KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
KOMPIUTERIŲ KATEDRA

Raimonda Makrickaitė

Balso signalo aptikimo ir triukšmo pašalinimo algoritmo tyrimas

Naudojant aukštesnės eilės statistiką

Magistro darbas

Darbo vadovas
prof. dr. E. Kazanavičius

Kaunas, 2006

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
KOMPIUTERIŲ KATEDRA

Raimonda Makrickaitė

Balso signalo aptikimo ir triukšmo pašalinimo algoritmo tyrimas

Naudojant aukštesnės eilės statistiką

Magistro darbas

Kalbos konsultantė
Lietuvių k. katedros lekt.
I. Mickienė
2006-05

Vadovas
prof. dr. E. Kazanavičius
2006-05

Recenzentas
doc. E. Toldinas
2006-05

Atliko
IFM-0/1 gr. stud.
Raimonda Makrickaitė
2006-05-15

Kaunas, 2006

Turinys

SUMMARY 4

ĮVADAS 5

2 ANALIZĖ 7

2.1 KALBOS ATPAŽINIMO TYRIMAI IR TAIKYMAI 7
2.1.1 Didelių balsų bazių rinkimas 10
2.2 BALSO SIGNALO APTIKIMAS 12
2.2.1 VAD algoritmas 13
2.2.2 VAD įvertinimas 14
2.3 TRIUKŠMAS 15
2.3.1 Pridėtinis triukšmas 17
2.4 AUKŠTESNĖS EILĖS STATISTIKOS (HOS) PASIRINKIMAS 18

3 ALGORITMO SUDARYMAS 19

3.1 AUKŠTESNĖS EILĖS SPEKTRAS IR STATISTIKOS 19
3.1.1 Apibrėžimai ir ypatybės 21
3.2 TRIUKŠMO KADRŲ ATPAŽINIMAS, NAUDOJANT AUKŠTESNĖS EILĖS STATISTIKĄ (HOS) 23
3.2.1 Balsui būtinos sąlygos 25
3.3 TIESINIS PROGNOZAVIMO KODAVIMAS (LPC) 25
3.3.1 „All Pole“ modelis 28
3.4 KALBOS ATPAŽINIMO ALGORITMAS, PAREMTAS AUKŠTESNĖS EILĖS STATISTIKA (HOS-VAD) 29

4 HOS-VAD ALGORITMO REALIZAVIMAS 34

4.1 HOS-VAD ALGORITMO KŪRIMAS MATLAB APLINKOJE 34
4.1.1 Įėjimo signalas 34
4.1.2 Skaidymas į kadrus 35
4.1.3 Skaidymas į langus 36
4.1.4 HOS parametrų skaičiavimas 36
4.1.5 SNR apskaičiavimas 36
4.1.6 Būsenų automatas 37
4.2 ALGORITMO MODELIO TESTAVIMO REZULTATAI 38
4.3 ALGORITMO REALIZAVIMAS DSK PLOKŠTĖJE 44
4.3.1 Algoritmo programavimas C programavimo kalba 45
4.3.3 C kodo įdiegimas DSK plokštėje 45

IŠVADOS 47

LITERATŪROS SĄRAŠAS 49

Voice Activity Detection and Noise Reduction Algortihm Analysis using Higher-Order statistics

Summary

This work presents a robust algorithm for voice activity detection (VAD) and noise reduction mechanism using combined properties of higher-order statistics (HOS) and an efficient algorithm to estimate the instantaneous Signal-to-Noise Ratio (SNR) of speech signal in a background of acoustic noise. The flat spectral feature of Linear Prediction Coding (LPC) residual results in distinct characteristics for the cumulants in terms of phase, periodicity and harmonic content and yields closed-form expressions for the skewness and kurtosis. The HOS of speech is immune to Gaussian noise and this makes them particularly useful in algorithms designed for low SNR environments. The proposed algorithm uses HOS and smooth power estimate metrics with second-order measures, such as SNR and LPC prediction error, to identify speech and noise frames. A voicing condition for speech frames is derived based on the relation between the skewness, kurtosis of voiced speech and estimate of smooth noise power. The algorithm presented and its performance is compared to HOS-only based VAD algorithm. The results show that the proposed algorithm has an overall better performance, with noticeable improvement in Gaussian-like noises, such as street and garage, and high to low SNR, especially for probability of correctly detecting speech. The proposed algorithm is replicated on DSK C6713.

Įvadas

Kalbos komunikacijose triukšmas yra nepastovus, papildantis išoriniais faktoriais informacijos (signalo) srautą, kurį gauna detektorius. Tai gali būti iš anksto numatyta, pvz. radijo ar vaizdo signalų trikdžiai, tačiau daugeliu atvejų tai yra traktuojama kaip nepageidaujamas numatytų veiksmų trukdymas. Daugelis kalbos apdorojimo sistemos vartotojų žino daugybę foninių triukšmų, esančių aplinkoje. Taip atsitinka todėl, kad jų laisvųjų rankų įranga sustiprina aplinkos triukšmą lygiai taip pat kaip ir pokalbį. Tebevyksta darbai, kuriais stengiamasi kuo įmanoma labiau susilpninti foninį triukšmą, teigiamai veikti kalbos aiškumą triukšmingoje aplinkoje.

Nors pastaruoju metu kalbos apdorojimas dirbtinai sukurtose sąlygose pasiekė didelį efektyvumą, tačiau realiame pasaulyje vis dar išlieka kalbos atpažinimo technologijų įdiegimo problemos. Viena iš problemų yra kalbos aptikimo pablogėjimas, tokiose aplinkose kaip biurai, automobiliai, gatvės, kompiuterių kambariai. Yra daugybė priežasčių, dėl kurių reikia triukšmą naikinti arba mažinti kalbos signale. Tačiau viena iš didžiausių problemų yra stengimasis išvengti kalbos elementų pašalinimo.

Šiuo darbu siekiama patobulinti ir įvertinti balso signalo aptikimo algoritmą, paremtą aukštesnės eilės statistika. Norint sukurti efektyvų, universalų kalbos atpažinimo metodą, reikia, kad triukšminga kalba vartojama realiame pasaulyje ir visi triukšmingoje aplinkoje atsiradę iškraipymai būtų įrašyti kalbos signalų duomenų bazėje. Bet neįmanoma surinkti kalbos duomenų, pačiose įvairiausiose triukšmo aplinkose. Kalbos ar balso signalo detektorius (VAD – Voice Activity Detector) siekia išskirti keletą akustinių foninio triukšmo tipų, netgi esant mažam signalo-triukšmo santykiui (SNR – signal-to-noise ratio).

Daugialypių terpių programų srityje VAD vienu metu sudaro galimybes balso ir duomenų programoms. Panašiai [1] jis kontroliuoja ir naikina vidutinę bitų perdavimo spartą ir gerina kalbos kodavimo kokybę universaliose mobiliųjų telekomunikacijų sistemose (UMTS – Universal Mobile Telecommunications Systems). Tinklinėse (cellular) radijo sistemose (pvz. GSM ir CDMA sistemose), pagrįstose nepertraukiamu perdavimo metodu (DTX – Discontinous Transmission mode), šis gebėjimas yra esminis, leidžiantis gerinti

sistemos galingumą, šalinant bendrųjų kanalų (co-channel) trikdžius ir galios sunaudojimą nešiojamuose skaitmeniniuose įrenginiuose [12], [7], [8].

Norint pagerinti balso atpažinimą, visų pirma reikia išanalizuoti tai, kas šioje srityje buvo daroma, kaip yra aptinkamas balso signalas ir pagal kokius kriterijus yra vertinamas atpažintas signalas. Esant foniniam triukšmui labai sunku išskirti triukšmą ir tylą, todėl kalbos aptikimui ir triukšmo mažinimui besikeičiančiame ir kenksmingame triukšmo akustiniame fone, reikalingi daug efektyvesni ir save sustiprinantys (self-sustainning) algoritmai. VAD algoritmuose kalbos aptikimui yra naudojami skirtingi parametrai. Tačiau pastaruoju metu aukštesnės eilės statistika (HOS – Higher-order Statistics) parodė potencialius rezultatus, esančius daugybėje signalo apdorojimo programų. HOS yra labai svarbi ir efektyvi, dirbant su maišytais gausiniais ir negausiniais procesais ir su netiesinėmis sistemomis [9].

Pirmoje darbo dalyje analizuojamos liekamojo kalbos signalo (po tiesinio prognozavimo kodavimo (LPC – Linear prediction coding)) trečios ir ketvirtos eilės kumulantų (cumulants) charakteristikos. Liekamojo signalo plokščias spektrinis vokas pasireiškia aiškiom šių kumulantų savybėmis: fazės, periodiškumo ir harmoniškumo išraiškoje ir atskleidžia asimetrijos ir eksceso koeficientams artimos formos išraiškas, pagrįstas harmoniniu kalbos modeliu.

Įvertinus visus svarbius kalbos atpažinimui parametrus, yra kuriamas ir tobulinamas kalbos atpažinimo algoritmas, paremtas aukštesnės eilės statistika. Pasiūlytasis algoritmas yra testuotas su skirtingo SNR lygio įvairiais triukšmo tipais, tokiais kaip gatvės, automobilio, garažo, traukinio. *Efektyvumo įvertinimas* – tai ne tik teisingai klasifikuojamų kalbos ir triukšmo kadrų tikimybė, bet ir blogos klasifikacijos tikimybė. Jos yra apskaičiuojamos lyginant su tikrąja pavyzdine byla (marker file), padaryta švariam kalbos signalui.

Šių parametrų apskaičiavimui ir triukšmingo kalbos signalo variantų generavimui, bei VAD algoritmo efektyvumo įvertinimui yra naudojama TIA duomenų bazės medžiaga (žr. 2 priedą). Sukūrus ir ištestavus algoritmo modelį, jis parašytas C programavimo kalba ir realizuotas TMS320C6713 DSP Starter Kit (DSK) plokštėje, naudojant Code Composer Studio (CCS) programinę įrangą.

Dalis šio darbo buvo daroma Aalborgo universiteto elektroninių sistemų instituto komunikacijos technologijų katedroje, Danija.

2 Analizė

Šiame skyriuje analizuojama tai, kas buvo iki šiol daroma kalbos atpažinimo srityje. Taip pat kaip yra aptinkamas balso signalas, balso atpažinimo algoritmas, jo charakteristikos. Kaip pridėtinis triukšmas (Gausinio triukšmo atveju), sugadinęs švarų kalbos signalą, yra suspaudžiamas ar izoliuojamas?

2.1 Kalbos atpažinimo tyrimai ir taikymai

Jau 1952 metais, atpažįstant kalbą, tyrinėtojai deklaravo 98% tikslumą ir įdomu tai, kad šis skaičius praktiškai nepakito iki dabar. Pirmi bandymai sukurti automatinio kalbos atpažinimo sistemą buvo atlikti Belo laboratorijose [26]. Tai buvo izoliuotų skaičių atpažinimo sistema, skirta vienam kalbėtojui ir rėmėsi kiekvieno skaičiaus balsių srityje spektrinių rezonansų matavimu. Sistema matavo dvejose plačiose dažnių juostose spektrinės energijos tam tikrą nesudėtingą funkciją laike, tokiu būdu grubiai aproksimuodama pirmas dvi formantes. Nors sistemos analizė buvo grubi, jos žodžio ilgio spektro matavimai buvo gana robastiški kalbos nepastovumui, galbūt labiau robastiški negu kai kurie vėliau naudojami laike kintančio spektro matavimo metodai. Ji matavo grubų formančių kelią, o ne patį spektrą. Tai yra potencialiai atsparu nereikšmingoms kalbos spektro modifikacijoms. Pvz., paprastas kalbėtojo galvos pasukimas nuo klausytojo dažnai sukelia pastebimus kalbos spektro pasikeitimus (ypač aukštesnių spektro komponenčių amplitudės sumažėjimą). Belo laboratorijos sistemos spektro įvertis buvo gana grubus, žemų ir aukštų dažnių spektro momentų histogramavimas per visą pasisakymą ir tokiu būdu kitimas laike buvo prarastas. Nors idėja gera, to meto technologija buvo per silpna, kad galima būtų šią sistemą stipriai tobulinti. Sistema naudojo analogines elektrines komponentes ir sunkiai buvo modifikuojama. Nepaisant to, išradėjai tvirtino, kad sistema dirba gana gerai, vienam kalbėtojui tariant skaičius, kurie buvo izoliuoti pauzėmis, pasiekia 2% klaidingumą. Kalbos signalas buvo

filtruojamas į žemų ir aukštų dažnių komponentes ir kiekviena komponentė stipriai ribojama taip, kad jos amplitudė nepriklausė nuo signalo stiprumo. Šiems signalams buvo skaičiuojami nulio kirtimai ir sistema naudojo nulių kirtimų reikšmes, vertinant kiekvienos dažnių juostos centrinį dažnį. Žemų dažnių juosta buvo kvantuojama į vieną iš šešių 100 Hz subjuostų (tarp 200 ir 800Hz), aukštų dažnių juosta buvo kvantuojama į vieną iš penkių 500 Hz subjuostų. Kartu šios dvi kvantuotos reikšmės atitinka vieną iš 30 galimų dažnių porų. Skaičiai turėjo atskiriamus dažnių porų pasiskirstymus ir taip buvo vienas nuo kito atskiriami.

Balsių srityje spektrų matavimai, gauti lygiagrečių filtrų pagalba, buvo panaudoti ir bandant atpažinti vieno kalbėtojo dešimt skirtingų skiemenų [27]. Dudley sukūrė klasifikatorių, kuris vertino spektro kitimą laike. Šis būdas buvo plačiai paplitęs, o dabar kalbos atpažinimui dažniausiai naudojama kuri nors iš kintančių laike lokalinių spektro įverčių funkcijų, kurios vaizduoja atpažįstamą kalbą. Fry ir Denes [28] bandė sukurti fonemų atpažinimo mechanizmą, kuris galėtų atpažinti keturias balses ir penkis priebalsius. Atpažinimo tikslumo gerinimui, čia buvo panaudota statistinė informacija apie galimas fonemų sekas ir tam tikrus spektrinių objektų tapatinimo būdus. Tai buvo bandymas šalia akustinės informacijos panaudoti gramatikos tikimybes. Buvo iškelta mintis, kad konkretaus lingvistinio vieneto tikimybė gali būti priklausoma nuo ankstesnio lingvistinio vieneto, taip, kad žodžio tikimybė nėra priklausoma vien tik nuo akustinio įėjimo. Sekantis to laikotarpio bandymas buvo Forgie balsių atpažinimo aparatas [29]. Šis aparatas galėjo atpažinti 10 balsių, esančių tarp priebalsių /b/ ir /t/ ir atpažinimas buvo nepriklausomas nuo diktoriaus. Spektrinei informacijai gauti buvo naudojamas juostinių filtrų analizatorius. Balso trakto rezonansų radimui buvo naudojamas kintantis laike rezonansų įvertis.

Sakai ir Doshita 1962 metais [30] realizavo fonemų atpažinimo įrenginį: sprendimų priėmimui buvo panaudotas aparatūrinis kalbos segmentavimas ir nulio kirtimų analizė. Tuo metu taip pat aktyviai buvo pradėta spręsti kalbos vienetų laiko skalės nevienodumo problema. Martin [31] sukūrė eilę nesudėtingų laiko skalės normalizavimo metodų, kurie rėmėsi patikimu kalbos pradžios ir galo momentų nustatymu. Beveik tuo pat metu, Vinciuk [32] laiko skalės vienodinimui pasiūlė naudoti dinaminio programavimo metodus. Svarbus pasiekimas buvo Reddy tyrimai [33], kuriuose nepertraukiamos kalbos atpažinimui panaudotas dinaminis fonemų kitimo sekimas. 6-tajame dešimtmetyje buvo sukurti trys spektro įverčių metodai, kurie vėliau tapo labai svarbūs kalbos atpažinimui. Šie metodai pirmiausiai buvo pritaikyti kalbos kodavimui: Greita Furjė transformacija (FFT), kepstrinė (arba homomorfinė) analizė ir tiesinio prognozavimo kodavimas (LPC). Buvo sukurti nauji pavyzdžių tapatinimo metodai:

deterministinis metodas, vadinamas dinaminiu laiko skalės kraipymu (DTW), ir statistinis metodas, vadinamas paslėptu Markovo modeliu (HMM).

Advanced Research Projects Agency (ARPA) finansavo didelį kalbos suvokimo projektą. Tikslas buvo sukurti 1000 žodžių automatinį kalbos atpažinimą, naudojant kelis kalbėtojus, ištisinę kalbą ir apribotą gramatiką su mažesniu nei 10% semantinių klaidų kiekiu. Tačiau tik HARPY sistema, sukurta CMU doktoranto Bruce Lowerre, tenkino keliamus reikalavimus. Jis naudojo LPC segmentus, gramatikos žinias ir „Baker Dragon system“, bei CMU sistemos Hearsey modifikuotus metodus. Tyrinėtojai savo atpažinimo sistemose naudojo spektrinius požymių vektorius, LPC ir fonetinius požymius. Buvo sukurti metodai, naudojantys DTW, HMM ir neuroninius tinklus. Buvo stengiamasi sutrumpinti pavyzdžių palyginimo trukmę. Dažnai buvo naudojami dirbtinio intelekto metodai, ypač ARPA programoje. Automatiniam kalbos atpažinimui buvo pritaikyta HMM teorija ir buvo sukurtos HMM besiremiančios sistemos.

Vėliau kalbos atpažinimo tyrinėtojų dėmesys persikėlė nuo izoliuotų žodžių atpažinimo prie kalbos, sudarytos iš sujungtų žodžių (connected word), atpažinimo [34-36]. Tyrimų tikslas buvo sukurti robastinę sistemą, sugebančią atpažinti sklandžiai pasakytą žodžių seką. Tyrimų objektu tapo statistiniai modeliavimo metodai, ypač Paslėptų Markovo Grandinių (HMM) metodas [37-38]. Kita fundamentalia tyrimų sritimi, kuri pradėjo sparčiai vystytis, tapo neuroninių tinklų panaudojimas kalbos atpažinimo problemų sprendimui [39-40]. Didelio žodyno nepertraukiamos kalbos atpažinimo sistemų kūrimui didelį impulsą suteikė DARPA (Defence Advanced Research Projects Agency) projektai, kuriuos plėtojant buvo labiau orientuojamasi į natūralios kalbos apdorojimą ir taikymus, tokius kaip telefoninių tinklų operatoriaus darbo palengvinimas. 1984 metais ARPA pradėjo finansuoti antrą programą, kurios tikslas buvo sakinių skaitymas iš 1000 žodžių žodyno. Sakiniai buvo klausimai ir komandos skirtos naudotis jūrų informacijos duomenų baze, nors sistema iš tikrųjų neturėjo sąsajos su jokia duomenų baze: vertinimas vyko pagal žodžių atpažinimą. Projekte dalyvaujančių sistemų vertinimas vyko kartą ar du per metus. Dalyviai gaudavo CD-ROM-ą su testiniais duomenimis ir siųsdavo į NIST atpažinimo sistemos generuotas frazes, kur rezultatai būdavo oficialiai įvertinami.

ARPA projektas sudarė labai geras sąlygas tobulinimams. Daug laboratorinių sistemų gali atpažinti naujo kalbėtojo kalbą (be apmokymo konkrečiam kalbėtojui) su 60000 žodžių žodynu realiame laike, gaunant mažesnį nei 10% klaidingumą. Tai įkvėpė kitus tyrinėtojus, kurie nebuvo finansuojami iš šio projekto, tame tarpe ir tyrinėtojus iš Europos. Pavyzdžiui,

Kembridžo universitetas Anglijoje dalyvavo įvertinime ir sukūrė HTK arba HMM įrankių komplektą (ToolKit), kuris buvo plačiai išdalintas. Dabar yra įmanoma naudoti HTK, kad būtų galima gauti didelio žodyno atpažinimo rezultatus panašius į tuos, kuriuos gavo daugelis projekto dalyvių.

Per paskutinį dešimtmetį kalbos atpažinimo srityje nebuvo sukurtas joks bazinis mechanizmas, tačiau buvo daug svarbių pasiekimų: apdorojimo (front-end) pasiekimai (pvz., melų arba barkų skalės kepstrų įverčiai, delta požymiai, kanalo normalizacijos būdai ir balso trakto normalizacija) ir tikimybinis įvertinimas (pvz., maksimalaus tikėtinumo tiesinė regresija, naujų kalbėtojų ar akustinių sistemų adaptacija, arba mokymas maksimizuojant tarpusavio informaciją tarp duomenų ir modelių). Todėl galima teigti, kad dirbančių toje srityje pastangos yra orientuotos link egzistuojančių idėjų efektyvumo pagerinimo nei link naujų idėjų generavimo. Tai turėjo tendenciją konverguoti į geras, panašias sistemas, kiekvienai laboratorijai bandant pasinaudoti patobulinimais, kurie pavykdavo kitiems. Nors tai vedė prie greito tobulėjimo, bet tai taip pat vedė metodų konvergavimą į vieną metodą.

Taip pat šis laikotarpis pasižymi masiniu kalbinių technologijų taikymu įvairiose srityse: telekomunikacijose, informacinėse tarnybose, mokyme, tarptautiniuose komerciniuose ryšiuose. Daugelyje šių taikymų kaip pagrindinė arba kaip sudedamoji sistemos dalis yra kalbos atpažinimas. Išsiplėtus taikymų sričiai, kartu labai pagriežtėjo reikalavimai kalbos atpažinimo tikslumui. Reikia patikimai atpažinti kalbą, esant triukšmingai aplinkai, dideliems ryšio kanalo iškraipymams, ryšio kanalo dažnių juostos apribojimams (pavyzdžiui telefonijoje). Dėl to labai suintensyvėjo tyrimai kalbos atpažinimo srityje. Bandoma surasti požymius, kurie leistų atpažinti kalbą esant sudėtingoms ryšio sąlygoms, triukšmui, panaudoti lingvistines žinias kalbos atpažinimo patikimumo padidinimui.

2.1.1 Didelių balsų bazių rinkimas

Iki 1986 metų kalbos atpažinimo tyrinėtojai sistemų mokymui ir testavimui neturėjo plačiai pripažintų balsų bazių atpažinimo. Dėl to buvo sunku palyginti gautus rezultatus. Su šia problema buvo susiję daug kalbos tyrinėtojų. Mokslininkai, dirbantys pramonėje (pvz., Texas Instruments ir Dragon Systems) dirbo su NIST (National Institute of Standards and Technology) ir surinko didelę balsų bazę.

1986 metais buvo pradėta kurti TIMIT balsų bazė, kuri vėliau tapo pirma plačiai naudojama standartine balsų baze. Fonetiniam savitumui pavaizduoti buvo parinktas 61 garso alfabetas. Buvo parinkti fonetiškai subalansuoti tekstai, kad mokymo aibėje būtų gerai atstovaujamas kiekvienas garsas. 630 kalbėtojų pasakė po 10 sakinių, iš kurių du buvo visiems kalbėtojams tie patys. Duomenis įrašė Texas Instruments, fonetiškai segmentavo MIT pirmiausiai naudodamas automatinį segmentatorių, vėliau magistrantai tikrino ir taisė klaidas. Taip buvo gauta bazė, kurioje kiekvieno bazės garso ribos buvo pažymėtos. Nors klaidų šioje bazėje dar yra, bet ji iki šiol plačiai naudojama.

1984 metais prasidėjus antrai ARPA kalbos programai (D)ARPA buvo suformuluotas naujas tikslas - resursų valdymas (RM – Resource Management) su nauja balsų baze. RM bazėje balso įrašai buvo sukurti skaitant tekstą. Sakiniai buvo sukonstruoti iš 1000 žodžių kalbos modelio. Buvo įrašyta 21000 pasakymų, kuriuos ištarė 160 kalbėtojų. RM tiksluose buvo nepriklausomas nuo kalbėtojo atpažinimas. Kai kurios sistemos buvo apmokytos daugelio kalbėtojų ir testuojamos su kalbėtojais, neįeinančiais į mokymo aibę. Vėliau, vykdant RM programą, dėmesys persikėlė į „Wall Street Journal“ tikslą - atpažinti skaitomą kalbą iš „Wall Street Journal“. Pirmas testas buvo atliktas su 5000 žodžių žodynu, be žodžių neesančių žodyne. Antras testas – su 20000 žodžių su žodžiais neesančiais žodyne. Vėlesni testai buvo su iš esmės neribotu žodynu. Tyrinėtojai sistemos testavimui dažnai naudojo 60000 žodžių dekoderius.

Lygiagrečiai RM buvo suformuluotas kitas tikslas „Air Travel Information System“ (ATIS), kuris rėmėsi spontaniškais užklausimais rezervuojant lėktuvų bilietus. ATIS tikslas yra kalbos supratimas (priešingai negu kalbos atpažinimas). Sistemos ne tik turėjo sukurti žodžių sekas, bet ir bandyti suteikti joms semantinę prasmę, kad galėtų atlikti atitinkamą funkciją. Pavyzdžiui, jeigu vartotojas pasakydavo "parodyk man lėktuvų skrydžius iš Bostono į San Franciską", sistema turėjo reaguoti parodydama skrydžių sąrašą. Bendravimas tęsdavosi tol kol būdavo pasiekiamas kažkoks tikslas; šiuo atveju užsakyti lėktuvo bilietus. Ši sritis buvo praktiškesnė negu „Wall Street Journal“ skaitymas, bet žodyno dydis buvo mažesnis.

DARPA finansavo šių bazių rinkimą. Rinkimo procesą koordinavo NIST. Ši ir kitos balsų bazės dabar vartotojams yra duodamos „Linguistic Data Consortium“, kuris yra Pensilvanijos universitete, Filadelfija.

2.2 Balso signalo aptikimas

Šnekamosios kalbos ir tylos atskyrimo procesas vadinamas balso signalo aptikimu (VAD – voice activity detection). Pirmiausiai jis buvo ištirtas, naudojant TASI (Time Assigned Speech Interpolation) sistemas. VAD yra svarbi technologija daugeliui kalba pagrįstų programų, įskaitant kalbos atpažinimą, kodavimą ir laisvų rankų telefoniją. Dėl šių priežasčių buvo pasiūlyti įvairūs VAD algoritmo tipai, kurie keitė vėlavimo, jautrumo, tikslumo ir skaičiavimų kainą.

Pirminė balso signalo aptikimo funkcija yra balso buvimo signale indikacija, siekiant palengvinti ne tik kalbos apdorojimą, bet ir kalbos pradžios ir pabaigos segmentų skyriklių indikaciją [5]. Didelei grupei programų, tokių kaip skaitmeninis mobilusis radijas, skaitmeniniai vienalaikiai balsas ir duomenys (DSVD – Digital Simultaneous Voice and Data) ar kalbos saugojimas, yra norima suteikti nenutrūkstamą kalbos kodavimo parametrų perdavimą. Privalumais gali būti žemesnis mobiliųjų telefonų rageliuose sunaudojamos galios vidurkis ar didesnis bitų perdavimo spartos vidurkis lygiagrečiam paslaugų teikimui (pvz. duomenų perdavimui), ar net didesnis atminties lustų talpumas. Tačiau tobulinimas labiausiai priklauso nuo kalbos metu padarytų pauzių procentinio kiekio ir VAD patikimumo atpažįstant šiuos intervalus. Iš vienos pusės, mažas kalbos signalo procentas yra privalumas, iš kitos pusės, turi būti išvengta kalbos signalo apkarpymo, norint išlaikyti kokybę. Tai yra kritinė VAD algoritmo problema, esant stipraus triukšmingumo sąlygoms [10].

Kalbos signalo aptikimas yra svarbus kalbos perdavimui, tobulinimui ir atpažinimui. Šią užduotį apsunkina gausybė kintančių kalbos rūšių ir foninių triukšmų [6]. Ankstesnieji VAD algoritmai yra pagrįsti Itakura LPC atstumų matu, energijos lygiais, laiko planavimu, tonu, nulio kirtimų dažniu, kepstrinėmis savybėmis, prisitaikančiu kalbos signalui triukšmo modeliavimu ir periodiškumo matu. Deja šie algoritmai turi problemų kai SNR reikšmės yra mažos, ypač kai triukšmas nestacionarus. Pastovaus tikslumo negalima pasiekti, kadangi daugelis algoritmų priklauso nuo palyginimui naudojamo rėžio lygio. Šis rėžio lygis dažnai laikomas fiksuotas arba apskaičiuojamas tylos (kai kalbos nėra) intervaluose [18]. Pastarąjį dešimtmetį daugelis tyrinėtojų studijavo skirtingas kalbos aptikimo triukšme strategijas ir VAD įtaką kalbos apdorojimo sistemose [2].

2.2.1 VAD algoritmas

VAD algoritmo pagrindinė funkcija yra iš įvesties signalo išskirti tam tikras savybes ar skaičius ir palyginti šias reikšmes su rėžiais, kurie paprastai surandami iš triukšmo ir kalbos signalo charakteristikų. Jei signalo matavimų reikšmės telpa tarp rėžių, laikoma, kad tai yra balsas. Esant nestacionariam triukšmui, VAD reikalingos laike kintančios rėžių reikšmės. Ši reikšmė paprastai yra apskaičiuojama segmentuose, kur nėra balso [18].

Pastaruoju metu pristatyti VAD metodai apibrėžia taisyklę, pagrįstą kadras po kadro atliekamais matavimais lyginant kalbą ir triukšmą [2]. VAD metode naudojami šie matavimai: spektrinis nuotakumas (spectral slope), koreliacijos koeficientai, registracijos tikėtinumo santykis (log likelihood ratio), kepstras, modifikuoti atstumo matavimai.

VAD galima suskaidyti į du etapus: parametrų skaičiavimas ir klasifikavimo programa. Nepriklausomai nuo VAD metodo, operacija yra kompromisas tarp balso atpažinto triukšmu arba triukšmo atpažinto balsu [6]. VAD veikiantis mobilioje aplinkoje turi sugebėti aptikti kalbą labai skirtingų akustinio foninio triukšmo tipų diapazone. Šiomis sudėtingomis aptikimo sąlygomis yra labai svarbu, kad VAD veiktų „be avarijų“, nustatant „kalba aptikta“ tuo atveju, kai yra abejonė, taip išvengiant kalbos apkarpymų. Didžiausias kalbos aptikimo sudėtingumas yra labai mažas signalo-triukšmo santykis (SNR). Kalbos ir triukšmo skirtumą įmanoma nustatyti, naudojant paprastas lygio aptikimo technikas, kai dalis kalbos pasireiškimų yra paslėpta žemiau triukšmo lygio [16].

Universalus balso signalo aptikimo algoritmas tapo reikalingas tuomet, kai tradiciniai sprendimai davė didelį klaidingo aptikimo koeficientą, esant foniniams triukšmams būdingiems mobiliosiose aplinkose. Vienas svarbus aspektas šiuolaikinėse skaitmeninėse sistemose yra kalbos kodavimo algoritmų universalumas, kuris yra reikalingas efektyviam kanalų panaudojimui. Algoritmai turi būti universalūs ne tik blogėjimui sumažinti, bet ir foninio triukšmo būdingo mobiliosiose aplinkose sumažinimui [3]. Universalumą galima apibūdinti taip: „VAD yra universalus tuo atveju, jei balso aptikimas yra artimas kontroliniam tyliose aplinkose lygiai taip pat kaip ir triukšmingose aplinkose“. Naujas apibrėžimas teigia, kad VAD yra universalus tada, kai gaunami panašūs rezultatai esant švariai ir triukšmingai kalbai. Universalumas gali būti įvertintas laikant VAD rezultatus, esant švariai kalbai, kontroliniais ir skaičiuojant klaidų statistiką, kai tas pats VAD apdoroja triukšmingą kalbą. Kuo universalesnis VAD, tuo rečiau pasitaiko klaidos [6].

2.2.2 VAD įvertinimas

VAD charakteristikomis galima vadinti aktyvumą, apkarpymų griežtumą ir laipsnį. Norint įvertinti apkarpymo kiekį ir kaip dažnai triukšmas palaikomas kalba, VAD išvestis yra lyginama su idealaus VAD išvestimi. VAD charakteristikos vertinamos pagal keturis įprastinius parametrus:

1. Pradžios apkarpymo (FEC – Front End Clipping): pereinant iš triukšmo į kalbą.
2. Kalbos apkarpymo (MSC – Mid Speech Clipping): kai kalba palaikoma triukšmu.
3. Pertekliaus: kai dėl užsilikusios aktyvios VAD vėliavėlės pereinant iš kalbos į triukšmą, triukšmas palaikomas kalba.
4. Triukšmo atpažinimo, kaip kalbos (NDS – Noise Detected as Speech): triukšmas interpretuojamas kaip kalba su tylos periodu.

Nors anksčiau apibūdintas metodas duoda naudingą objektyvią informaciją, tačiau tai tik pradinis įvertinimas subjektyvaus efekto atžvilgiu. Todėl yra svarbu perkelti subjektyvius testus į VAD, kur pagrindinis tikslas yra pastebėtų apkarpymų kiekio priimtinumas. Šios rūšies testai reikalauja tam tikro kiekio klausytojų, kurie vertintų VAD rezultatų įrašus. Klausytojai turi atkreipti dėmesį į šias savybes:

1. Kokybę.
2. Supratimo sunkumą.
3. Apkarpymų girdimumą.

Šios savybės, gautos keletą kartų klausantis kalbos, yra naudojamos, apskaičiuojant rezultatų vidurkį kiekvienai savybei atskirai, taip yra gaunamas bendras VAD elgsenos įvertinimas. Nors objektyvūs metodai yra labai naudingi pradinėje VAD kokybės stadijoje, bet subjektyvūs metodai yra labiau reikšmingi. Tačiau jie yra ir brangesni, kadangi kelioms dienoms yra reikalingas tam tikras žmonių kiekis. Paprastai jie yra naudojami tuo atveju, kai pasiūlytą algoritmą ruošiamasi standartizuoti [3].

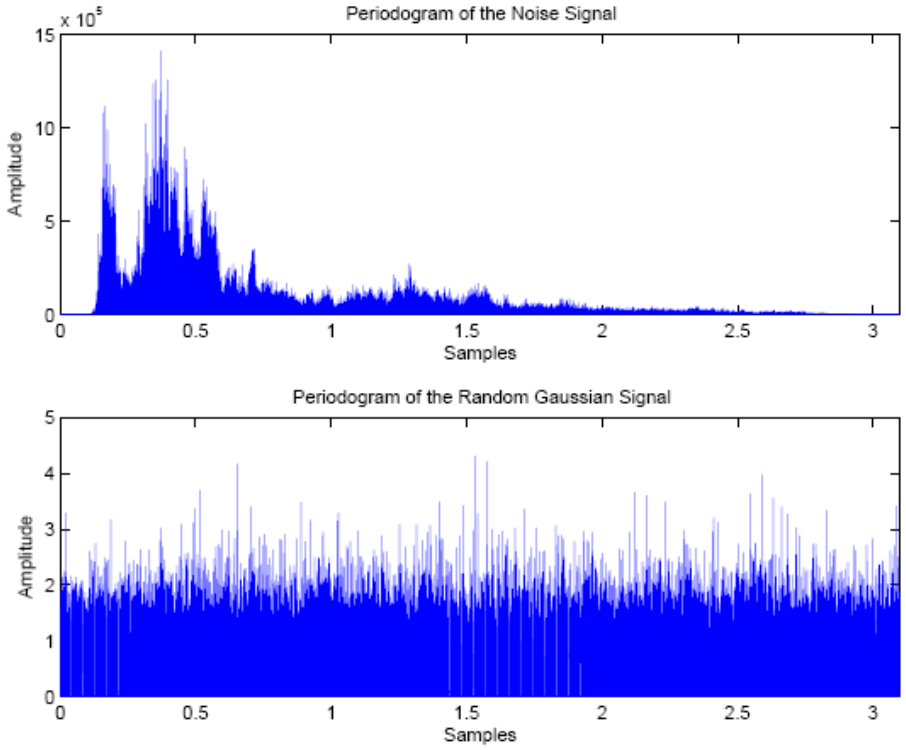
Viena iš pagrindinių HOS VAD naudojimo priežasčių yra gebėjimas nuslopinti spalvotą triukšmą.

2.3 Triukšmas

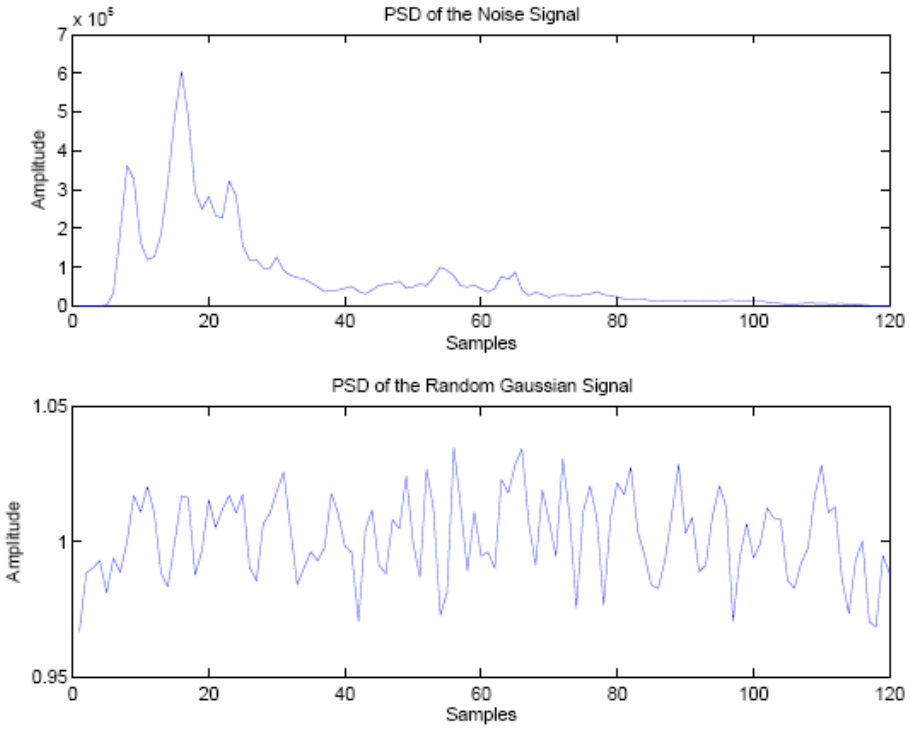
Triukšmą galima apibūdinti kaip trokštamo ar nenorimo signalo užteršimą. Natūralaus ir dirbtinio triukšmo šaltiniai gali atlikti arba atsitiktinį, arba modeliuotą įsikišimą. Tik pastarasis gali būti efektyviai pašalintas analoginėse sistemose. Tačiau skaitmeninės sistemos yra sukonstruotos taip, kad jų kvantuotieji signalai gali būti tobulai atkurti tol, kol triukšmo lygis lieka žemiau apibrėžto maksimumo, kuris kinta iš programos į programą. Yra daugybė triukšmo su įvairiomis dažnių charakteristikomis formų, kurios klasifikuojamos pagal „spalvas“ [19].

Baltas triukšmas – tai signalas (ar procesas), turintis vienodą dažnių spektrą. Kitaip tariant, signalas yra vienodos galios bet kuriame diapazone, bet kurio dažnio centre, turintis užduotą pralaidumą. Praktikoje signalas gali būti „baltas“, esant vienodam spektrui už apsibrėžto dažnių diapazono. „Baltas“ signalas dažnių srityje turi turėti tam tikras svarbias statistines savybes laike. Pavyzdžiui, jis, laikui bėgant, turi turėti nulinę autokoreliaciją, išskyrus nuliniame laiko poslinkyje (at zero timshift). 2 paveiksle matyti, kad automobilio triukšmas (ištyrus 10000 mėginių) nėra baltas. Iš periodinės diagramos (periodogram, 1 pav.) matyti, kad spektras yra nevienodas, kai tuo tarpu atsitiktinai sugeneruotas Gausinis triukšmas turi tolygų pasiskirstymą. Galios spektrinis tankumas yra glotni periodų diagramos versija.

Triukšmas, turintis tolydų pasiskirstymą, tokį kaip normalusis pasiskirstymas, gali būti baltas [20]. Gausinis triukšmas kartais klaidingai laikomas baltu gausiniu triukšmu, tačiau taip nėra. Gausinis triukšmas reiškia triukšmą su tikimybine pasiskirstymo funkcija (PDF – Probability Distribution Function), kuri rodo triukšmo nekoreliaciją laike. Gausinis triukšmas, vadinimas baltu, apibūdina triukšmo koreliaciją.



1 pav. Testas automobilinio triukšmo baltumui



2 pav. Testas automobilinio triukšmo tipui

Kitas dažniausiai vartojamas spalvotas triukšmas yra rausvas triukšmas (pink noise). Jo dažnių spektras yra nelygus, bet turi vienodą galią diapazone, kuri yra proporcingai plati. Rausvas triukšmas yra subjektyviai baltas. Taip yra todėl, kad žmogaus klausos sistema jaučia apytiksliai lygų visų dažnių didumą. Galios tankumas sumažėja iki -3dB per oktavą, padidindamas dažnį (tankumas proporcingas 1/f). Taip pat yra daugybė „mažiau oficialių“ triukšmo spalvų tokių kaip ruda, mėlyna, violetinė, pilka, raudona, oranžinė, žalia ir juoda.

2.3.1 Pridėtinis triukšmas

Yra daugybė akustinio iškraipymo šaltinių, kurie gali sumenkinti kalbos atpažinimo sistemų veiklumą. Daugeliui kalbos atpažinimo programų priemaišinis triukšmas yra svarbiausias akustinio iškraipymo šaltinis [14]. Daug mokslininkų pastangų buvo atiduota, bandant kompensuoti priemaišinio triukšmo efektą universaliame kalbos atpažinime.

Jei kalbos signalas *s(k)* yra paveiktas nekoreliuotu triukšmu *n(k)* [17], tai gautas signalas dažnių srityje bus išreikštas:

$$Y(e^{j\omega}) = X(e^{j\omega}) + N(e^{j\omega}) \quad (1)$$

Jei *s(t)* yra originalus švarus kalbos signalas, tai gautas kalbos signalas *y(t)* laiko srityje gali būti išreikštas:

$$y(t) = s(t) * h(t) + n(t) = x(t) + n(t) \quad (2)$$

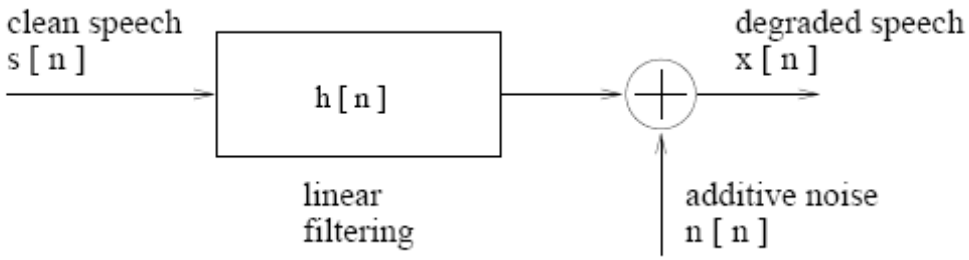
, kur *h(t)* – kanalo iškraipymo staigi reakcija (impulse response),

n(t) – aplinkos triukšmas,

* žymi sujungimo operaciją,

x(t) – kalba be triukšmo (2.2.1 paveikslas).

Tipiški struktūriniai kintamumo adaptacijai modeliai priima, kad kalba yra iškraipyta, jungiant priemaišinį triukšmą ir tiesinį filtravimą.



3 pav. Priemaišinis triukšmas

Kalbos apdorojime, kalba yra traktuojama kaip naudingi duomenys, o visi kiti signalai yra laikomi triukšmu. Daug algoritmų ir programų sukurta, siekiant sumažinti ar šalinti triukšmą iš signalo, vienas iš jų ir yra balso signalo aptikimas.

2.4 Aukštesnės eilės statistikos (HOS) pasirinkimas

Ankstesniuose VAD algoritmuose trumpalaikė energija, nulio kirtimų dažnis ir LPC koeficientai buvo vieni iš pagrindinių parametrų, naudojamų kalbos aptikimui. Kepstro savybių ir mažiausiųjų kvadratų periodiškumo matavimai yra dažniausiai naudojami VAD projektavime. G.729B VAD turi parametrų rinkinį, įtraukiantį ir linijinius spektro dažnius (LSF – line spectral frequencies), mažą diapazono energiją (low band energy), nulinio lygmens dažnį ir pilną diapazono energiją (full band energy).

Trumpalaikė ar spektrinė energija tradiciškai buvo naudojama kaip esminis parametras išskiriantis kalbos segmentus iš kitų signalo formų. Tačiau šios savybės tapo mažiau patikimos ir universalios triukšmingose aplinkose, ypač esant nestacionariam triukšmui ir tokiems garsams kaip lūpų pliaukštelėjimas, stiprus kvėpavimas ir burnos takštelėjimas ir kt. [11].

HOS davė gerus rezultatus daugelyje signalo apdorojimo programų ir yra labai reikšminga dirbant su Gausinio ir negausinio procesų sumaišymais, ir netiesinėse sistemose. Kalbos apdorojime HOS programa yra Gausinio suspaudimo ir fazės išsaugojimo ypatybės.

3 Algoritmo sudarymas

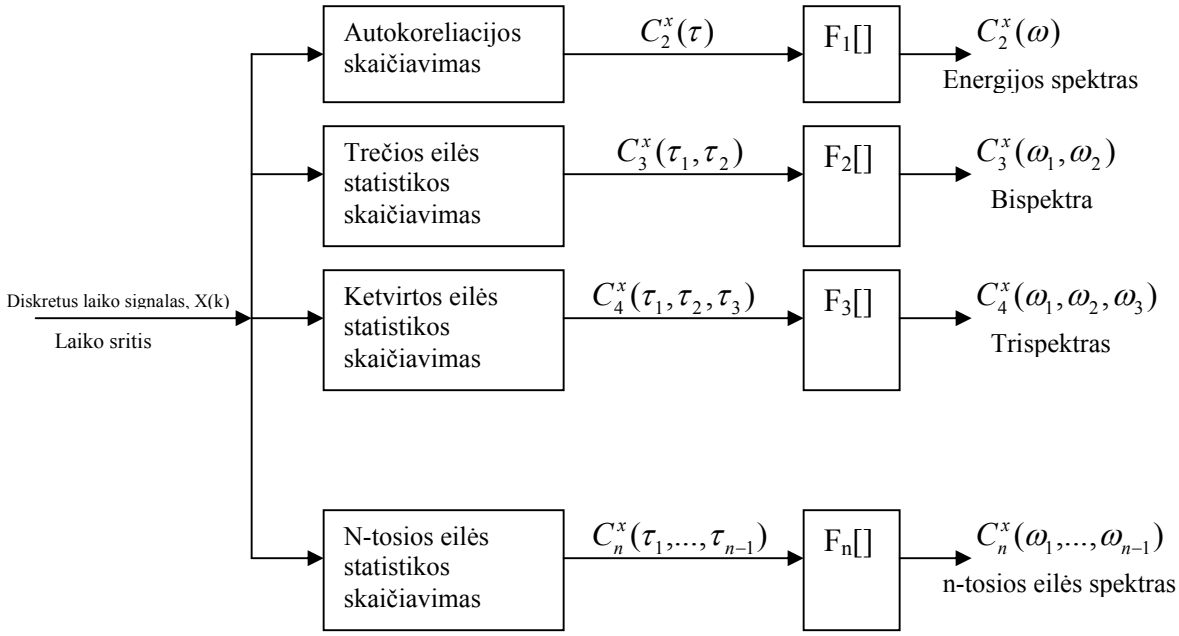
Šiame skyriuje aptariami įvairūs algoritmai, naudoti projektuojant balso signalo aptikimo algoritmą. Atlikus tiesinį prognozavimo kodavimą, liekamojo balsinio signalo asimetrijos ir eksceso koeficientai yra išreikšti harmonikų M skaičiumi ir signalo energija. Kiekvienai M reikšmei, kuri yra garso aukščio funkcija, šie parametrai yra geresni ir gali būti naudojami aptikti balsą. Normalizuotų parametrų privalumas yra tai, kad jie nepriklauso nuo signalo energijos, dėl to ir yra naudojami absoliutiniai slenksčiai. Norint gauti bloko dispersijos įvertį, asimetrijos ir eksceso koeficientų įverčių pasiskirstymas yra normalizuojamas. Balso signalui apskaičiuotas sąryšis tarp asimetrijos ir eksceso koeficientų yra naudojamas atpažinti balso kadrams. Visa tai sudaro VAD algoritmo pagrindą, naudojant HOS.

3.1 Aukštesnės eilės spektras ir statistikos

Diskretaus laiko deterministinio ar stochastinio signalo energijos spektrinio tankumo, ar paprasčiau energijos spektro, apskaičiavimas yra naudinga priemonė skaitmeninių signalų apdorojimui. Energijos spektro skaičiavimo būdai yra esminiai, kuriant pažangias radarų, sonarų, komunikacijų, kalbos, biomedicinos, geofizines ir kitas duomenų apdorojimo sistemas [13]. Energijos spektro skaičiavimui signalas apdorojamas taip, kad apskaičiuotas energijos pasiskirstymas tarp dažnio komponenčių, suspaudžia fazės ryšius tarp šių komponenčių. Energijos spektro teikiama informacija yra esminė tai, kuri yra pateikiama autokoreliacijos sekoje. To pakanka pilnam gausinio signalo statistiniam apibūdinimui. Tačiau praktikoje žiūrima plačiau nei energijos spektras ir siekiama išskirti informaciją apie fazės ryšius ir nukrypimus nuo gausinio signalo.

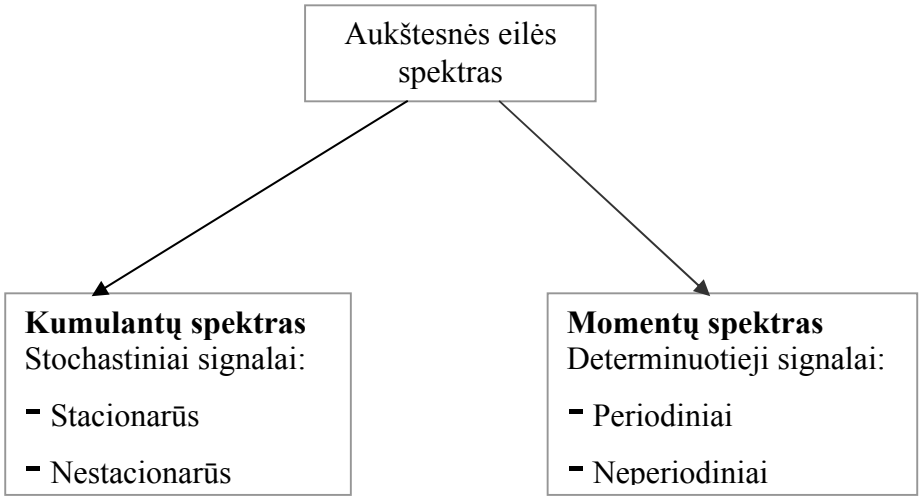
Aukštesnės eilės spektras (polispektras) apibūdinamas signalo aukštesnės eilės statistikomis (kumulantais). Aukštesnės eilės spektro skirtingi variantai yra trečios eilės

spektras (bispektras), kuris pagal apibrėžimą yra trečios eilės statistikos Furjė transformacija, ir ketvirtos eilės spektras (trispektras), kuris yra stacionaraus signalo ketvirtos eilės statistikos Furjė transformacija. Energijos spektras yra vienas iš elementų, kurie priklauso diskretaus laiko signalo aukštesnės eilės spektrų klasifikacijai (4 pav.).



4 pav. Diskretaus signalo X(k) aukštesnės eilės spektrų klasifikacija. F[] – n-matmens Furjė transformacija

Signalo aukštesnės eilės statistikos ir spektras gali būti apibrėžiami momentais ir kumulantais. Momentai ir momentų spektras gali būti labai naudingi analizuojant deterministinį signalą, o kumulantai ir kumulantų spektras yra labai svarbūs analizuojant stochastinius signalus [4] (5 pav.).



5 pav. Polispektro klasifikacija

Signalų apdorojime aukštesnės eilės spektras naudojamas dėl to, kad:

1. Silpnina pridėtinį spalvotą gausinį triukšmą, kurio energijos spektras nežinomas, bispektras taip pat silpnina negausinį triukšmą su simetrine tikimybės galimumo funkcija (pdf – probability density function).
2. Atpažįsta neminimalios fazės signalus.
3. Išskiria informaciją susijusią su nukrypimais nuo gausinio signalo.
4. Aptinka ir charakterizuoja signalų netiesinės ypatybes, taip pat atpažįsta netiesines sistemas.

3.1.1 Apibrėžimai ir ypatybės

Jei X(k), k = 0, ±1, ±2, ±3,… yra stacionarus diskretaus laiko signalas ir jo momentai pagal eiliškumą n egzistuoja, tai

$$m_n^x(\tau_1,\tau_2,...,\tau_{n-1}) = E\{X(k)X(k+\tau_1)...X(k+\tau_{n-1})\} \qquad (3)$$

aprašo stacionaraus signalo n-tosios eilės momentų funkciją, kuri priklauso tik nuo laiko skirtumo $\tau_1,\tau_2,...,\tau_{n-1}$, $\tau_i = 0, \pm1, ...$ visiems *i*. Taip pat antros eilės momento funkcija $m_2^x(\tau_1)$ yra X(k) autokoreliacija, o $m_3^x(\tau_1,\tau_2)$ ir $m_4^x(\tau_1,\tau_2,\tau_3)$ yra atitinkamai trečios ir ketvirtos eilės momentai.

Negausinio stacionaraus atsitiktinio X(k) signalo n-osios eilės kumulanto funkciją galima užrašyti taip (tik kai n = 3,4):

$$c_n^x(\tau_1,\tau_2,...,\tau_{n-1}) = m_n^x(\tau_1,\tau_2,...,\tau_{n-1}) - m_n^G(\tau_1,\tau_2,...,\tau_{n-1}) \quad (4)$$

, kur $m_n^x(\tau_1,\tau_2,...,\tau_{n-1})$ – X(k) n-osios eilės momento funkcija,

$m_n^G(\tau_1,\tau_2,...,\tau_{n-1})$ – ekvivalentaus gausinio signalo n-osios eilės momento funkcija, kuri turi tokius pačius vidurkį ir autokoreliacijos eiliškumą kaip X(k).

Gausiniam signalui

$$m_n^x(\tau_1,\tau_2,...,\tau_{n-1}) = m_n^G(\tau_1,\tau_2,...,\tau_{n-1}) \quad (5)$$

Tai $c_n^x(\tau_1,\tau_2,...,\tau_{n-1}) = 0$. Nors išraiška (5) yra teisinga tik kai n = 3 ar 4, $c_n^x(\tau_1,\tau_2,...,\tau_{n-1}) = 0$ su visais n, jei X(k) yra gausinis. Santykis tarp X(k) momento ir kumulanto sekų egzistuoja kai n = 1, 2, 3, 4.

Pirmos eilės kumulantai:

$$c_1^x = m_1^x = E\{X(k)\} \quad \text{(vidurkis)} \quad (6)$$

Antros eilės kumulantai:

$$c_2^x(\tau_1) = m_2^x(\tau_1) - (m_1^x)^2 \quad \text{(kovariacijos seka)}$$
$$= m_2^x(\tau_1) - (m_1^x)^2 = c_2^x(-\tau_1) \quad (7)$$

, kur $m_2^x(-\tau_1)$ – autokoreliacijos seka.

Taigi antros eilės kumulanto seka yra kovariacija, o antros eilės momento seka yra autokoreliacija.

Trečios eilės kumulantai:

$$c_3^x(\tau_1,\tau_2) = m_3^x(\tau_1,\tau_2) - m_1^x[m_2^x(\tau_1) + m_2^x(\tau_2) + m_2^x(\tau_1-\tau_2)] + 2(m_1^x)^3 \quad (8)$$

, kur $m_3^x(\tau_1,\tau_2)$ - trečios eilės momento seka.

Ketvirtos eilės kumulantai:

$$\begin{aligned} c_4^x(\tau_1,\tau_2) &= m_4^x(\tau_1,\tau_2,\tau_3) - m_2^x(\tau_1)\cdot m_2^x(\tau_3-\tau_2)\cdot m_2^x(\tau_2)\cdot m_2^x(\tau_3-\tau_1)\cdot m_2^x(\tau_3) \\ & m_2^x(\tau_2-\tau_1)\cdot m_1^x[m_3^x(\tau_2-\tau_1,\tau_3,\tau_1) + m_3^x(\tau_2,\tau_3)m_3^x(\tau_2,\tau_4) + m_3^x(\tau_1,\tau_2)](m_2^x)^2 \\ & [m_1^x(\tau_1) + m_2^x(\tau_2) + m_2^x(\tau_3) + m_2^x(\tau_3-\tau_1) + m_2^x(\tau_3-\tau_2) + m_2^x(\tau_2-\tau_1)] - 6(m_1^x)^4 \end{aligned} \quad (9)$$

Jei signalo X(k) vidurkis nulinis $m_1^x = 0$, ir tada iš išraiškų (7), (8) seka, kad antros ir trečios eilės kumulantai yra identiški atitinkamai antros ir trečios eilės momentams. Bet tam, kad būtų galima apskaičiuoti ketvirtos eilės kumulantus, reikia (9) išraiškoje įvertinti ketvirtos ir antros eilės momentus.

$$c_4^x(\tau_1,\tau_2) = m_4^x(\tau_1,\tau_2,\tau_3) - m_2^x(\tau_1)\cdot m_2^x(\tau_3-\tau_2)\cdot m_2^x(\tau_2)\cdot m_2^x(\tau_3-\tau_1) - m_2^x(\tau_3)\cdot m_2^x(\tau_2-\tau_1) \quad (10)$$

Į išraiškas (7), (8),(9) įrašius $\tau_1 = \tau_2 = \tau_3 = 0$ ir priėmus, kad $m_1^x = 0$, gaunama

$$\begin{aligned} \gamma_2^u &= E\{x^2(k)\} = c_2^x(0) && \text{(dispersija)} \\ \gamma_3^u &= E\{x^3(k)\} = c_3^x(0,0) && \text{(asimetrijos koeficientas)} \\ \gamma_4^u &= E\{x^4(k)\} - 3[\gamma_2^u]^2 = c_4^x(0,0,0) && \text{(eksceso koeficientas)} \end{aligned} \quad (11)$$

Normalizuotas eksceso koeficientas yra $\gamma_4^x / [\gamma_2^x]^2$. Dispersija, asimetrijos ir eksceso koeficientų įverčiai skaičiuojami iš kumulantų su nuliniu vėlinimu.

3.2 Triukšmo kadrų atpažinimas, naudojant aukštesnės eilės statistiką (HOS)

Gausinio triukšmo asimetrijos ir eksceso koeficientai yra lygūs nuliui tik statistinio vidurkio prasme. Paprastai yra naudojami baigtinio ilgio kadrai, todėl ar kadras yra triukšmas sprendžiama tikėtinumo metodu, remiantis asimetrijos ir eksceso koeficientų įverčių dispersija ir pasiskirstymu. Duotajam gausiniam procesui *g(n)*, antros, trečios ir ketvirtos eilės momentų įverčiai yra

$$M_{kg} = \frac{1}{N}\sum_{n=0}^{N-1}[g(n)]^k \quad (12)$$

Lygybė apskaičiuoja $E[\{x(n)\}]^k$ įvertį, kai $k = 2, 3, 4$, o N yra kadrų skaičius. Šie įverčiai yra objektyvūs [9]. Balto gausinio triukšmo atveju, jų vidurkis ir dispersija gali būti išreikšti

$$\begin{aligned} &E[M_{3g}] = 0; \\ &E[M_{4g}] = 3v_g^2; \\ &Var[M_{3g}] = \frac{15v_g^3}{N}; \\ &Var[M_{4g}] = \frac{96v_g^4}{N} \end{aligned} \quad (13)$$

Asimetrijos koeficiento įvertis $SK = M_{3g}$ yra objektyvus, su nuliniu vidurkiu ir žinoma dispersija. Šis įvertis – tai daugybės nepriklausomų identiškai pasiskirsčiusių (iid – independent identically distributed) atsitiktinių kintamųjų suma. Tokiu atveju panaudojus centrinės ribos teoremą (central limit theorem), gaunamas normalizuotas įvertis

$$SK_a = \frac{M_{3g}}{\sqrt{15v_g^3 / N}} \tag{14}$$

yra gausinis kintamasis su nuliniu vidurkiu ir blokine dispersija. Taip apskaičiavus kadro asimetrijos koeficientą ir atitinkamai sužymėjus reikšmes pažymėtas „*a*“, tikimybė, kad kadras yra gausinis triukšmas, yra

$$prob[Noise] = prob[|\, SK_a \,| \geq a] \tag{15}$$

Ji ekvivalenti apskaičiuotam plotui, kurį sudaro sritis esanti po Gausine kreive. Ši sritis gali būti apskaičiuota kaip *erfc(x)* funkcija[1]. Kai $a = 0$ tai plotas yra vienetinis, o kai $a > 0$ - $prob[Noise] = 2/\sqrt{2\pi}\int_a^{\infty} e^{x^2/2}dx$. Taigi *prob[Noise] = erfc(|a|)*.

Neigiamas asimetrijos koeficientas nenurodo triukšmo buvimo tol, kol kalbos signalo HOS yra teigiama, kadangi tarpiniai segmentai gali turėti neigiamą HOS. Taip pat ir eksceso koeficientas pirmiausiai yra apskaičiuojamas iš antros ir ketvirtos eilės momentų. Tam, kad būtų galima užtikrinti skaičiavimo objektyvumą, yra naudojamas modifikuotas skaičiavimas

$$KU_U = (1 + \frac{2}{N})M_{4g} - 3(M_{2g})^2 \tag{16}$$

Šis apskaičiavimas yra objektyvus su nuliniu vidurkiu ir žinoma dispersija. Pasiskirstymas susideda iš dviejų kintamųjų skirtumo: gausinio ir chi kvadrato (chi-square). Vis dėl to čia yra naudojama aproksimacija ir įvertinimas tada laikomas normaliai pasiskirsčiusiu.

Kintamojo su nuliniu vidurkiu bloko dispersija yra aprašoma kaip

$$KU_{Ub} = \frac{KU_U}{\sqrt{\frac{3v_g^4}{N}(104 + \frac{452}{N} + \frac{596}{N^2})}} \tag{17}$$

Taip apskaičiavus duotojo kadro eksceso koeficientą ir atitinkamai sužymėjus reikšmes pažymėtas „*b*“, tikimybė, kad kadras yra triukšmas, yra: *prob[Noise] = erfc(|b|)*. Naudojant normalizuotus asimetrijos ir eksceso koeficientus, ir „erfc“ funkcijos reikšmes, tikimybė, kad kadras yra triukšmas, gali būti nustatyta.

[1] Klaidos funkcija (error function)

3.2.1 Balsui būtinos sąlygos

Balso signalo asimetrijos (skewness) ir eksceso (kurtosis) koeficientai yra apibrėžiami energija ir harmonikų skaičiumi, bei gali būti naudojami balso kadrams atpažinti [9]. Norint pašalinti energijos efektą, reiktų normalizuoti γ_3 ir γ_4 parametrus, tačiau tokiu atveju šie parametrai taptų mažiau efektyvesni, atpažįstant triukšmą. Todėl santykis atitinkamai asimetrijos koeficiento energijos su eksceso koeficiento yra naudojamas pašalinti signalo energijos efektą, tuo pačiu išvengiant triukšmo efekto.

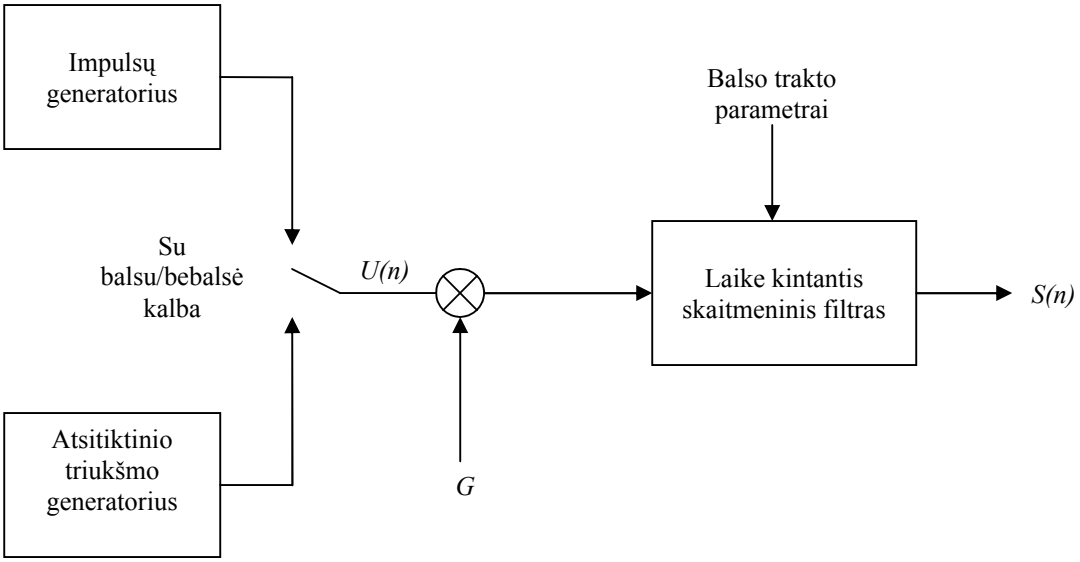
$$SKR = \frac{skewness^2}{kurtosis^{1.5}} = \frac{9(M-1)^2}{8M\left[\frac{4}{3}M - 4 + \frac{7}{6M}\right]^{1.5}} \tag{18}$$

, kur M – harmonikų skaičius (žingsnio funkcija – function of pitch).

Santykis SKR nepriklauso nuo signalo ir yra tik M funkcija. Esant gausiniam triukšmui, šis santykis yra neapibrėžtas, kadangi abu operandai yra nuliniai. Tačiau ši nulio sąlyga niekada neatsiranda dėl įverčių dispersijos. SKR santykis gali turėti bet kokią reikšmę, įskaitant ir balso signalo ribose, todėl jo neužtenka balso kadrams atpažinti.

3.3 Tiesinis prognozavimo kodavimas (LPC)

LPC (Linear Prediction Coding) yra laikoma viena iš galingiausių kalbos analizės metodikų. Jis yra kaip pagrindas kituose, naujesniuose ir sudėtingesniuose algoritmuose, kurie yra naudojami kalbos parametrams apskaičiuoti, pavyzdžiui, žingsniams, formantams, spektrui, balso traktui ir kalbos atvaizdavimas žemais bitais (low bit). Pagrindinis tiesinio prognozavimo principas teigia, kad kalba gali būti modeliuojama kaip tiesinės, laike kintančios sistemos išėjimas, kuri sužadinama arba periodinių impulsų, arba atsitiktinio triukšmo (6 pav.). Šie du akustiniai šaltiniai yra vadinami atitinkamai su balsu ir be balso. Pagal tai, balso sklidimas yra generuojamas balso stygų, einant oro srautui, o bebalsis garsas yra generuojamas tuomet, kai balso stygos yra atsipalaidavę.



6 pav. Kalbos sintezė sugeneruota su LPC

Tarkime, kad lygtis

$$s(n) \approx a_1 s(n-1) + a_2 s(n-2) + ... + a_p s(n-p) \tag{19}$$

, kur $a_1, a_2, ..., a_p$ – pastovūs koeficientai.

(19) lygybę galima pertvarkyti, įtraukiant sužadinimo narį *Gu(n)*:

$$s(n) = \sum_{i=1}^{p} a_i s(n-i) + Gu(n) \tag{20}$$

, kur G – stiprinimas,

$u(n)$ – normalizuotas sužadinimas.

Pertvarkant (20) lygybę į z sritį, gaunama:

$$S(z) = \sum_{i=1}^{p} a_i z^{-i} S(z) + GU(z) \tag{21}$$

Tokiu atveju persiuntimo funkcija bus:

$$H(z) = \frac{S(z)}{GU(z)} = \frac{1}{1 - \sum_{i=1}^{p} a_i z^{-1}} = \frac{1}{A(z)} \tag{22}$$

Tai atitinka skaitmeninio laike kintančio filtro persiuntimo funkcijai. Pagrindiniai parametrai gaunami su LPC modeliu yra: su balsu/be balso klasifikacija, žingsnio periodas, sustiprinimas ir koeficientai $a_1, ..., a_p$. Svarbu tai, kad kuo modelis aukštesnės eilės, tuo geriau

„all-pole“ modelis atvaizduoja kalbos garsą. Tiesinį prognozavimą su koeficientais a_k apibūdina daugianaris *P(z)*:

$$P(z) = \sum_{k=1}^{p} a_k z^{-k} \tag{23}$$

, kurio išėjimai yra

$$\tilde{s}(n) = \sum_{k=1}^{p} a_k s(n-k) \tag{24}$$

Prognozavimo klaidingumas *e(n)* aprašomas kaip:

$$e(n) = s(n) - \tilde{s}(n) = s(n) - \sum_{k=1}^{p} a_k s(n-k) \tag{25}$$

Tai yra sistemos $A(z) = 1 - \sum_{k=1}^{p} a_k z^{-k}$ išėjimai, ir jei a_k = a_k, tada $H(z) = \frac{G}{A(z)}$.

Pagrindinis tikslas yra gauti koeficientus a_k tam, kad būtų galima sumažinti prognozavimo klaidingumo kvadratą trumpuose kalbos kadrų segmentuose (paprastai 10-30ms). Trumpalaikis prognozavimo klaidingumas kadrui aprašomas:

$$E_n = \sum_m e_n^2(m) = [s_n(m) - \sum_{k=1}^{p} a_k s_n(m-k)]^2 \tag{26}$$

, kur $s_n(m)$ – kalbos segmentas, paimtas iš kaimyninio mėginio n: $s_n(m) = s(m+n)$.

Koeficientų a_k reikšmės, minimizuojančios prognozavimo klaidingumą E_n, gaunamos atsižvelgiant į $\frac{dE_n}{da_i} = 0$, *i = 1, 2, ..., p*, tada:

$$\sum_m s_n(m-i)s_n(m) = \sum_{k=1}^{p} a_k^l \sum_m s_n(m-i)s_n(m-k) \text{ , } 1 \le i \le p \tag{27}$$

, kur a_k^l – a_k reikšmės, kurios minimizuoja E_n.

Pažymint $\phi_n(i,k) = \sum_m s_n(m-i)s_n(m-k)$, (27) lygybę galima užrašyti

$$\sum_{k=1}^{p} a_k \phi_n(i,k) = \phi_n(i,0) \text{ , } \qquad i = 1, 2, ...,p \tag{28}$$

Tai yra *p* lygčių sistema su *p* kintamaisiais, kuri sprendžiama segmentui s_m ieškant a_k koeficientų. Taip galima įrodyti, kad:

$$E_n = \sum_m s_n^2(m) - \sum_{k=1}^{p} a_k \sum_{k=1}^{p} s_n(m-k) \tag{29}$$

O supaprastinus:

$$E_n = \phi_n(0,0) - \sum_{k=1}^{p} a_k \phi_n(0,k) \tag{30}$$

Tada apskaičiavus $E_n(i, k)$ reikšmes, kai $1 \le i \le p$, $1 \le k \le p$, a_k koeficientai gaunami sprendžiant (28) lygybę. (30) lygčių sistemą galima spręsti keletu metodų:

- Autokoreliacijos metodas.
- Kovariacijos metodas.

3.3.1 „All Pole“ modelis

Tiesinis prognozavimas ir autoregresinis modeliavimas yra dvi skirtingos problemos, kurios duoda tokius pačius skaitmeninius rezultatus. Abiem atvejais galutinis tikslas yra tiesinio filtro parametrų apibrėžimas. Tačiau kiekvienos problemos atveju naudojamas filtras yra skirtingas.

Tiesinio prognozavimo atveju, tikslas yra apibrėžti FIR filtrą, kuris gali optimaliai prognozuoti autoregresinio proceso būsimus mėginius, paremtus buvusių mėginių tiesiniu deriniu. Skirtumas tarp tikrojo autoregresinio signalo ir prognozuoto signalo yra vadinamas prognozavimo klaidingumu. Idealiu atveju šis klaidingumas yra baltas triukšmas.

Autoregresinio modeliavimo atveju, tikslas yra apibrėžti „all-pole“ IIR filtrą, kuris, sužadintas balto triukšmo, pateikia signalą su tokiomis pačiomis statistikomis kaip ir autoregresinis procesas.

Nagrinėjant lygtį:

$$s(n) = -\sum_{k=1}^{p} a_k s(n-k) + G\sum_{l=0}^{q} b_l u(n-l) \qquad 1 \le k \le p,\ 1 \le l \le p \tag{31}$$

, jei $b_l = 0$, tai modelis vadinamas kaip „all pole“ modelis arba autoregresinis modelis (AR – autoregressive model) (Jei $a_k = 0$, tai modelis tampa „all zero“ modeliu). Tokiame modelyje signalas $s[n]$ gali būti gaunamas kaip ankstesnių reikšmių ir įėjimo $u[n]$ derinys:

$$s(n) = -\sum_{k=1}^{p} a_k s(n-k) + Gu(n) \tag{32}$$

, kur G – stiprinimo rodiklis.

Taip pat galima sumažinti (30) išraiškoje perdavimo funkciją $H(z)$ į „all pole“ modelio perdavimo funkciją:

$$H(z) = \frac{S(z)}{U(z)} = \frac{G}{1 + \sum_{k=1}^{p} a_k z^{-k}} = \frac{G}{A(z)} \tag{33}$$

3.4 Kalbos atpažinimo algoritmas, paremtas aukštesnės eilės statistika (HOS-VAD)

Signalas, ilgą laiką esantis be balso, turi charakteristikas panašias į gausines, jo negalima atskirti nuo gausinio triukšmo naudojant HOS [9]. Tačiau realybėje bebalsis signalas atsiranda kalbos pereinamosiose ribose, turėdamas nenulinę HOS. Todėl VAD siūlomas remiantis HOS ir formuojamas kaip dviejų būsenų baigtinis automatas. Algoritmas apjungia asimetrijos, eksceso koeficientus, jų normalizuotas versijas γ_3 ir γ_4, SNR, LPC prognozavimo klaidingumą, ir SKR santykį kalbos kadrų atskyrimui nuo triukšmo.

Algoritmo etapai:

1. *Duomenys:*

   Naudojant 8kHz kalbos mėginius, dešimtos eilės LPC analizė yra atliekama kartą per 20ms, tokiu būdu generuojamas 20ms liekamasis signalas. Naudojant liekamąjį signalą, VAD yra vykdomas kad 10ms ir yra 20% persidengimas.

2. *HOS apskaičiavimas*:

   Naudojant (12) formulę, kiekvienai iteracijai (10ms) yra skaičiuojami antros, trečios ir ketvirtos eilės momentų įverčiai, kai *N = 100*. Autoregresinė schema yra naudojama momentų įverčių lyginimui. Iš jų yra nustatomi objektyvūs eksceso koeficiento įverčiai (16). Trečios eilės momentas yra asimetrijos koeficiento įvertis (12). Po to šie įverčiai yra normalizuojami pagal signalo energiją

$$\gamma_3 = \frac{SK}{M_{2x}^{1.5}}$$

$$\gamma_4 = \frac{KU_u}{M_{2x}^{2}} \tag{34}$$

3. *Triukšmo ir signalo-triukšmo santykio (SNR) skaičiavimas*:

Naudojant bebalsius kadrus apskaičiuojama triukšmo energija. Be to, yra priimta, kad pirmi trys kadrai yra be balso ir naudojami pradiniai triukšmo energijai apskaičiuoti. Kiekvieną kartą kai nustatoma, kad kadras yra bebalsis, jo energija yra naudojama pagal autoregresinį vidurkį atnaujinti triukšmo energiją

$$v_g(k) = (1-\beta)v_g(k-1) + \beta M_{2X} \tag{35}$$

, kur k – iteracijos indeksas,

M_{2X} – kadro energija,

v_g – triukšmo energija,

β – 0.1*prob[Noise].

Per kiekvieną iteraciją tuo momentu esama triukšmo energija yra naudojama skaičiuojant kadro SNR.

$$SNR = Pos\left[\frac{M_{2X}}{v_g} - 1\right] \tag{36}$$

, kur $Pos[x] = x$, kai $x > 0$ ir 0 , kai $x < 0$.

M_{2X} – kalbos su triukšmu energija

v_g – triukšmo energija.

Kadangi liekamasis signalas yra filtruotas žemų dažnių filtru ties 2kHz, tai anksčiau aprašytas SNR yra taikomas tik mažesniam spektrui. Bendras SNR apskaičiuojamas naudojant nefiltruotą liekamąjį signalą ir visos juostos energiją.

4. *Tikimybė, kad kadras yra vien tik triukšmas*:

   Kai asimetrijos ir eksceso koeficientai yra suskaičiuoti ir dispersija šių įverčių apskaičiuota, naudojant triukšmo energiją v_g, pagal (14) ir (17) formules. Tada, norint gauti nulinį vidurkį, atitinkamų vienetų SK_a ir KU_{Ua} dispersijos yra skaičiuojamos. Iš šių dviejų reikšmių yra nustatoma tikimybė, ar kadras yra triukšmas

$$prob[Noise] = [erfc(a) + erfc(b)]/2 \tag{37}$$

   , kur a ir b – atitinkamai SK_a ir KU_{Ub} apskaičiuotos reikšmės.

5. *Asimetrijos-eksceso koeficientų santykio (SKR) skaičiavimas*:

   SKR santykis apskaičiuojamas tiesiogiai iš nenormalizuotų asimetrijos ir eksceso koeficientų

$$SKR = \frac{[SK]^2}{[KU_U]^{1.5}} \tag{38}$$

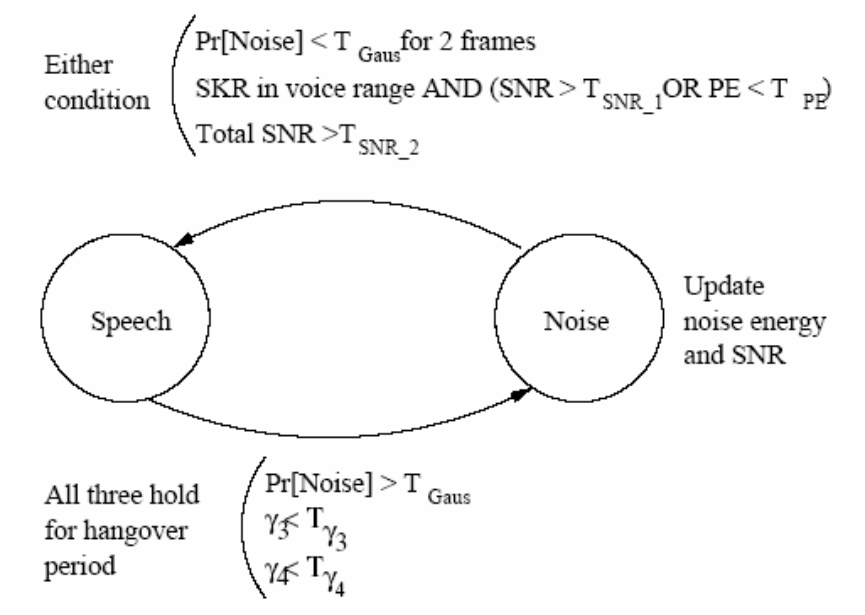
6. *LPC prognozavimo klaidingumas*:

   LPC prognozavimo klaidingumas yra prognozavimo didėjimo inversija ir skaičiuojamas iš atspindžio koeficientų (r_i) aibės, gautos iš LPC analizės

$$PE = \Pi_{i=0}^{10}(1 - r_i^2) \tag{39}$$

7. *Balso/triukšmo būsenų automatas*:

   VAD algoritmas vykdomas kaip dviejų būsenų automatas (7 pav.).



7 pav. VAD, paremto HOS, būsenų automatas

Būsenose vykdomos operacijos:

a) Triukšmo būsena:

   Triukšmo energija atnaujinama pagal *prob[Noise]* (37). SKR, gausinio signalo galimumo (triukšmo tikimybė), SNR ir klaidingumo tikimybės reikšmės naudojamos sprendžiant ar kadras yra kalba. Sekančių trijų sąlygų atsiradimas lemia perėjimą:

   1. *prob[Noise]* < T_{Gaus} dviems iš eilės einantiems kadrams.
   2. SKR kalbos ribose ir $SNR > T_{SNR1}$ ar $PE < T_{PE}$ reiškia kalbos kadrą.
   3. Bendras $SNR > T_{SNR2}$ reiškia, kad kadre daug kalbos.

b) Kalbos būsena:

Triukšmo tikimybė kartu su γ_3 ir γ_4 reikšmėmis yra naudojamos sprendžiant ar kadras yra gausinis signalas. Po pauzės periodo vyksta perėjimas į triukšmo būseną, jei $prob[Noise] > T_{Gaus}$ ir $\gamma_3 < T_{\gamma 3}$ ir $\gamma_4 < T_{\gamma 4}$.

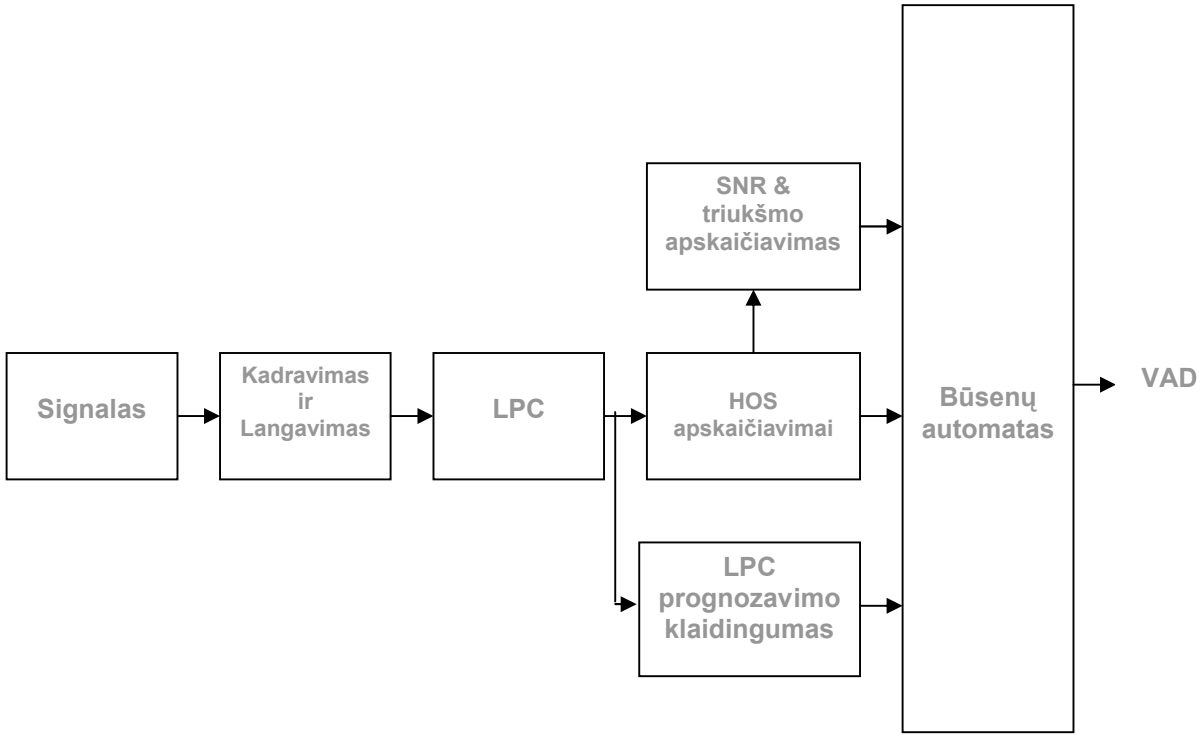
Pagrindiniai HOS-VAD algoritmo etapai surašyti 1 lentelėje, o 8 paveiksle – sistemos modelis.

1 lentelė. HOS-VAD algoritmo etapai

Procesas	Įėjimas	Išvestis	Apibūdinimas
Buferizacija	Kalbos signalas	Atrankiniai kadrai	Kalbos signalas pateikiamas matricos forma, kur kadras yra eilutė.
LPC	Kadrai	Liekamasis signalas, LPC koeficientai	Skaičiuoja LPC koeficientus ir liekamąjį signalą.
HOS skaičiavimas	Liekamasis signalas	Normalizuoti skewness ir kurtosis	Skaičiuojami antros, trečios ir ketvirtos eilės momentai, pagal kuriuos po to skaičiuojami skewness ir kurtosis.
Triukšmo ir SNR skaičiavimas	Liekamasis signalas	Triukšmo energija ir SNR	Naudojant kadro energiją skaičiuojama triukšmo energija ir SNR.
Tikimybės, kad kadras vien tik triukšmas, skaičiavimas	Normalizuoti skewness ir kurtosis, atsižvelgiant į triukšmo energiją	Tikimybė, kad kadras yra triukšmas	Tikimybė apskaičiuojama naudojant normalizuotų skewness ir kurtosis klaidos funkciją.
SKR skaičiavimas	Normalizuoti skewness ir kurtosis	SKR	Skaičiuojamas skewness kurtosis santykis.
LPC prognozavimo klaidingumo skaičiavimas	Iš LPC analizės gauti atspindžio koeficientai	LPC prognozavimo klaidingumas	Skaičiuojamas tiesinio prognozavimo kodavimo klaidingumas
Triukšmo būsena	Triukšmo tikimybė, SKR, SNR ir klaidingumo tikimybė	Kadrai pripažinti kaip triukšmas	Priklausomai nuo rėžių daroma išvada, kad kadras yra triukšmas.
Kalbos būsena	Triukšmo tikimybė, normalizuoti skewness ir kurtosis	Kadrai pripažinti kaip kalbos	Priklausomai nuo rėžių daroma išvada, kad kadras yra kalba.

1 lentelė. HOS-VAD algoritmo etapai. Tęsinys

Tikimybių skaičiavimas	Teisingai atpažinti kalbos kadrai, teisingai atpažinti triukšmo kadrai, neteisingai suskirstyti triukšmo ir kalbos kadrai	Tikimybės teisingai atpažintų kalbos, triukšmo kadrų, tikimybė klaidingo atpažinimo	Skaičiuojamos teisingai atpažintų kalbos, triukšmo kadrų bei klaidingo atpažinimo tikimybės $P_{Cspeech}$, P_{Cnoise}, $P_{f.}$



8 pav. HOS-VAD sistemos modelis

Taigi signalas pirmiausiai yra skaidomas į kadrus ir langus, po to filtruojamas. Po tiesinio prognozavimo kodavimo yra skaičiuojama aukštesnės eilės statistikos, signalo-triukšmo santykis, bei LPC prognozavimo klaidingumas. Visi apskaičiuoti parametrai paduodami būsenų automatui, kuriame yra sprendžiama ar kadras yra kalba, ar triukšmas.

4 HOS-VAD Algoritmo realizavimas

Šiame skyriuje aprašomas HOS-VAD algoritmo realizavimas, kuris vykdomas trim etapais: algoritmo sukūrimu ir testavimu Matlab aplinkoje, perrašymu į C programavimo kalbą. C kodas pasitelkiant Code Composer Studio sąsają buvo įdiegtas TMS320C6713 DSK.

4.1 HOS-VAD algoritmo kūrimas Matlab aplinkoje

Modeliuojant procesus buvo priimta, kad programa įėjimui paima signalą, jį suskaido į 20ms kadrus ir rezultatą atiduoda modeliavimui. Modeliavimo metu algoritmas veikia realiu laiku ir nuosekliai, imdamas po vieną kadrą. Be to RVAD (Robust VAD) algoritmas, paremtas aukštesnės eilės statistika, ne tik dirba su esamu kadru, bet ir gauna kito kadro mėginius.

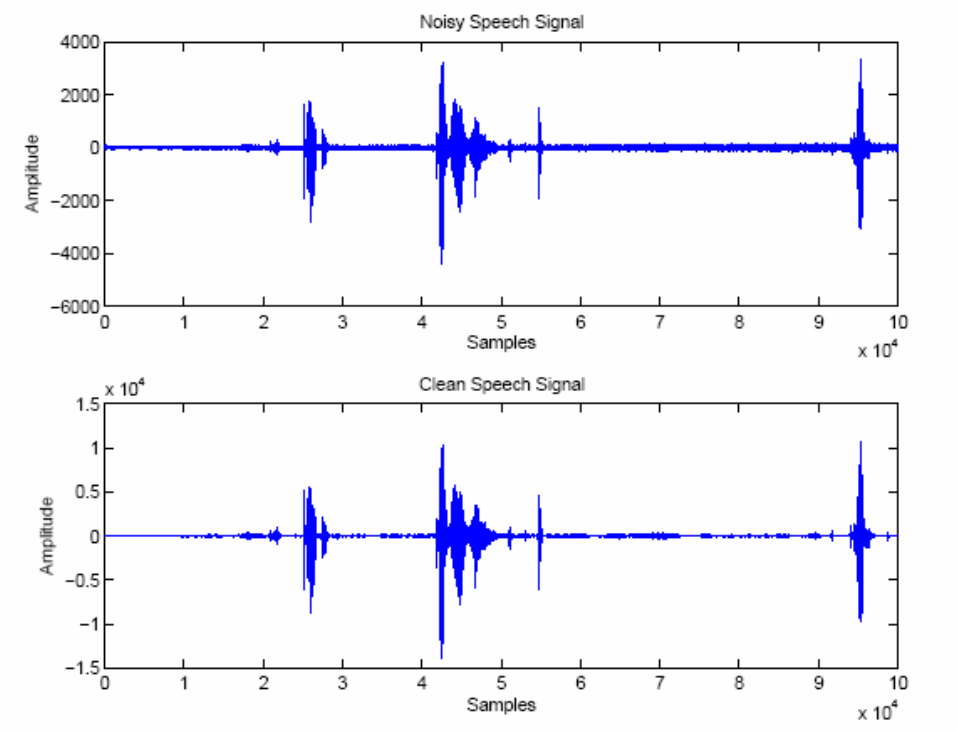
4.1.1 Įėjimo signalas

Sistema yra kuriama su skirtingais kalbos signalais:

- Triukšmu užterštas signalas, t.y. testavimo blokai.
- Švarus kalbos signalas.
- Žymės bylos (mark files) arba nuorodos signalas.

Kalbos signalai yra 10 skirtingų variantų, iš kurių penki įkalbėti moteriškų balsu ir penki – vyrišku balsu. Yra naudojami keturių rūšių triukšmo signalai: automobilio, garažo, traukinio ir gatvės. Kombinuojant kalbos ir triukšmo signalus įvairiu santykiu buvo gauta 80 skirtingų testavimo blokų. Kiekvienas blokas yra skirtinga kalbos normalizacijos lygio, triukšmo tipo ir SNR kombinacija. Šie blokai turi skirtingus SNR lygius: 6dB, 12dB, 18dB ir

∞. Pavyzdžiui, 6 blokas yra gautas sukombinavus *m1left1.nom* kalbos bylą ir *car.nom* triukšmo bylą, bei papildžius *6dB* SNR.



9 pav. Kalbos signalas su triukšmu ir be triukšmo

Šie testavimo blokai ir yra triukšmu užteršti kalbos signalai, naudojami kaip algoritmo įėjimo signalai. 9 paveiksle pavaizduotas kalbos signalas su triukšmu ir be triukšmo. Nuorodos signalas arba žymės bylos yra sugeneruoti rezultatų palyginimui.

4.1.2 Skaidymas į kadrus

Kadangi yra laikoma, kad kadro ilgis gali būti tarp 10ms – 30ms, darbe nustatytas kadro ilgis 20ms. Jei ilgis yra mažesnis nei 10ms, gaunamas grubumas, o jei ilgis – didesnis nei 30ms, sumažėja suvokimo kokybė.

4.1.3 Skaidymas į langus

Lango ilgis nustato kalbos signalo dalį, kuri bus pasirinkta. Idealaus lango dažnio charakteristika turi labai siaurą pagrindinę skiltį, kuri padidina gebą ir sumažina skilties šonus ar dažnio nutekėjimą. Kadangi idealus langas praktiškai neegzistuoja, tai kompromisų ieškoma atskirai kiekvienam taikymui.

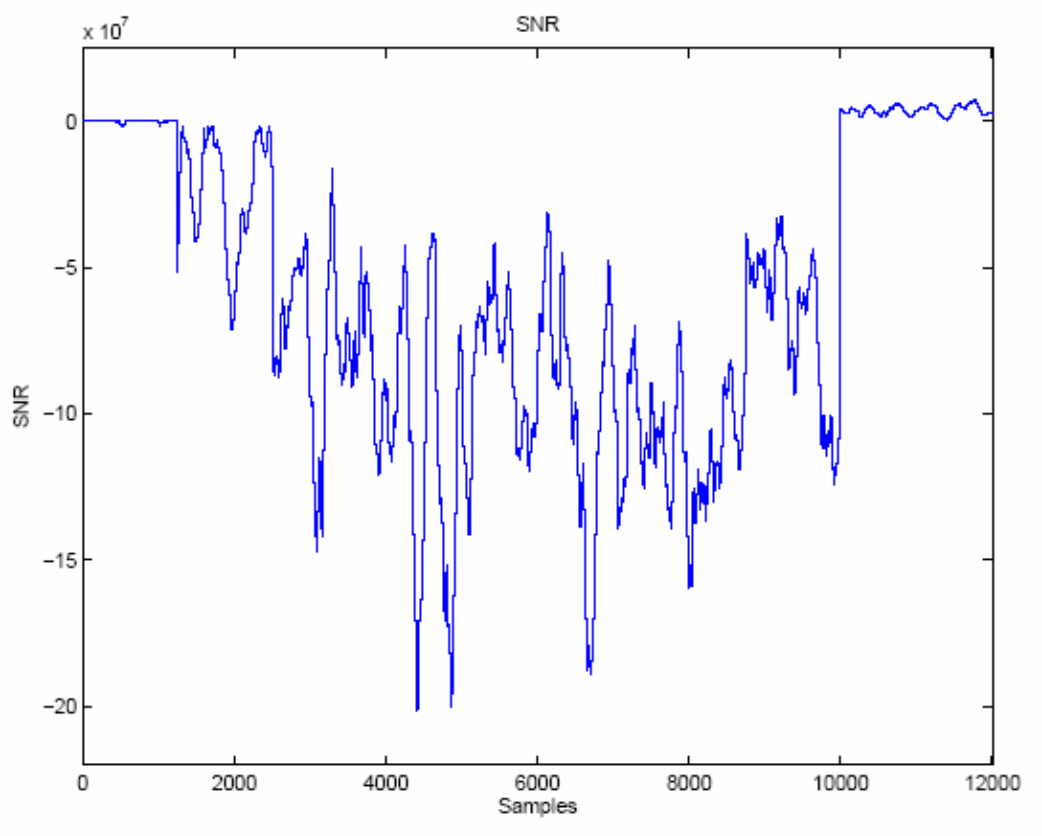
Yra galimi skirtingi langų tipai: stačiakampio, „hanning“ ar „hamming“. Stačiakampiai langai, dėl labai siauros pagrindinės skilties ir didelio dažnio nutekėjimo, turi didžiausią dažnio gebą. Didelių šonų skiltys duoda aukštų dažnių nutekėjimą, tokiu būdu stačiakampiais sulanguota kalba yra triukšmingesnė. Dėl to stačiakampiai langai nėra naudojami kalbos spektro analizei. Trapezoidiniai langai, tokie kaip „hamming“ ir „hanning“, yra žinomi kaip turintys mažesnį dažnio nutekėjimą, bet ir su mažesne geba. Todėl yra gaunamas glotnesnis spektras nei naudojant stačiakampius langus. Šiame darbe yra naudojami „hanning“ langai.

4.1.4 HOS parametrų skaičiavimas

Tam, kad atpažintų ar esamas kadras yra kalba, ar ne, kadrams yra skaičiuojami normalizuoti asimetrijos ir ekscesо koeficientai. Šios reikšmės yra gaunamos iš (34) išraiškų. Dėl šių priežasčių, naudojant formules (13) iš anksto yra apskaičiuojami antros, trečios ir ketvirtos eilės momentai.

4.1.5 SNR apskaičiavimas

Signalo-triukšmo santykis skaičiuojamas naudojant esamą triukšmo energiją (36). Jei pagal triukšmo energiją esamas kadras pripažįstamas kaip ne kalba, tai triukšmo energija yra skaičiuojama pagal (35) formulę. Kitu atveju, triukšmo energija išlieka tokia pati kaip ir praėjusio kadro.



10 pav. Signalo-triukšmo santykis SNR

Po to kai SNR apskaičiuojamas kiekvienam kadrui, yra atnaujinamas bendras SNR. 10 paveiksle pavaizduotas 67 bloko pirmų 12000 mėginių SNR.

4.1.6 Būsenų automatas

Po normalizuotų asimetrijos ir eksceso koeficientų, SNR, triukšmo tikimybės, LPC prognozavimo klaidingumo ir SKR apskaičiavimų, algoritmas sprendžia ar kadras yra kalba, ar ne. Sprendimas yra priimamas naudojant būsenų automato modelį, kuris turi dvi būsenas: triukšmo ir kalbos. Esama būsena priklauso nuo ankstesnio kadro.

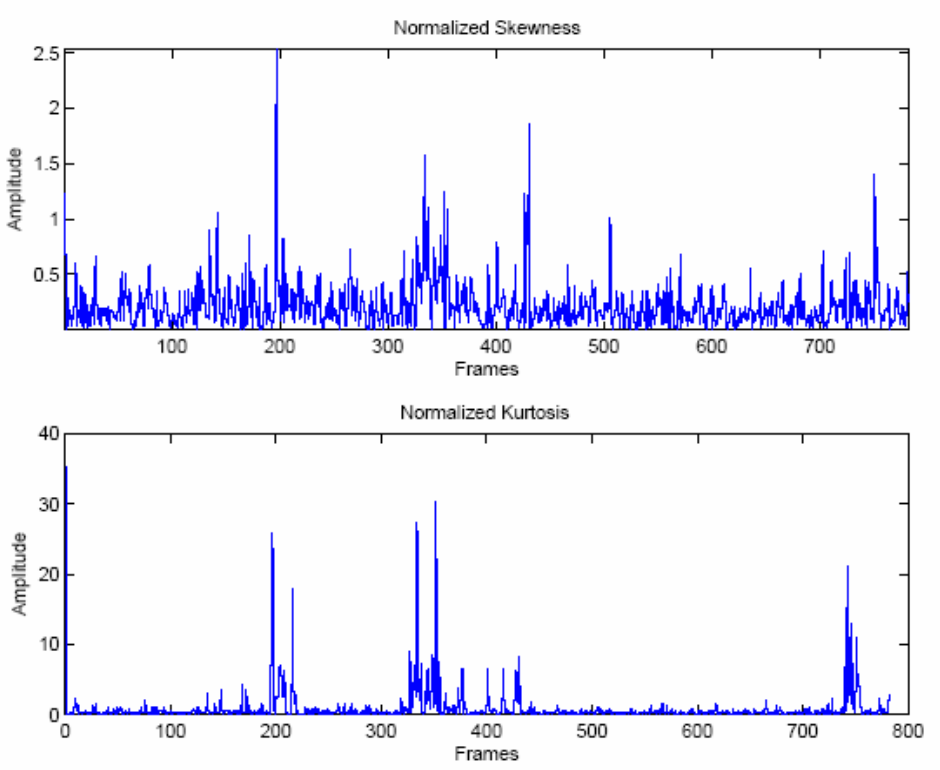
Jei būsenų automatas yra triukšmo būsenoje, tikrinama ar esamas kadras vis dar triukšmas, ar jau ne. Sprendimas priimamas pagal esamo kadro triukšmo tikimybės, prognozavimo klaidingumo, SKR, SNR ir bendrą SNR reikšmes, juos lyginant su nustatytais atitinkamais rėžiais.

Jei būsenų automatas yra kalbos būsenoje, sprendžiama ar esamas kadras vis dar kalba, ar ne. Sprendimas priimamas atsižvelgiant į esamo kadro triukšmo tikimybės, normalizuotų asimetrijos ir eksceso koeficientų reikšmes, juos lyginant su atitinkamais rėžiais.

4.2 Algoritmo modelio testavimo rezultatai

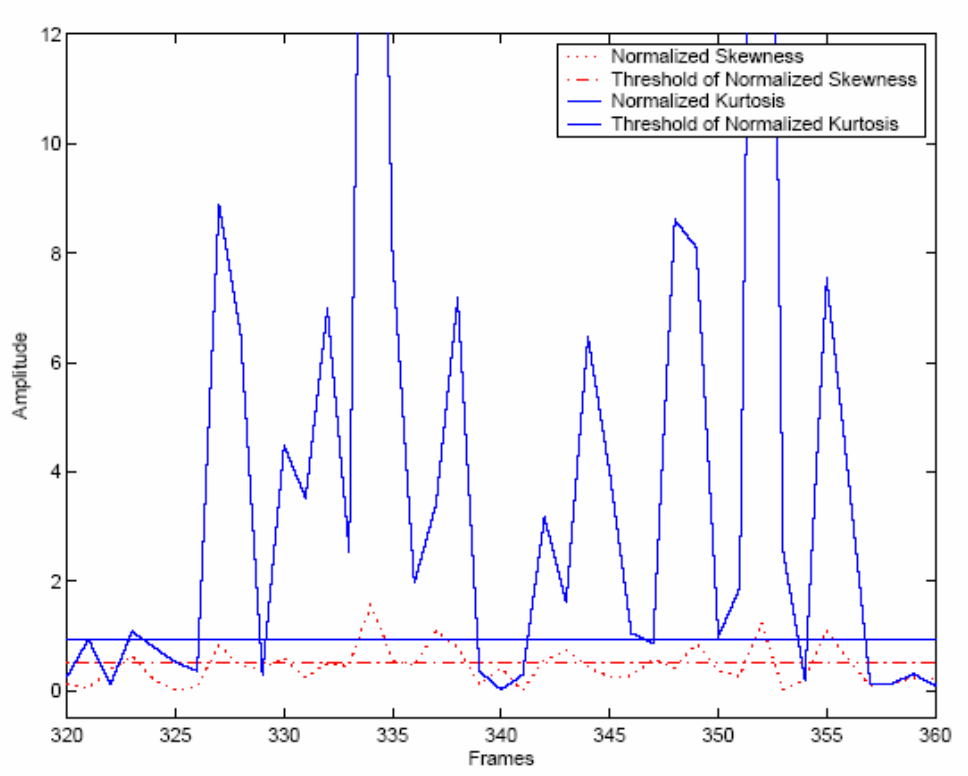
Algoritmo modelio testavimo rezultatams parodyti naudojamas 67 bloko 100000 mėginių įėjimo signalas. 67 blokas yra moters kalbos ir automobilinio triukšmo junginys (9 pav.)

Šiuo atveju, aukščiausios normalizuotų asimetrijos ir eksceso koeficientų viršūnės rodo kalbos kadrus (11 pav.). Apjungiant normalizuotų asimetrijos ir eksceso koeficientų rezultatas yra sprendžiama ar kadras yra kalba, ar triukšmas.



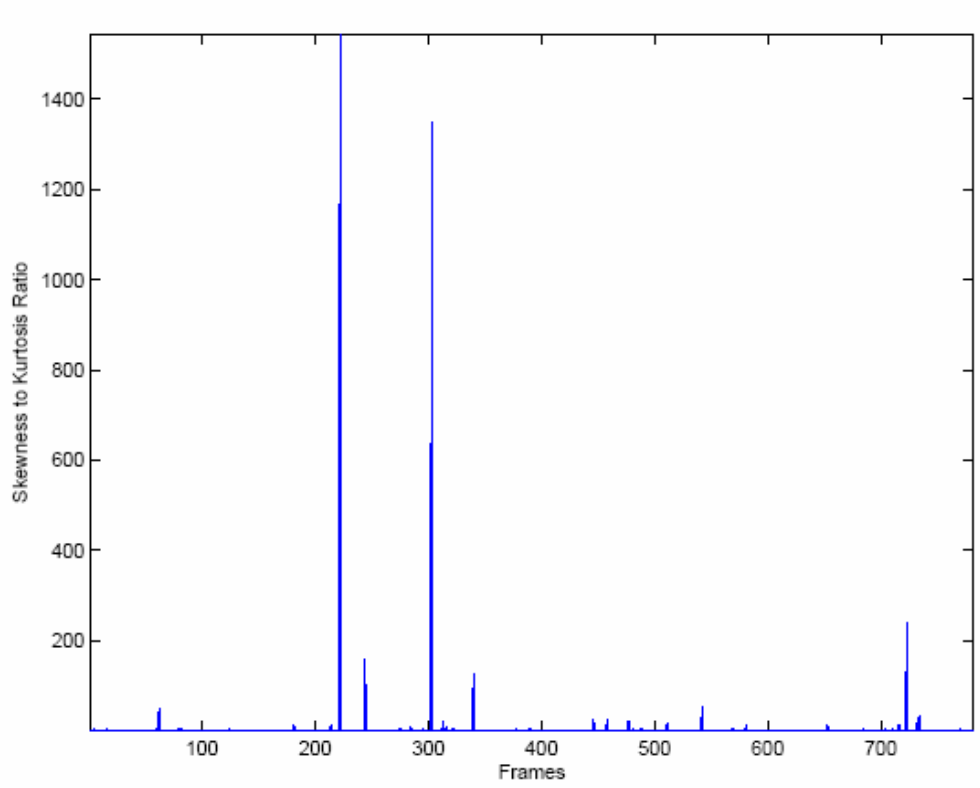
11 pav. Normalizuoti asimetrijos ir eksceso koeficientai

12 pav. yra pavaizduota 40 signalo kadrų. Kadrai, esantys žemiau 0,5 asimetrijos ir 0.94 eksceso koeficientų amplitudžių, yra triukšmas. Taigi kadrai nuo 339 iki 341 yra pripažįstami triukšmu, kadangi asimetrijos ir eksceso koeficientai yra žemiau rėžių, o kadrai nuo 354 iki 357 yra kalbos. Jei reikšmės yra žemiau rėžių, tai kadras neabejotinai yra pripažįstamas triukšmu, tačiau ar kadras yra kalba sprendžiama ne tik iš asimetrijos ir eksceso koeficientų, bet ir iš kitų parametrų.



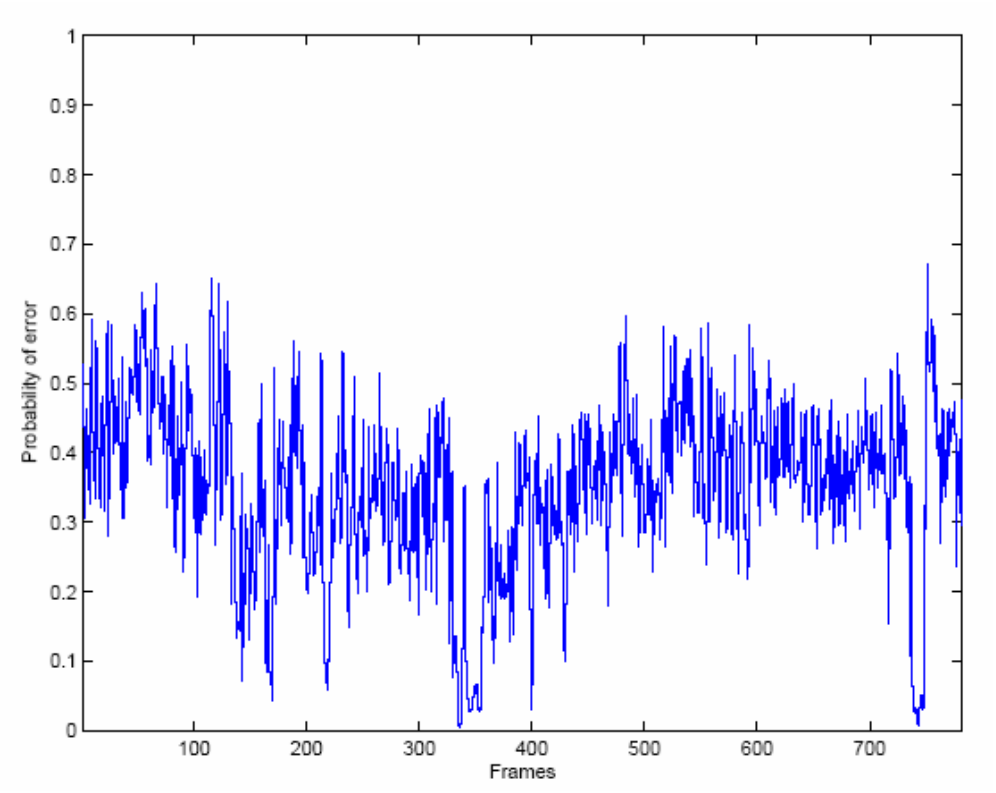
12 pav. Normalizuoti asimetrijos ir eksceso koeficientai ir jų rėžiai nuo 320 iki 360 kadro

Asimetrijos ir eksceso koeficientų santykis (SKR) (13 pav.) yra vienas iš parametrų naudojamų kalbos kadrams atpažinti. Priešingai nei normalizuotų asimetrijos ir eksceso koeficientų grafike, čia aukštos SKR viršūnės rodo negausinį triukšmą. Analizuojant grafiką, galima teigti, kad kalbos kadrai priklauso tam tikroms amplitudės reikšmių riboms.

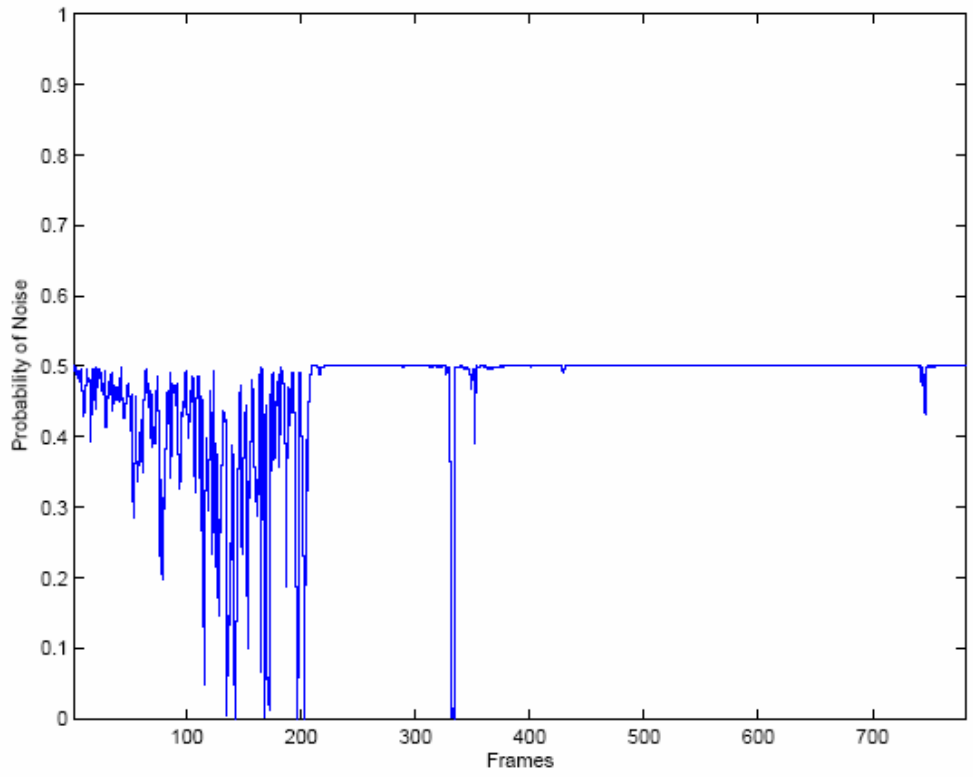


13 pav. Asimetrijos-ekscesо koeficientų santykis, SKR

Prognozavimo klaidingumo ir triukšmo tikimybės grafikai, pavaizduoti atitinkamai 14 ir 15 pav., rodo kadrų statistinę informaciją. Kuo aukštesnė prognozavimo klaidingumo reikšmė, tuo labiau tikėtina, kad kadras yra triukšmas. Kuo mažesnė triukšmo tikimybė, tuo didesnė galimybė, kad kadras yra kalba.

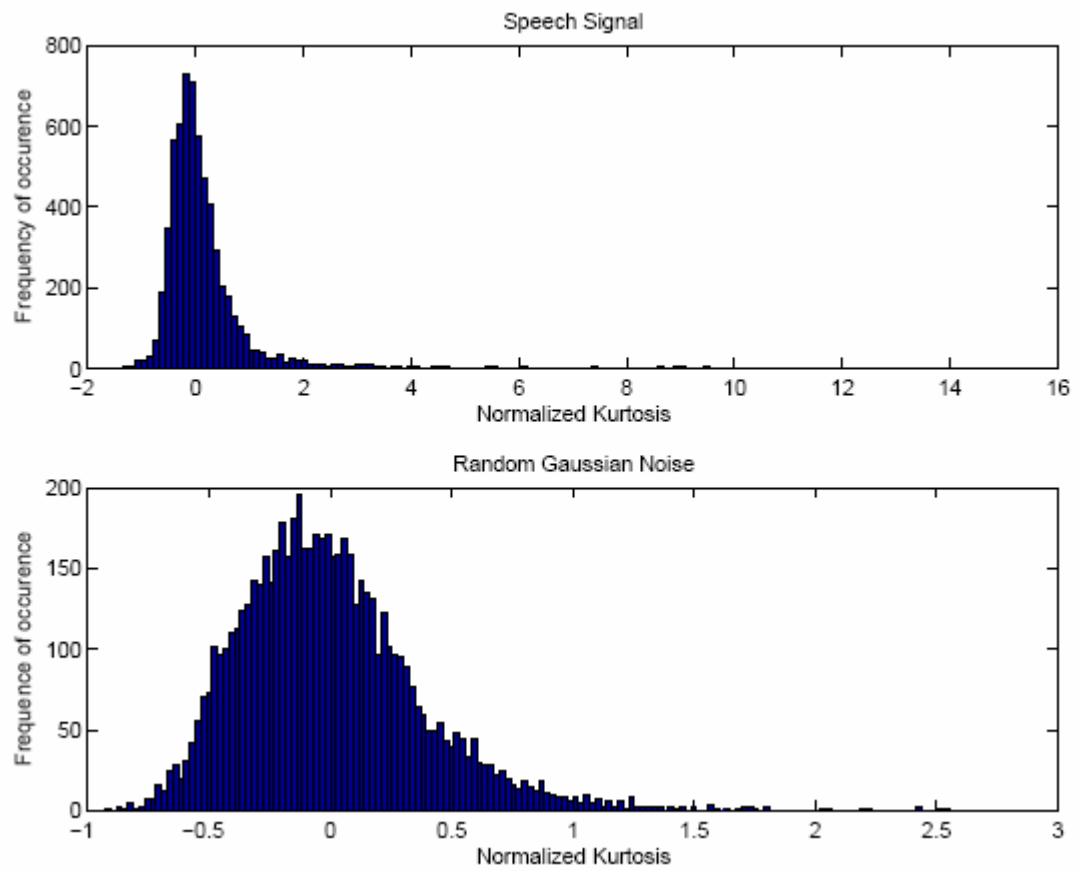


14 pav. Prognozavimo klaidingumas



15 pav. Triukšmo tikimybė

16 pav. pavaizduota normalizuoto eksceso koeficiento kadras po kadro (frame-by-frame) reikšmių histograma, sugeneruota iš LPC liekamojo signalo 6250 kadrų. Kita histograma yra gauta iš atsitiktinai sugeneruoto gausinio triukšmo normalizuoto eksceso koeficiento prieš LPC filtravimą. Histogramos rodo ketvirtos eilės statistikų skirtumą tarp kalbos ir gausinio triukšmo. Iš paveikslo galima matyti, kad kalbos pasisakymai turi tylos periodus, kai eksceso koeficientas yra nulinis.



16 pav. LPC liekamojo signalo normalizuotas eksceso koeficiento histogramos (kalbos signalas lyginamas su atsitiktiniu gausiniu triukšmu)

Remiantis visais anksčiau aprašytais parametrais, yra vertinamas algoritmo efektyvumas ir apskaičiuojami trys našumo parametrai:

$P_{cSpeech}$ – teisingai atpažintų kalbos kadrų tikimybė, kuri apskaičiuojama kaip teisingos kalbos atpažinimo ir bendro rankiniu būdu pažymėtų kalbos kadrų skaičiaus santykis.

P_{cNoise} – teisingai atpažintų triukšmo kadrų tikimybė, kuri apskaičiuojama kaip teisingo triukšmo atpažinimo ir bendro rankiniu būdu pažymėtų triukšmo kadrų skaičiaus.

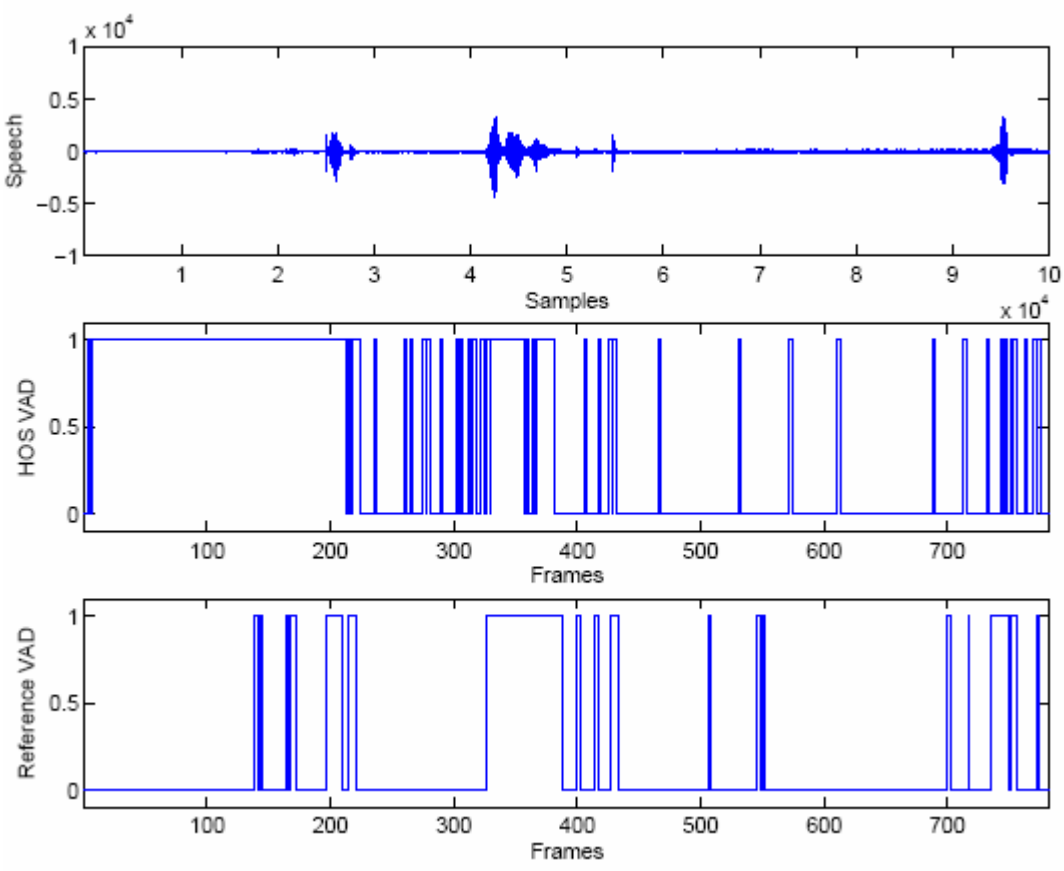
P_f – blogo atpažinimo tikimybė, kuri apskaičiuojama kaip neteisingai klasifikuotų kalbos ar triukšmo ir bendro kadrų skaičiaus santykis.

2 lentelėje yra automobilio triukšmo signalo įvertinimas kai SNR 0dB. Kai signale nėra kalbos, $P_{cSpeech}$ neskaičiuojamas.

2 lentelė. Automobilio triukšmas, SNR – 0 dB

Triukšmo aplinka	$P_{cSpeech}$ **(%)**	P_{cNoise} **(%)**	P_f **(%)**
Automobilio (SNR 0dB)	∞	91,2668	8,7332

Be to, kalbos signalui buvo sukurtos nuorodos bylos, su kuriomis lyginami algoritmo rezultatai (17 pav.). Pagal kadro energiją yra priimamas sprendimas, ar kadras yra kalba, ar ne. Nuorodos grafikuose vienetas yra kalba, o nulis – tyla.



17 pav. Palyginimas su nuorodos byla

Buvo naudojamos skirtingos triukšmo aplinkos, tokios kaip gatvės, garažo, automobilio, ir taip pat šie triukšmai su skirtingais SNR lygiais. Atitinkamai buvo suskaičiuoti

$P_{cSpeech}$, P_{cNoise}, P_f (3 lentelė). Kaip buvo minėta, $P_{cSpeech}$ ir P_f skaičiuojami remiantis nustatytais rėžiais, pagal kuriuos buvo sprendžiama ar kadras kalba, ar ne. Skirtingoms triukšmo aplinkoms ir net skirtingiems SNR lygiams yra skirtingi rėžiai. Vietoje adaptyvių rėžių buvo pasirinkti fiksuoti, kadangi su adaptyviais rėžiais gaunami rezultatai yra blogesni. Pagrindinis aspektas yra, kad kadrų atpažintų kaip kalba, tikimybė būtų kuo didesnė, kadangi svarbu, kad kuo mažiau kalbos būtų atpažinta kaip triukšmo.

3 lentelė $P_{cSpeech}$, P_{cNoise}, P_f

Triukšmo aplinka		$P_{cSpeech}$ **(%)**	P_{cNoise} **(%)**	P_f **(%)**
Tipas	SNR			
Gatvė	18 dB	95,02624	95,27732	19,94640
Gatvė	12 dB	94,81300	97,22586	24,24436
Gatvė	6 dB	96,27360	94,24034	29,57412
Garažas	18 dB	90,89854	96,50380	17,41288
Garažas	12 dB	91,22174	96,31108	21,28208
Garažas	6 dB	92,55860	98,21720	29,88722
Automobilis	18 dB	97,49856	93,31110	23,55938
Automobilis	12 dB	90,20506	96,81272	21,59166
Automobilis	6 dB	93,12092	94,75996	27,59700
Traukinys	18 dB	94,18450	97,72330	19,86844
Traukinys	12 dB	86,78462	98,22220	23,31566
Traukinys	6 dB	92,81294	94,00732	29,6381

4.3 Algoritmo realizavimas DSK plokštėje

Šiame poskyryje aprašomas algoritmo C kodas, jo realizacija TI C6713 DSK[2] plokštėje. „Code studio composer“ (CCS) aplinka yra papildyta keletu C bylų.

[2] DSK apibūdinimą žiūrėti 3 priede

4.3.1 Algoritmo programavimas C programavimo kalba

Kadangi C kodas konvertuotas tiesiogiai iš Matlab, nebuvo tinkamas diegimui į plokštę, algoritmas rankiniu būdu buvo perrašytas C programavimo kalba. Ši programa naudoja įėjimo bylą, kurioje 16 bitų duomenys, aukšto/žemo baito (high-byte/low-byte) žodžio formatas.

HOS VAD programą sudaro šios pagrindinės funkcijos:

- *HanningWindow*: funkcija generuoja nustatyto dydžio „Hanning“ langus. Skirtingai nuo Matlab, čia „Hanning“ langas yra be nulio užpildymo (zero padding).
- *SignalFraming*: funkcija formuoja nustatyto dydžio kadrą.
- *LPC*: tiesinio prognozavimo modelio funkcija naudoja dar *AutoCorrelation* ir *Levinson-Recursion* funkcijas. Ji generuoja tiesinio prognozavimo ir atspindžio koeficientus.
- *AllPoleFilter*: funkcija iš esamo kadro ir tiesinio prognozavimo koeficientų generuoja liekamąjį signalą (residue).
- *HOSCompute*: aukštesnės eilės statistikos funkcija.
- *GetVgSNR*: funkcija skaičiuoja triukšmo energiją ir signalo-triukšmo santykį.
- *GetSKR*: funkcija skaičiuoja asimetrijos ir eksceso koeficientų santykį.
- *PredictionError*: Naudojant atspindžio koeficientus funkcija skaičiuoja LPC prognozavimo klaidingumą.
- *StateMachine*: funkcija vykdo dviejų būsenų automatą, naudojamą sprendžiant ar kadras yra kalba, ar triukšmas.

Programos rezultatai yra išvedami komandiniame lange.

4.3.3 C kodo įdiegimas DSK plokštėje

Algoritmo testavimas siekia patvirtint C kodo operacijas DSK plokštėje. Įėjimo signalas (švari kalba sugadinta triukšmu) yra suskaidomas į kadrus. Būsenų automato išėjimas yra reikiami parametrai. DSK išėjimas (pateikiamas: išvedant kintamųjų reikšmes, žybčiojant lemputei ar garsu) turi patvirtinti išėjimą gautą Matlab aplinkoje.

Diegiant C kodą DSK plokštėje, buvo padaryta keletas kodo pakeitimų:

- HOS-VAD programa DSK plokštėje nenaudoja jokių įėjimo duomenų. Signalo 400 mėginių yra naudojami kaip programos globalus kintamasis.
- Sprendžiant klaidą buvo parašyta papildoma funkcija. C programa naudoja įdiegtą *erfc()* funkciją, o CCS šios funkcijos neturi. Dėl to HOS-VAD kodui DSK plokštėje buvo parašyta *erfc()* funkcija.

CCS HOS-VAD algoritmo programa sėkmingai buvo sukompiliuota ir įkelta į DSK plokštę testavimui. Po skaičiavimų rezultatai išvesti *CCS stdout window* lange, patvirtino, kad programa veikia teisingai.

Išvados

Šio darbo tikslas, išnaudojant aukštesnės eilės statistikos ypatybes, patobulinti balso signalo aptikimo ir triukšmo pašalinimo algoritmą. HOS-VAD algoritmas buvo sukurtas ir ištestuotas Aalborgo universitete, Danijoje. HOS atskleidžia svarbias kumulantų ypatybes, kurių reikšmingumas praplečia VAD taikymo ribas:

- Trečios eilės HOS gausiniam signalui yra nulis, tačiau signalo su balsu asimetrijos ir eksceso koeficientai yra nenuliniai. Taigi gali būti panaudoti kaip pagrindas atpažįstant kalbą ar balso klasifikacijai. Normalizuoti pagal atitinkamą signalo energiją, šie parametrai yra nepriklausomi nuo signalo lygių. Dėl to yra patogu juos naudoti atpažįstant kalbą, kai naudojami absoliutiniai rėžiai.
- Balso signalo atitinkamų asimetrijos ir eksceso koeficientų energijų santykis yra nepriklausomas nuo signalo energijos ir yra apribotas iki mažo diapazono bet kuriam praktiniam žingsnio diapazonui.
- Kalbos signalui be balso galima nemodeliuoti LPC liekamojo signalo kaip harmoninio proceso, bet geriau kaip bendrą baltą procesą.
- Lyginant su kitais darbais, kur HOS naudojama kalbai, šiame darbe yra paimtas labiau fundamentinis metodas, pagal kurį pirmiausiai daromi analitiniai išvedimai, pagrįsti kalbos modeliu. Tokiu būdu yra pagrindžiamas eksperimentinių rezultatų patvirtinimas arba atmetimas.
- Pagrindinė priežastis, kodėl yra atsižvelgiama į LPC liekamąjį signalą, yra tai, ar jis plokščio spektrinio voko. Kadangi tai kalbos aukštesnės eilės kumulantų išvedimus daro lengviau apdorojamus, bei leidžia išmatuoti ir įvertinti gausinio triukšmo HOS įverčių poslinkį ir dispersiją.
- VAD algoritmo vykdymui buvo parinkti skirtingi triukšmo scenarijai. Buvo naudojami neadaptyvūs slenksčiai, o kiekvienam triukšmo tipui ir SNR lygiui

buvo surandami fiksuoti optimalūs slenksčiai, taip padidinant rezultatų kokybiškumą.

- Pasiūlytas algoritmas paremtas daugiau analitine struktūra, todėl skaičiavimai, naudojantys panašių parametrų rinkinį, tampa sudėtingesni (jei skaičiavimai k-osios eilės, tai sudėtingumas $O(n^k)$) ir užtrunka ilgiau. Tačiau kaip tik ši struktūra net keletu procentu pagerina rezultatus. Nors ir sudėtingumas yra didelis, tačiau ir rezultatai yra daug geresni net kai SNR yra mažas.

Algoritmas buvo įdiegtas C6713 DSK plokštėje. Dirbant su 400 mėginių, būsenų automato rezultatai yra panašūs su gautais modeliuojant Matlab aplinkoje.

Negalima teigti, kad šios statistikos yra kažkuo geresnės, tačiau jos suteikia papildomos informacijos apie signalą ir taip apsaugo nuo triukšmo išlikimo. Kaip tik dėl to yra gaunami geresni rezultatai, esant ir mažam SNR. Efektyvūs algoritmai yra tie, kuriuose galima apjungti ir išnaudoti skirtingų parametrų ir statistikų teikiamą informaciją apie signalą.

Literatūros sąrašas

[1] Beritelli, F.; Casale, S.; Cavallaero, A, *"A robust voice activity detector for wireless communications using soft computing"*, Selected Areas in Communications, IEEE Journal on, vol.16, issue 9, Dec. 1998, pp. 1818 – 1829.

[2] Chen Dong; Kuang Jingming, *"A robust voice activity detector applied for AMR"*, Department of Electronic Engineering, Beijing Institute of technology, 21 – 25 August 2000.

[3] Philippe Renevey and Andrzej Drygajlo, *"Entropy based voice activity detection in very noisy conditions"*, Swiss center for electronics and microtechnology, Swiss federal institute of technology, March, 2005.

[4] Chrysostomos L. Nikias; Anthina P. Petropulu, *"Higher-Order Spectra Analysis a nonlinear signal processing framework"*, PTR Prentice Hall, Englewood Cliffs, NJ: Prentice-Hall, 1993.

[5] Stephen W. Laverty; Donald R. Brown, *"Improved voice activity detection in the presence of passing vehicle noise"*, Worcester Polytechnic Institute, March, 2005.

[6] Virginie Gilg; Christophe Beaugeant; Martin Schonle; Bernt Andrassy; *"Methodology for the design of a robust voice activity detector for speech enhancement"*, International workshop on acoustic echo and noise control (IWAENC2003).

[7] G. Gabor; Z. Gyorfi, *"On the higher order distributions of speech signals"*, IEEE Trans. Acoust., Speech, Signal Processing, vol.36, pp. 602 – 603, April 1988.

[8] Beritelli. F; Casale. S; Ruggeri. G; Serrano. S, *"Performance evaluation and comparison of G.729/AMR/fuzzy voice activity detectors"*, Signal Processing Letters, IEEE, vol.9, issue 3, March 2002, pp. 85 – 88.

[9] Elias Nemer; Rafik Gooubran; Samy Mohmoud, *"Robust Voice Activity Detection Using Higher-Order Statistics in the LPC Residual Domain"*, IEEE Transactions on Speech and Audio Processing, vol.9, no.3, March 2001, pp. 217 - 231.

[10] Joachim Stegmann; Gerhard Schroder, *"Robust voice-activity detection based on the wavelet transform"*, Deutsche Telekom Berkom, Deutsche Telekom AG, 7 – 10 Sept, 1997.

[11] Jia-Lin Shen; Jein-Weih Hung; Lin-Shan Lee, *"Robust Entropy based endpoint detection for speech recognition in noisy environments"*, Institute of Information Science, Academia Sinica, Taipei, Taiwan, Republic of China, April, 2005.

[12] R. Fulchiero; A. Spanias, *"Speech enhancement using the bispectrum"*, in Proc. Int. Conf. Acoustics, Speech, Signal Processing, vol.4, 1993, pp. 488 – 491.

[13] C. Nikias; J. Mendel, *"Signal Processing with Higher-Order spectra"*, IEEE signal processing magazine, July 1993.

[14] J. Chen; K.K. Paliwal; S. Nakamura, *"Subtraction of Additive noise from corrupted speech for robust speech recognition"*, school of Microelectronics Engineering, Griffith University, Brisbane, QLD 4111, Australia, April 2005.

[15] Jerry M. Mendel, *"Tutorial on Higher-Order Statistics (Spectra) in Signal Processing and System Theory: Theoretical results and Some Applications"*, IEEE Transactions on Signal Processing, vol.79, no.3, March 1991.

[16] D.K. Freeman; G. Cosier; C.B. Southcott; I. Boyd, *"The voice activity detection for the pan-european digital cellular mobile telephone service"* in Proc. Int. Conf. acoustics, speech, signal processing, May 1989, pp. 369 – 372.

[17] Qifeng Zhu; Abeer Alwan, *"The effect of additive noise on speech amplitude spectra: a quantitative analysis"*, IEEE Signal Processing Letters, vol.9, no.9, September 2002.

[18] S. Gokhun Tanyer; Hamza Ozer, *"Voice activity detection in nonstatinary noise"*, IEEE transactions on speech and audio processing, vol.8, no.4, July 2000.

[19] http://en.wikipedia.org/wiki/Noise_(physics), April 2005.

[20] http://en.wikipedia.org/wiki/White_noise, April 2005.

[21] http://cnx.rice.edu/content/m0087/latest, April 2005.

[22] http://www.kanecomputing.co.uk/pdfs/dsk_6713a.pdf, April 2005.

[23] http://focus.ti.com/lit/ds/symlink/tms320c6713.pdf, April 2005.

[24] http://www.historicalvoices.org/spokenword/resources/audiotech/lpc.php, April 2005.

[25] Texas Instruments, *Code composer studio guide 2004*, May 2004.

[26] K.H. Davis; R. Biddulph; S. Balashek, *"Automtic Recognition of Spoken Digits"*, J. Acoust. Soc. Am., 24(6), 637-642, 1952.

[27] H.F. Olson; H. Belar, *"Phonetic Typewriter"*, J. Acoust. Soc. Am., 28(6), 1072-1081, 1956.

[28] D.B. Fry, *"Theoretical Aspects of Mechanical Speech Recognition"*, and P. Denes, *"The Design and Operation of the Mechanical Speech Recognizer at University College London"*, J. British Inst. Radio Engr., 19(4), 211-229, 1959.

[29] J.W. Forgie; C.D. Forgie, *"Results Obtained From a Vowel Recognition Computer Program"*, J. Acoust. Soc. Am., 31(11), 1480-1489, 1959.

[30] T. Sakai; S. Doshita, *"The Phonetic Typewriter, Information Processing 1962"*, Proc. IFIP Congress, Munich, 1962.

[31] T.B. Martin; A.L. Nelson; H.J. Zadell, *"Speech recognition by Feature Abstraction Techniques"*, Tech. Report AL-TDR-64-176, Air Force Avionics Lab, 1964.

[32] T.K. Vintsyuk, *"Speech Discrimination by Dynamic Programming"*, Kibernetika, 4(2), 81-88, Jan.-Feb., 1968.

[33] D.R. Reddy, *"An Approach to Computer Speech Recognition by Direct Analysis of the Speech Wave"*, Tech. Report No. C549, Computer Science Dept., Stanford Univ., September 1966.

[34] H. Sakoe, *"Two Level DP Matching - A Dynamic Programming Based Pattern Matching Algorithm for Connected Word Recognition"*, IEEE Trans. Acoustics, Speech, Signal Proc., ASSP-27, 588-595, December 1979.

[35] C.S. Myers; L.R. Rabiner, *"A Level Building Dynamic Time Warping Algorithm for Connected Word Recognition"*, IEEE Trans. Acoustics, Speech, Signal Proc., ASSP-29, 284-297, April 1981.

[36] C.H. Lee; L.R. Rabiner, *"A Frame Synchronous Network Search Algorithm for Connected Word Recognition"*, IEEE Trans. Acoustics, Speech, Signal Proc., ASSP-37 (11), 1649-1658, November, 1989.

[37] J. Ferguson, Ed., *"Hidden Markov Models for Speech"*, IDA, Princeton, NJ, 1980.

[38] L.R. Rabiner, *"A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition"*, Proc. IEEE, 77(2), 257-286, February 1989.

[39] R.P. Lippmann, *"An Introduction to Computing with Neural Nets"*, IEEE ASSP Magazine, 4(2), 4-22, April 1987.

[40] A. Weibel; T. Hanazawa; G. Hinton; K. Shikano; K. Lang, *"Phoneme Recognition Using Time-Delay Neural Networks"*, IEEE Trans. Acoustics, Speech, Signal Proc., ASSP-37, 393-404, 1989.

[41] *„TDMA minimum performance standards for discontinous transmission operation of mobile stations"*, TIA doc. and database IS-727, June 1998.