sadrži dvije kategorije, govor i negovor. Točnost ovog klasifikatora prikazana je tablicom 4.

TABLICA 4. TOČNOST MODELSPEECH KLASIFIKATORA

	Govor	Glazba
Klasificirano kao govor	89.84%	28.14%
Klasificirano kao negovor	10.16%	71.86%

Kao i u prethodnom slučaju, glavni razlog za razmjerno nisku točnost klasifikatora je činjenica da su intervali dugački i da među njima nema preklapanja.

Sljedeći testirani model je model4 koji sadrži četiri kategorije: glazbu, muški govor, ženski govor i tišinu. S obzirom da testirani zvučni zapisi sadrže i muški i ženski govor podjednako, obje kategorije se uvrštavaju pod govor prilikom testiranja. Njegova je točnost prikazana tablicom br. 5.

TABLICA 5. TOČNOST MODEL4 KLASIFIKATORA

	Govor	Glazba
Klasificirano kao govor	83.06%	19.18%
Klasificirano kao glazba	14.33%	71.08%
Klasificirano kao tišina	2.61%	9.74%

I u ovom slučaju je točnost relativno niska. Općenito govoreći, govor ima puno više razdoblja u kojima je riječ o tišini u odnosu na glazbu, no u slučaju ovog klasifikatora se to ne primjećuje, zbog toga što su pauze veće od jedne sekunde rijetke u govoru, a klasifikator može prepoznati isključivo takve pauze.

Posljednji testirani model je modelMusicGenres3 koji sadrži kategorije glazbe, klasičnu glazbu, jazz i elektroničku glazbu.

TABLICA 5. TOČNOST KLASIFIKACIJE ŽANRA GLAZBE

	Žanr	Klasificirani žanr
1	Škotska narodna glazba	Klasična glazba
2	Klasična glazba	Jazz
3	Klasična glazba	Klasična glazba
4	Jazz	Jazz
5	Rock	Elektronička glazba
6	Jazz	Klasična glazba
7	Jazz	Jazz
8	Rock	Jazz
9	Elektronička glazba	Elektronička glazba
10	Klasična glazba	Klasična glazba
11	Narodna glazba (udaraljke)	Jazz
12	Jazz	Jazz
13	Jazz	Jazz
14	Klasična glazba	Klasična glazba
15	Klasična glazba	Klasična glazba
16	Klasična glazba	Klasična glazba
17	Električna violina	Klasična glazba/Elektronička glazba
18	Klasična glazba	Klasična glazba
19	Jazz	Jazz

20	Jazz	Jazz
21	Rock	Jazz
22	Klasična glazba	Klasična glazba
23	Jazz	Jazz
24	Jazz	Jazz
25	Jazz	Jazz
26	Klasična vokalna glazba (zbor)	Klasična glazba
27	Jazz	Jazz
28	Jazz	Elektronička glazba
29	Klasična glazba (gitara)	Jazz
30	Klasična glazba	Klasična glazba
31	Elektronička glazba	Elektronička glazba
32	Jazz	Jazz
33	Narodna glazba	Klasična glazba
34	Klasična glazba (gitara)	Jazz
35	Rock	Klasična glazba
36	Klasična glazba (gitara)	Klasična glazba
37	Jazz	Jazz
38	Jazz	Jazz
39	Elektronička glazba	Elektronička glazba
40	Klasična vokalna glazba	Klasična glazba
41	Narodna glazba (harmonika)	Klasična glazba
42	Jazz	Klasična glazba
43	Klasična glazba (orkestar)	Klasična glazba
44	Klasična glazba	Klasična glazba
45	Jazz	Jazz
46	Jazz	Jazz
47	Narodna glazba (bliski istok)	Jazz
48	Jazz	Jazz
49	Narodna glazba (bliski istok)	Jazz
50	Jazz	Jazz
51	Jazz	Jazz
52	Klasična glazba (opera)	Elektronička glazba
53	Klasična glazba (opera)	Jazz
54	Jazz	Klasična glazba
55	Elektronička glazba	Elektronička glazba
56	Rock	Elektronička glazba
57	Rock	Elektronička glazba
58	Elektronička glazba	Elektronička glazba

59	Jazz	Klasična glazba
60	Jazz	Jazz
61	Rock	Klasična glazba
62	Jazz	Jazz
63	Klasična glazba	Klasična glazba
64	Klasična glazba	Klasična glazba

Uzimajući u obzir da dio žanrova koji se nalaze u testiranim glazbenim zapisima ni ne postoje u modelu koji je testiran, postignuta je prilično dobra točnost klasifikatora. Ovdje se pokazuje da klasifikator općenito obavlja prilično dobar posao u klasificiranju čitavih zvučnih zapisa, što mu i jest namijenjena svrha te da je glavni razlog relativnog neuspjeha prilikom klasifikacije govora i glazbe činjenica da obavlja klasifikaciju nad nezgrapnim vremenskim intervalom, predugim da se zadrži stacionarnost signala, a prekratkim da se može smatrati čitavom snimkom.

Zbog toga je napravljena ponovna analiza, u kojoj su se klasificirale čitave poluminutne snimke govora i glazbe, s istim modelima koji su prethodno korišteni. Ponovno se ustanovilo da je optimalan iznos parametra $k = 3$. Pritom su dobiveni dvojaki rezultati. Svi modeli su izuzetno dobro prepoznali snimke govora, ali su jako podbacili prilikom prepoznavanja glazbe. Sljedeći primjer prikazuje rezultate za modelSM.

TABLICA 6. TOČNOST KLASIFIKACIJE ČITAVIH SNIMKI

	Govor	Glazba
Klasificirano kao govor	96.88%	31.25%
Klasificirano kao glazba	3.12%	68.75%

Iz ovih i prethodnih rezultata može se zaključiti da ovaj klasifikator ima modele koji su trenirani na govoru koji je sličan onome što se testiralo, dok su testirani glazbeni segmenti različiti od onog čime je model treniran.

Zbog toga se postavlja idući korak u testiranju ovog klasifikatora, izrada novog modela za klasificiranje govora i glazbe. Polovicom se snimki trenira model klasifikatora dok preostale snimke služe za testiranje klasifikatora. Za kreiranje

novog modela koristi se funkcija kNN_model_add_class.m, a za klasifikaciju funkcija fileClassification.m.

Pritom su dobiveni sljedeći rezultati:

TABLICA 7. TOČNOST KLASIFIKACIJE ČITAVIH SNIMKI

	Govor	Glazba
Klasificirano kao govor	100%	14.06%
Klasificirano kao glazba	0%	85.94%

Savršena točnost prilikom klasifikacije govora nije iznenađujuća, s obzirom da su sve snimke govore radijske snimke sličnih govornika i slične kvalitete. S druge strane, upotrebom *Marsyas* snimki za treniranje klasifikatora povećana je učinkovitost klasifikatora nad preostalim *Marsyas* snimkama.

U slučaju daljnjeg povećanja broja snimki kojima se trenira klasifikator poveća se točnost, što prikazuje tablica br 8. U tom su slučaju govor i glazba trenirani sa 68% snimki.

	Govor	Glazba
Klasificirano kao govor	100%	10.00%
Klasificirano kao glazba	0%	90.00%

Pokazuje se da je učenjem i metodom k-NN moguće postići vrlo dobru točnost klasificiranja u slučaju kad je riječ o snimkama koje su prilično jednolične, kao što su radijski zapisi. Raznovrsni zapisi, kao što je glazba ostvaruju upitnu točnost prilikom ovakve klasifikacije.

Zaključak

Pouzdana klasifikacija govora i glazbe prije svega ovisi o primjeni. Sustavi koji zahtjevaju rad u stvarnom vremenu mogu s prilično velikom točnošću (85%) ovisiti o klasifikaciji korištenjem najjednostavnijih značajki zvuka, kao što su broj prolaza kroz ništicu i postotak okvira niske energije.

S druge strane, nešto veća točnost se postiže upotrebom klasifikatora koji izračunava veći broj značajki zvuka, ali takvi algoritmi troše puno više procesorskog vremena.

Pokazuje se da je klasifikator temeljen na učenju moguće lako podučiti da klasificira nove kategorije zvukova ispravno, kao i da poboljša klasificiranje zvukova koje konzistentno loše klasificira dodavanjem novih snimki u testni model. To je nemoguće napraviti kod klasičnog klasifikatora koji se temelji na provjeravanju fiksnih pragova za svaki od parametara i pokazuje se kao glavni nedostatak u odnosu na klasifikator temeljen na učenju.

Literatura

[1] J. E. Cutting, B. S. Rosner, *Categories and boundaries in speech an music**, Perception & Psychophysics Vol. 16 (3), 1974, str. 564-570

[2] C. T. Best, H. Hoffman, B. B. Glanville, *Development of infant ear asymmetries for speech and music*, Perception & Psychophysics Vol. 31 (1), 1982, str. 75-85

[3] J. Sanders, *Real-Time Discrimination of Broadcast Speech/Music*, IEEE International Conference on Acoustics, Speech, and Signal Processing (Volume:2), 1996, str. 993-996

[4] E. Scheirer, M. Slaney, *Construction and Evaluation of a Robust Multifeature Speech/Music Discriminator*, IEEE International Conference on Acoustics, Speech, and Signal Processing (Volume:2), Apr. 1997, str. 1331-1334

[5] M. J. Carey, E. S. Parris, H. Lloyd-Thomas, *A Comparison of Features for Speech, Music Discrimination.*., IEEE International Conference on Acoustics, Speech, and Signal Processing (Volume:1), Mar. 1999, str. 149-152

[6] George Tzanetakis, *Marsyas – Music Analysis*, Retrieval and Synthesis for Audio Signals, *Music Speech Data Set*, http://marsyasweb.appspot.com/download/data_sets/, 23. 3. 2015.

[7] J. Lyons, *Mel Frequency Cepstral Coefficients (MFCC) tutorial*, Practical Cryptography, http://practicalcryptography.com/miscellaneous/machine-learning/guide-mel-frequency-cepstral-coefficients-mfccs/, 26. 5. 2015.

[8] D. Ellis, *PLP and RASTA (and MFCC, and inversion) in Matlab using melfcc.m, invmelfcc.m*, 3. 9. 2012., http://labrosa.ee.columbia.edu/matlab/rastamat/, 26. 5. 2015.

[9] O. Lartilllot, P. Toiviainen, T. Eerola, MIRtoolbox, *Matlab Toolbox for Music Information Retrieval*, in C. Preisach, H. Burkhardt, L. Schmidt-Thieme, R. Decker (Eds.), Data Analysis, Machine Learning and Applications, Studies in Classification, Data Analysis, and Knowledge Organization, Springer-Verlag, 2008.

[10] J. M. Grey, J. W. Gordon, *Perceptual effects of spectral modifications on musical timbres*, The Journal of the Acoustical Society of America (Volume 63, Issue 5), 1978, str. 1493

[11] T. Giannakopoulos, *Silence removal in speech signals*, http://www.mathworks.com/matlabcentral/fileexchange/28826-silence-removal-in-speech-signals, 6. 6. 2015.
[12] I. Ivek, *Probabilistic Formulations of Nonnegative Matrix Factorization*, http://across.fer.hr/_download/repository/KDI,_Ivan_Ivek.pdf, 13. 6. 2015.
[13] W. Chen, X. Z. Fang, Y. Cheng, *A maximum a posteriori super resolution algorithm based on multidimensional Lorentzian distribution*, Journal of Zheijang University SCIENCE A (Volume 10, Issue 12), Dec. 2009, str. 1705 - 1713
[14] *Mixture model*, https://en.wikipedia.org/wiki/Mixture_model, 15. 6. 2015.
[15] *k-d tree*, https://en.wikipedia.org/wiki/K-d_tree. 13. 6. 2015.
[16] *K-nearest neighbours algorithm*, https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm, 16. 6. 2015.
[17] C. Panagiotakis & G. Tziritas, *A Speech/Music Discriminator Based on RMS and Zero-Crossings*, IEEE Transactions on Multimedia (Volume:7, Issue:1), Feb. 2005, str. 155 - 166
[18] T. Giannakopoulos, A. Pikrakis, *Introduction to Audio Analysis: A MATLAB Approach,* First Edition, Elsevier, 2014.
[19] T. Giannakopoulos, Matlab Audio Analysis Library, 10. 3. 2014.

METODE RAZLIKOVANJA GOVORA I GLAZBE U DIGITALNIM ZVUČNIM ZAPISIMA

Sažetak

U sklopu ovog diplomskog rada obrađen je povijesni pregled na metode razlikovanja govora i glazba te je napravljen pregled različitih čimbenika koji utječu na uspješnost klasifikacije. To su prije svega parametri zvuka, u vremenskoj domeni postotak okvira niske energije i broj prolaza kroz ništicu, u frekvencijskoj domeni spektralni centroid, spektralni tok, frekvencija većinske spektralne snage te u kepstralnoj domeni modulacijska energija na 4 Hz i modul razlike spektra i rekonstruiranog spektra iz kepstra. Rad se zatim fokusira na različite klasifikatore koje je moguće koristiti u sklopu razlikovanja govora i glazbe te obrađuje četiri različita klasifikatora. Na kraju se diskutiraju dvije različite gotove metode koje se mogu koristiti za klasificiranje govora i glazbe.

Ključne riječi: digitalna obrada zvuka, razlikovanje govora i glazbe, broj prolaza kroz ništicu, spektralni tok, spektralni centroid, MFCC koeficijenti, MAP estimator, Gaussove mješavine, k-d stabla, k-NN algoritam.

METHODS FOR SPEECH AND MUSIC DISCRIMINATION IN DIGITAL SOUNDTRACKS

Abstract

In the first part of this Masters' Thesis there is a historical overview of the methods for Speech/Music Discrimination and a survey of various factors that affect the classification performance. These factors consist primarily of parameters of sound. In the time domain, they include percentage of low-energy frames and zero crossing rate, in the frequency domain there are spectral centroid, spectral flux, spectral rolloff point and in the cepstral domain there are 4 Hz modulation energy. The Thesis then focuses on different classifiers that can be used for speech/music discrimination, pinpointing four different classifiers. Finally, two different methods that can be used to classify speech and music are discussed.

Key words: digital sound processing, speech/music discrimination, zero-crossing rate, spectral flux, spectral centroid, MFCC coefficients, the MAP estimator, Gaussian mixture, k-d trees, k-NN algorithm.

Privitak

Funkcija za slučajan odabir zvučnih zapisa

```
clc;
clear all;
close all;

music_dire = 'music_wav\';
speech_dire = 'speech_wav\';
music_pre = 'glazba (';
speech_pre = 'govor (';
suf = ')';
broj_zapisa = 10;

podaci = zeros(128,661500);
podaci1 = zeros(64,661500);

for i = 1:64;
    file_name = sprintf('%s%s%d%s.wav', music_dire, music_pre, i,
suf);
    %disp(file_name);
    [podaci(i,:), fs(1,:), nbits(1,:)] = wavread(file_name);
    %disp(max(abs(podaci(i,:))));
    podaci(i,:) = podaci(i,:) ./ max(abs(podaci(i,:)));
    %disp(max(abs(podaci(i,:))));
end

% for i = 0:63
%     for j = 1:661500
%         glazba(i*661500+j) = podaci((i+1), j);
%     end
% end
% wavwrite(glazba, fs, nbits, 'glazba.wav');

for i = 1:64;
    file_name = sprintf('%s%s%d%s.wav', speech_dire, speech_pre, i,
suf);
    %disp(file_name);
    [podaci(i+64,:), fs(1,:), nbits(1,:)] = wavread(file_name);
    [podaci1(i,:), fs(1,:), nbits(1,:)] = wavread(file_name);
    %disp(max(abs(podaci(i+64,:))));
    podaci(i+64,:) = podaci(i+64,:) ./ max(abs(podaci(i+64,:)));
    podaci1(i,:) = podaci1(i,:) ./ max(abs(podaci1(i,:)));
    %disp(max(abs(podaci(i+64,:))));
end

% for i = 0:63
%     for j = 1:661500
%         govor(i*661500+j) = podaci1((i+1), j);
%     end
% end
% wavwrite(govor, fs, nbits, 'govor.wav');
% wavwrite(glazba, fs, nbits, 'glazba.wav');
% clear podaci1;
% clear glazba;
% clear govor;
```

```
pokusaj = 0;
brojac = 0;
while(brojac~=0.5*broj_zapisa)
    brojac = 0;
    izbor = randi(128, broj_zapisa, 1);

    % figure;
    % plot(izbor, 'x');
    % hold on;
    % y=zeros(1,30);
    % for i = 1:30
    %     y(i) = 64;
    % end
    % plot(y);


    for i = 1:broj_zapisa
        if izbor(i)>64
            brojac = brojac+1;
        end
    end
    pokusaj = pokusaj+1;
    % disp(pokusaj);
end

% disp('Broj odabranih govorenih datoteka:');
% disp(brojac)
% disp('Broj odabranih glazbenih datoteka:');
% disp(30-brojac);

izab_podaci = zeros(1, broj_zapisa*661500);
for i = 0:(broj_zapisa-1)
    for j = 1:661500
        izab_podaci(i*661500+j) = podaci(izbor(i+1), j);
    end
end

clear podaci;

wavwrite(izab_podaci, fs, nbits, 'odabrani_podaci.wav');
wavwrite(izab_podaci, fs, nbits, 'Metode\odabrani_podaci.wav');
% Data clipping warning jer vrijednost podataka
% koji se pišu mora biti -1.0 <= y < +1.0 pa se +1.0 clippa
% u najbližu vrijednost manju od +1.0

ID = fopen('izbor.txt','w');
fprintf(ID, '%2f \n', izbor);
fclose(ID);

clear all;
```

Funkcija za skaliranje zapisa snagom

```
clear all;
close all;
clc;

[govor, fs, nbits] = wavread('govor.wav');
[glazba, fs, nbits] = wavread('glazba.wav');

snaga_govor = (1./length(govor)) * sum(govor.^2);
snaga_glazba = (1./length(glazba)) * sum(glazba.^2);

razmjer = snaga_govor/snaga_glazba;
glazba = glazba*sqrt(razmjer);

snaga_govor = (1./length(govor)) * sum(govor.^2);
snaga_glazba = (1./length(glazba)) * sum(glazba.^2);

wavwrite(glazba, fs, nbits, 'glazba_skalirana_snagom.wav');
wavwrite(govor, fs, nbits, 'govor_skaliran_snagom.wav');
```

Funkcija koja služi za razdvajanje zvučnog segmenta na segmente trajanja 20 ms (na primjeru izračuna karakteristike za zero-crossing rate

```
clear all;
close all;
clc;

[govor, fs, nbits] = wavread('govor.wav');
[glazba, fs, nbits] = wavread('glazba.wav');


zcr_go_uk = sum(abs(diff(govor>0)))/length(govor);
zcr_gl_uk = sum(abs(diff(glazba>0)))/length(glazba);

go = zeros(96000, 441);
gl = zeros(96000, 441);
go_zcr = zeros(96000,1);
gl_zcr = zeros(96000,1);

for i = 1:96000
    go(i,:) = govor( ((i-1)*441)+1 : ((i-1)*441)+441 );
    gl(i,:) = glazba( ((i-1)*441)+1 : ((i-1)*441)+441 );

    go_zcr(i) = sum(abs(diff(go(i,:)>0)))/length(go(i,:));
    gl_zcr(i) = sum(abs(diff(gl(i,:)>0)))/length(gl(i,:));

    disp(i/96000 * 100);
end

figure;
plot(go__zcr,'r');
hold on;
plot(gl_zcr,'b');
```

```
os_glazba = linspace(1,96000,96000);
os_govor = linspace(1,96000,96000);
figure;
plot(os_govor./50, go__zcr, 'b');
title('Prosječan broj prolaza kroz ništicu za govor');
xlabel('Vrijeme [s]');
ylabel('Amplituda parametra');
figure;
plot(os_glazba./50, gl_zcr, 'b');
title('Prosječan broj prolaza kroz ništicu za glazbu');
xlabel('Vrijeme [s]');
ylabel('Amplituda parametra');
axis([0 2000 0 0.8]);
```

Funkcija za poziv rastamat MFCC koeficijenata

```
clc;
close all;

[d,sr] = wavread('glazba.wav');
samples = d(1:length(d)/50);

wintime = 0.025;
hoptime = 0.010;
numcep = 100;
lifterexp = 0.6;
sumpower = 1;
preemph = 0.97;
dither = 0;
minfreq = 0;
maxfreq = 100;
nbands = 150;
bwidth = 1.0;
dcttype = 2;
fbtype = 'mel';
usecmp = 0;
modelorder = 0;
broaden = 0;
useenergy = 0;

if preemph ~= 0
  samples = filter([1 -preemph], 1, samples);
end

 [pspectrum,logE] = powspec(samples, sr, wintime, hoptime, dither);

[aspectrum,wts,binfrqs] = audspec(pspectrum, sr, nbands, fbtype,
minfreq, maxfreq, sumpower, bwidth);

if (usecmp)
  aspectrum = postaud(aspectrum, maxfreq, fbtype, broaden);
end
```

```
if modelorder > 0

  if (dcttype ~= 1)
    disp(['warning: plp cepstra are implicitly dcttype 1 (not ',
num2str(dcttype), ')']);
  end

  lpcas = dolpc(aspectrum, modelorder);
  cepstra = lpc2cep(lpcas, numcep);

else
    cepstra = spec2cep(aspectrum, numcep, dcttype);
end

cepstra = lifter(cepstra, lifterexp);

if useenergy
  cepstra(1,:) = logE;
end
```

Funkcija za izdvajanje segmenata trajanja 60 msn na primjeru izračuna frekvencije većinske energije signala

```
clear all;
close all;
clc;

[govor, fs, nbits] = wavread('govor.wav');
govor = govor(17640001:(17640000+66150));

for i = 1:50
    go(i,:) = govor( ((i-1)*1323)+1 : ((i-1)*1323)+1323 );
    filename = sprintf('%s%d.wav', 'dat\gov',i);
    wavwrite(go(i,:), fs, nbits, filename);
    pom = specgram(filename);
    r(i) = mirrolloff(filename, 0.95);
    r(1,i)
end
```