LATVIJAS UNIVERSITĀTE

BAKALAURA DARBS

RĪGA 2010

LATVIJAS UNIVERSITĀTE
DATORIKAS FAKULTĀTE

RUNAS ATPAZĪŠANA

BAKALAURA DARBS

Autors: **Ansis Atteka**
Stud. apl. num. aa06033
Darba vadītājs: M. dat. Andris Atteka

RĪGA 2010

Anotācija

Ar runas atpazīšanas palīdzību ir iespējams panākt efektīvāku datu apmaiņu starp datoru un tā lietotāju, nekā ar ierastajām ievadierīcēm, piemēram, peli un klaviatūru. Kā dažus efektīvus runas atpazīšanas pielietošanas piemērus varētu minēt teksta diktēšanu, komandu padošanu datoram ar balss palīdzību un ļoti ātru runas fragmentu meklēšanu audio datnēs.

Potenciālie klienti runas atpazīšanas sistēmām varētu būt mājsaimniecības, uzziņu pakalpojuma sniedzēji, drošības dienesti u.t.t. Runas atpazīšana paver arī iespēju efektīvākai mašīnmācīšanai, jo šāda sistēma var tikt ieintegrēta sabiedrībā un mācīties autonomi.

Pateicoties arvien jaudīgākiem datoriem, ir iespējams realizēt precīzākas, lielu vārdnīcu, reāla laika runas atpazīšanas sistēmas.

Darbā ir aplūkotas problēmas, ar kurām ir jāsaskaras runas atpazīšanas sistēmu veidotājiem. Darbā ir piedāvāti veidi, kā saskatīt likumsakarības audio signālos.

Atslēgvārdi: Runas atpzīšana, skaņa, balss, īpašību vektori.

Abstract

With assistance of Speech to Text Recognition, it is possible to achieve much efficient data exchange rate between the user and computer. Some examples to consider are text dictation, voice commands and searching for text patterns in audio files.

Potential customers of speech recognition should be households, information service providers, intelligence agencies and so forth. Speech recognition also provides more effective way to design autonomous machine learning systems in other disciplines, because they can go unsupervised and try to interact with random real people.

As computers become more faster it is possible to implement speech recognition systems that achieves high accuracy, works real-time on large dictionaries.

In this thesis are considered issues that must be solved by a speech recognition system. There are also given tips how to look for patterns in audio signal.

Keywords: Speech recognition, sound, voice, feature vectors.

Saturs

Apzīmējumu saraksts

API – (no angļu valodas *Application Programming Interface)*

MFC – Mela Frekveņču Cepstrāli

HMM – Slēptie Markova modeļi (no angļu valodas *Hidden Markov Models*)

Hz - Hercs (frekvences mērvienība)

CMU – Universitāte Kalifornijā, kurā tika izstrādāta CMU Sphinx runas atpazīšanas sistēma(no angļu valodas *Carnegie Mellon University*)

Ievads

Runas atpazīšana ar skaitļotāju palīdzību jau kopš pagājušā gadsimta ir realitāte. Vislielākais ierobežojums runas atpazīšanas veiksmīgai nākšanai plašakā sabiedrībai ir tas, ka runas atpazīšana ir sarežģīta un tai ir nepieciešami ļoti jaudīgi datori.

Šajā darbā mēģināsim paši pārliecināties par to, *vai ir iespējams saskatīt runas pazīmes (fonēmas, vārdus) audio signālā.* Un, ja to varēsim izdarīt, tad drošivien arī eksistēs algoritms, kuru varam implementēt uz datora.

Darbs sastāv no divām nodaļām, pirmā nodaļa iepazīstina lasītāju ar audio signāliem un runas atpazīšanu vispārīgā līmenī. Tiek paskaidrots, kā skaņa izskatās datoram un tiek nodemonstrēti paņēmieni kā visvieglāk saskatīt likumsakarības audio signālos.

Otrā nodaļa iepazīstina lasītāju ar jau populārākajām esošajām runas atpazīšanas sistēmām un to, kādas iezīmes tām var piemist. Vislielākais akcents ir likts tieši uz atvērtā koda CMU Sphinx runas atpazīšanas sistēmu, jo tās kods ir publiski pieejams jebkuram. Jāpiebilst, ka tieši CMU Sphinx tika ņemts par iedvesmas avotu šim darbam.

Ievads runas atpazīšanā

Šajā nodaļā tiks iztirzātas fundamentālas problēmas, kuras ir jārisina katrai runas atpazīšanas sistēmai. Tiks paskaidorts "Kā izskatās skaņa analogajā un digitālajā pasaulē?", "Kas ir balss un runa?", īpašības vektoru izveidošana u.t.t.

Skaņa

Skaņa pēc būtības nav nekas cits, kā viļņi, kuri rodas spiediena svārstību iespaidā un pārvietojas kādā vielā, piemēram, gaisā vai ūdenī. Skaņa vakuumā nav iespējama, jo tur nav vielas, kuras būtu iespējams iesvārstīt.

Def: skaņa ir ceļojoši viļni, kuri rodas spiediena svārstību iespaidā un pārvietojas kādā vielā, bet frekvences, kuras veido šo skaņu, atrodas dzirdamo frekveņču diapozonā.[1]

Šādai definīcijai varētu arī uzreiz nepiekrist, jo nav skaidrs, kas ir dzirdamo frekveņču diapozons. Katrai dzīvajai būtnei dzirdamo frekveņču diapozons var būt atšķirīgs, piemēram, cilvēkam tas ir parasti no 12Hz līdz 20kHz, bet sunim 67Hz līdz 45kHz[2]. Pie tam ar vecumu pastāv varbūtība, ka katram cilvēkam šis diapozons mainās. Tātad, kas vienam pēc šīs definīcijas ir skaņa, otram var vairāk nebūt skaņa. Būtu vēlams vienoties, kur sākas un cik plaša tad beigās būs šī interesējošā skaņas frekveņču josla. Runas atpazīšanā par skaņas frekveņču diapozonu visērtāk būtu uzskatīt tās frekvences, kuras ir iespējams identificēt digitālajā signālā. Nākošajās nodaļās, kad būsim iepazinušies ar Nikvista teorēmu, tas arī tiks precizēts.

Skaņa laika domēnā

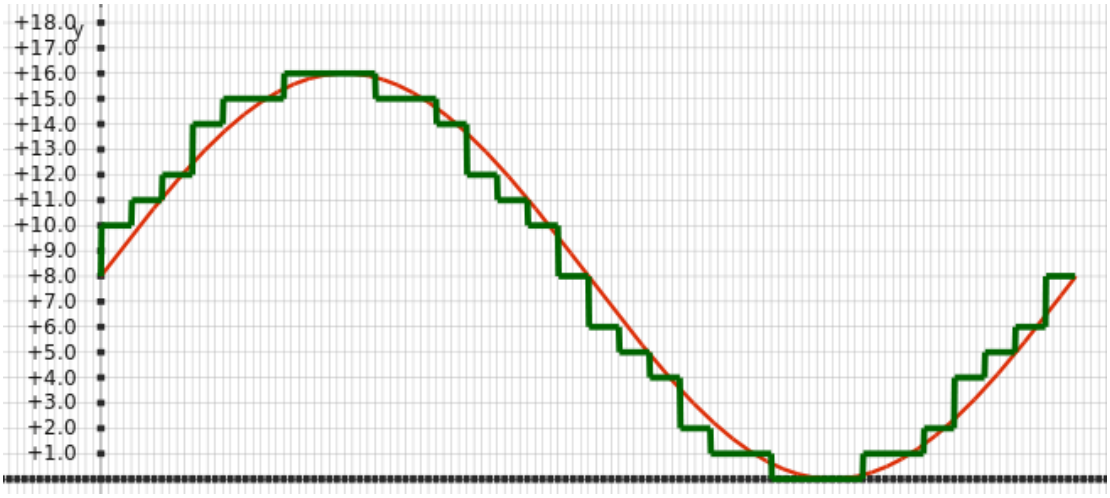
Skaņa, kura rodas runājot, ir vairāku šādu viļņu apvienojums. Šis viļņu kopums veido nepārtrauktu vienargumenta funkciju $f(t)$, kur šī funkcija mainās laikā. Ja vēlēsimies attēlot stereo signālu, tad jau mums būs nepieciešamas divas šādas vienargumenta funkcijas. Tieši nepārtrauktības dēļ, šo funkciju nav iespējams ideāli tādu pašu saglabāt datora atmiņā. Tāpēc ir nepieciešams veids, kā digitalizēt šādus analogos signālus, lai arī dators būtu spējīgs ar tiem strādāt. Katrai skaņas kartei, kas atrodas datorā, ir ADC *(Analog to Digital Converter)* iekārta. Šī iekārta veic PCM *(Pulse Code Modulation)* modulāciju jeb analogā uz digitālā signāla pārvēršanu. Ierīce, kas veic pārvēršanu pretējā virzienā, ir DAC *(Digital to Analog Converter)*. Tiesa gan, ka DAC pats izdomā, kā aizpildīt analogo izejas signālu starp diviem secīgiem

digitāliem mērījumiem, tātad nav iespējams pilnībā atjaunot oriģinālo signālu.

Def: PCM ir digitāla reprezentācija analogajam signālam, kur ik pa noteiktam laika brīdim tiek noteikta tā amplitūda no analogā signāla.[3]

Skatīt ***1.1.att***, kur ir dots piemērs, kā tiek modulēta nepārtraukta funkcija:

$$f(x)=8*\sin(\pi*\frac{x}{16})+8 \quad .$$



1.1.att. Analogā signāla pārvēršana digitālajā

Datorā šī sinusoīda pēc modulācijas tiek saglabāta ar 32 veseliem skaitļiem (X-ass), kur katrs no šiem 32 skaitļiem ir robežās no 0 līdz 16 (Y-ass). Uzreiz ir skaidrs, ka diezgan daudz informācijas tiek pazaudēta, jo, ja sinusoīda kādā konkrētā punktā nav vesels skaitlis, tad tā tiek noapaļota uz tuvāko veselo skaitli (atkarībā no ADC implementācijas), bet, ja, veicot divus secīgus mērījumus, sinusoīdai pa vidu būtu "pīķis", tad digitālajā formātā tas vairāk netiktu piefiksēts, jo mērījumi tika veikti pārāk reti, lai to pamanītu. Tātad, lai veiktu modulāciju, ir nepieciešams noskaidrot:

1. Frekvenci, cik bieži vienā sekundē ir nepieciešams veikt interesējošos mērījumus,
2. Dažādo vērtību skaitu, kuras noteiks amplitūdas precizitāti uz y-ass;

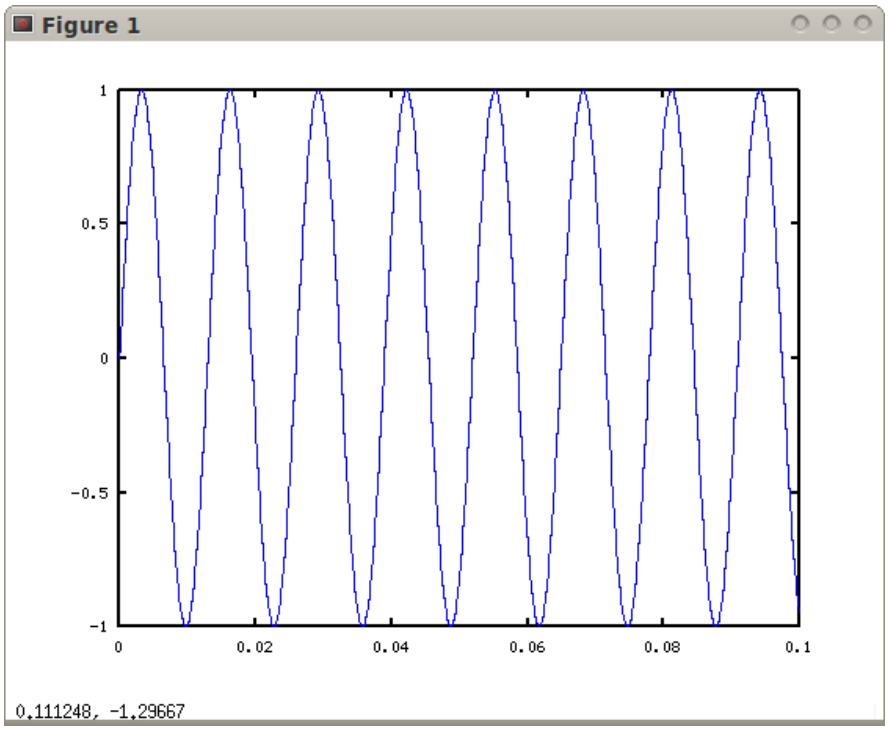
Papildus nestandarta raksturlielumi, kuri var piemist ADC, bet mūsu gadījumā būtu pārāk "eksotiski" ir:

1. kā pieļaujamās amplitūdas vērtības ir izkārtotas uz y-ass (lineāri, logaritmiski),
2. kāda ir minimālā/maksimālā amplitūdas vērtība, kuru var nomērīt uz y-ass,

3. vai mērījumi pa x-asi tiek veikti ar konstantu laika intervālu.

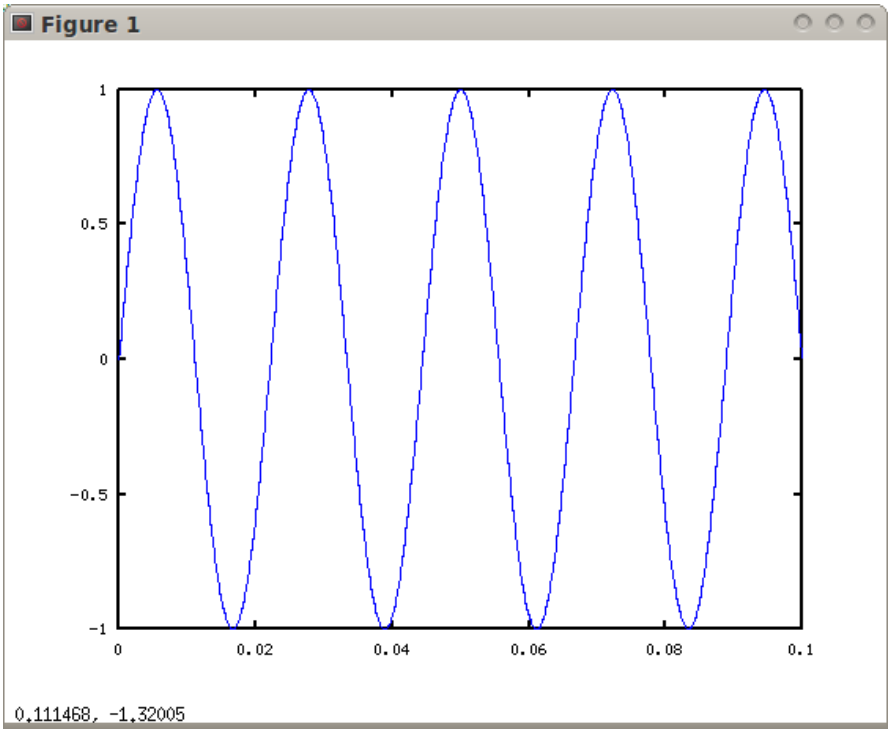
Skaņa frekveņču domēnā

Skaņu no laika domēna ir iespējams attēlot arī citā, vizuāli ērtākā veidā, proti, frekveņču domēnā, kas ir arī pazīstams kā spektrs. Lai nodemonstrētu sakarību starp laika domēnu un frekveņču domēnu, saskaitīsim divus "tīrus" viļņus (tādus, kuri nes tikai vienu konkrētu frekvenci) laika domēnā. Pēc tam to abu summu aplūkosim frekveņču domēnā un izdarīsim attiecīgus secinājumus.



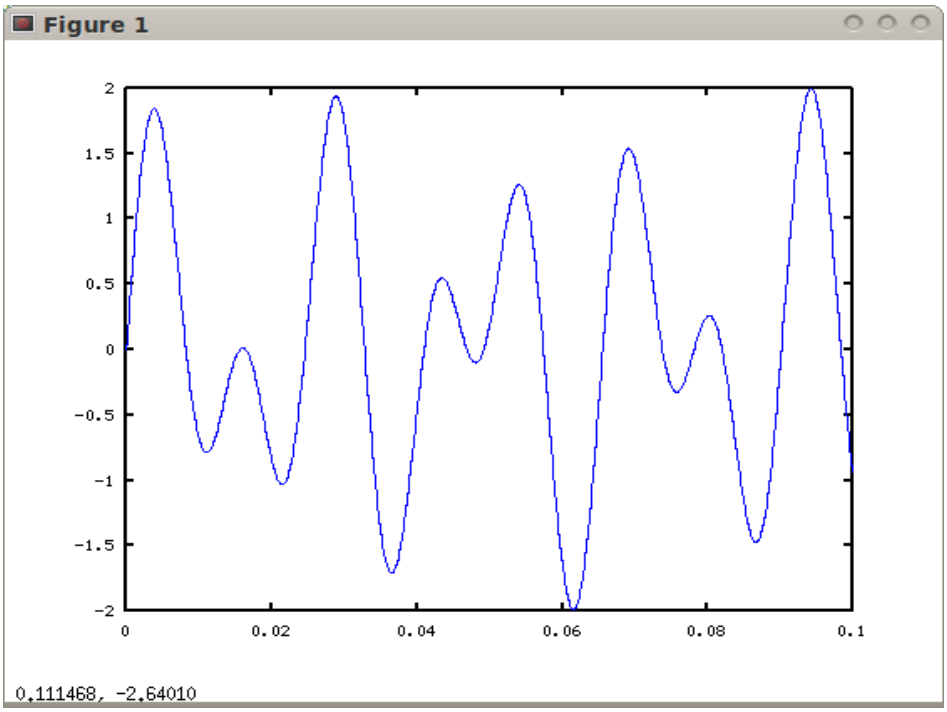
1.2. att. Tīrs 77Hz skaņas vilnis

Kā redzams **attēlā nr 1.2**. ir dota sinusoīda ar frekvenci 77Hz, kas veic 77 pilnus ciklus vienā sekundē (x-ass mērvienība ir sekunde). Šīs sinusoīdas amplitūda ir 1 vienība pa y-asi. Jo lielāka amplitūda, jo skaļāk ir dzirdams konkrētais skaņas vilnis.



1.3. att. Sinusoīda ar 45Hz frekvenci

Otra sinusoīda, kuru saskaitīsim, ir attēlota **1.3. att.** Tās frekvence ir 45Hz un tā veiktu 45 svārstības vienā sekundē. Tā pat kā iepriekšējā attēlā x-ass mērvienība ir sekunde. Saskaitot abus signālus, mēs iegūsim kaut ko šādu:



1.4. att. signāls, kurš iegūts saskaitot sinusoīdas no 1.2. att. un 1.3. att.

Būtu interesanti paskatīties, kā izskatīsies šis kombinētais signāls frekveņču domēnā. Bet pirms tam būtu jānoskaidro, kā veikt pārvēršanu starp šīm abām reprezentācijām. Talkā nāks Furjē transformācija, bet par cik mums interesē tieši digitālie signāli, tad speciāli pievērsīsimies Diskrētajai Furjē (DFT) transformācijai.

DFT dekompozīcīju jeb spektru var atrast trijos dažādos veidos – ar vienādojuma sistēmas, ar korelācijas vai arī ar Ātrās Furjē Transformācijas (*FFT*) palīdzību[4]. Klasiskā pieeja demonstrācijas nolūkiem ir izmantot korelāciju.

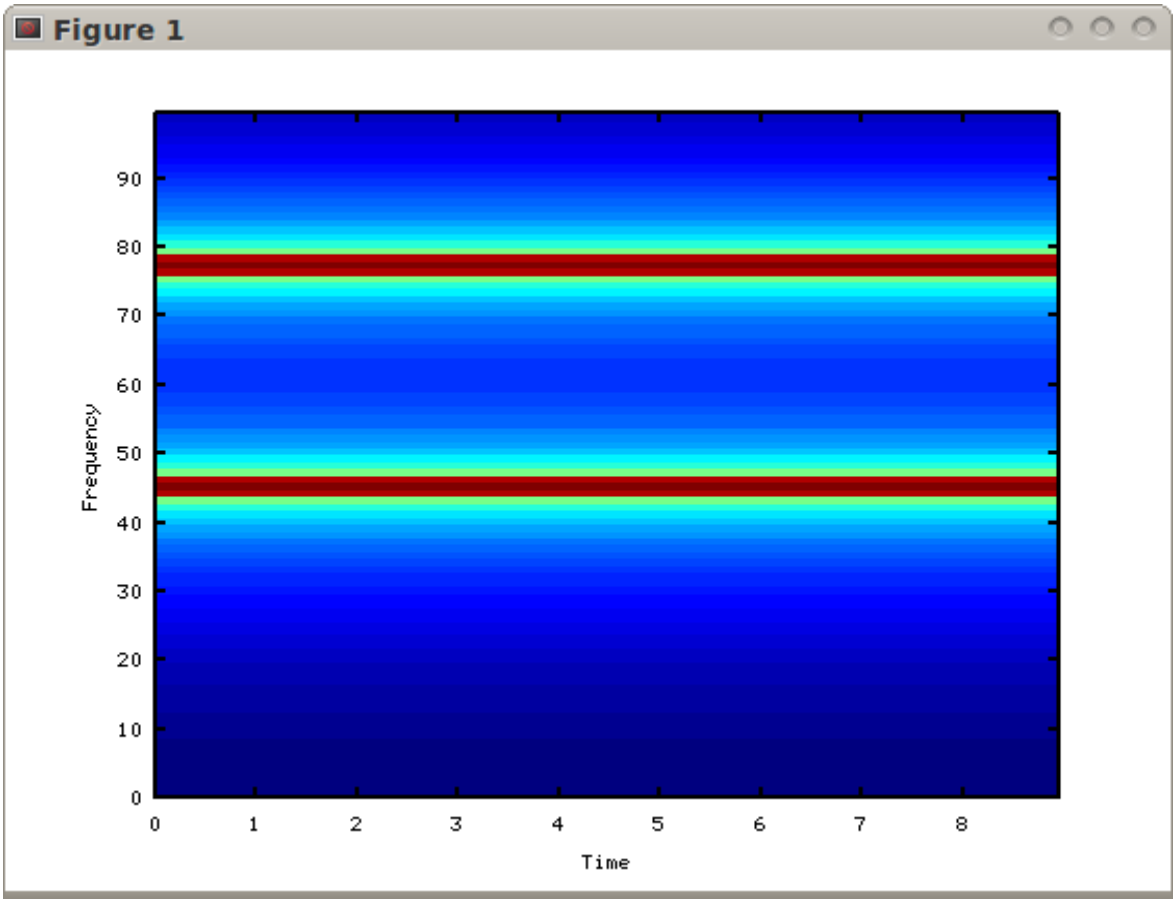
Korelācija ir darbība, kurā divi signāli tiek sareizināti un pēc tam visi iegūtie punkti šajā sareizinātajā signālā tiek sasummēti – tātad tiek iegūts viens vienīgs skaitlis. Pirmais signāls ir tas, kuram ir jāveic analīzi (jānosaka frekvenču komponenetes), bet otrais ir DFT primitīvfunkcija (ar kuru mēģināsim izteikt oriģinālo signālu). Pēc saskaitīšanas iegūtais skaitlis ir amplitūda katrai primitīvfunkcijai, ar kuru reizinājām oriģinālo signālu (protams, šo skaitli vēl ir jānormalizē, izdalot to ar konstanti).

*Def: Par divu diskrētu signālu X[1..N] un Y[1..N] reizinājumu sauc Z[1..N], kur Z[i] = X[i] * Y[i].*

Mūsu gadījumā primitīvfunkcijas ir $\sin(\frac{2*\pi*k*i}{N})$ un $\cos(\frac{2*\pi*k*i}{N})$, kur *k* ir vesels skaitlis robežās no *0* līdz $\frac{N}{2}$ (ieskaitot), bet pats *N* ir mērījumu skaits analizējamā signālā. Par cik *Octave* masīvu (vektoru) numurēšana sākas ar viens, indeksi *i* un *k* arī tiek samazināti par 1.

```
function [Re, Im] = DFT(X)

    N = length(X);

    for k = [1: (N / 2) + 1]
        Im(k) = 0;
        Re(k) = 0;

        for i = [1: N ]
            Re(k) = Re(k) + X(i) * cos(2 * pi * (k - 1) * (i - 1) / N );
            Im(k) = Im(k) - X(i) * sin(2 * pi * (k - 1) * (i - 1) / N );
        endfor

    endfor

endfunction
```

Ar augstāk aprakstīto pseido kodu ir iespējams no PCM signāla, kas atrodas masīvā *X[1..N]*, iegūt sinusa un kosinusa primitīvfunkciju amplitūdu koeficientus (attiecīgi *Im[1..(N/2) + 1]* un *Re[1..(N/2) + 1]*). Kopā ir *N/2 + 1* koeficienti katrai primitīvfunkcijai - gan sinusam, gan kosinusam. Pats pirmais un pēdējais sinusa koeficients vienmēr būs 0. Aplūkojam sasummēto signālu spektru un secinam, ka ir jūtams piesātinājums tieši pie 45Hz un 77Hz. Tie ir viļņi kas arī iepriekš tika saskaitīti.



1.5. att. Spektrs signālam no 1.4. attēla

Pretējā darbība DFT ir inversā Furjē transformācija. Ar tās palīdzību ir iespējams no frekveņču domēna pāriet atpakaļ uz laika domēnu.

```
function [X] = iDFT(Re, Im)

    N = (length(Re) - 1) * 2;

    for k = [1: (N /2) + 1]
        Im(k) = - Im(k) / (N / 2);
        Re(k) = Re(k) / (N / 2);
    endfor

    Re(1) = Re(1) / 2;
    Re((N / 2) + 1) = Re((N / 2) + 1) / 2;



    for i = [1: N]
        X(i) = 0;
        for k = [1: (N / 2) + 1]
            X(i) = X(i) + (Re(k) * cos(2 * pi * (k -1) * (i - 1) / N));
            X(i) = X(i) +(Im(k) * sin(2 * pi * (k -1) * (i - 1) / N));
        endfor
    endfor

endfunction
```

Algoritms ir pavisam vienkāršs – sareizinam katru primitīvfunkciju ar amplitūdas koeficientu un tad saskaitam visas tās kopā. Jāpiemin, ka iDFT ir arī jānormalizē amplitūdu koeficienti. Normalizēšana arī tiek izdarīts pirmajā iterāciju ciklā un nākošajās divās rindiņās tūlīt aiz cikla.

Nikvista teorēma

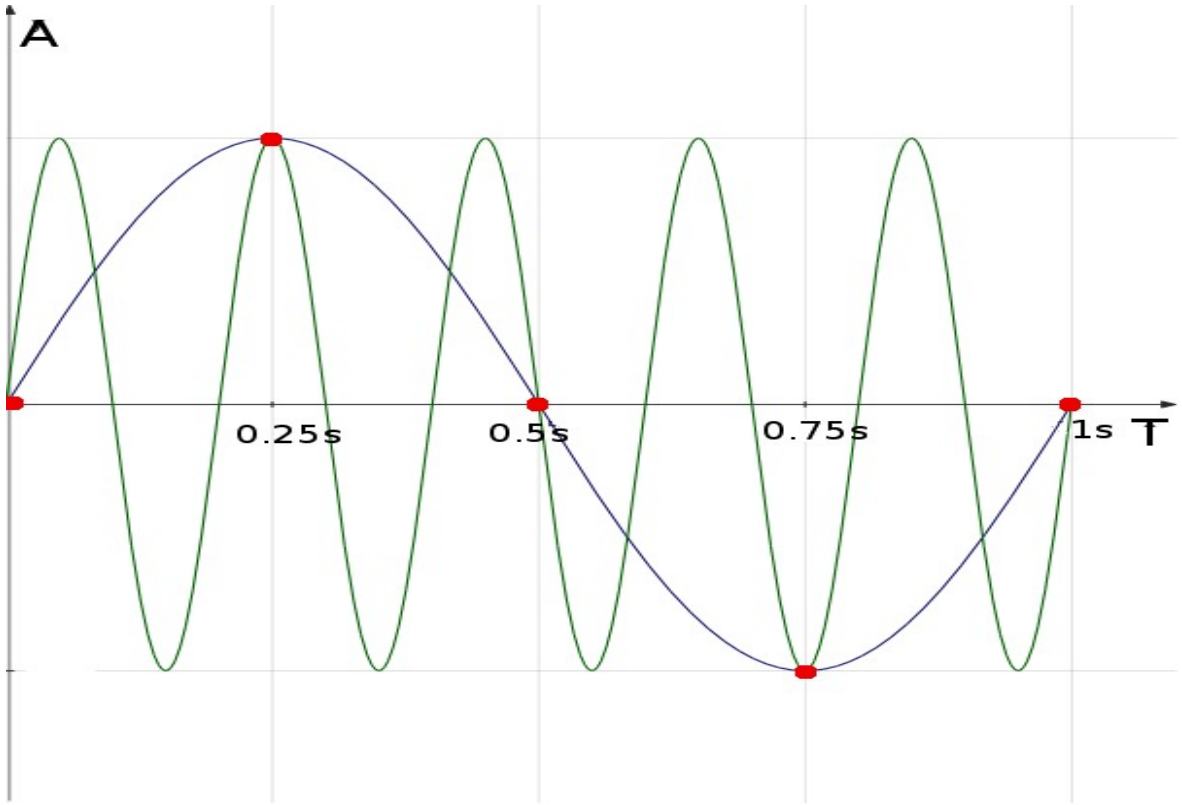
Iepriekš apskatījām, kā ar Diskrētās Furjē Transformācijas palīdzību var pāriet no laika-domēna uz frekveņču-domēnu, un arī otrādāk.

Teorēma: Nikvista teorēma nosaka, ka, ja funkcija f(t) nesatur frekvences augstākas par B Herciem, tad pietiek veikt mērījumus ar 2B Herciem, lai f(t) frekvenču informācija saglabātos arī pēc modulācijas.[5]

Darba sākumā pieminēju, ka cilvēks dzird skaņas no frekvenču joslas 20Hz līdz 20000Hz. Cilvēka balss frekvenču josla parasti atrodas zemāk sākot no 80Hz līdz pāris tūkstošiem Hercu[6]. Rodas jautājums, cik bieži ir nepieciešams veikt mērījumus, lai mūsu interesējošās frekvences saglabātos? Tas ir jāpieskaņo dzirdes vai runas aparātam? Patiesībā, tas ir atkarīgs arī

no apkārtējās vides. Apakšējai robežai vajadzētu būt divreiz lielākai kā augstākā frekvence no balss diapozona. Šādi tiek nodrošināts, ka visas frekvences, kas piedalīsies balss veidošanā savā starpā tiks identificētas kā unikālas. Bet ir arī jāņem vērā fakts, ka apkārtnē var būt trokšņi ar augstākām frekvencēm, piemēram 3300Hz, 5500Hz, kas visi tiks kļūdaini uzskatīti par viļņiem ar frekvenci 1100Hz pēc tam, kad būs veikta modulācija, piemēram, ar frekvenci 2200Hz.

Rodas jautājums – kāpēc tā? Piemēram, ja veicam 4 mērījumus sekundē, tad funkcijām $f(t)=\sin(2*\pi*t)$ un $g(t)=\sin(2*\pi*5*t)$ punktos *t = 0s; 0,25s; 0,5s; 0.75s*; u.t.t. vērtības vienmēr būs vienādas (skatīt ***1.6. att***), tāpēc nav iespējams izšķirt šos divus viļņus. Tas pats notiks arī ar augstākas frekvences trokšņiem, kuri kļūdaini papildinās balss frekvenču amplitūdas.



1.6. att. Divas dažādas sinusoīdas, kuras pēc modulācijas ar frekvenci 4Hz vairāk nevar atšķirt

Trokšņi

Trokšņi ir ļoti apgrūtinoša problēma, ar kuru jārēķinās katrai runas atpazīšanas sistēmai. Viens veids kā no tiem izvairīties, ir lietot augstākas kvalitātes mikrofonus, kuri ļauj atfiltrēt

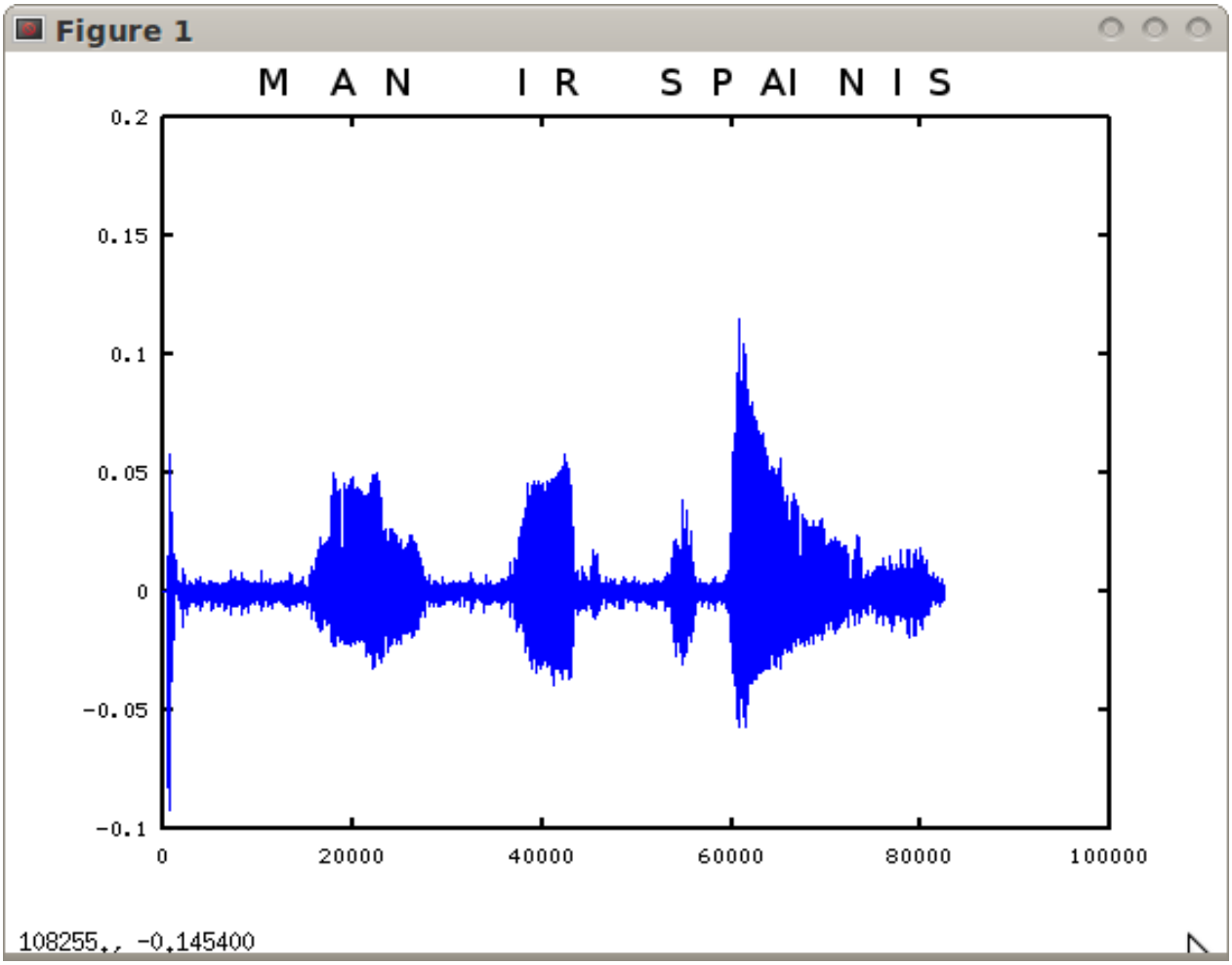
trokšņus jau analogajā pasaulē.

Jāpiemin, ka nav iespējams atbrīvoties pilnībā no visiem trokšņiem tik triviālā veidā, kā vienkārši samazinot interesējošo frekveņču joslu līdz runātāja balss frekveņču diapozonam. Pamatojums tam ir, ka trokšņus var radīt arī citi cilvēki, kas sarunājas blakus līdzīgās frekvencēs vai arī pat paša runātāja atbalss, kas rodas telpas dēļ.

Efektīvāks veids, ar kuru varētu izvairīties no trokšņiem, ir izmantot divus mikrofonus, kur viens atrodas tuvāk mutei, bet otrs tālāk. Tādejādi, ja no pirmā signāla atņems otro, rezultāta signālā dominēs pārsvarā runātāja balss.

Cilvēka balss un dzirde

Iepriekšējā apakšnodaļā apskatījām skaņu vispārīgā līmenī. Runas atpazīšana ir visvairāk ieinteresēta tieši balss frekvenču diapozonā, jo tieši šis diapozons nodrošina runas informācijas pārnesi no runas avota līdz dzirdes aparātam. Šajā nodaļā tiks paskaidrots, kā no audio signāla iegūt tieši to informāciju, kura būtu interesanta runas atpazīšanas sitēmai. Lai radītu labāku priekšstatu par cilvēka runu, apskatīsim divus piemērus no reālas dzīves. Skatīt ***att. 1.7.*** kurā ir attēlots audio signāls "Man ir Spainis" laika domēnā.

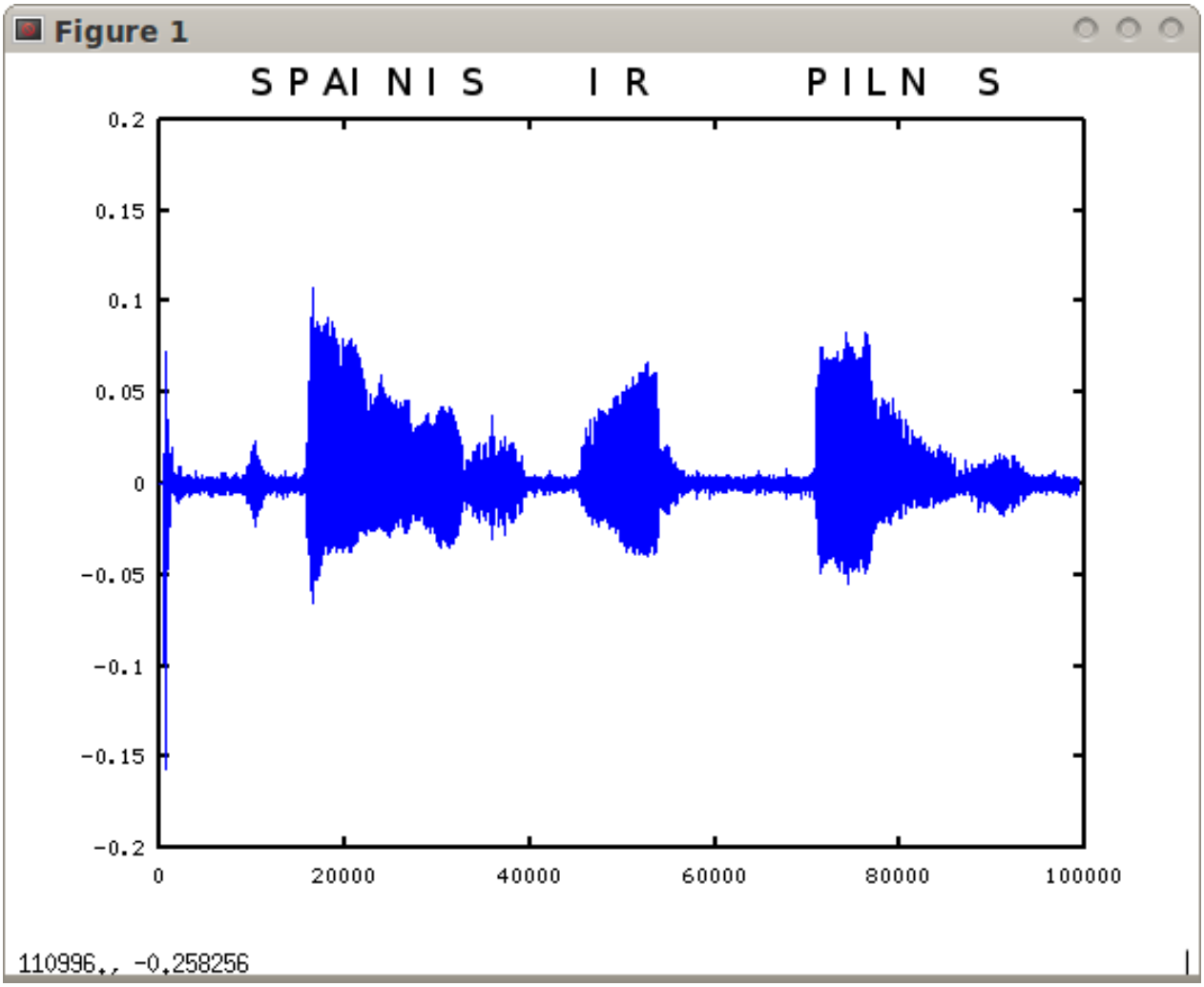


1.7. att. Audio signāla "Man ir spainis" attēlojums laika domēnā

Konkrētajā gadījumā vārdi ir ierunāti ar nelielām pauzēm (skatīt teikuma izrunu augšā par to, kurā brīdī kura skaņa ir dzirdama). Signāls tika mērīts ņemot paraugus ar 32KHz frekvenci. Uz x-ass ir attēlots konkrētais merījums, tātad, ja kopā ir 80000 mērījumi, bet signāls tikai mērīts ar biežumu 32000 paraugi/sekundē, tad sanāk, ka kopējais signāls ir aptuveni 2.5 sekundes ilgs. Ir

interesanti novērot, ka starp diviem secīgiem vārdiem, kad bija jābūt it kā klusumam, funkcija patiesībā nav vienāda ar 0 – tas ir fona troksnis.

Apskatīsimies vēl vienu audio signālu, kurā ir ierunāts teksts “Spainis ir pilns” (skatīt ***1.8.att.***).

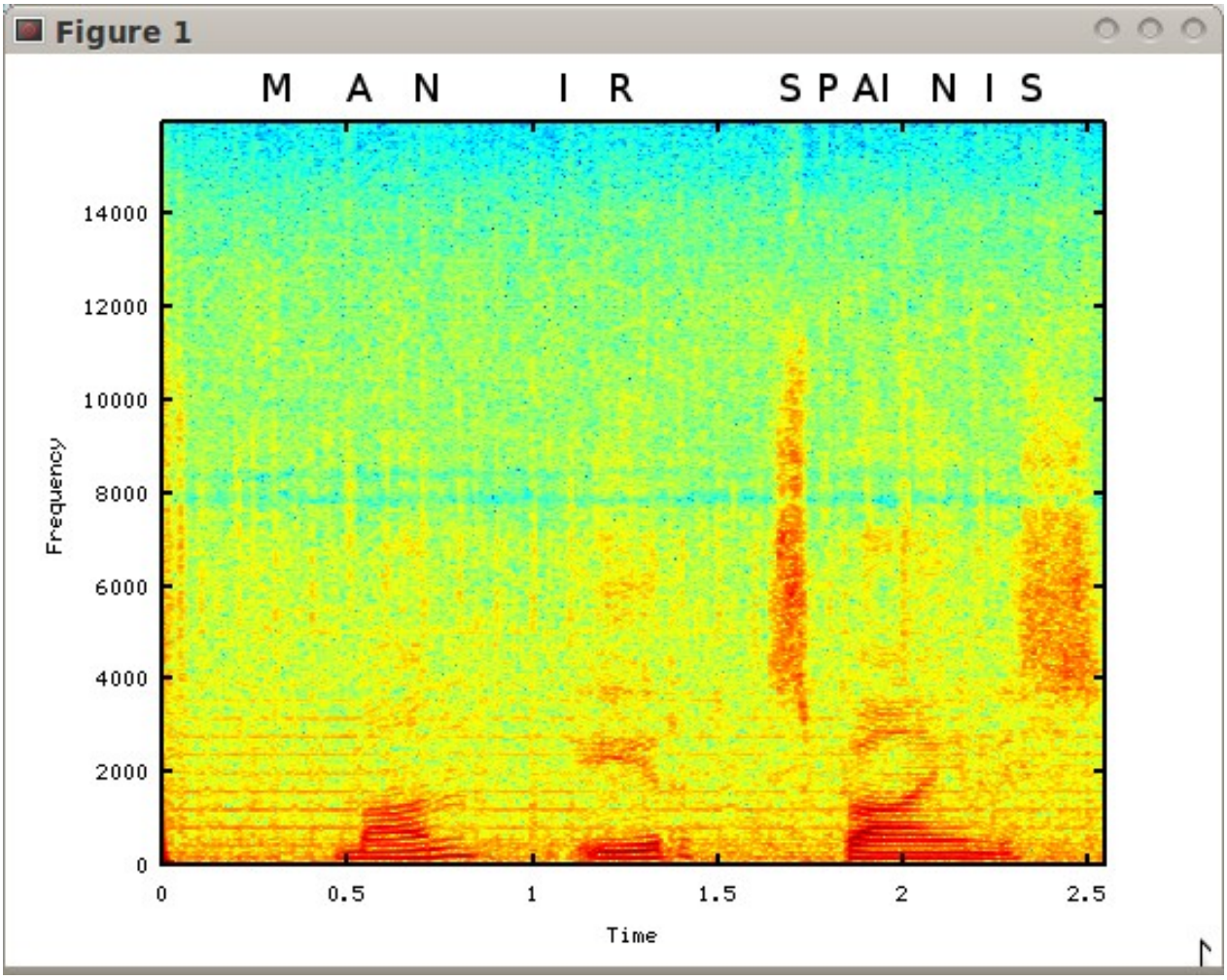


1.8. att. Audio signāla "Spainis ir pilns" attēlojums laika domēnā

Ja palūkojamies vērīgāk, tad signāls tur, kur ir ierunāts teksts “spainis” un “ir”, izskatās nedaudz līdzīgi abos attēlos, bet tas ir tikai tāpēc, ka abi vārdi tika ierunāti lēni un ļoti skaidri. Vislielākā problēma ar signālu pētīšanu laika domēnā ir tā, ka, ja viena un tā pati skaņa tika ierunāta dažādās frekvencēs, vai arī ja viens un tas pats vārds tika ierunāts dažādos laika intervālos, tad vairāk nav tik vienkārši saskatīt likumsakarības. Nākošajā nodaļā apskatīsimies, vai varam sameklēt likumsakarības labāk.

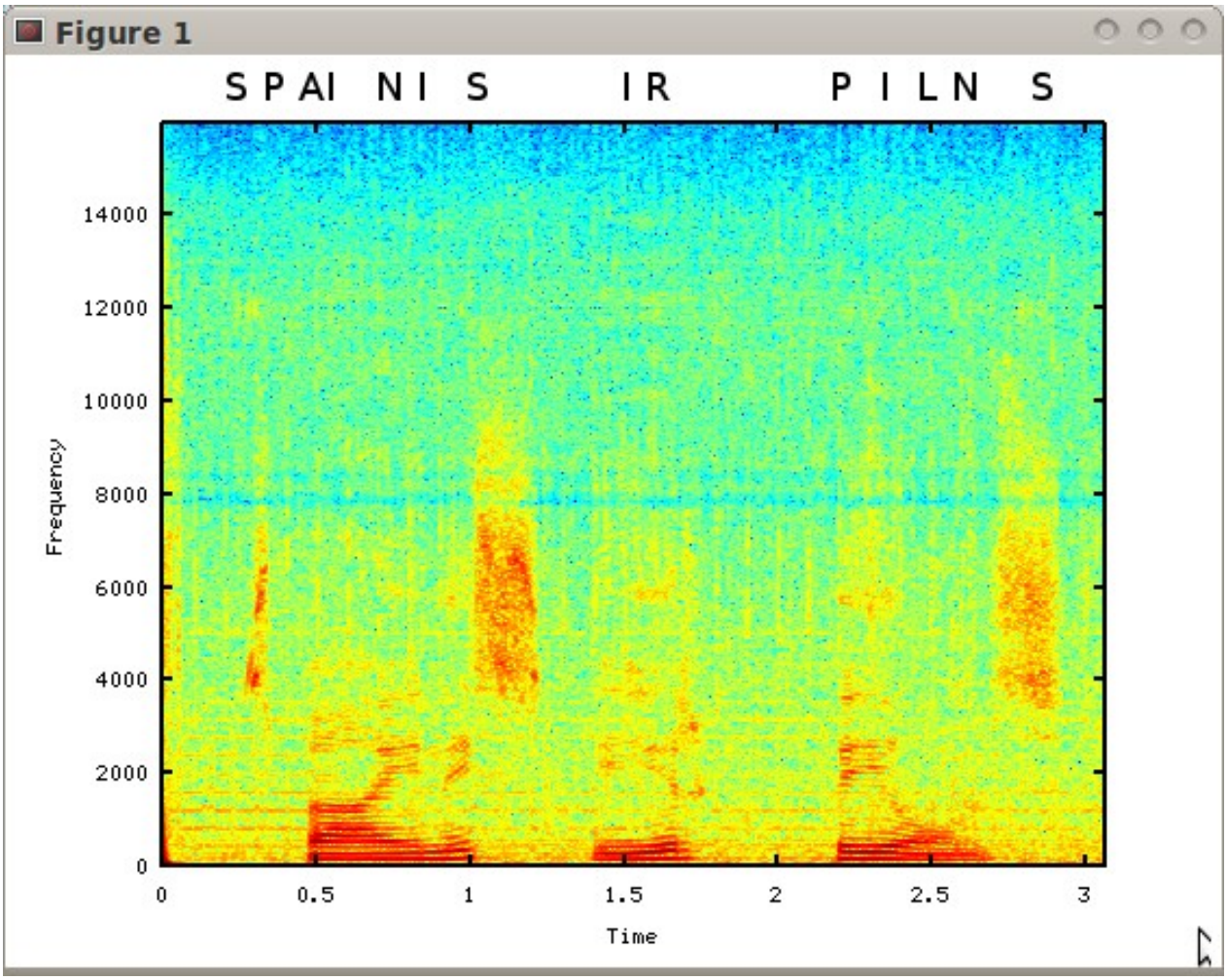
Frekvences

Nodaļā par skaņu mēs secinājām, ka jebkuru audio signālu var aplūkot arī frekveņču domēnā. Iespējams, ka tieši tajā reprezentācijā varētu saskatīt likumsakarības efektīvāk. Apskatīsimies runas "Man ir spainis" reprezentāciju frekveņču domēnā (skatīt ***1.9.att***).



1.9. att. Audio signālā "Man ir spainis" attēlojums frekveņču domēnā

Sarkanie punkti ir piesātinātie reģioni, kuros ir izteikti jūtama konkrētā frekvence (Y-ass). Varam pamanīt, ka runātāja balss diapozons ir frekvencēs līdz 4000 Hz. Izņēmums ir skaņa "s", kura piesātina frekvences intervālā no 4000Hz līdz 11000Hz (t = 1,6s un t = 2,4s). Aplūkosim arī otro signālu, kurā ir ierunāts "Spainis ir pilns" (skatīt ***1.10. att.***).



1.10. att. Audio signāla "Spainis ir pilns" attēlojums frekveņču domēnā

Secinājumi, kurus mēs varam izdarīt, ir tādi, ka

1. skaņa “P” vārdā “Pilns” un “Spainis” abos gadījumos neizskatās nemaz tik līdzīgi,
2. toties vārdi “Spainis” un “ir” abos attēlos izskatās diezgan līdzīgi;

Vēl rodas jautājums, vai cilvēks skaņu spēj izšķirt lineāri, logaritmiski vai arī pēc citas likumsakarības atkarībā no tās frekvences Hercos. Piemēram, atšķirt Do (262Hz) no Re (294Hz) var gandrīz jebkurš muzikāls cilvēks, bet vai viņam būtu pa spēkam atšķirt arī 15000Hz un 15032Hz skaņas signālus – diezin vai. Tieši tāpēc zinātniekie Stenlijs Stevens, Džons Folkmans un Edvīns Ņūmens] ieviesa jēdzienu, kā logaritmiskā Mela Frekveņču skala, kas ir iegūta statistiskā veidā, analizējot klausītāju spēju izšķirt skaņas dažādās frekvencēs[7].

Lietojot Mela frekveņču skalu ierasto hercu vietā, būtu tāda priekšrocība, ka runas atpazīšanas sistēma būtu spējīga koncentrēties uz skaņām tieši tā pat kā cilvēks – tai nebūtu

svarīgi izšķirt skaņas tik precīzi pie pie augstākām frekvencēm kā pie zemākām. Formula ar kuru no Herciem var pāriet uz *Meliem* ir šāda:

$$m = 1127 \ln\left(\frac{f}{700} + 1\right)$$

Inversā funkcija, lai pārietu no *Meliem* atpakaļ uz Herciem, ir šāda:

$$f = 700\left(e^{m/1127} - 1\right)$$

Īpašību vektori

Tik tālu mēs esam apskatījuši skaņas signālus laika un frekveņču domēnā. Tagad, kad esam iepazinušies arī ar Mela frekveņču skalu, būtu izdevīgi mēģināt aplūkot ko efektīvāku par tipisku spektru (jeb pilnu frekveņču dekompozīciju), lai varētu attēlotu objektus no akustiskās telpas.

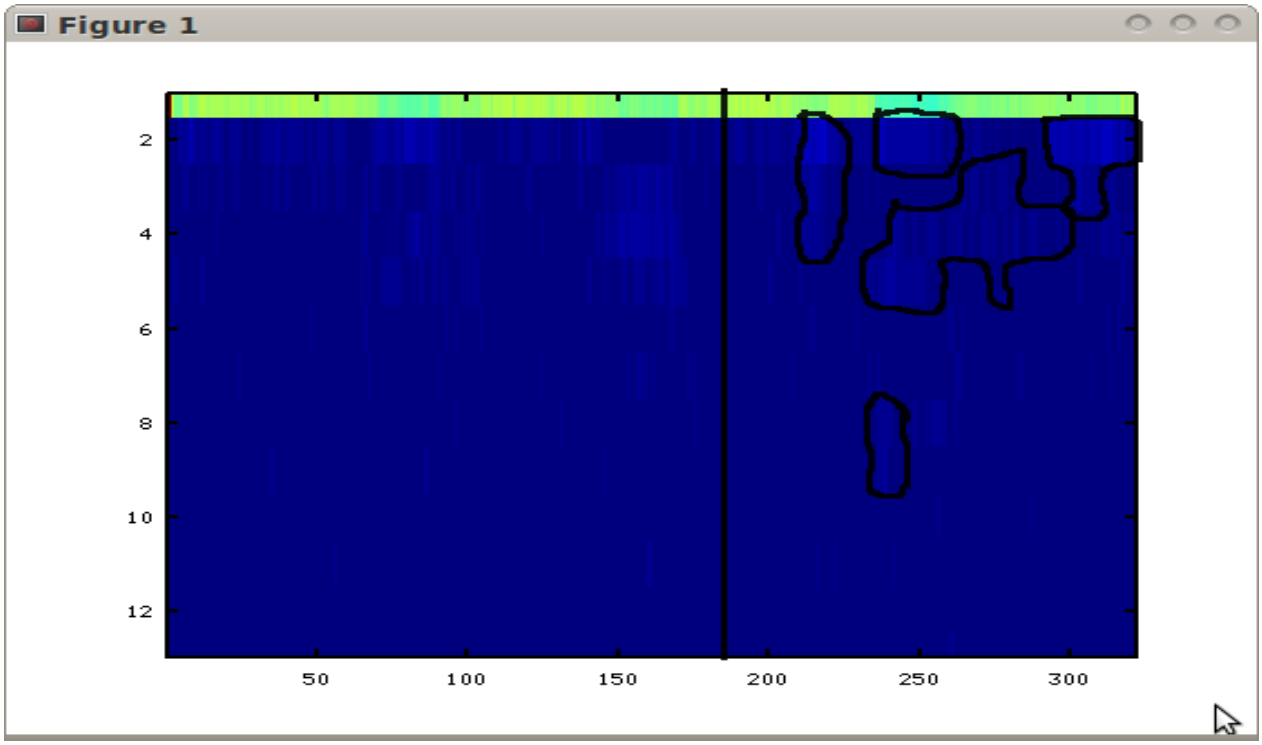
Def: Īpašību vektors ir n-dimensionāls vektors, ar kuru tiek attēlots kāds objekts.

Mūsu gadījumā katrs īpašību vektors dod iespēju kompakti attēlot "spektru" īsam (parasti ap 10 ms) laika momentam. Piemēram, ja kāds konkrēts vārds tiek izrunāts 2 sekundes, tad to vārdu var attēlot ar 200 šādiem īpašību vektoriem (pieņemot, ka īpašību vektori nemaz nepārklājas viens ar otru).

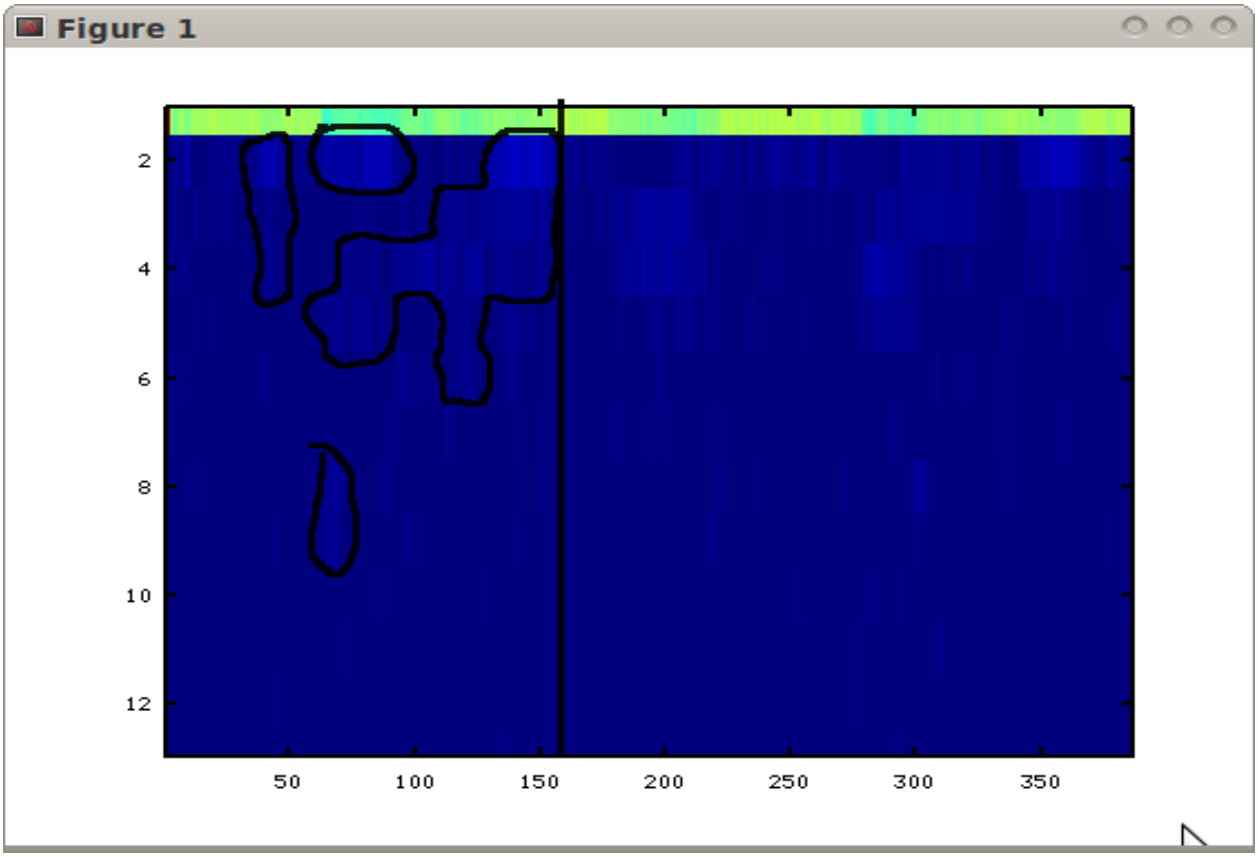
Kāpēc lai gan nevarētu izmantot pilno spektru kā īpašības vektoru? Patiesībā var izmantot (un var izmantot arī daudzus citus paņēmienus[8]), bet pilnais spektrs nav kompakts, jo, lai attēlotu objektu no akustiskās telpas 10 milisekundēm, ir nepieciešams milzīgs n-dimensionāls vektors (n>1000).

Tātad teorētiski runas atpazīšanā varētu pielietot jebkuru paņēmienu, lai iegūtu īpašības vektoru, kamēr netiek pazaudēta būtiskākā informācija, kuru tie satur. Viens veids ir izmantot Mela Frekveņču Cepstrālus (MFC). Algoritms MFC iegūšanai no PCM audio signāla ir ļoti labi aprakstīts Internetā[8]. Vislielākā priekšrocība Mela-frekveņču cepstrāliem ir tā, ka 10ms ar skaņu var attēlot tikai ar 13 skaitļiem.

Apskatīsimies, kā izskatās abi iepriekš aplūkotie signāli, ja tos izsaka MFC formā (skatīt ***1.11. att.*** un ***1.12. att.***). Tie MFC koeficienti, kuri ir izteiktāk jūtami, ir apvilkti ar melnu krāsu. Joprojām pastāv redzama līdzība starp šiem abiem vārdiem "Spainis" divos dažādos audio signālos.



1.11. att. Audio signāls "Man ir spainis" izsteikts MFC formā



1.12. att. Audio signāls "Spainis ir pilns" izteikts MFC formā

Runa

Par runu varētu uzskatīt visas tās skaņas, kuras tika izdvestas ar nolūku sazināties, izmantojot kādu valodu.

Fonēmas, zilbes un vārdi

Nodaļā par balsi secinājām, ka ir ļoti ērti skatīties uz audio signālu frekveņču domēnā. Lai arī datoram būtu ērti strādāt ar audio signālu, to nepieciešams attēlot kompaktā veidā ar īpašības vektoriem, kas patiesībā ir ļoti līdzīgs spektram.

Mazākā vienība, kuru cilveks ir spējīgs izrunāt, ir fonēma. Problēma ar tikai uz fonēmu orientētu atpazīšanu ir tāda, ka secīgās skaņas saplūst, piemēram, vārdā "rasa" otrais "a" tiek izrunāts savādāk nekā vārdā "naba". Lai atrisinātu šo problēmu un mūsu izrunas vārdnīcā būtu spējīga atšķirt visus šos dažādos "a", ir ieviests jēdziens kā konteksta fonēma. Aplūkosim abus šos vārdus un to izrunas vēlreiz:

1. [r a s a] - [$\_r_{a\ r}a_{s\ a}s_{a\ s}a\_$],
2. [n a b a] - [$\_n_{a\ n}a_{b\ a}b_{a\ b}a\_$];

Cena, kuru mēs maksājam par konteksta fonēmām, ir tā, ka to skaits dramatiski pieaug. Piemēram, ja latviešu valodā ir 26 līdzskaņi (b, c, č, d, dz, dž, f, g, ģ, h, j, k, ķ, l, ļ, m, n, ņ, p, r, s, š, t, v, z, ž), 12 patskaņi (a, ā, e (šaurais un platais), ē (šaurais un platais), i, ī, u, ū, o, ō) un 10 divskaņi (ai, au, ei, eu, ie, iu, oi, ou, ui, uo), tad sanāk, ka kopā ir iespējamas 26+12+10=48 dažādas fonēmas. Papildus mums ir jāpieskaita arī "klusums" pie fonēmām, jo, ja fonēma seko pēc izrunas pauzes, tad sanāk, ka tai nav kreisais konteksts. Tātad ieskaitot klusuma jeb pauzes fonēmu, kopā sanāk ap $(48+1)^3=117649$ dažādas trijfonēmas. Protams, ka trijfonēma $_{dž}dž_{dž}$ latviešu valodā nav sastopama, tāpēc mūsu sistēma varētu nebūt apmācīta atpazīt tādas netipiskas skaņas.

Runas atpazīšana var būt balstīta arī uz zilbēm, bet tad zilbe būs minimālā vienība, kuru tai būs iespējams atpazīt. Ja šādai sistēmai runātājs centīsies pateikt vārdu pa fonēmām, tad tas vairāk nebūs iespējams, jo fonēma ir mazāka vienība par zilbi. Tāpēc šāda sistēma tik un tā parasti tiek papildināta arī ar fonēmu atpazīšanu.

Ir iespējams realizēt arī runas atpazīšanu, kuras minimālā atpazīšanas vienība ir vārds.

Priekšrocības šādai sistēmai ir, ka tā var sasniegt ļoti augstu precizitāti, jo tā ir apmācīta uz konkrētiem vārdiem un zin, ko sagaidīt. Turklāt vislielākā problēma ir tā, ka pastāv gandrīz neierobežots skaits ar vārdiem, ar kuriem būtu jāapmāca šāda sistēma. Pie tam nepārtraukti tiek darināti jauni vārdi.

Runātāja īpatnības

Rūnātāja īpatnības ir visi tie faktori, kas var ietekmēt runas atpazīšanas sistēmas precizitāti. Mūsu smadzenes ir tik sarežģītas, ka tās bieži vien bez piepūles spēj tikt galā ar runas īpatnībām, lai saprastu teiktā jēgu, bet datoram sekojoši faktori ir ļoti nozīmīgi:

1. Dialekts – ja sistēma ir pielāgota konkrētam valodas dialektam, tad tā vairs tik labi nederēs citiem šīs valodas dialektiem,
2. Emocionālais stāvoklis – ja runātājs ir uztraucies vai apslimis, viņa valoda var datoram izklausīties nedaudz savādāk kā parasti,
3. Dzimums – parasti sieviešu balss ir augstākā frekveņču diapozonā nekā vīriešiem,
4. Runas defekti – dažiem cilvēkiem ir grūtāk izteikt kādas konkrētas skaņas;

Datoram ir jāparedz visi šie apsvērumi, pretējā gadījumā tiks pieļautas kļūdas.

Valodu īpatnības

Bez runātāja īpatnībām pastāv arī valodu īpatnības. Dažas valodas ir vairāk līdzīgas, bet citas mazāk. Piemēram, latviešu valoda ir specifiska ar locījumiem. Tas nozīmē, ka, lai sasniegtu pēc iespējas labākus atpazīšanas rezultātus, tai vai nu būtu jāzin par galotnēm un priedēkļiem, vai nu tai būtu nepieciešams uzskaitīt visus vārdus visos iespējamajos locījumos.

Citas valodas ir arī tonālas, piemēram, Mandarīnu dialekts, kuru runā Ķīnas ziemeļos. Šajā valodā ir svarīgs arī tonis kādā skaņa tiek izrunāta.

Runas atpazīšana sadalīta pa slāņiem

Uz runas atpazīšanu varētu arī skatīties kā uz procesu, kurš ir sadalīts pa slāņiem, kur katrs no tiem ir atbildīgs par kādu konkrētu jomu un cenšas uzlabot kopējo atpazīšanas precizitāti. Pie tam daži slāņi var būt ciešāk apvienoti viens ar otru, bet citi var vispār tikt izlaisti, piemēram, pārbaude konteksta atbilstībai.

Slānis	**Veicamais uzdevums**	**Piemērs**
Runas digitalizēšana.	Pārvērst analogo signālu digitālajā, lai nepazaudētu runas īpašības, bet tajā pašā laikā, lai atfiltrētu pēc iespējas vairāk trokšņus.	Nikvista teorēmas pielietošana un trokšņu filtri.
Īpašības vektoru izveide.	Jāizveido kompakti vektori, kas varētu pēc iespējas precīzāk pārstāvēt visus interesējošos objektus no akustiskās telpas.	Mela Frekveņču cepstrāli.
Īpašības vektoru sakopošana.	Īpašību vektori ir izkliedēti pa visu iespējamo akustisko telpu – līdzīgos vektorus ir nepieciešams sakopot.	Gausa Mikstūru modeļi.
Fonēmu/zilbju veidošana.	Pārbaude, vai virkne ar īpašības vektoriem varētu būt bijusi emitēta no konkrēta Slēptā markova modeļa, kam atbilst kāda fonēma vai zilbe.	Slēptie Markova modeļi.
Vārdu veidošana no fonēmām/zilbēm.	Pārbaude vai fonēmas veido pareizu vārdu	Piemēram, “drikubabs” nav pareizs vārds, bet fonēmas ir saliktas kopā diezgan veiksmīgi.
Teikumu veidošana no vārdiem.	Šis slānis ir atbildīgs par to, lai teikuma struktūra atbilstu gramatikas likumiem.	Piemēram, ir ļoti apšaubāmi, ka teikums varētu izskatīties šādi: “pilniem mežs ar zvēriem”. Latviešu valodai ir īpatnēja ar to, ka teikumā var mainīt vārdus vietām, bet tik un tā ir iespējams uztvert, kas tika sacīts. Savukārt vācu valodai ir stingrāki likumi teikumu uzbūvē, kas ierobežo runas atpazīšanā izvirzīto hipotēžu skaitu.

Teikumu atbilstība kontekstam.	Izvirzīto hipotēžu ierobežošana konkrētas vārdnīcas ietvaros, kas vislabāk atbilst sarunas tematam.	Ir maz ticams, ka sarunā vienlaicīgi parādās medicīnas, ekonomikas un būvzinātņu specifiski termini. Runa parasti ir orientēta uz konkrētu vārdnīcu, kuras vārdi atkārtojas biežāk. Daži cilvēki savā runā mēdz arī biežāk lietot konkrētus vārdus nekā citus (piemēram, “ķocis” un “grozs”).

Esošās programmatūras un to vēlamās raksturiezīmes

Šajā nodaļa tiks apskatītas jau esošās runas atpazīšanas sistēmas. Tiks apspriests tas, kā izvērtēt runas atpazīšanas sistēmas īpašības – kam jāpievērš uzmanība. Akcents lielākoties tiek likts tieši uz CMU Sphinx programmu saimi.

Šī tabula neiekļauj dažādos aplikāciju programmēšanas interfeisus (API), kuri izmanto vienu un to pašu runas atpazīšanas sistēmu.

Nosaukums	**Licence**	**Tiek uzturēta**
CMU Sphinx 3	BSD	Iespējams, ka drīz pārtrauks uzturēt CMU Sphinx 4 dēļ, kura kods ir rakstīts no jauna ievērojot kodēšanas standartus.
CMU Sphinx 4	BSD	Jā
PocketSphinx	BSD	Jā
Julius	BSD	Jā
Dragon Naturally Speaking	Privāta	Jā
Microsoft speech recognition engine	Privāta	Jā

Šajā tabulā ir uzskaitītas vairākas CMU Sphinx versijas, jo tās nav domātas, lai aizvietotu viena otru, bet gan katra no tām kalpo dažādiem mērķiem, piemēram, iegultām sistēmām, precizitātei u.t.t.

Sekojošiem aspektiem būtu jāpievērš uzmanība:

1. Vai runas atpazīšanas sistēma ir domāta lielām (~20000 vārdi) vai mazām vārdnīcām (~100 vārdi)?
2. Vai tiek atpazīti izolēti vārdi, runa ar pauzēm vai arī tekoša runa?
3. Vai sistēma ir pielāgota konkrētam runātājam?
4. Vai sistēma spēj atpazīt spontānu vai arī tikai diktētu runu?
5. Vai sistēma māk tikt galā ar trokšņiem?

Rezultāti

Darbā tika mēģināts noskaidrot, *vai ir iespējams saskatīt runas pazīmes (fonēmas, vārdus) audio signālā*. Lai nonāktu līdz šādam secinājumam bija nepieciešams:

1. *saprast, vai skaņu ir iespējams reprezentēt matemātiski?*
   Jā, to ir iespējams izdarīt. Tiek piedāvāti divi veidi, kā uzdot audio signālu matemātiski – frekveņču domēns un laika domēns;
2. *saprast, vai audio signālos parādās likumsakarības?*
   Jā, ir novērojamas likumsakarības. Aplūkojām divus audio signālus, kuros ir ierunāti teksti “Man ir spainis” un “Spainis ir pilns”. Nonācām pie secinājuma, ka ir zināmas līdzības abos gadījumos, kad tiek sacīts vārds “spainis”. Šīs sakarības bija viegli novērot tieši frekveņču domēnā;
3. *saprast, vai audio signālu ir iespējams uzdot ar ko kompāktāku kā pilnu spektru?*
   Jā, joprojām varējām saskatīt piesātinātos Mela Frekveņču Cepstrāla koeficientus, kas lielā mērā sakrita vārdam “spainis” abos signālos. Pie tam vektors sastāvēja tikai no 13 skaitļiem katrām 10 milisekundēm.

Secinājumi

Lai saprastu runas atpazīšanu, ir nepieciešamas milzīgas priekšzināšanas, kas balstās uz varbūtību teoriju, statistiku, digitālo signālu apstrādi, mašīnmācīšanos, valodniecību un pat uz psiholoģiju (lai runas atpazīšanas sistēma neizvestu no pacietības runātāju, kad tā nesaprot, ko tai liek darīt).

Darbā interesanta bija audio signālu izpēte tieši frekveņču domēnā, jo tur varēja pamanīt sakarības starp vienādām skaņām, zilbēm un vārdiem.

Lai veiktu dziļāku statistisku analīzi, būtu bijis ļoti noderīgi, izmantot kādu datubāzi, kurā jau glabātos augstākas kvalitātes audio faili un attiecīgais teksts, kas tika ierunāts šajā signālā. Jāpiemin, ka ir sastopamas šādas datubāzes jau angļu valodā[9], bet būtu bijis interesantāk darboties tieši dzimtajā, latviešu valodā.

Nākošais posms būtu, aplūkot statistisko modelēšanu, ar kuras palīdzību būtu iespējams meklēt likumsakarības secīgos mela frekveņču cepstrālos ar datora palīdzību (piemēram, izmantojot Slēptos Markova modeļus, MFC vektoru kvantizāciju akustiskajā telpā un Gausa mikstūru modeļus).

Izmantotās literatūras avoti

[1] Wikipedia – Sound – [tiešsaiste]. - [atsauce 20.05.2010].

Pieejams: http://en.wikipedia.org/wiki/Sound

[2] Louisiana State Univeristy – Frequency Hearing Ranges– [tiešsaiste]. - [atsauce 10.05.2010].

Pieejams: http://www.lsu.edu/deafness/HearingRange.html

[3] Wikipedia – Pulse Code-Modulation – [tiešsaiste]. - [atsauce 12.05.2010]

Pieejams: http://en.wikipedia.org/wiki/Pulse-code_modulation

[4] DSP guide – Discrete Fourier Transform – [tiešsaiste]. - [atsauce 15.05.2010]

Pieejams: http://www.dspguide.com/

[5] DSP guide – ADC and DAC – [tiešsaiste]. - [atsauce 23.05.2010]

Pieejams: http://www.dspguide.com/

[6] Wikipedia – Vocal Range – [tiešsaiste]. - [atsauce 21.05.2010]

Pieejams: http://en.wikipedia.org/wiki/Vocal_range

[7] Wikipedia - Mel Scale - [tiešsaiste]. - [atsauce 24.05.2010]

Pieejams: http://en.wikipedia.org/wiki/Mel_scale

[8] Center for Spoken Language Understanding – Lecture 5 - [tiešsaiste]. - [atsauce 21.05.2010]

Pieejams:http://www.cslu.ogi.edu/

[9] VoxForge - VoxForge - [tiešsaiste]. - [atsauce 31.05.2010]

Pieejams: http://www.voxforge.org/

Dokumentārā lapa

Bakalaura darbs

„Runas atpazīšana”

Ar savu parakstu apliecinu, ka pētījums veikts patstāvīgi, izmantoti tikai tajā norādītie informācijas avoti un iesniegtā darba elektroniskā kopija atbilst izdrukai. Piekrītu sava darba publicēšanai internetā.

Autors: Ansis Atteka _____
(Autora paraksts)

Ar savu parakstu apliecinu, ka esmu lasījis augšminēto bakalaura darbu un atzīstu to par **piemērotu** aizstāvēšanai Latvijas Universitātes datorzinātņu bakalaura studiju programmas gala pārbaudījuma komisijas sēdē.
Darba vadītājs: M. Dat. Andris Atteka _____
(Vadītāja paraksts)

Darbs iesniegts Datorikas fakultātē _____
(Iesniegšanas datums)

Ar šo es apliecinu, ka darba elektroniskā versija ir augšupielādēta LU informatīvajā sistēmā.
Metodiķe: Ārija Sproģe _____
(Metodiķes paraksts)

Recenzents: _____

Darbs aizstāvēts bakalaura darbu gala pārbaudījuma komisijas sēdē

_____ prot. Nr. _____, vērtējums _____
(Darba aizstāvēšanas datums)

Komisijas sekretārs: _____ _____
(Sekretāra paraksts)