

Sažimajuće očitavanje signala

Grozdanić, Karlo

Master's thesis / Diplomski rad

2023

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Science / Sveučilište u Zagrebu, Prirodoslovno-matematički fakultet**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:217:303523>

Rights / Prava: [In copyright](#)/[Zaštićeno autorskim pravom.](#)

Download date / Datum preuzimanja: **2024-09-12**



Repository / Repozitorij:

[Repository of the Faculty of Science - University of Zagreb](#)



SVEUČILIŠTE U ZAGREBU
PRIRODOSLOVNO–MATEMATIČKI FAKULTET
MATEMATIČKI ODSJEK

Karlo Grozdanić

SAŽIMAJUĆE OČITAVANJE SIGNALA

Diplomski rad

Voditelj rada:
doc. dr. sc. Azra Tafro

Zagreb, veljača, 2023.

Ovaj diplomski rad obranjen je dana _____ pred ispitnim povjerenstvom u sastavu:

1. _____, predsjednik
2. _____, član
3. _____, član

Povjerenstvo je rad ocijenilo ocjenom _____.

Potpisi članova povjerenstva:

1. _____
2. _____
3. _____

Svim dragim ljudima

Sadržaj

| | |
|-----------------------------------------------------------------|-----------|
| Sadržaj | iv |
| Uvod | 1 |
| 1 Osnovni pregled sažimajućeg očitavanja | 3 |
| 1.1 Standardni pristupi kompresiji signala | 3 |
| 1.2 Sažimajuće očitavanje | 5 |
| 1.3 Zamjena normi | 7 |
| 2 Svojstva sažimajućeg očitavanja | 9 |
| 2.1 Egzaktna rekonstrukcija i minimalni broj mjerenja | 9 |
| 2.2 Greška rekonstrukcije | 12 |
| 2.3 Izbor matrica | 13 |
| 2.4 Algoritmi sažimajućeg očitavanja | 16 |
| 3 Primjeri | 19 |
| 3.1 Korištene transformacije | 20 |
| 3.2 Proširenje teorije | 24 |
| 3.3 Usporedba klasične i poboljšane metode | 29 |
| 4 Dodatak | 41 |
| Bibliografija | 61 |

Uvod

Problem sažimanja podataka možemo naći u skoro svim slučajevima slanja podataka. Tako se jedan od prvih primjera sažimanja podataka javio 1838. godine, kada je osmišljen Morseov kod. U Morseovom kodu svako slovo predstavljeno je nizom kratkih i dugih signala određene duljine, najčešće 3 ili 4 signala. Kako bi prosječna poruka bila što kraća, najčešće korištena slova engleskog jezika, *e* i *i*, reprezentirana su samo jednim, odnosno dvoma kratkim signalima.

U ovom radu bavit ćemo se sažimajućim očitavanjem, jednom od metoda sažimanja signala. Metoda je nastala ranih 2000-ih godina, suradnjom Terencea Taoa i Emmanuela Candèsa. Signalom smatramo svaku pojavu koju možemo izmjeriti i reprezentirati vektorom, a to može biti temperatura i vlažnost zraka kroz vrijeme, razina šećera u krvi, zvučni zapis, fotografija i sl.

U najčešćem slučaju sažimanje signala dolazi tek nakon što je taj signal u potpunosti izmjeren. Problem nastaje u trenucima kada nije lako doći do velikog broja mjerenja. Jedan takav primjer je liječnički pregled pacijenta magnetskom rezonancom gdje snimanje može trajati dulje od 30 minuta, vrijeme u kojem pacijent mora biti što je više moguće miran. Predstavljena metoda omogućit će egzaktnu ili približnu rekonstrukciju signala iz značajno manjeg broja mjerenja. Glavni uvjet koji će signal trebati ispuniti je rijetkost, svojstvo signala da se sastoji od malog broja temeljnih signala.

U prvom poglavlju formulirat ćemo problem sažimajućeg očitavanja, usporediti ga sa standardnim pristupima kompresiji podataka i ponuditi intuiciju zašto tako formuliran problem ima smisla. U drugom poglavlju predstaviti ćemo razrađenu teoriju sažimajućeg očitavanja, uvjete pod kojima će postupak biti uspješan te dva algoritma koji taj problem rješavaju. Treće poglavlje bavit će se primjenom sažimajućeg očitavanja u svrhu rekonstrukcije slika te će biti uvedena poboljšanja na standardni postupak koje dovode do značajno boljih rezultata.

Poglavlje 1

Osnovni pregled sažimajućeg očitavanja

Počinjemo s osnovnim pregledom metode kako bi čitatelju kasnije bilo lakše pratiti detaljnu razradu.

Osnovni objekt koji ćemo koristiti je signal reprezentiran vektorom iz \mathbb{R}^n . Taj će signal najčešće predstavljati zvučni zapis ili sliku. Dvodimenzionalne signale, kao što su slike, vektoriziramo. Zbog jednostavnosti koncentrirat ćemo se na \mathbb{R}^n , no rezultati analogni onima koje ćemo pokazati vrijedit će i na \mathbb{C}^n .

Definicije iz ovog poglavlja preuzete su iz [10], dok razrada problema većinom slijedi [23] i [13].

1.1 Standardni pristupi kompresiji signala

Definicija 1.1.1. l_0 normu definiramo kao $\|\cdot\|_0 : \mathbb{F}^n \rightarrow \mathbb{R}$, $\|\mathbf{x}\|_0 = \sum_{i=1}^n \mathbb{1}_{x_i \neq 0}$ za $\mathbf{x} = (x_i)_{i=1:n}$.

l_0 norma je broj elemenata vektora različitih od nula. Uočimo da je stvarno norma.

Definicija 1.1.2. Vektor $\mathbf{x} \in \mathbb{R}^n$ je **k-rijedak** ako ima najviše k elemenata različitih od 0, odnosno ako $\|\mathbf{x}\|_0 \leq k$.

Definicija 1.1.3. $\Sigma_k := \{\mathbf{x} \in \mathbb{R}^n, \|\mathbf{x}\|_0 \leq k\}$ je skup svih k -rijetkih vektora.

Definicija 1.1.4. Nosač vektora $\mathbf{x} = (x_i)_{i \leq n} \in \mathbb{R}^n$ je skup indeksa njegovih ne-nul elemenata, tj

$$\text{supp}(\mathbf{x}) := \{i : x_i \neq 0, i = 1, \dots, n\}.$$

Uvedimo i kompresibilnost signala kao svojstvo signala da bude dobro procijenjen nekim rijetkim signalom. Ovisno o objektu kojeg modeliramo, može biti poznato da je mjereni signal stvarno rijedak. Kad to neće biti slučaj, vrlo često će signal biti kompresibilan u nekoj bazi.

Definicija 1.1.5. *Neka $1 \leq p < \infty$ i $r > 0$. $\mathbf{x} \in \mathbb{R}^n$ je **p-kompresibilan s konstantom C i brzinom r** ako*

$$\sigma_k(\mathbf{x})_p := \min_{\hat{\mathbf{x}} \in \Sigma_k} \|\mathbf{x} - \hat{\mathbf{x}}\|_p \leq C \cdot k^{-r},$$

za svaki $k \in \{1, \dots, n\}$.

Jedan način kompresije signala \mathbf{x} bio bi pronalaženje ortonormirane baze (ONB) Ψ u kojoj bi signal \mathbf{x} imao rijetku reprezentaciju, odnosno $\mathbf{s} = \Psi\mathbf{x}$, gdje je \mathbf{s} k -rijedak vektor. Kasnije se \mathbf{x} može rekonstruirati iz $\mathbf{x} = \Psi^*\mathbf{s}$. Pri takvoj kompresiji ne bi došlo do gubitka informacija.

Ako smo spremni na određen gubitak informacija u svrhu veće kompresije, za danu bazu Ψ i vektor \mathbf{s} tako da je $\mathbf{s} = \Psi\mathbf{x}$, može se odabrati samo k najvećih koeficijenata od \mathbf{s} , a sve ostale postaviti na 0. Takav način kompresije često je korišten, uz klasične odabire ONB kao što su diskretna Fourierova transformacija (DFT), valić baze, ...

Uočimo određenu rasipnost koja se događa pri dohvaćanju signala. Potrebno je očitati cijeli signal, izračunati potrebne koeficijente, manji broj njih zadržati, a ostatak odbaciti. Glavna motivacija sažimajućeg očitavanja je zaobići tu rasipnost i u trenutku očitavanja dohvatiti manji broj podataka iz kojih će se, kasnije, moći rekonstruirati originalni signal s dovoljnom točnošću.

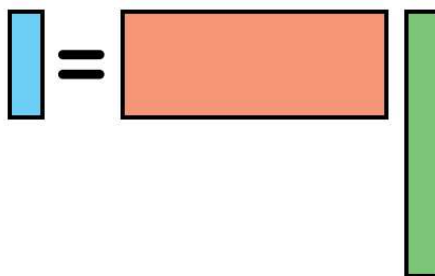
1.2 Sažimajuće očitavanje

Neka je $\mathbf{x} \in \mathbb{R}^n$ originalni signal koji želimo očitati, Ψ ortonormirana baza $\mathbb{R}^{n \times n}$ i $\mathbf{s} = \Psi \mathbf{x}$ prikaz signala u bazi koji je k -rijedak. Umjesto početnog signala \mathbf{x} , pretpostavimo da imamo m slučajnih projekcija oblika $y_i = \langle \mathbf{x}, \hat{\phi}_i \rangle$, gdje je m mnogo manji od n .

Dani problem je kako iz tih slučajnih projekcija rekonstruirati početni signal, točno ili približno.

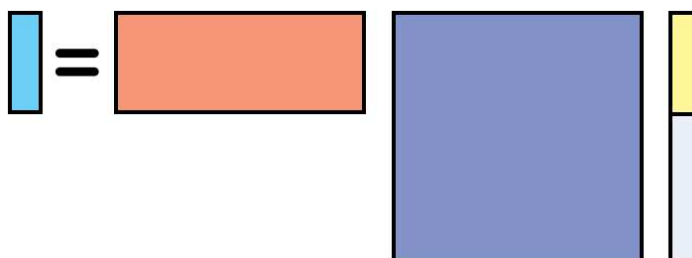
Preciznije, neka je $\Phi = [\hat{\phi}_1, \hat{\phi}_2, \dots, \hat{\phi}_m]^T \in \mathbb{R}^{m \times n}$ matrica dizajna, gdje su vektori $\hat{\phi}_i$ poslagani u redove, i $\mathbf{y} = [y_1, \dots, y_m]^t \in \mathbb{R}^m$. Problem je zadan sustavom $\mathbf{y} = \Phi \mathbf{x}$.

To što je m mnogo manji od n bitno otežava dani problem zato što tada sustav nema jedinstveno rješenje. Jedinstvenost rješenja je bitna jer je cilj egzaktno rekonstruirati početni signal \mathbf{x} .



Slika 1.1: Početni neodređen linearni sustav $\mathbf{y} = \Phi \mathbf{x}$

Ono što olakšava problem i daje mogućnost uspjeha je početna pretpostavka da \mathbf{x} ima rijedak zapis $\mathbf{s} = \Psi \mathbf{x}$.



Slika 1.2: Ekvivalentni prikaz sustava, $\mathbf{y} = \Phi \Psi^t \mathbf{s}$, gdje je \mathbf{s} rijedak

Definirajmo problem:

$$\min_{\tilde{\mathbf{x}} \in \mathbb{R}^n} \|\Psi \tilde{\mathbf{x}}\|_0 \text{ uz uvjet } \mathbf{y} = \Phi \mathbf{x}.$$

Zbog ortogonalnosti matrice Ψ i uz $\mathbf{A} := \Phi \Psi^* \in \mathbb{R}^{m \times n}$, vrijedi

$$\Phi \mathbf{x} = \Phi \Psi^* \Psi \mathbf{x} = (\Phi \Psi^*)(\Psi \mathbf{x}) = \mathbf{A} \mathbf{s}.$$

Sada možemo problem napisati u terminima vektora koeficijenata \mathbf{s} :

$$\min_{\tilde{\mathbf{s}} \in \mathbb{R}^n} \|\tilde{\mathbf{s}}\|_0 \text{ uz uvjet } \mathbf{y} = \mathbf{A} \tilde{\mathbf{s}}. \quad (P_0)$$

Problem stvarna minimizacija l_0 norme što je kombinatorno težak problem bez efikasnog algoritma rješavanja. Iz tog razloga se uvodi relaksacija gdje se l_0 norma zamijeni s l_1 normom te se dobiva lakše rješiv konveksan problem za kojeg postoje efikasni algoritmi. Kasnije će se vidjeti da ova relaksacija ima smisla i da novi problem može dati ista rješenja kao i početni.

$$\min_{\tilde{\mathbf{s}} \in \mathbb{R}^n} \|\tilde{\mathbf{s}}\|_1 \text{ uz uvjet } \mathbf{y} = \mathbf{A} \tilde{\mathbf{s}}. \quad (P_1)$$

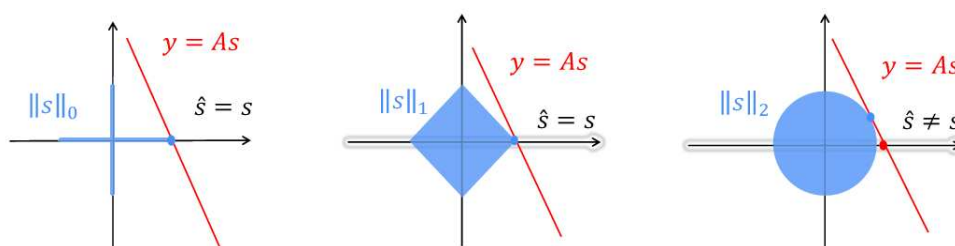
U praktičnoj primjeni metode, možemo očekivati da mjerenja sadrže grešku. Tada se problem može modificirati na ovaj način:

$$\min_{\tilde{\mathbf{s}} \in \mathbb{R}^n} \|\tilde{\mathbf{s}}\|_1 \text{ uz uvjet } \|\mathbf{A} \tilde{\mathbf{s}} - \mathbf{y}\|_2^2 \leq \varepsilon. \quad (P_{1,\varepsilon})$$

Ostaje pitanje uz koje uvjete sažimajuće očitavanje daje zadovoljavajuće rezultate: koji su povoljni izbori matrica Φ i Ψ , koliki broj očitavanja m treba imati, postoje li garancije na točnost rekonstrukcije, koliko je metoda stabilna uz prisutnost greške...

1.3 Zamjena normi

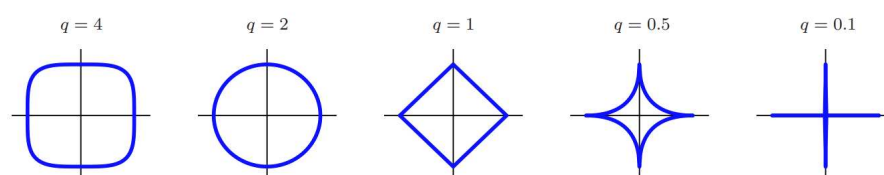
Iz (P_0) dolazimo do (P_1) tako da l_0 zamijenimo l_1 normom. Razlog tome je NP-težina pripadne minimizacije u l_0 normi. Detaljna razrada i dokaz te tvrdnje može se naći u [11], u poglavlju **2.3 NP-Hardness of 0-Minimization**. Da se uvjerimo da ta zamjena ima smisla te da ti problemi mogu dati ekvivalentna rješenja, pogledajmo geometrijski primjer u dvije dimenzije. Skup rješenja $y = As$, bit će pravac. Sa \hat{s} označimo rješenje problema minimizacije l_0 norme.



Slika 1.3: Grafički prikaz minimizacije u normama l_0 , l_1 , l_2

U slučaju minimizacije l_1 norme, rješenje će biti jednako kao ono koje bismo dobili minimizacijom l_0 norme, dok kod l_2 to nije slučaj.

Gledajući kako izgledaju jedinične sfere za ostale polunorme, mogli bismo doći do zaključka da svaka (polu)norma l_q , $0 < q \leq 1$ daje rijetka rješenja. l_1 se odabire zbog konveksnosti problema i postojanja efikasnih algoritama rješavanja.



Slika 1.4: Jedinične sfere u raznim normama

Poglavlje 2

Svojstva sažimajućeg očitavanja

U ovom poglavlju bit će uvedeni dovoljni uvjeti za matricu sustava \mathbf{A} pod kojima će se iz (P_1) moći egzaktno rekonstruirati početni signal \mathbf{x} . U praksi se ti uvjeti neće provjeravati nego će se s njima na umu konstruirati matrica \mathbf{A} , odnosno cijeli sustav mjerenja s Φ i Ψ .

Matrica \mathbf{A} može biti slučajna ili deterministička. Kad je slučajna, bit će izabrana iz klase slučajnih matrica koje s velikom vjerojatnošću ispunjavaju neke od navedenih uvjeta.

Teorijska razrada u ovom poglavlju većinom slijedi definicije i teoreme iz [13] i [10]. Dokazi svih tvrdnji prisutni su u navedenim izvorima, osim ako nije drugačije navedeno.

2.1 Egzaktna rekonstrukcija i minimalni broj mjerenja

Definicija 2.1.1 (Definicija 3.1 iz [13]). *Spark matrice \mathbf{A} , označen sa $\text{spark}(\mathbf{A})$, najmanji je broj linearno zavisnih stupaca matrice.*

Teorem 2.1.2 (Teorem 3.3 iz [13]). *Neka $\mathbf{A} \in \mathbb{R}^{m \times n}$, $k \in \mathbb{N}$. Tada je ekvivalentno:*

1. *Ako je rješenje (P_0) k -rijedak vektor, tada je to rješenje jedinstveno.*
2. $k < \text{spark}(\mathbf{A})/2$.

Definicija 2.1.3 (Definicija 3.4 iz [13]). *Neka $\mathbf{A} \in \mathbb{R}^{m \times n}$. Kažemo da \mathbf{A} zadovoljava svojstvo jezgre (eng. nullspace property, NSP) reda k , ako za sve $\mathbf{h} \in \text{Ker}(\mathbf{A}) \setminus \{0\}$ i za sve skupove indeksa $|\Lambda| \leq k$,*

$$\|1_{\Lambda} \mathbf{h}\|_1 < \frac{1}{2} \|\mathbf{h}\|_1.$$

Kako je $\|\mathbf{h}\|_1 = \|1_{\Lambda}\mathbf{h}\|_1 + \|1_{\Lambda^c}\mathbf{h}\|_1$, gornju nejednakost možemo ekvivalentno zapisati kao $\|1_{\Lambda}\mathbf{h}\|_1 < \|1_{\Lambda^c}\mathbf{h}\|_1$. Vidimo da matrica zadovoljava svojstvo jezgre ako niti jedan netrivialan element jezgre nije niti približno k -rijedak.

Sada tvrdnja slična prošlom teoremu vrijedi i za rješenje dobiveno l_1 minimizacijom:

Teorem 2.1.4 (Teorem 3.5 iz [13]). *Neka $\mathbf{A} \in \mathbb{R}^{m \times n}$, $k \in \mathbb{N}$. Tada je ekvivalentno:*

1. *Ako je rješenje (P_1) k -rijedak vektor, tada je to rješenje jedinstveno.*
2. *\mathbf{A} zadovoljava svojstvo jezgre reda k .*

Koherentnost matrice

Definicija 2.1.5 (Definicija 3.6 iz [13]). *Neka je $\mathbf{A} = (a_i)_{i=1}^n \in \mathbb{R}^{m \times n}$. **Koherentnost matrice \mathbf{A}** definirana je kao*

$$\mu(\mathbf{A}) = \max_{i \neq j} \frac{|\langle \mathbf{a}_i, \mathbf{a}_j \rangle|}{\|\mathbf{a}_i\|_2 \|\mathbf{a}_j\|_2}.$$

Lema 2.1.6 (Lema 3.7 iz [13]). *Neka $\mathbf{A} \in \mathbb{R}^{m \times n}$. Vrijedi:*

$$\mu(\mathbf{A}) \in \left[\sqrt{\frac{n-m}{m(n-1)}}, 1 \right].$$

Maksimalna vrijednost postiže se kada su stupci linearno zavisni.

Teorem 2.1.7 (Teorem 3.8 iz [13]). *Neka $\mathbf{A} \in \mathbb{R}^{m \times n}$, i neka je $\mathbf{x} \in \mathbb{R}^n \setminus \{0\}$ rješenje (P_0) koje zadovoljava:*

$$\|\mathbf{x}\|_0 < \frac{1}{2} \cdot \left(1 + \frac{1}{\mu(\mathbf{A})} \right). \quad (2.1)$$

Tada je \mathbf{x} jedinstveno rješenje (P_0) i (P_1) .

Izraz (2.1) je ekvivalentan $\mu(\mathbf{A}) < 2\|\mathbf{x}\|_0 - 1$ zbog čega je, kako bi bilo moguće rekonstruirati što više različitih signala, poželjna što manja koherentnost \mathbf{A} .

Svojstvo ograničene izometričnosti

Definicija 2.1.8 (Definicija 3.9 iz [13]). *Neka $\mathbf{A} \in \mathbb{R}^{m \times n}$. \mathbf{A} zadovoljava svojstvo ograničene izometričnosti (eng. restricted isometry property, RIP) reda k ako postoji $\delta_k \in (0, 1)$ tako da*

$$(1 - \delta_k)\|\mathbf{x}\|_2^2 \leq \|\mathbf{Ax}\|_2^2 \leq (1 + \delta_k)\|\mathbf{x}\|_2^2,$$

vrijedi za sve $\mathbf{x} \in \Sigma_k$.

Ako matrica zadovoljava svojstvo ograničene izometričnosti reda $2k$, onda je garantirano da za $\mathbf{x}, \mathbf{x}' \in \Sigma_k, \mathbf{x} \neq \mathbf{x}'$ vrijedi $\mathbf{Ax} \neq \mathbf{Ax}'$, odnosno da će se rekonstrukcijom moći dobiti točno početni signal. To se lagano dobije ako se u gornju formulu uvrsti $\mathbf{x} - \mathbf{x}'$. Primijetimo da se u narednim teoremima koristi upravo svojstvo ograničene izometričnosti reda $2k$, što dolazi iz $\mathbf{x} - \mathbf{x}' \in \Sigma_{2k}$.

Teorem 2.1.9 (Teorem 1.4 iz [10]). *Neka $\mathbf{A} \in \mathbb{R}^{m \times n}$ zadovoljava svojstvo ograničene izometričnosti reda $2k$ za $\delta_{2k} \in (0, \frac{1}{2}]$. Tada:*

$$m \geq C \cdot k \cdot \log\left(\frac{n}{k}\right),$$

za $C = \frac{1}{2 \ln(\sqrt{24}+1)} \approx 0.28$.

Teorem 2.1.10 (Teorem 1.5 iz [10]). *Neka $\mathbf{A} \in \mathbb{R}^{m \times n}$ zadovoljava svojstvo ograničene izometričnosti reda $2k$ za $\delta_{2k} < \sqrt{2} - 1$. Tada \mathbf{A} zadovoljava svojstvo jezgre reda $2k$ s konstantom $C = \frac{2}{1 - (1 + \sqrt{2})\delta_{2k}}$.*

Korolar 2.1.11. *Neka $\mathbf{A} \in \mathbb{R}^{m \times n}$ zadovoljava svojstvo ograničene izometričnosti reda $2k$ za $\delta_{2k} < \sqrt{2} - 1$. Ako je rješenje (P_1) $2k$ -rijedak vektor, tada je to rješenje jedinstveno.*

2.2 Greška rekonstrukcije

U ovom potpoglavlju vidjet ćemo da je greška rekonstrukcije razumno ograničena te da je metoda stabilna u slučaju kada je prisutna greška.

Teorem 2.2.1 (Rekonstrukcija bez greške mjerenja, Teorem 1.8 iz [10]). *Neka $\mathbf{A} \in \mathbb{R}^{m \times n}$ zadovoljava svojstvo ograničene izometričnosti reda $2k$ za $\delta_{2k} < \sqrt{2} - 1$. Neka $\mathbf{x} \in \mathbb{R}^n$ i neka je $\hat{\mathbf{x}}$ rješenje problema (P_1) . Tada:*

$$\|\mathbf{x} - \hat{\mathbf{x}}\|_2 \leq C_0 \cdot \left(\frac{\sigma_k(\mathbf{x})_1}{\sqrt{k}} \right),$$

$$\text{za } C_0 := 2 \cdot \frac{1 - (1 - \sqrt{2})\delta_{2k}}{1 - (1 + \sqrt{2})\delta_{2k}}.$$

Teorem 2.2.2 (Rekonstrukcija uz prisutnu grešku 1, Teorem 1.9 iz [10]). *Neka $\mathbf{A} \in \mathbb{R}^{m \times n}$ zadovoljava svojstvo ograničene izometričnosti reda $2k$ za $\delta_{2k} < \sqrt{2} - 1$. Neka $\mathbf{x} \in \mathbb{R}^n$ i neka je $\hat{\mathbf{x}}$ rješenje problema $(P_{1,\varepsilon})$. Tada:*

$$\|\mathbf{x} - \hat{\mathbf{x}}\|_2 \leq C_0 \cdot \left(\frac{\sigma_k(\mathbf{x})_1}{\sqrt{k}} \right) + C_1 \cdot \varepsilon,$$

$$\text{za prije definirani } C_0 \text{ i } C_1 := 4 \frac{\sqrt{1 + \delta_{2k}}}{1 - (1 + \sqrt{2})\delta_{2k}}.$$

Prisutna greška može se i drugačije modelirati. Umjesto da pretpostavimo da $\|\mathbf{e}\|_2 \leq \varepsilon$, možemo pretpostaviti da vrijedi $\|\mathbf{A}^T \mathbf{e}\|_\infty \leq \varepsilon$. Tada problem $(P_{1,\varepsilon})$ modificiramo na način:

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^n} \|\boldsymbol{\beta}\|_1 \quad \text{uz uvjet} \quad \|\tilde{\mathbf{Z}}^T (\tilde{\mathbf{Z}}\boldsymbol{\beta} - \mathbf{y})\|_\infty \leq \varepsilon, \quad (P_{\infty,\varepsilon})$$

i dobivamo sličan rezultat, iskazan u sljedećem teoremu.

Teorem 2.2.3 (Rekonstrukcija uz prisutnu grešku 2, Teorem 1.10 iz [10]). *Neka $\mathbf{A} \in \mathbb{R}^{m \times n}$ zadovoljava svojstvo ograničene izometričnosti reda $2k$ za $\delta_{2k} < \sqrt{2} - 1$. Neka $\mathbf{x} \in \mathbb{R}^n$ i neka je $\hat{\mathbf{x}}$ rješenje problema $(P_{\infty,\varepsilon})$. Tada:*

$$\|\mathbf{x} - \hat{\mathbf{x}}\|_2 \leq C_0 \cdot \left(\frac{\sigma_k(\mathbf{x})_1}{\sqrt{k}} \right) + C_2 \cdot \sqrt{k} \cdot \varepsilon,$$

$$\text{za prije definirani } C_0 \text{ i } C_2 := \frac{4\sqrt{2}}{1 - (1 + \sqrt{2})\delta_{2k}}.$$

2.3 Izbor matrica

Matrica očitavanja Φ može biti slučajna ili deterministička. Iz ranije prikazanog vidjeli smo da se uspješnost rekonstrukcije temelji na svojstvu ograničene izometričnosti i svojstvu jezgre. Eksplicitna konstrukcija matrice koja će zadovoljavati ta svojstva za proizvoljne n i m nije jednostavan problem. U praksi se čak niti ne provodi direktna provjera ta dva svojstva zbog njene računske neefikasnosti. Više o tome može se naći u [22].

Iz tih razloga, slučajne matrice biramo iz poznatih distribucija tako da postoji velika vjerojatnost da zadovoljavaju navedena svojstva. Ako Φ konstruiramo deterministički, odabiramo ju u paru s Ψ .

Ponekad je matrica očitavanja već zadana prirodom problema. To je slučaj kod magnetske rezonance i kod sažimajućeg fluorescentnog mikroskopa [21] (eng. *Compressive Fluorescence Microscopy*). Unutar takvog mikroskopa postoji $n \times n$ malih zrcala koja kombiniraju snop svjetlosti u konačni. Za svako od tih zrcala možemo odrediti hoće li, u danom mjerenju, reflektirati snop svjetlosti i time pridonijeti završnom rezultatu mjerenja.

Slučajne matrice

Slučajnu matricu biramo iz distribucije za koju je poznato da se dobro ponaša s postavljenim postupkom rekonstrukcije, točnije da zadovoljava svojstvo ograničene izometričnosti. Dodatno će matrica $\mathbf{A} = \Phi\Psi^*$ u većini slučajeva biti male koherentnosti.

Postoji nekoliko problema kada je matrica očitavanja slučajna. Prvi je da postoji mogućnost da matrica ne zadovoljava sva svojstva koja bi trebala. Kako nikada nećemo provoditi provjeru tih svojstava, može se dogoditi da uspješna rekonstrukcija neće biti moguća. Nadalje, ako signal koji očitavamo ima n elemenata, matrica očitavanja imat će $c \times n^2$ elemenata, gdje uzimamo da je $m = c \times n$, za $c > 0$. U primjenama gdje je n velik, kao što je rekonstrukcija slika i dužih signala, pohrana matrice Φ i njena primjena na vektor u trenutku rješavanja problema minimizacije su memorijski, odnosno računski zahtjevni problemi.

Iduće definicije podrazumijevaju vjerojatnosne prostore $(\mathbb{R}, \mathcal{B}(\mathbb{R}), \mathbb{P})$ i $(\mathbb{R}^{m \times n}, \mathcal{B}(\mathbb{R}^{m \times n}), \mathbb{P})$.

Definicija 2.3.1. Za $\mathbf{X} = (X_{ij})$ ćemo reći da je *normalno distribuirana matrica* ako su X_{ij} normalno distribuirane varijable.

Napomena 2.3.2. Normalnu distribuciju s očekivanjem 0 i varijancom 1, $\mathcal{N}(0, 1)$, zvat ćemo *standardnom normalnom distribucijom*.

Definicija 2.3.3. Za $\mathbf{X} = (X_{ij})$ ćemo reći da je *matrica s Bernoullijevom distribucijom* ako su X_{ij} Bernoullijeve slučajne varijable.

Koncentrirat ćemo se na slučajne matrice sa subgaussovskom distribucijom, širu klasu matrica kojoj pripadaju normalno distribuirana i slučajna matrica s Bernoullijevom distribucijom. Glavni rezultat bit će teorem koji će dati ogradu na broj mjerenja tako da matrica zadovoljava svojstvo ograničene izometričnosti s velikom vjerojatnošću.

Definicija 2.3.4. *Slučajna varijabla X ima subgaussovsku distribuciju ako postoje konstante $\beta, \kappa > 0$ takve da*

$$\mathbb{P}(|X| \geq t) \leq \beta e^{-\kappa t^2}, \quad (2.2)$$

za sve $t > 0$.

Definicija 2.3.5. *Za $\mathbf{X} = (X_{ij})$ ćemo reći da je matrica sa subgaussovskom distribucijom ako su X_{ij} slučajne varijable sa subgaussovskom distribucijom.*

Teorem 2.3.6 (Teorem 9.2 iz [11]). *Neka je $\mathbf{A} \in \mathbb{R}^{m \times n}$ realizacija slučajne matrice sa subgaussovskom distribucijom za koju vrijedi (2.2) s konstantama β, κ i neka $\varepsilon > 0, \delta > 0$. Tada postoji konstanta $C > 0$, koja ovisi samo o β i κ , tako da za δ_k uz koju matrica $\frac{1}{\sqrt{m}}\mathbf{A}$ zadovoljava svojstvo ograničene izometričnosti, vrijedi $\delta_k \leq \delta$ s vjerojatnošću barem $1 - \varepsilon$ ako:*

$$m \geq C \cdot \delta^{-2} \cdot (k \ln(e \cdot n/k) + \ln(2\varepsilon^{-1})).$$

Ako stavimo $\varepsilon = e^{-\frac{\delta^2 m}{2C}}$, dobivamo:

$$m \geq 2 \cdot C \cdot \delta^{-2} \cdot k \cdot \ln(e \cdot n/k),$$

što garantira da je $\delta_k \leq \delta$ s vjerojatnošću barem $1 - 2e^{-\frac{\delta^2 m}{2C}}$.

Za normalno distribuirane i matrice s Bernoullijevom distribucijom konstanta C je univerzalna.

Determinističke matrice

Uz determinističku konstrukciju matrice očitavanja Φ nije potrebno pohranjivati cijelu matricu, nego ju je moguće ponovno konstruirati u trenutku kada nam je potrebna. U praksi za matricu Φ odabiremo neku od poznatih matrica za koju postoje efikasni algoritmi primjene takve matrice na proizvoljan vektor. Zbog toga možemo provesti cijeli postupak rekonstrukcije, a da nikada eksplicitno ne konstruiramo matricu Φ .

Modificirajmo postavku problema. Neka je i dalje $\mathbf{x} \in \mathbb{R}^n$ originalni signal, Ψ ortonormirana baza $\mathbb{R}^{n \times n}$, $\mathbf{s} = \Psi \mathbf{x}$ k -rijedak, ali neka je $\Phi \in \mathbb{R}^{n \times n}$ ortogonalna matrica. Sada je i $\mathbf{A} = \Phi \Psi^* \in \mathbb{R}^{n \times n}$ ortogonalna matrica.

Koristit ćemo samo neke retke matrice Φ . Neka je **strategija očitavanja** $\Omega \subseteq \{1, \dots, n\}$ kardinalnosti m te neka su njeni elementi izabrani slučajno s uniformnom distribucijom. Ako imamo mjerenja oblika $\mathbf{y} = \mathbf{P}_\Omega \Phi$, gdje je $\mathbf{P}_\Omega \in \{0, 1\}^{n \times n}$ dijagonalna matrica projekcije čiji je (j, j) -ti element 1 ako je $j \in \Omega$, a 0 inače, tada se \mathbf{x} može egzaktno rekonstruirati iz \mathbf{y} s vjerojatnosti barem $1 - \varepsilon$ ako:

$$m \geq C \cdot \mu(\mathbf{A}) \cdot n \cdot k \cdot (1 + \ln(1/\varepsilon)) \cdot \ln(n), \quad (2.3)$$

gdje je $C > 0$ konstanta neovisna o ostalim elementima nejednakosti. Problem koji minimiziramo je:

$$\min_{\tilde{\mathbf{s}} \in \mathbb{R}^n} \|\tilde{\mathbf{s}}\|_1 \quad \text{uz uvjet} \quad \|\mathbf{P}_\Omega \mathbf{A} \tilde{\mathbf{s}} - \mathbf{y}\|_2^2 \leq \varepsilon.$$

Obratimo pozornost na faktor $\mu(\mathbf{A})$. Imamo:

$$\mu(\mathbf{A}) = \max_{i \neq j} |\mathbf{A}_{i,j}| = \max_{i \neq j} |\langle \phi_i, \psi_j \rangle| = \frac{1}{\sqrt{n}} \mu(\Phi, \Psi),$$

gdje se $\mu(\Phi, \Psi)$ naziva koherentnost matrica Φ i Ψ . Što je koherentnost manja, to će manje mjerenja biti potrebno za egzaktnu rekonstrukciju. U praksi se koriste parovi baza za koje je poznato da su nekoherentne.

2.4 Algoritmi sažimajućeg očitavanja

U ovom poglavlju su navedena dva osnovna algoritma rješavanja problema sažimajućeg očitavanja. U [18] i [8] može se naći više algoritama te njihove usporedbe. Oba predstavljena algoritma rješavaju problem (P_1) ili $(P_{1,\varepsilon})$.

Traženje baze

Algoritam traženja baze (*eng. basis pursuit*) rješava problem (P_1) , to jest nalazi vektor $\tilde{\mathbf{s}}$ najmanje l_1 norme uz uvjet $\mathbf{y} = \mathbf{A}\tilde{\mathbf{s}}$, tako da riješi problem linearnog programiranja koja će dati ekvivalentno rješenje. Pretvorba u problem linearnog programiranja i rješavanje te zadaće moguće je računski efikasno izvesti.

Za $\tilde{\mathbf{s}} = (\tilde{s}_i)_i^n$ vrijedi

$$\|\tilde{\mathbf{s}}\|_1 = \sum_i^n |\tilde{s}_i|.$$

Uvedemo li dodatne varijable $\tilde{s}_i^+, \tilde{s}_i^- \geq 0$ za $i = 1, \dots, n$ tako da

$$\tilde{s}_i = \tilde{s}_i^+ - \tilde{s}_i^-,$$

vrijedit će

$$|\tilde{s}_i| = \tilde{s}_i^+ + \tilde{s}_i^-,$$

odnosno

$$\|\tilde{\mathbf{s}}\|_1 = \sum_i^n (\tilde{s}_i^+ + \tilde{s}_i^-).$$

Neka $\tilde{\mathbf{s}}^+ := (\tilde{s}_i^+)_i^n$, $\tilde{\mathbf{s}}^- := (\tilde{s}_i^-)_i^n$.

Uočimo još da je uvjet $\mathbf{y} = \mathbf{A}\tilde{\mathbf{s}}$ ekvivalentan $\mathbf{A}\tilde{\mathbf{s}}^+ \geq \mathbf{y}$ i $-\mathbf{A}\tilde{\mathbf{s}}^- \geq -\mathbf{y}$.

Sada imamo zadaću:

$$\begin{cases} \mathbf{1}^T \tilde{\mathbf{s}}^+ + \mathbf{1}^T \tilde{\mathbf{s}}^- & \rightarrow \min \\ \mathbf{A}\tilde{\mathbf{s}}^+ - \mathbf{A}\tilde{\mathbf{s}}^- & \geq \mathbf{y} \\ -\mathbf{A}\tilde{\mathbf{s}}^+ + \mathbf{A}\tilde{\mathbf{s}}^- & \geq -\mathbf{y} \\ \tilde{\mathbf{s}}^+, \tilde{\mathbf{s}}^- & \geq \mathbf{0}. \end{cases}$$

U slučaju problema $(P_{1,\varepsilon})$, slično se može dobiti zadaća konusnog programiranja drugog reda koju je isto moguće računski efikasno riješiti.

Ortogonalno traženje sličnosti

Algoritam ortogonalnog traženja sličnosti (*eng. orthogonal matching pursuit, OMP*) rješava probleme (P_1) i $(P_{1,\varepsilon})$ tako da u svakoj iteraciji pronalazi stupac matrice \mathbf{A} s najvećom sličnosti s trenutnim rezidualom, gdje sličnost mjerimo pomoću skalarnog produkta. Trenutni rezidual u svakom koraku je razlika između vektora mjerenja \mathbf{y} i vektora koji se dobije kada \mathbf{A} djeluje na trenutnu aproksimaciju \mathbf{s} , osim u prvom kada rezidualom smatramo \mathbf{y} . Uočimo da se norma reziduala u svakom koraku smanjuje.

Izvršavanje zaustavljamo nakon ograničenog broja iteracija ili kada trenutna aproksimacija \mathbf{s}_i zadovoljava $\mathbf{y} \approx \mathbf{A}\mathbf{s}_i$. Nakon m koraka algoritam će pronaći \mathbf{s} takav da $\mathbf{y} = \mathbf{A}\mathbf{s}$.

Algoritam 1 Orthogonal matching pursuit

Ulaz: matrica $\mathbf{A} \in \mathbb{R}^{m \times n}$, vektor mjerenja $\mathbf{y} \in \mathbb{R}^m$

```

 $\mathbf{s}_0 = \mathbf{0}, \mathbf{r}_0 = \mathbf{y}, \Lambda_0 = \emptyset$ 
for  $i = 1; i := i + 1$  do
   $\mathbf{g}_i \leftarrow \mathbf{A}^t \mathbf{r}_{i-1}$ 
   $\Lambda_i \leftarrow \Lambda_{i-1} \cup \operatorname{argmax}_{j \leq n} |g_{ij}|$ 
   $\mathbf{s}_i|_{\Lambda_i} \leftarrow A_{\Lambda_i}^\dagger \mathbf{y}$ 
   $\mathbf{s}_i|_{\Lambda_i^c} \leftarrow 0$ 
   $\mathbf{r}_i \leftarrow \mathbf{y} - \mathbf{A}\mathbf{s}_i$ 
  if kriterij zaustavljanja then
     $\mathbf{s} = \mathbf{s}_i$ 
    stop
  end if
end for

```

Izlaz: rijedak signal $\mathbf{s} \in \mathbb{R}^n$, za koji $\mathbf{y} \approx \mathbf{A}\mathbf{s}_i$

Poglavlje 3

Primjeri

Pokazana teorija sažimajućeg očitavanja garantira mogućnost rekonstrukcija originalnog signala uz pretpostavke rijetkosti, nekoherentnosti i svojstva ograničene izometričnosti. U praksi su te pretpostavke rijetko ispunjene, no metoda se i dalje uspješno koristi.

Poglavlje prati [5], [19] i [6] gdje su uvedene dodatne definicije i teorija koji obrazlažu neočekivanu uspješnost sažimajućeg očitavanja. Dodatno je predstavljena modifikacija metode koja dovodi do veće uspješnosti.

Za primjenu metoda koristit ćemo crno-bijele slike koje ćemo reprezentirati matricama $\mathbf{A} \in [0, 1]^{m \times n}$. Svaki element matrice predstavlja bit jedan piksel slike, a njegova vrijednost bit će intenzitet bijele boje. Najmanji intenzitet bit će označen s 0 i njime će biti prikazana crna boja, dok će 1 označavati najveći intenzitet i njime će biti prikazana bijela boja.

Korištene slike uzete su iz skupa podataka COCO [16] ili su napravljene za potrebe rada. Skup podataka COCO sadrži 330 000 raznovrsnih slika, koje mogu prikazivati ljude, životinje, pojave, okoline i slično. Uz slike, sadrži oznake objekata koji se nalaze na slici te njihove granice zbog čega se primarno koristi u problemima vezanim uz označavanja i segmentaciju objekata na slici. Ovdje je korišten jer prikazuje slike iz stvarnog života što nam omogućuje da dobijemo bolju procjenu kako metoda radi u stvarnoj primjeni. Slike koje nisu crno-bijele, transformirat ćemo u crno-bijele slike.

Programi korišteni za implementaciju svih metoda u ovom poglavlju nalaze se u poglavlju 4, a mogu se naći i na [12]. Korišten je programski jezik MATLAB [17], paketi CILib [7], CurveLab [1], fastwht [2], ShearLab [15], ShearletReweighting [3], spg11 [9], subaxis [4].

3.1 Korištene transformacije

Sljedeće transformacije imaju dva svojstva koja ih čine pogodnim za korištenje u svrhu rekonstrukcije slika. Prvo je da slike pokazuju svojstvo rijetkosti ili približne rijetkosti nakon primjene tih transformacija, odnosno u pripadnim ONB. Drugo je da za primjenjivanje navedenih transformacija na vektor ili matricu postoje efikasni algoritmi pa u praksi nikad neće biti potrebno direktno množiti s matricom transformacije. Definicije su većinom uzete iz [20].

Diskretna Fourierova transformacija

Fourierova transformacija dani signal u vremenskoj domeni pretvara u signal u frekvenzijskoj domeni. Drugim riječima, za dani signal i frekvenciju, možemo odrediti koliko je sinusoida te frekvencije pridonijela ukupnom signalu. Diskretna Fourierova transformacija (DFT) signala $x = (x_k) \in \mathbb{R}^n$ definirana je kao:

$$X_{j+1} = \sum_{k=0}^{n-1} x_{k+1} \cdot e^{\frac{-2\pi i k j}{n}} = \sum_{k=0}^{n-1} x_{k+1} \cdot \omega^{kj}, \quad \text{gdje je } \omega = e^{-2\pi i/n}, \quad \text{za } j = 0, \dots, n.$$

Normalizirana matrica transformacije izgleda ovako:

$$U_{\text{dft}} = \left(\frac{\omega^{(k-1)(j-1)}}{\sqrt{n}} \right)_{k,j=1,\dots,n}.$$

Diskretna kosinusna transformacija

Diskretna kosinusna transformacija (*eng. discrete cosine transform*, DCT) će signal izraziti kao sumu kosinusa s različitim frekvencijama. Postoji više vrsta DCT-a, od kojih ćemo mi koristiti sljedeću:

$$X_{j+1} = \sum_{k=0}^{n-1} x_{k+1} \cdot \cos \left[\frac{\pi}{n} \left(k + \frac{1}{2} \right) \cdot j \right], \quad \text{za } j = 0, \dots, n.$$

Postoje dva načina kako bismo mogli primijeniti spomenute transformacije na sliku. Neka $\mathbf{X} \in \mathbb{R}^{n \times n}$ predstavlja sliku. Prvi način je da sliku vektoriziramo te na nju primijenimo jednodimenzionalnu transformaciju. Drugi je da na sliku primijenimo dvodimenzionalnu verziju transformacije, danu s \mathbf{UXU}^* , gdje je \mathbf{U} matrica jednodimenzionalne transformacije.

Diskretna valićna transformacija

Neprekidna valićna transformacija (*eng. continuous wavelet transform*) signala $x(t)$ definirana je kao:

$$[W_{\psi}x](\tau, a) = |a|^{-1/2} \int_{-\infty}^{\infty} x(t) \cdot \psi^* \left(\frac{t-\tau}{a} \right) dt,$$

gdje je $\tau \in \mathbb{R}$ vremenski pomak, $a \neq 0$ faktor skaliranja, $\psi \in L^2(\mathbb{R})$ prototipni valić (*eng. mother wavelet*), a ψ^* kompleksna konjugacija od ψ .

Skup $\{\psi_{\tau,a}(t) = |a|^{-1/2} \cdot \psi \left(\frac{t-\tau}{a} \right) : \tau, a \in \mathbb{R}, a \neq 0\}$ zovemo skup funkcija razlaganja. Na valićnu transformaciju $[W_{\psi}x](\tau, a)$ možemo gledati kao na mjeru doprinosa valića $\psi_{\tau,a}$ signalu $x(t)$, gdje valić $\psi_{\tau,a}$ dobijemo tako da prototipni valić $\psi(t)$ translateramo za τ , te skaliramo za faktor a .

Diskretnu valićnu transformaciju (*eng. discrete wavelet transform, DWT*) dobit ćemo ako izraz evaluiramo u diskretnim vrijednostima $a = a_0^m$ i $\tau = n\tau_0 a_0^{-m}$, za $m, n \in \mathbb{Z}$ i fiksne $\tau_0, a_0 > 0$. Imamo:

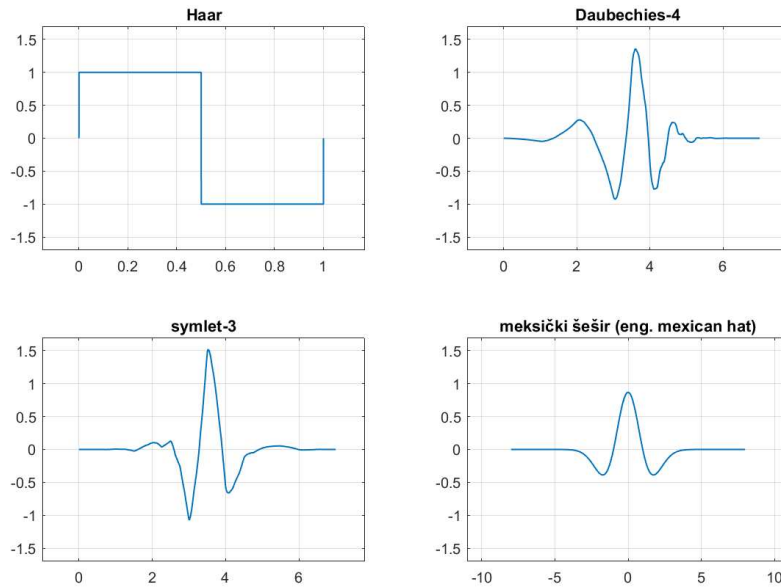
$$[W_{\psi}x](m, n) = a_0^{-m/2} \int_{-\infty}^{\infty} x(t) \cdot \psi^* \left(\frac{t}{a_0^m} - n\tau_0 \right) dt.$$

Za $m \in \mathbb{Z}$ fiksna, skupove oblika $\{\psi_{m,n}(t) = a_0^{-m/2} \cdot \psi \left(\frac{t}{a_0^m} - n\tau_0 \right) : n \in \mathbb{Z}\}$ zvat ćemo **razinama razlučivosti**. Za veći m dobit ćemo grublje, a za manji finije razine razlučivosti.

Na slici 3.1 prikazani su neki od mogućih izbora prototipnih valića. Različiti skupovi funkcija razlaganja mogu imati svojstvo ortogonalnosti, neprekidnosti i sl. U nastavku rada kao prototipne valiće koristit ćemo Haarov i Daubechies valić.

Dvodimenzionalna diskretna valićna transformacija dobiva se tako da uzastopno pri-mjenjujemo jednodimenzionalne diskretne valićne transformacije na retke pa na stupce.

Sustav sličan valićima čini shearlet sustav, gdje je još dodan koeficijent skaliranja u smjeru i smicanja prototipnog oblika. Više o shearlet sustavu može se naći u [14].

Slika 3.1: Različiti prototipni valići ψ

Hadamardova matrica

Hadamardove matrice koristit ćemo kao determinističke matrice očitavanja. One su ortogonalne i simetrične matrice čiji su elementi 1 ili -1 . Zbog toga će ih biti jednostavnije implementirati u sustavu očitavanja, slično kao što je bio slučaj kod sažimajućeg fluorescentnog mikroskopa. Isto tako, uz Hadamardove matrice veže se Walsh-Hadamardova transformacija za čiju primjenu postoji efikasan algoritam.

Definicija 3.1.1. *Dijadsko proširenje od $x \in [0, 1)$ je red:*

$$\sum_{i=1}^{\infty} x_i 2^{-i}, \quad \text{za } (x_i)_{i \in \mathbb{N}} \in \{0, 1\}^{\mathbb{N}}.$$

Dijadsko proširenje od $n \in \mathbb{N}_0$ je red:

$$\sum_{i=1}^{\infty} n_i 2^{i-1}, \quad \text{za } (n_i)_{i \in \mathbb{N}} \in \{0, 1\}^{\mathbb{N}}.$$

Definicija 3.1.2. *Redosljedom određene Walshove funkcije (sequency-ordered Walsh functions) definirane su kao:*

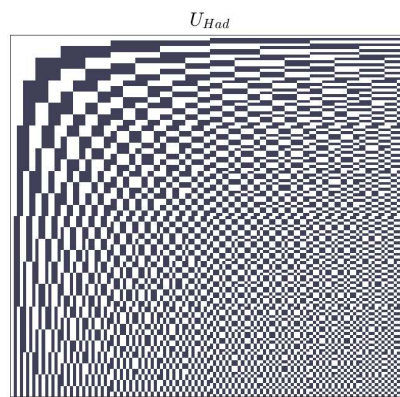
$$v_n(x) = -1^{\sum_{i=1}^{\infty} (n_i + n_{i+1}) x_i}, \quad x \in [0, 1), \quad n \in \mathbb{N}_0,$$

gdje su $(x_i)_{i \in \mathbb{N}}$ i $(n_i)_{i \in \mathbb{N}}$ dijadska proširenja od x , odnosno n .

Hadamardove matrice su simetrične i ortogonalne (do na konstantu) matrice dimenzije $2^k \times 2^k$ čiji su elementi 1 i -1 . Definirane su kao:

$$U_{Had} = \left(v_{i-1} \left(\frac{j-1}{2^k} \right) \right)_{i,j=1,\dots,2^k},$$

gdje su v_{i-1} redosljedom određene Walshove funkcije. Na slici 3.2 prikazana je Hadamardova matrica dimenzija 128×128 .



Slika 3.2: Hadamardova matrica

3.2 Proširenje teorije

Do sada pokazana teorija temelji se na pojmovima rijetkosti, koherentnosti i svojstva ograničene izometričnosti. Vidjet ćemo da često korišteni signali i matrice sustava imaju dodatnu strukturu za koju je trenutna teorija nedostatna. Isto tako, u slučajevima kada nisu ispunjeni neki od postavljenih uvjeta, vrlo dobra rekonstrukcija je moguća. Uvodimo dodatne pojmove koji uzimaju u obzir strukturu koja se javlja, uz koje ćemo moći dobiti drugačije uvjete pod kojima očekujemo uspješnu rekonstrukciju. Motivirani time, modificirat ćemo strategiju očitavanja što će dovesti do veće uspješnosti rekonstrukcije.

Lokalna rijetkost kod valić baza, ali i šire

U 3.1 navedena su dva bitna svojstva korištenih transformacija. Uz njih, dodatno još vrijedi da će koeficijenti koji pripadaju manjim frekvencijama (tj. grubljim razinama razlučivosti kod valić baza) biti većih apsolutnih vrijednosti, dok će oni koji pripadaju većim frekvencijama imati manje apsolutne vrijednosti. Isto tako, apsolutne vrijednosti koeficijenata brzo će padati kako frekvencije rastu kod Fourierove i kosinusne transformacije, odnosno kako razine razlučivosti idu od grubljih prema finijima kod valićnih transformacija. Uzmimo sada sliku iz stvarnog života te pokažimo da ilustriramo ovu tvrdnju primjerom. Korištena slika uzeta je iz COCO skupa podataka.

Na slici 3.3 prikazane su vrijednosti koeficijenata koji se dobiju nakon primjene diskretne kosinusne, odnosno diskretne valićne transformacije s Haarovim valićem. Ono što možemo vidjeti je brzi pad apsolutnih vrijednosti koeficijenata, što ukazuje na približnu rijetkost slika u bazama odgovarajućih transformacija.

Uzmimo valić bazu $\{\phi_n\}_{n \in \mathbb{N}}$. Poznato je da postoji dekompozicija \mathbb{N} u konačne podskupove koji odgovaraju različitim razinama razlučivosti baze valića, $\mathbb{N} = \bigcup_{k \in \mathbb{N}} \mathcal{M}_k$, gdje je $\mathcal{M}_k = \{M_{k-1} + 1, \dots, M_k\}$ skup indeksa koji odgovara k -toj razini, uz $0 = M_0 < M_1 < M_2 < \dots$. Neka su $s \in l^2(\mathbb{N})$ koeficijenti signala x u ovoj bazi. Neka je $\varepsilon \in \langle 0, 1 \rangle$, definiramo **globalnu rijetkost** i **rijetkost na k -toj razini** kao:

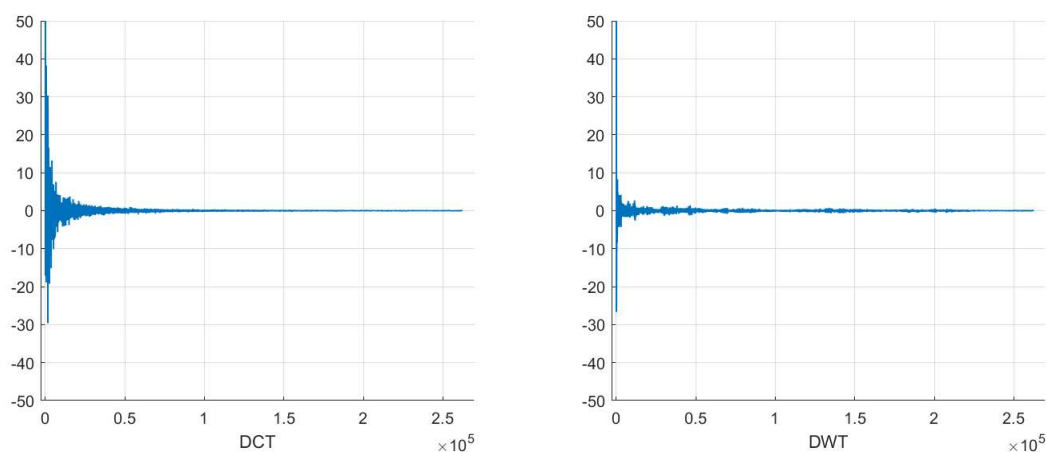
$$s = s(\varepsilon) = \min \left\{ n : \left\| \sum_{i \in \mathcal{M}_n} s_i \phi_i \right\| \geq \varepsilon \left\| \sum_{i=1}^{\infty} s_i \phi_i \right\| \right\}$$

$$s_k = s_k(\varepsilon) = \left| \mathcal{M}_{s(\varepsilon)} \cap \{M_{k-1} + 1, \dots, M_k\} \right|,$$

gdje je \mathcal{M}_n skup n indeksa koeficijenata najvećih po apsolutnoj vrijednosti. Prikazano svojstvo govori da:

$$s_k / (M_k - M_{k-1}) \rightarrow 0, \quad (3.1)$$

kada $k \rightarrow \infty$ za proizvoljan $\varepsilon \in \langle 0, 1 \rangle$.



Slika 3.3: Početna slika i pripadni koeficijenti nakon kosinusne (lijevo) i valične (desno) transformacije

Svojstvo analogno 3.1 prisutno je i kod drugih često korištenih baza, ne samo kod valića. Zbog toga nam u nastavku neće od važnosti biti globalna rijetkost signala, nego rijetkost po razinama.

Definicija 3.2.1. Neka je $\mathbf{x} \in \mathbb{R}^n$. Za $r \in \mathbb{N}$ neka je $\mathbf{M} = (M_1, \dots, M_r) \in \mathbb{N}^r$ i $\mathbf{s} = (s_1, \dots, s_r) \in \mathbb{N}^r$ i neka je $s_k \leq M_k - M_{k-1}$ za $k = 1, \dots, r$, $M_0 = 0$. \mathbf{x} je (\mathbf{s}, \mathbf{M}) -rijedak ako za svaki $k = 1, \dots, r$:

$$\text{card}\left(\text{supp}(\mathbf{x}) \cap \{M_{k-1} + 1, \dots, M_k\}\right) \leq s_k.$$

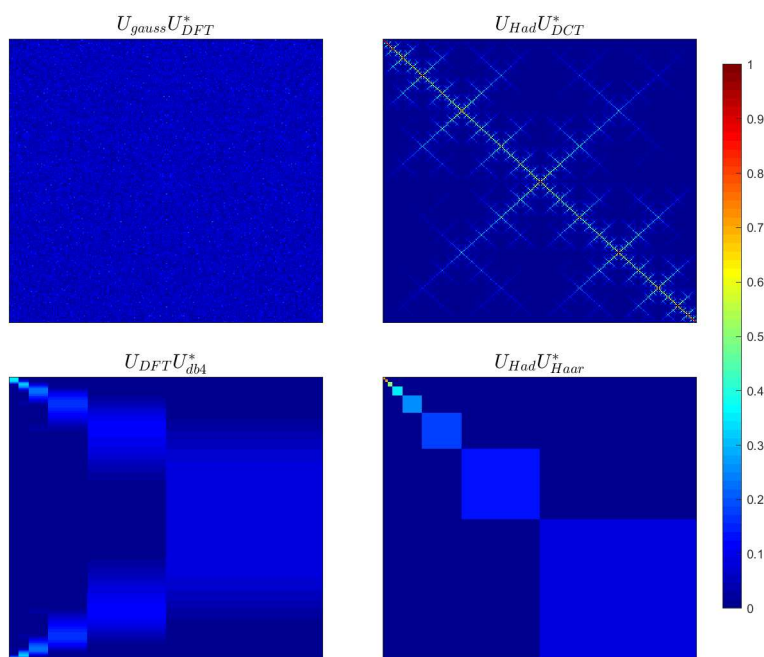
Skup svih (\mathbf{s}, \mathbf{M}) -rijetkih vektora označavamo s $\Sigma_{\mathbf{s}, \mathbf{M}}$.

Asimptotska nekoherentnost

U 2.1.7 i 2.3 vidjeli smo da je nekoherentnost matrice $\mathbf{A} = \mathbf{\Phi}\mathbf{\Psi}^t$ poželjno svojstvo. Za razliku od slučajnih matrica, determinističke matrice očitavanja koje se koriste u praksi često su koherentne s matricom transformacije. Pogledajmo kako izgleda matrica \mathbf{A} za neke izbore matrica $\mathbf{\Phi}$ i $\mathbf{\Psi}$.

Korištene matrice očitavanja bit će matrica s normalnom distribucijom, Hadamardova matrica i matrica diskretne Fourierove transformacije, dok će korištene matrice transformacije biti matrica diskretne Fourierove, kosinusne ili valične transformacije s Haarovim i Daubechies-4 prototipnim valićem. Dodatno ćemo matricu s normalnom distribucijom skalirati tako da joj stupci budu l_2 norme 1, a označit ćemo ju s $\mathbf{U}_{\text{gauss}}$. To radimo zbog jasnoće prikaza.

Na slici 3.4 prikazane su apsolutne vrijednosti matrica: $\mathbf{U}_{\text{gauss}}\mathbf{U}_{\text{DFT}}^*$ (gore lijevo), $\mathbf{U}_{\text{Had}}\mathbf{U}_{\text{DCT}}^*$ (gore desno), $\mathbf{U}_{\text{DFT}}\mathbf{U}_{\text{db4}}^*$ (dolje lijevo), $\mathbf{U}_{\text{Had}}\mathbf{U}_{\text{Haar}}^*$ (dolje desno).



Slika 3.4: Apsolutne vrijednosti elemenata matrice

U slučaju kada je $\mathbf{\Phi}$ deterministička matrica, možemo vidjeti da matrica \mathbf{A} ima veliku koherentnost, ali da elementi veće apsolutne vrijednosti čine neku strukturu. Izbacimo li

određene retke ili stupce, koherentnost dobivene matrice je mala. To je motivacija sljedeće definicije. U slučaju kada je Φ slučajna matrica, odnosno kada je $\Phi = \mathbf{U}_{\text{gauss}}$, takva struktura nije vidljiva.

Definicija 3.2.2. Neka je $\{U_N\}_{N \in \mathbb{N}}$ niz izometrija, $U_N \in \mathbb{C}^{N \times N}$ za svaki $N \in \mathbb{N}$. Neka je P_K projekcija na $\left\{ \left\{ e_j : j = 1, \dots, K \right\} \right\}$, gdje je $\{e_j\}$ kanonska baza od \mathbb{C}^N , a P_K^\perp njegov ortogonalni komplement. Kažemo da je $\{U_N\}_{N \in \mathbb{N}}$ **asimptotski nekoherentan** ako $\mu(P_K^\perp U_N)$, $\mu(U_N P_K^\perp) \rightarrow 0$ kada $K \rightarrow \infty$ i $N/K = c$, $\forall c \geq 1$.

Definicija je dana za nizove matrica gdje broj redaka i stupaca tih matrica idu u $+\infty$. U našem slučaju gledamo nizove oblika $\{\Phi_n \Psi_n^*\}$, gdje Φ_n i Ψ_n predstavljaju matrice transformacija iz $\mathbb{R}^{n \times n}$ ili $\mathbb{C}^{n \times n}$ iz 3.1. Ako je $\{\Phi_n \Psi_n^*\}$ asimptotski nekoherentan, reći ćemo da su Φ_n i Ψ_n asimptotski nekoherentne.

Niz matrica bit će asimptotski nekoherentan, ako koherentnost matrica koji dobijemo tako da maknemo mali broj početnih redaka i mali broj početnih stupaca, konvergira u 0. Na slici 3.4 možemo vidjeti da bi \mathbf{U}_{Had} i \mathbf{U}^{Haar} ; \mathbf{U}_{DFT} i \mathbf{U}^{db4} mogle biti asimptotski nekoherentne, dok za \mathbf{U}_{Had} i \mathbf{U}_{DCT} to nije slučaj. Naime, elementi velike apsolutne vrijednosti koncentrirani su oko dijagonale te na njenu koherentnost zanemarivanje početnih redaka i stupaca nema veliki utjecaj.

Ova definicija zahtijeva promjenu sustava očitavanja. Naime, velika koherentnost u prvih nekoliko redaka matrice \mathbf{A} ukazuje da bi bitna informacija u signalu mogla biti sadržana u mjerenjima koja odgovaraju tim redcima. Zbog toga bismo htjeli povećati broj mjerenja koja odgovaraju redcima velike koherentnosti i postupno smanjivati ona koja odgovaraju redcima manje koherentnosti.

Definicija 3.2.3. Neka je $r \in \mathbb{N}$, $\mathbf{N} = (N_1, \dots, N_r) \in \mathbb{N}^r$, gdje je $1 \leq N_1 < \dots < N_r$, $\mathbf{m} = (m_1, \dots, m_r) \in \mathbb{N}^r$, gdje je $m_k \leq N_k - N_{k-1}$, $k = 1, \dots, r$. Pretpostavimo da su

$$\Omega_k \subseteq \{N_{k-1} + 1, \dots, N_k\}, \quad |\Omega_k| = m_k, \quad k = 1, \dots, r,$$

uniformno odabrani, gdje je $N_0 = 0$. Skup

$$\Omega = \Omega_{\mathbf{N}, \mathbf{m}} := \bigcup_{k=1}^r \Omega_k$$

zovemo (\mathbf{N}, \mathbf{m}) -strategija očitavanja na više razina (koristeći r razina).

Neka je $\mathbf{z} := \Phi \mathbf{x} \in \mathbb{R}^n$ i matrica očitavanja neka je ortogonalna. Razdvojimo \mathbf{z} na r razina očitavanja, gdje indekse granica svake razine označimo s $N_{k-1} + 1$ i N_k . Imamo da k -ta

razina obuhvaća elemente $\{\mathbf{z}_{N_{k-1}+1}, \dots, \mathbf{z}_{N_k}\}$. Od tih $N_k - N_{k-1}$ elemenata, očitat ćemo njih m_k . Ω_k označava koje smo elemente k -te razine očitali.

Definicija 3.2.4. Neka je \mathbf{U} izometrija na $\mathbb{R}^{n \times n}$ ili $\ell^2(\mathbb{N})$. Neka je $\mathbf{N} = (N_1, \dots, N_r) \in \mathbb{N}^r$ i $\mathbf{M} = (M_1, \dots, M_r) \in \mathbb{N}^r$, gdje je $1 \leq N_1 < \dots < N_r$ i $1 \leq M_1 < \dots < M_r$.

(k, l)-ta lokalna koherentnost od U s obzirom na N i M definirana je kao

$$\mu_{\mathbf{N}, \mathbf{M}}(k, l) = \sqrt{\mu(\mathbf{P}_{N_{k-1}}^{N_k} \mathbf{U} \mathbf{P}_{M_{l-1}}^{M_l}) \cdot \mu(\mathbf{P}_{N_{k-1}}^{N_k} \mathbf{U})},$$

za $k, l = 1, \dots, r$ i \mathbf{P}_b^a matrica projekcije koja odgovara indeksima $\{a + 1, \dots, b\}$.

k-ta relativna rijetkost definirana je kao:

$$S_k = S_k(\mathbf{N}, \mathbf{M}, \mathbf{s}) = \max_{\mathbf{z} \in \Theta} \|\mathbf{P}_{N_k}^{N_{k-1}} \mathbf{U} \mathbf{z}\|^2,$$

gdje je $\Theta = \{\mathbf{z} \in \mathbb{C}^n : \|\mathbf{z}\|_\infty \leq 1, \|\mathbf{P}_{M_l}^{M_{l-1}} \mathbf{z}\|_0 = s_l, l = 1, \dots, r\}$.

Glavni rezultat

Dolazimo do glavnog rezultata proširene teorije, iskazanog u [6], koji će objediniti no-vodefinirane pojmove te dati realističnije granice za uspješnost rekonstrukcije.

Rezultat garantira da ćemo moći rekonstruirati (\mathbf{s}, \mathbf{M}) -rijedak vektor $\mathbf{x} \in \mathbb{R}^n$ iz mjerenja dobivenih očitavanjem na više razina, ako broj mjerenja svake razine m_k zadovoljava:

$$1 \geq C_1 \cdot \frac{N_k - N_{k-1}}{m_k} \cdot \left(\sum_{l=1}^r \mu_{\mathbf{N}, \mathbf{M}}(k, l) \cdot s_l \right) \cdot \ln(n),$$

gdje $m_k \geq C_2 \cdot \hat{m}_k \cdot \ln(n)$, a za svaki \hat{m}_k vrijedi:

$$1 \geq C_3 \cdot \sum_{k=1}^r \left(\frac{N_k - N_{k-1}}{\hat{m}_k} - 1 \right) \cdot \mu_{\mathbf{N}, \mathbf{M}}(k, l) \cdot \tilde{s}_k,$$

$\forall l = 1, \dots, r$ i $\forall \hat{s}_1, \dots, \hat{s}_r \in (0, \infty)$ za koje $\hat{s}_1 + \dots + \hat{s}_r \leq s_1 + \dots + s_r$ i $\hat{s}_k \leq S_k(\mathbf{N}, \mathbf{M}, \mathbf{s})$. C_1, C_2, C_3 su poznate konstante.

3.3 Usporedba klasične i poboljšane metode

U ovom dijelu usporedit ćemo klasično i poboljšano sažimajuće očitavanje na stvarnim slikama. Prvo ćemo pokazati kakve rezultate možemo očekivati od standardne rekonstrukcije, a zatim ćemo uvesti spomenuta poboljšanja i vidjeti kako to mijenja uspješnost rekonstrukcije. Usporedit ćemo i kako različite matrice transformacije i strategije očitavanja utječu na rekonstrukciju.

Gaussova jezgra i klizajući prozor

Za $x, y \in \mathbb{R}$ i $\sigma \in \mathbb{R}$ definiramo **Gaussovu jezgru** kao $\kappa(x, y) = \exp\left(-\frac{\|x-y\|^2}{2\sigma^2}\right)$.

Neka $\mathbf{X} = (x_{ij}) \in \mathbb{R}^{n \times n}$ predstavlja sliku. Za $m \in \mathbb{N}$ neparan definiramo **prozor veličine $m \times m$ s Gaussovom jezgrom od \mathbf{X} u w, z , za $w, z \in \{1, \dots, n\}$** kao:

$$\mathbf{x}_{w,z} = \left\{ x_{i,j} \kappa(x_{w,z}, x_{i,j}) : i = w - \frac{m+1}{2}, \dots, w + \frac{m+1}{2}, j = z - \frac{m+1}{2}, \dots, z + \frac{m+1}{2} \right\}$$

gdje je κ Gaussova jezgra. Primijetimo da se u prošloj definiciji može dogoditi slučaj u kojem je $(i, j) \notin [1, n]^2$. U tom slučaju je definiranje vrijednosti $x_{i,j}$ stvar dogovora, u kojem je jedna od opcija gledati kao da je slika zrcaljena preko svojih rubova.

Neka je $m \in \mathbb{N}$ neparan, $\mathbf{X} = (x_{ij}), \mathbf{Y} = (y_{ij}) \in \mathbb{R}^{n \times n}$ predstavljaju slike, a κ Gaussova jezgra. Definiramo:

- **sredinu prozora veličine $m \times m$ s Gaussovom jezgrom od \mathbf{X} u w, z , za $w, z \in \{1, \dots, n\}$** kao:

$$\mu_{\mathbf{x}} = \sum_{i=w-(m+1)/2}^{w+(m+1)/2} \sum_{j=z-(m+1)/2}^{z+(m+1)/2} x_{i,j} \cdot \kappa(x_{w,z}, x_{i,j}),$$

- **uzoračku varijancu prozora veličine $m \times m$ s Gaussovom jezgrom od \mathbf{X} u w, z , za $w, z \in \{1, \dots, n\}$** kao:

$$\sigma_{\mathbf{x}}^2 = \left(\sum_{i=w-(m+1)/2}^{w+(m+1)/2} \sum_{j=z-(m+1)/2}^{z+(m+1)/2} x_{i,j}^2 \cdot \kappa(x_{w,z}, x_{i,j}) \right) - \mu_{\mathbf{x}}^2,$$

- **uzoračku kovarijancu prozora veličine $m \times m$ s Gaussovom jezgrom od \mathbf{X} i \mathbf{Y} u w, z , za $w, z \in \{1, \dots, n\}$** kao:

$$\sigma_{\mathbf{xy}}^2 = \left(\sum_{i=w-(m+1)/2}^{w+(m+1)/2} \sum_{j=z-(m+1)/2}^{z+(m+1)/2} x_{i,j} \cdot y_{i,j} \cdot \kappa(x_{w,z}, x_{i,j}) \right) - \mu_{\mathbf{x}} \cdot \mu_{\mathbf{y}},$$

SSIM

Uspješnost rekonstrukcije slika mjerit ćemo **mjerom indeksa strukturne sličnosti** [24] (*eng. structural similarity index measure, SSIM*). Razlog tome je što SSIM bolje prati vizualno percipiranu razliku između slika od drugih sličnih mjera. Sličnost slika bit će iskazana u vrijednostima od 1 za iste, do -1 za potpuno neslične slike.

Za prozore veličine 11×11 s Gaussovom jezgrom od \mathbf{X} i \mathbf{Y} , $\mathbf{x} = \mathbf{x}_{w,z}$ i $\mathbf{y} = \mathbf{y}_{w,z}$, lokalna sličnost definirana je kao:

$$\text{SSIM}(\mathbf{x}, \mathbf{y}) = \frac{(2\mu_x\mu_y + 0.01^2)(2\sigma_{xy} + 0.03^2)}{(\mu_x^2 + \mu_y^2 + 0.01^2)(\sigma_x^2 + \sigma_y^2 + 0.03^2)},$$

gdje je μ_x uzoračka sredina od \mathbf{x} , μ_y uzoračka sredina od \mathbf{y} , σ_x^2 uzoračka varijanca od \mathbf{x} , σ_y^2 uzoračka varijanca od \mathbf{y} , σ_{xy} uzoračka kovarijanca od \mathbf{x} i \mathbf{y} . Ukupnu sličnost dobit ćemo kao aritmetičku sredinu svih lokalnih sličnosti.

Klasično sažimajuće očitavanje

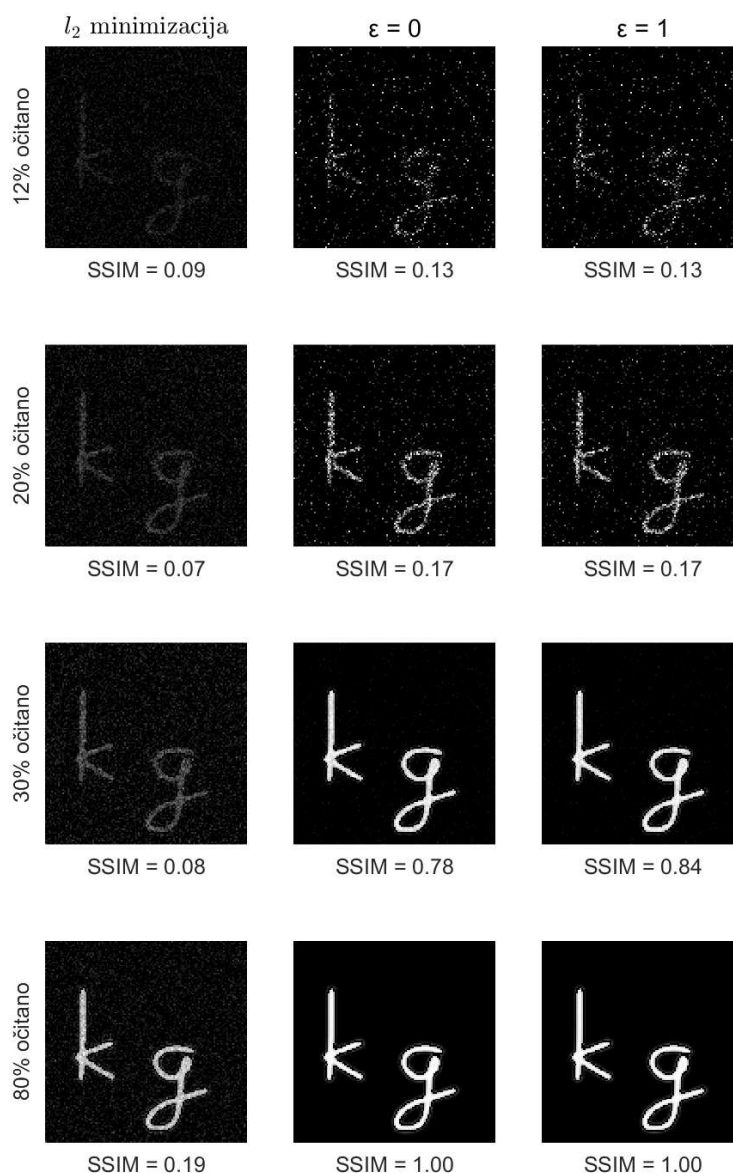
Koristimo sliku dimenzija 128×128 , prikazanu na 3.5. Možemo vidjeti da je rijetka u kanonskoj bazi, a sadrži otprilike 11% ne-nul koeficijenata. Zbog toga koristimo $\Psi = \mathbf{I}$. Sliku vektoriziramo, a kao matricu očitavanja $\Phi \in \mathbb{R}^{m \times 128^2}$ koristimo matricu sa standardnom normalnom distribucijom.

Prvo rješavamo problem minimizacije l_2 norme, a onda (P_1) , odnosno $(P_{1,\varepsilon})$ s $\varepsilon = 1$. U eksperimentima mijenjamo postotak očitanih koeficijenata.



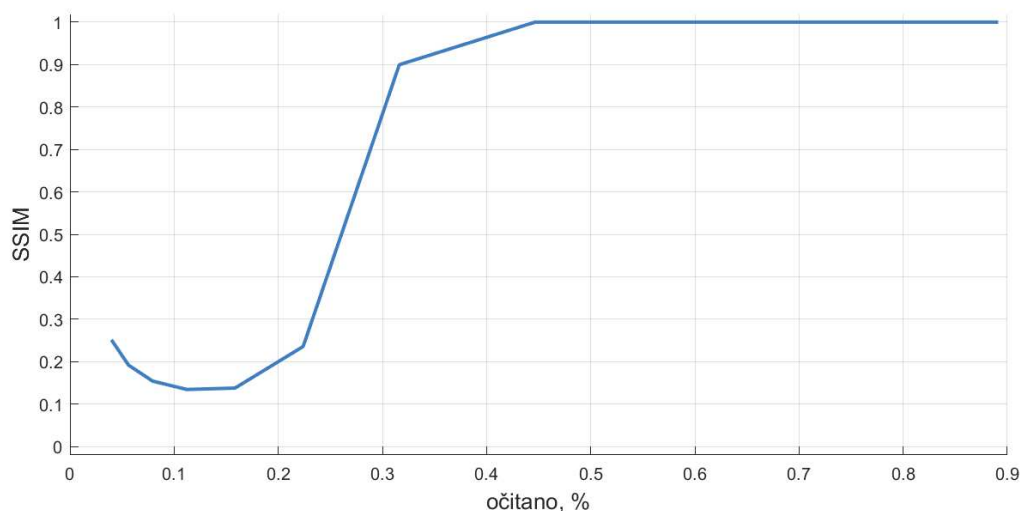
Slika 3.5: Originalna slika

Na slici 3.7 prikazani su rezultati rekonstrukcije, a ispod svake rekonstrukcije prikazana je uspješnost izražena preko SSIM. Možemo vidjeti da uz relativno mali broj očitavanja, l_1 minimizacija uspješno rekonstruira početnu sliku, dok niti s velikim brojem očitavanja to nije slučaj za problem l_2 minimizacije. Problemi (P_1) i $(P_{1,\varepsilon})$ daju približno jednake rezultate uz $\varepsilon = 1$.



Slika 3.6: Usporedba rezultata klasičnog sažimajućeg očitavanja

Promotrimo kako broj uzoraka utječe na sličnost originalne slike i njene rekonstrukcije. Rekonstruirajmo istu sliku kao u prošlom primjeru, uzmimo $\varepsilon = 1$ i postupak ponovimo za $m = cn$, za razne $c \in \langle 0, 1 \rangle$. Rezultati su prikazani na slici 3.7. Na apscisi je označen postotak očitanih uzoraka u odnosu na sve elemente slike, a na ordinati uspješnost rekonstrukcije, izraženu preko SSIM.



Slika 3.7: Odnos broja uzoraka i uspješnosti rekonstrukcije

Odmah su vidljiva dva svojstva grafa. Prvo je pad sličnosti kako povećavamo c od 0 do 0.15. Razlog tome je što će pri malom broju očitanih uzoraka, postupak rekonstrukcije vraćati većinom crnu sliku. Kako broj uzoraka raste, nastaju razne greške svuda po slici. Takve greške dobro su vidljive na 3.7, u slučaju kada je očitano 12% ukupne slike. Kako je potpuno crna slika sličnija originalnoj nego rekonstrukcija s takvim greškama, uočavamo pad sličnosti.

Bitnije svojstvo je nelinearan, skoro logaritamski, odnos broja uzetih uzoraka i sličnosti rekonstrukcije s originalnom slikom. Kada je broj očitanih uzoraka relativno mali, i mali broj novih uzoraka doprinijet će povećanju kvalitete rekonstrukcije. Kako se broj očitanih uzoraka povećava, doprinos novih uzoraka sve je manji.

Uzmimo sada sliku koja nije rijetka u kanonskoj bazi te pogledajmo kako različite matrice transformacija Ψ utječu na rezultate rekonstrukcije. Na slici 3.8 prikazana je korištena slika, a nastaviti ćemo ju koristiti do kraja ovog poglavlja. Slika je veličine 512×512 i preuzeta je iz COCO skupa podataka. Za Φ uzmimo matricu čiji su elementi Bernoullijeve slučajne varijable.

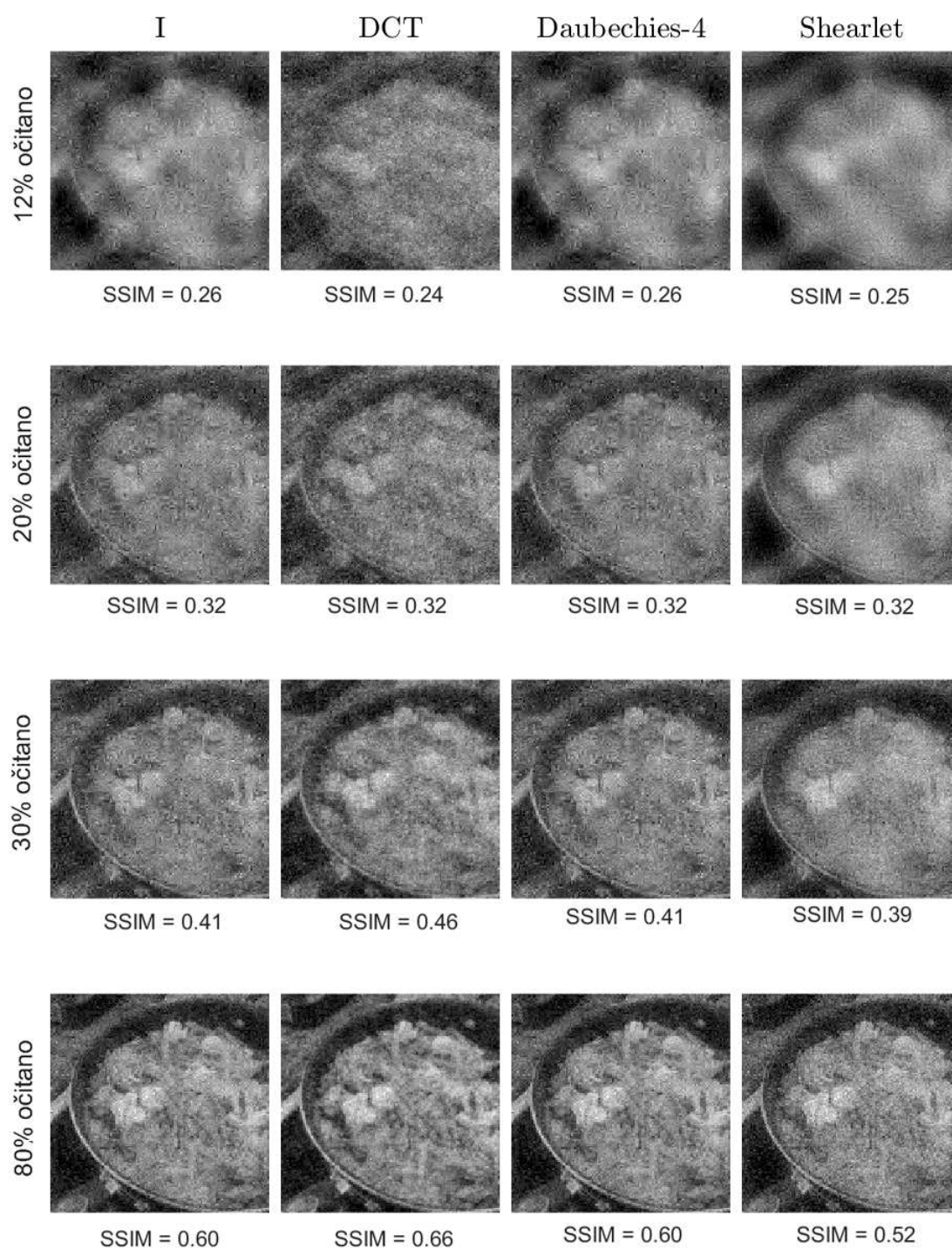


Slika 3.8: Korištena slika

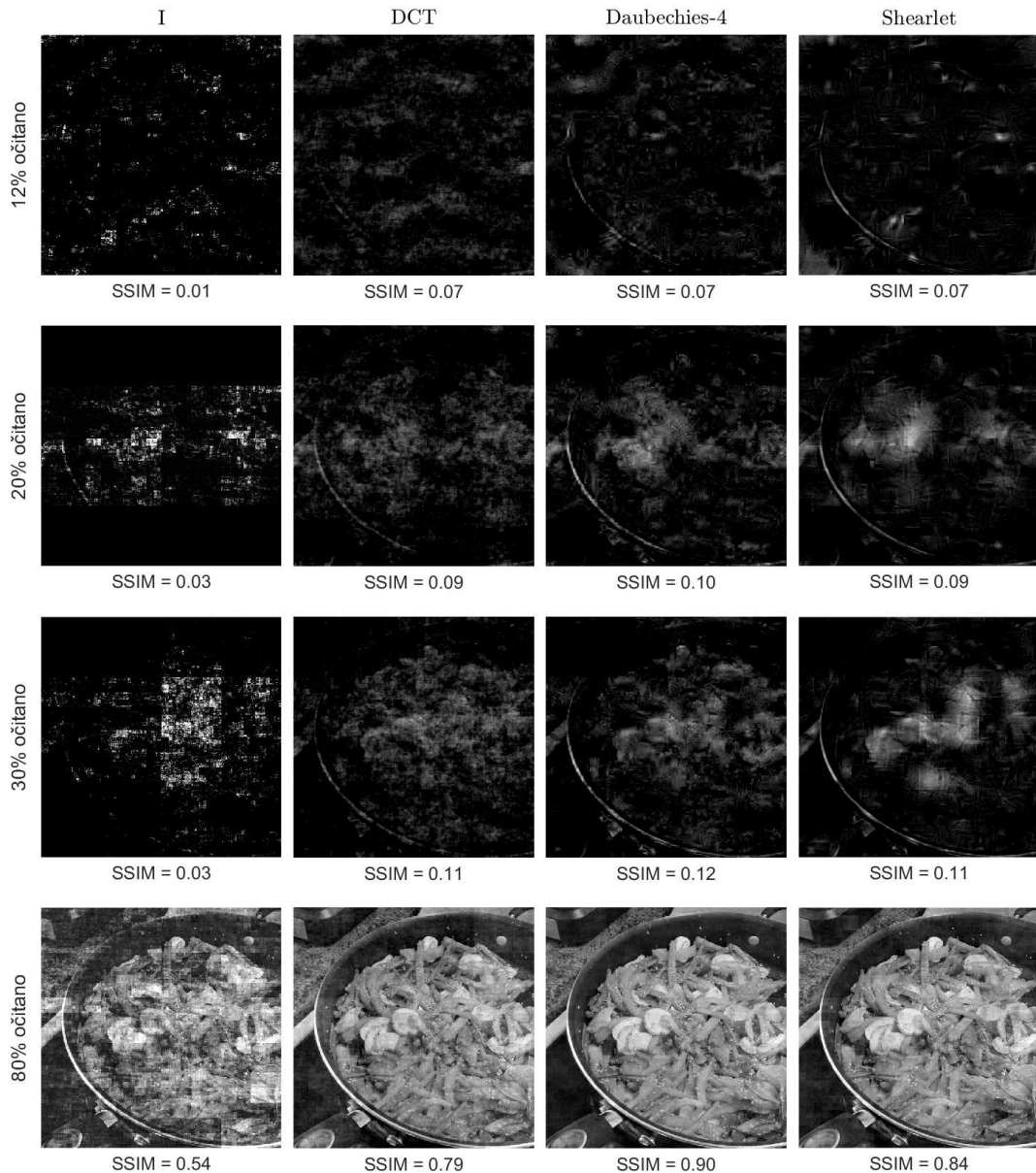
Na slici 3.9 prikazani su rezultati rekonstrukcije slike prikazane na 3.8 za različit broj dobivenih mjerenja te različite matrice transformacija Ψ . Treba nadodati da smo, isključivo u ovom primjeru, originalnu sliku skalirali na veličinu 128×128 zbog dugog izvršavanja algoritma rekonstrukcije.

Vidimo da ovdje izbor matrice nema velik utjecaj na uspješnost postupka. Sličnu uspješnost dobivamo neovisno o izabranoj matrici transformacije. S druge strane, prisutna je vizualna razlika u dobivenim rekonstrukcijama. Na nekim rekonstrukcijama dobro se raspoznaju osnovni oblici originalne slike, dok se na drugima bolje vide manji oblici i njihovi obrisi.

Ukupno najbolje rezultate dobivamo kada je Ψ matrica valične transformacije s Daubechies-4 prototipnim valićem te vrlo dobru rekonstrukciju možemo dobiti već s malim brojem očitanih uzoraka.

Slika 3.9: Rezultati rekonstrukcije s različitim matricama Ψ

Pretpostavimo sada da nemamo izbor matrice Φ , nego da je ona već zadana u postavci problema. Prelazimo na slučaj determinističkih matrica Φ i Ψ . Zbog jednostavnosti, pretpostavimo da je matrica očitavanja Φ Hadamardova matrica.

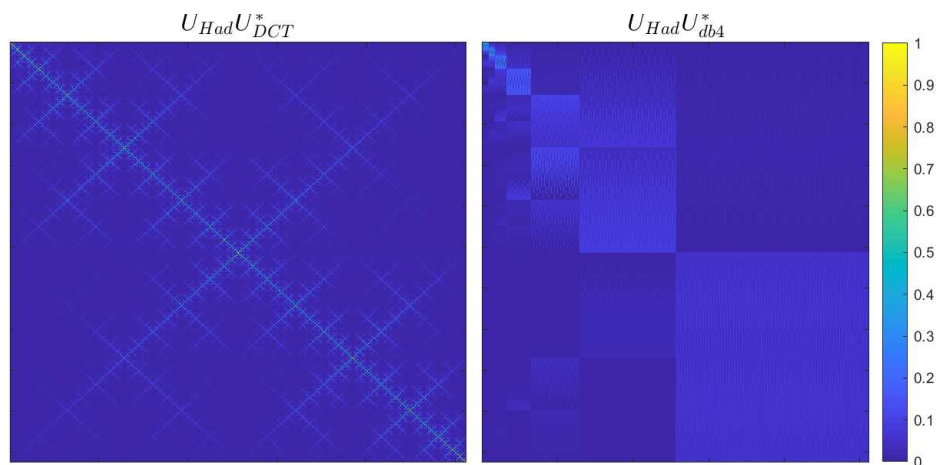


Slika 3.10: Rezultati rekonstrukcije s različitim matricama Ψ , $\mathbf{A} = \mathbf{U}_{\text{Had}} \Psi^*$

Na slici 3.10 prikazani su rezultati rekonstrukcije slike 3.8 kada je matrica očitavanja $\Phi = \mathbf{U}_{\text{Had}}$, a uzorci su uzeti uniformno. Uspoređujući uspješnost ove rekonstrukcije naspram uspješnosti rekonstrukcije kada su elementi matricom očitavanja imali Bernoullijevu razdiobu, možemo vidjeti puno lošije rezultate kada je prisutan mali broj uzoraka. Na 3.10 vidimo da, kada je dostupno manje ili jednako uzoraka od 30% elemenata cijele slike, početna slika nije niti približno razlučiva iz ponuđenih rekonstrukcija.

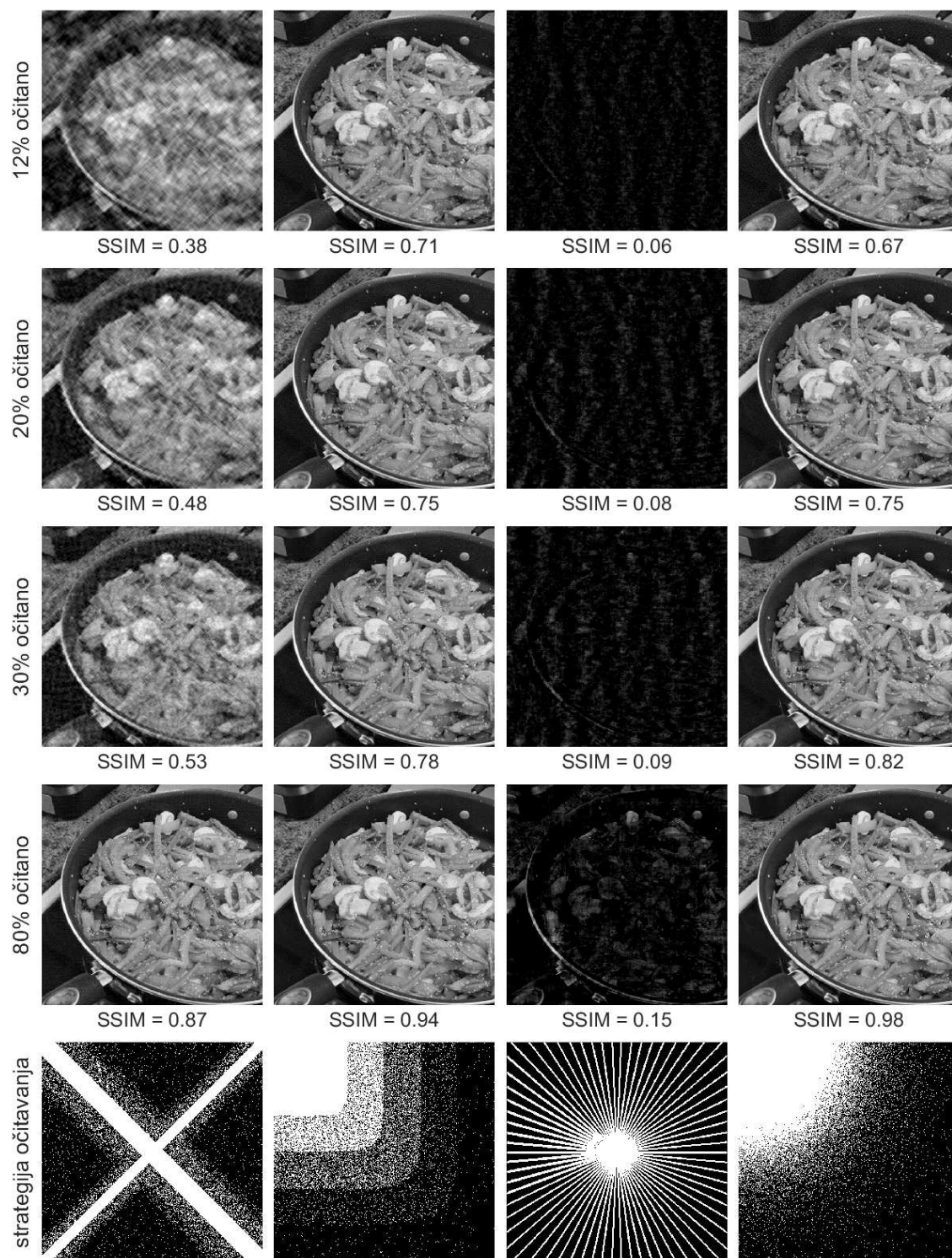
Poboljšano sažimajuće očitavanje

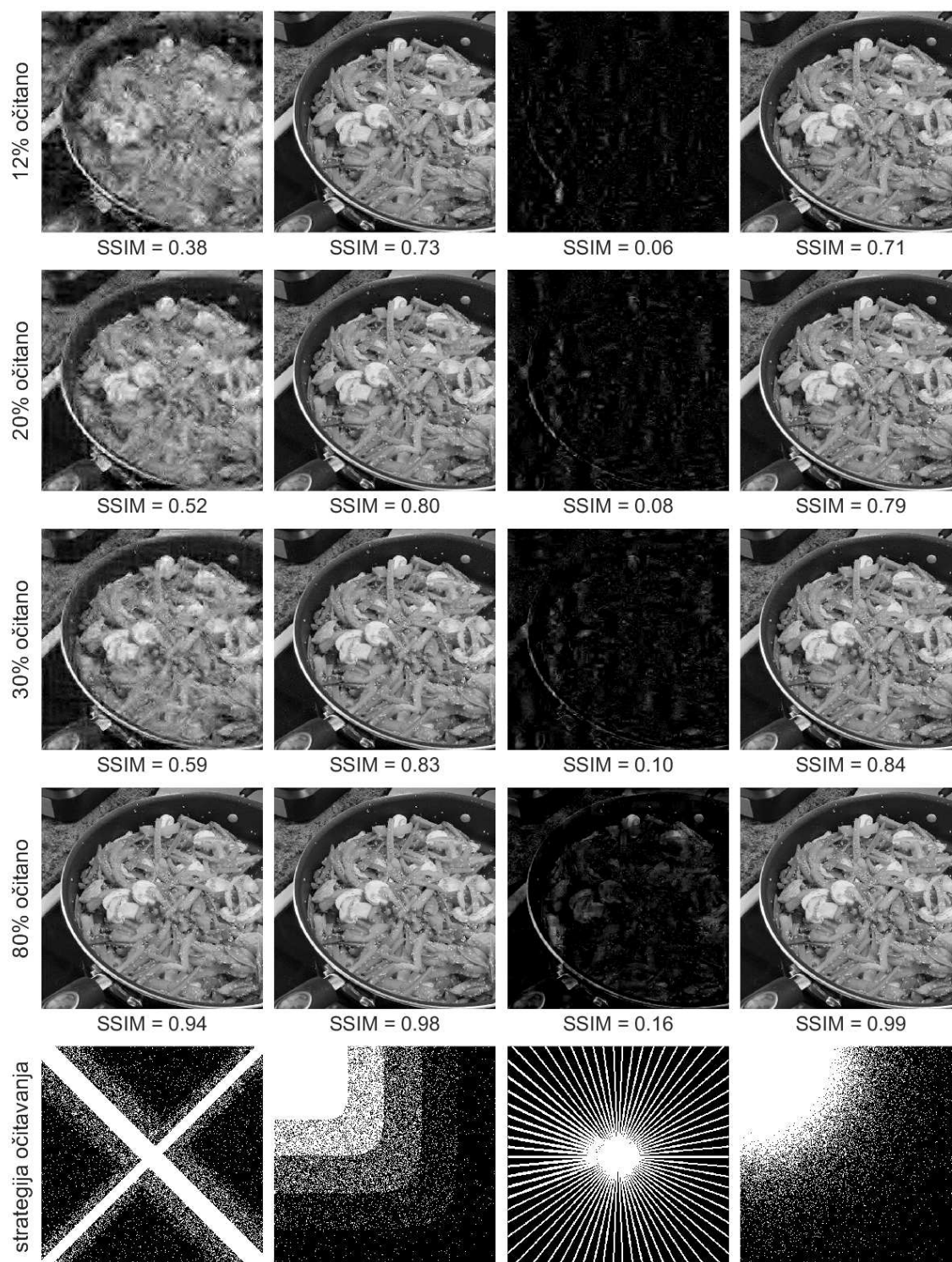
Zbog jednostavnosti, ograničit ćemo se na \mathbf{U}_{DCT} i $\mathbf{U}_{\text{Daubechies-4}}$ (skraćeno \mathbf{U}_{db4}), matrice diskretne kosinusne transformacije i valične s Daubechies-4 prototipnim valićem. Prisjetimo se kako izgledaju matrice $\mathbf{A} = \mathbf{U}_{\text{Had}}\Psi^*$ za različite baze \mathbf{U}_{DCT} i \mathbf{U}_{db4} . Na slici 3.11 prikazane su apsolutne vrijednosti matrice \mathbf{A} .



Slika 3.11: Vizualizacija koherentnosti

Ono što možemo uočiti je da su u prvom slučaju vjerojatno važniji uzorci očitani blizu glavne i sporedne dijagonale od onih ostalih. U drugom slučaju vjerujemo da će biti važni uzorci koji se nalaze blizu gornjih dijagonalnih blok matrica. Kako bismo ispitali te pretpostavke, napraviti ćemo rekonstrukciju originalne slike s četiri različite strategije očitavanja te usporediti rezultate. Prva strategija će očitavati više uzoraka oko glavne i sporedne dijagonale, druga i četvrta će koncentrirati uzorke oko gornjeg lijevog dijela slike, a treća će koncentrirati uzorke oko centra slike.

Slika 3.12: Rezultati rekonstrukcije uz $\mathbf{A} = \mathbf{U}_{\text{Had}} \mathbf{U}_{\text{DCT}}^*$

Slika 3.13: Rezultati rekonstrukcije uz $\mathbf{A} = \mathbf{U}_{\text{Had}} \mathbf{U}_{\text{db4}}^*$

Na slikama 3.12 i 3.13 prikazani su rezultati rekonstrukcije uz četiri opisane strategije očitavanja za $\mathbf{A} = \mathbf{U}_{\text{Had}}\mathbf{U}_{\text{DCT}}^*$, odnosno $\mathbf{A} = \mathbf{U}_{\text{Had}}\mathbf{U}_{\text{db4}}^*$. Očitani uzorci, kada je očitano 30% slike, prikazani su na dnu slike.

Uočimo neočekivanu uspješnost ove rekonstrukcije kada je odabrana prikladna strategija očitavanja. Već sa samo 12% očitane originalne slike možemo dobiti vrlo točnu rekonstrukciju, dok sa 30% očitane slike dobivamo rekonstrukciju usporedivu s onom koja se dobije kada očitamo 80% originalne slike bez dodanog uvjeta na matricu očitavanja, prikazanu na slici 3.9. Isto tako, treba imati na umu da korištene strategije očitavanja vrlo vjerojatno nisu optimalne te da je moguće dobiti bolje rezultate uz neke modifikacije.

Vodeći se idejom da elementi matrice $\Phi\Psi^*$ veće apsolutne vrijednosti sadrže više informacija o originalnoj slici od ostalih, pretpostavili bismo da će prva strategija očitavanja, ona koja uzorke koncentrira oko glavne i sporedne dijagonale, biti značajno bolja od ostalih. Zanimljivo je da to nije slučaj te da druge strategije daju bolje rezultate.

Poglavlje 4

Dodatak

generate_figures/draw_hadamard.m

```
1 n = 128;
2 U_Had = sqrt(n) * fwht(eye(n), n, 'sequency');
3
4 f = figure();
5 f.Position = 1.0e+03 * [ 1.4263    0.5737    0.5333
    0.4613];
6 colormap bone;
7 imagesc(U_Had, [-0.16, 0.0884]);
8 set(gca, 'xtick', []); set(gca, 'ytick', []);
9 set(gca, 'xticklabel', []); set(gca, 'yticklabel', []);
10
11 title('$U_{Had}$', 'interpreter', 'latex', 'FontSize', 15);
12
13 saveas(gcf, 'plots/hadamard.png');
```

generate_figures/draw_mother_wavelets.m

```
1 wnames = {'haar', 'db4', 'sym4', 'mexh'};
2 names = {'Haar', 'Daubechies-4', 'symlet-3', 'meksicki sesir
    (eng. mexican hat)'};
3
4 f = figure();
5 f.Position = 1.0e+03 * [1.0003    0.7423    0.8740
    0.5960];
```

```

6 for i = 1:4
7     ax = subplot(2, 2, i);
8     if i == 4
9         [psi, xval] = wavefun(wnames{i}, 14);
10    else
11        [~, psi, xval] = wavefun(wnames{i}, 14);
12    end
13
14    plot(xval, psi, 'LineWidth', 1); grid();
15    ylim([-1.7 1.7]);
16    xlim(xlim * 1.1 - mean(xlim) * 0.25);
17    title([names{i}])
18 end

```

generate_figures/visualize_asymptotic_coherence.m

```

1 im = rescale(rgb2gray(imread('data/cocol.png')));
2 n = size(im, 1);
3 y = dct2(im);
4
5 figure();
6 imshow(im);
7
8 f = figure();
9 f.Position = 1.0e+03 * [1.0003    0.9483    1.08    0.3900];
10 ax2 = subplot(1, 2, 1);
11 plot(y(:), 'LineWidth', 1); grid(); box();
12 xlim([-3000 2.7*1e5]);
13 ylim([-50 50]);
14
15 xlabel('DCT')
16
17 wave_name = 'haar';
18 nres = wmaxlev(n, wave_name);
19 [y, s] = wavedec2(im, nres, wave_name);
20 ax3 = subplot(1, 2, 2);
21 plot(y, 'LineWidth', 1); grid(); box();
22 xlim([-3000 2.7*1e5]);
23 ylim([-50 50]);

```

```

24 xlabel('DWT')
25
26 linkaxes([ax2 ax3], 'x');

```

generate_figures/visualize_coherence.m

```

1 N = 512*8;
2
3 Had = sqrt(N) * fwht(eye(N), N, 'sequency');
4
5 Cosin = dctmtx(N);
6 Haar = generate_wavelet(N, 'haar');
7 DB4 = generate_wavelet(N, 'db4');
8
9 mats = {Cosin, DB4};
10 names = {'$U_{Had}U_{DCT}^*$', '$U_{Had}U_{db4}^*$'};
11 f = figure();
12 f.Position = [798.3333 444.3333 880.0000 396.0000];
13
14 for i = 1:length(mats)
15     subaxis(1, length(mats), i, 1, 'Spacing', 0.015, '
        Padding', 0, 'Margin', 0.06);
16     imagesc(abs(Had * mats{i}'), [0 1]);
17     title(names{i}, 'interpreter', 'latex', 'FontSize', 16,
        'FontWeight', 'Normal');
18     set(gca, 'xticklabel', []);
19     set(gca, 'yticklabel', []);
20
21     if i == length(mats)
22         colorbar;
23     end
24 end
25
26 saveas(gcf, 'data/coh.png');

```

generate_figures/visualize_coherence_all.m

```

1 %% vizualizacija koherentnosti

```



```

2 n = 1024 * 4;
3
4 Hada = sqrt(n) * fwht(eye(n), n, 'sequency');
5 Four = (1 / sqrt(n)) * dftmtx(n);
6 Cosin = dctmtx(n);
7 Haar = generate_wavelet(n, 'haar');
8 DB1 = generate_wavelet(n, 'db4');
9
10 Psi = {Cosin, Four, "haar", "db4"};
11 names = {'DCT', 'DFT', 'Haar', 'db4'};
12
13 f = figure();
14 f.Position = 1.0e+03 * [1.0003    0.5630    0.8580
    0.7753];
15 colormap(jet);
16
17 for i = 1:length(Psi)
18     if isstring(Psi{i})
19         U2 = generate_wavelet(n, char(Psi{i}));
20     else
21         U2 = Psi{i};
22     end
23
24     if i == 4
25         U1 = Four;
26         U1_name = '$U_{DFT}U_{';
27     elseif i == 2
28         U1 = normc(randn(n, n));
29         U1_name = '$U_{gauss}U_{';
30         mutual_coherence(U1)
31     else
32         U1 = Hada;
33         U1_name = '$U_{Had}U_{';
34     end
35
36     max(max(abs(U1 * U2')))
37
38     subaxis(2, 2, mod(i, 2) + 1, ceil(i / 2), 'Spacing',
    0.05, 'Padding', 0.01, 'Margin', 0.09);

```

```

39     imagesc(abs(U1 * U2'), [0, 1]);
40
41     title([ U1_name, names{i}, '^*$', ], 'interpreter', '
         latex', 'FontSize', 15);
42
43     set(gca, 'xtick', []); set(gca, 'ytick', []);
44     set(gca, 'xticklabel', []); set(gca, 'yticklabel', []);
45 end
46
47 h = axes(f, 'visible', 'off');
48 colorbar(h, 'Position', [0.93 0.168 0.022 0.7]);
49
50 saveas(gcf, 'plots/coherence.png');

```

CS.m

```

1 function [s] = CS(x, opA, Phi, idx, epsilon)
2     n = size(x, 1);
3     size(x)
4
5
6     if idx ~= -1
7         y = Phi(x, 1);
8         y = y(idx);
9     else
10        y = Phi(x, 1);
11    end
12
13    y = y(:);
14
15    opts = spgSetParams('iterations', 2000);
16    s = spg_bpdn(opA, y, epsilon, opts);
17 end

```

generate_wavelet.m

```

1 function [W] = generate_wavelet(N, wname)
2     dim = log2(N);

```

```

3     nres = wmaxlev(N, wname);
4     if strcmp(wname, 'haar')
5         nres = nres - 1;
6     end
7     s = 2.^[dim-nres, (dim-nres):dim]';
8
9     W = zeros(N);
10    e_i = zeros(N, 1);
11    for i=1:N
12        e_i(i) = 1;
13        W(i, :) = waverec(e_i, s, wname);
14        e_i(i) = 0;
15    end

```

subsampling_schemes_DCT.m

```

1 %% dijagonalna strategija ocitavanja
2 function [idx] = subsampling_schemes_DCT(n, m, mode)
3     e = ones(n,1);
4
5     idx = full(spdiags([e e e e e e e e], -3:3, n, n));
6     idx = idx(n:-1:1, :);
7     idx = idx + full(spdiags([e e e e e e e e e e], -4:4, n, n
8         ));
9
10    x = 1:n;
11    levels = round([0, 0.06, 0.17, 0.225, 1] * n);
12
13    remainder = m - sum(sum(idx == 1));
14    percent = 1.0;
15
16    for k=2:length(levels)
17        new_idx = [];
18
19        for i = levels(k-1):levels(k) % pravci
20            x_ = x(1 : (n - i));
21            new_idx = [new_idx sub2ind([n n], x_, x_ + i)];
22            new_idx = [new_idx sub2ind([n n], x_ + i, x_)];
23        end

```

```

23
24     if mode == 1
25         for i = round((levels(k-1):levels(k)) / 2) %
26             pravci
27                 x_ = x(1 : (n - i));
28                 new_idx = [new_idx sub2ind([n n], n + 1 - x_,
29                     x_ + i)];
30                 new_idx = [new_idx sub2ind([n n], x_, n + 1 -
31                     x_ - i)];
32             end
33         elseif mode == 2
34             warning('off','all')
35             for i = (round(levels(k-1):levels(k)) / 4) %
36                 pravci
37                     x_ = x(1 : (n - i));
38                     new_idx = [new_idx sub2ind([n n], n + 1 - x_,
39                         x_ + i)];
40                     new_idx = [new_idx sub2ind([n n], x_, n + 1 -
41                         x_ - i)];
42                 end
43                 warning('on','all')
44             end
45
46         new_idx = setdiff(new_idx, find(idx == 1));
47
48         remainder = m - sum(sum(idx == 1));
49         n_sampled = min(length(new_idx), round(remainder *
50             percent));
51         sample = randsample(new_idx, n_sampled);
52         idx(sample) = 1;
53
54         remainder = m - sum(sum(idx == 1));
55         percent = percent / 1.5;
56     end
57
58     if sum(sum(idx == 1)) ~= m
59         idx(randsample(find(idx ~= 1), round(m - sum(sum(idx
60             == 1))) )) = 1;
61     end

```

```

54
55     % imshow(idx);
56     idx = find(idx == 1);
57 end

```

l1_l2_comparison.m

```

1 %% učitaj sliku
2 im = rescale(imread('data/kg_.png'));
3 im((im > 0.3) & (im < 0.7)) = 0;
4 x = im(:);
5 n = numel(x);
6
7 %% eksperiment 1 – uspješnost rekonstrukcije s različitim
   uvjetima
8 f = figure();
9 start = 400; w = 120; h = 200;
10 f.Position = [1.0063e+03 226.3333 578 926];
11 perc = [0.12, 0.2, 0.3, 0.8];%
12 for row=1:4
13     m = round(n * perc(row));
14     A = A_(idx(1:m), :);
15     y = A * x;
16
17     for epsilon=-1:1:1
18         if epsilon == -1
19             s = lsqr(A, y);
20         else
21             s = spg_bpdn(A, y, epsilon);
22         end
23
24         subaxis(4, 3, epsilon+2, row, 'Spacing', 0.06, '
           Padding', 0, 'Margin', 0.05);
25         imshow(reshape(s, size(im)));
26
27         label_v = xlabel(sprintf('SSIM = %.2f', round(ssim(
           reshape(s, size(im)), reshape(x, size(im))), 2)),
           'FontSize', 12);
28         label_v.Position(2) = label_v.Position(2) - 12;

```

```

29
30     if row == 1
31         if epsilon == -1
32             title('$I_{2}$ minimizacija', 'FontSize',
33                 14, 'FontWeight', 'Normal', 'interpreter',
34                 'latex');
35         else
36             title(sprintf('%c = %d', char(949), round(
37                 epsilon)), 'FontSize', 14, 'FontWeight',
38                 'Normal');
39         end
40     end
41
42     if epsilon == -1
43         label_h = ylabel(sprintf('%d%% ocitano', round(
44             perc(row) * 100)), 'FontSize', 12);
45         label_h.Position(1) = label_h.Position(1) + 15;
46     end
47
48     end
49
50 saveas(gcf, 'plots/11_12.png');
51
52
53 %% dodatak – odnos uspjesnosti rekonstrukcije i broja
54   uzoraka
55 n_tests = 5;
56
57 f = figure();
58 f.Position = [910.3333 568.3333 988.0000 420.0000];
59 perc = 10.^(-1.4:0.15:0);
60 ssims = zeros(n_tests, length(perc));
61 for test=1:n_tests
62     A_ = randn(n, n);
63     idx = randperm(n);
64     for i = 1:length(perc)
65         m = round(n * perc(i));
66         A = A_(idx(1:m), :);

```

```

62         y = A * x;
63         im_rec = reshape(spg_bpdn(A, y, 1), size(im));
64         ssims(test, i) = ssim(im_rec, reshape(x, size(im)));
65     end
66 end
67
68 colormap parula;
69 plot(perc, mean(ssims), 'LineWidth', 2, 'Color',
70      [65,125,193] / 255);
71 grid();
72 box off;
73 xlabel('ocitano, %', 'FontSize', 13);
74 ylabel('SSIM', 'FontSize', 13);
75 ylim([-0.02, 1.02]);
76 saveas(gcf, 'plots/ssim.png');

```

new_CS_methods.m

```

1 %% učitaj sliku
2 im = rescale(rgb2gray(imread('data/coco5.png')));
3
4 % ako se koristi Bernoullijeva matrica, treba smanjiti sliku
5 % im = imresize(im, 0.25);
6
7 n = size(im, 1);
8
9
10 %% postavke
11 epsilon = 1;
12 dwtmode('per', 'nodisp');
13 shearlet_system = SLgetShearletSystem2D(0, size(im, 1), size
14      (im, 2), 4);
15
16 %% matrica ocitavanja
17 Had = sqrt(n) * fwht(eye(n), n, 'sequency');
18 opHad = @(x, mode) fastwht(fastwht(x, [], 'sequency'),'', [],
19      'sequency') * size(x, 1);

```

```

19 %% matrice transformacije
20 opI = @(x, mode) I_(x, mode, n);
21 opDCT = @(x, mode) dct2_(x, mode, n);
22 opHaar = @(x, mode) dwt2_(x, mode, 'haar', n);
23 opDaub = @(x, mode) dwt2_(x, mode, 'db4', n);
24 opShear = @(x, mode) dst2_(x, mode, shearlet_system, n);
25
26 %% eksperiment1 - razlicite baze i Bernoullijeva matrica
27 f = figure();
28 f.Position = 1e+03 * [1.0003    0.2043    0.9840
    1.1340];
29
30 Bernoulli = randsrc(round(n*n*0.8), n*n, [0, 1]);
31 percents = [0.12, 0.2, 0.3, 0.5];
32 Psis = {opDaub, opDCT, opDaub, opShear};
33 Psi_names = {'I', 'DCT', 'Daubechies-4', 'Shearlet'};
34 idx = -1;
35
36 for i = 1:4
37     perc = percents(i);
38     m = round(n * n * perc);
39     Phi = @(x, mode) bernoulli_(x, mode, Bernoulli(1:m, :),
    n);
40     for j = 1:length(Psis)
41         Psi = Psis{j};
42         opA = @(x, mode) op_(x, mode, Phi, Psi, idx, n);
43
44         s = CS(im, opA, Phi, idx, epsilon);
45
46         subaxis(length(percents), length(Psis), j, i, '
    Spacing', 0.011, 'Padding', 0, 'Margin', 0.05);
47         im_rec = sc(Psi(s, 2));
48         imshow(im_rec);
49
50         label_v = xlabel(sprintf('SSIM = %.2f', round(ssim(
    im_rec, im), 2)), 'FontSize', 12);
51         label_v.Position(2) = label_v.Position(2) - 10;
52
53         if j == 1

```



```

54         label_h = ylabel(sprintf('%d%% ocitano', round(
55             perc * 100)), 'FontSize', 13);
56         label_h.Position(1) = label_h.Position(1) + 10;
57     end
58     if i == 1
59         title(Psi_names{j}, 'FontSize', 14, 'FontWeight'
60             , 'Normal', 'interpreter', 'latex');
61     end
62 end
63 saveas(gcf, 'plots/p1.png');
64
65 %% eksperiment2 - razlicite baze i Hadamardova matrica,
66    uniformno %%
67
68 f = figure();
69 f.Position = 1e+03 * [1.0003    0.2043    0.9840
70    1.1340];
71 percents = [0.12, 0.2, 0.3, 0.8];
72 Psis = {opI, opDCT, opDaub, opShear};
73 Psi_names = {'I', 'DCT', 'Daubechies-4', 'Shearlet'};
74 Phi = opHad;
75
76 for i = 1:4
77     perc = percents(i);
78     m = round(numel(im) * perc);
79     idx = randperm(n*n, m);
80     for j = 1:length(Psis)
81         Psi = Psis{j};
82         opA = @(x, mode) op_(x, mode, Phi, Psi, idx, n);
83
84         s = CS(im, opA, Phi, idx, epsilon);
85
86         subaxis(length(percents), length(Psis), j, i, '
87             Spacing', 0.011, 'Padding', 0, 'Margin', 0.05);
88         im_rec = sc(Psi(s, 2));

```

```

88         imshow(im_rec);
89
90         label_v = xlabel(sprintf('SSIM = %.2f', round(ssim(
91             im_rec, im), 2)), 'FontSize', 12);
92         label_v.Position(2) = label_v.Position(2) - 45;
93
94         if j == 1
95             label_h = ylabel(sprintf('%d%% ocitano', round(
96                 perc * 100)), 'FontSize', 13);
97             label_h.Position(1) = label_h.Position(1) + 45;
98         end
99         if i == 1
100             title(Psi_names{j}, 'FontSize', 14, 'FontWeight',
101                 , 'Normal', 'interpreter', 'latex');
102         end
103     end
104
105     saveas(gcf, 'plots/p2.png');
106
107     %% eksperiment 3. – hadamard, dct s posebnom strategijom
108     o itavanja %%
109     f = figure();
110     f.Position = 1e+03 * [1.0003    0.2043    0.8160
111         1.1340];
112
113     percents = [0.12, 0.2, 0.3, 0.8];
114     Psi = opDCT;
115     Phi = opHad;
116
117     for i = 1:4
118         perc = percents(i);
119         m = round(numel(im) * perc);
120         for mode = 1:4
121             switch mode

```

```

122         idx = cil_sph2_exp(n, m, 2, 1, 18, 3, 8);
123     case 3
124         idx = cil_sph2_radial_lines(n, m, 35, true);
            idx = idx(randperm(numel(idx))); idx =
                idx(1:m);
125     case 4
126         idx = cil_sph2_power_law(n, m, 3);
127     end
128
129     opA = @(x, mode) op_(x, mode, Phi, Psi, idx, n);
130     s = CS(im, opA, Phi, idx, epsilon);
131
132
133     subaxis(length(percents) + 1, 4, mode, i, 'Spacing',
            0.011, 'Padding', 0, 'Margin', 0.05);
134     im_rec = sc(Psi(s, 2));
135     imshow(im_rec);
136
137     label_v = xlabel(sprintf('SSIM = %.2f', round(ssim(
            im_rec, im), 2)), 'FontSize', 12);
138     label_v.Position(2) = label_v.Position(2) - 55;
139
140     if mode == 1
141         label_h = ylabel(sprintf('%d%% ocitano', round(
            perc * 100)), 'FontSize', 13);
142         label_h.Position(1) = label_h.Position(1) + 55;
143     end
144
145     if i == 3
146         z = zeros(n, n);
147         z(idx) = 1;
148         subaxis(5, 4, mode, 5, 'Spacing', 0.011, '
            Padding', 0, 'Margin', 0.05);
149         imshow(z);
150         if mode == 1
151             label_h = ylabel('strategija o itavanja', '
            FontSize', 13);
152             label_h.Position(1) = label_h.Position(1) +
                55;

```

```

153         end
154     end
155 end
156 end
157
158 saveas(gcf, 'plots/p3.png');
159
160
161 %% eksperiment 4. – hadamard, dwt-db4 s posebno strategijom
    o itavanja %%
162 f = figure();
163 f.Position = 1e+03 * [1.0003    0.2043    0.8160
    1.1340];
164
165 percents = [0.12, 0.2, 0.3, 0.8];
166 Psi = opDaub;
167 Phi = opHad;
168
169 for i = 1:4
170     perc = percents(i);
171     m = round(numel(im) * perc);
172     for mode = 1:4
173         switch mode
174             case 1
175                 idx = subsampling_schemes_DCT(n, m, mode);
176             case 2
177                 idx = cil_sph2_exp(n, m, 2, 1, 18, 3, 8);
178             case 3
179                 idx = cil_spf2_radial_lines(n, m, 35, true);
180                 idx = idx(randperm(numel(idx))); idx =
181                     idx(1:m);
182             case 4
183                 idx = cil_sph2_power_law(n, m, 3);
184         end
185     end
186     opA = @(x, mode) op_(x, mode, Phi, Psi, idx, n);
187     s = CS(im, opA, Phi, idx, epsilon);

```

```

188     subaxis(length(percents) + 1, 4, mode, i, 'Spacing',
189             0.011, 'Padding', 0, 'Margin', 0.05);
190     im_rec = sc(Psi(s, 2));
191     imshow(im_rec);
192
193     label_v = xlabel(sprintf('SSIM = %.2f', round(ssim(
194         im_rec, im), 2)), 'FontSize', 12);
195     label_v.Position(2) = label_v.Position(2) - 55;
196
197     if mode == 1
198         label_h = ylabel(sprintf('%d%% ocitano', round(
199             perc * 100)), 'FontSize', 13);
200         label_h.Position(1) = label_h.Position(1) + 55;
201     end
202
203     if i == 1
204         title('DWT', 'FontSize', 14, 'FontWeight', '
205             Normal', 'interpreter', 'latex');
206     end
207
208     if i == 3
209         z = zeros(n, n);
210         z(idx) = 1;
211         subaxis(5, 4, mode, 5, 'Spacing', 0.011, '
212             Padding', 0, 'Margin', 0.05);
213         imshow(z);
214         if mode == 1
215             label_h = ylabel('strategija ocitavanja', '
216                 FontSize', 13);
217             label_h.Position(1) = label_h.Position(1) +
218                 55;
219         end
220     end
221 end
222 end
223 end
224
225 end
226
227 saveas(gcf, 'data/p4.png');
228
229

```

```

220
221
222 %% pomocne funkcije %%
223 function [im] = sc(im)
224     im(im < 0) = 0;
225     im(im > 1) = 1;
226 end
227
228 function [y] = op_(x, mode, Phi, Psi, idx, n)
229     if mode == 1
230         z = Psi(x, 2);
231
232         y = Phi(z, 1);
233         if idx ~= -1
234             y = y(idx);
235         end
236         y = y(:);
237     else
238         if idx ~= -1
239             z = zeros(n, n);
240             z(idx) = x(:);
241         else
242             z = x;
243         end
244         z = Phi(z, 2);
245         y = Psi(z, 1);
246     end
247 end
248
249
250 function [y] = bernoulli_(x, mode, Phi, n)
251     if mode == 1
252         y = Phi * x(:);
253     else
254         y = Phi' * x;
255         y = reshape(y, [n, n]);
256     end
257 end
258

```

```

259
260
261 function [y] = I_(x, mode, n)
262     if mode == 1
263         y = x(:);
264     else
265         y = reshape(x, [n, n]);
266     end
267 end
268
269
270 function [y] = dct2_(x, mode, n)
271     if mode == 1
272         y = dct2(x);
273         y = y(:);
274     else
275         x = reshape(x, [n, n]);
276         y = idct2(x);
277     end
278 end
279
280
281 function [y] = dwt2_(x, mode, wave_name, n)
282     nres = wmaxlev(n, wave_name);
283
284     if mode == 1
285         [y, ~] = wavedec2(x, nres, wave_name);
286         y = y';
287     else
288         dim = round(log2(n));
289         S = 2.^[dim-nres, (dim-nres):dim; dim-nres, (dim-
290             nres):dim]';
291         x = reshape(x, [], 1);
292         y = waverec2(x, S, wave_name);
293     end
294 end
295
296 function [y] = dst2_(x, mode, shearlet_system, n)

```

```
297     dim3 = length(shearlet_system.RMS);
298
299     if mode == 1
300         y = SLsheardec2D(x, shearlet_system);
301         y = reshape(y, [n*n*dim3, 1]);
302     else
303         x = reshape(x, [n, n, dim3]);
304         y = SLshearrec2D(x, shearlet_system);
305     end
306 end
```


Bibliografija

- [1] *CurveLab*, <http://www.curvelet.org/software.html>.
- [2] *fastwht*, <https://github.com/vegarant/fastwht>.
- [3] *ShearletReweighting*, <https://github.com/jky-ma/ShearletReweighting>.
- [4] *subaxis*, <https://www.mathworks.com/matlabcentral/fileexchange/3696-subaxis-subplot>.
- [5] B. Adcock i A. C. Hansen, *Compressive Imaging: Structure, Sampling, Learning*, Cambridge University Press, 2021.
- [6] B. Adcock, A. C. Hansen, C. Poon i B. Roman, *Breaking the coherence barrier: A new theory for compressed sensing*, (2013).
- [7] Vegard Antun, *Cilib – A software library for compressive imaging*, 2020, <https://github.com/vegarant/cilib>.
- [8] Y. Arjoune, N. Kaabouch, H. E. Ghazi i A. Tamtaoui, *Compressive Sensing: Performance Comparison Of Sparse Recovery Algorithms*, (2018), <https://arxiv.org/abs/1801.09744>.
- [9] E. van den Berg i M. P. Friedlander, *SPGL1: A solver for large-scale sparse reconstruction*, December 2019, <https://friedlander.io/spgl1>.
- [10] Y. C. Eldar i G. Kutyniok, *Compressed Sensing Theory and Applications*, (2012).
- [11] S. Foucart i H. Rauhut, *A Mathematical Introduction to Compressive Sensing*, (2013).
- [12] K. Grozdanić, *CompressiveSensing*, <https://github.com/kgrozdanic/CompressiveSensing>.
- [13] G. Kutyniok, *Theory and applications of compressed sensing*, GAMM-Mitteilungen **36** (2013), br. 1, 79–101, <https://onlinelibrary.wiley.com/doi/abs/10.1002/gamm.201310005>.

- [14] G. Kutyniok i D. Labate, *Shearlets: Multiscale Analysis for Multivariate Data*, ožujak 2012, ISBN 0817683151.
- [15] G. Kutyniok, W. Q. Lim i R. Reisenhofer, *ShearLab 3D: Faithful Digital Shearlet Transforms Based on Compactly Supported Shearlets*, **42** (2016), br. 1, ISSN 0098-3500, <https://doi.org/10.1145/2740960>, www.shearlab.org.
- [16] T. Y. Lin, M. Maire, S. J. Belongie, L. D. Bourdev, R. B. Girshick, J. Hays, P. Perona, D. Ramanan, Dollár P. i Zitnick C. L., *Microsoft COCO: Common Objects in Context*, CoRR **abs/1405.0312** (2014), <http://arxiv.org/abs/1405.0312>.
- [17] MATLAB, *version 9.4.0 (R2018a)*, The MathWorks Inc., Natick, Massachusetts, 2018.
- [18] S. Qaisar, R. M. Bilal, W. Iqbal, M. Naureen i S. Lee, *Compressive sensing: From theory to applications, a survey*, Journal of Communications and Networks **15** (2013), br. 5, 443–456.
- [19] B. Roman, A. C. Hansen i B. Adcock, *On asymptotic structure in compressed sensing*, (2014).
- [20] D. Seršić, *Valići i filtarski slogovi*, Element, 2015.
- [21] V. Studer, J. Bobin, M. Chahid, H. S. Mousavi, E. Candes i M. Dahan, *Compressive fluorescence microscopy for biological and hyperspectral imaging*, Proceedings of the National Academy of Sciences of the United States of America **109** (2012), br. 26, E1679–87.
- [22] A. M. Tillmann i M. E. Pfetsch, *The Computational Complexity of the Restricted Isometry Property, the Nullspace Property, and Related Concepts in Compressed Sensing*, IEEE Transactions on Information Theory **60** (2014), br. 2, 1248–1259.
- [23] M. J. Wainwright, T. Hastie i R. Tibshirani, *Statistical Learning with Sparsity: the Lasso and Generalizations*, (2015).
- [24] Z. Wang, A.C. Bovik, H.R. Sheikh i E.P. Simoncelli, *Image quality assessment: from error visibility to structural similarity*, IEEE Transactions on Image Processing **13** (2004), br. 4, 600–612.

Sažetak

U ovome radu prezentiramo metodu sažimajućeg očitavanja koja nam omogućuje rekonstrukciju signala iz vrlo malog broja mjerenja, uz uvjet rijetkosti signala u nekoj bazi. Rekonstrukciju dobivamo preko problema minimizacije l_1 norme, koja pod određenim uvjetima daje ista rješenja kao problem minimizacije l_0 norme.

U prvom poglavlju formuliran je problem sažimajućeg očitavanja i dana je usporedba s klasičnim pristupima sažimanju podataka. Uvedene su definicije i pojmovi koji se kasnije koriste.

U drugom poglavlju detaljno je predstavljena teorija sažimajućeg očitavanja: pokazani su uvjeti pod kojima će postupak biti uspješan te su dane granice točnosti rekonstrukcije ovisno o parametrima mjerenja.

U trećem poglavlju pokazali smo kako sažimajuće očitavanje radi u problemima rekonstrukcije slika, te smo uveli poboljšanja na sam postupak koja dovode do boljih rezultata rekonstrukcije.

Summary

In this paper, we investigate compressive sensing, a technique that allows for the reconstruction of a signal from a limited number of measurements, provided that the signal exhibits sparsity in some orthonormal basis. The signal is reconstructed by solving a l_1 minimization problem is solved, which in the right conditions, yields a solution that is equivalent to a l_0 minimization problem.

In the first chapter, we formulate the compressive sensing problem and give compare it to the standard approach for signal compression. Key definitions and concepts are introduced.

In the second chapter, we present a comprehensive theory of compressive sensing, demonstrating the conditions under which the method is guaranteed to be successful, and we providing the bounds on the reconstruction error based on the measurement process.

In the third chapter, we demonstrate the application of compressive sensing to image reconstruction and modifications that significantly improve the reconstruction accuracy.

Životopis

Rođen sam 19. siječnja 1990. godine u Zagrebu. Nakon završene osnovne škole Josipa Račića, upisujem XV. gimnaziju u Zagrebu. Maturiram 2017. godine, nakon čega upisujem preddiplomski studij matematike na Prirodoslovno-matematičkom fakultetu u Zagrebu. Studij završavam 2020. godine i upisujem diplomski sveučilišni studij Matematička statistika na istom fakultetu. Za izniman uspjeh u studiju dodijeljene su mi tri nagrade Matematičkog odsjeka.