

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA
ZAVOD ZA ELEKTRONIČKE SUSTAVE I OBRADBU INFORMACIJA

**SEMINAR IZ NMDOS-a:
Potiskivanje šuma korištenjem wavelet transformacije i optimalnih filtara**

Franjo Tonković
Ivan Krsnik

Siječanj 2005.

Sadržaj:

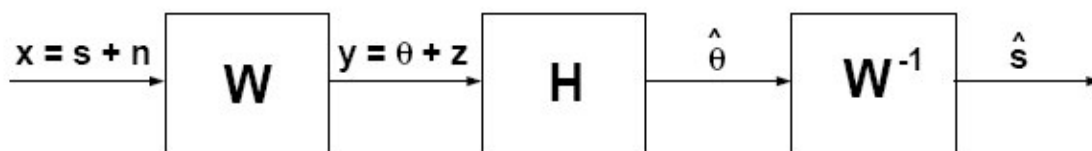
1. Uvod.....	2
2. Empirijsko Wiener filtriranje u wavelet domeni.....	3
3. Testiranje sustava.....	4
3.1 Simulacije u MATLABU:.....	4
3.1.1 Puštanje zašumljenog test signala kroz sustav.....	4
3.1.2 Simulacija sa zvučnim signalom.....	7
3.1.3 Propuštanje slike kroz naš sustav.....	8
3.2 Matlab kod korišten u testiranju.....	10
4. Zaključak.....	12

1 Uvod

Ovaj seminar iznosi **jednostavan način potiskivanja šuma** rezanjem wavelet koeficijenata, koristeći empirijski wienerov filter u wavelet domeni.

Wavelet transformacija je vrlo brzo postala dominantna u obradi signala i slike. Wavelet ekspanzija teži ka sakupljanjem energije u relativno mali broj velikih koeficijenata. To svojstvo kompaktnosti energije kod wavelet transformacije čini wavelet domenu privlačnom za obradu signala.

Osnovna ideja:



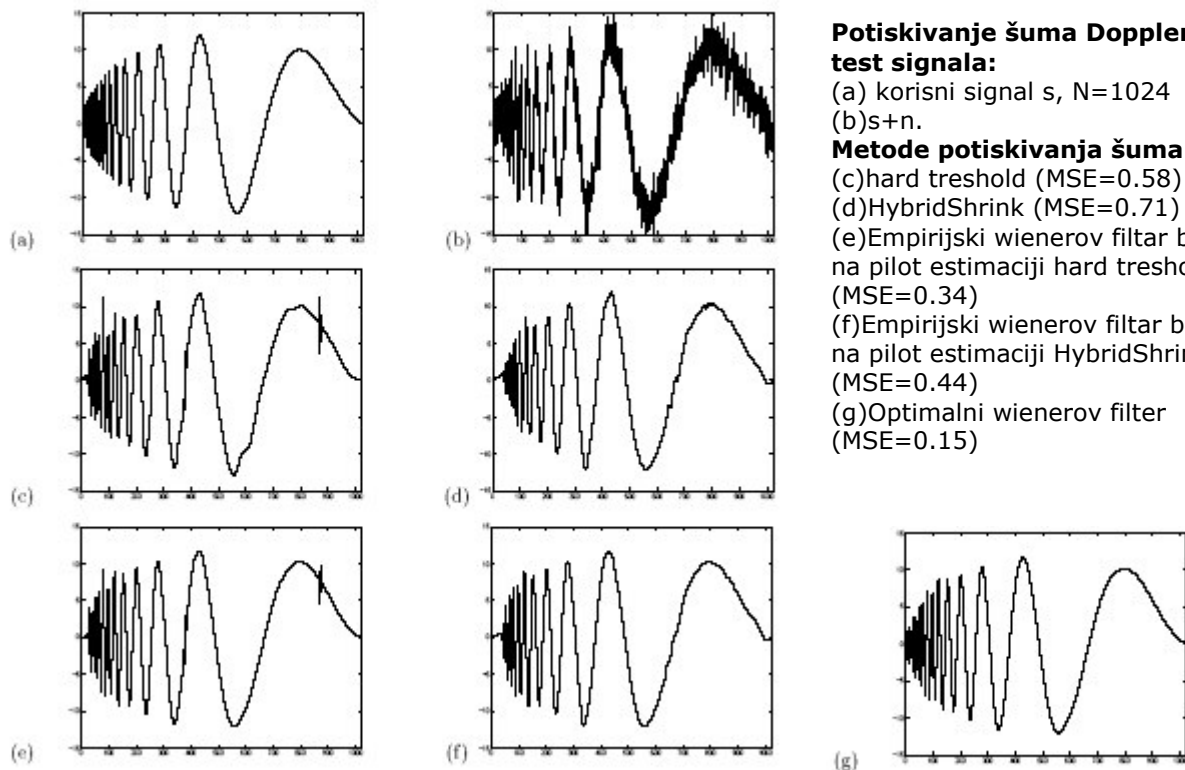
s-korisni signal (nepoznati deterministički kontinuirani vremenski signal s(t))

n-bijeli gausov šum

y-signal x u wavelet domeni

H-filer pomoću kojeg u wavelet domeni dobivamo wavelet koeficijente korisnog signala theta, iz zašumljenog y. Filter koji se pokazuje kao najbolji, u smislu minimizacije maksimalne srednje kvadratne pogreške (MSE) je wienerov filter H_w

$$h_w(i) \stackrel{def}{=} \frac{\theta^2(i)}{\theta^2(i) + \sigma^2}$$



Potiskivanje šuma Dopplerovog test signala:

(a) korisni signal s, N=1024

(b) s+n.

Metode potiskivanja šuma:

(c) hard treshold (MSE=0.58)

(d) HybridShrink (MSE=0.71)

(e) Empirijski wienerov filtar baziran na pilot estimaciji hard treshold (MSE=0.34)

(f) Empirijski wienerov filtar baziran na pilot estimaciji HybridShrink (MSE=0.44)

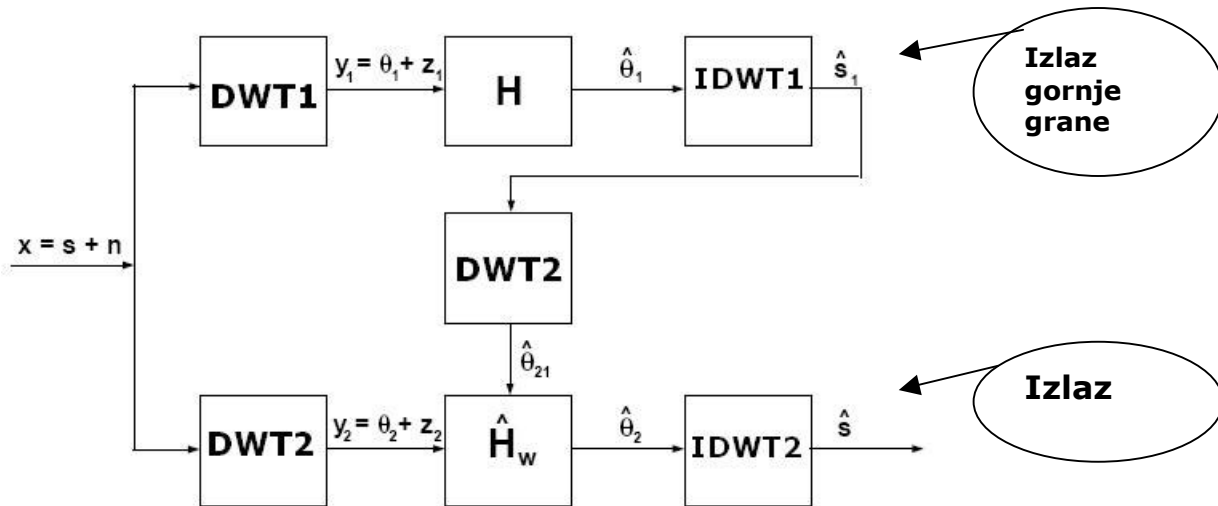
(g) Optimalni wienerov filter (MSE=0.15)

SLIKA 1: Usporedba različitih tipova filtera

2 Empirijsko Wiener filtriranje u wavelet domeni

IDEJA:

Imamo rezanje wavelet filtarskih koeficijenata temeljeno na wavelet transformaciji, a tada se taj rezultat uzima kao model (pilot estimacija) za dobivanje wienerovog filtra u odvojenoj (druga grana) wavelet domeni.



SLIKA 2: Shema sustava

Koriste se dvije wavelet transformacije DWT1 i DWT2. Wavelet u prvoj grani se koristi da se dobije standardna wavelet estimacija signala s_1 , bazirana na wavelet koeficijentima N_t . Taj signal s_1 predstavlja model signala koji se u drugoj grani koristi za filtriranje wienerovim filtrom u wavelet domeni, a za tu granu imamo drugačiju wavelet transformaciju-DWT2.

Kod wienerovog filtra treba znati i spektar šuma i spektar signala, a to nije uvijek slučaj i zna predstavljati problem. Mi računamo wavelet, a ne obične spektre.

Da dobijemo pouzdanu estimaciju svih koeficijenata signala $N_s = N_t + N_d$, signal s_1 transformiramo sa DWT2, da se dobije θ_{21} , koja se koristi za dizajn empirijskog wienerovog filtra u DWT2 domeni.

Razlika između DWT1 i DWT2 garantira da operator DWT2/DWT1 rasteže koeficijente sa θ_{21} u veći broj koeficijenata θ_{21} .

Ovdje opisana metoda potiskivanja šuma se zove *WienerShrink*, a ona je u usporedbi sa metodom praga fina metoda i daje bolje rezultate.

Klasično rezanje koeficijenata previše odreže wavelet koeficijente, a WienerShrink zbog rastezanja koeficijenata daje kvalitetnije rezultate.

3 Testiranje sustava:

Estimator	MSE (Doppler test signal)	MSE (Heavisine test signal)	MSE (Bumps test signal)	MSE (Blocks test signal)
Hard treshold	0.209	0.102	0.392	0.121
Wiener shrink (hard)	0.121	0.075	0.242	0.118
Hybrid shrink	0.274	0.094	0.439	0.226
Wiener shrink (Hybrid)	0.140	0.069	0.285	0.167
Ideal Wiener	0.069	0.037	0.183	0.064

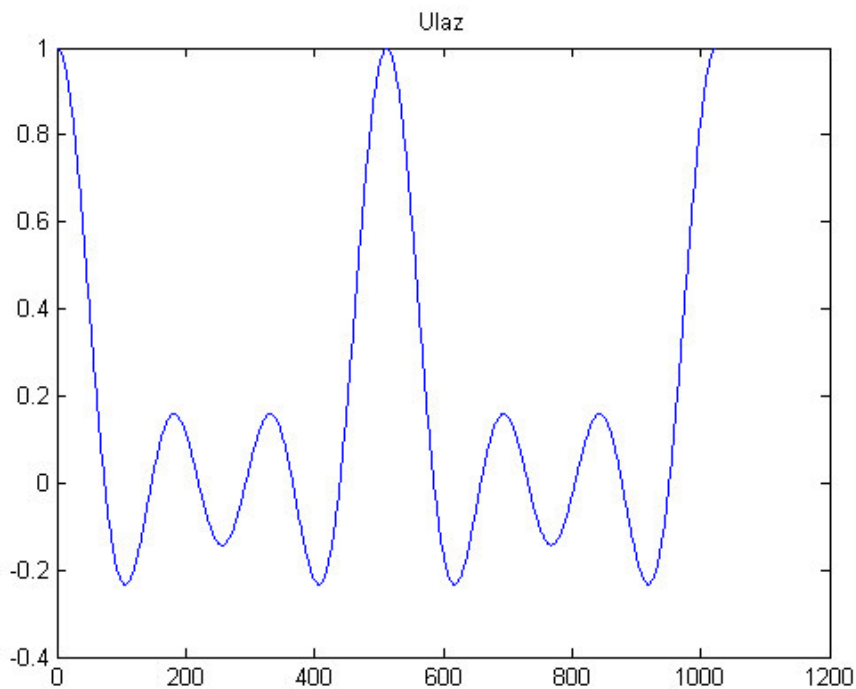
TABLICA 1: MSE za različite tehnike estimacije, uz 5 različitih test signala

3.1 Simulacije u MATLABU:

3.1.1 Puštanje zašumljenog test signala kroz sustav

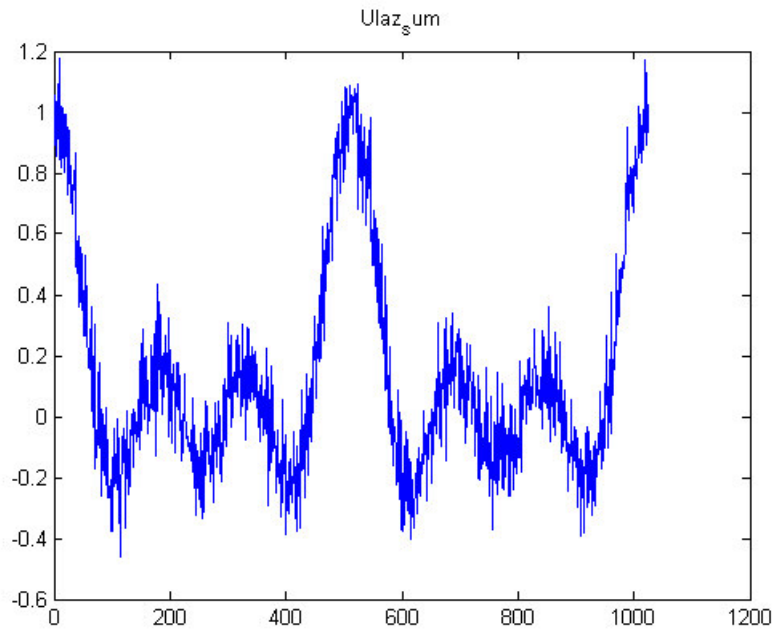
Signal sa kojime smo mi u matlabu proveli simulaciju, dobili smo ovako:

```
x = linspace(0,4*pi,1024);  
ulaz_orig=diric(x,7);
```



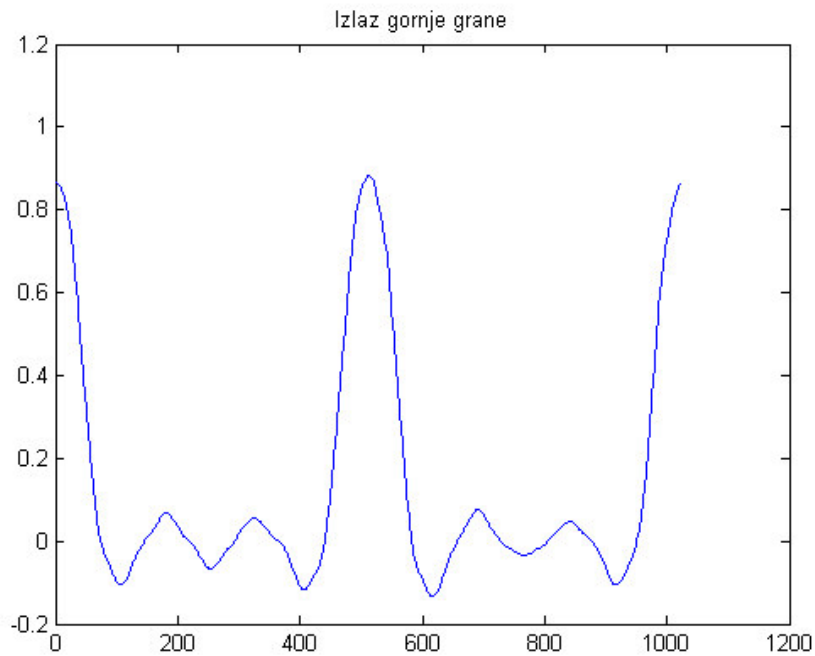
SLIKA 3: Čisti test signal na ulazu

Nakon što smo ga zašumili sa bijelim šumom, dobili smo sljedeći signal:

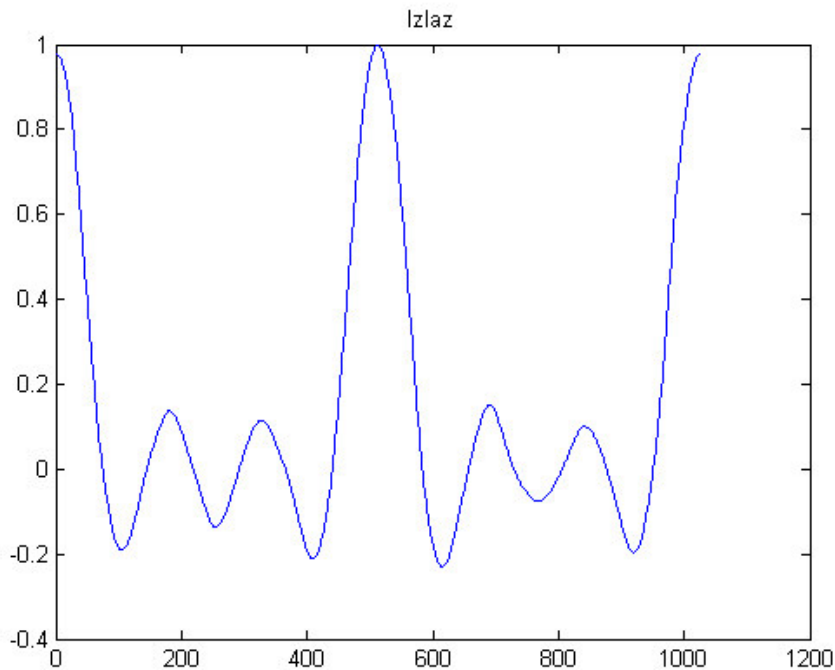


SLIKA 4: Zašumljeni test signal na ulazu

Nakon što smo isprobavanjem ustanovili da najbolje rezultate dobivamo za threshold faktor 0.4, uz $SNR_{ulaz} = 17.8982$, $SNR_{gornja_grana} = -25.7928$, $SNR_{izlaz} = -42.8576$, signali na označenim mjestima na glavnoj shemi izgledaju:



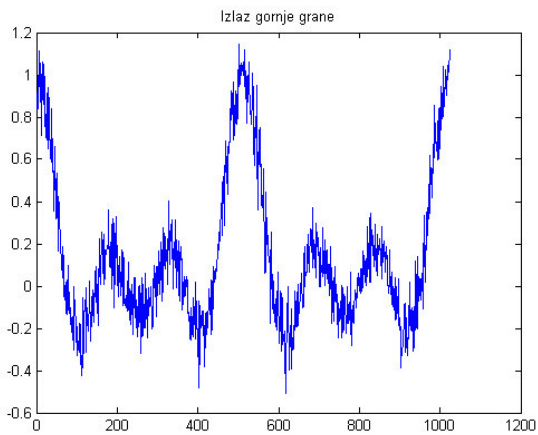
SLIKA 5: Signal na izlazu iz gornje grane



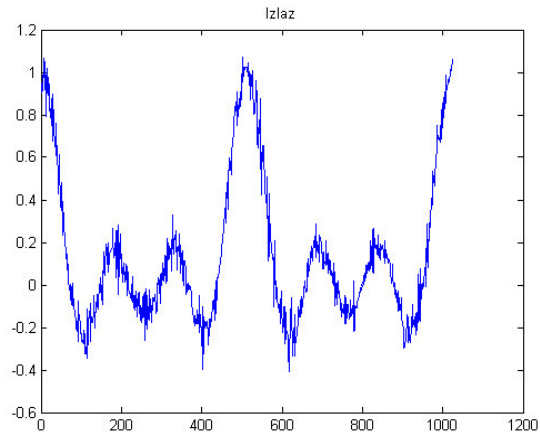
SLIKA 6: *Signal na izlazu*

Sada ćemo pokazati kakvi su rezultati dobiveni uz neadekvatno izabrane threshold faktore:

Threshold faktor=0.1, SNR_ulaz =18.2912db, SNR_gornja_grana = -22.9808db, SNR_izlaz =-32.2638db.

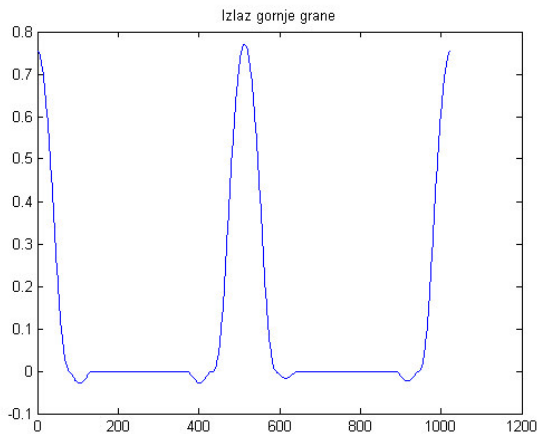


SLIKA 7: *Signal na izlazu iz gornje grane uz premali Threshold faktor*

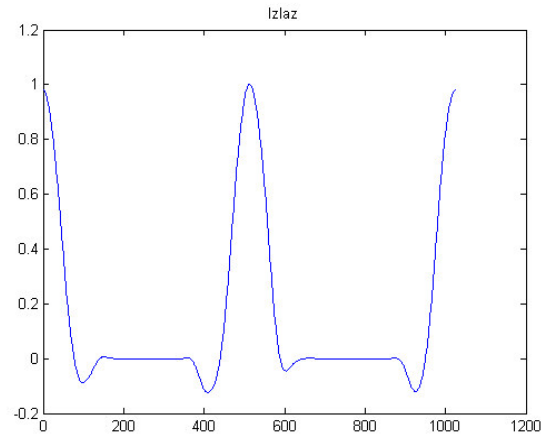


SLIKA 8: *Signal na izlazu uz premali Threshold faktor*

Threshold faktor=1, SNR_ulaz =17.7879db, SNR_gornja_grana =-14.9846db, SNR_izlaz =-22.9371db.



SLIKA 9: Signal na izlazu iz gornje grane uz preveliki Threshold faktor



SLIKA 10: Signal na izlazu uz preveliki Threshold faktor

3.1.2 Simulacija sa zvučnim signalom:

Kroz naš sustav propustili smo zašumljeni zvuk gitare, koristeći dosta veliki bijeli šum, uz treshold faktor = 0.014.

Na ulazu je odnos signal/šum: SNR_ulaz = -37.0543 db.
Nakon prolaska kroz gornju granu, odnos signal/šum je:
SNR_gornja_grana = -40.3923 db.
Odnos signal/šum na izlazu je: SNR_izlaz = -49.4227 db.

Smanjivši bijeli šum na ulazu 100 puta, uz treshold faktor od 0.006 dobili smo:

Na ulazu je odnos signal/šum: SNR_ulaz = -49.0201 db.
Nakon prolaska kroz gornju granu, odnos signal/šum je:
SNR_gornja_grana = -53.4234db.
Odnos signal/šum na izlazu je: SNR_izlaz = -59.6028 db.

3.1.3 Propuštanje slike kroz sustav

Ovakvi sustavi za odšumljivanje se uz neke modifikacije mogu koristiti i za filtriranje slika:



(a) Originalna slika



(b) Slika zašumljena sa bijelim šumom

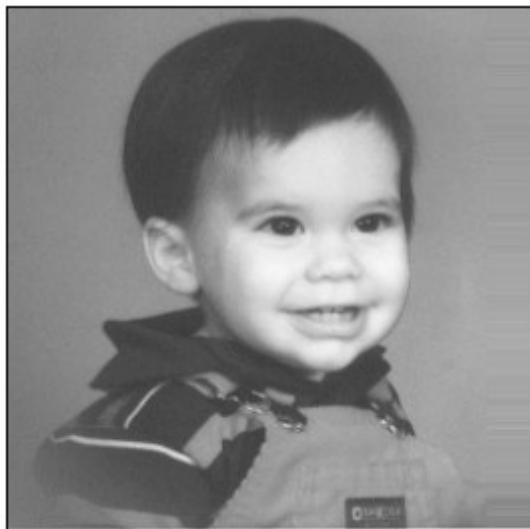


(c) Hard treshold filtriranje (MSE=92.44)

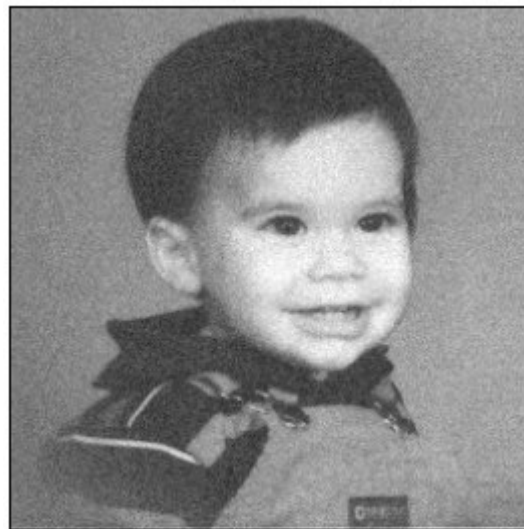


(d) WienerShrink filtriranje (MSE=61.37)

SLIKA 11: *Potiskivanje šuma slike*



(a) Originalna slika



(b) Slika zašumljena sa bijelim šumom



(c) Hard treshold filtriranje (MSE=74.93)



(d) WienerShrink filtriranje (MSE=48.07)

SLIKA 12: *Potiskivanje šuma slike*

3.2 Matlab kod korišten u testiranju

```
% (1) Propuštanje zašumljenog signala kroz sustav

%zadavanje čistog signala
x = linspace(0,4*pi,1024);
ulaz_orig=diric(x,7)

%zadavanje bijelog šuma
bijeli_sum=randn(1,1024);

%ulazni zašumljeni signal
ulaz=ulaz_orig+bijeli_sum/10;

figure,plot(ulaz_orig),title('Ulaz')
figure,plot(ulaz),title('Ulaz_sum')

%wavelet transformacija(prva) ulaznog signala
a=swt(ulaz,5,'db1');

%različite metode računanja optimalne razine praga
thr1 = thselect(a, 'rigrsure')
thr2 = thselect(a, 'heursure')
thr3 = thselect(a, 'sqtwolog')
thr4 = thselect(a, 'minimaxi')

%soft threshold filtriranje u wavlet domeni
a_tresh=wthresh(a,'s',thr);

%mi smo sami izabrali prag koji daje najbolje rezultate
a_tresh=wthresh(a,'s',0.01);

%inverzna wavelet transformacija
s1=iswt(a_tresh,'db1'); %signal na izlazu iz gornje grane

figure, plot(s1); title('Izlaz gornje grane')

%wavelet transformacija(druga) signala na izlazu iz gornje grane i
%ulaznog signala
a2=swt(s1,5,'sym4');
a3=swt(ulaz,5,'sym4');

%računanje wienerovog filtra
broj=a2.^2;
naz=broj+(var(ulaz))^2;
hw=broj./naz;

theta2=a3.*hw;

%inverzna wavelet transformacija
izlaz=iswt(theta2,'sym4'); %signal na izlazu
```

```

figure, plot(izlaz); title('Izlaz');

% (2) prupuštanje zašumljenog zvuka gitare kroz sustav

%učitavanje wav datoteke
[snd,fs, nbits] = wavread('gitara.wav');
snd=(snd(1:114880,1))';

%zadavanje bijelog šuma
bijeli_sum=randn(1,114880);
bijeli_sum=bijeli_sum/1000;

%zašumljeni signal gitare
ulaz=snd+bijeli_sum;

%zapisivanje signala u wav datoteku, da se kasnije može poslušati
wavwrite(ulaz,fs,nbits,'zasumljeni_ulaz')

%wavelet transformacija(prva) ulaznog signala
a=swt(ulaz,5,'sym5');

%različite metode računanja optimalne razine praga
% thr1 = thselect(a, 'rigrsure')
% thr2 = thselect(a, 'heursure')
% thr3 = thselect(a, 'sqtwolog')
% thr4 = thselect(a, 'minimaxi')

%hard threshold filtriranje u wavlet domeni
a_tresh=wthresh(a,'s',thr1);

%mi smo sami izabrali prag koji daje najbolje rezultate
a_tresh=wthresh(a,'h',0.006);

%inverzna wavelet transformacija
s1=iswt(a_tresh,'sym5'); %signal na izlazu iz gornje grane

%zapisivanje signala u wav datoteku, da se kasnije može poslušati
wavwrite(s1,fs,nbits,'izlaz_gornje_grane');

%wavelet transformacija(druga) signala na izlazu iz gornje grane i
%ulaznog signala
a2=swt(s1,5,'sym4');
a3=swt(ulaz,5,'sym4');

%računanje wienerovog filtra
broj=a2.^2;
naz=broj+(var(ulaz))^2;
hw=broj./naz;

theta2=a3.*hw;

%inverzna wavelet transformacija
izlaz=iswt(theta2,'sym4'); %signal na izlazu

%zapisivanje signala u wav datoteku, da se kasnije može poslušati

```

```
wavwrite(izlaz, fs, nbits, 'izlaz');
```

4 Zaključak

Ovdje prikazana *WienerShrink* metoda za potiskivanje šuma pomoću waveleta je hibridna metoda standardnog wavelet filtriranja pomoću praga i empirijskih Wienerovih tehnika filtriranja. Rezultat je filter u wavelet domeni koji se približio optimalnom Wienerovom filteru. Iako je kod navedene metode potrebno dva puta računati wavelet transformacije, rezultati koji se tako dobivaju znatno su bolji od klasičnih metoda, tako da to ne bi smjelo stvarati veći problem.

Mi smo *WienerShrink* metodu realizirali u Matlabu. Realizirali smo Wienerov filter koristeći pilot estimaciju koju smo pak realizirali tako da smo primjenili fiksni prag kod rezanja koeficijenata u wavelet domeni. Metodu je moguće primijeniti za potiskivanje šuma kod signala, zvuka i slike. *WienerShrink* metoda u svim tim slučajevima daje bolje rezultate od klasičnog wavelet filtriranja pomoću praga.

Daljnje usavršavanje *WienerShrink* metode može ići u smjeru određivanja najoptimalnijih wavelet transformacija za pojedinu granu