

Sveučilište u Zagrebu  
Fakultet elektrotehnike i računarstva  
Zavod za elektroničke sustave i obradbu informacija

Projekt iz kolegija  
*„Napredne metode digitalne obradbe signala“*

# **Sažeto uzorkovanje (Compressive sensing)**

**Davor Bonači**

0036409319

Zagreb, 2008.

# SADRŽAJ

Sadržaj.....	2
1. Uvod .....	3
2. Problem .....	4
3. Rješenje .....	6
4. Primjeri .....	7
5. Zaključak.....	10
Literatura.....	11
Kod u Matlabu .....	12

# 1. UVOD

Sažeto uzorkovanje (engl. *compressed sensing*, *compressive sensing*) je nova metoda otipkavanja signala sa smanjenim brojem uzoraka s obzirom na Shannon-Nyquistov teorem. Naime, taj teorem otipkavanja kaže da je potrebno otipkavati signal najmanje dvostrukom frekvencijom od najveće frekvencijske komponente dostupne u signalu. Standardno, signal se otipkava tom dvostrukom frekvencijom, sprema se u kratkotrajnu memoriju, komprimira nekim algoritmom i takav, sažeti signal, sprema u dugotrajnu memoriju ili šalje komunikacijskim kanalom. Ovakva metoda (engl. *transform coding*) zahtjeva jako visoku frekvenciju otipkavanja, što otežava proizvodnju.

Sažeto uzorkovanje pruža drugačiji pristup – otipkavanje „komprimiranog signala“ s frekvencijom otipkavanja znatno nižom od one koju zahtjeva teorem otipkavanja. Naravno, ova metoda se može primijeniti samo na određenim klasama signala i u određenim uvjetima. Ovaj projekt opisuje metodu sažetog uzorkovanja i eksperimentalno provjerava učinkovitost na određenim primjerima.

## 2. PROBLEM

Promatrat će se realni, konačni, jednodimenzionalni, vremenski-diskretni signal  $x$ . Višedimenzionalni signali mogu se donekle analizirati ovom metodom tako da se njihovi uzorci smjeste u jedan vektor-stupac. Neka signal  $x$  ima  $N$  uzoraka.

Signal  $x$  možemo promatrati kao vektor, koji je element vektorskog prostora  $\mathcal{R}^N$ . Uzorci signala su tada smješteni u vektor-stupac  $x$ . Taj vektor može se izraziti kao linearna kombinacija vektora baze vektorskog prostora:

$$x = \sum_{i=1}^N s_i \psi_i, \text{ ili, ekvivalentno, } x = \psi s, \quad (1)$$

gdje skup vektora  $\{\psi_i, i \in \{1, 2, 3, \dots, N\}\}$  čini neku bazu vektorskog prostora  $\mathcal{R}^N$ , a  $\psi$  je kvadratna matrica reda  $N$  koja sadrži navedene vektore baze kao svoje stupce. Koeficijenti  $s_i$ , složeni u vektor-stupac  $s$ , predstavljaju razlaganje vektora  $x$  po bazi vektorskog prostora. Jasno je da su vektori  $x$  i  $s$  ekvivalentne reprezentacije signala, s tim da je  $x$  reprezentacija u vremenu ili prostoru, a  $s$  u  $\psi$ -domeni.

**Definicija.** Za vremenski diskretni signal  $x$  od  $N$  uzoraka kažemo da je **K-rijedak** ako postoji neka baza vektorskog prostora  $\mathcal{R}^N$ , takva da se signal  $x$  može zapisati kao linearna kombinacija samo  $K$  vektora baze (od  $N$  mogućih).

Tada vektor  $s$ , prema raspisu (1), sadržava  $N-K$  koeficijenata koji su jednaki nuli i  $K$  koeficijenata različitih od nule. Metodu je moguće efikasno upotrijebiti za signale za koje je  $K \ll N$ . Pokazuje se da većina signala s kojima se susrećemo u praksi približno zadovoljava gornji uvjet. Naime, najčešće postoji baza u kojoj se signal može dobro aproksimirati samo s nekoliko vektora baze. Tada vektor  $s$ , prema raspisu (1), sadržava samo  $K$  značajnih (velikih) koeficijenata, a ostalih  $N-K$  koeficijenata je približno jednako nuli. Za takav signal kažemo da je kompresibilan.

Standardna metoda (engl. transform coding) uzorkovanja i kompresije signala, prema Shannon-Nyquistovom teoremu, temelji se na kompresibilnosti signala. Takva metoda otipkava signal  $x$  u  $N$  uzoraka. Zatim se signal raspisuje prema formuli (1). Značajni koeficijenti i njihove lokacije je kodiraju u novi, komprimirani signal, a mali koeficijenti odbacuju. Nedostaci ove metode su:

1. broj  $N$  može biti velik, iako je konačni  $K$  mali,
2. svih  $N$  koeficijenata u raspisu (1) mora biti izračunato, iako će se njih  $N-K$  odbaciti,
3. lokacije značajnih koeficijenata se moraju kodirati u rezultatni signal.

Sažeto uzorkovanje nudi rješenje svih ovih nedostataka, tako da se direktno uzorkuje komprimirani signal.

Pretpostavimo da je moguće mjeriti neka svojstva signala  $x$ . Preciznije rečeno, pretpostavimo da možemo mjeriti sličnost vektora  $x$  s proizvoljnim vektorima, odnosno da

možemo računati skalarni produkt vektora  $x$  s  $M$  proizvoljnih vektora  $\varphi_i$  duljine  $N$ . Složimo navedene mjerne vektore kao retke u mjernu matricu  $\Phi$ .

Dakle, možemo definirati novi, komprimirani, signal:

$$y = \Phi x = \Phi \psi s = \Theta s. \quad (2)$$

Signal  $y$  predstavlja komprimiranu sliku signala  $x$ , ali sastoji se od  $M$  uzoraka, čime je uz  $M < N$  postignuto sažimanje. Mjerenje nije adaptivno, tj. matrica  $\Phi$  je fiksna i ne ovisi o ulaznom signalu.

Problem sažetog uzorkovanja možemo razbiti u dva manja problema:

1. generiranje mjerne matrice  $\Phi$ , tako da važna informacija bude sačuvana, tj. da svih  $K$  koeficijenata u raspisu (1) različitih od nule bude sačuvano u mjernom rezultatu  $y$ ,  
i
2. rekonstrukcija signala  $x$  iz signala  $y$  koji sadrži samo  $M$  uzoraka.

### 3. RJEŠENJE

Prvi problem je generirati mjernu matricu  $\Phi$ , koja se sastoji od  $M$  redaka i  $N$  stupaca, ali na način da svih  $K$  važnih koeficijenata u raspisu (1) bude sačuvano u mjernom rezultatu  $y$ , prema zapisu (2). To je moguće uz uvjet  $M \geq K$ .

Pokazuje se da se dobra mjerna matrica  $\Phi$  dobije jednostavnim generiranjem njenih elemenata generatorom slučajnih brojeva. Uzmimo da elementi mjerne matrice budu Gaussove slučajne varijable sa srednjom vrijednošću jednakom nuli i varijancom  $1/N$ . Tada je velika vjerojatnost generiranja dobre mjerne matrice. Pojašnjenje ove činjenice moguće je naći u literaturi pod brojem 1.

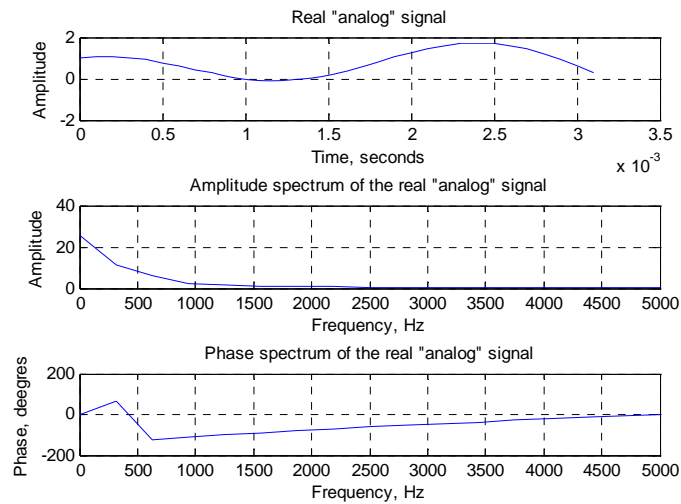
Preostao je problem rekonstrukcije vektora  $x$  ili, ekvivalentno, vektora  $s$  iz vektora  $y$  prema jednadžbi (2). Vidljivo je da se tu radi u sustavu od  $M$  linearnih jednadžbi s  $N$  nepoznanica. Takav sustav može nemati rješenje, u slučaju da su jednadžbe u kontradikciji, ili može imati više rješenja. Ako je mjerna matrica dobro generirana, kontradikcija se ne može dogoditi. Ostaje za odabrati pravo rješenje od dobivenih rješenja. Pokazuje se da optimizacija bazirana na  $l_1$  normi daje potpunu rekonstrukciju:

$$\hat{s} = \operatorname{argmin} \|s'\|_1, \quad \Phi s' = y. \quad (3)$$

Dobiva se potpuna rekonstrukcija za  $K$ -rijetke signale i bliska aproksimacija za kompresibilne signale. Vremenska složenost ovakve optimizacije je  $O(N^3)$ . Detaljniju raspravu o drugim normama i razlozima moguće je naći u literaturi pod brojem 1.

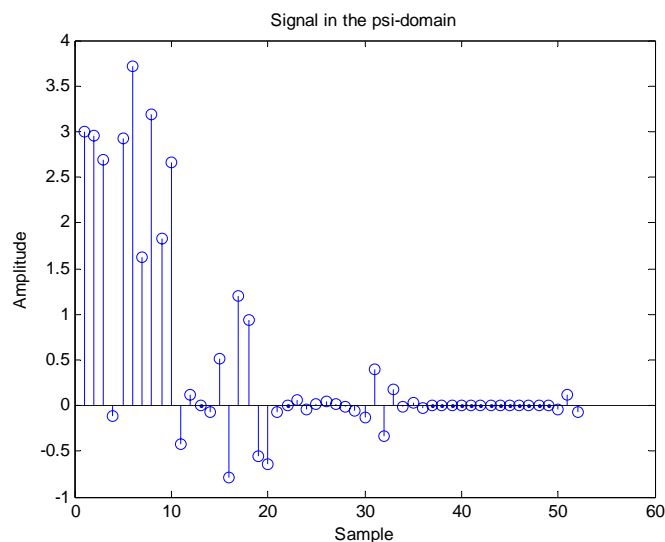
## 4. PRIMJERI

Provjerimo, za početak, potpunu rekonstrukciju  $K$ -rijetkog signala. Generirajmo 32 uzorka signala koji je zbroj sinusa frekvencije 150 Hz i kosinusa frekvencije 400 Hz, frekvencijom otipkavanja od 10 kHz. Taj signal i njegov Fourierov spektar su prikazani na slici 1.



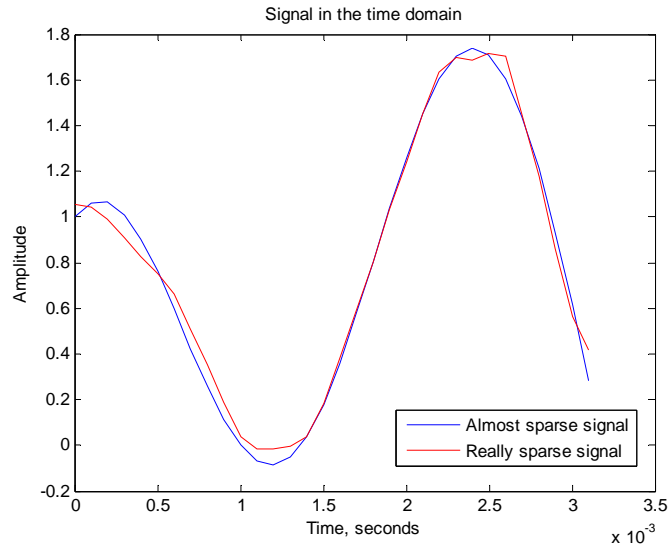
Slika 1. Početni signal i njegov Fourierov spektar.

Transformirajmo taj signal u wavelet domenu pomoću trirazinske *sym4* diskretne wavelet transformacije. Koeficijenti (aproximacije i detalja) su prikazani na slici 2.



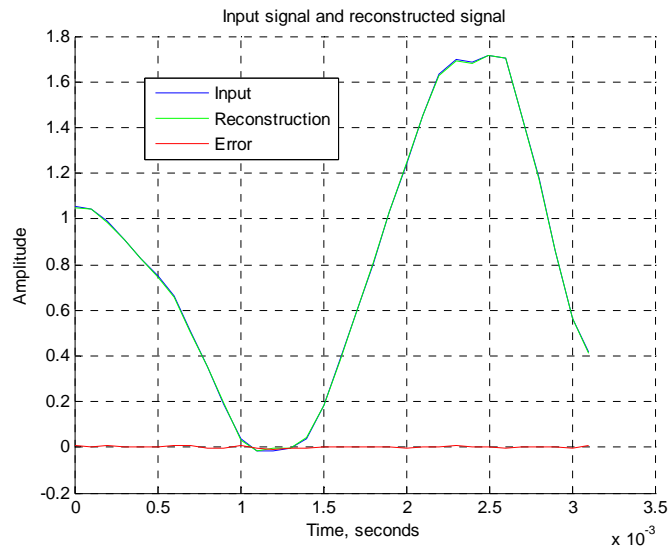
Slika 2. Signal u wavelet domeni.

Vidimo da taj signal nije  $K$ -rijedak. Poništimo koeficijente manje od 0,5 i rekonstruirajmo signal. Početni i rekonstruirani signal su prikazani na slici 3.



Slika 3. Početni i rekonstruirani signal.

Vidimo da rekonstruirani signal nije identičan početnom, ali jest  $K$ -rijedak. Provedimo postupak mjerenja pomoću slučajne mjerne matrice s  $M = 27$  i zatim algoritam rekonstrukcije. Rekonstrukcijski signal i ulazni signal su prikazani na slici 4.

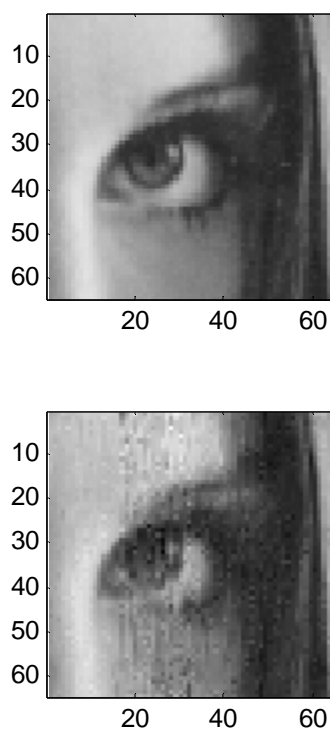


Slika 4. Rekonstruirani i ulazni signal.

Najveća razlika rekonstruiranog signala i ulaznog iznosi 0,0083 što je posljedica numeričke pogreške. Na temelju dobivenih rezultata zaključujemo da je ostvarena potpuna rekonstrukcija.

Provjerimo sada algoritam na stvarnom signalu – slici. Ulazna slika je crno-bijela 8-bitna, veličine 64x64 piksela i prikazana je na slici 5. Algoritam je proveden pomoću identične wavelet trofazinske *sym4* transformacije. Zbog uštede na vremenu, algoritam je zasebno proveden na četiri segmenta slike veličine 32x32 piksela uz  $M = 600$ . Rezultat rekonstrukcije je prikazan na slici 5.



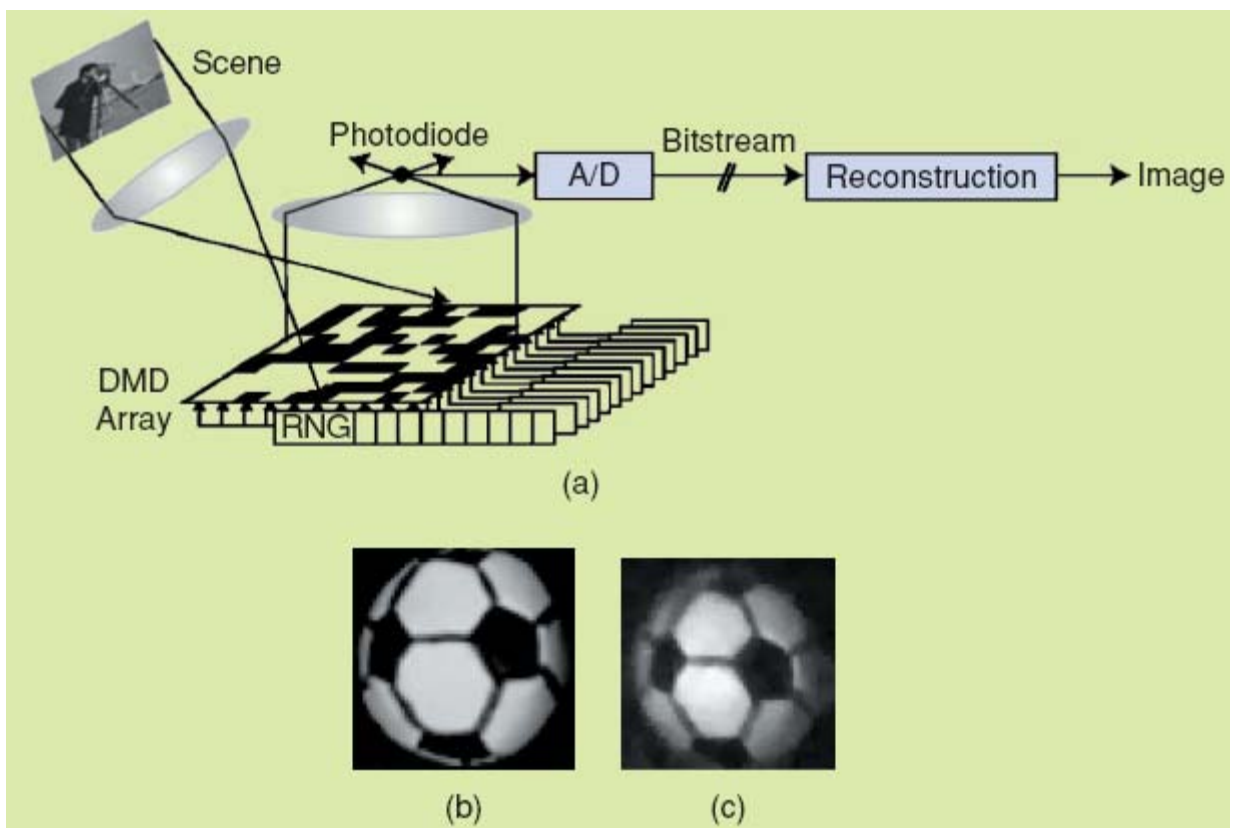


Slika 5. Početna i rekonstruirana slika.

Vidljivo je da rekonstrukcija nije potpuna, nego samo djelomična. Razlog tome je što slika u odabranoj domeni nije  $K$ -rijetka.

## 5. ZAKLJUČAK

Iz navedenih primjera može se zaključiti da sažeto uzorkovanje nudi mogućnost otipkavanja signala uz smanjeni broj uzoraka. Brzina izvođenja rekonstrukcije trenutno predstavlja problem za primjenu metode u stvarnom vremenu. Ako se metoda ne primjenjuje u stvarnom vremenu, onda ne donosi mnogo u odnosu na standardnu metodu (engl. transform coding). Kod digitalnih fotoaparata moguća je upotreba ove metode. Snimanje se radi u stvarnom vremenu, a rekonstrukcija ne predstavlja velik problem. Moguće bi bilo konstruirati fotoaparat s velik brojem „mega-piksela“ koristeći samo jednu fotodiodu prema donjoj slici.



Slika (a) prikazuje digitalni fotoaparat koji primjenjuje metodu sažetog uzorkovanja. Umjesto velikog broja foto-dioda, dovoljno je samo jedna koja na slučajni mjestima uzorkuje sliku. Slika (b) prikazuje nogometnu loptu fotografiranu sa standardnim fotoaparatom, a slika (c) pomoću fotoaparata (a). To je slika s 4096 piksela rekonstruirana od samo 1600 slučajnih mjerenja slike (b).

## LITERATURA

1. Richard G. Baraniuk, „Compressive Sensing“, IEEE Signal Processing Magazine, July 2007, 118-124

## KOD U MATLABU

```

% Napredne metode digitalne obradbe signala - Projekt
% Compressed sensing - Potpuna rekonstrukcija uz smanjeni broj uzoraka
% Davor Bonaci, davor.bonaci@fer.hr

%% 1. Initialization
clear all;
close all hidden;
pack;

fs = 1e4; % Sampling frequency, in Hz
N = 32; % Number of samples
M = 27; % Number of measurements
f1 = 150; % Sine frequency one, in Hz
f2 = 400; % Sine frequency two, in Hz

%% 2. Input signal and its spectrum
t = [0 : 1/fs : (N-1)/fs];
x = sin(2*pi*f1*t) + cos(2*pi*f2*t);
f = fs * (0 : N/2) / N;
X = fft(x);

figure,
subplot(3, 1, 1), plot(t, x), grid, title('Real "analog" signal'),
xlabel('Time, seconds'), ylabel('Amplitude'),
subplot(3, 1, 2), plot(f, abs(X(1 : N/2+1))), grid,
title('Amplitude spectrum of the real "analog" signal'),
xlabel('Frequency, Hz'), ylabel('Amplitude'),
subplot(3, 1, 3), plot(f, rad2deg(angle(X(1 : N/2+1)))), grid,
title('Phase spectrum of the real "analog" signal'),
xlabel('Frequency, Hz'), ylabel('Phase, deegres');

%% 3. Computing representation of the input signal
%% in the psi-domain and the psi-matrix
%psi_inv = WavMat(wfilters('sym4'), N, 3);
%psi = inv(psi_inv);
%psi_inv = dctmtx(N)';
%psi = dctmtx(N);

psi_inv = WaveMtx(N, 3, 'sym4');
psi = pinv(psi_inv);

s = (psi_inv * x.').';
figure, stem(s), hold on;

%% Generating sparse signal
s_sparse = s;
s_sparse(find(abs(s_sparse) < 0.5)) = 0;

stem(s_sparse, 'r'), title('Signal in the psi-domain'),
xlabel('Sample'), ylabel('Amplitude'),
legend('Almost sparse signal', 'Really sparse signal');

x_sparse = (psi * s_sparse.').';

figure, plot(t, x), hold on, plot(t, x_sparse, 'r'),
title('Signal in the time domain'), xlabel('Time, seconds'),
ylabel('Amplitude'), legend('Almost sparse signal', 'Really sparse signal');

x = x_sparse;
s = s_sparse;

%% 4. Computing measurement matrix and performing measurements
randn('state', 0);
fi = 1/N .* randn(M, N);
y = fi * x. ';

%% 5. Reconstructing initial signal
theta = fi * psi;

cvx_begin
variable s_est(length(s), 1);

```

```

    minimize( norm(s_est, 1) );
    subject to
        theta * s_est == y;
cvx_end

x_est = (psi * s_est).';

%% 6. Verification
max_error_s = max(abs(s - s_est.))
max_error_x = max(abs(x - x_est))

figure, hold on, grid, title('Input signal and reconstruction in the psi-domain'),
stem(s, 'b'), stem(s_est, 'g'), stem(s_est.' - s, 'r'),
legend('Input', 'Reconstruction', 'Error');

figure, grid, hold on, title('Input signal and reconstructed signal'),
xlabel('Time, seconds'), ylabel('Amplitude'),
plot(t, x, 'b'), plot(t, x_est, 'g'), plot(t, x-x_est, 'r'),
legend('Input', 'Reconstruction', 'Error');

```

---

```

% Napredne metode digitalne obradbe signala - Projekt
% Compressed sensing - Potpuna rekonstrukcija uz smanjeni broj uzoraka
% Davor Bonaci, davor.bonaci@fer.hr

%% 1. Initialization
clear all;
close all hidden;
pack;

N = 1024;
M = 600;

%% 2. Generation of the matrixes
%psi_inv = dctmtx(N)';
%psi = dctmtx(N);
psi_inv = WaveMtx(N, 3, 'sym4');
psi = pinv(psi_inv);

N2 = size(psi);
N2 = N2(2);

randn('state', 0);
fi = 1/N .* randn(M, N);

theta = fi * psi;

%% 3. Read input image
im_rgb = imread('lenna_oko.tiff');
im_gray = rgb2gray(im_rgb);
image = double(reshape(im_gray, 1, prod(size(im_gray))));
figure, imshow(im_gray);
len = length(image);

%% 4. Compress and decompress
image_est = [];
for i = 1 : floor(len / N)
    signal = image((i - 1) * N + 1 : i*N);

    % Performing measurements
    y = fi * signal. ';

    % Reconstructing initial signal
    cvx_begin
        variable s_est(N2, 1);
        minimize( norm(s_est, 1) );
        subject to
            theta * s_est == y;
    cvx_end

    signal_est = (psi * s_est). ';

    image_est = [ image_est signal_est ];
end

%% 6. Verification
im_gray_est = uint8(round(reshape(image_est, sqrt(len), sqrt(len))));
max_error_signal = max(abs(signal - signal_est))

```

```
figure,  
subplot(2, 1, 1), subimage(im_gray),  
subplot(2, 1, 2), subimage(im_gray_est);
```

---

```
function WavMtx = WaveMtx(N, D, wavelet)  
%WAVEMTX Compute wavelet transform matrix.  
% M = WAVEMTX(N, D, wavelet) returns the N-by-M wavelet transform matrix.  
%  
% Example  
% -----  
% M = WaveMtx(32, 3, 'sym4');  
% output = M * input;  
for i = 1 : N  
    ulaz = zeros(1, N);  
    ulaz(i) = 1;  
  
    izlaz = wavedec(ulaz, D, wavelet);  
  
    WavMtx(1 : length(izlaz), i) = izlaz;  
end  
return
```