

June 17, 2024

## 1 Prvi zadatak

### 1.1 Postavka

1. Korišćenjem komercijalnog mikrofona u programskom okruženju MATLAB, snimiti govornu sekvencu u dužini od 20-ak sekundi. Sekvencu snimiti sa frekvencijom odabiranja 8 ili 10 kHz i ona treba da se sastoji desetak jasno segmentiranih reči.
2. Korišćenjem kratkovremenske energije i kratkovremenske brzine prolaska kroz nulu izvršiti određivanje početka i kraja pojedinih reči. Dobijeni rezultat prikazati grafički. Preslušati segmentirane delove zvučne sekvence i komentarisati dobijeni rezultat. (Po želji se ovaj postupak može ponoviti primenom Teager energije).
3. Snimiti novu sekvencu od par reči (bogatih samoglasnicima, recimo onomatopeja...) i na osnovu tako snimljene sekvence proceniti pitch periodu sopstvenog glasa. Koristiti dve različite metode pa uporediti i komentarisati dobijene rezultate.

### 1.2 Teorijski uvod

Jedna od značajnih karakteristika govornog signala jeste **kratkovremenska energija** (short-time signal energy). Ukupna energija se definiše sledećim izrazom:

$$E = \sum_{k=-\infty}^{\infty} x^2[k]$$

Međutim, ovako definisana veličina ne nosi nikakvu informaciju o pojavi zvučnih i bezvučnih signala, kao i o njihovoj vremenski promenljivoj prirodi. Odatle potreba da se definiše kratkovremenska energija, koja se računa kao zbir kvadrata  $N$  poslednjih odbiraka iz intervala  $(n-N+1)$  do  $n$ -tog odbirka:

$$E_n = \sum_{k=n-N+1}^n x^2[k]$$

Ovo je zapravo identično uvođenju prozorske funkcije koja ima vrednost 1 za opseg  $0 \leq n \leq N-1$ , a 0 za ostale vrednosti, pa se izraz može napisati i na sledeći način:

$$E_n = \sum_{k=-\infty}^{\infty} x^2[k]w[n-k]$$

gde  $w$  predstavlja prozorsku funkciju. Na sledećoj slici prikazan je tipičan izgled kratkovremenske energije.

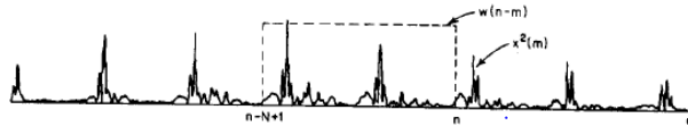


Figure 1: Tipičan oblik kratkovremenske energije

Za prozorsku funkciju se najčešće bira Hammingova:  $w[n] = 0.54 - 0.46\cos(2\pi/(N-1))$  za  $0 \leq n \leq N-1$ , odnosno 0 za ostale vrednosti, ili pravougaona prozorska funkcija:  $w[n] = 1$  za  $0 \leq n \leq N-1$ , odnosno 0 za ostale vrednosti. Što se tiče dužine prozora  $N$ , važno je da ona ne bude suviše mala, recimo reda dužine pitch periode ili manje, jer to dovodi do značajnog oscilovanja kratkovremenske energije zavisno od pojedinih oscilacija u vremenskom obliku signala. Takođe,  $N$  ne treba biti ni suviše veliko, reda veličine nekoliko pitch perioda, jer će za posledicu kratkovremenska energija biti isuviše spora i neće nositi informacije o značajnim promenama u obliku signala. Uobičajeno je da se uzima da je dužina prozorske funkcije 1 do 3 pitch periode, pa dobijamo sledeće relacije:

$$NT = (1 \div 3) \Rightarrow N = (1 \div 3) \frac{T_{pitch}}{T} = (1 \div 3) \frac{F_s}{F_{pitch}}$$

Uzimajući u obzir da se pitch frekvencija ljudskog glasa kreće između 120Hz (kod muškaraca) i 300Hz (kod dece), i da je tipična frekvencija odabiranja 10kHz, znamo da je dobar izbor za dužinu prozorske funkcije  $N \in [100, 200]$ .

Drugi parametar koji dobro oslikava sadržaj snimljenog signala je **brzina prolaska signala kroz nulu**. Ova veličina je zapravo mera frekvencijskog sadržaja signala, pa ukoliko je signal bogat visokim učestanostima, ova mera će biti veća i obrnuto. Brzina prolaska  $Z$  definiše se kao:

$$Z = \frac{N_0}{N}$$

gde je  $N$  broj posmatranih odbiraka, a  $N_0$  broj odbiraka koji nisu istog znaka kao njihovi prethodnici. Na primer, ukoliko posmatramo prostoperiodičnu komponentu učestanosti  $F_0$  koja je odabirana sa učestanošću  $F_s$ , tokom jedne periode  $T_0$  imaćemo  $N = \frac{T_0}{T_s} = \frac{F_0}{F_s}$ , pri čemu će signal dva puta promeniti znak, pa je brzina prolaska kroz nulu  $Z$ :

$$N = \frac{2}{N} = \frac{2F_0}{F_s}$$

Međutim, govorni signal sadrži frekvencijske komponente iz širokog spektra učestanosti, pa sama vrednost mere  $Z$  nije tako precizna. Zbog ovoga se koristi kratkovremenska forma ova mere:

$$Z_n = \sum_{k=-\infty}^{\infty} \frac{1}{2} |sgn(x[k]) - sgn(x[k-1])| w[n-k]$$

gde je  $sgn$  funkcija znaka, a  $w[n]$  prozorska funkcija definisana kao:  $w[n] = \frac{1}{2N}$  za  $0 \leq n \leq N-1$ , a u suprotnom 0. Kada ovu meru primenjujemo na govorne signale polazi se od pretpostavke da je zvučni deo signala na učestanostima oko 3kHz, dok je bezvučni deo i pozadinski šum na višim učestanostima. Zbog ovoga možemo očekivati da je mera  $Z$  za zvučni signal značajno niža u odnosu na bezvučni. Na sledećoj slici je prikazan histogram parametra  $Z$  za zvučni i bezvučni signal kao i njihove odgovarajuće Gausovske funkcije gustine verovatnoće.

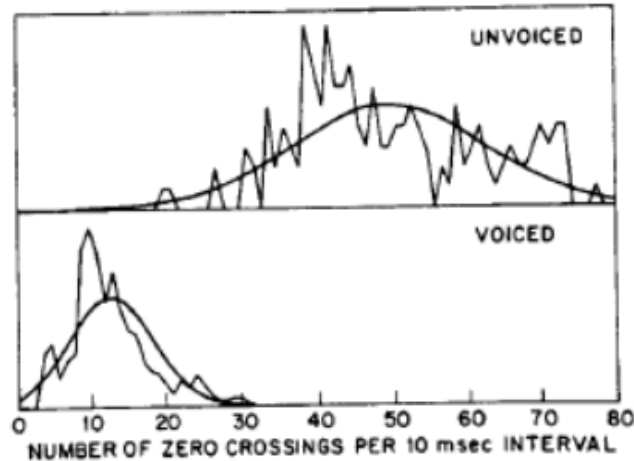


Figure 2: FGV brzine prolaska kroz nulu za zvučne i bezzvučne segmente govornog signala

Za bezzvučni signal, srednja vrednost parametra  $Z$  je 49 prolaska kroz nulu na periodu od 10ms, dok je za zvučni samo 14. Iako je razlika očigledna, varijanse su velike i funkcije gustine verovatnoće se preklapaju, pa  $Z$  ne može biti jedini parametar po kome se vrši segmentacija na zvučni i bezzvučni deo govornog signala. Jedan od glavnih problema za rešavanje prilikom obrade govornog signala je izdvajanje korisnog, govornog dela od raznih vrsta šumova koji su prisutni. Ono što dodatno otežava ovaj problem je i postojanje reči koje počinju ili se završavaju frikativima (f,h) ili slabim plozivima (p,t,k), ili reči koje se završavaju nazalima (m,n,nj). Snaga ovih glasova ne leži u očekivanom opsegu, i pri tome je prilično mala, pa se često teško izdvaja od pozadinskog šuma. Algoritam koji rešava ovaj problem koristi kombinaciju kratkovremenske energije i kratkovremenske brzine prolaska kroz nulu signala i prikazan je na sledećoj slici.

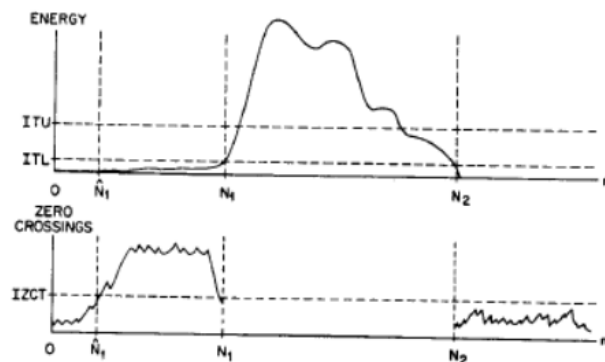


Figure 3: Primer kratkovremenske energije i kratkovremenske brzine prolaska kroz nulu u cilju određivanja početka i kraja govornog signala

Prvo što određujemo je ITU koji predstavlja gornji prag za koji ćemo biti sigurni da ako je energija signala veća od njegove vrednosti, znamo da je u tom intervalu sigurno govorni signal. Međutim, s obzirom da je prag uglavnom dovoljno veliki, znamo i da je govor započeo i pre nego što je KVE postala veća od ITU, i traje i nakon što je KVE opala ispod ITU. Pored gornjeg praga biramo i donji prag ITL, koji će biti manje konzervativan i pomoći će nam da približnije odredimo početak i kraj govornog signala. Od trenutka kada je KVE postala veća od ITU iterativnim postupkom se pomeramo ulevo u vremenu i određujemo trenutak  $N_1$  kada KVE prvi put postane manja od ITL-a i na sličan način, od trenutka kada je KVE postala manja od ITU iterativnim postupkom se pomeramo udesno u vremenu i određujemo trenutak  $N_2$  kada je KVE postala

manja od ITL-a. Ova dva trenutka predstavljaju naš inicijalni početak i kraj reči. U cilju finog podešavanja inicijalnog početka reči  $N_1$  posmatra se interval signala STZCR 25 frejmova (prozorskih funkcija) levo od tačke  $N_1$ . Ukoliko u tom intervalu signal STZCR više od tri puta preseče usvojeni prag (između 20 i 30) tada se  $N_1$  koriguje u  $\hat{N}_1$ , u tačku gde je prag prvi put presečen. Analogno se sprovodi postupak za prag  $N_2$ , s tim što je  $\hat{N}_2$  tačka poslednjeg preseka.

Estimacija pitch periode je jedan od najvažnijih zadataka u obradi govornog signala. Metode za procenu pitch periode su brojne, a ovde će biti izložene dve: Metoda paralelnog procesiranja i Autokorelaciona metoda.

Metodu paralelnog procesiranja predložili su Rabiner i Gold 1969. godine. Metoda je oslikana na sledećem grafiku.

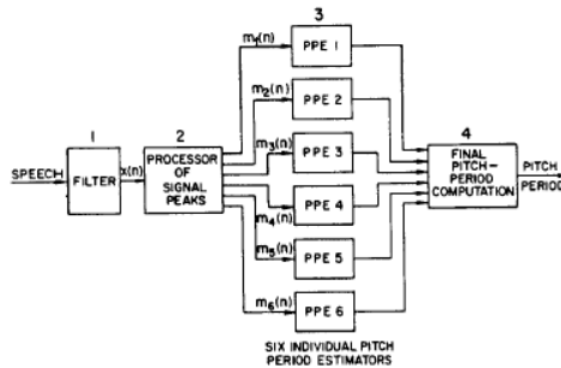


Figure 4: Šema za estimaciju potch periode

Blok 1 predstavlja filtar čiji je propusni opseg obično  $[0-900]$ Hz, koji je prikladan s obzirom da očekujemo da se pitch perioda nađe u intervalu od 100 do 300Hz. Blok 2 je procesor pikova u signalu, koji na izlazu generiše 6 različitih povorki impulsa. Dakle, na osnovu filtriranog signala na ulazu, na izlazu se formira šest signala  $m_i(n)$  definisanih na sledeći način:

- $m_1(n)$  sadrži impulse jednake vrednostima maksimuma originalnog signala na lokacijama u kojima su se maksimumi pojavili
- $m_2(n)$  sadrži impulse jednake vrednostima razlike maksimuma i prethodnog minimuma na mestu gde se pojavio maksimum
- $m_3(n)$  sadrži impulse intenziteta jednakim razlici maksimuma i prethodnog maksimuma na mestima gde se pojavio tekući maksimum
- $m_4(n)$  sadrži impulse intenziteta jednakim negativnim vrednostima minimuma na pozicijama gde se minimum pojavio
- $m_5(n)$  sadrži impulse jednakim negativnoj razlici tekućeg minimuma i prethodnog maksimuma, na lokaciji pojave prvog minimuma
- $m_6(n)$  sadrži impulse intenziteta negativne razlike rekućeg minimuma i prethodnog minimuma na lokaciji tekućeg

Nijedan od ovih signala ne može imati negativnu vrednost, pa se u skučaju dobijanja negativne vrednosti upisuje nula. Blok broj 3 su estimatori pitch periode PPE, koji na osnovu ulaznih signala generišu sopstvenu procenu pitch periode. Njihov rad je prikazan na sledećoj slici.

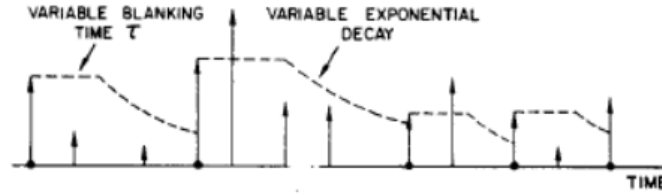


Figure 5: Rad estimator PPE

Estimator od prvog impulsa kreće da broji vreme, i ne radi ništa sve dok ne prođe 'blanking time'  $\tau$ , a nakon isteka ovog vremena, generiše funkciju  $Ae^{-\lambda t}$ , gde je vrednost A zapravo vrednost intenziteta impulsa od koga je počelo brojanje, a parametar  $\lambda$  se bira tako da PPE ne može da generiše pitch periodu veću od očekivane. Blanking time  $\tau$  se bira obrnuto, tako da pitch perioda ne bude manja od očekivane. Poslednji blok 4 donosi konačnu odluku o estimaciji pitch periode tako što prikuplja sve pojedinačne procene i na njih primenjuje medijanu.

Još jedna česta metoda za procenu pitch periode koristi autokorelacionu funkciju. Autokorelaciona funkcija nekog stohastičkog signala računa se po sledećoj formuli:

$$R_x(k) = E\{x(n)x(n+k)\}$$

U nedostatku poznavanje funkcije gustine verovatnoće, definiše se i estimacija autokorelacione funkcije:

$$\hat{R}_x(k) = \lim_{N \rightarrow \infty} \frac{1}{2N+1} \sum_{n=-N}^N x(n)x(n+k)$$

S obzirom da u realnim aplikacijama nije moguće sprovesti limes, autokorelaciona funkcija se računa na sledeći način:

$$\hat{R}_x(k) = \frac{1}{2N+1} \sum_{n=-N}^N x(n)x(n+k)$$

Ukoliko je signal periodičan sa periodom P, tada je i atukorelaciona funkcija periodična sa istom periodom, i dodatno ima izraženi pik na vrednosti P. S obzirom da u govornom signalu postoje prostoperiodične komponente, u autokorelacionoj funkciji će biti izraženi pikovi, dok su kod bezvučnog signala ovi pikovi daleko manje izraženi. Često se pre sračunavanja autokorelacione funkcije koristi klipovanje, kako bi se jasnije videlo koji pik potiče od pitch periode, a koji od karakteristika vokalnog trakta. Jedna od najčešćih vrsta klipovanja je three-level klipovanje, prikazano na sledećem grafiku.

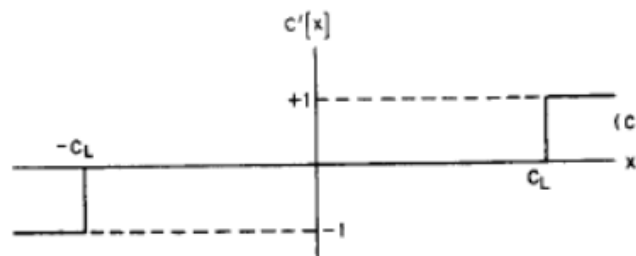


Figure 6: Three-level klipovanje

### 1.3 Rešenje

Najpre snimamo potrebnu sekvencu od 20 sekundi sa frekvencijom odabiranja od 8000Hz.

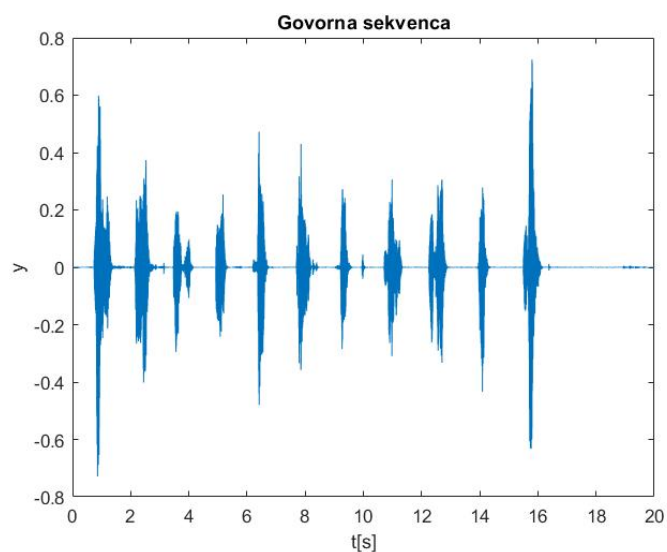


Figure 7: Snimljena govorna sekvenca

Zatim po gore navedenim formulama, i usvajanjem odgovarajuće širine prozora, iscrtavamo kratkovremensku energiju i kratkovremensku brzinu prolaska kroz nulu.

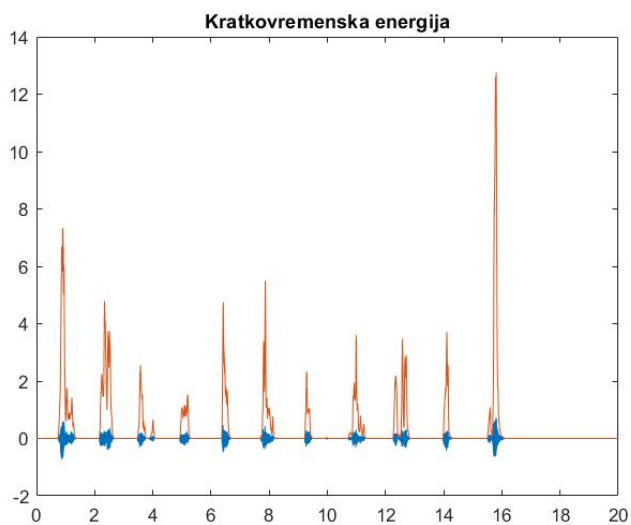


Figure 8: Kratkovremenska energija

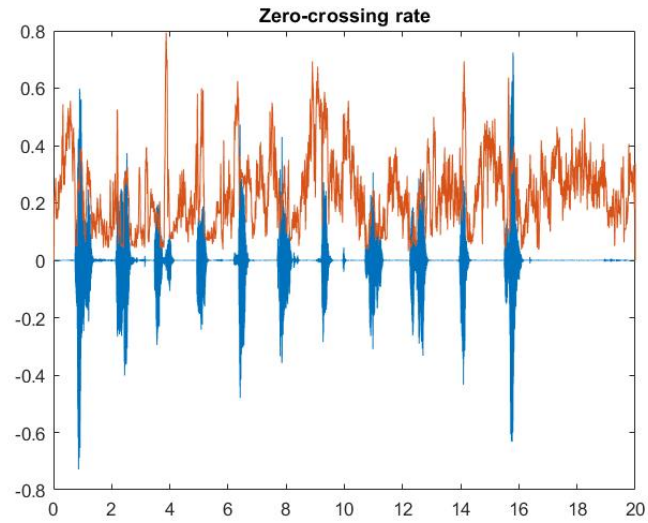


Figure 9: Kratkovremenska brzina prolaska kroz nulu

Primenom gore pomenutog algoritma za segmentaciju reči, dobijamo rezultate prikazane na sledećem grafiku.

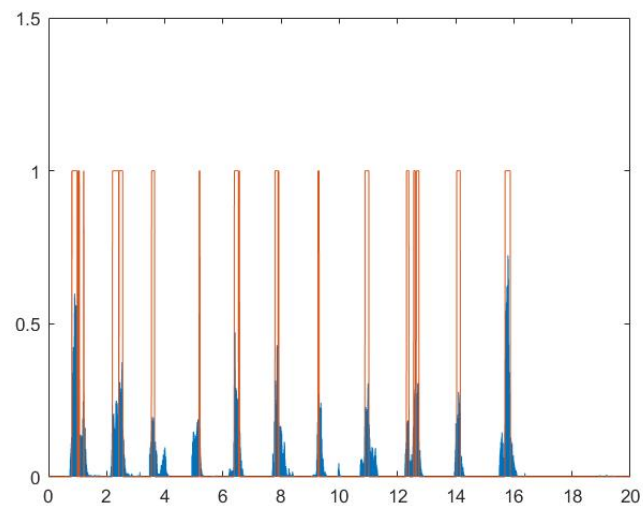


Figure 10: Inicijalna segmentacija reči u sekvenci

Primetimo da reči nisu baš najbolje segmentirane, i zbog toga odlučujemo primenimo fino podešavanje i dodatno proširimo reči, i time dobijamo zadovoljavajuće rezultate.

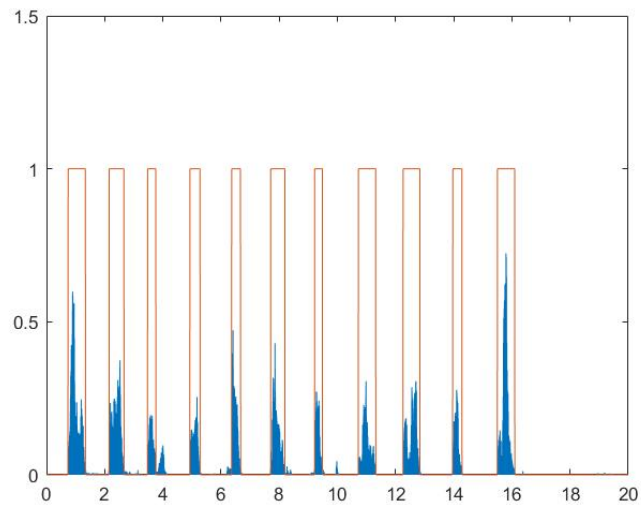


Figure 11: Finalna segmentacija reči u sekvenci

Prelazimo da određivanje pitch periode. Snimljena je sekvenca koja je bogata samoglasnicima, i koja je prikazana na sledećim graficima, prvo originalna, pa i filtrirana.

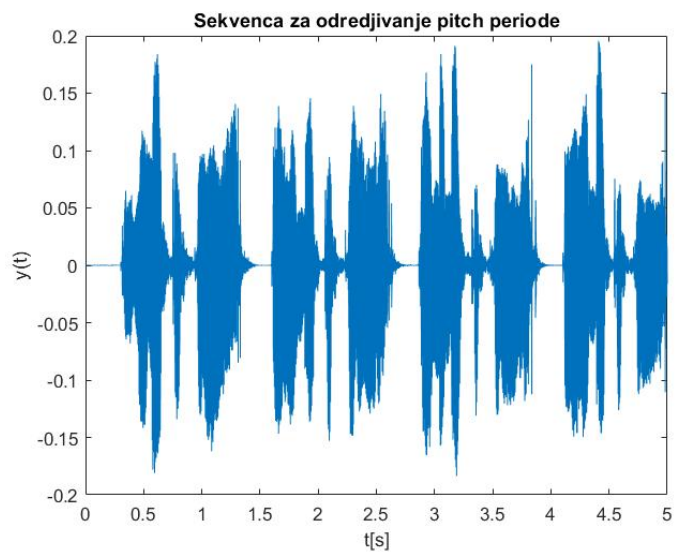


Figure 12: Govorna sekvenca



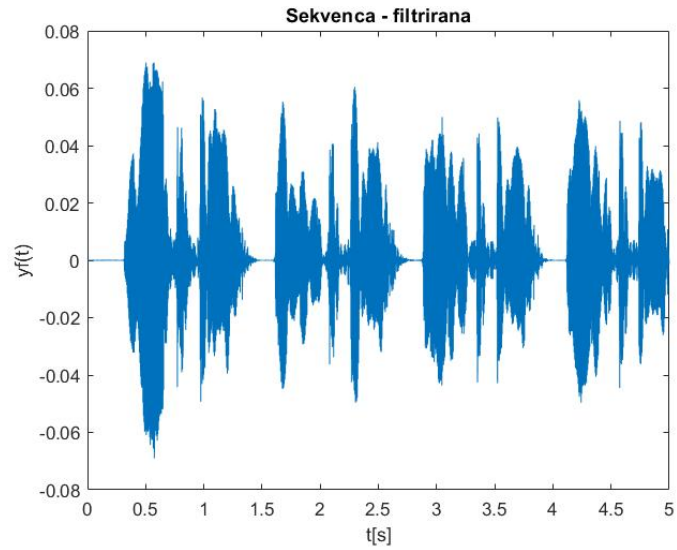


Figure 13: Filtrirana govorna sekvenca

Pitch perioda dobijena ugrađenom funkcijom za ovu sekvenču je 195.122Hz. Pokušaćemo prvo sa metodom paralelnog procesiranja.

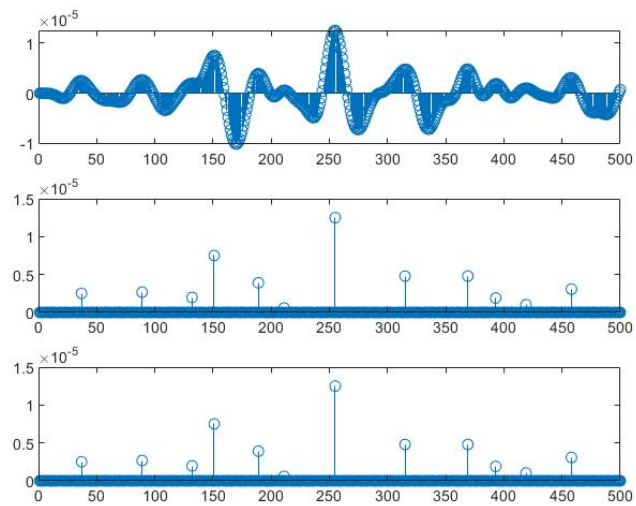


Figure 14: Funkcije m1, m2 i m3

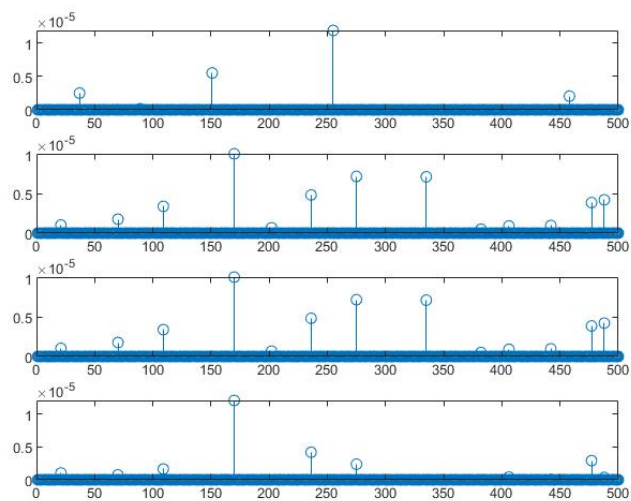


Figure 15: Funkcije m4, m5 i m6

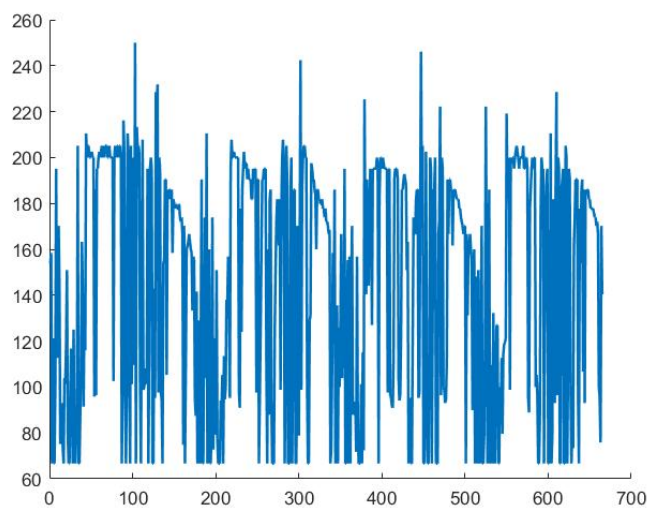


Figure 16: Metoda paralelnog procesiranja

Metoda paralelnog procesiranja za rezultat daje pitch periodu od 166.67Hz, dakle nije potpuno precizna. Sada primenjujemo i autokorelacionu metodu, i kao rezultat dobijamo vrednost od 200Hz, što je donekle bolje.

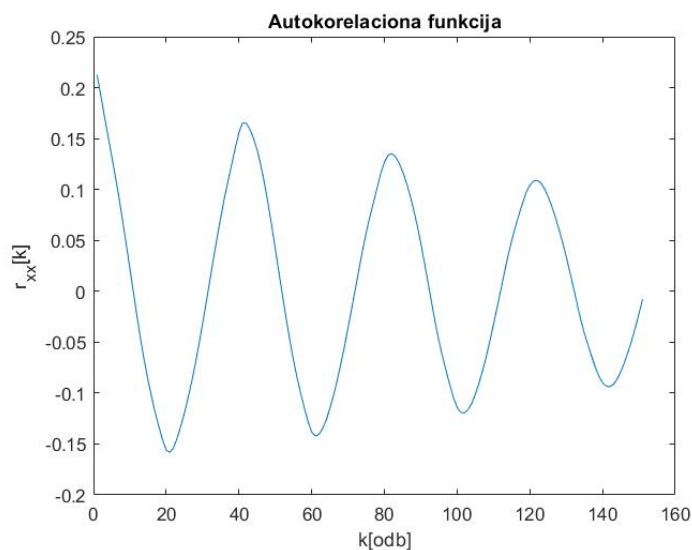


Figure 17: Autokorelaciona metoda

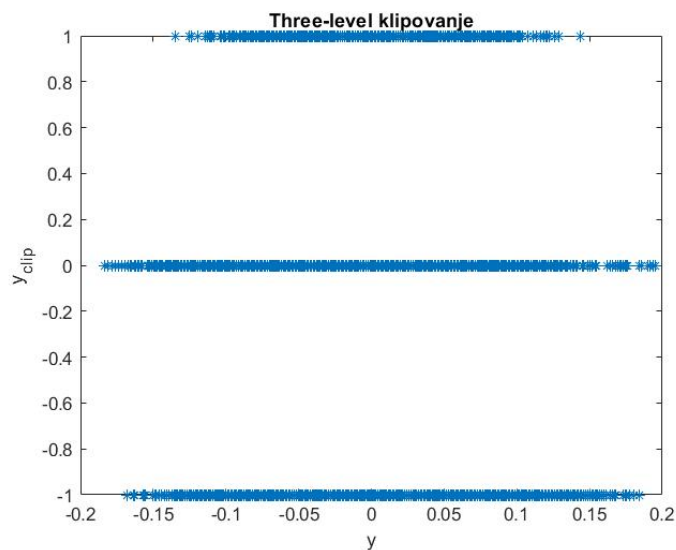


Figure 18: Three-level klipovanje

Isti postupak ćemo sada ponoviti za drugačiju sekvencu, koja sadrži samo izgovor jednog samoglasnika dužine 5 sekundi. Na sledećim graficima je prikazana data sekvenca, kao i njena filtrirana verzija.

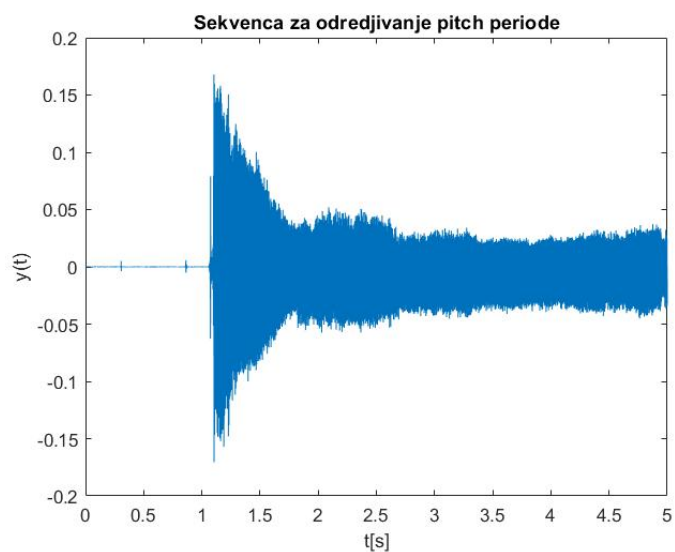


Figure 19: Govorna sekvenca

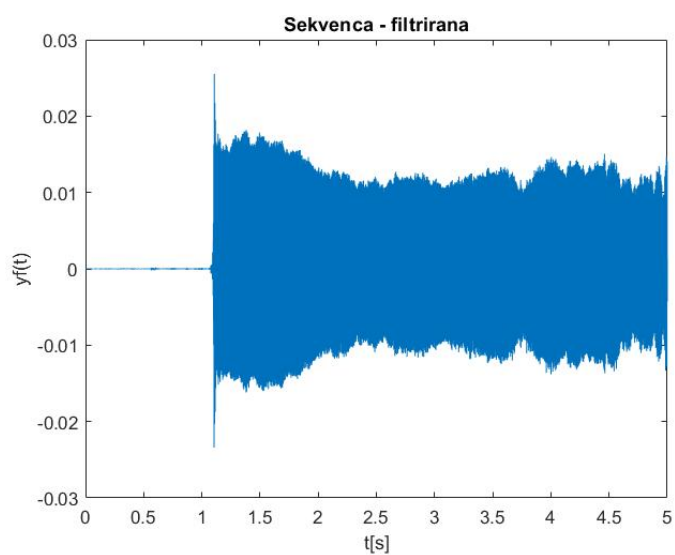


Figure 20: Filtrirana govorna sekvenca

Na filtriranu govornu sekvencu primenjujemo paralelnu metodu procesiranja, i dobijamo sledeće grafike odbiraka 6 potrebnih povorki impulsa:

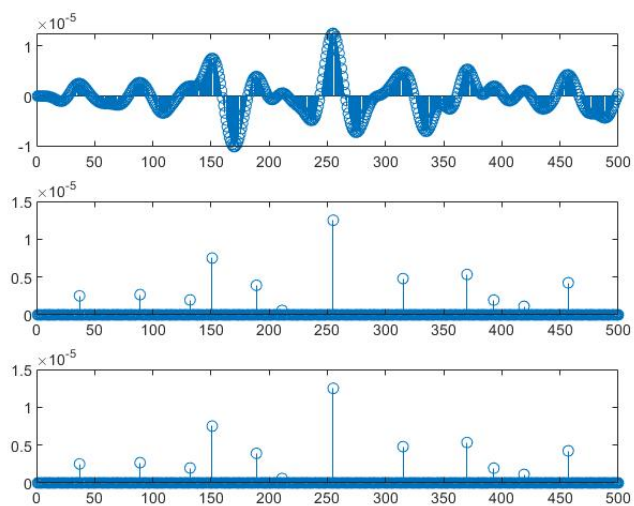


Figure 21: Funkcije m1, m2 i m3

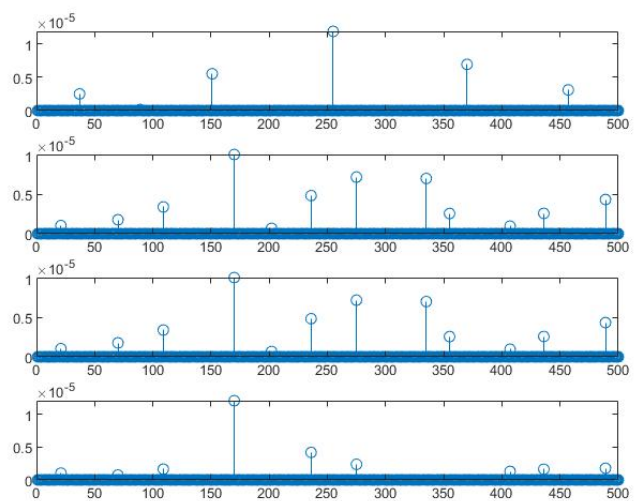


Figure 22: Funkcije m4, m5 i m6

Kao rezultat ove metode dobijamo da pitch perioda iznosi 200Hz, a istu vrednost dobijamo i uz pomoć ugrađene funkcije. Ova vrednost svakako ima smisla jer se nalazi u razumnom opsegu za žene, a znamo da je govornik ženska osoba. Rezultat je prikazan i na sledećem grafiku.

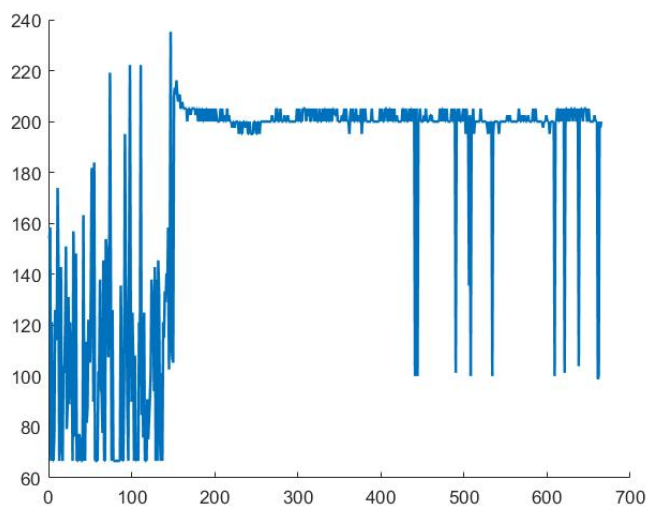


Figure 23: Metoda paralelnog procesiranja

Zatim primenjujemo i metodu koja koristi autokorelacionu metodu, i dobijamo ponovo isti rezultat od 200Hz, prikazan na sledećem grafiku. Možemo zaključiti da ove metode rade odlično, i da su prethodno ne tako sjajni dobijeni rezultati nastali zbog govorne sekvence koja nije bila idealna za analizu.

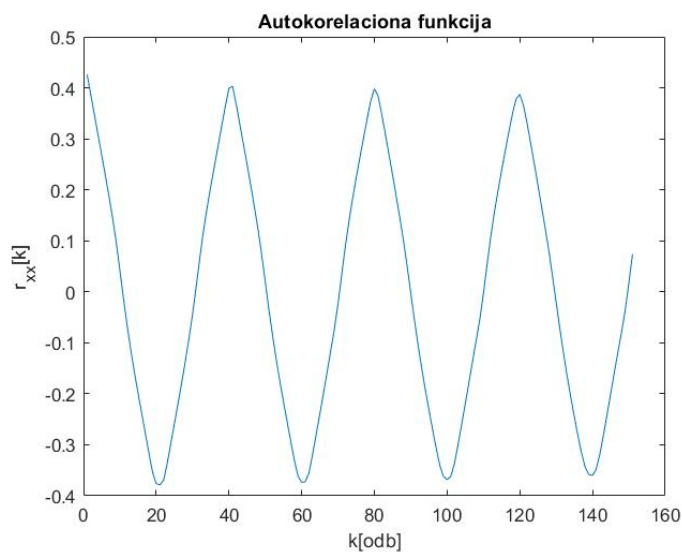


Figure 24: Autokorelaciona metoda

## 2 Drugi zadatak

### 2.1 Postavka

1. Korišćenjem komercijalnog mikrofona u programskom okruženju MATLAB, snimiti govornu sekvencu u dužini od 20-ak sekundi. Sekvencu snimiti sa frekvencijom odabiranja 8 kHz u šesnaestobitnoj (default) rezoluciji.
2. Isprojektovati  $\mu = 100$  i  $\mu = 500$  kompanding kvantizator sa 4, 8 i 12 bita i za njih odrediti zavisnost odnosa signal-šum za različite vrednosti odnosa ( $X_{max}/\sigma_x$ ). Ovaj odnos menjati promenom varijanse korisnog signala, prostim skaliranjem početne snimljene sekvence. Prikazati rezultate grafički.
3. Isprojektovati Delta kvantizator za sekvencu iz tačke 1. Adekvatno podesiti parametar  $\Delta$  tako da se dobije što bolji kvalitet kvantizacije. Uporediti oblike originalnog i kvantizovanog signala. Šta se dešava kada je korak kvantizacije  $\Delta$  previše mali ili previše veliki? Da li se histogram priraštaja može koristiti za određivanje adekvatnog parametra  $\Delta$ ? Pratiti kvalitet zvuka i promene u amplitudi za svaki slučaj.

### 2.2 Teorijski uvod

Kada je govorni signal tiši, najveći broj odbiraka se nalazi u blizini i s obzirom na ravnomernu, uniformnu kvantizaciju, kvantizovani signal će biti okarakterisan samo sa par kvantizacionih nivoa. Da bi se rešio ovaj problem, pojavila se ideja da se signal pre procesa kvantizacije provuče kroz logaritamsku funkciju, kako bi ekspandovao signal-vrednosti oko nule se razvuku na veći opseg. Ova ideja je sjajna i rešava ovaj problem, međutim logaritamska funkcija nije definisana u nuli, pa je njena direktna primena nemoguća. Kao rešenje ovog problema uvodi se nova funkcija  $F$  koja će modelirati ponašanje logaritamske funkcije, a opet biti definisana u nuli:

$$F(x[n]) = X_{max} \frac{\log[1 + \mu \cdot \frac{|x[n]|}{X_{max}}]}{\log[1 + \mu]} \cdot \text{sgn}(x[n])$$

Na sledećem grafiku možemo videti kako vrednost parametra  $\mu$  utiče na oblik funkcije.

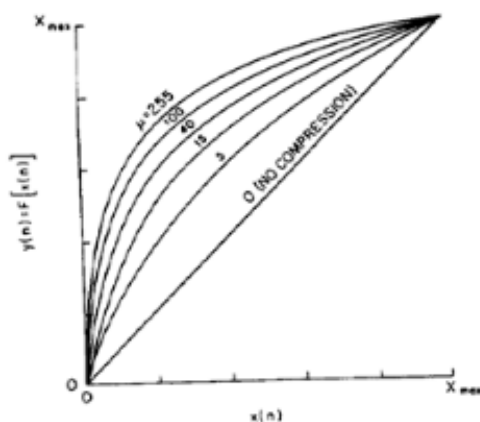


Figure 25: Zavisnost funkcije  $F$  od promene  $\mu$

Analizom možemo izvesti i izraz sa odnos signal/šum:

$$SNR[dB] = 6B + 4.77 - 20 \log_{10} \ln(1 + \mu) - 10 \log_{10} \left[ 1 + \left( \frac{X_{max}}{\mu \sigma_x} \right)^2 + \sqrt{2} \frac{X_{max}}{\mu \sigma_x} \right]$$

Oдавde možemo zaključiti da se prednost ovog kvantizatora ogleda u poslednjem izrazu u kome je opadanje odnosa signal/šum sa povećanjem odnosa  $\frac{X_{max}}{\sigma_x}$  umanjeno  $\mu$  puta. Efekat je onoliko izraženiji koliko je  $\mu$  veće. Dodatno, karakteristika SNR-a gotovo da miruje u opsegu  $\frac{X_{max}}{\sigma_x}$ , što je velika prednost u odnosu na uniformni kvantizator, kod koga karakteristika naglo opada.

Delta kvantizator predstavlja najjednostavniji oblik diferencijalnog kvantizatora. Zbog činjenice da je perioda odabiranja manja od Nikvistove periode, smatra se da su promene u signalu male, te je sasvim dovoljno čuvati informaciju o promeni znaka. Zbog ovoga je delta kvantizaciji dovoljan samo 1 bit, što predstavlja veliku uštedu. Ovaj bit ima vrednost 1 ukoliko je pomena pozitivna, a 0 ukoliko je negativna. Na sledećoj slici je prikazana struktura delta kvantizatora.

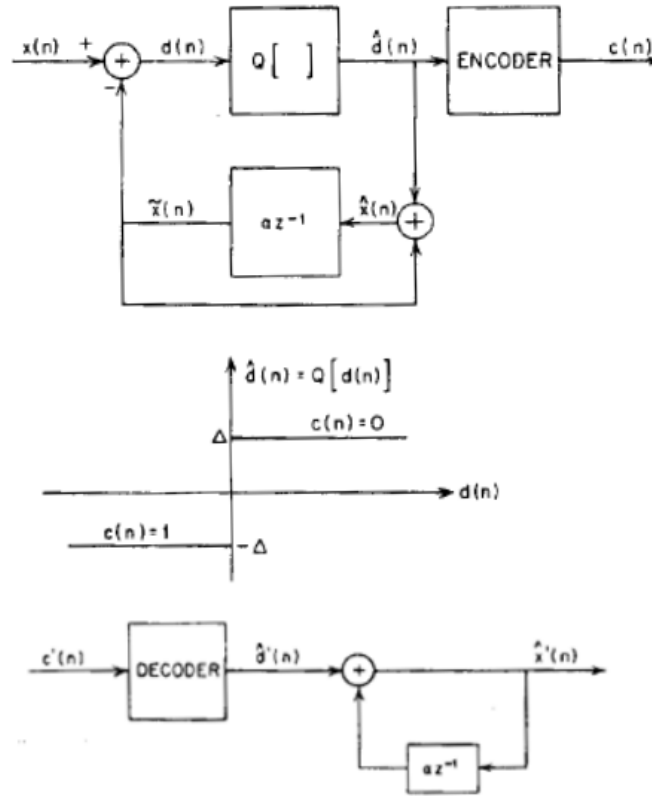


Figure 26: Šema delta kvantizatora

Kvantizovani signal priraštaja je definisan na sledeći način:

$$d[n] > 0 \Rightarrow \hat{d}[n] = \Delta, c[n] = 0$$

$$d[n] < 0 \Rightarrow \hat{d}[n] = -\Delta, c[n] = 1$$

dok je prediktor definisan kao:

$$\tilde{x}[n] = \alpha \hat{x}[n-1]$$

$$\hat{x}[n] = \alpha \hat{x}[n-1] + d[n]$$



## 2.3 Rešenje

Na početku snimamo sekvencu u trajanju od 20 sekundi, sa frekvencijom odabiranja 8000Hz u šesnaestobitnoj rezoluciji, koja je prikazana na sledećoj slici. Zatim ispitujemo rad  $\mu$  companding kvantizatora. Projektujemo

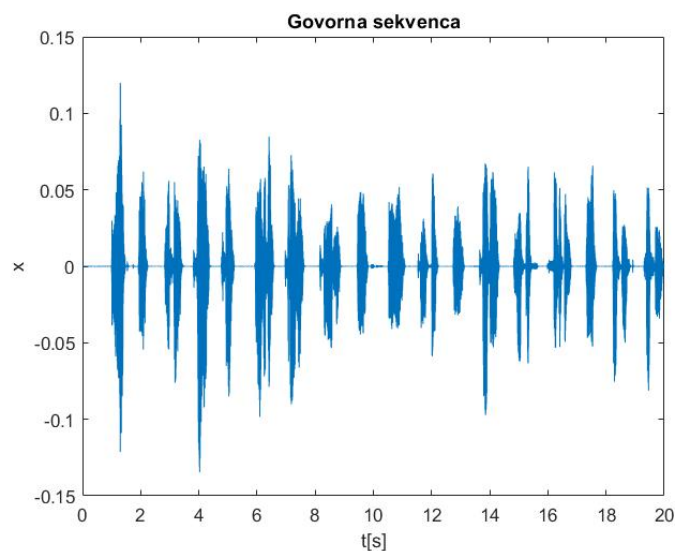


Figure 27: Govorna sekvenca

kvantizator za  $\mu = 100$  i  $\mu = 500$ , kao i za 4, 8 i 12 bita. Rezultati su prikazani grafički na sledećim slikama.

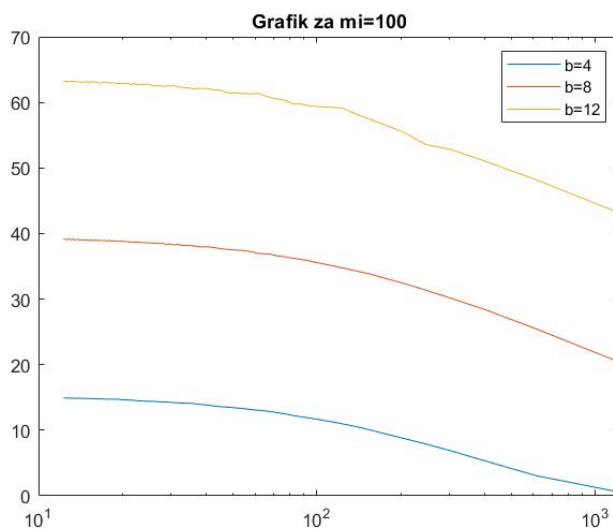


Figure 28:  $\mu = 100$

Možemo zaključiti da za manje  $\mu$  SNR ima veću vrednost, ali karakteristika ranije doživljava pad. Takođe, primećujemo da veći broj bita obezbeđuje i veću vrednost SNR-a.

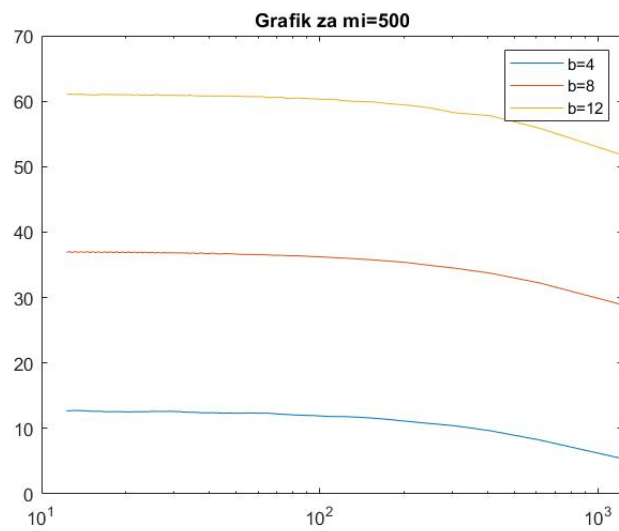


Figure 29:  $\mu = 500$

Sledeći zahtev je bio konstruisanje adekvatnog delta kvantizatora. Odabrali smo parametar delta kao  $\Delta = 0.01$  i dobili sledeći rezultat:

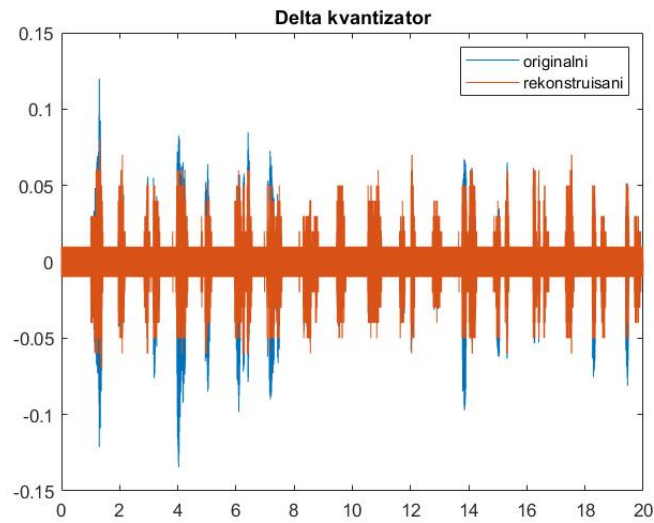


Figure 30: Originalni i kvantizovani signal

Rezultati deluju zadovoljavajuće. Bilo je moguće i odabrati korak kvantizacije pomoću histograma. Za uslov da je pokriveno 90% površine histograma, dobijamo da je  $\Delta = 0.0101$ . Da se desilo da smo uzeli isuviše

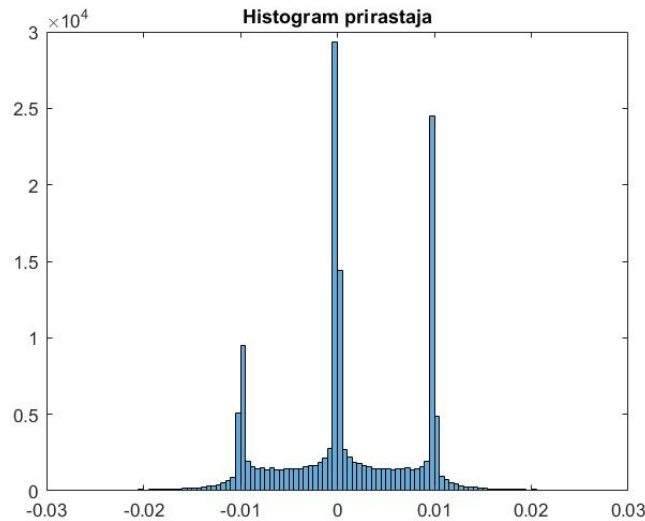


Figure 31: Histogram

malu vrednost za delta, došlo bi do slope overload-a - pojave kada rekonstruisani signal konstantno kasni za originalnim. Rešenje problema bi bilo usvajanje veće vrednosti za  $\Delta$ . Sa druge strane, ako je delta pak preveliku, dolazi do pojave zvane granular noise-rekonstruisani signal počinje da osciluje oko originalnog signala. Rešenje je logično-smanjiti  $\Delta$ .

## 3 Treći zadatak

### 3.1 Postavka

Snimiti bazu sa 3 izgovorene cifre, gde je svaka cifra izgovorena 10 puta od strane istog govornika (30 sekvenci u bazi).

1. Napisati funkciju *preprocessing* koja prima govornu sekvencu i vraća je nakon izvršene predobrade (segmentacija i filtriranje).
2. Implementirati funkciju *feature\_extraction* koja za prosleđenu sekvencu vraća obeležja zasnovana na LPC i/ili keprstralnim koeficijentima (dozvoljeno je korišćenje ugrađenih funkcija uz teorijski opis).
3. Konačna funkcija *cifer\_recognition* treba da pokrene kod za snimanje govorne sekvence, i zatim da obeležja snimljene sekvence dobijena na osnovu funkcija iz tačaka 1. i 2. prosledi klasifikatoru po izboru. Kada klasifikator donese odluku, ispisati je u komandnom prozoru.
4. Uspešnost klasifikacije testirati na po 5 novosnimljenih sekvenci iz svake klase i prikazati u obliku konfuzione matrice. Takođe prikazati konfuzionu matricu za trening skup. Za svaku od navedenih tačaka dati sažet pregled teorije na kojoj se zasniva, kao i detaljan opis implementacije same funkcije. Rezultate svake tačke prikazati grafički na odabranoj sekvenci i prokomentarisati uticaj izbora obeležja i klasifikatora na ishod klasifikacije. Izdvojiti i prokomentarisati primere tačno i pogrešno klasifikovanih sekvenci.

### 3.2 Teorijski uvod

LPC (Linear Predictive Coefficients) analiza polazi od pretpostavke da se govorni signal može opisati AR modelom, odnosno modelom koji poseduje samo polove. Zapravo, signal se može opisati kao linearnom kombinacijom prethodnih odbiraka. Rezultat LPC analize je  $p$  konjugovano-kompleksnih parova polova, a svaki par polova je određen sa 2 parametra, što znači da je potrebno  $2p$  LPC koeficijenata za određivanje svih polova. Pokazuje se da je u većini slučajeva dovoljno uzeti 14-16 LPC koeficijenata. Kako su prethodne metode izvedene pod pretpostavkom o stacionarnosti signala, a sam govorni signal to nije, LPC analiza se sprovodi nad prozorovanim sekvencama, u kojima se pretpostavlja da je signal vremenski invarijantan. Finalni rezultati LPC analize zavise od izbora prozora (čest izbor je Hamingov prozor), kao i dužine i stepena preklapanja prozora.

Sam postupak za prepoznavanje reči se može opisati kroz par koraka:

1. Prvo se vrši segmentacija govornog signala primenom neke od metoda
2. Zatim se usvaja tip, veličina i stepen preklapanja prozora tako da se obezbedi stacionarnost signala na prozoru
3. Za svaki prozor se određuju LPC koeficijenti nekom od metoda za AR spektralnu estimaciju
4. Dobijeni LPC koeficijenti se dalje koriste kao ulazi u sistem za prepoznavanje izgovorenih reči

### 3.3 Rešenje

Nakon što smo snimili bazu sa potrebni ciframa, prikazujemo jedan od signala kako bismo isprobali predobradu signala.

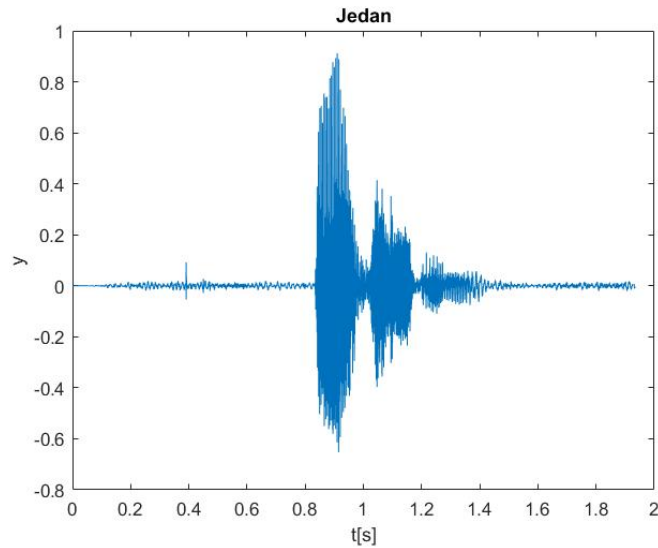


Figure 32: Jedan

Predobrada je odrađena korišćenjem Rabiner i Gold algoritma. Dodatno, zbog nekih delova signala koji su bili manje zvučni, odrađeno je i fino podečavanje i proćirivanje reći. Takode, signal se na kraju i filtrira low-pass filtrom. Rezultat je prikazan na sledećoj slici.

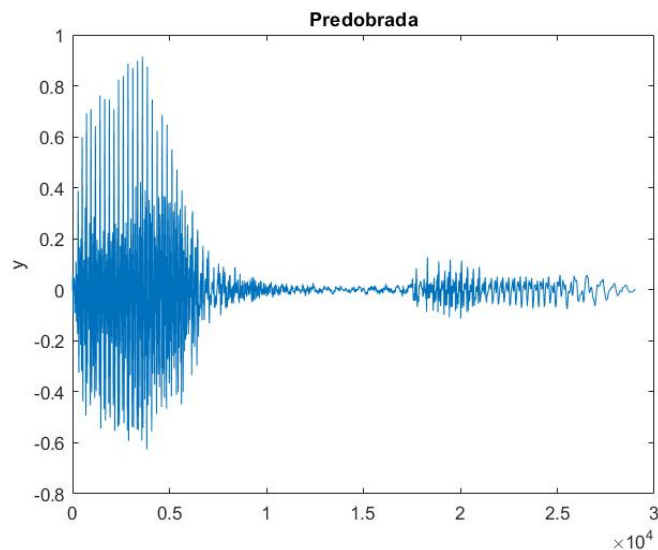


Figure 33: Signal posle predobrade

Zatim prelazimo na izraćunavanje LPC koeficijenata. Koristimo red modela  $p=16$  i raćunamo koeficijente uz pomoć ugrađene funkcije `aryule`, koja lako rešava Yule-Walker-ove jednaćine. Nakon izraćunavanja koeficijenata, biramo najreprezentativnije koeficijente, one koji će najuspešnije razdvojiti svaku od ovih klasa jednu od druge. Eksperimentalnim putem dobijamo da ove uslove ispunjavaju 9., 11. i 16. koeficijent, pa biramo njih. Na osnovu ovih koeficijenata isctavamo našu trening bazu cifara u 3D prostoru.

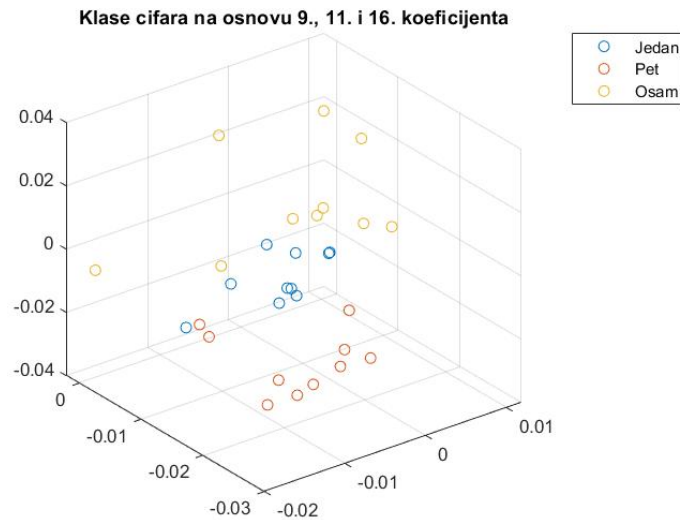


Figure 34: Klase

Sada kada imamo izdvojena obeležja, možemo da prediktuemo klase test skupa. Računamo koeficijente za svaki signal i uz pomoć KNN algoritma, sa parametrom  $k=10$ , dobijamo solidnu tačnost klasifikacije. 14 od 15 cifara je tačno klasifikovano, što je prikazano kroz konfuzionu matricu na sledećem grafiku.

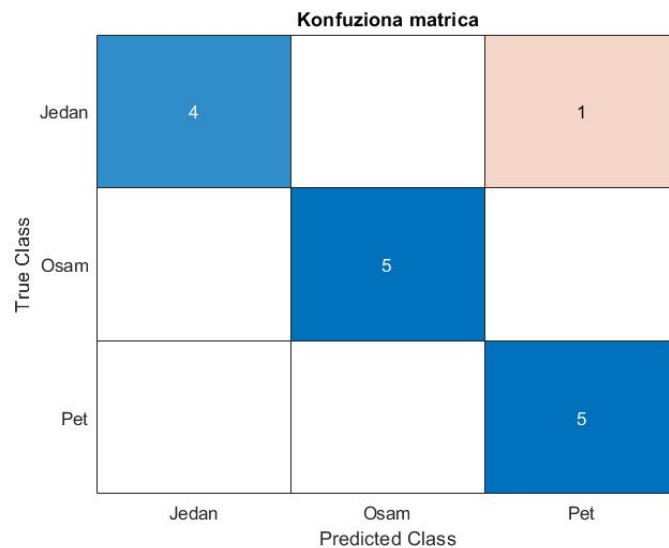


Figure 35: Konfuzionna matrica