

UNIVERSITATEA „POLITEHNICA” DIN BUCUREȘTI

FACULTATEA DE ELECTRONICĂ, TELECOMUNICAȚII ȘI TEHNOLOGIA
INFORMAȚIEI

SISTEM DE RECUNOAȘTERE ȘI CLASIFICARE A PLÂNSETELOR
NOU-NĂSCUȚILOR

Proiect de diplomă

prezentat ca cerință parțială pentru obținerea titlului de *Inginer* în domeniul
Electronică Aplicată și Ingineria Informației
Programul de studii *Ingineria Informației*

Conducători științifici

Prof. Corneliu BURILEANU

Absolvent

Georgiana RĂNGHIUC

București
2017

Universitatea "Politehnica" din București
Facultatea de Electronică, Telecomunicații și Tehnologia Informației
Departamentul: Electronică Aplicată și Ingineria Informației

Aprobat Director de Departament *:


Prof. Dr. Ing. Sever PAȘCA

TEMA PROIECTULUI DE DIPLOMĂ a studentei Georgiana Rânghiuc, 441A

1. Titlul temei: **"Sistem de recunoaștere și clasificare a plânsetelor nou-născuților"**

2. Contribuția practică, originală a studentului va consta în (în afara părții de documentare):

- Această lucrare își propune studiul asupra modelelor de producere a vorbirii și, în particular, a plânsetelor nou-născuților și realizarea unui sistem de recunoaștere automată a cifrelor rostite utilizate din ce în ce mai des în cadrul transcrierii speech-to-text.
- Se va realiza extragerea mai multor tipuri de parametri, care să caracterizeze diversele tipuri de plânsete, și anume: parametri wavelet ce sunt diferențiați prin familiile wavelet corespunzătoare. Acești parametri vor fi utilizați în discriminarea plânsetelor nou-născuților.
- Se vor realiza teste pe două baze de date diferite: baza de date DUNSTAN care face referire la cinci tipuri de plânsete și baza de date SPLANN care face referire la plânsetele prezente și în baza DUNSTAN, dar care mai adaugă încă alte două tipuri. De asemenea, baza de date SPLANN este o baza de date extinsă care ne permite o clasificare mai bună. Se vor face clasificări pentru a face o diferențiere între diversele tipuri de plânsete ale nou-născuților, cu clasificatorul: Suport Vector Machine.

3. Proiectul se bazează pe cunoștințe dobândite în principal la următoarele 3-4 discipline: Recunoașterea formelor și inteligența artificială, Programare orientată pe obiect, Prelucrarea digitală a semnalelor.

4. Proprietatea intelectuală asupra proiectului aparține: Laborator „Speed”, UPB

5. Locul de desfășurare a activității: CAMPUS, sala 204

6. Realizarea practică rămâne în proprietatea: Laborator „Speed”, UPB

7. Data eliberării temei: 28 iulie 2016

CONDUCĂTOR LUCRARE:

Prof. Corneliu BURILEANU

Ș.L. Dr. Ing. Prof. Horia CUCU

STUDENT:

Georgiana Rânghiuc

DECLARAȚIE DE ONESTITATE ACADEMICĂ

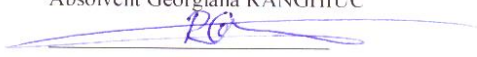
Prin prezenta declar că lucrarea cu titlul "Sistem de recunoaștere și clasificare a plâșetelor nou-născuților", prezentată în cadrul Facultății de Electronică, Telecomunicații și Tehnologia Informației a Universității "Politehnica" din București ca cerință parțială pentru obținerea titlului de *Inginer* în domeniul *Electronică Aplicată și Ingineria Informației*, programul de studii *Ingineria Informației*, este scrisă de mine și nu a mai fost prezentată niciodată la o facultate sau instituție de învățământ superior din țară sau străinătate.

Declar că toate sursele utilizate, inclusiv cele de pe Internet, sunt indicate în lucrare, ca referințe bibliografice. Fragmentele de text din alte surse, reproduse exact, chiar și în traducere proprie din altă limbă, sunt scrise între ghilimele și fac referință la sursă. Reformularea în cuvinte proprii a textelor scrise de către alți autori face referință la sursă. Înțeleg că plagiatul constituie infracțiune și se sancționează conform legilor în vigoare.

Declar că toate rezultatele simulărilor, experimentelor și măsurătorilor pe care le prezint ca fiind făcute de mine, precum și metodele prin care au fost obținute, sunt reale și provin din respectivele simulări, experimente și măsurători. Înțeleg că falsificarea datelor și rezultatelor constituie fraudă și se sancționează conform regulamentelor în vigoare.

București, 04.09.2017

Absolvent Georgiana RĂNGHIUC


(semnătura în original)

CUPRINS

Cuprins.....	7
Lista Figurilor.....	9
Lista Tabelelor.....	12
Acronime.....	13
Capitolul 1 Introducere	15
Motivația tezei	15
Capitolul 2 Descrierea Bazelor de Date	17
2.1 Baza de date Dunstan.....	17
2.2 Baza de date SPLANN.....	19
Capitolul 3 Teoria Wavelet-urilor.....	23
3.1 Introducere în teoria Wavelet-urilor	23
3.2 Influența numărului de momente nule	29
3.3 Scurtă introducere despre noțiunea de plânsete ale nou-născuților și studii realizate	30
3.4 Analiza semnalelor existente în baza de date	31
3.5 Familii de Wavelet-uri	32
3.6 Implementare matematică – Analiza Wavelet a plânsetelor nou-născuților(Metoda I)	35
Capitolul 4 Extragerea de trăsături robuste pentru clasificarea plânsetelor nou-născuților(Metoda II).....	41

4.1 Transformata Wavelet Pentru Cazul Procesării de semnale.....	41
4.2 Implementarea Software a funcției de extragere și calcul al coeficienților wavelet	42
4.3 Împărțirea bazei de date.....	43
4.4 Calculul Trăsăturilor.....	45
Capitolul 5 Experimente	47
Capitolul 6 Concluzii.....	51
6.1 Concluzii generale	51
6.2 Contribuții Personale	52
6.3 Experimente ulterioare.....	53
Referințe.....	55
Anexe.....	59

LISTA FIGURILOR

Figura 3.1 Exemplificarea semnalelor staționare.....	24
Figura 3.2 Exemplificarea semnalelor nestaționare.....	25
Figura 3.3 Undișoara mama Daubechies 12	27
Figura 3.4 Funcția de autocorelație a unei undișoare mamă.....	27
Figura 3.5 Comparație între factorii de compresie $c=1$ (stânga) și $c=1/4$ (dreapta) ai aceleiași funcții de bază.....	28
Figura 3.6 STFT vs. WT.....	28
Figura 3.7 Comparație între transformatele Fourier și Wavelet	29
Figura 3.8 Familia Daubechies – funcția de scalare și funcția Wavelet corespunzătoare	32
Figura 3.9 Familia Biortonormal – funcția de scalare și funcția Wavelet corespunzătoare.	33
Figura 3.10 Familia Biortonormal – funcția de scalare și funcția Wavelet corespunzătoare reconstrucției.....	33
Figura 3.11 Familia Symlet – funcția de scalare și funcția Wavelet corespunzătoare reconstrucției.....	34
Figura 3.12 Familia Coiflet 3– funcția de scalare și funcția Wavelet corespunzătoare reconstrucției.....	34
Figure 3.13 Familia Coiflet 5 – funcția de scalare și funcția Wavelet corespunzătoare reconstrucției.....	35

Figure 3.14 Familia Discret Meyer – funcția de scalare și funcția Wavelet corespunzătoare reconstrucției.....	35
Figure 3.15 Implementarea DWT folosind bancuri de filtre	36
Figure 3.16 Implementarea IDWT folosind bancuri de filtre	38
Figure 3.17 Matricea de parametri corespunzătoare primei metode	39
Figure 4.1 Descompunerea semnalului cu ajutorul funcției wavedec	43
Figure 4.2 Setul de antrenare(stânga) și setul de testare(dreapta)	44
Figure 4.3 Matricea corespunzătoare parametrilor extrași	46

LISTA TABELELOR

Tabel 3.1 Selecție a performanțelor BER pentru $f_m=0.005$ și o singură iterație IDWT	30
Tabel 6.1 Rezultate pentru metoda I.....	48
Tabel 6.2 Rezultate pentru metoda II	49

Copyright © 2017, Georgiana RĂNGHIUC

Toate drepturile rezervate

Autorul acordă laboratorului „Speed” din cadrul UPB dreptul de a reproduce și de a distribui public copii pe hârtie sau electronice ale acestei lucrări, în formă integrală sau parțială.

ACRONIME

CTDR – Closure interval-transient detail ratio

DWT – Discrete Wavelet Transform

FBI – Federal Bureau of Investigation

IDWT – Inverse Discrete Wavelet Transform

MRA – Multi Resolution Analysis

PVD – Power variation of detail coefficients

SAR – Short-term logarithmic average energy

SPLANN – Sistem de recunoaștere automată a plânsetelor nou-născuților

SVM – support Vector Machine

WPR – Wavelet Power Ratio

ZCR – Zero Crossing Rate

CAPITOLUL 1

INTRODUCERE

MOTIVAȚIA TEZEI

De-a lungul timpului am dorit să îmbin partea medicală cu cea a electronicii pentru a-i ajuta pe cei din jurul meu și pentru a aduce un aport de lucruri folositoare lumii în care trăim. În anii petrecuți în spital de-a lungul vremii și din cunoștințele pe care le-am acumulat, am observat că există tinere mame care au nevoie de ajutorul unei alte persoane, care să fi trecut deja prin etapa de a avea în propriile brațe un copil abia născut, care să le explice ceea ce se întâmplă de fapt cu copilul lor atunci când acesta se exprimă prin metoda plânsului și ce nevoi se ascund în spatele plânsului propriu-zis.

În ultimii ani s-au dezvoltat tot felul de studii pe această parte de neonatologie din dorința de a ajuta tinerii părinți să își înțeleagă mai bine copilul atunci când acesta încă nu se poate folosi de aparatul fonator pentru a vorbi, ci se folosește pentru a scoate diverse sunete care îi indică necesitățile. Mulți din cei care au pentru prima dată grijă de un nou-născut întâmpină probleme în a înțelege ceea ce se petrece cu copilul lor atunci când acesta plânge. Deseori este necesară apariția unei persoane în vârstă, fiind reprezentată de unul din bunici sau fiind în interiorul spitalului să fie reprezentată de unul din specialiștii în domeniul neonatologiei, sau chiar de un medic pediatru. Atunci când mama părăsește maternitatea și merge cu copilul în interiorul

căminului ei, de cele mai multe ori are nevoie de o persoană care să o ajute să soluționeze în timp util nevoile pe care le arată bebelușul său. Pentru aceasta am realizat o serie de teste pe baza de date Dunstan pentru a obține o serie de rezultate experimentale pentru a clasifica cât mai bine plânsurile nou-născuților, astfel aflând ceea ce se petrece cu adevărat în spatele plânsului celui mic.

În studiile ce s-au realizat până acum s-a dedus faptul că putem analiza informația ce se regăsește în semnal pentru bebelușii ce nu au împlinit vârsta de trei luni, de atunci aparatul lor schimbându-se treptat și focalizându-se pe producerea altor sunete din care nu se pot extrage informații concludente care să ne ajute la observarea necesităților bebelușului.

Pentru toate acestea și din dorința de a ajuta părinții ce se află la început de drum în a crește un nou-născut ne-am gândit să dezvoltăm un algoritm care să recunoască și să clasifice plânsurile nou-născuților.

Acest algoritm presupune două metode ce se bazează pe extragerea coeficienților wavelet din semnalele ce aparțin bazei de date DUNSTAN. După extragerea coeficienților wavelet propriu-ziși, vom lua în calcul două metode după cum urmează: metoda I presupune calculul energiilor pe toată durata semnalului și pentru toate nivelele pentru care s-au extras coeficienții wavelet și asocierea unui label corespunzător tipului de semnal, după care vom antrena clasificatorul SVM cu o parte din date, ca mai apoi să îl testăm pe date ce nu fac parte din etapa de antrenare. Metoda II presupune calculul următorilor parametri: WPR, PVD, SAR, ZCR, CTDR, care sunt calculați cu ajutorul coeficienților de aproximație și de detaliu ce au fost extrași cu ajutorul transformatei wavelet pentru trei nivele.

În cazul ambelor metode se vor alege familii diferite de wavelet-uri pentru a se putea observa care tip de funcție se potrivește mai bine cu tipul semnalelor noastre. De aceea ca și rezultate experimentale în cadrul acestei lucrări de diplomă s-au obținut următoarele rezultate experimentale, după cum urmează:

- În cadrul primei metode, cea mai bună acuratețe s-a obținut pentru familia de Wavelet-uri Discrete Meyer(dmey) și are o valoare de 61.53%;
- În cadrul celei de-a doua metode, cea mai bună acuratețe s-a obținut pentru familia de Wavelet-uri Symlet5(sym5) și are o valoare de 65.21%

Precizez că aceste rezultate sunt raportate la un număr de 37 de bebeluși, fiecare din ei având tipuri diferite de plânsuri și experimentele realizându-se pe o bază de date sigură, etichetată corect de către specialiștii în domeniu.

CAPITOLUL 2

DESCRIEREA BAZELOR DE DATE

2.1 BAZA DE DATE DUNSTAN

În ultimii ani s-au realizat mai multe experimente din care a reieșit faptul că se poate extrage informație utilă din plânsetele nou-născuților. S-a ajuns la concluzia că în spatele acestor plânsete se află de fapt o necesitate, iar bebelușul nu se poate exprima altfel decât prin intermediul acestor tipuri de sunete.

Încă din copilărie Perscilla Dunstan a fost înzestrată cu abilități de recunoaștere, interpretare și reproducere a unor tipuri de semnale complexe ca și conținut și de-a lungul vieții a observat un comportament similar și în cazul nou-născuților, unde a dedus un oarecare tipar pentru fiecare tip de plânset, analizând și observând că acesta se repetă de la un nou-născut la altul, indiferent de rasă sau sex. Aceste lucruri au fost expuse în practică și astfel s-a realizat etichetarea bazei de date DUNSTAN, de care am dispus și noi în cadrul acestei lucrări de diplomă. Perscilla a analizat fiecare tip de plânset și astfel s-a dedus faptul că putem eticheta cu o precizie satisfăcătoare cinci tipuri diferite de plânset. Aceste cinci tipuri au fost etichetate cu ajutorul medicilor de specialitate: medici neonatologi și medici pediatri, o echipă întreagă de oameni care s-au ocupat nu doar de etichetare, ci și de clasificarea corectă a acestor tipuri de plânsete, în spatele cărora se înfățișează de fapt o nevoie a nou-născutului. Este de precizat faptul că studiile s-au realizat pentru început și pe imagini ale nou-născuților care plâng, deoarece se pot extrage și de aici informații care să ateste necesitatea celui mic. Studiul s-a realizat pe un număr de 37 de bebeluși care au fost înregistrați pentru cinci tipuri de plânsete(disconfort, foame, somn, eructație, dureri de burtă), după cum urmează:

- Pentru bebelușul cu ID-ul 1 regăsim în baza de date plânsete pentru: disconfort(HEH), foame(NEH), somn(OWH);
- Pentru bebelușul cu ID-ul 2 regăsim în baza de date plânsete pentru: foame(NEH);

- Pentru bebelușul cu ID-ul 3 regăsim în baza de date plânsuri pentru: eructație(EH), foame(NEH), somn(OWH), disconfort(HEH);
- Pentru bebelușul cu ID-ul 4 regăsim în baza de date plânsuri pentru: foame(NEH);
- Pentru bebelușul cu ID-ul 5 regăsim în baza de date plânsuri pentru: durere de burtă(EAIRH), eructație(EH), foame(NEH), disconfort(HEH);
- Pentru bebelușul cu ID-ul 6 regăsim în baza de date plânsuri pentru: eructație(EH);
- Pentru bebelușul cu ID-ul 7 regăsim în baza de date plânsuri pentru: foame(NEH);
- Pentru bebelușul cu ID-ul 8 regăsim în baza de date plânsuri pentru: foame(NEH);
- Pentru bebelușul cu ID-ul 9 regăsim în baza de date plânsuri pentru: foame(NEH), disconfort(HEH), somn(OWH);
- Pentru bebelușul cu ID-ul 10 regăsim în baza de date plânsuri pentru: foame(NEH), disconfort(HEH), somn(OWH);
- Pentru bebelușul cu ID-ul 11 regăsim în baza de date plânsuri pentru: foame(NEH), dureri de burtă(EAIRH), foame(NEH), somn(OWH);
- Pentru bebelușul cu ID-ul 12 regăsim în baza de date plânsuri pentru: foame(NEH), eructație(EH), disconfort(HEH), somn(OWH);
- Pentru bebelușul cu ID-ul 14 regăsim în baza de date plânsuri pentru: foame(NEH);
- Pentru bebelușul cu ID-ul 15 regăsim în baza de date plânsuri pentru: foame(NEH), somn(OWH);
- Pentru bebelușul cu ID-ul 16 regăsim în baza de date plânsuri pentru: eructație(EH), disconfort(HEH), somn(OWH);
- Pentru bebelușul cu ID-ul 17 regăsim în baza de date plânsuri pentru: foame(NEH), somn(OWH);
- Pentru bebelușul cu ID-ul 18 regăsim în baza de date plânsuri pentru: foame(NEH), dureri de burtă(EAIRH), eructație(EH), somn(OWH);
- Pentru bebelușul cu ID-ul 19 regăsim în baza de date plânsuri pentru: foame(NEH);
- Pentru bebelușul cu ID-ul 20 regăsim în baza de date plânsuri pentru: foame(NEH), eructație(EH);
- Pentru bebelușul cu ID-ul 21 regăsim în baza de date plânsuri pentru: foame(NEH), eructație(EH), disconfort(HEH), dureri de burtă(EAIRH);
- Pentru bebelușul cu ID-ul 22 regăsim în baza de date plânsuri pentru: foame(NEH), eructație(EH), disconfort(HEH);
- Pentru bebelușul cu ID-ul 23 regăsim în baza de date plânsuri pentru: foame(NEH), eructație(EH);
- Pentru bebelușul cu ID-ul 25 regăsim în baza de date plânsuri pentru: foame(NEH);
- Pentru bebelușul cu ID-ul 26 regăsim în baza de date plânsuri pentru: somn(OWH);
- Pentru bebelușul cu ID-ul 28 regăsim în baza de date plânsuri pentru: dureri de burtă (EAIRH), somn(OWH);
- Pentru bebelușul cu ID-ul 29 regăsim în baza de date plânsuri pentru: somn(OWH);
- Pentru bebelușul cu ID-ul 30 regăsim în baza de date plânsuri pentru: eructație(EH), somn(OWH);

- Pentru bebelușul cu ID-ul 31 regăsim în baza de date plânsuri pentru: somn(OWH);
- Pentru bebelușul cu ID-ul 32 regăsim în baza de date plânsuri pentru: eructație(EH), disconfort(HEH), somn(OWH);
- Pentru bebelușul cu ID-ul 35 regăsim în baza de date plânsuri pentru: dureri de burtă(EAIRH);
- Pentru bebelușul cu ID-ul 36 regăsim în baza de date plânsuri pentru: dureri de burtă(EAIRH);
- Pentru bebelușul cu ID-ul 37 regăsim în baza de date plânsuri pentru: dureri de burtă(EAIRH);
- Pentru bebelușul cu ID-ul 38 regăsim în baza de date plânsuri pentru: dureri de burtă(EAIRH);
- Pentru bebelușul cu ID-ul 39 regăsim în baza de date plânsuri pentru: dureri de burtă(EAIRH);
- Pentru bebelușul cu ID-ul 40 regăsim în baza de date plânsuri pentru: disconfort(HEH);
- Pentru bebelușul cu ID-ul 41 regăsim în baza de date plânsuri pentru: disconfort(HEH).

În cadrul acestei baze de date se găsesc, pentru fiecare clasă, următorul număr de bebeluși care au fost înregistrați pentru următoarele tipuri de plânsuri:

- pentru tipul plânsurilor ce ilustrează durerea de burtă, regăsim un număr de 10 bebeluși;
- pentru tipul plânsurilor ce ilustrează eructația, regăsim un număr de 12 bebeluși;
- pentru tipul plânsurilor ce ilustrează foamea, regăsim un număr de 23 de bebeluși;
- pentru tipul plânsurilor ce ilustrează disconfortul, regăsim un număr de 12 bebeluși;
- pentru tipul plânsurilor ce ilustrează somnul, regăsim un număr de 16 bebeluși.

Precizez faptul că în cazul unor bebeluși există două sau mai multe plânsuri care fac parte din aceeași clasă, dar care au fost înregistrate la momente diferite de timp.

2.2 BAZA DE DATE SPLANN

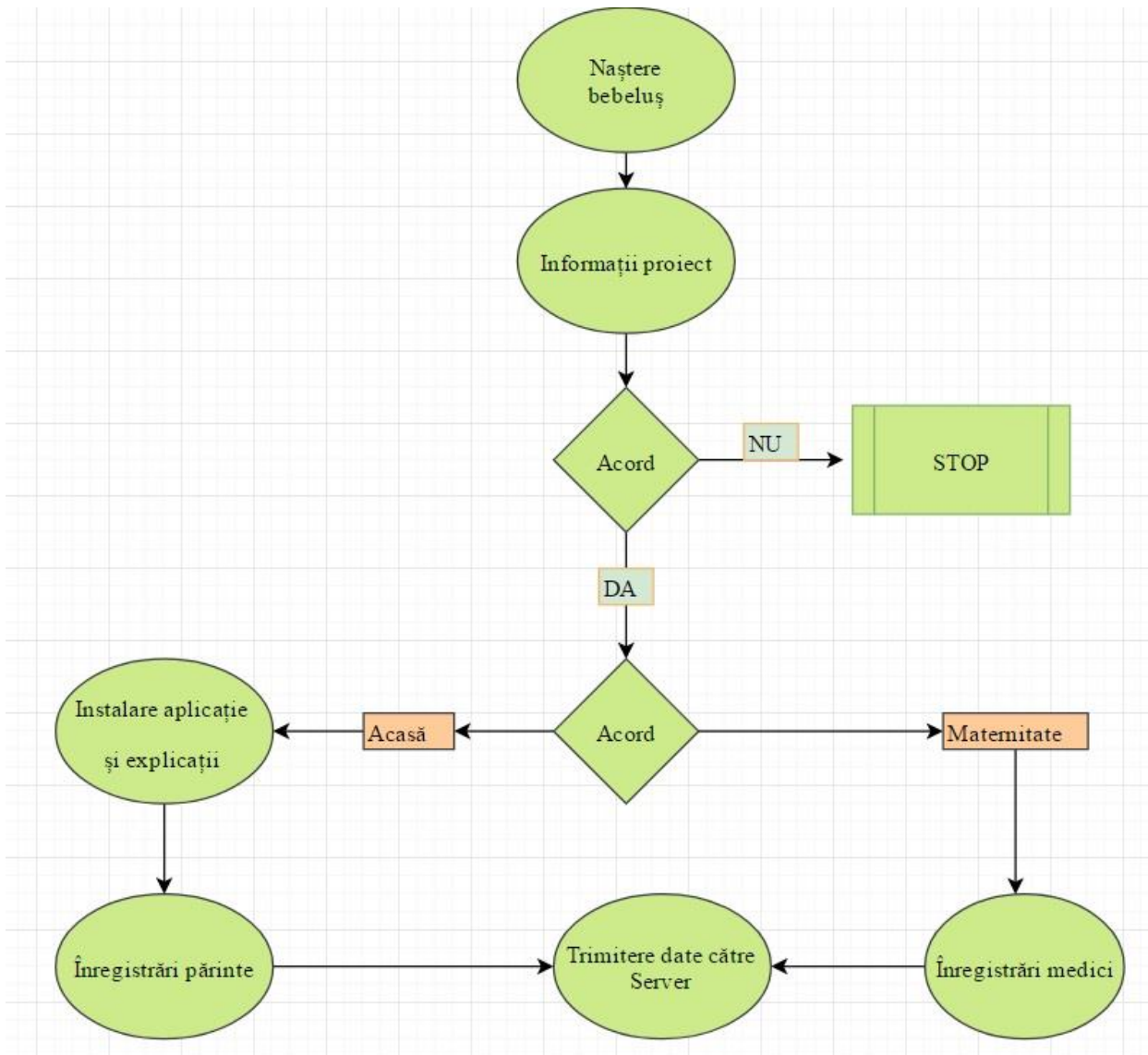
Din dorința de a extinde baza de date DUNSTAN și de a reuși clasificarea cât mai precisă a plânsurilor nou-născuților s-a inițiat proiectul SPLANN împreună cu echipa celor de la SOFTWIN în care se dorește punerea în folosul oamenilor a unei aplicații care să recunoască și să clasifice cât mai bine aceste semnale în funcție de nevoile pe care le are nou-născutul. Pentru aceasta s-a realizat o colaborare cu Spitalul Sf. Pantelimon din București, cu medicii ce fac parte din secția de neonatologie.

Acest proiect presupune înregistrarea nou-născuților atât în cadrul maternității, cât și în afara ei, atunci când mama părăsește spitalul. Acest lucru se poate realiza doar dacă mama sau tatăl își exprimă acordul pentru acest lucru prin semnarea unui acord de confidențialitate în care noi ne luăm angajamentul față de ei că vom înregistra nou-născutul cu ajutorul medicilor din spital cu ajutorul aplicației care este încă în dezvoltare, că vom păstra datele lor confidențiale și că nu vor fi folosite în alte scopuri decât cele care au fost prevăzute în cadrul acestui proiect. Dacă părintele își dă acordul pentru a face el înregistrări acasă cu copilul atunci când acesta plânge atunci noi îi instalăm aplicația pe telefon și ne luăm totodată responsabilitatea de a-i explica cum funcționează și de a-l ajuta când acesta întâmpină probleme, iar ei își iau responsabilitatea de a

ne transmite datele către server și de a ne anunța de eventualele nereguli în ceea ce privește aplicația dezvoltată până în momentul de față. Menționez că părinții nu sunt constrânși pentru a participa în cadrul acestui studiu, ei fiind informați de ceea ce se va face și doar cu acordul lor se vor face înregistrările, atât în maternitate, cât și acasă.

Pentru a păstra datele nou-născutului intacte și confidențiale după completarea documentelor necesare de către medici, acestuia i se va atribui un ID care va face conexiunea între baza de date și numele său propriu-zis atunci când vor apărea înregistrările.

Proiectul SPLANN s-a desfășurat după următoarea structură în vederea înregistrării plânsurilor nou-născuților:



În momentul în care părintele își exprima acordul pentru a face înregistrări în momentul în care nou-născutul plânge, noi ne-am luat responsabilitatea de a-i instala aplicația pe un telefon ce dispunea de un sistem de operare Android și de a-i explica ceea ce se întâmpla în spatele înregistrării propriu-zise. Aplicația este disponibilă pe site-ul: www.softwinresearch.ro/Apps/ de unde noi descărcăm un apk pentru telefonul mamei, după care instalăm aplicația și îi atribuim ID-ul bebelușului cu care realizăm, de asemenea, partea de login. Acestui ID îi este atribuită o parolă care se găsește într-un document securizat ce este pus la dispoziție doar persoanelor autorizate pentru a realiza instalarea aplicației și care au acces la baza de date și la fișele confidențiale. De asemenea, în cadrul acestui proiect s-au semnat acorduri de confidențialitate,

atât față de părinți, cât și față de corporația SOFTWIN, că datele sunt confidențiale și divulgarea numelor pacienților, copiilor, parolelor, ID-urilor sau înregistrărilor propriu-zise implică nerespectarea unor articole din Consimțământ, iar încălcarea lor atrage după sine și pedeapsa corespunzătoare.

Manipularea aplicației este una foarte simplă ce nu îi va pune dificultăți proaspetei mămici, de altfel acest lucru fiind luat în considerare de către noi atunci când ne-am gândit la aceasta. Pentru informații suplimentare, sau pentru dificultăți în vederea folosirii aplicației, după ce părăsesc spitalul, părinții au primit datele noastre de contact, pentru a rezolva eventualele neplăceri create de aplicație.

Etichetarea bazei de date se face cu o dificultate ridicată deoarece aceste semnale trebuiesc etichetate de medici specializați în domeniu, nu se poate realiza de către altcineva deoarece aceasta ar presupune etichetarea greșită și ar duce inevitabil către erori mari atunci când s-ar dori recunoașterea automată și etichetarea acestor semnale. Aceasta presupune realizarea unor înregistrări cât mai clare și cu aparatură specializată, de aceea pentru partea în care înregistrările au loc în cadrul maternității au fost puse la dispoziție microfoane și telefoane pentru fiecare medic în parte pentru a face înregistrările cât mai concludente și cu cât mai puțin zgomot. După realizarea acestor înregistrări medicul care s-a ocupat de aceasta, trimite către server datele și face etichetarea în funcție de ceea ce aude și interpretează. Aici există următoarea situație prin care medicul realizează etichetarea: după ce s-a realizat înregistrarea, acesta poate aștepta să observe ceea ce se întâmplă cu nou-născutul și abia apoi trimite datele corecte către server, adică etichetarea corectă a plânsului ce ilustrează una din cele cinci nevoi. Există posibilitatea ca acesta să greșească o etichetare și aceasta să ajungă în baza de date alterată, de aceea atunci când această parte de achiziție se termină, semnalele sunt procesate și este ascultată fiecare înregistrare în vederea verificării corectitudinii etichetărilor făcute.

Pentru instalarea fiecărei aplicații s-a păstrat constant legătura între medicii din Spitalul Sf. Pantelimon, București, echipa Softwin și echipa din cadrul Laboratorului de Cercetare Speed, din cadrul Universității Politehnica, București.

Această bază de date, față de baza de date DUNSTAN conține în plus două clase separate care ilustrează durerea patologică și durerea exercitată de corpul nou-născutului. Ea conține în total șapte clase, după cum urmează: Colici, Disconfort, Durere, Durere patologică, Oboseală/Somn, Foame, Eructație.

În aceste cazuri pentru fiecare nevoie dispunem la momentul actual de o cantitate de semnale distribuite în funcție de nevoi după cum urmează:

- Pentru nevoia care ilustrează plânsete specifice Colicilor dispunem de 225 de semnale etichetate;
- Pentru nevoia care ilustrează plânsete produse de Disconfort dispunem de 2210 de semnale etichetate;
- Pentru nevoia care ilustrează plânsete specifice Durerii dispunem de 4404 de semnale etichetate;
- Pentru nevoia care ilustrează plânsete specifice Durerii patologice dispunem de 459 de semnale etichetate;
- Pentru nevoia care ilustrează plânsete produse de Oboseală/Somn dispunem de 34 de semnale etichetate;
- Pentru nevoia care ilustrează plânsete produse de Foame dispunem de 5536 de semnale etichetate;
- Pentru nevoia care ilustrează plânsete specifice Eructației dispunem de 505 de semnale etichetate.

În această bază de date dispunem de un total de 12914 semnale ce fac parte din cele șapte clase enumerate mai sus. Acestea au fost etichetate de medicii care au fost implicați în cadrul acestui proiect, segmentate și verificate de către echipa Softwin împreună cu echipa din cadrul Laboratorului de Cercetare Speed din cadrul Universității Politehnica, București.

De asemenea, trebuie precizat faptul că a crescut și numărul bebelușilor de la 37 la 143 și că există bebeluși care pot ilustra un singur tip de nevoie în cadrul acestor înregistrări.

CAPITOLUL 3

TEORIA WAVELET-URILOR

3.1 INTRODUCERE ÎN TEORIA WAVELET-URILOR

În ultimele două decenii unul din subiectele ce se bucură de un interes ridicat este cel specific teoriei semnalelor, mai exact cel al transformatei Wavelet. Cu excepția notabilă a semnalelor ale căror caracteristici nu pot fi separate de cele ale sistemelor care le-au generat, tehnicile de analiză multirezoluție, în special cele care au fost axate pe utilizarea transformatei Wavelet, s-au aflat mereu în centrul preocupărilor unor numeroase centre de cercetare, dar au pătruns cu succes și în aplicații practice dintre cele mai diverse. Ca și aplicație practică este suficient să amintim apariția standardului FBI de stocare a amprentelor sau standardul de compresie JPEG2000 pentru a putea intui plaja largă de posibilități de utilizare a acestor instrumente de procesare.

În ceea ce privește modelul teoretic analiza de tip wavelet poate fi considerată ca fiind ajunsă la maturitate, odată ce a apărut un set semnificativ de cărți și lucrări de specialitate publicate în reviste de prestigiu. Din nefericire, majoritatea resurselor bibliografice au fost elaborate de către matematicieni, astfel încât înțelegerea lor riguroasă se dovedește dificilă pentru cei ce doresc să dezvolte aplicații și să implementeze algoritmi ce pot fi utilizați mai apoi în practică. Din acest punct de vedere în acest capitol vom folosi instrumente familiare teoriei semnalelor și sistemelor, făcând referire la noțiuni binecunoscute de filtrare sau algebră liniară, cu scopul de a ne putea forma o imagine cât mai clară asupra acestui subiect.

Semnalele produse de nou-născuți ca și celelalte semnale sunt reprezentate ca manifestări purtătoare de informație și, de aceea, suntem interesați să putem modela, clasifica și caracteriza aceste semnale prin intermediul unor mărimi obiective. În cazul în care discutăm despre semnale unidimensionale amintim, de regulă, o caracterizare în domeniul timp, pe când în cazul semnalelor bidimensionale (spre exemplu, cazul imaginilor) se amintește domeniul spațial. Informații asemănătoare sunt oferite pe baza analizei compoziției spectrale (a informației din domeniul frecvență), ce sunt puse la dispoziția noastră prin intermediul transformatei Fourier. Deși este o metodă bine cunoscută, nelipsită din cărțile de prelucrări de semnale, această transformată este dificil de utilizat atunci când ne dorim să cunoaștem compoziția spectrală la un anumit moment dat de timp.

Astfel de exemple întâlnim în practică, spre exemplu în cazul prelucrării semnelor seismice, a seriilor de timp financiare sau a unor semnale biomedicale, aici încadrând și semnalele specifice nou-născuților cu vârsta cuprinsă între zero și trei luni. Mai mult decât atât, există aplicații în care ne dorim să aplicăm operații de filtrare care să ne permită menținerea cât mai nedeteriorată a momentelor de timp în care apar modificări semnificative ale semnalului original (deoarece aspectul general, prezența și duratele unor astfel de segmente sunt purtătoare de informație), însă în acest caz transformata Fourier ne asigură cu dificultate aceste obiective.

În literatură întâlnim două tipuri de semnale care după tipul de staționaritate se împart în semnale staționare și semnale nestaționare. Cazul semnalelor staționare presupune că semnalele cu compoziție în frecvență sunt invariate în timp și toate componentele de frecvență există pe toată durata semnalului.

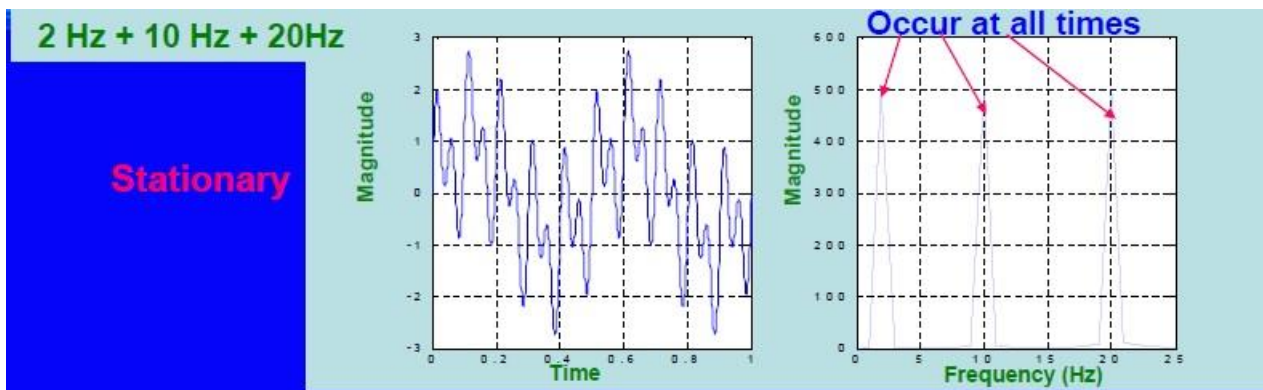


Figura 3.1 Exemplificarea semnalelor staționare

Cazul semnalelor nestaționare ne este ilustrat prin faptul că frecvența se modifică tot timpul.

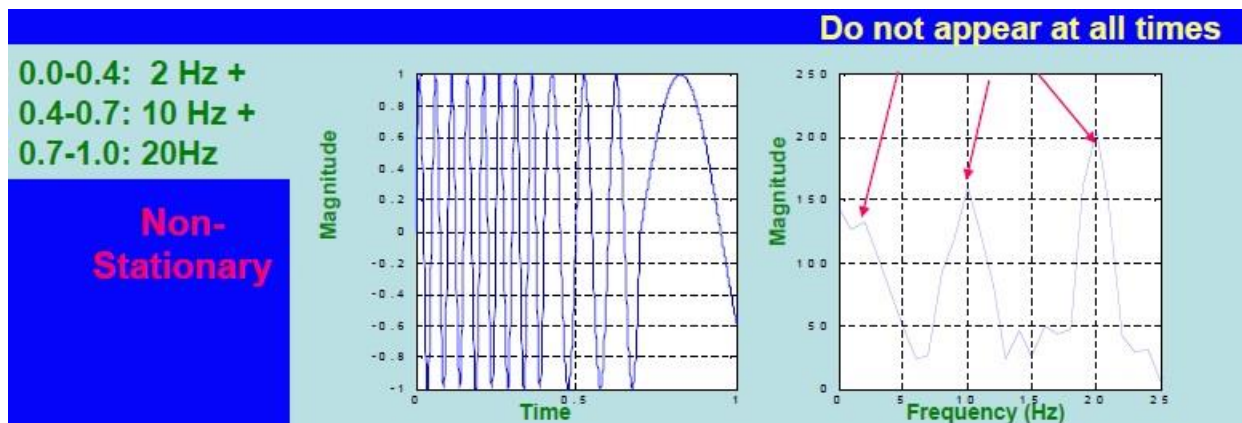


Figura 3.2 Exemplificarea semnalelor nestaționare

În funcție de semnalele pe care ne dorim să le clasificăm și să extragem o serie de parametri cât mai reprezentativi am ales această transformată din varii motive, printre care găsim atât avantaje, cât și dezavantaje: ni se permite convertirea unui semnal în serii de unde wavelet și ni se asigură un mod de a le analiza atât în timp, cât și în frecvență. De asemenea față de transformata Fourier putem stoca mult mai ușor semnalele ce sunt obținute și pot fi approximate mult mai eficient semnalele întâlnite în viața cotidiană. Spre deosebire de transformata Fourier despre care se întâlnesc date mult mai repede, în cantitate mult mai mare și care se bucură de o popularitate mult mai mare, astfel putând fi aplicată mult mai ușor, transformata Wavelet presupune să obținem informații suplimentare din semnal, care nu sunt disponibile în forma brută a acestuia.

O alternativă a transformatei Fourier pe termen scurt (STFT) pentru a putea depăși problema de rezoluție este cea a folosirii transformatei Wavelet.

Funcția wavelet este rezultatul unui filtru terțe bandă și a unei scalări care are ca scop înjumătățirea lungimii de bandă (apar probleme pentru acoperirea întregului spectru, pentru a rezolva această cerință ar fi necesar un număr infinit de nivele).

Funcția de scalare are rolul de a filtra nivelul cel mai scăzut al transformatei și asigură acoperirea întregului spectru.

Analiza multirezoluție ne permite să analizăm semnalul la frecvențe diferite cu rezoluții diferite. Undișoarele pot oferi o rezoluție temporală bună la frecvențe înalte, și o rezoluție în frecvență bună pentru cazul frecvențelor joase. Acest lucru nu este de dorit atunci când vorbim despre analiza semnalelor, deoarece frecvențele joase presupun o evoluție lentă a semnalului, în timp ce acelea înalte se regăsesc în tranziții bruște în semnal, a căror „captare” este favorizată de o rezoluție temporală bună.

Această modalitate de partajare a planului timp-frecvență poate fi obținută prin translatarea și scalarea pe axa timpului a unei funcții unice ce se numește undișoara mamă

$\psi(t) : \psi_{s,\tau} = \frac{1}{\sqrt{s}} \psi\left(\frac{t-\tau}{s}\right)$, variabilele de scară (s) și cele de poziționare pe axa timpului (τ) sunt variabile continue. Dacă se va încerca discretizarea acestor variabile, vom putea obține o versiune discretă de undișoară mamă, $\psi_{j,k}(t)$. Este de remarcat că nu variabila timp este cea care

ne oferă versiunea discretizată a undișoarei, ci ceilalți doi parametri ai acesteia:

$$\psi_{j,k}(t) = s_0^{-\frac{j}{2}} \psi(s_0^{-j} \cdot t - k\tau_0) \quad (1)$$

Pentru a putea obține această versiune discretizată a familiei de undișoare, $\{\psi_{j,k}(t)\}$, relațiile folosite au fost: $s = s_0^j$ și $\tau = ks_0^j\tau_0$. O alegere des întâlnită pentru s_0 este $s_0=2$, care ne conduce la undișoare folosite în cazul transformatei numite Transformare Wavelet Diadică. Dacă facem referire acum la un semnal de timp continuu $x(t)$, versiunea discretizată a transformării wavelet continue va fi de forma:

$$DWT_x(j,k) = \int_{-\infty}^{\infty} x(t) \cdot \psi_{j,k}^*(t) \cdot dt \quad (2)$$

Relația amintită mai sus definește de fapt un produs scalar între semnalul nostru, $x(t)$ și o funcție din familia $\{\psi_{j,k}(t)\}$. Ea se aseamănă cu relația ce permite calculul coeficienților Fourier ai unui semnal periodic. Daubechies a arătat că pentru a exista o funcție undișoară $\{\psi(t)\}$, atunci trebuie să existe o altă funcție denumită funcție de scalare și care are notația $\varphi(t)$. Versiunile scalate ale acestei funcții sunt $\phi_j(t) = \phi(2^{-j}t)$. Orice funcție wavelet de la scara j poate fi generată ca o combinație liniară a funcțiilor de scara, de la scara $j-1$. Spre exemplu, o undișoară mamă de la scara 0 poate fi scrisă astfel:

$$\psi_0(t) = \sum_k a_k \varphi(2t - k)$$

O proprietate ce este cunoscută foarte bine atunci când discutăm despre funcții wavelet este cea că ele pot genera baze ortogonale ale lui $L^2(\mathcal{R})$. Pentru a putea demonstra că această proprietate este validă avem nevoie ca familia $\psi_{j,k}(t)$ să satisfacă două condiții, și anume:

1. Condiția de ortogonalitate
2. Să constituie o bază completă.

Pentru cea de-a doua condiție este necesar ca orice semnal din $L^2(\mathcal{R})$ să poată fi scris ca o combinație liniară a funcțiilor din familia de undișoare. Această proprietate din punct de vedere matematic poate fi formulată ca și în relația de mai jos:

$$\langle \psi_{j,k}, \psi_{m,n} \rangle = \begin{cases} 1, & \text{dacă } j = m \text{ și } k = n \\ 0, & \text{altfel} \end{cases}$$

Un caz particular ce poate fi evidențiat, această proprietate este valabilă și dacă vom restrânge la o singură scară de descompunere (adică $j=m$ în ecuația de mai sus). În această situație se poate spune că familia ce se obține prin translatarea în timp a undișoarei mamă $\Psi(t)$ și anume $\{\psi(t-k)\}_{k \in \mathbb{Z}}$ reprezintă o familie ortonormată.

Un exemplu practic ce a putut fi generat cu ajutorul programului Matlab este ilustrat în figura de mai jos pentru undișoara Daubechies-12 (corespondentă undișoarei Daubechies cu 6 momente nule):

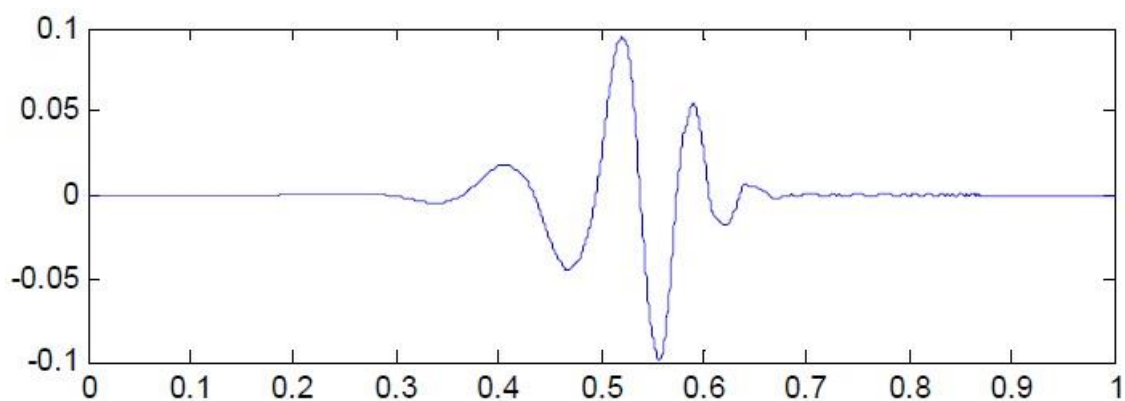


Figura 3.3 Undișoara mama Daubechies 12

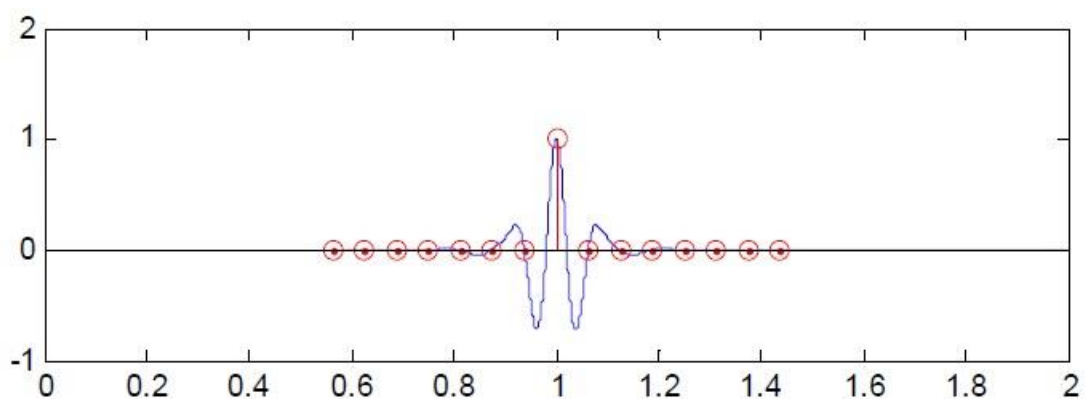


Figura 3.4 Funcția de autocorelație a unei undișoare mamă

Proprietatea de ortogonalitate a unor versiuni ce sunt translatate în timp generate de o funcție unică este specifică și funcțiilor de scară $\varphi(x)$, nu doar undișoarelor mamă.

Transformata Wavelet face parte din una din modalitățile naturale de a implementa un mod particular de reprezentare a semnalelor denumit analiză multirezoluție. Principalul aspect al acestei abordări constă în faptul că semnalul ce urmează a fi analizat este descris printr-o succesiune de aproximări ce conțin din ce în ce mai multă informație. Un exemplu clasic ce poate ilustra intuitiv acest principiu este următorul: privind din depărtare o pădure o putem caracteriza doar prin aspectul de ansamblu. Pe măsură ce parcurgem distanța care ne desparte de aceasta, începem să distingem copacii, apoi ramurile acestora, apoi frunzele. Acesta este de fapt specificul analizei de tip multirezoluție: fiecare nivel de aproximare conține pe de o parte întreaga informație ce a este disponibilă la nivelul anterior, la care se adaugă o componentă suplimentară de detaliu. Din punct de vedere matematic, analiza multirezoluție presupune să se aproximeze unele funcții prin proiecții succesive pe un ansamblu de spații vectoriale liniare care sunt incluse unele în altele.

În matematică, o serie wavelet este o reprezentare a unei funcții de o anumită serie ortonormată generată de un wavelet. O funcție este definită ca fiind un wavelet ortonormat dacă poate fi folosită pentru a defini o bază Hilbert, care este un sistem complet ortonormat, pentru spațiul Hilbert de funcții pătratic integrabile.

Principiul care se află la baza acestei transformări constă în faptul că transformarea produsă ar trebui să permită schimbări în domeniul timp, dar nu ar trebui să afecteze forma semnalului original.

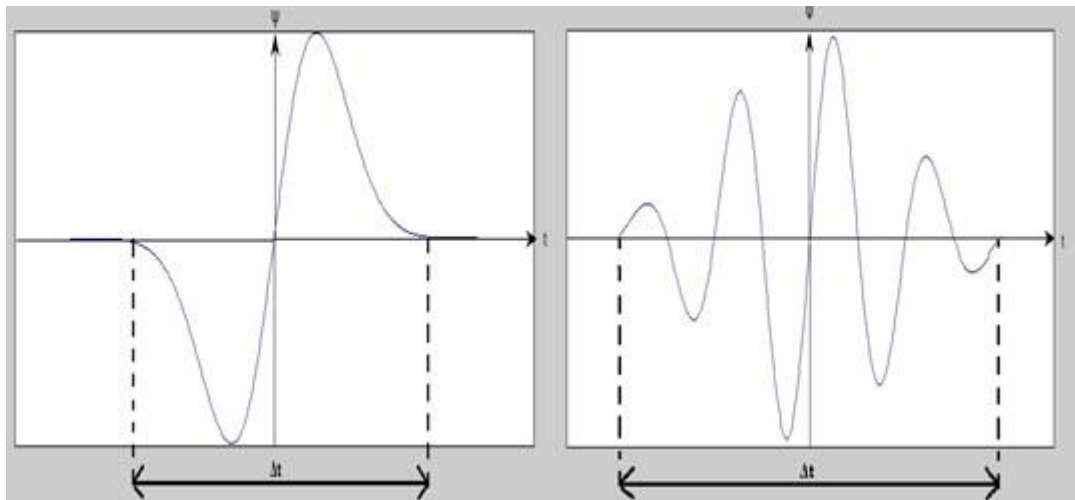


Figura 3.5 Comparație între factorii de compresie $c=1$ (stânga) și $c=1/4$ (dreapta) ai aceleiași funcții de bază

Acest lucru este afectat doar de alegerea funcțiilor de bază. Modificările ce au loc în domeniul timp sunt de așteptat să fie conforme cu frecvența de analiză corespunzătoare funcției de bază. Cu alte cuvinte, funcția de bază ψ poate fi considerată ca fiind răspunsul la impuls al unui sistem cu care a fost filtrată funcția $x(t)$, în cazul nostru fiind reprezentată de un semnal. Semnalul transformat oferă informații despre timp și frecvență. De aceea, transformata Wavelet conține informații similare cu cele obținute în cazul transformatei Fourier pe termen scurt (Short Time Fourier Transform), dar cu proprietăți speciale suplimentare ale transformatei wavelet, care apar la rezoluția în timp la frecvențe mai mari de analiză a funcției de bază. [1]

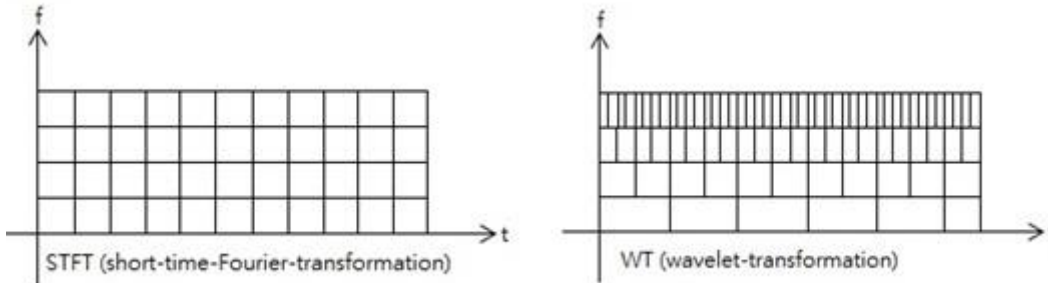


Figura 3.6 STFT vs. WT

Unul din avantajele folosirii analizei de tip Wavelet în comparație cu cea analiza Fourier este constituit din flexibilitatea alegerii unei funcții prototip „mother wavelet”. Proprietățile pe care trebuie să le avem în vedere pot face referire la caracterul compact în timp sau frecvență al funcției alese, la simetria formei de undă sau la caracterul ei lipsit de variații bruște. Pentru lucrarea prezentată, aspectele legate de micșorarea simultană a suportului (duratei) în domeniile timp și frecvență sunt considerate a fi esențiale (să ne amintim că un semnal wavelet trebuie să reprezinte totuși o undă „mică”), iar acestea sunt asigurate de așa-numitele condiții de regularitate.

Putem face o comparație între metodele existente, în acest moment, ce pot fi aplicate pentru a extrage diferite trăsături din interiorul unui semnal ca în figura de mai jos.

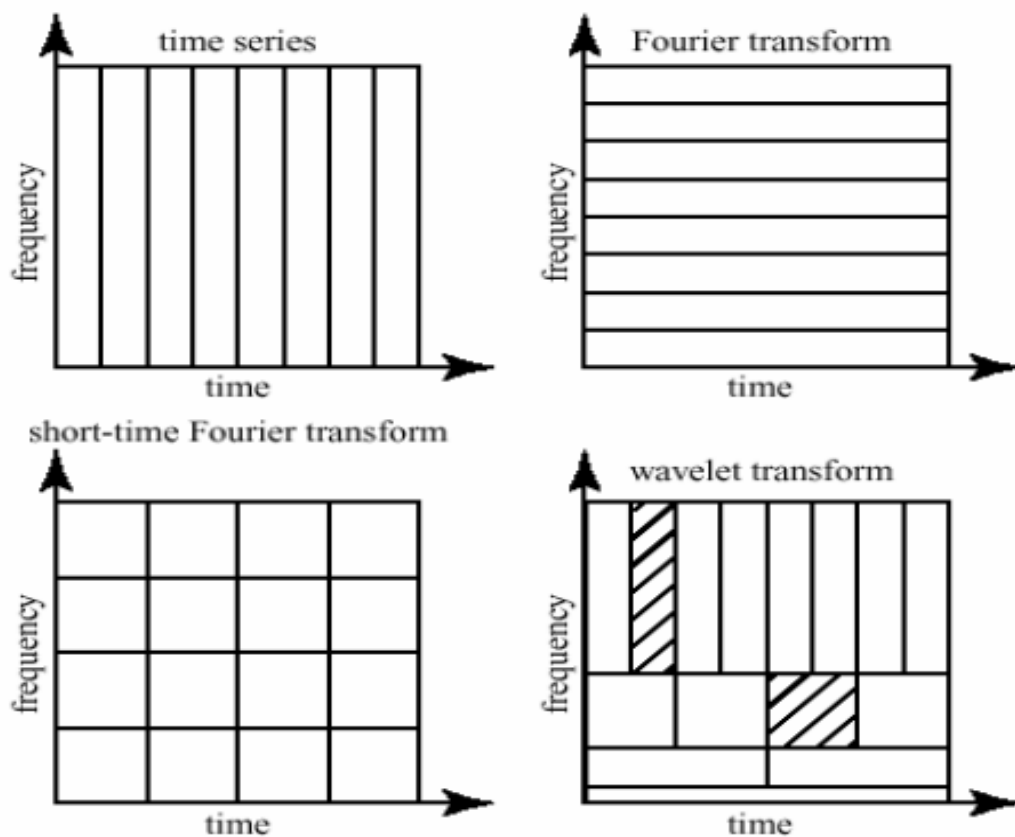


Figura 3.7 Comparație între transformatele Fourier și Wavelet

3.2 INFLUENȚA NUMĂRULUI DE MOMENTE NULE

Se poate aprofunda mai departe analiza undișoarelor folosite. Luând de această dată în considerare un alt parametru al transmisiei, parametru reprezentat de numărul de momente nule. În acest sens, o selecție a rezultatelor obținute pentru toate undișoarele este aratăată în tabelul imediat următor.

Tabel 3.1 Selecție a performanțelor BER pentru $f_m=0.005$ și o singură iterație IDWT

Eb/No	0	2	4	6	8	10	12	14	16	18	20
Haar	0.1255	0.0879	0.0580	0.0366	0.0218	0.0127	0.0072	0.0041	0.0025	0.0016	0.0011
Daub4	0.1466	0.1086	0.0776	0.0536	0.0361	0.0241	0.0160	0.0109	0.0075	0.0053	0.0040
Daub8	0.1468	0.1088	0.0774	0.0535	0.0363	0.0244	0.0166	0.0115	0.0082	0.0063	0.0051
Daub12	0.1466	0.1086	0.0775	0.0537	0.0365	0.0248	0.0170	0.0122	0.0093	0.0073	0.0061
Daub16	0.1465	0.1086	0.0775	0.0536	0.0368	0.0251	0.0178	0.0131	0.0100	0.0082	0.0071
Daub20	0.1464	0.1085	0.0773	0.0539	0.0370	0.0256	0.0183	0.0137	0.0109	0.0091	0.0081
Symmlet4	0.1469	0.1088	0.0775	0.0536	0.0362	0.0242	0.0162	0.0110	0.0077	0.0056	0.0043
Symmlet6	0.1468	0.1088	0.0777	0.0538	0.0362	0.0243	0.0164	0.0113	0.0081	0.0061	0.0048
Symmlet8	0.1467	0.1088	0.0776	0.0536	0.0365	0.0245	0.0166	0.0116	0.0083	0.0064	0.0053
Symmlet10	0.1467	0.1088	0.0777	0.0536	0.0365	0.0245	0.0168	0.0118	0.0087	0.0067	0.0055
Coiflet1	0.1467	0.1089	0.0776	0.0537	0.0363	0.0242	0.0163	0.0110	0.0078	0.0057	0.0044
Coiflet2	0.1467	0.1089	0.0775	0.0537	0.0362	0.0242	0.0162	0.0110	0.0078	0.0057	0.0044
Coiflet3	0.1468	0.1087	0.0776	0.0538	0.0362	0.0244	0.0165	0.0113	0.0080	0.0060	0.0047
Coiflet4	0.1467	0.1089	0.0776	0.0537	0.0364	0.0245	0.0167	0.0115	0.0083	0.0064	0.0051
Coiflet5	0.1465	0.1086	0.0777	0.0536	0.0364	0.0245	0.0167	0.0117	0.0085	0.0066	0.0054

Referitor la influența numărului de momente nule, ea este evidentă în cazul undișoarelor din familia Daubechies, confirmând că undișoarele ce sunt mai bine localizate în timp oferă performanțe mai bune în cazul transmisiei printr-un canal variant în timp cu fading plat.

Ca o vedere per ansamblu performanța cea mai bună (exceptând undișoara Haar) la un anumit RSZ este marcată prin intermediul unui fundal gri. Se observă din tabel că, făcând excepția mai sus amintită, pe locul al doilea putem regăsi undișoara Daubechies 4, care este implementată prin intermediul unor filtre cu 4 coeficienți. Această diferențiere este importantă la valori mari ale RSZ, unde se poate presupune că ponderea cea mai importantă a erorilor este dată de împrăștierea Doppler.

3.3 SCURTĂ INTRODUCERE DESPRE NOȚIUNEA DE PLÂNSETE ALE NOU-NĂSCUȚILOR ȘI STUDIUL REALIZATE

Așa cum am specificat până acum în lucrarea prezentată mă voi dedica analizei semnalelor produse de nou-născuți. Sunetele produse de oamenii maturi cu ajutorul aparatului fonator, sunt produse diferit de către nou-născuți, dar conțin informație utilă. În momentul nașterii aceștia nu cunosc altă formă de exprimare decât cea a plânsetului.

Din totdeauna s-a dorit ajutarea părinților pentru a se înțelege mai ușor necesitățile nou-născutului, acest lucru ducând la realizarea unor studii și a unor sisteme care să recunoască automat nevoia care se află în spatele plânsetului nou-născutului, încă din anii '60. Unele exemple pentru acest sistem se referă, de asemenea, și la recunoașterea facială și interpretarea acesteia în funcție de nevoia exprimată de bebeluș. Un alt tip de sistem de acest fel este acela în care se recunoștea tipul plânsetului de durere. Alte sisteme ce au fost dezvoltate sunt cele care recunosc trei tipuri de plânsete diferite: de durere, de foame și frică, folosind pentru determinarea acestora trei tipuri de rețele neuronale. De actualitate sunt și experimentele ce au scos la iveală faptul că se poate determina dacă un nou-născut suferă de boli patologice, aceste probleme putând fi descoperite într-un timp scurt și altfel ajutând la soluționarea problemelor

cât mai rapid. De exemplu la întâlnirea unor tipuri de plânsuri ce sunt răgușite se poate lua în calcul și faptul că bebelușul poate suferi de o boală patologică (bebelușii ce suferă de sindromul Down au un plâns specific ce le poate indica medicilor pediatri să recunoască această anomalie și astfel să ia măsuri mult mai rapide, față de momentul în care acest lucru s-ar descoperi la o vârstă mai înaintată). Medicii pediatri, persoane care au experiență în acest domeniu, pot diferenția aceste tipuri de plânsuri, în fiecare tip existând un tipar diferit.

Un plâns este alcătuit din patru tipuri de sunet: sunetul care se produce în faza de inspirație urmat de o pauză și un sunet produs în etapa de expirație care la rândul său este urmat de o pauză. Durata momentelor de inspirație, expirație și momentele de pauză pot să difere de la bebeluș la bebeluș. Plânsul propriu-zis se produce în faza de expirație.

Plânsul bebelușilor face parte din clasa semnalelor staționare pe timp scurt ca și în cazul vorbirii adulților. Aceste semnale pot fi considerate mai staționare decât cele produse de adulți deoarece aceștia nu au control atât de bun asupra aparatului fonator. Pentru a se putea caracteriza și recunoaște aceste tipuri de plânsuri a fost nevoie de introducerea unor noi trăsături, în afară de cel clasice: timpul și frecvența. Aceste noi mărimi sunt: latența, durata componentei expiratorii, intervalul dintre plânsuri, valoarea de vârf a frecvenței fundamentale, amplitudinea componentei, frecvența dominantă, amplitudinea frecvenței dominante și amplitudinea maximă globală.

Abordarea care se folosește în cazul recunoașterii plânsurilor nou-născuților este cea care se aplică și în cazul recunoașterii de vorbitor, și anume considerăm că fiecare tip de plâns face parte dintr-o clasă diferită.

3.4 ANALIZA SEMNALELOR EXISTENTE ÎN BAZA DE DATE

După mai multe studii efectuate de diverse centre de cercetare printre care numărăm și studiul companiei australiene Dunstan Baby Language condusă de Perscilla Dunstan care se referă la informațiile utile ce pot fi extrase dintr-un semnal de tip plâns, se poate constata că nou-născutul își exprimă de fapt o stare fizică sau emoțională sau anumite nevoi. Perscilla Dunstan a creat un tutorial video prin care urmărește să îi ajute pe părinții bebelușilor să identifice cele cinci nevoi de bază ale acestora: foame, disconfort, eructație, flatulență, somn. Această codificare a fost realizată de Perscilla, care încă din frageda copilărie putea să reproducă compoziții muzicale pe care le auzise o singură dată și avea talentul de a găsi anumite tipare în diferite sunete.

Baza de date Dunstan conține optzeci și trei de fișiere ce aparțin unui număr de 37 de bebeluși. Fișierele conținute în această bază de date nu sunt egale ca și durată, iar numărul de bebeluși este diferit de la fișier la fișier, distribuit în următoarea formă:

- Clasa „EAIRH” conține plânsuri ce reprezintă durerea de burtă aparținând unui număr de 12 bebeluși;
- Clasa „EH” conține plânsuri pentru eructație aparținând unui număr de 14 bebeluși;
- Clasa „NEH” conține plânsuri pentru foame aparținând unui număr de 25 bebeluși;
- Clasa „HEH” conține plânsuri pentru disconfort aparținând unui număr de 13 bebeluși;
- Clasa „OWH” conține plânsuri pentru somn aparținând unui număr de 19 bebeluși.

3.5 FAMILII DE WAVELET-URI

Selectarea unei baze wavelet optimale

Punând accent încă o dată pe avantajul flexibilității în selectarea unei funcții prototip de tip wavelet și, respectiv, a bazei generate de aceasta, este de menționat că avem posibilitatea de a alege după criteriul eficienței unui semnal dat în raport cu o astfel de bază. Prin eficiență înțelegem proprietatea ca în descompunerea semnalului ce trebuie analizat să fie un număr cât mai redus de coeficienți ce trebuie să conțină informație utilă, restul putând fi neglijați.

În urma acestor studii, pentru cazul plânsurilor nou-născuților unde am observat proprietăți asemănătoare cu cele ale semnalelor din cadrul transformatelor wavelet, am analizat pentru experimente următoarele familii de wavelet-uri ce vor fi descrise în paragrafele ce urmează.

Familia Daubechies(,db')

Familia de wavelet-uri Daubechies au fost descoperite de Ingrid Daubechies și sunt o familie ortogonală de wavelet-uri care definesc o transformare discretă wavelet ce este caracterizată printr-un număr maxim al momentelor de dispariție al unor puncte suport. Cu fiecare tip de wavelet ce aparține acestei familii există o funcție de scalare (numită „funcție wavelet tată”) care generează o analiză ortogonală multirezoluție.

Această familie de wavelet-uri este aleasă cu un număr de momente de dispariție A , cât mai mare (acestea nu implică neapărat o aproximare mai bună) pentru suportul dat, de lățime $2A-1$. Din cele 2^{A-1} soluții posibile ale ecuațiilor algebrice pentru momentele de dispariție și condițiile de ortogonalitate, soluția aleasă este cea a cărei filtru de scalare are faze extremale. Această transformată este utilizată în sens larg pentru rezolvarea unei extinse game de probleme, spre exemplu, proprietăți de auto-similaritate ale unui semnal sau probleme fractale, discontinuități de semnal, etc.

Daubechies wavelet ortogonale D2-D20 respectiv d1-d10, sunt frecvent utilizate. Numărul de index se referă la numărul N de coeficienți. Fiecare wavelet are un număr de la zero momente de dispariție până la jumătate din numărul de coeficienți. De exemplu, D2(Haar Wavelet) are un moment de dispariție, D4 are două momente de dispariție, etc.

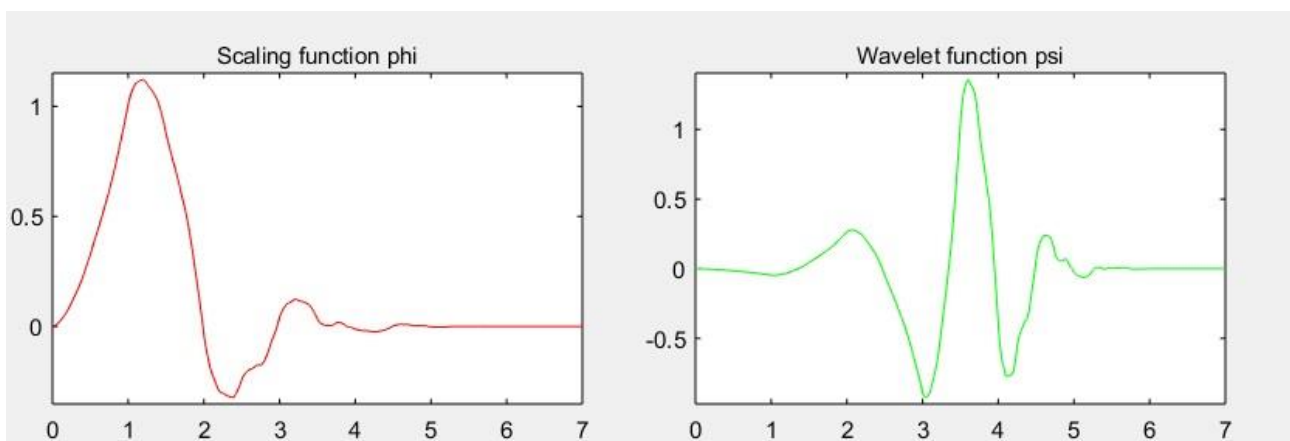


Figura 3.8 Familia Daubechies – funcția de scalare și funcția Wavelet corespunzătoare

Familia Biortonormal 3.5(,bior 3.5')

Un wavelet biortonormal este un wavelet ce are asociată o transformată wavelet ce poate fi inversabilă, dar nu este neapărat să fie și funcție ortogonală. Această familie de wavelet-uri permite folosirea unor grade de libertate mult mai variate decât cele ortonormale. Un grad de libertate ce poate fi adăugat este cel în care putem construi funcții wavelet simetrice.

În acest caz vom discuta despre două funcții de scalare ϕ și $\tilde{\phi}$, în loc de una ca în cazul celor de tip ortonormal. Aceste două funcții pot genera mai multe tipuri de analiză multirezoluție și, în consecință cu acest lucru, două funcții wavelet diferite, deci și coeficienții Celor două funcții, M și N pot fi diferiți. În consecință cu cele scrise mai sus, cele două funcții de scalare trebuie să îndeplinească condiția următoare de ortonormalitate:

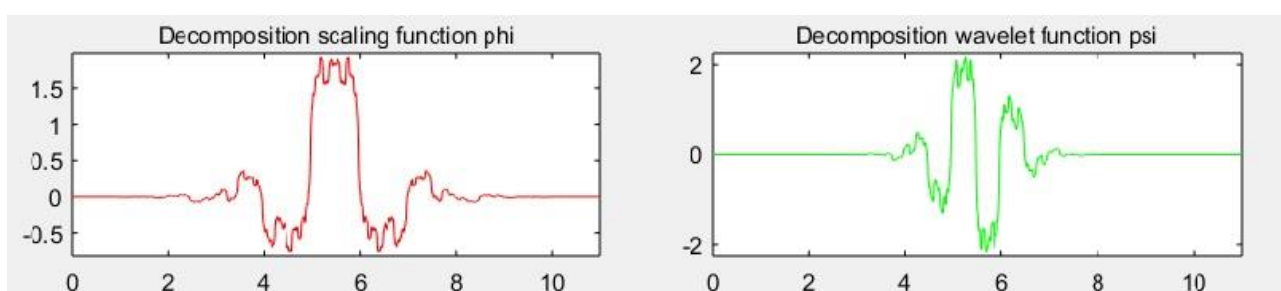


Figura 3.9 Familia Biortonormal – funcția de scalare și funcția Wavelet corespunzătoare.

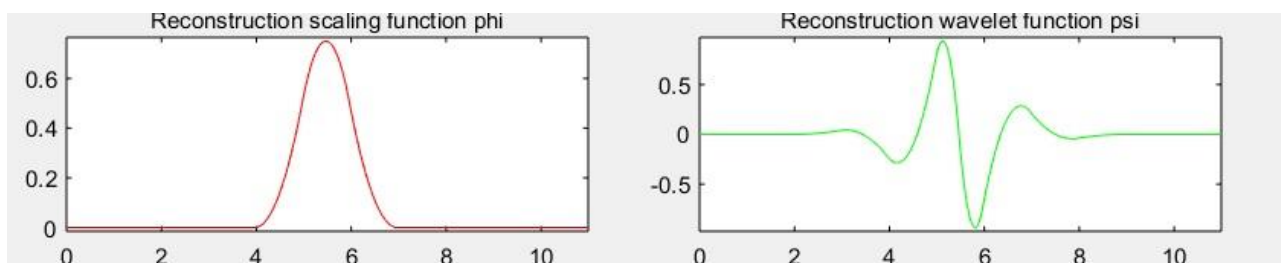


Figura 3.10 Familia Biortonormal – funcția de scalare și funcția Wavelet corespunzătoare reconstrucției

Familia Symlet(,sym')

Această familie de wavelet-uri este o versiune modificată a familiei Daubechies ce are ca și proprietate simetria.

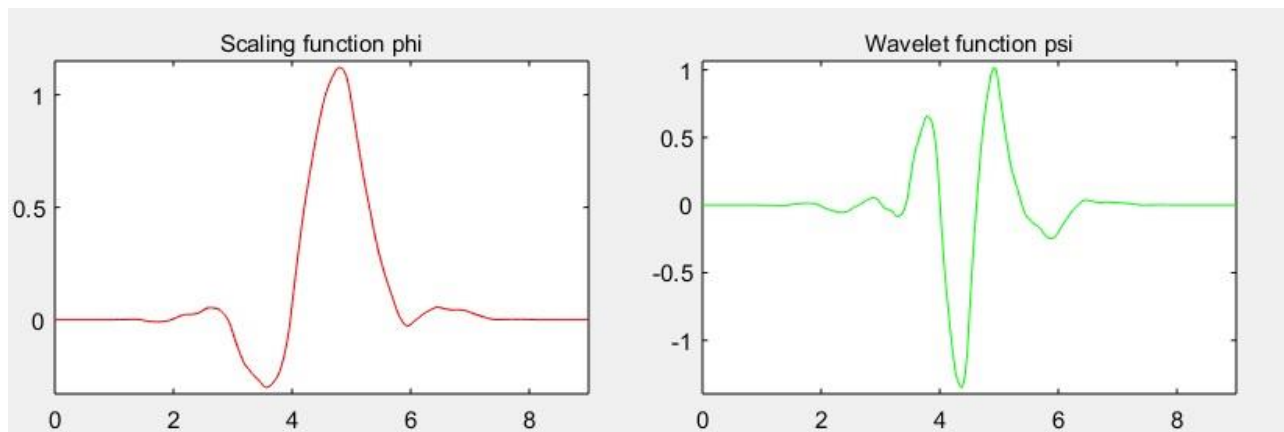


Figura 3.11 Familia Symlet – funcția de scalare și funcția Wavelet corespunzătoare reconstrucției.

Familia Coiflet(,coif^o)

Această familie a fost concepută de Ingrid Daubechies la cererea lui Ronald Coifman din dorința de a avea funcții de scalare cu momente estompate. Funcția sa wavelet este aproape simetrică, având $N/3$ momente de dispariție, iar ca și număr de funcții de scalare regăsim $N/3-1$.

Regăsim folosirea acestei familii de wavelet-uri și alături de operatorii Calderón-Zygmund.

Pentru cazul aferent plânsetelor nou-născuților am selectat această familie de wavelet-uri pentru trei, respectiv cinci momente de dispariție, așa cum ilustrează și figurile de mai jos.

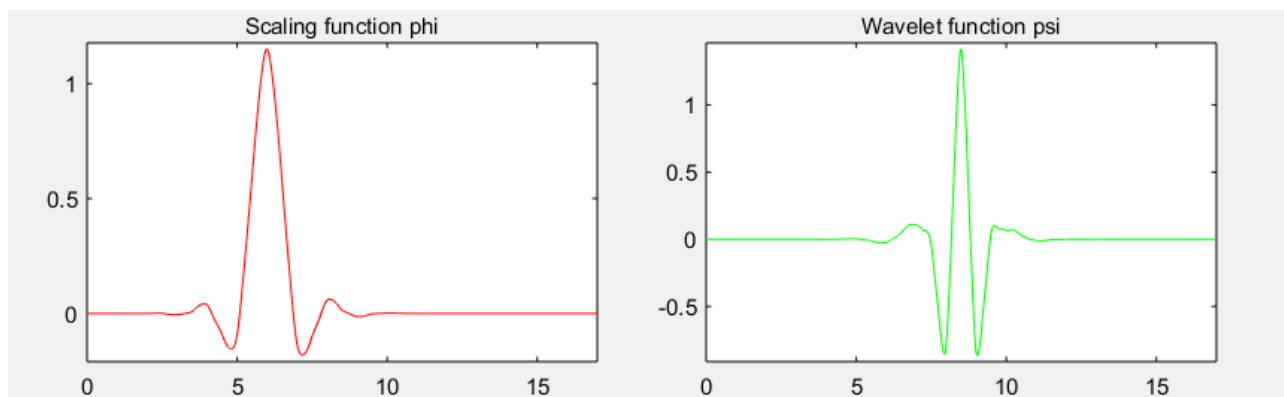


Figura 3.12 Familia Coiflet 3– funcția de scalare și funcția Wavelet corespunzătoare reconstrucției.

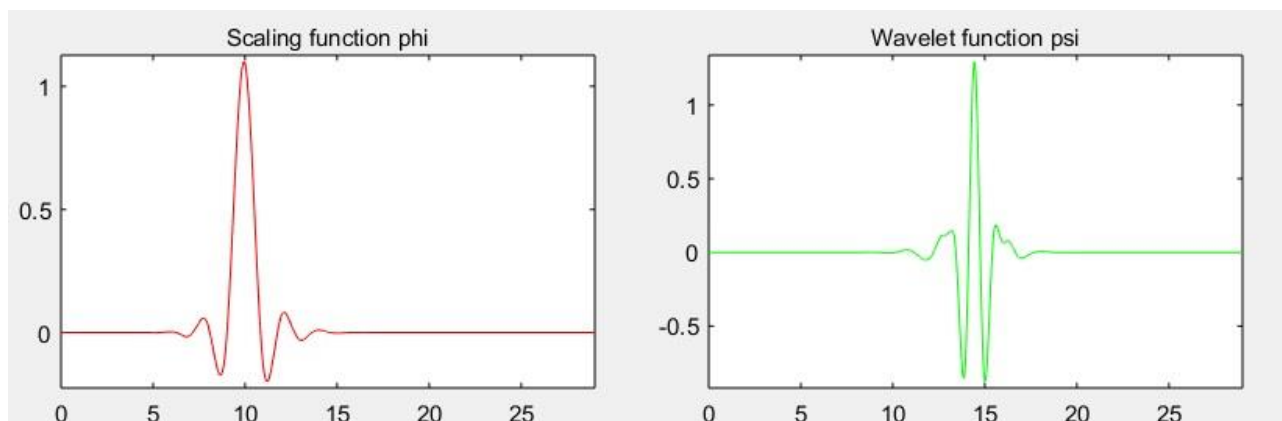


Figure 3.13 Familia Coiflet 5 – funcția de scalare și funcția Wavelet corespunzătoare reconstrucției

Familia Discrete Meyer('dmey')

Apar ca și descoperiri după anul 1980 și au ca și scop o abordare diferită și anume: nu au suport de contact și sunt continuu diferentiabile.

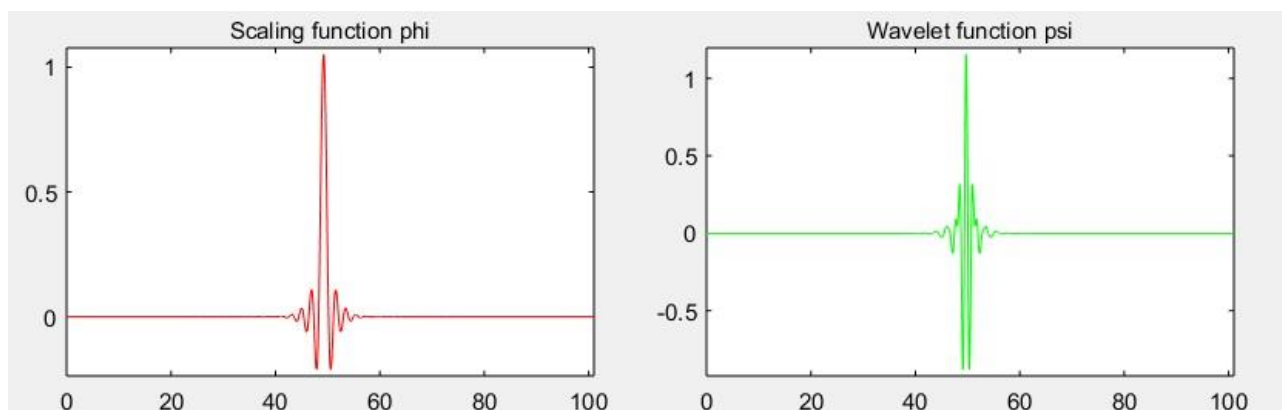


Figure 3.14 Familia Discret Meyer – funcția de scalare și funcția Wavelet corespunzătoare reconstrucției.

3.6 IMPLEMENTARE MATEMATICĂ – ANALIZA WAVELET A PLÂNSETELOR NOU-NĂSCUȚILOR(METODA I)

Atunci când ne dorim să implementăm toate aceste concepte pe calculator, transformata wavelet indicată prin formula (2) impune o serie de constrângeri specifice. Principala constrângere manifestată față de versiunea continuă a transformării wavelet este constituită de redundanța sa ridicată. Pentru a scoate în evidență acest lucru, ilustrăm exemplul unui semnal de intrare de N eşantioane pe care dorim să aplicăm transformata Wavelet continuă pe un număr de scări. Rezultatul obținut va fi o matrice de $N \times S$ numere(coeficienți wavelet), ceea ce face această transformată să fie dificil de aplicat în multe din situațiile practice. Transformata Wavelet

Discretă(DWT), pe de altă parte, oferă suficientă informație pentru a analiza semnalul, respectiv sintetiza.

Termenul de transformată wavelet este în strânsă legătură cu acela de analiză multi-rezoluție. Ideea pe care este bazată analiza multi-rezoluție este aceeași care stă la baza transformării wavelet continue, respectiv obținerea unei reprezentări timp-scară a semnalului, de această dată folosind tehnici de filtrare digitală. Atunci când vorbim despre transformata wavelet discretă vorbim de fapt despre o serie de filtre cu diverse caracteristici ce sunt aplicate pentru a examina semnalul la diferite stări de decompoziție. Astfel, semnalul este trecut printr-o serie de filtre trece-sus pentru analiza frecvențelor înalte din semnal, iar pe de altă parte, printr-o serie de filtre trece-jos care permit analiza frecvențelor joase din semnal.

Rezoluția semnalului(aici fiind reprezentată printr-o măsură a nivelului de detaliu care poate fi analizat în semnalul respectiv) este schimbată prin aceste seturi de operații de filtrare. Din alt punct de vedere, scara este schimbată prin operații de sub-eșantionare(în cazul analizei), respectiv supra-eșantionare(în cazul în care discutăm despre sinteză). Sub-eșantionarea unui semnal se referă la reducerea ratei de eșantionare(coeficienții sunt distribuiți mai rar pe axa timpului) și se realizează prin înlăturarea unor eșantioane din semnal de la o anumită rezoluție. Supra-eșantionarea, pe de altă parte, corespunde creșterii ratei semnalului prin adăugarea de noi eșantioane în acesta.

Coeficienții DWT corespund unei eșantionări a transformării continue, eșantionare care se produce pentru cele două variabile: pentru variabila de scară, cât și pentru variabila de poziție. Spre deosebire de ceea ce avem ilustrat în formula (2) trebuie specificat că pentru o implementare pe calculator semnalul de analizat va fi unul discretizat. Schema de implementare a DWT directă va fi ilustrată în figura imediat următoare. Este ilustrată implementarea de bancuri de filtre propusă de Mallat. Procedura este inițiată prin filtrarea semnalului cu ajutorul unui filtre digital trece-jos, $g[n]$ care are lățimea de bandă $\pi/2$. Reamintim că frecvența maximă din spectrul unui semnal în timp discret este π , frecvență ce ar corespunde jumătății frecvenței de eșantionare specifice unui semnal analogic.

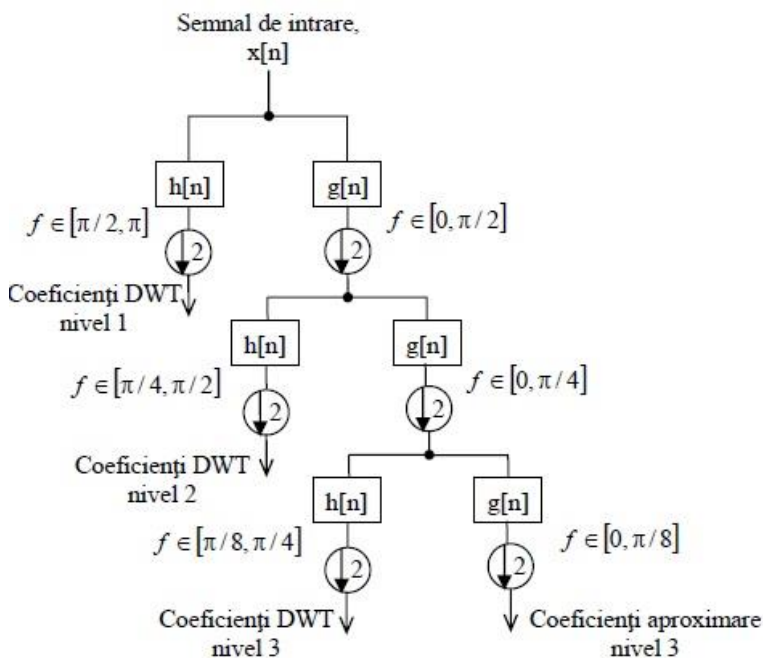


Figure 3.15 Implementarea DWT folosind bancuri de filtre

Filtrarea ce se efectuează corespunde efectuării unei operații de convoluție a semnalului cu răspunsul la impuls al filtrului. După ce se aplică filtrul are loc o decimare a ieșirii filtrului cu 2, care nu afectează informația semnificativă, deoarece în acest moment banda semnalului discret

este de doar $\pi/2$ ceea ce reprezintă jumătate din banda originală. Scara semnalului este acum dublul celei originale, sau altfel spus s-a obținut un anumit nivel de aproximație a semnalului. Este de remarcat că în acest caz filtrarea care s-a realizat a eliminat doar conținutul de înaltă frecvență, fără a duce la o modificare propriu-zisă a scării, care este realizată prin procesul de sub-eșantionare(decimare).

Privind lucrurile dintr-o altă perspectivă, rezoluția semnalului este legată de cantitatea de informație care o poate conține și este prin urmare afectată de filtrarea care se realizează. Se poate afirma că eliminând jumătate din bandă, se pierde jumătate din informația pe care semnalul o conținea, deci și rezoluția s-a înjumătățit. Pentru o imagine de ansamblu clară trebuie precizat faptul că sub-eșantionarea nu diminuează rezoluția. Operația anterioară de filtrare a avut ca rezultat un semnal supra-eșantionat, în care jumătate din eșantioane sunt redundante.

Procedura de filtrare și decimare poate fi exprimată și prin intermediul următoarei relații:

$$y[n] = \sum_k h[k]x[2n - k]$$

Odată cu precizarea acestor lucruri, vom urmări în continuare, pas cu pas, modalitatea matematică în care este calculată DWT. Trebuie luat în calcul faptul că DWT realizează analiza semnalului în diferite benzi de frecvență cu rezoluții diverse, prin descompunerea acestuia în informație (coeficienți) de aproximare, respectiv de detaliu. Pentru a realiza ceea ce am descris adineauri, DWT utilizează două seturi de funcții despre care s-au făcut precizări mai sus(funcții de scară și undișoare), care sunt asociate cu filtrele trece-jos, respectiv trece-sus. Răspunsurile la impuls ale acestor filtre sunt $g[n]$, respective $h[n]$. Descompunerea semnalului în diverse subbenzi este obținută prin aceste operații, aplicate succesiv, de filtrare trece sus și trece jos, ilustrate în relațiile imediat următoare:

$$y_{TS} = \sum_n x[n]h[2k - n]$$

$$y_{TJ} = \sum_n x[n]g[2k - n]$$

Cu fiecare iterație realizată, rezoluția temporară devine mai slabă, iar cea frecvențială devine mai bună, așa cum este ilustrat și în figura de mai sus.

O altă proprietate importantă folosită în cazul transformatei DWT este că răspunsurile la impuls ale filtrelor folosite sunt independente, ele fiind corelate prin relația:

$h[L - 1 - n] = (-1)^n g[n]$, unde L reprezintă lungimea în eșantioane a răspunsului la impuls al filtrelor. Conversia de la caracteristica trece-jos la cea trece-sus este redată prin factorul $(-1)^n$.

În acest caz reconstrucția este ușor de aplicat, deoarece folosirea acestor filtre duce la formarea unor baze ortonormale. Fiecare semnal poate fi interpretat ca o combinație liniară a componentei sale trece-sus, respectiv trece-jos. La fiecare iterație ce se realizează aplicând transformata inversă pentru reconstrucție, semnalul este supra-eșantionat și trecut prin filtrele de sinteză. Formula pentru fiecare nivel în parte, în cazul reconstrucției, este ilustrată mai jos:

$$x[n] = \sum_k y_{TS}[k]h[2k - n] + y_{TJ}[k]g[2k - n]$$

Dacă vom analiza mai atent filtrele cu care se face reconstrucția vom realiza că sunt asemănătoare cu cele folosite pentru descompunere, fiind de fapt o versiune reflectată în timp a acestora:

$$h_1[n] = h^*[-n]$$

$$g_1[n] = g^*[-n].$$

Fiind date operațiile de supra sau sub-eșantionare cu 2, o implementare facilă a transformării se poate face atunci când semnalul căruia i se aplică transformata are un număr de eșantioane de 2^m . În această situație se pot aplica un număr maxim de m iterații, deoarece pentru fiecare nivel de descompunere numărul coeficienților ce se obțin este de două ori mai mic decât la scara precedentă, ajungându-se astfel la situația în care, dacă numărul de iterații este maxim, la scara cea mai brută de aproximare să avem un singur coeficient.

În figura de mai jos este ilustrată reconstrucția pentru 3 nivele de descompunere. Filtrele de reconstrucție sunt notate cu $g_1[n]$, respectiv $h_1[n]$.

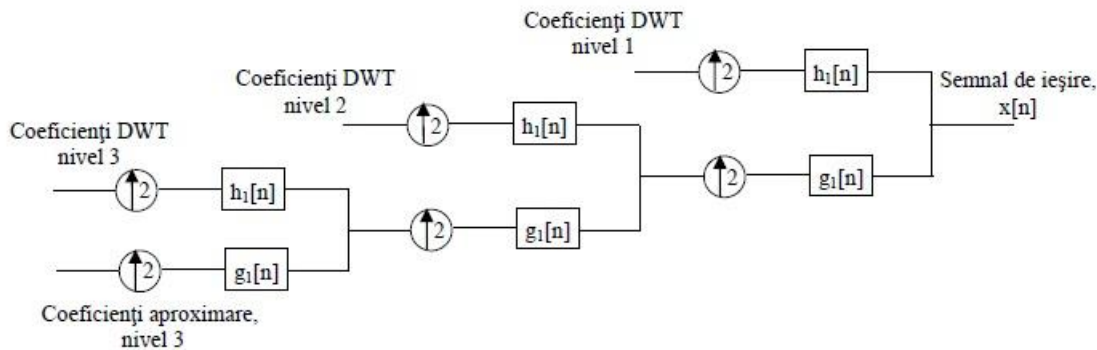


Figure 3.16 Implementarea IDWT folosind bancuri de filtre

În cazul ideal, reconstrucția se poate realiza exact dacă cele două filtre, trece-sus și trece-jos sunt ideală. Cu toate că impementarea unor filtre ideale este destul de dificilă, există posibilitatea ca în anumite condiții să se definească filtre care să permită o reconstrucție cât mai exactă a semnalului. Unele din cele mai cunoscute asemenea filtre au fost cele descoperite de Ingrid Daubechies.

Pentru că interpretarea coeficienților rezultați în urma aplicării transformatei wavelet discrete(DWT) poate fi destul de dificilă, o vom analiza printr-un exemplu, datorită importanței acestui aspect asupra subiectului principal, și anume utilizarea funcțiilor wavelet.

Bazându-ne pe figurile 0.14 și 0.15 ilustrate anterior, vo presupune că avem de analizat un semnal cu 1024 de eșantioane, eșantionat cu 10MHz. Așa cum am precizat într-un paragraf anterior fiind vorba despre această frecvență de eșantionare, frecvența maximă a semnalului de analizat va fi de 5 MHz(frecvența de eșantionare trebuie să fie dublul frecvenței maxime a semnalului de analiză). Atunci când aplicăm transformata wavelet, după prima iterație a acesteia se obțin 512 eșantioane la ieșirea din filtrul trece-sus, eșantioane ce sunt denumite aici coeficienți wavelet ai primului nivel. Ei vor reprezenta informația ce este conținută în banda [2.5, 5] MHz. La ieșirea din filtrul trece-jos(FTJ) se obțin 512 coeficienți de aproximare, ce corespund benzii [0,2. 5] MHz, care vor fi supuși în continuare descompunerii. Acest procedeu se repetă asupra acestor coeficienți, obținându-se detaliile la cea de-a doua iterație și coeficienți de aproximare, iar la cea de-a treia iterație vom obține în final un set de alți 128 de coeficienți de detaliu, respectiv aproximare. Acești coeficienți de aproximare vor aparține benzii [0,622.5]kHz. Rezultatul care se obține va fi un vector de coeficienți concatenați astfel: {A3(128), D3(128), D2(256), D1(512)}. Se poate observa încă o dată că folosirea transformatei wavelet discretă nu este redundantă, celor 1024 de eșantioane le vor corespunde tot 1024 de coeficienți de ieșire. Rămâne valabilă afirmația făcută în paragrafele de mai sus, în care susțineam că la fiecare iterație se reduce numărul de eșantioane(scade rezoluția temporală), dar se îmbunătățește banda(se îmbunătățește rezoluția frecvențială).

În cadrul primei metode ne-am propus calculul a cinci energii. Pentru fiecare nivel obținut în partea de descompunere noi calculăm energiile corespunzătoare astfel:

- dacă coeficienții pentru care se realizează calculul energiilor aparțin primului nivel atunci vom reține coeficienții aferenți acestui nivel și vom realiza ridicarea la pătrat, după care ne vom deplasa în interiorul vectorului L pentru a reține coeficienții aferenți următorului nivel.

- dacă coeficienții se află pe alt nivel atunci vom mai avea o variabilă care ne va memora poziția din vectorul L pentru a vedea nivelul pentru care vom calcula energia corespunzătoare. Această variabilă va porni inițial de la valoarea 0.

În final vom obține o matrice ce va avea dimensiunea de 83x6, unde prima coloană va fi reprezentată de label-ul asociat fiecărui tip de semnal în parte, urmată de cinci coloane corespunzătoare energiilor calculate (patru energii pentru cazul coeficienților de detaliu și una pentru cazul coeficientului de aproximație aferentă nivelului 4). Matricea va arăta de forma:

83x6 double

	1	2	3	4	5	6
1	1	0.0082	0.0158	0.2209	0.4728	0.2823
2	1	0.0071	0.1947	0.3492	0.2275	0.2215
3	1	0.0509	0.4658	0.4193	0.0424	0.0217
4	1	0.0014	0.0103	0.1766	0.5069	0.3049
5	1	0.0460	0.3271	0.3092	0.2294	0.0882
6	1	0.0100	0.5266	0.3182	0.0957	0.0495
7	1	0.0244	0.5090	0.2697	0.1231	0.0738
8	1	0.0079	0.1130	0.3090	0.4039	0.1662
9	1	0.0487	0.3013	0.2310	0.3491	0.0699
10	1	0.1698	0.1798	0.3374	0.2440	0.0690
11	1	0.0175	0.4296	0.4957	0.0441	0.0131
12	1	0.0189	0.0412	0.4321	0.3556	0.1522
13	2	0.2326	0.4205	0.1925	0.1251	0.0293
14	2	0.1384	0.6846	0.0988	0.0681	0.0101
15	2	0.0088	0.1201	0.2132	0.2374	0.4205
16	2	0.1331	0.6674	0.1366	0.0424	0.0204
17	2	0.0259	0.4579	0.4064	0.0920	0.0178
18	2	0.0676	0.3808	0.4484	0.0806	0.0226
19	2	0.1000	0.2050	0.3208	0.3191	0.0551
20	2	0.0702	0.4267	0.3662	0.0834	0.0535
21	2	0.0846	0.4055	0.2788	0.1726	0.0586
22	2	0.0569	0.4286	0.3374	0.1255	0.0515
23	2	0.2429	0.3207	0.1504	0.2104	0.0756
24	2	0.0956	0.3832	0.2216	0.1989	0.1007
25	2	0.0337	0.3313	0.2994	0.1446	0.1910
26	2	0.0337	0.3313	0.2994	0.1446	0.1910
27	3	0.2205	0.4619	0.2770	0.0259	0.0148

Figure 3.17 Matricea de parametri corespunzătoare primei metode

CAPITOLUL 4

EXTRAGEREA DE TRĂSĂTURI ROBUSTE PENTRU CLASIFICAREA PLÂNSETELOR NOU- NĂSCUȚILOR (METODA II)

În acest capitol ne propunem să expunem o nouă metodă de extragere a coeficienților cu care urmează a fi antrenat modelul SVM și ulterior testat, pentru baza de date existentă în cadrul acestui proiect. De asemenea, vom aborda și câteva lucruri despre coeficienții de aproximație și de detaliu obținuți cu ajutorul transformatei Wavelet, cât și parametrii calculați după ce au fost extrași coeficienții.

4.1 TRANSFORMATA WAVELET PENTRU CAZUL PROCESĂRII DE SEMNALE

Bazat pe analiza multirezoluție, un semnal este descompus în coeficienți de aproximație și de detaliu la diferite scări. Cu alte cuvinte, pot fi extrase informații de la nivele diferite prin descompunerea semnalului inițial folosind o bază ortonormată de unde. Ca o transformare convențională se utilizează transformata Fourier pe termen scurt (STFT) atât în domeniul matematicii, cât și cel al ingineriei. O limitare a acestei transformate este că, chiar dacă are o singură fereastră utilizată pentru toate frecvențele, analiza rezoluției este aceeași la toate locațiile

în planul frecvenței temporale. Din cauză că sistemul nostru auditiv utilizează o frecvență dependentă de rezoluție, limitarea transformatei reprezintă un handicap pentru semnalele audio. DWT poate rezolva acest dezavantaj cu planul timp-frecvență dreptunghiular.

Din punct de vedere matematic este necesar să se cerceteze singularitatea structurilor neregulate deoarece conțin informații utile în semnale (spre exemplu semnale discontinue, nestăționare, etc). Cea mai mare provocare este selectarea unei tehnici adecvate care să ne permită studierea neregularităților structurilor de semnale.

Până acum transformata Fourier a constituit principalul instrument matematic pentru analiza acestor singularități. Cu toate acestea, transformata Fourier cu forme de undă sinusoidale ce se întinde pe o lungime fixă oferă doar o descriere a regularității globale a semnalelor fără a se adapta bine la localizarea singularităților din domeniul frecvenței temporale. Acest fapt motivează studierea transformatei wavelet care poate realiza caracterizarea regularității locale de semnale prin descompunerea semnalelor în componente bine-localizate în timp. Așa cum s-a demonstrat în [MH92], se bazează pe detectarea tuturor singularităților semnalului cu proprietatea maximului local care este măsurat de-a lungul evoluției scării acestor maxime locale. Detectarea acestor singularități au fost studiate nu numai în matematică, ci și în domeniul procesării de semnale și al aplicațiilor acestuia.

Analiza flexibilă în planul timp-frecvență a DWT ne ilustrează avantajele pe care le are față de STFT, în vederea prelucrării semnalelor. Cu ajutorul principiului lui Heisenberg știm cu certitudine că nici o transformare nu poate oferi o rezoluție bună atât în domeniul timp, cât și în domeniul frecvență, acestea realizându-se în același moment de timp. Deoarece funcțiile wavelet de bază sunt unde mai scurte generate prin scalarea funcției wavelet „mamă”, ele sunt bine localizate atât în domeniul timp, cât și cel de scară. Acest comportament automat al transformatei wavelet se potrivește perfect pentru semnalele de voce care necesită o rezoluție de înaltă frecvență pentru a analiza componentele cu frecvențele reduse (semnale vocalizate) și o rezoluție temporală ridicată pentru analiza componentelor de frecvență înaltă (punem accent pe semnalele nevocalizate, spre exemplu: semnalele seismice, semnalele produse de nou-născuți, etc). Acest comportament inteligent al DWT este utilizat pentru reprezentări ale semnalelor acustice [YWS92], codări ale vocii [Lit98, CD99] folosind pentru reprezentare pachete wavelet în vederea modelării, segmentării semnalelor [TLS+94] și clasificarea fonetică a acestora [CG05, PK04, PK05a].

În ultimii ani, abordarea diminuată a wavelet-ului a fost dezvoltată rapid și îmbunătățită, toate acestea pornind de la pragurile fine și simple [Don95]. Multe din îmbunătățirile ce s-au adus pragurilor de tip wavelet pentru a spori eficiența analizei semnalelor de vorbire, au fost realizate ca prag semisoft cu pragul selectat pentru regiuni de semnal nevocalizate [SB97], praguri dure și fine [SMM02] bazate pe μ -legea [SA01, CKYK02, PK05b]. În [LGG04], combinația de fin și dur este aplicată pragurilor pentru a se adapta la proprietăți diferite ale semnalului de vorbire. Prin intermediul distorsiunilor mai mici, metodele de prag sunt integrate cu alte tehnici, cum ar fi operatorul de energie Teger și adaptorul de prag mascat [BR01].

4.2 IMPLEMENTAREA SOFTWARE A FUNCȚIEI DE EXTRAGERE ȘI CALCUL AL COEFICIENȚILOR WAVELET

Ne propunem să extragem coeficienții de aproximație și de detaliu după cum este ilustrat și în figura de mai jos. Așa cum expuneam în al doilea capitol acești coeficienți sunt obținuți în urma aplicării transformatei wavelet, mai precis după aplicarea a două filtre după cum urmează:

- Pentru coeficienții de aproximație se aplică un filtru trece-jos, după care se realizează down-sampling. Acest pas continuă până când se parcurge complet numărul nivelelor impuse de către programator și asignate funcției de descompunere
- Pentru coeficienții de detaliu se aplică un filtru trece-sus. Algoritmul este continuat la pasul cu coeficienții de aproximație, aici obținându-se coeficienții de detaliu pentru primul nivel.

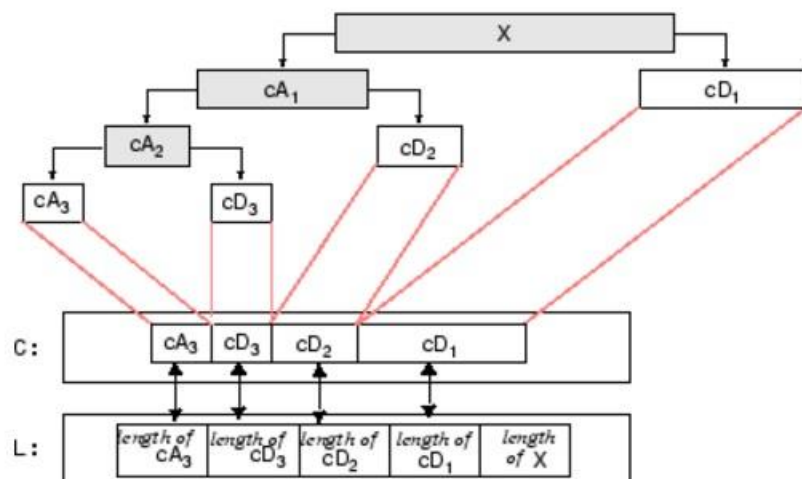


Figure 4.1 Descompunerea semnalului cu ajutorul funcției wavedec

După cum ne este ilustrat în figura de mai sus, semnalul nostru x este descompus în primă instanță prin aplicarea succesivă atât a unui filtru trece-jos pentru calculul coeficienților de aproximație, cât și a unui filtru trece-sus pentru calculul coeficienților de detaliu. După obținerea coeficienților de detaliu ai primului nivel se continuă descompunerea semnalului pe nivelul următor așa cum ne este prezentat și în imagine. Pentru primul nivel de aproximație se aplică din nou o serie de filtre al căror rezultat să ne ilustreze coeficienții de aproximație și de detaliu ai nivelului 2. La ultimul pas se ilustrează același lucru doar că filtrele sunt aplicate de pe nivelul 2. Vectorul de coeficienți ce se creează este format din (cA_3, cD_3, cD_2, cD_1) , iar vectorul L reține lungimea coeficienților aferenți fiecărui nivel.

După extragerea acestor coeficienți ne propunem să calculăm o serie de parametri cu care ne dorim să antrenăm modelul SVM pentru baza de date Dunstan, aferentă acestui proiect și pe care am realizat până acum experimente.

Prin aplicarea transformatei Wavelet discretă pentru $M=3$ pentru fiecare fereastră de vorbire se va obține o serie de coeficienți de aproximație și patru serii de coeficienți de detaliu care vor forma secvența coeficienților wavelet $X_{3,i}(n) = \{a_{3,p}, d_{3,p}, d_{2,p}, d_{1,p}\}$, unde $m=3,2,1$.

Coeficientul ce aparține nivelului 4 și cei trei coeficienți de detaliu obținuți se regăsesc dispuși astfel: $\{N_3=N_f/8, N\}$

4.3 ÎMPĂRȚIREA BAZEI DE DATE

Baza de date Dunstan este împărțită în cinci clase (plânsete de foame, somn, disconfort, colici, eructație), așa cum am specificat și în primul capitol. Pentru a putea realiza clasificarea propriu-zisă a fost însă necesară realizarea mai multor seturi de teste atât de antrenare, cât și de testare. Aceste seturi au fost realizate cu ajutorul funcției implementate în cadrul acestui proiect de diplomă, denumită Cross-validare. Aceasta are rolul de a selecta din toată baza de date atât fișiere pentru etapa de antrenare, cât și fișiere pentru etapa de testare. Acest lucru se realizează

cu ajutorul calculării minimului unui vector în care se concatenează vectorul și verificarea dacă ID-ul bebelușului selectat din listă se regăsește în numele fișierului. Dacă minimul acestui vector este zero atunci fișierul meu va fi distribuit pentru lista de testare, altfel el va fi atribuit listei de antrenare.

Cele două liste rezultate vor fi folosite mai departe pentru etapa de antrenare și, respectiv, etapa de testare. Din imaginea ilustrată mai jos se poate observa că pentru etapa de antrenare avem un procentaj mai mare de fișiere decât pentru cea de testare, fiind necesar acest lucru deoarece avem nevoie să antrenăm cât mai bine baza de date în vederea recunoașterii unui număr cât mai mare de plânsete și clasificarea lor clară pe nevoi.

```

eairh_36_4_Bior3.txt
eairh_37_6_Bior3.txt
eairh_38_8_Bior3.txt
eh_12_1_Bior3.txt
eh_16_8_Bior3.txt
eh_21_14_Bior3.txt
eh_23_6_Bior3.txt
eh_30_7_Bior3.txt
eh_32_4_Bior3.txt
eh_3_2_Bior3.txt
eh_6_10_Bior3.txt
eh_6_11_Bior3.txt
heh_10_3_Bior3.txt
heh_12_8_Bior3.txt
heh_16_4_Bior3.txt
heh_1_9_Bior3.txt
heh_21_13_Bior3.txt
heh_32_7_Bior3.txt
heh_3_2_Bior3.txt
heh_40_5_Bior3.txt
heh_41_11_Bior3.txt
neh_10_1_Bior3.txt
neh_12_1_Bior3.txt
neh_14_1_Bior3.txt
neh_15_1_Bior3.txt
neh_16_1_Bior3.txt
neh_17_1_Bior3.txt
neh_19_1_Bior3.txt
neh_1_1_Bior3.txt
neh_21_1_Bior3.txt
neh_23_1_Bior3.txt
neh_2_1_Bior3.txt
neh_2_2_Bior3.txt
neh_3_1_Bior3.txt
neh_4_1_Bior3.txt
neh_6_1_Bior3.txt
neh_7_1_Bior3.txt
owh_10_6_Bior3.txt
owh_12_2_Bior3.txt
owh_15_13_Bior3.txt
owh_16_4_Bior3.txt
owh_17_3_Bior3.txt
owh_1_15_Bior3.txt
owh_26_2_7_Bior3.txt
owh_26_5_Bior3.txt
owh_28_10_Bior3.txt
owh_29_14_Bior3.txt
owh_30_1_Bior3.txt
owh_31_18_Bior3.txt
owh_32_19_Bior3.txt
owh_3_1_11_Bior3.txt
owh_3_2_16_Bior3.txt

set_1_test_Bior3 - Notepad
File Edit Format View Help
eairh_11_3_Bior3.txt
eairh_18_9_Bior3.txt
eairh_35_1_Bior3.txt
eairh_35_2_5_Bior3.txt
eairh_35_3_7_Bior3.txt
eairh_39_10_Bior3.txt
eairh_5_2_Bior3.txt
eh_18_9_Bior3.txt
eh_20_13_Bior3.txt
eh_22_12_Bior3.txt
eh_5_2_5_Bior3.txt
eh_5_3_Bior3.txt
heh_22_12_Bior3.txt
heh_5_10_Bior3.txt
heh_9_1_Bior3.txt
heh_9_6_Bior3.txt
neh_11_1_Bior3.txt
neh_18_1_Bior3.txt
neh_20_1_Bior3.txt
neh_22_1_Bior3.txt
neh_22_2_Bior3.txt
neh_25_1_Bior3.txt
neh_5_1_Bior3.txt
neh_8_1_Bior3.txt
neh_9_1_Bior3.txt
owh_11_12_Bior3.txt
owh_18_9_Bior3.txt
owh_9_17_Bior3.txt
owh_9_1_6_Bior3.txt

```

Figure 4.2 Setul de antrenare(stânga) și setul de testare(dreapta)

4.4 CALCULUL TRĂSĂTURILOR

În cadrul celei de-a doua metode ne-am propus ca din semnalul inițial să păstrăm acele porțiuni în care avem informație utilă. Pentru acest lucru am ales un prag de 0,12. Dacă media pe un bloc este mai mare decât acest prag atunci noi vom partiționa semnalul inițial în subsemnale/blocuri ce vor cuprinde 256 de eșantioane fiecare. Numărul total de blocuri în care poate fi partiționat semnalul este 10, deoarece semnalele nu sunt egale ca și durată, astfel că dacă am lua un număr mai mare de blocuri când pragul nostru nu ar mai fi depășit am obține valori nule, ceea ce nu ar avea ca rezultat un nou semnal. Pentru fiecare bloc în parte vor fi calculați coeficienții wavelet de aproximație, respectiv detaliu și va fi extras setul de parametri după cum este ilustrat mai jos.

Setul de parametri ce sunt calculați după extragerea coeficienților wavelet.

- Wavelet power ratio (WPR) – este raportul dintre energia calculată pentru coeficienții de aproximație ai primului nivel și puterea tuturor coeficienților:

$$WPR(i) = \frac{2N_3 + N_2 + N_1}{2N_3 + N_2} \frac{\sum_{n=1}^{N_1} X_{3,i}^2(n)}{\sum_{n=1}^{N_1} X_{3,i}^2(n) + \sum_{n=N_1+1}^{N_f} X_{3,i}^2(n)}$$

- Power variation of detail coefficient (PVD) – diferența dintre energia calculată pentru coeficienții de detaliu ai primului nivel și cei de pe al treilea nivel:

$$PVD(i) = \frac{1}{N_1} \sum_{n=N_1+1}^{N_f} X_{3,i}^2(n) - \frac{1}{N_3} \sum_{n=N_1+1}^{N_2} X_{3,i}^2(n)$$

- Short-term logarithmic average energy (SAE) – este calculată pentru fiecare bloc ce s-a extras din semnalul inițial:

$$SAE(i) = 10 \log \left(\frac{1}{N_f} \sum_{l=1}^{N_f} x(l)^2 \right)$$

- Zero crossing rate (ZCR) – este numărul de schimbări de semn pentru fiecare subsemnal extras din semnalul inițial:

$$ZCR(i) = \sum_{l=1}^{N_f} |sgn(x(l)) - sgn(x(l-1))|$$

- Closure interval-transient detail ratio (CTDR) – este raportul dintre energia calculată pentru coeficienții de detaliu ai nivelului i și energia calculate pentru coeficienții de detaliu de pe nivelul $(i-1)$:

$$CTDR_m(i) = \frac{\sum_{n=N_m+1}^{N_{m-1}} X_{3,i}^2(n)}{\sum_{n=N_m+1}^{N_{m-1}} X_{3,i-1}^2(n)}$$

După calculul acestor parametri îi vom distribui într-un vector în care:

1. prima coloană va fi dată de labelul corespunzător nevoii după cum urmează:
 - Dacă aparține clasei de semnale corespunzătoare nevoii ce ilustrează durerea de burtă (EAI RH) atunci acesta va avea labelul 1;

- Dacă aparține clasei de semnale corespunzătoare nevoii ce ilustrează durerea de burtă (EH) atunci acesta va avea labelul 2;
 - Dacă aparține clasei de semnale corespunzătoare nevoii ce ilustrează durerea de burtă (HEH) atunci acesta va avea labelul 3;
 - Dacă aparține clasei de semnale corespunzătoare nevoii ce ilustrează durerea de burtă (NEH) atunci acesta va avea labelul 4;
 - Dacă aparține clasei de semnale corespunzătoare nevoii ce ilustrează durerea de burtă (OWH) atunci acesta va avea labelul 5.
2. Următoarele coloane sunt formate din blocuri de câte cinci parametri după cum urmează: WPR, PVD, SAE, ZCR, CDTR. Vor exista 10 blocuri a câte cinci parametri, deci un total de 50 de parametri.

În final va rezulta o matrice ce are ca dimensiune 83x50, unde 83 reprezintă numărul total de semnale inițiale pentru care se vor calcula cei cinci parametri expuși mai sus, iar 50 reprezintă concatenarea dintre labelul asociat fiecărui tip de semnal și coeficienții calculați pentru cele 10 blocuri extrase din semnalul inițial.

Matricea de parametri rezultată va arăta de această formă:

The screenshot shows the MATLAB interface with a variable viewer window titled 'Variables - feat_param'. The variable is an 83x50 double matrix. The matrix data is as follows:

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	1	0.0131	0.9405	0.9114	1	0.0408	0.8545	0.6095	0.7813	0.0076	0.0127	0.8776	0.4969	0.7704	0.3059	0.0761	0.9450	0.6075	0.7643	0.1266	0.0
1	1	0.0846	0.6993	0.9462	0.5565	0.2775	0.5125	0.8340	0.2891	0.0073	0.1912	0.4276	0.7573	0.2593	0.3446	0.2725	0.6154	0.8602	0.2429	0.1259	0.3
1	1	0.4810	0.5783	0.7393	0.2339	0.5993	0.7634	0.5244	0.2500	0.0087	0.7036	0.7780	0.4237	0.3259	0.5719	0.6262	0.6825	0.7392	0.2929	0.0915	0.4
1	1	0.0081	0.8324	0.9363	0.7742	0.0190	0.7332	0.6972	0.6484	0.0158	0.0447	0.8991	0.5074	0.7111	0.3135	0.0077	0.9307	0.6335	0.7000	0.0784	0.0
1	1	0.3688	0.7104	0.8221	0.6290	0.3741	0.7672	0.5785	0.4766	0.0047	0.3854	0.7936	0.5276	0.4222	0.3516	0.3337	0.8731	0.5756	0.4071	0.1096	0.1
1	1	0.4655	0.6702	0.7827	0.2984	0.6456	0.6782	0.8371	0.2500	0.0139	0.6273	0.7670	0.6967	0.1852	0.4519	0.6814	0.8271	0.7744	0.1786	0.0859	0.8
1	1	0.6251	0.6750	0.7428	0.2500	0.5602	0.8462	0.5080	0.2031	0.0266	0.8549	0.8372	0.5841	0.1852	0.1979	0.6219	0.9351	0.5066	0.2143	0.0930	0.7
1	1	0.0439	1	0.9088	0.9919	0.0214	1	0.7183	0.9453	0.0098	0.0058	1	0.6303	0.8667	0.3313	0.0476	0.9597	0.7832	0.7500	0.1154	0.0
1	1	0.1894	0.5473	0.9070	0.4516	0.2427	0.7660	0.6419	0.4531	0.0071	0.1886	0.7113	0.5204	0.4593	0.2541	0.2086	0.8483	0.5694	0.3571	0.1560	0.3
1	1	0.0830	0.2472	0.7925	0.5081	0.1121	0.4272	0.6312	0.4453	0.0068	0.2447	0.7220	0.4048	0.4444	0.2722	0.0314	0.7854	0.5204	0.4000	0.1190	0.0
1	1	0.5359	0.6002	0.7378	0.2581	0.4135	0.4683	0.6434	0.2500	0.0107	0.4238	0.6883	0.5061	0.2444	0.4074	0.6926	0.8354	0.5170	0.2286	0.0646	0.5
1	1	0.1995	0.4312	0.8704	0.5726	0.0192	0.6007	0.7240	0.5000	0.0130	0.0379	0.6153	0.6282	0.5333	0.2351	0.0648	0.4525	0.9572	0.5143	0.2178	0.0
2	2	0.3326	0.5025	0.9605	0.2742	0.5055	0.6453	0.8338	0.2656	0.0251	0.4481	0.6602	0.7465	0.1926	0.5114	0.4341	0.7471	0.8842	0.2143	0.0304	0.7
2	2	0.6561	0.8088	0.8119	0.2177	0.7431	0.8789	0.6209	0.2188	0.0199	0.7276	0.8062	1	0.1556	0.5421	0.8813	0.8994	0.7415	0.2357	0.0735	0.8
2	2	0.0756	0.7330	0.8756	0.9435	0.1115	0.9033	0.7114	0.9375	0.0166	0.2301	0.9228	0.5377	0.7778	0.2166	0.0469	0.9668	0.5607	1	0.1417	0.1
2	2	0.6766	0.7388	0.7018	0.2097	0.7970	0.8483	0.4832	0.1953	0.0175	0.8774	0.9089	0.4405	0.1704	0.0591	0.8446	0.9020	0.8848	0.1357	0.4277	
2	2	0.6719	0.6100	0.7319	0.2339	0.5335	0.7716	0.5760	0.2266	0.0099	0.4700	0.7824	0.4371	0.2222	0.3107	0.7570	0.8444	0.5683	0.2071	0.1076	0.5
2	2	0.3576	0.2432	0.7798	0.2823	0.4546	0.6987	0.5028	0.2656	0.0256	0.4226	0.8289	0.4177	0.2667	1	0.5313	0.6860	0.7806	0.2143	0.0629	0.6
2	2	0.1550	0.4956	0.8909	0.5403	0.4058	0.5554	0.7833	0.3828	0.0077	0.7647	0.7545	0.6322	0.3111	0.2787	0.3760	0.7845	0.7264	0.3714	0.0645	0.7
2	2	0.4569	0.5403	0.9018	0.3306	0.3815	0.5546	0.6648	0.2734	0.0246	0.4797	0.7699	0.4591	0.3556	0.4019	0	0.9811	0	0	0	
2	2	0.3326	0.5025	0.9606	0.2742	0.5055	0.6453	0.8338	0.2656	0.0251	0.4481	0.6602	0.7465	0.1926	0.5114	0.4341	0.7471	0.8842	0.2143	0.0304	0.7

Figure 4.3 Matricea corespunzătoare parametrilor extrași

CAPITOLUL 5

EXPERIMENTE

În cadrul acestui capitol ne-am propus să ilustrăm ceea ce am realizat în materie de experimente pe baza de date DUNSTAN. Atât în cadrul primei metode, cât și în cazul celei de-a doua metode s-au realizat zece seturi de antrenare și zece seturi de testare cu ajutorul funcției de cross-validare. Acest lucru s-a realizat pentru fiecare familie de Wavelet-uri în parte, fișierele fiind de fiecare dată altele, iar datele conținute fiind specifice familiei pentru care s-au calculat coeficienții de aproximație, respectiv detaliu. În cadrul acestei lucrări de diplomă am lucrat cu șase familii de wavelet-uri, deci vom avea 60 de seturi de antrenare și 60 de seturi de testare pentru fiecare metodă în parte.

În cadrul primei metode, după calculul coeficienților wavelet de aproximație, respectiv detaliu, se vor calcula energiile corespunzătoare fiecărui nivel în parte. Matricea ce va rezulta va avea dimensiunea de 83×6 , unde 83 reprezintă numărul total de semnale inițiale pentru care vom asocia un label ce corespunde fiecărui tip de nevoie în parte, iar numărul 6 reprezintă concatenarea între labelul asociat semnalului inițial și cele cinci energii corespunzătoare fiecărui nivel în parte.

În cadrul celei de-a doua metode vom avea o matrice ce va avea dimensiunea de 83x50, unde vom discuta despre același număr de semnale inițiale(83), iar 50 reprezintă concatenarea dintre labelul asociat fiecărui semnal inițial în parte și numărul de parametri calculat în cadrul celei de-a doua metode.

În cadrul ambelor metode există atât funcție ce realizează cross-validarea, cât și funcții specifice pentru antrenarea bazei de date și testarea acesteia.

De asemenea, în cazul ambelor metode parametrii obținuți pentru fiecare semnal sunt stocați în fișiere de tipul txt ce au ca denumire numele semnalului inițial. Fiecare astfel de fișier se va găsi în folderul specific fiecărei metode în parte. Aceste fișiere conțin câte o linie aferentă fiecărui semnal în parte. Pentru etapa de antrenare, cât și pentru etapa de testare acestea sunt încărcate într-o matrice cu ajutorul funcției csvread existentă în cadrul mediului de dezvoltare Matlab.

Această matrice, așa cum am specificat în paragrafele de mai sus este constituită dintr-o parte care face referire la label-urile aferente fiecărui tip de semnal în parte și o parte specifică parametrilor calculați pentru fiecare metodă în parte.

În cadrul primei metode am obținut o serie de rezultate ce sunt specificate în tabelul de mai jos:

Tabel 6.1 Rezultate pentru metoda I

Acuratețe						
Fam.Wavelet Nr_test	Coiflet3	Coiflet5	Bior3.5	Daubechies	Symlet5	Dmey
1	28.57%	31.81%	41.37%	53.48%	50%	37.93%
2	34.37%	43.75%	47.36%	28%	32%	55%
3	33.33%	44%	40.00%	34.78%	61.11%	32%
4	45.83%	61.11%	57.14%	41.66%	44.44%	43.47%
5	44%	38.88%	42.85%	60%	52.94%	61.53%
6	42.85%	52.94%	42.30%	27.27%	28%	28%
7	44.44%	36%	53.33%	55%	52%	34.78%
8	38.46%	52%	36.66%	33%	35.29%	41.66%
9	40%	41.17%	44.44%	38.09%	38.46%	50%
10	38.09%	34.61%	40.90%	37.5%	30.76%	40.90%

În tabelul de mai sus sunt prezentate rezultatele pentru prima metodă. Cu culoarea roșie este pusă în evidență cea mai bună acuratețe obținută pentru un tip de familie wavelet, iar culoarea albastră ilustrează cea mai slabă acuratețe obținută pentru setul de antrenare, testare și familia specificate.

Pentru această metodă am obținut cea mai bună acuratețe în cazul familiei Discrete Meyer și are valoarea de 61,53%, pentru setul de antrenare și testare cu numărul 5. Cea mai slabă acuratețe dintre toate a fost obținută pentru familiile Daubechies, Symlet5 și Discrete Meyer pentru seturile de antrenare și testare 2, 6, 6 și are un procent de 28%.

Am notat cu Nr_test numărul setului de antrenare, respectiv testare care s-a obținut în urma apelării funcției de cross-validare, iar cu Fam. Wavelet am notat familia de wavelet-uri cu ajutorul căreia s-au extras coeficienții de aproximație și de detaliu.

Tabel 6.2 Rezultate pentru metoda II

Acuratețe						
Fam.Wavelet Nr_test	Coiflet3	Coiflet5	Bior3.5	Daubechies	Symlet5	Dmey
1	61.90%	40%	62.5%	44%	55%	61.90%
2	51.85%	50%	34.78%	44.4%	40.90%	57.14%
3	42.85%	53.33%	56.52%	52%	65%	62.5%
4	57.69%	64%	52%	60%	45.83%	61.53%
5	60%	61.90%	52.63%	47.05%	43.47%	40%
6	47.61%	40%	37.93%	53.84%	48%	43.33%
7	48%	50%	52.38%	38.46%	53.84%	59.25%
8	56%	55.55%	53.12%	50%	48.14%	50%
9	34.78%	52.38%	45.83%	23.07%	65.21%	62.5%
10	41.66%	48%	54.16%	48.27%	65%	42.85%

În cadrul celui de-al doilea tabel, observăm o acuratețe mai bună decât în cazul primei metode. Aici am obținut cea mai bună acuratețe pentru cazul familiei Symlet5 ce s-a realizat pentru testul cu numărul 5. De asemenea observăm că cea mai mică acuratețe obținută se păstrează în cazul familiei Daubechies, aceasta având de această dată valoarea de 23.07% și experimentele având loc pentru setul de date cu numărul 9.

Precizez că aceste rezultate sunt obținute pentru un număr de 37 de nou-născuți, iar setul de test pentru care am obținut această acuratețe reprezintă 10% din valoarea tuturor fișierelor din baza de date.

CAPITOLUL 6

CONCLUZII

6.1 CONCLUZII GENERALE

În cadrul acestei lucrări de diplomă am avut ca obiective următoarele:

- Extinderea bazei de date SPLANN
- Implementarea unor funcții care să recunoască plânsetele nou-născuților
- Clasificarea cu ajutorul clasificatorului SVM a plânsetelor nou-născuților

Ne-am propus abordarea a două metode care să recunoască și să clasifice plânsetele nou-născuților în funcție de nevoile pe care aceștia le au, iar în urma aplicării acestora pe baza de date DUNSTAN am obținut următoarele rezultate ce sunt ilustrate în cele două tabele de mai sus.

Realizând o comparație între cele două tabele de mai sus, pe coloană, observăm faptul că trăsăturile extrase în cazul celei de-a doua metode sunt mai concludente, duc către o acuratețe mai bună, deoarece acestea transpun legătura ce se realizează între blocuri. Noi am împărțit semnalul inițial în zece blocuri, rezultatele obținute în acest caz ne indică faptul că de fapt

acestea constituie niște semnale cvasi-staționare, deci sunt ilustrate proprietăți statistice de sine stătătoare pentru fiecare bloc în parte.

În cadrul acestei lucrări am obținut următoarele rezultate, privind acuratețea, pentru următoarele familii de wavelet-uri, după cum urmează:

- pentru familia de wavelet-uri Daubechies o acuratețe de 60%, atât pentru cazul primei metode, cât și pentru cazul celei de-a doua metode;
- pentru familia de wavelet-uri Symlet5 o acuratețe de 61,11% în cazul primei metode, pe când în cazul celei de-a doua metode am obținut cea mai mare acuratețe reieșită din toate experimentele efectuate, aceea de 65,21%;
- pentru familia de wavelet-uri Dmey, în cadrul primei metode am obținut o acuratețe de 61,53%, iar în cadrul celei de-a doua metode am obținut o acuratețe mai bună cu un procent: 62,5%;
- pentru familia de wavelet-uri Coiflet3, cazul celei de-a doua metode ne-a adus odată cu rularea experimentelor și un procent mult mai mare în ceea ce privește acuratețea: 61,90%, față de cea obținută în cazul primei metode care avea o valoare de 45,83%;
- pentru familia de wavelet-uri Coiflet5, cele două metode se deosebesc printr-un număr de aproximativ trei procente, în cazul primei metode obținându-se o acuratețe de 61,11%, pe când în cazul celei de-a doua metode s-a obținut o acuratețe de 64%;
- pentru familia de wavelet-uri Biorthogonal 3.5, observăm o diferență de cinci procente, în cadrul primei metode obținându-se o acuratețe de 57,14%, pe când pentru cea de-a doua s-a obținut o acuratețe de 62,5%.

Per ansamblu, putem concluziona spunând că pentru cazul celei de-a doua metode rezultatele privind atât trăsăturile extrase, cât și acuratețea obținută mult mai bune decât în cadrul primei metode folosite în această lucrare de diplomă, toate acestea conducând către rezultatul final ce are valoarea de 65,2%.

6.2 CONTRIBUȚII PERSONALE

Contribuțiile personale ale autorului acestei lucrări de diplomă sunt organizate de-a lungul întregii lucrări.

În cadrul acestei lucrări au fost scrise și executate programe atât pentru partea de configurare a fișierelor, cât și pentru partea de antrenare și testare a modelelor generate cu ajutorul funcției svmtrain. Aceste programe definesc următoarele părți ale lucrării:

- a) partea de configurare, script ce este rulat de fiecare dată pentru un test nou;
- b) partea de extragere a coeficienților de aproximație și de detaliu cu ajutorul transformatei wavelet;
- c) partea de calcul a trăsăturilor propriu-zise: calculul energiilor/calculul celor cinci trăsături descrise în cadrul capitolului 4
- d) partea de antrenare și generare a modelului SVM pentru un set de date;
- e) partea de testare a modelului SVM antrenat la pasul anterior.

6.3 EXPERIMENTE ULTERIOARE

Pe viitor ne dorim să extindem experimentele și pe baza de date SPLANN, astfel sperând să având mai multe seturi cu care putem să antrenăm mai bine modelul SVM, când vom realiza testarea acestuia rezultatele obținute să fie mai bune, vom obține o acuratețe cât mai mare.

De asemenea avem în vedere realizarea unor experimente care să includă antrenarea bazei de date și testarea cu ajutorul rețelelor neuronale.

REFERINȚE

[Bertoldi, 2009] Bertoldi, N., Haddow, B., Fouet, J.-B., “Improved Minimum Error Rate Training in Moses,” *The Prague Bulletin of Mathematical Linguistics*, pp. 1-11, February 2009.

[Academic Press,1992] Akansu, Ali N.; Haddad, Richard A. (1992), *Multiresolution signal decomposition: transforms, subbands, and wavelets*, Boston, MA: Academic Press, ISBN 978-0-12-047141-6

[Kingsbury, 2005] Selesnick, I.W.; Baraniuk, R.G.; Kingsbury, N.C., 2005, *The dual-tree complex wavelet transform*

[Academic, 1999] S. Mallat, *A Wavelet Tour of Signal Processing*, 2nd ed. San Diego, CA: Academic, 1999.

[Mallat, 1992] S. G. Mallat and S. Zhong, “Characterization of signals from multiscale edges,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 7, pp. 710– 732, Jul. 1992.

[Gabbouj, 2009] Ince, Kiranyaz, Gabbouj, 2009, *A generic and robust system for automated patient-specific classification of ECG signals*

[Broughton, 2017] S. Allen. "Wavelet Based Methods in Image Processing". *www.rose-hulman.edu*. Retrieved 2017-05-02.

[Kluwer Academic Publishers, 1995] A.N. Akansu and M.J.T. Smith, *Subband and Wavelet Transforms: Design and Applications*, Kluwer Academic Publishers, 1995.

[Kluwer Academic Publishers, 1999] A.N. Akansu and M.J. Medley, *Wavelet, Subband and Block Transforms in Communications and Multimedia*, Kluwer Academic Publishers, 1999.

[IEEE, 1998] A.N. Akansu, P. Duhamel, X. Lin and M. de Courville *Orthogonal Transmultiplexers in Communication: A Review*, *IEEE Trans. On Signal Processing*, Special Issue on Theory and Applications of Filter Banks and Wavelets. Vol. 46, No.4, pp. 979–995, April, 1998.

[Elsevier, 2010] A.N. Akansu, W.A. Serdijn, and I.W. Selesnick, *Wavelet Transforms in Signal Processing: A Review of Emerging Applications*, *Physical Communication*, Elsevier, vol. 3, issue 1, pp. 1–18, March 2010.

[Sixth Indian Conference on Computer Vision, 2008] Pragada, S.; Sivaswamy, J. (2008-12-01). "Image Denoising Using Matched Biorthogonal Wavelets". 2008 Sixth Indian Conference on Computer Vision, Graphics Image Processing: 25–32. *doi:10.1109/ICVGIP.2008.95*.

[MATLAB, 2017] "Thresholds for wavelet 1-D using Birgé-Massart strategy - MATLAB wdcbm". *www.mathworks.com*. Retrieved 2017-05-03.

[Burhan, 2012] Ergen, Burhan (2012-01-01). *Signal and Image Denoising Using Wavelet Transform*. InTech. doi:10.5772/36434.

[BR01] M. Bahoura and J. Rouat. Wavelet speech enhancement based on the teager energy operator. In *IEEE Signal Processing Letter*, volume 8, pages 10–12, 2001.

[CD99] B. Carnero and A. Drygajlo. Perceptual speech coding and enhancement using frame-synchronized fast wavelet packet transform algorithms. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 47:1622–1635, 1999.

[CG05] G.F. Choueiter and J.R. Glass. A wavelet and filter bank framework for phonetic classification. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 933–936, 2005.

[CKYK02] S. Chang, Y. Kwon, S. Yang, and I. Kim. Speech enhancement for nonstationary noise environment by adaptive wavelet packet. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 561–564, 2002.

[Don95] D. L. Donoho. De-noising by soft thresholding. *IEEE Trans. Information Theory*, 41:613–627, 1995.

[Lit98] Jr Litwin, L.R. Speech coding with wavelets. *IEEE Potentials*, 17(2):38–41, 1998.

[LGG04] A. Lallouani, M. Gabrea, and C.S. Gargour. Wavelet based speech enhancement using two different threshold-based denoising algorithms. In *Proceedings of the Canadian Conference on Electrical and Computer Engineering*, pages 315–318, 2004.

[PK04] T. V. Pham and G. Kubin. DWT-based classification of acoustic-phonetic classes and phonetic units. In *Proceedings of International Conference on Spoken Language Processing*, pages 985–988, 2004.

[PK05a] T. V. Pham and G. Kubin. DWT-based phonetic groups classification using neural networks. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 401–404, 2005.

[PK05b] T. V. Pham and G. Kubin. WPD-based noise suppression using nonlinearly weighted threshold quantile estimation and optimal wavelet shrinking. In *Proceedings of Interspeech*, pages 2089–2092, 2005.

[SA01] H. Sheikhzadeh and H. R. Abutalebi. An improved wavelet-based speech enhancement system. In *Proceedings of Eurospeech*, pages 1855–1858, 2001.

[SB97] J. W. Seok and K. S. Bae. Speech enhancement with reduction of noise components in the wavelet domain. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, volume 2, pages 1323–1326, 1997.

[SMM02] Zilany M. S., Hasan Md., and Khan M. Efficient hard and soft thresholding for wavelet speech enhancement. In *Proceedings of the European Signal Processing Conference*, 2002.

[TLS+94] B. Tan, R. Lang, H. Schroder, A. Spray, and P. Dermody. Applying wavelet analysis to speech segmentation and classification. In *Proceedings of the SPIE Conference on Wavelet Applications in Signal and Image Processing*, pages 750–761, 1994.

[YWS92] X. Yang, K. Wang, and S.A. Shamma. Auditory representations of acoustic signals. *IEEE Transactions on Information Theory*, 38(2):824–839, 92.

ANEXE

Calculul trăsăturilor pentru cazul primei metode

```

function [list_coef, wavs]=extractWavelet(pathToWavs, pathToWavelet, cryClasses, cryLabels,
user_folder, lista_dir)
wavs = [];
wavs = [wavs ; cell(size(wavs, 1),1)];
list_coef = []; % matrice a indexilor asociati cu energiile corespunzatoare
contor = 1;

for m = 3 : (size(user_folder, 1))
    director = fullfile(sprintf('%s', pathToWavs), sprintf('%s', lista_dir{m})); %luam fiecare folder al
    fiecarui vorbitor
    lista_wavs = dir([director, '/*.wav']);% lista tuturor fisierelor wav ale unui vorbitor.
    for i = 1:(size(lista_wavs, 1))
        % Read the data of the WAV file and store it.
        [wavs{contor}] = audioread(lista_wavs(i).name);
        [C,L] = wavedec(wavs{contor},4,'db4');
        result = Energie(C,L);
        result = result';

        for j = 1:length(cryClasses)
            label = strfind(lista_wavs(i).name, cryClasses{j});
            if(label == 1)
                current_label = cryLabels(j);
                list_coef(contor,:) = [current_label result];
                lista = fopen(sprintf('%s',pathToWavelet,'',lista_wavs(i).name(1:end-4),'_DB.txt'),'wt');
                fprintf(lista,'%1.4f' ,list_coef(contor,:), ' ');%printeaza linie cu linie in fisie
                contor = contor + 1;
            end
        end
    end
end
fclose all;
end

```

Funcția corespunzătoare calculului energiilor

```
function result = Energie(C, L)
nr_nivel = size(L,1)-1;
% nr_nivel=size(l,1)-1;
offset = 0;
%offsetL=0;
energy=0;
for i = 1:nr_nivel
    if(i == 1)
        energy(i) = sum(C(1:L(1)).^2);
        offset = offset + L(1);
    else
        energy(i) = sum(C(offset + 1 : offset + L(i)).^2);
        offset = offset + L(i);
    end
end
energy = energy / sum(energy);
result = energy.';
end
```

Calculul trăsăturilor pentru cazul celei de-a doua metode

```
function [feat] = CALC_parametri(y)
lb = 256; % lungime bloc
N = 10; % nr blocuri
x = cell(N,1);
prag=0.12;
t = 1;
j = 1;
while(t<=length(y)-lb)
    if(j <= N)
        if(mean(abs(y(t:t+lb)))> prag)
            x{j,1} = y(t:t+lb-1);
            fprintf('Sample cut-off: %d \n', t);
            t = t+lb;
            j = j+1;
        else
            t = t+1;
        end
    else
        break;
    end
end
figure(),
for j = 1:10
    subplot(5,2,j),plot(x{j,1});
    hold on;
end
cd1 = cell(length(x),1);
cd2 = cell(length(x),1);
cd3 = cell(length(x),1);
WPR = zeros(length(x),1);
```



```

PVD = zeros(length(x),1);
SAE = zeros(length(x),1);
ZCR = zeros(length(x),1);
CDTR = ones(length(x),1);
for i=1:length(x)
    [C,L] = wavedec(x{i,1},4,'sym5');
    [cd1{i,1},cd2{i,1}, cd3{i,1}] = detcoef(C,L,[1 2 3]);
    A = appcoef(C,L,'sym5',3);
    N1 = length(cd1{i,1});
    N2 = length(cd2{i,1});
    N3 = length(cd3{i,1});

    WPR(i,1) = (lb/(N1+N2+N3))* sum(A.^2)/(sum(x{i,1}.^2));
    PVD(i,1) = 1/N1 * sum(cd1{i,1}.^2)- 1/N3 * sum(cd3{i,1}.^2);
    SAE(i,1) = 10*log10(1/lb*sum(x{i,1}.^2)+0.01);
    ZCR(i,1) = sum(abs(sign(x{i,1}(2:end))-sign(x{i,1}(1:end-1))));
end
for i=2:length(x)
    CDTR(i,1) = sum(cd1{i,1}.^2)/(sum(cd1{i-1,1}.^2))*0.00001;
end
feat = [WPR, PVD, SAE, ZCR, CDTR];
feat = reshape(feat.', 1,[]);

feat(5) = []; % scoatere CDTR al primului bloc

feat (isnan(feat)) = 0;
end

```

Crearea matricii cu care urmează a fi antrenat și testat modelul SVM

```

function [feat, current_label, feat_param]= citire_wavs(pathToWavs,
user_folder,cryClasses,cryLabels, lista_dir)
wavs = [];
wavs = [wavs ; cell(size(wavs, 1),1)];
contor = 1;
flag = 1;
for m = 3 : (size(user_folder, 1))
    director = fullfile(sprintf('%s', pathToWavs), sprintf('%s', lista_dir{m})); %luam fiecare folder al
fiecarii vorbitor
    lista_wavs = dir([director, '/*.wav']);% lista tuturor fisierelor wav ale unui vorbitor.
    for i = 1:size(lista_wavs, 1)
        % Read the data of the WAV file and store it.
        [wavs{contor}] = audioread(lista_wavs(i).name);
        contor = contor + 1;
        for j = 1:length(cryClasses)
            label = strfind(lista_wavs(i).name, cryClasses{j});
            if(label == 1)
                current_label(flag) = cryLabels(j);
                flag = flag + 1;
            end
        end
    end
end
end
for j = 1: size(wavs,2)

```

```

    feat(j,:) = CALC_parametri(wavs{1,j});
end
end

```

```

m = min(feat,[],1);
m = repmat(m, size(feat,1),1);
M= max(feat,[],1);
M = repmat(M, size(feat,1),1);

```

```

feat = (feat - m)./(M-m);
feat_param = [current_label' feat];
end

```

Împărțirea bazei de date în fișiere de antrenare, respectiv testare

```

function Cross_wavelet( pathToWavs, test_number, user_folder, lista_dir, n_testSpeakers,
pathToWavelet )
lista_antrenare =
fopen(strcat(pathToWavelet,sprintf('set_%d',test_number),sprintf('_train_Meyer.txt')),'wt');
lista_testare =
fopen(strcat(pathToWavelet,sprintf('set_%d',test_number),sprintf('_test_Meyer.txt')),'wt');
spks = [1:12];
spks = [spks 14:23];
spks = [spks 25 26 28:32];
spks = [spks 35:41];
r = rand(size(spks,2), 1);
[~, idx] = sort(r);
key_words = [];
for t=1:n_testSpeakers
    key_words = [key_words spks(idx(t))];
end
for i = 3: (size(user_folder,1))
    director = dir(fullfile(sprintf('%s', pathToWavs), sprintf('%s', lista_dir{i})));
    lista = {director.name};
    for j = 3:(size(lista,2))
        d=[];
        for r=1:n_testSpeakers
            key_word =sprintf('%s', lista_dir{i}, '_' ,num2str(key_words(r)), '_ ');
            verif = isempty(strfind(lista{j},key_word));
            d = [d verif];
        end
        if(min(d)==0)
            fprintf(lista_testare,sprintf('%s',lista{j}(1:end-4),'_Meyer.txt'));
            fprintf(lista_testare,'\n');
        else
            fprintf(lista_antrenare,sprintf('%s',lista{j}(1:end-4),'_Meyer.txt'));
            fprintf(lista_antrenare,'\n');
        end
    end
end
end
fclose all;
end

```

Funcția de antrenare a modelului SVM

```
function [model] = antrenare_SVM( cale_input)
    cale_input = 'C:/Users/lenovo/Desktop/email/';
    contor = 1;
    TrainMat = [];
    results = fopen(strcat(cale_input,sprintf('matrice_600_11.txt')), 'wt');
    fis_antrenare_m2 = textread(sprintf('%s',cale_input, '/matrice_600_11.txt'), '%s', 'delimiter',
    '\n', 'whitespace', '');
    for i = 1:size(fis_antrenare_m2,1)
        TrainMat(contor,:) = reshape(csvread(strcat(cale_input,fis_antrenare_m2{i})), 1, []);
        contor = contor + 1;
    end

    labels_train_mat = TrainMat(:,1);
    trainMat = TrainMat(:,2:end);
    model = svmtrain(labels_train_mat, trainMat, '-c 1 -g 0.07 -b 1');
end
```

Funcția de testare a modelului SVM

```
function [label_test_mat] = testare_semnele_partitionate(model, test_number, cale_input)
    flag = 1;
    results_test =
    fopen(strcat(cale_input,sprintf('results_%d',test_number),sprintf('_test_coif3.txt')), 'wt');
    fisiere_evaluare = textread(sprintf('%s',cale_input, '/set_10_test_coif3.txt'), '%s', 'delimiter',
    '\n', 'whitespace', '');

    for i = 1:size(fisiere_evaluare,1)
        test_mat(flag,:) = reshape(csvread(strcat(cale_input,fisiere_evaluare{i})), 1, []);
        flag = flag + 1;
    end
    label_test_mat = test_mat(:,1);
    testMat = test_mat(:,2:end);

    [predict_label, accuracy, prob_values] = svmpredict(label_test_mat, testMat, model, '-b 1');
    fprintf(results_test,sprintf('%0.4f',accuracy));
    fprintf(results_test,'\n');

end
```