SVEUČILIŠTE U ZAGREBU

FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 1166

NAPREDNA REKONSTRUKCIJA SLIKE U POZITRONSKOJ EMISIJSKOJ TOMOGRAFIJI

Ankica Gogić

Zagreb, lipanj 2015.

SVEUČILIŠTE U ZAGREBU FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA ODBOR ZA DIPLOMSKI RAD PROFILA

Zagreb, 9. ožujka 2015.

DIPLOMSKI ZADATAK br. 1166

| Pristupnik: | Ankica Gogić (0036455529) |
|-------------|---------------------------|
| Studij: | Računarstvo |
| Profil: | Računalno inženjerstvo |

Zadatak: Napredna rekonstrukcija slike u pozitronskoj emisijskoj tomografiji

Opis zadatka:

Pozitronska emisijska tomografija (PET) jedna je od osnovnih metoda 3D slikovnog prikaza metaboličke aktivnosti organizma nakon unosa radioizotopom obilježene bioaktivne molekule (primjer: analogon šećera - fluordeoksiglukoza). Prilikom snimanja, po dva gama fotona se kreću u aproksimativno suprotnim smjerovima te pogađaju kristalne detektore postavljene u rotirajući prsten. Koincidentne detekcije (unutar 6 - 12 ns) odgovaraju linijama u prostoru iz kojih se rekonstruira 3D objekt, a linije se mogu grupirati u projekcijske slike - sinograme. Rekonstrukcija se često izvodi algoritmom filtrirane povratne projekcije (eng. filtered back projection - FPB). Rekonstruirana 3D slika ograničene rezolucije redovito sadrži šum i artefakte uslijed nedostataka mjerne metode. S druge strane, sažeto očitavanje (eng. compressive sensing) je moderan pristup koji se oslanja na dvije linearne transformacije i optimizacijsku metodu za vrlo uspješnu rekonstrukciju mjernih podataka. Jedna transformacija osigurava rijetku reprezentaciju promatranih 3D objekata, a druga je dio mjernog sustava. Uz zadovoljene pretpostavke, rezultati mogu biti značajno iznad Nyquistove granice. U okviru diplomskog rada potrebno je osmisliti i realizirati napredne rekonstrukcijske metode za rekonstrukciju PET snimaka visoke kvalitete.

Zadatak uručen pristupniku: 13. ožujka 2015. Rok za predaju rada: 30. lipnja 2015.

Mentor:

Prof. dr. sc. Damir Seršić

Djelovođa: Dana

Prof. dr. sc. Danko Basch

Predsjednik odbora za diplomski rad profila:

Prof. dr. sc. Mario Žagar

Posebna zahvala kolegi Robertu Bagariću na pomoći sa interpretacijom podataka, te kolegama Janji Paliski, Antoniu Zemuniku i Nikolini Očić na pomoći pri interpretaciji compressive sensing metode.

Sadržaj

| 1. | Uvod | | 8 |
|----|-----------|---|----|
| 2. | Teorijs | ka podloga | 10 |
| 2 | 2.1. Po | zitronska emisijska tomografija | 10 |
| | 2.1.1. | Priprema subjekta | 10 |
| | 2.1.2. | Princip rada | 11 |
| 2 | 2.2. Lis | t Mode Format | 13 |
| | 2.2.1. | LMF alat | 14 |
| | 2.2.2. | Format zapisa događaja | 14 |
| | 2.2.3. | Geometrija adresiranja kristala | 15 |
| 2 | 2.3. Sir | nogram | 18 |
| | 2.3.1. | Generiranje sinograma | 18 |
| | 2.3.2. | Pretvorba LMF formata u sinogram | 19 |
| | 2.3.3. | Pretvorba sinograma u sliku | 22 |
| 2 | 2.4. Co | mpressive sensing | 26 |
| | 2.4.1. | Podaci | 26 |
| | 2.4.2. | Metodologija | 27 |
| | 2.4.3. | Rekonstrukcija | 28 |
| 3. | Izvedba | a rekonstrukcije Compressive sensing metodom | 30 |
| 3 | 3.1. Ula | azni podaci i interpretacija | 30 |
| | 3.1.1. | LMF datoteka sa pojedinačnim događajima | 30 |
| | 3.1.2. | LMF datoteka sa koincidencijama | 31 |
| 3 | 3.2. Kre | eiranje matrica za compressive sensing metodu | 32 |
| 3 | 3.3. Izra | ada i provedba compressive sensing metode | 33 |
| 4. | Rezulta | ati | 35 |
| 5. | Zaključ | ak | 39 |

| 6. | Literatura | 40 |
|------|----------------|----|
| Prog | ramska potpora | 44 |

Popis oznaka i kratica

| engl | engleski |
|-------|---|
| СТ | Computed tomography |
| MRI | Magnetic Resonance Imaging |
| PET | Pozitronska emisijska tomografija |
| SPECT | Single Photon Emission Computed Tomography |
| CS | Compressive sensing |
| FDG | Flordeoksiglukoza |
| FLT | F-flortimidin |
| LOR | Line of response (Linija odgovora) |
| LMF | List Mode Format |
| BP | Back Projection |
| FBP | Filtered Back Projection |
| ART | Algebraic Reconstruction Technique |
| MLEM | Maximum Likelihood Expectation Maximization |
| DWT | Discrete Wavelet Transform |
| SWT | Stationary Wavelet Transform |

Popis slika

| 12 |
|----|
| 13 |
| 15 |
| 15 |
| 16 |
| 17 |
| 18 |
| 19 |
| 20 |
| 21 |
| 23 |
| 24 |
| 24 |
| 25 |
| 27 |
| 33 |
| 35 |
| 36 |
| 37 |
| 37 |
| |

1. Uvod

Počevši od 1895. godine kada je Wilhelm Conrad Röntgen pronašao revolucionarno otkriće - X-zrake, zasnovan je pojam nuklearne medicine, te se ona nevjerojatnom brzinom razvija u smjeru što neinvazivnijih metoda pregledavanja pacijenata u svrhu dijagnosticiranja njihova stanja. Razvitku su pojma pridonijeli i Henri Becquerel sa svojim pokusima s uranijem 1896. godine, te Frederick Proescher koji 1913. godine u svojoj studiji o tretiranju pojedinih bolesti spominje intravenozno injektiranje radioterapije. Sâm pojam radioaktivnosti postavila je 1903. godine glasovita znanstvenica Marie Curie.

Daljnji je razvoj napredovao u smjeru otkrivanja pogodnih radioaktivnih izotopa, a prvi su radioaktivni izotop ustanovili 1934. godine Irene Curie i Frederic Joliot. Tim je činom, u kombinaciji sa izumom Lawrence-ovog ciklotrona 1932. godine, službeno zasnovana nuklearna medicina. Konačne formacije radioaktivnih izotopa pogodnih za upotrebu u medicini formirane su tek trideset godina kasnije, nakon čega njihova upotreba postaje globalno raširena.

Usporedno s razvijanjem radioaktivnih izotopa razvijaju se i strojevi koji su sposobni detektirati emitirane valove izotopa, te tako prikazati pojavu promatranu pomoću izotopa. Razvoj kreće sa uređajem za detekciju X-zraka 1957. godine, te se nastavlja preko mamografije 1969. godine, CT (engl. *Computed Tomography*) uređaja iz 1972. godine i MRI (engl. *Magnetic Resonance Imaging*) uređaja iz 1977. godine. Navedeni uređaji fokusirali su se na prikazivanje anatomije i strukture tijela koji se nalazi unutar njih.

Novije generacije uređaja koji su se dalje razvijali fokusirali su se na prikaz funkcionalnosti dijelova tijela koje se nalazi unutar njih. Tim uređajima pripadaju PET (engl. *Positron Emission Tomography*) uređaj koji je otkriven 1970.-ih, SPECT (engl. *Single Photon Emission Computed Tomography*) uređaj iz 1980. godine, te f - MRI (funkcijski MRI) uređaj iz 1990. godine.

Osim podjele po funkcionalnosti prikaza dijelova tijela, uređaji za detekciju radioaktivnog zračenja dijele se na ionizirajuće i neionizirajuće, te na uređaje koji detektiraju emisiju zračenja koja dolazi iz tijela (na primjer PET i SPECT) i na

uređaje koji zrače prema tijelu koje se nalazi unutar uređaja te sakupljaju informacije detektorima koji se nalaze iza tijela (na primjer CT i mamografija).

Svi se izlazni podaci iz dotičnih uređaja na neki način više ili manje uspješno transformiraju u format slike prihvatljiv za analizu. Pritom se koriste metode rekonstrukcije slike, za koje je potrebna velika količina podataka kako bi se rekonstrukcija mogla obaviti. Stoga će u ovom radu biti govora o metodi izrade slike kvalitete veće no što je trenutna, a sa manje potrebnih podataka.

Naime, jedan od problema pri rekonstrukciji slike je i količina detektora koji se nalaze na strojevima. Što je detektora više, to je detekcija, a time i sam broj uzoraka slike veći i rekonstrukcija je uspješnija. No nabava pojedinog detektora ili njihovog potrebnog skupa novčano je zahtjevniji pothvat, pa se traži alternativa na rekonstrukcijskoj strani. Pretpostavka je ovog rada da je tome moguće doskočiti uporabom compressive sensing metode u rekonstrukciji slike. Stoga će ta pretpostavka biti razrađena na primjeru PET skenera i podataka dobivenih njegovom upotrebom.

2. Teorijska podloga

2.1. Pozitronska emisijska tomografija

Pozitronska emisijska tomografija jedna je od metoda koje se koriste u nuklearnoj medicini za neinvazivni pregled stanja unutarnjih organa. Zasniva se na promatranju zračenja koje dolazi iz tijela pacijenta i određivanja gdje se radioaktivni materijal nakuplja nakon što je unesen u tijelo. Naime, neke vrste tkiva više iskorištavaju supstance koje se koriste za provedbu PET skeniranja, što se na samoj slici nastaloj skeniranjem očituje vidljivim promjenama na mjestu gdje se dotično tkivo nalazi u vidu izraženijeg zračenja.

O kojem se tkivu radi ovisi o tome koja se radioaktivna supstanca koristi jer će veće količine supstance koristiti ona tkiva koja taj izotop u njegovoj neizotopnoj varijanti koriste na svakodnevnoj bazi. Primjerice, supstanca koja se bazira na šećeru, a koji je temeljni izvor energije za rad mozga, biti će po ubrizgavanju u pacijenta vidljiva na čitavom području mozga, te se stoga ne preporuča za dijagnosticiranje problema s mozgom.

2.1.1. Priprema subjekta

Proces skeniranja započinje i prije no što subjekt uđe u uređaj jer metoda zahtijeva da se u subjekt ubrizga radioaktivna supstanca željenih performansi, takozvani **radiofarmak**.

Postoji nekoliko vrsti radiofarmaka, primjerice:

Flordeoksiglukoza (FDG) koja se koristi kod tkiva koja troše mnogo energije (glukoze), kao što su tumori, upalni procesi, epileptogena žarišta koja nastaju između dvaju napada epilepsije i slično. Ovaj radiofarmak također se može koristiti za detekciju perforiranih tkiva, te dijelova srčanog mišića za koji postoji mogućnost oporavka.

F-fluorid koji se koristi za detekciju metaboličkih promjena u kostima. Za razliku od FDG koja otkriva oblik i ponašanje samog tumora u kostima, F-fluorid pokazuje

reakciju kosti na tumor ili terapiju protiv tumora, čime se može pratiti napredak terapije nad tumorom.

F-flortimidin (FLT) i C-timidin koji se koriste kao indikatori sinteze DNA. Naime, nastanak tumora za sobom povlači i pojačane sinteze DNA i RNA, što omogućuje praćenje rasta tumora. Ovi se radiofarmaci najčešće koriste kod dijagnostike tumora na mozgu.

Flor-estradiol (FES) koji se koristi za prikaz estrogenskih receptora, što pomaže u dijagnostici tumora dojke.

Nakon što se subjektu ubrizga radiofarmak započinje proces snimanja. Snimanju mogu prethoditi i dodatne pripreme subjekta, nakon čega ga se postavlja u ležeći položaj na pokretni krevet unutar uređaja i pokreće snimanje. Dodatne pripreme ovise o ponašanju radiofarmaka, jer učinkovitost detekcije radiofarmaka ovisi o tome koliko je vremena prošlo od njegova ubrizgavanja. Ukoliko se pregled ne obavi dovoljno brzo, točnije prije no što se radiofarmak raspadne, proces se mora započeti ispočetka, što stvara probleme jer se može dogoditi da je za dotičnu vrstu radiofarmaka koji se upotrebljava već zadovoljen maksimalan dopušten unos u tijelo subjekta.

2.1.2. Princip rada

Snimanje se odvija na način da se radiofarmak u tijelu počinje raspadati, čime dolazi do pojave takozvanog **anihilacijskog zračenja**. Anihilacijsko zračenje (slika 1) nastaje kao posljedica β -plus raspada. β -plus ili pozitronski raspad nastaje kao dio prirodnog procesa raspada radioaktivnog materijala na proton koji se pretvara u neutron, te pozitron (pozitivna β -čestica) i neutrino, koji dijele energiju raspada. Nakon toga pozitron usporava prolaskom kroz materiju i pritom gubi kinetičku energiju. Na kraju pozitron praktički dođe do suprotnog mu elektrona, pri čemu se oni ponište (anihiliraju) i umjesto njih nastaje zračenje u obliku energijskog ekvivalenta njihovih masa mirovanja.



Slika 1. Proces anihilacije prilikom raspada radiofarmaka (Galera, 2013.)

To zračenje u obliku je dva fotona jednake energije koja iznosi 511 keV, koji odlaze u suprotnim smjerovima. Upravo detekcija ovih fotona na kristalima prstena PET skenera čini osnovu PET-a, točnije omogućava prikaz položaja pozitronskih emitera u tijelu. Naime, PET se zasniva na detekciji dotičnih fotona unutar određenog, što minimalnijeg, vremenskog prozora na suprotnim kristalima unutar prstena. Taj se događaj onda proziva detekcijom, a linija na kojoj se dogodio taj događaj linijom odgovora (engl. *Line of response - LOR*). Detekcija se u određenom vremenskom okviru događa na više no jednom paru kristala, što znači da se dobiva i više no jedna linija odgovora. Presjek svih linija odgovora unutar vremenskog okvira snimanja daje informaciju o tome na kojoj se poziciji u prostoru dogodio događaj.

Tijekom snimanja vrši se praćenje svih ovih događaja, te se zapisuju podaci o njima koji su kasnije potrebni za pretvorbu podataka u ljudima interpretabilan oblik. Postoje dva načina zapisa podataka: List Mode Format i sinogram. Daljnji opis ovih metoda slijedi u nastavku.

2.2. List Mode Format

List Mode Format (u daljnjem tekstu LMF) naziv je jednog od trenutno upotrebljavanih formata za spremanje podataka PET skeniranja, a koji zbog svoje jednostavnosti sve više istiskuje drugi od upotrebljavanih formata, zapis u obliku sinograma.

Svaki dokument zapisan u LMF formatu u stvarnosti se sastoji od dvije fizičke datoteke sa ekstenzijama .cch i .ccs.

Dokument sa .cch ekstenzijom predstavlja datoteku sa zaglavljem i sadrži podatke vezane uz sam proces snimanja te uz uređaj koji je korišten za snimanje. Ti podaci govore o izgledu, to jest geometriji uređaja kojim se vršilo snimanje (slika 2). Tako je iz zaglavlja moguće iščitati dimenzije, a time i broj kristala po sektorima i modulima u prstenima uređaja. Više o geometriji biti će rečeno u poglavlju 2.2.3. Dotično zaglavlje je u ASCII formatu i proizvoljne veličine.

| 😑 Nema | a231014-1min.cch 🗵 |
|--------|------------------------------------|
| 1 | geometrical design type: 1 |
| 2 | ring diameter: 13.56 cm |
| 3 | rsector axial pitch: 0 cm |
| 4 | rsector azimuthal pitch: 18 degree |
| 5 | rsector tangential size: 1.91 cm |
| 6 | rsector axial size: 11.11 cm |
| 7 | module axial size: 1.91 cm |
| 8 | module tangential size: 1.91 cm |
| 9 | module axial pitch: 2.76 cm |
| 10 | module tangential pitch: 0 cm |
| 11 | submodule axial size: 1.91 cm |
| 12 | submodule tangential size: 1.91 cm |
| 13 | submodule axial pitch: 0.0 mm |
| 14 | submodule tangential pitch: 0 cm |
| 15 | crystal axial size: 0.2 cm |
| 16 | crystal tangential size: 0.2 cm |
| 17 | crystal radial size: 2.0 cm |
| 18 | crystal axial pitch: 0.23 cm |
| 19 | crystal tangential pitch: 0.23 cm |
| 20 | layer0 radial size: 1.0 cm |
| 21 | in layer0 interaction length: 5 mm |
| 22 | layer1 radial size: 1.0 cm |
| 23 | in layer1 interaction length: 5 mm |
| 24 | clock time step: 390 pg |
| 25 | azimuthal step: -0.1 degree |
| 26 | z shift sector 0 mod 2: 9.2 mm |
| 27 | axial step: 0 cm |
| 28 | scan file name: Nema231014-1min |

Slika 2. Prikaz sadržaja .cch datoteke

Datoteka sa .ccs ekstenzijom predstavlja binarnu datoteku koja se sastoji od dva dijela. Prvi dio jest binarno zaglavlje u kojem je opisan format u kojem su strukturirani zapisi koji slijede nakon zaglavlja. Format ovisi o vrsti zapisa koji slijede, a koji mogu biti zapis događaja (engl. *event record*), zapis pobrojavanja (engl. *count rate record*) ili simulacijski zapis (engl *simulation record*). Također, samo zaglavlje ovisi i o tome da li se radi o zapisima pojedinih događaja ili zapisima koincidencija. Sukladno tome, binarno je zaglavlje .ccs datoteke proizvoljne duljine. Nakon zaglavlja slijede sami zapisi, po jedan u retku u formi sukladnoj onoj koja je opisana u zaglavlju .ccs datoteke.

U ovom radu analizirani su podaci dobiveni zapisima događaja, te pojašnjenje njihovih formi slijedi u nastavku.

2.2.1. LMF alat

Paralelno sa razvitkom LMF formata, razvijen je i LMF alat kojim je moguće izraditi LMF datoteku, manevrirati njome, te ju pretvarati u ljudima čitljiv ASCII format. LMF alat djelo je znanstvenika sa *ClearPET Project*-a, te su pri razvitku dotičnog projekta izrađeni i primjeri za rad sa alatom. Korisniku su za učenje rada s alatom dostupne testne datoteke, o čemu je više moguće doznati u (Krieguer, 2005.).

2.2.2. Format zapisa događaja

Pri izgradnji ASCII zapisa LMF datoteke dobiveni su zapisi događaja koji se, kao što je već rečeno, dijele na zapis pojedinih događaja i zapis koincidencija.

Svaki zapis pojedinih događaja sastoji se od slijedećih dijelova (slika 3):

- Vrijeme kada se dogodio događaj u piko sekundama. Događaji se promatraju od trenutka kada je započeto snimanje i poredani su uzlazno s obzirom na vrijeme.
- 2. Adresa kristala koji je pogođen u heksadekadskom zapisu.
- 3. Adresa kristala koji je pogođen u ljudski čitljivom zapisu, na način da su prikazani indeksi dijelova prstena prema kojima je moguće locirati kristal.
- 4. 3D-koordinate kristala na kojem se dogodio događaj.
- 5. Energija događaja.

 Energija događaja pomnožena sa zadanim korakom step. Korak je zapisan u zaglavlju datoteke.

| 1 | Ti: | 125151149290 | 7878 | Ad: | 30 | 0 | 0 | 0 | 60 | <u>Pn</u> : | -27.77 | -127.06 | 16.80 | Ey: | 167 | 501 | keV |
|---|-----|---------------|------|-----|----|---|---|---|----|-------------|---------|---------|-------|-----|-----|-----|-----|
| 2 | Ti: | 125151149310 | 277a | Ad: | 9 | 3 | 0 | 0 | 61 | Rn: | 25.43 | 127.59 | 74.40 | Ey: | 166 | 498 | keV |
| 3 | Ti: | 807418741287 | 5d7c | Ad: | 23 | 1 | 0 | 0 | 62 | Rn: | -122.50 | -44.07 | 36.00 | Ey: | 182 | 546 | keV |
| 4 | Ti: | 1188828750328 | 2642 | Ad: | 9 | 2 | 0 | 0 | 33 | Rn: | 34.79 | 125.45 | 48.00 | Ey: | 188 | 564 | keV |
| 5 | Ti: | 3256506946170 | 8b42 | Ad: | 34 | 3 | 0 | 0 | 33 | <u>Pn</u> : | 41.93 | -123.25 | 67.20 | Ey: | 203 | 609 | keV |

Slika 3. Primjer zapisa pojedinih događaja

S druge strane, zapis koincidencija ima slijedeću strukturu (slika 4):

- Vrijeme kada se dogodio događaj, u piko sekundama. Događaji se promatraju od trenutka kada je započeto snimanje i poredani su uzlazno s obzirom na vrijeme.
- 2. Adresa prvog od dva kristala koji je pogođen, u heksadekadskom zapisu.
- Adresa drugog od dva kristala koji je pogođen, u heksadekadskom zapisu.
- 4. Adresa prvog kristala u obliku 3D koordinata unutar sustava promatranja.
- 5. Energija događaja.
- Energija događaja pomnožena sa zadanim korakom step. Korak je zapisan u zaglavlju datoteke.

| 1 | 0 | 0.0 | 253 | c70 999 | 58.97 5 | 58.41 3 | 39.10E 0 0 | keV |
|---|---|-----|---------|---------|-----------|----------|------------|---------|
| 2 | 0 | 0.0 | c3e | 6212 | 999 21.40 | 70.05 | 62.10E | 0 0 keV |
| 3 | 0 | 0.0 | a17 | 5c6c | 999 33.81 | 1 75.59 | 29.90E | 0 0 keV |
| 4 | 0 | 0.0 | 5a1c | 6208 | 999 -23.5 | 51 -69.1 | 4 29.90E | 0 0 keV |
| 5 | 0 | 0.0 | 29 5a18 | 3 999 | 53.78 (| 52.96 | 4.60E 0 0 | keV |

Slika 4. Primjer zapisa koincidencija

2.2.3. Geometrija adresiranja kristala

Kako bi adresiranje kristala bilo jednoznačno bez obzira na konstrukciju PET skenera kojim se izvodi mjerenje, pri definiciji LMF formata dogovorena je konvencija po kojoj se adresiraju pojedini dijelovi uređaja. Konvencija će u nastavku biti objašnjena na primjeru geometrije skenera iz *ClearPET* projekta (Krieguer, 2002.).

Svaki se PET uređaj sastoji od jednog ili više prstena, pri čemu se prsteni indeksiraju počevši od indeksa nula, te indeks raste u smjeru aksijalne osi $\vec{e_z}$ (slika 5). Prstene se dalje promatra kao linije dobivene na način da se prstene 'prereže' i razmota u 2D prostoru.



Slika 5. Podjela PET skenera na prstene (Krieguer, 2002.)

Svaki se prsten dijeli na jednak broj sektora, a tvorba prstenova i sektora tvori pojam *rsectors* koji u ovoj geometriji označavaju geometrijski najveće jedinice uređaja. Indeksiranje *rsector-*a također počinje od nula, a nultim se *rsector-*om smatra onaj koji se nalazi na nultom prstenu na onom mjestu gdje pozitivna strana x-osi siječe prsten (to jest sektor). Indeksi *rsector-*a rastu najprije tangencijalno na prsten u smjeru kazaljke na satu, a zatim u smjeru aksijalne osi (slika 6). Pritom je potrebno napomenuti da se koordinatni sustav PET skenera nešto razlikuje od Kartezijevog koordinatnog sustava. Naime, u Kartezijevom je sustavu x-os horizontalna, a y-os vertikalna. U sustavu PET skenera osi su zamijenjene, pa tako x-os leži vertikalno, a y-os horizontalno.



Slika 6. Podjela PET skenera na sektore i module (Krieguer, 2002.)

Nadalje se sektori dijele na module, na način da se u ovoj izvedbi geometrije sektori (točnije *rsector*-i) tangencijalno dijele svaki na tri modula. Moguća je i drugačija podjela, no ona će uvijek unutar *rsector*-a činiti matričnu formu. Svaki modul potom se dijeli na podmodule, u ovom slučaju na četiri podmodula aksijalno (slika 7). Podmoduli se, kao i moduli, mogu podijeliti i na drugi način, ali također uvijek tvoreći matričnu formu unutar modula. Na kraju se svaki podmodul dijeli na kristale. U ovom primjeru kristali tvore matricu dimenzija 8x8 u jednom sloju. Naime, moguće je da kristali budu u više od jednog sloja, pri čemu indeksiranje slojeva započinje od onog sloja koji je najbliži centru prstena, te se on naziva nultim slojem.



Slika 7. Podjela PET skenera na module i podmodule (Krieguer, 2002.)

2.3. Sinogram

Za razliku od LMF formata koji je zbog svoje lagane mogućnosti zapisa i interpretacije sve popularniji kao način spremanja rezultata u novijim PET uređajima, sinogram je bio bazični način zapisivanja rezultata. Bazira se na korištenju Radonove transformacije kako bi se vizualno prikazale linije odgovora, nakon čega slijedi primjena neke od transformacija za pretvorbu podataka iz forme sinograma u oblik slike.

2.3.1. Generiranje sinograma

Sinogram je takav zapis linija odgovora u kojem se tvori matrica dimenzija $r \propto \varphi$, gdje su φ kutovi koji govore za koliko je linija odgovora rotirana od pozitivnog dijela osi x, a *r* udaljenosti linije odgovora od centra koordinatnog sustava. Koordinatni je sustav postavljen u centar promatranih prstenova i tumači se kao klasični Kartezijev koordinatni sustav, sa x osi postavljenom horizontalno i y osi postavljenom vertikalno. Za sve se linije računaju vrijednosti φ i *r*, te se na poziciju određenu tim vrijednostima ili upisuje nova vrijednost ili nadodaje vrijednost na postojeću. Dotične se vrijednosti mogu skalirati sa odabranim faktorom, što jedino utječe na tumačenje podataka pri rekonstrukciji slike.

Kada se takva matrica projicira, dobije se karakterističan sinusoidalni izgled. Sinusoide koje se vide na slici 8 proizlaze iz definicije Radonove transformacije i njena izračuna za točkasti izvor. Ona se za pojedini izvor definira kao

 $Rf(r,\varphi) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \delta(x-x_0) \cdot \delta(y-y_0) \cdot \delta(r-x \cdot \cos(\varphi) - y \cdot \sin(\varphi)) dx dy$ (1)

što za točkasti izvor rezultira sa

$$Rf(r,\varphi) = \delta(r - x_0 \cdot \cos(\varphi) - y_0 \cdot \sin(\varphi))$$
(2)

odnosno sinusoidom. Shodno svojstvima sinusoida, kada se transformacija primjeni na ostale točke tijela, sinusoide se zbrajaju i rezultiraju u sinogramu.



Slika 8. Sinogram, preuzeto iz (Galera, 2013.)

2.3.2. Pretvorba LMF formata u sinogram

Kako i LMF i sinogram predstavljaju iste podatke, sasvim je razumljivo da je moguća i transformacija iz jednog u drugi oblik. Transformacija se zasniva na određivanju linija odgovora i njihovu smještanju u pripadni sinogram.

Za određivanje linija odgovora potrebno je pronaći zapise unutar LMF datoteke koji se mogu smatrati koincidencijama, te koristeći njihove položaje dobivene iz geometrije prstenova i kristala, izračunati jednadžbu linije odgovora. Koincidencijama se smatraju oni zapisi koji su se dogodili u dovoljno malom vremenskom okviru, primjerice desetak nano sekundi.

Po određivanju koincidencija, potrebno ih je podijeliti u ravnine u kojima su se dogodile. Naime, PET snimanje proizvodi više od jednog sinograma jer je dozvoljeno postojanje linija odgovora između bilo koja dva kristala (slika 9). Točnije, jedno snimanje proizvede 2n - 1 sinogram, pri čemu je n broj promatranih ravnina. One se određuju dijeljenjem aksijalne širine prstena sa aksijalnom širinom kristala. Sinograma je točno 2n - 1 jer je n ravnina okomitih na kristale, te n - 1 ravnina smještenih okomito na pola između svaka kristala.

Liniju se odgovora sprema u dotičnu ravninu na način da se pogledaju krajnje z koordinate točaka koje određuju liniju, te nađe njihova sredina. Ukoliko izračunata sredina pada na cijeli broj, liniju se proglašava dijelom ravnine okomite na kristale čiji je indeks dotična sredina. Ukoliko sredina pak pada na polovicu između dva cijela broja, liniju se proglašava dijelom ravnine između kristala čiji je indeks dotična sredina. Kada se linije odgovora podijele u ravnine, u daljnjem se izračunu promatraju njihove projekcije u ravninu određenu x i y koordinatnim osima. Ovakav se sinogram naziva 3D sinogramom.



Slika 9. Dozvoljene su sve linije odgovora (Badawi, 1999.)

Nakon sortiranja koincidencija za svaku se ravninu iz zapisa u LMF formatu čitaju x i y koordinate kristala (vidjeti poglavlje 2.2.2) za svaki od zapisa u paru. Pomoću tih se koordinata računa linija odziva jednostavnim matematičkim postupkom računanja pravca kroz dvije točke u prostoru:

$$y - y_1 = \frac{y_2 - y_1}{x_2 - x_1} (x - x_1) \tag{3}$$

gdje su $T_1(x_1, y_1)$ i $T_2(x_2, y_2)$ dotične koordinate.

Jednom kada je poznata jednadžba linije odgovora, potrebno je odrediti kut φ i udaljenost *r*. To se može napraviti na dva načina: vektorski ili analitički. U ovom je radu odlučeno da će se zbog svoje jednostavnosti proučavati analitički način. Kod

analitičkog načina sve što je potrebno jest razmotati geometrijske odnose između linije odgovora i koordinatnog sustava.

Neka se linija odgovora nalazi na poziciji kao na slici (slika 10). Tada je potrebno izračunati vrijednosti r_0 koja predstavlja udaljenost r i φ .



Slika 10. LOR geometrija (Galera, 2013.)

Za r_0 vrijedi odnos:

$$r_0 = r_a \cdot \sin(\alpha) \tag{4}$$

a za φ vrijedi:

$$\varphi = tan^{-1}(\frac{y_1 - y_2}{x_1 - x_2}) \tag{5}$$

Za računanje kuta φ svi podaci su već poznati, dok za računanje udaljenosti treba još pronaći iznos za r_a i α . Udaljenost r_a moguće je izračunati koristeći Pitagorin poučak, pa slijedi:

$$r_a = \sqrt{x_1^2 + y_1^2}$$
(6)

Da bi se mogao izračunati kut α postavlja se slijedeća jednadžba:

$$\alpha + \beta + \varphi + 90 = 180 \tag{7}$$

iz čega proizlazi da je

$$\alpha = 90 - \beta - \varphi \tag{8}$$

 β se dobiva iz odnosa

$$\beta = \tan^{-1}(\frac{x_1}{y_1}) \tag{9}$$

a odnos za ϕ je već izračunat u izrazu 5 , pa ga se zajedno sa izrazom 9 samo uvrsti u izraz za 8. Za α se onda dobije:

$$\alpha = 90 - \tan^{-1}\left(\frac{x_1}{y_1}\right) - \tan^{-1}\left(\frac{y_1 - y_2}{x_1 - x_2}\right) \tag{10}$$

Uvrštavanjem jednadžbi 6 i 10 u jednadžbu 4, dobije se :

$$r_0 = \sqrt{x_1^2 + y_1^2} \cdot \cos(\tan^{-1}\left(\frac{x_1}{y_1}\right) + \tan^{-1}\left(\frac{y_1 - y_2}{x_1 - x_2}\right)) \tag{11}$$

čime se izračuna udaljenost linije odgovora od centra koordinatnog sustava. Tako izračunati kut φ i udaljenost r mogu se spremiti u matricu sinograma.

2.3.3. Pretvorba sinograma u sliku

Kako bi se dobiveni izlaz iz PET uređaja mogao interpretirati, potrebno ga je transformirati u oblik lakši za shvaćanje, a to podrazumijeva slikovni oblik. Tijekom godina razvijeno je nekoliko metoda rekonstrukcije, od kojih se najviše koriste analitička metoda poznata pod skraćenicom FBP (engl. Filtered BackProjection), te iterativne metode ART (engl. Algebraic Reconstruction Technique) i MLEM (engl. Maximum Likelihood Expectation Maximization).

FBP METODA

FBP metoda jedna je od prvih metoda razvijenih za namjenu rekonstrukcije PET slike. To je poboljšani način provedbe BP (engl. Back Projection) metode, pa ju i u potpunosti sadrži.

BP metoda začeta je u dobu nakon izuma prvog stroja za snimanje X-zraka, kada je pojam standardnog računala podrazumijevao daleko slabije računalo od onog koje se danas standardno koristi. Iz tih je razloga za računalnu obradu bilo važno algoritam što više svesti na računalne operacije spremanja u memoriju i zbrajanja, koje su koristile najmanje procesorskog vremena i resursa. Ideja koja je proizašla iz dotičnih uvjeta bila je veoma jednostavna - napraviti upravo obratan postupak od onog kojim je dobiven sinogram. Postupak je, naime, slijedeći: za svaku točku u matrici [r, ϕ] izračuna se linija određena kutom ϕ i udaljenošću od centra koordinatnog sustava r. Linija se zatim nacrta u sustavu, te se prelazi na slijedeću točku. Svaka se slijedeća točka matrice ponovno pretvara u liniju određenu koordinatama svoje pozicije, te ucrta u isti koordinatni sustav kao i prethodna. Preklapanjem ucrtanih linija dobiva se slika, kao što je vidljivo na slici 11.



Slika 11. Iteracije BP metode za točkasti izvor (Badawi, 1999.)

No, kao što je vidljivo nakon zadnje iteracije, dotična metoda ima problema sa preklapanjima linija odgovora. Naime, dolazi do svojevrsnog razmazivanja slike oko izvora. Razlog tome jest što se kod zbrajanja linija događa da se mnoge linije preklapaju jedna preko druge i time povećavaju vrijednosti koeficijenata na kojima se dogodio događaj, ali i nekoliko okolnih. Kako bi se to izbjeglo, potrebno je obaviti takvu filtraciju koja bi sačuvala vrijednost na mjestu presjeka, a otklonila sve smetnje nastale oko toga područja.

To se izvodi koristeći takozvani *ramp* filtar, koji je reprezentiran apsolutnom vrijednošću funkcije u vremenskoj, odnosno valićnim filtrom u frekvencijskoj domeni (slika 12).



Slika 12. Prikaz ramp filtra (Germano, 2001)

Kada se takav filtar primjeni na svaku od zraka pri iscrtavanju, dobije se rezultat prikazan na slici 13.



Slika 13. Iteracije FBP metode (Platten, 2003.)

ART metoda

Kod ART metode pretpostavlja se da se presjeci sastoje od nizova nepoznanica, pa se problem rekonstrukcije može postaviti kao sustav linearnih nejednadžbi. Kako bi se takav sustav riješio, potrebno je odabrati kolekciju baznih funkcija. Najčešće se za takvu kolekciju bazne funkcije određuju koristeći piksele kao funkcije pripadanja koordinata određenom pikselu.

Rekonstrukcija se zatim odvija iterativno, pri čemu se za dobivanje skupa rješenja koristi slijedeća formula:

$$x^{k+1} = x^k + \lambda_k \cdot \frac{b_i - \langle a_i, x^k \rangle}{||a_i||^2}$$
(12)

U jednadžbi su a_i redovi matrice A koja sadrži sliku, a b_i članovi vektora kutnih projekcija.

MLEM metoda

MLEM algoritam procjenjuje maksimalnu vjerojatnost postojanja podataka koji fale ili nisu prisutni. MLEM algoritam može pronaći vezu između novih parametara i prave informacije. To znači da će nakon svake iteracije biti izračunat dio podataka vezan uz neki dio slike.

Svaka iteracija odvija se u dva koraka. Prvi je korak procijeniti skrivene parametre koristeći opservirane i prave podatke. Nakon toga primjenjuje se algoritam maksimalne vjerojatnosti. Vjerojatnost će porasti nakon svake iteracije, što znači da će ujedno i konvergirati.

Kako bi se postigla što bolja kvaliteta, potrebno je povećati broj iteracija, što se vidi iz slike 14.



Inicijalno rješenje







7. iteracija

Slika 14. Iteracije MLEM metode (Galera, 2013.)

2.4. Compressive sensing

Compressive sensing metoda nastala je kao odgovor na svakodnevni problem koji se događa pri rekonstrukciji signala, a to je činjenica da će rekonstrukcija biti moguća jedino ukoliko je zadovoljen Shannon-Nyquistov teorem. Taj teorem kaže da je uvjet uspješne rekonstrukcije uzorkovanje signala frekvencijom barem dva puta većom od najveće frekvencije u signalu (Gogić, 2015.).

No, to često podrazumijeva veliku količinu podataka, pa se u današnje vrijeme kada su podaci svuda oko nas i kada dotični signali postaju sve veći i kompleksniji za obradu, postavlja pitanje postoji li metoda koja bi koristila manje podataka. Tu je na scenu stupio compressive sensing.

2.4.1. Podaci

Compressive sensing metoda traži da ulazni podaci budu u formi jednodimenzionalnog vremenski diskretnog signala x od N elemenata x[n] poredanih u vektor stupac, te da se svi višedimenzionalni signali vektoriziraju u taj oblik. Nakon rekonstrukcije, višedimenzionalni se signali vraćaju u prvotni oblik.

Kao i kod svih metoda vezanih za vremensko-frekvencijsku domenu, i ovdje je definiran odnos između podataka signala u obje domene. Tako vrijedi da, ako signal x promatramo u frekvencijskoj domeni, dobivamo signal s, s kojim je x u slijedećem odnosu:

$$x = \sum_{i=1}^{N} s_i \psi_i \tag{13}$$

gdje je ψ matrica inverzne transformacije.

Ono što čini osnovu compressive sensing metode jest činjenica da ulazni podaci moraju biti *K*-rijetki. Za signal se kaže da je *K*-rijedak ukoliko je samo *K* ($K \ll N$) njegovih članova različitih od 0, a svi ostali su 0. Na takvim je signalima ujedno moguće i napraviti dobru kompresiju jer je samo nekolicina važnih podataka za sačuvati prisutna u signalu (Gogić, 2015.). Ovakav je signal sada moguće povoljnije iskoristiti jer se neće obrađivati svih *N* članova, već samo njih *K*.

2.4.2. Metodologija

Kao što je rečeno, za procesiranje u compressive sensing metodi koristiti će se samo dio ulaznih podataka. Da bi se odredilo koje podatke analizirati koristi se matrica ψ , koja sadrži *M* uzoraka (*M* < *N*), te je dimenzija *Mx*1. Ta se matrica naziva mjernom ili rijetkom matricom, te se njome dobiva vektor kojeg tvorac metode naziva vektorom *y*. On je definiran sa:

$$y = \phi x = \phi \psi s = \theta s \tag{14}$$

Promatrajući definiciju *y*-ona, evidentno je da matrica θ ima dimenzije $M \times N$, budući da je definirana kao umnožak matrice inverzne transformacije i mjerne matrice.

Mjerna je matrica neovisna o ulaznom signalu x, te se ne mijenja tijekom postupka. Stoga je prvi i ključan korak ovog algoritma definiranje takve mjerne matrice kojom neće biti nepovratno izgubljeni podaci. Jedino ako je taj korak izveden pravilno, moguće je raditi rekonstrukciju signala x iz prorijeđenog vektora y.



Slika 15. Prikaz algoritma Compressive sensing-a (Baraniuk, 2007)

Mjerna matrica

Kako bi se odabrala adekvatna mjerna matrica potrebno je zadovoljiti dva uvjeta: mjerna matrica mora biti ortonormirana, te mjerna i transformacijska matrica moraju biti nekoherentne. Iz dotičnih svojstava, te rijetkosti matrice dolazi se do toga da se računa samo (100 - p)% uzoraka signala, jer mjerna matrica odbacuje p% redaka.

Postoji nekoliko vrsti matrica koje odgovaraju ulozi mjerne matrice, od kojih su prema (Gogić, 2015.) najuspješnije :

- Nasumična matrica koja se sastoji od nasumičnih brojeva. Njena uspješnost leži u tome što je veoma mala vjerojatnost da će ovako izgrađena matrica biti koherentna sa transformacijskom matricom. Naravno, ova će matrica zadovoljavati zahtjeve ukoliko je ujedno i ortogonalna.
- 2. Diskretna valićna transformacija koja predstavlja diskretizaciju kontinuirane valićne transformacije i, kao i ona, ima svojstvo istovremenog prikaza signala u vremenu i frekvenciji. Njeno je veoma povoljno svojstvo to što u valićnoj domeni tvori rijetke signale. Postoji više vrsta diskretnih valićnih transformacija, kao što su DWT i SWT.

Transformacijska matrica

Transformacijska matrica ne odabire se na temelju definiranih pravila kao mjerna, već ona ovisi o primjeni algoritma. Tako treba uzeti u obzir cilj primjene matrice kako bi se efikasno odredilo što bi ona trebala predstavljati.

2.4.3. Rekonstrukcija

Daljnji postupak algoritma zahtijeva rekonstrukciju, što znači da od vektora y treba naći sva rješenja s' koja zadovoljavaju jednadžbu

$$\theta \cdot s' = y \tag{15}$$

što i nije tako lagan zadatak uzevši u obzir da je u vektoru y sadržano mnogo manje podataka nego u početnom signalu, pa ova jednadžba daje beskonačno mnogo s' koji ju zadovoljavaju. Cilj je rekonstrukcije naći upravo onaj s' koji najmanje odstupa od početnog signala x.

Da bi se to izvelo potrebno je napraviti optimizaciju algoritma pronalaska, što se može postići koristeći nekoliko normi. U ovom je radu odabrana minimizacija preko l_1 norme iz razloga što se minimizacija po l_2 normi ne pokazuje dobrom jer ne radi dobro sa rijetkim signalima. Također, l_0 norma dolazi u obzir kao dobar odgovor, no ona je računalno previše složena za izvedbu, pa nije razmatrana.

Dotična obrazloženja dovode do zaključka da će l_1 norma polučiti zadovoljavajuće rezultate. l_1 norma dana je slijedećim odnosom:

$$\hat{s} = argmin ||s'||_0 takav \, da \, je \, \theta s' = y \tag{16}$$

3. Izvedba rekonstrukcije Compressive sensing metodom

Provedba ideje opisane u prethodnom poglavlju izvedena je u nekoliko većih koraka, a to su:

- 1. Interpretacija ulaznih podataka u ovisnosti o tipu zapisa unutar LMF datoteke. Ulazni se podaci pretaču u oblik linija odgovora.
- Iščitavanje podataka iz linija odgovora radi konstrukcije tražene slike u compressive sensing metodi.
- 3. Izrada i provedba compressive sensing metode.

3.1. Ulazni podaci i interpretacija

Kao što je navedeno u prethodnim poglavljima, LMF format već je naveliko istisnuo sinogram. Tako su i podaci s kojima je manipulirano u ovom radu iščitani iz LMF formata.

Podaci su iščitavani u dvije verzije, ovisno o tome sadrži li LMF datoteka podatke o pojedinačnim događajima (takozvanim *singles eventima*) ili podatke o koincidencijama. U nastavku slijede opisi interpretacija LMF datoteka u ovisnosti o dotičnim tipovima podataka.

3.1.1. LMF datoteka sa pojedinačnim događajima

LMF datoteka sa pojedinačnim događajima sastoji se od niza redova od kojih svaki predstavlja jednu anihilaciju detektiranu na nekom od kristala. Na popisu su sve anihilacije detektirane prilikom snimanja, u vremenski sortiranom poretku.

Sadržaj svakog reda već je pojašnjen u poglavlju 2.2.2, te ovdje neće biti ponovno naveden. Iz dotičnog sadržaja za konstrukciju linija odgovora potrebni su podaci o vremenu kada se događaj dogodio, te 3D koordinate kristala na kojem se događaj dogodio.

Vrijeme je nužno kako bi se pomoću zadanog vremenskog okvira odredilo koja dva susjedna zapisa o događaju potencijalno čine događaj koincidencije. Naime,

da bi se anihilacija detektirala, potrebno je da se unutar veoma malenog vremenskog okvira dogode dvije detekcije na dvama različitim kristalima. Nije nužno da ti kristali budu nasuprotni jedan drugome, nužno je samo da se linija odziva dobivena pomoću njihovih koordinata prostire unutar prstena na razumnoj udaljenosti od samog ruba prstena.

Uvjetom o razumnoj udaljenosti se postiže odbacivanje situacija kada su primjerice dva susjedna kristala gotovo istovremeno detektirala zračenje, te ih se proglasi parom koji čini liniju odziva. Takva situacija fizički nije moguća zbog konstrukcije PET skenera.

Nakon određivanja potencijalnih parova, jednostavnom primjenom osnova linearne algebre može se izračunati jednadžba linije odziva.

3.1.2. LMF datoteka sa koincidencijama

Za razliku od LMF datoteke sa pojedinačnim događajima, redak u LMF datoteci sa koincidencijama označava već interpretiran događaj anihilacije. Interpretaciju obavlja program unutar *ClearPET* LMF alata, te je kod interpretacije ove datoteke moguće preskočiti dio sa vremenskim prozorom.

Pa ipak, interpretacija iz ovog tipa datoteke nije sasvim jednostavna. Koristeći oblik zapisa kakav je trenutno, nad svakim je retkom potrebno izvršiti niz operacija kako bi se dobile 3D koordinate drugog kristala. Naime, kao što je vidljivo u poglavlju 2.2.2, u zapisu retka LMF datoteke sa koincidencijama samo su 3D koordinate prvog od dva kristala nad kojima je detektiran događaj. Koraci za pronalazak koordinata drugog kristala bili bi slijedeći:

- Interpretacijom .cch datoteke vezane za LMF datoteku koincidencija iščitavaju se podaci o geometriji PET skenera - broj, dimenzije i raspored *rsector*-a, modula, podmodula i kristala.
- Određuje se pozicija u trenutku nula za nulti kristal nultog podmodula, u nultom modulu i nultom rsektoru. Trenutkom nula smatra se stanje mirovanja upravo prije no što snimanje kreće.
- Za svaki se zapis korištenjem 3D koordinata prvog kristala računa kut otklona koji je napravio uređaj.

- 4. Za svaki se zapis pomoću heksadekadskog zapisa koordinata drugog kristala i interpretirane geometrije iz prvog koraka, određuje se i nulta pozicija drugog kristala u trenutku nula.
- Izračunati kut otklona nadodaje se na maloprije izračunatu nultu poziciju drugog kristala, te se dobivaju 3D koordinate drugog kristala.

Primjenom ovog postupka dobivaju se potrebne 3D koordinate, te je moguće izračunati potrebnu liniju odziva.

No, zahvaljujući činjenici da je *ClearPET*-ov LMF alat otvorenog koda, moguće je ovome problemu doskočiti na drugačiji način. Jednostavnom promjenom onoga što se ispisuje kao 3D koordinate moguće je umjesto 3D koordinata prvog kristala dobiti 3D koordinate drugog kristala. Tim postupkom izrađena je nova datoteka sa zapisima 3D koordinata drugog kristala, te su obje datoteke interpretirane paralelno.

Interpretacija se u ovoj situaciji bazirala na jednostavnom iščitavanju 3D koordinata prvog kristala iz prve, te 3D koordinata drugog kristala iz druge datoteke. Nakon toga podaci su spremljeni u dvije matrice, gdje svaki redak matrice predstavlja koordinate jednog od kristala koji je detektirao događaj, a u istim se recima nalaze koordinate kristala koji su detektirali istu liniju odgovora.

3.2. Kreiranje matrica za compressive sensing metodu

U poglavlju 2.4.2 bilo je govora o mjernoj i transformacijskoj matrici te o tome kako ih je potrebno pravilno odabrati kako bi algoritam compressive sensing-a bio pravilno proveden.

Za matricu mjerenja izabrana je DWT matrica čiji je valić Symlets 4 ili skraćeno sym4 (slika 16). Sym4 je gotovo simetričan, ortogonalan i biortogonalan valić čije je povoljno svojstvo to da je signal nakon primjene takvog valića, dakle u valićnoj domeni, izrazito rijedak.

Za transformacijsku matricu izabrana je matrica pretvorbe LMF formata direktno u oblik slike. Time se dobiva rijetka reprezentacija signala kojeg tražimo, točnije bolje rekonstrukcije slike koju se traži. No, izrada takve matrice bila bi vremenski i resursno previše zahtjevna, te se za potrebe ovog rada pribjeglo metodi prerade mjernih podataka u oblik slike, te izjednačavanju takvih podataka sa umnoškom matrice mjerenja i traženog signala s.



Slika 16. Symlet 4 valić (Wasilewski, 2008.)

Ovakva izvedba bila bi analogna provedbi slijedeće formule:

$$\phi^{-1} \cdot y = \psi \cdot s \tag{17}$$

s tom iznimkom da matrica φ nikada nije ni izrađena već je zamijenjena funkcijom koja vrši pretvorbu signala y-on, čime je ušteđeno na prostoru i vremenu.

3.3. Izrada i provedba compressive sensing metode

Alat kojime je moguće sprovesti compressive sensing metodu može se pronaći na poveznici navedenoj u (Grant, 2013.). No, prije poziva kôda potrebno je pripremiti matrice koje se koriste .

Kôd za izradu matrice DWT transformacije moguće je pronaći u dodatku, a zasniva se na izradi matrica DWT transformacije za transformaciju u oba smjera. Matrice se izrađuju provedbom transformacije na jediničnoj matrici i spremanja dotičnih transformacija u novu matricu. Množenjem tako nastale matrice sa signalom, dobiva se transformirani signal.

Kao i prethodni, kôd za izradu transformacijske matrice, odnosno modifikaciju mjerenja u matricu rezultantne slike moguće je naći u dodatku. Ovaj kôd sastoji se od nekoliko dijelova, a to su redom:

- Određivanje kojoj ravnini od mogućih devedeset i pet, točnije kojem sinogramu bi pripala koja pročitana linija koincidencije.
- Pretvaranje svih linija koincidencija u linije odgovora za svaki od sinograma. Naime, ako su na nekom sinogramu između neka dva kristala za isti kut postojale dvije i više linija koincidencije, njih se sve stapa u jednu. Time se izbjegava razmazivanje linija.
- Za svaku ravninu, odnosno potencijalni sinogram, radi se projekcija linija odgovora pribavljenih prijašnjim postupcima, te razmazivanje dotičnih linija. Razmazivanje se provodi gausovskim filtrom, koji omogućava da linije poprime kontinuiranu formu.
- Skup projekcija za jednu ravninu proglašava se mjerenom slikom. Naime, svaki se sinogram u 3D zapisu sinograma pretvara u jednu sliku, te PET snimanje proizvodi onoliko slika koliko je sinograma u njegovu zapisu.

Tako dobivene slike se pojedinačno predaju u kôd za compressive sensing metodu, te se pokušavaju dobiti bolje slike dotičnom metodom rekonstrukcije.

Metodi razmazivanja linija prišlo se i na drugi način, a taj je bio da se linije jednostavno sve projiciraju bez da se odbace one koje se preklapaju, te se nad tako projiciranim linijama napravi razmazivanje gausovskim filtrom.

Sve dobivene mjerene slike predalo se na korištenje programskom alatu CVX (Grant, 2013.), (Grant, 2008.). CVX je alat kojem se predaje varijabla koju se želi minimizirati, a koja može biti do u maksimalno dvije dimenzije. Predaje mu se i funkcija koju treba normalizirati, te mu se u tijelu napišu odnosi modela kojeg treba zadovoljiti. Nakon što su svi podaci pripremljeni, pokreće se sam alat.

Alat koristi predefinirane programe za rješavanje zadanog problema, bilo da je to minimizacija ili maksimizacija. Programi za rješavanje su dostupni skupa sa alatom, ili ih se može dodatno instalirati. Za potrebe ovog rada nisu bili potrebni dodatni programi za rješavanje, već je korišten predefinirani program SDPT3.

4. Rezultati

Kao što je objašnjeno u prethodnom poglavlju, izrada slike nad kojom se provodila compressive sensing metoda napravljena je u dvije verzije.

Zbog zahtjevnosti obrade i obujma podataka, rezultati neće biti prikazani za svih devedeset i pet ravnina koliko je sadržano u sinogramu, već samo za jedan. Odabrana je centralna ravnina direktnih ravnina, ravnina pod indeksom dvadeset i četiri. Prikaz rekonstrukcije jednostavnim nanošenjem linija u ravnini, bez redukcije preklapanja, prikazan je na slici 17.



Slika 17. Mjerna slika dobivena izravnim nanošenjem linija

Kako bi se iz slike što preciznije izvukle bitne značajke, odlučeno je da će na sliku biti primijenjen prag i to na dva načina:

1. Primjenom praga na mjerenu sliku te provedba compressive sensing metode na takvoj slici.

2. Primjena compressive sensing metode na postojećoj slici te primjena praga na dobivenim podacima, netom prije rekonstrukcije.



Primjenom prvog načina dobiva se rezultat prikazan na slici 18.

Slika 18. Rezultat uz primjenu prvog načina praga na prethodnu sliku

Iz dotične se slike vidi kako se glavnina informacija nalazi u centru slike, što je u skladu sa činjenicom da se promatrano tijelo nalazi u centru prstena PET skenera. Uvećavanjem ovih podataka dobiva se slika pojave koju se promatra u tijelu.

Primjenom praga nakon primjene compressive sensing metode ne dobivaju se suvisli rezultati, što otvara prostor za daljnji razvoj metode i otkrivanje zašto su rezultati nesuvisli. Na dotičnim rezultatima je evidentno da je došlo do nekakve greške kojom se vidno gube podaci. Stoga je za izvedbu ove metode u 3D verziji odabrana primjena praga prije no što se mjerena slika pošalje u alat za primjenu compressive sensing metode.

Pri izvođenju ovih metoda, razmatrane su mogućnosti provedbe valićne dekompozicije u što više razina, primjerice četiri. No, zbog ograničenja alata CVX uspješno je provedena dekompozicija do u maksimalno dvije razine.

Na slici 19 vidljiva je slika nastala nanošenjem linija sa redukcijom preklapanja koja prikazuje drugačiju ravninu nego što je ona na slici 17. Na tu je sliku primijenjena samo metoda primjene praga na mjerenu sliku pa provedba compressive sensing metode. Rezultati dobiveni ovim putem vidljivi su na slici 20.



Slika 19. Mjerna slika dobivena nanošenjem linija uz redukciju preklapanja



Slika 20. Rezultat uz primjenu prvog načina praga na prethodnu sliku

Pri izradi ovoga rada u planu je bila i demonstracija 3D verzije sinograma. U tu je svrhu razvijen kôd koji radi pripremu mjerenih podataka pomoću valićne transformacije u 3D prostoru, te kôd za provedbu compressive sensing metode nad takvim mjerenim podacima. Dotične je kodove moguće pronaći u dodatku.

No, s obzirom da se radi o veoma velikom broju podataka koji za obradu traže i procesorsku i memorijsku moć, ali i vrijeme, do trenutka kada je ovaj rad trebao

biti završen, kôd se nije uspio izvršiti do kraja. Uređaj na kojem se kod vrtio sastoji se od 32 Intel(R) Xeon(R) CPU E5-2640 v2 @ 2.00GHz procesora, sa 378GB memorije.

5. Zaključak

Promatranjem rezultata dobivenih primijenjenim metodama za zaključiti je da je pristup dobar, no da ga se treba poboljšati. Tijekom izrade rada naišlo se na nekoliko pojedinosti na koje bi se u nastavku trebalo obratiti pažnju.

Tim pojedinostima svakako pripada detaljnije promatranje izvedbe metode projiciranja linija preklapanjem. Naime, u toj je metodi potrebno obratiti pažnju na potencijalnu širinu linije, te tome prilagoditi i gausovsku funkciju kojom se linija razmazuje. Također, u postupku bi trebalo uzeti u obzir i međusobne odnose među ravninama, kao i činjenicu da linije indirektnih ravnina utječu i na direktne ravnine. Razlog tome jest da su pravci sadržani u indirektnim ravninama nakošeni u z dimenziji, pa probadaju i direktne ravnine.

Daljnje poboljšanje svakako je u provedbi compressive sensing metode. U ovom je radu korišten alat CVX, čija su ograničenja navedena ranije. Najviše problema stvaraju ograničenja vezana uz memorijsko zauzeće koja nastaju kada alat zauzme veću količinu memorije no što je alatu Matlab, koji izvodi kod alata CVX, dodijeljen. Uz ta ograničenja problem predstavlja i vrijeme izvedbe minimizacije, koje je poprilično dugotrajno ukoliko se radi o nekom većem broju podataka. Po rješenju svih ovih problema rezultati bi trebali biti višestruko bolji.

Vlastoručni potpis:

Ankica Gogić

6. Literatura

Citirana literatura:

[1] Moreno Galera, A. *Data processing toolbox for PET scenners*. Magistarski rad- Tampere University of technology, 2013.

[2] Krieguer, M., Simon, L., Strul, D., Morel, C. *List Mode Format Version 2.0: Getting started and using GATE with LMF.* Universite de Lusanne. 2005.

[3] Krieguer, M., Simon, L., Strul, D., Morel, C. *List Mode Format Implementation: scanner geometry description, Verision 4.1.* Universite de Lusanne. 2002.

[4] Badawi, R. Introduction to PET Physics. Datum izrade: 12. siječnja 1999.
http://depts.washington.edu/nucmed/IRL/pet_intro/toc.html. Datum pristupa:
20. svibnja 2015.

[5] Germano, G. Technical aspects of myocardial SPECT imaging. *The Journal of Nuclear Medicine* 42, 10 (2001), 1499-1507.

[6] Platten, D. *Multi-slice helical CT physics and technology.* 2003.
http://www.impactscan.org/slides/eanm2002/sld014.htm. Datum pristupa:
20. svibnja 2015.

[7] Baraniuk, R. G. Compressive sensing, *IEEE Signal processing Magzine 24,* 4 (2007), 118-121.

[8] Gogić, A., Očić, N., Paliska J., Zemunik, A. Primjena compressive sensing metode na rekonstrukciju slika. http://www.fer.unizg. hr/_download/repository/Primjena_compressive_sensing_

metode_na_rekonstrukciju_slika.pdf. Projektni rad. Fakultet elektrotehnike i računarstva. 2015.

[9] Wasilewski, F. *Wavelet browser*. 2008. - 2015. http://wavelets.pybytes. com/wavelet/sym4/ Datum pristupa: 12. lipnja 2015.

[10] Grant, M. i Boyd, S. CVX: Matlab software for discipline convex programming, version 2.0 beta. http://cvxr.com/cvx, rujan 2013.

[11] Grant, M. i Boyd S. Graph implementation for nonsmooth convex progams, Recent Advances in Learning and Control (a tribute to M. Vidyasagar), Blondel, V; Boyd, S i Kimura, H, editori. Lecture notes in Control and Information Sciences Springer, 2008. http://stanford.edu/~boyd/graph_dcp.html. 95-110

Necitirana literatura:

- [1] Akram, W., Gee, S., Gamiz, C., Pan, C., Romberg, J. *Backpropagation Filters*. https://www.clear.rice.edu/elec431/projects96/DSP/filters.html Datum pristupa: 20. svibnja 2015.
- [2] Marković, V., Punda, A., Eterović, D. Pozitronska emisijska tomografija -PET i PET-CT. http://neuron.mefst.hr/docs/katedre/nuklearna_medicina/ nastavni_materijali/PET_i_PET-CT.pdf. Datum pristupa: travanj 2015.
- [3] Rezevani, N. Reconstruction Algorithms in Computerized Tomography, http://www.cs.toronto.edu/~nrezvani/CAIMS2009.pdf. Datum pristupa: 20. svibanj 2015.
- [4] Biersack, H. J.; Freeman, LM. Clinical Nuclear Medicine. Prvo izdanje. Njemačka: Springer Science & Business Media, 3. siječnja 2008.

Sažetak

Napredna rekonstrukcija slike u pozitronskoj emisijskoj tomografiji

U ovom radu proučena je problematika analiziranja podataka u pozitronskoj emisijskoj tomografiji. Proučeni su načini tumačenja podataka, te izrađene skripte za dotično tumačenje. Rastumačene su forme ulaznih podataka dobivenih od stroja, te su one pretvorene u razumljiviji oblik. Podaci su pripremljeni za provedbu napredne metode rekonstrukcije slike. Razvijena je i primijenjena napredna metoda rekonstrukcije slike bazirana na compressive sensing metodi. Podaci su tijekom višestrukih provođenja metode pomno promatrani, te su sve nepravilnosti ispravljane pri radu. Na kraju, provedena je analiza dobivenih rezultata te izvedeni zaključci o trenutnom stanju i plan za daljnje razvijanje i poboljšanje metode.

Ključne riječi

Pozitronska emisijska tomografija, List Mode Format, sinogram, Compressive sensing, rekonstrukcija

Abstract

Advanced image reconstruction in positron emission tomography

In this paper the problem of analyzing data in positron emission tomography was studied. The modes of interpreting of the data were studied and scripts for the given interpreting were made. The forms of input data given by machine were explained and transformed into more understanding form. Data were prepared for applying advanced method of image reconstruction. The advanced method for reconstruction based on compressive sensing was developed and applied. Data were closely examined during the multiple runs of the method, and all irregularities were removed in the process. In the end, the analysis of given results was made and conclusions about current status and a plan for upcoming development and improvement of the method were developed.

Keywords

Positron emission tomography, List Mode Format, sinogram, Compressive sensing, reconstruction

Programska potpora

U ovom dodatku korisniku će biti prikazani kôdovi korišteni za izradu ovog rada. Isti se kôdovi mogu iskoristiti za rekreiranje procesa koji je proveden pri izradi. Pri tome je potrebno imati na umu neka ograničenja koda, a to su slijedeća:

- Kôd je izrađen u programskom jeziku Matlab, te izvođen u programu Matlab u verzijama Matlab2013b i Matlab2014a. Ukoliko se koristi druga verzija programa, potrebno je pregledati kôd te učiniti preinake ako je potrebno.
- Kôd je koristio vanjsku knjižnicu cvx, dostupnu na <u>http:/cvxr.com/cvx/download</u>, čije se upute za instalaciju nalaze u instalaciji dobavljenoj sa stranice.
- Kôd za izvedbu knjižnice cvx izvodio se na serveru sa 32 Intel(R) Xeon(R) CPU E5-2640 v2 @ 2.00GHz procesora, iz razloga što je na računalu sa procesorom Intel i3, te 6GB RAM memorije, ponestalo memorije za izvedbu programa već kod slike veličine 32x32 piksela.

Učitavanje podataka

Na samom početku potrebno je učitati adrese kristala. Postupak zahtijeva da se učitavaju jedan po jedan skup adresa, koji se spremaju u datoteke 'nema_first.mat' i 'nema_second.mat'. Nazivi matrica ne moraju biti takvi, ali ih je onda potrebno promijeniti u daljnjim kodovima. Učitavanje je provedeno slijedećim odsječcima koda:

```
xyz_first = ucitaj_adrese_kristala('/Nema231010-1min_singles_ascii.txt');
save('nema_first.mat', 'xyz_first');
xyz_second = ucitaj_adrese_kristala('/Nema231010-layer1.txt');
save('nema_second.mat', 'xyz_second');
```

```
function [ xyz_adrese ] = ucitaj_adrese_kristala( putanja )
% Datum: 30.05.2015.
% Autor: Ankica Gogić
%
% Funkcija : UCITAJ_ADRESE_KRISTALA
%
% Svrha : Učitava 3D koordinate kristala iz LMF datoteke sa
```

```
8
             koincidencijama
8
% Parametri :
        putanja - putanja do LMF datoteke
0
9
% Vraća
        xyz adrese - matrica sa svim adresama iz LMF datoteke, po jedna u
9
Ŷ
                     svakom retku
        fid = fopen(putanja);
        tline = fgetl(fid);
        xyz adrese = [];
        while ischar(tline)
            tLineTrimmed = tline(5:end);
            lineSplitted = strsplit(tLineTrimmed);
            x val = lineSplitted(5);
            y val = lineSplitted(6);
            z val = lineSplitted(7);
            x = str2double(x val{1});
            y = str2double(y val{1});
            z = str2double(z val{1}(1:end-1));
            xyz_adrese = [xyz_adrese; x y z];
            tline = fgetl(fid);
            i=i+1;
        end
end
```

Podjela linija po ravninama

Podatke dobivene iz LMF datoteka potrebno je podijeliti u ravnine, kao što je objašnjeno u radu. Za to se koristi slijedeća skripta, koja po završetku rada proizvodi dvije matrice: *direct_bins_3D* i *indirect_bins_3D*.

U prvoj se matrici nalazi 48 matrica, pri čemu svaka predstavlja jednu direktnu ravninu. U svakoj su od njih sadržane koordinate koje određuju linije odgovora pripadne dotičnoj direktnoj ravnini. Analogno tome, u drugoj se matrici nalazi 47 matrica sa koordinatama koje određuju linije odgovora za indirektne ravnine.

```
%%FINALNA BINNING SKRIPTA
% ucitaj si podatke
load('nema first.mat');
load('nema second.mat');
% radimo binning direktnih i indirektnih plane-ova
num rays = size(xyz first, 1);
num bins = 95;
direct bins 3D = zeros(1, 4, 1, num bins);
indirect bins 3D = zeros(1, 4, 1, num bins-1);
% inicijaliziraj te matrice
for j = 1:num bins
   direct bins 3D(:,:,1,j) = zeros(1,4);
   if (j<num bins)
        indirect_bins_3D(:,:,1,j) = zeros(1,4);
    end
end
for i=1:num rays
    % pročitaj podatke o zraci
   x1 = xyz first(i,1);
   y1 = xyz_first(i,2);
    z1 = xyz_first(i,3);
   x^2 = xyz second(i,1);
   y^2 = xyz second(i,2);
    z2 = xyz_second(i,3);
    % pripremi podatke za spremanje
    data_to_store = [x1 y1 x2 y2];
    % odredi vrijednost bin-a u koji treba spremiti
   bin = round(((ceil(z1/2.3) + ceil(z2/2.3))/2)*10);
    if (bin==0)
```

```
bin =10;
   end
   % provjeri je li spremanje u direktni ili indirektni bin
   % direktni bin je ako je bin cijeli broj, a indirektni ako je bin
   % između dva cijela broja
   if (mod(bin, 10) == 0)
        % odredi kamo spremiti u matrici s podacima u pojedinom bin-u
       index = size(direct bins 3D(:,:,1,round(bin/10)),1);
       if (index == 1 && any(direct_bins_3D(:,:,1,round(bin/10)) == 0) )
            index = 0;
        end
        index = index + 1;
        %spremi u bin
       direct bins 3D(index, :, 1, round(bin/10)) = data to store;
   else
        % odredi kamo spremiti u matrici s podacima u pojedinom bin-u
        index = size(indirect bins 3D(:,:,1,round(bin/10)),1);
        if (index == 1 && any(indirect bins 3D(:,:,1,round(bin/10)) ==0))
            index = 0;
        end
        index = index + 1;
        %spremi u bin
       indirect_bins_3D(index, :, 1, round(bin/10)) = data to store;
   end
end
```

Priprema mjerene matrice bez redukcije preklapanja

Nakon što su podaci učitani, potrebno ih je obraditi i pripremiti u oblik matrice za provedbu compressive sensing metode. Za metodu podjele linija odgovora po kutovima bez redukcije preklapanja, ali uz razmazivanje gausovskim filtrom se koristi slijedeća skripta:

```
imsize = 256; % zadaj željenu veličinu slike
scale = (imsize/2)/85;
data = direct_bins_3D(:,:,1,24); % pročitaj podatke
number_of_lines = size(data,1);
mat = zeros(imsize,imsize); % alociraj memoriju
h = fspecial('gaussian',2,1); % stvori filter
```

```
filteredImg = imfilter(mat,h);
output = zeros(imsize, imsize); % stvori matricu izlaznih podataka
for l=1:number of lines
    % druga točka pravca
    second = [round(imsize/2) - data(1,1) * scale
              scale * data(1,2) + round(imsize/2)];
    % prva točka pravca
   first = [ round(imsize/2) - data(1,3) * scale
              scale * data(1,4) + round(imsize/2)];
    if (abs(second(1) - first(1)) < 1)
       continue;
    end
    % izračunaj pravac u obliku y = ax+b
    % za pravac kroz 2 točke vrijedi:
    % a = (y2-y1) / (x2-x1)
    % b = a * (-x1) + y1
   a = (second(2) - first(2)) / (second(1) - first(1));
   b = a * (0-first(1)) + first(2);
    % odredi otkud do kud može u slici ići x
    x \min = 1;
    if ((a*x min+b)>imsize)
       x min = ceil((imsize-b)/a);
   else
        if ((a*x min+b)<1)
            x min = ceil((1-b)/a);
        end
   end
   x max = imsize;
   if ((a*x max+b)>imsize)
        x_max = round((imsize-b)/a-1);
    else
```

```
if ((a*x_max+b)<1)
            x max = round((1-b)/a)-1;
        end
    end
    if (x_max < x_min)</pre>
      continue;
    end
    x = linspace(x min, x max,1000); % x-evi u višoj rezoluciji
    y = a * x + b;
                                 % odgovarajući y-oni
    index = sub2ind(size(mat),ceil(x),round(y)); % indeksi linija
    mat(index) = mat(index) + 255; % povećavanje vrijednosti piksela
end
\max m = \max(mat(:));
mat = mat./max m;
filteredImg = imfilter(mat,h); % slika filtrirana gaussovim filtrom
```

Priprema mjerene matrice uz redukciju preklapanja

Za metodu sa redukcijom preklapanja uz razmazivanje gausovskim filtrom se koristi slijedeća skripta:

```
h = fspecial('gaussian',2,1); % gauss filter
% čitaj podatke iz bin-a
input_data = direct_bins_3D(:,:,1,24);
% stvaramo angle matrice - svaka angle matrica ima svoje parove
koordinata
create_matrix_with_angle_matrices
% radimo binnanje po linijama odgovora
LOR_binning
% stvara se matrica mjerenja
measure_matrix = create_measure_matrix_no_z( 256,256, all_LORS,
lor_intensity);
% te ju se prebacuje u sliku
bp_im = iradon(measure_matrix, []);
% i razmazuje
filteredImg manji gauss = imfilter(bp im,h);
```

Pritom se koristi pomoćna skripta za stvaranje angle matrica:

```
matrix of angle matrices d1 = zeros(1, 4, 1, 360);
written_to = zeros(1, 360);
for i=1:size(input_data,1);
    if (all(input data(i,:)) == 0)
        continue;
    end
    x1 = input_data(i,1);
    y1 = input_data(i,2);
    x2 = input data(i,3);
    y2 = input data(i, 4);
    data to store = [x1 y1 x2 y2];
    angle = atan2d(y1-y2,x1-x2);
    if (round(angle) <= 0)</pre>
        angle = angle + 360;
    end
    written to(1,round(angle)) = written to(1,round(angle)) + 1;
    index = written to(1, round(angle));
    matrix of angle matrices d1( index, :, 1, round(angle)) =
data to store;
end
```

Koristi se i pomoćna skripta za podjelu preklapanja:

```
all LORS = [];
lor intensity = [];
bp_im = zeros(256, 256);
%granice kuteva za svaki kristal
boundries = [0.5938 0.58013 0.5662 0.5522 0.5380 0.5236 0.50919 0.4945
0.47976 0.2543 0.2379 0.2214 0.2083 0.1882 0.1716 0.1543 0.1381 0.1214];
for theta=1:360
  LOR indexes = zeros(1, 35);
  LOR coordinates = zeros(1, 4, 1, 35);
  measurements = matrix_of_angle_matrices_d1( :, :, 1, theta);
   % ubaci svako mjerenje u pripadni bin
   for k=1: size(measurements,1)
       current measure = measurements(k, :);
       if (all(current measure) == 0)
           continue;
       end
       % ubaci u pravi bin
```

```
x = current_measure(1);
       y = current measure(2);
       r = sqrt(x*x + y*y);
       dist from centre = r*\cos(atan(x/y) + theta*pi/180);
       for num = 1:17
           if ( abs(round(dist from centre)) <= round(boundries</pre>
                (1,num)*r) && abs(round(dist from centre)) >
                round(boundries (1,num+1)*r))
               if (dist from centre > 0)
                   num = num + 18;
               end
               if (LOR indexes(1, num) == 0)
                   LOR_coordinates(:,:,1,num) = current_measure;
               end
               LOR_indexes(1,num) = LOR_indexes(1,num) + 1;
               break;
           end
           if (num == 17 )
                if (LOR indexes(1,num+1) == 0)
                   LOR coordinates(:,:,1,num+1) = current measure;
               end
               LOR indexes(1,num+1) = LOR indexes(1,num+1) + 1;
               break;
            end
       end
   end
   % makni prazne bin-ove
   ind = [];
   for 1=1:35
       if (LOR\_indexes(1,1) \sim = 0)
           all_LORS = [all_LORS; LOR_coordinates(:,:, 1, 1)];
           lor intensity = [lor intensity; LOR indexes(1,1)];
       end
   end
end
```

I na kraju pomoćna funkcija za izradu mjerene matrice:

```
function [ measure matrix] = create measure matrix no z( picture width,
picture height, data, lor intensity)
%CREATE MEASURE MATRIX Creates measure matrix from data provided in
LMF('s)
9
    Input:
9
     picture_width - width of the given picture
8
     picture height - height of the given picture
     data - data to interprete as lines
%
8
     lor intensity - intensities of the lines (number of them on same
8
     bin)
8
   Output:
8
     measure_matrix - matrix suitable for compressive sensing code
   num adresses = size(data, 1);
   phi = zeros(num_adresses, 1);
    diagonal = ceil(sqrt(picture width^2 + picture height^2));
   measure matrix = zeros(diagonal+1,360);
```

```
angles = zeros(1, 360);
    for i=1 : num adresses
        x1 = data(i, 1);
        x^2 = data(i,3);
        y1 = data(i,2);
        y^2 = data(i, 4);
        current phi = round(atan2d(y2-y1,x2-x1));
        if (current phi <= 0)
            current phi = current phi + 360;
        end
        dist_from_centre = (sqrt(x1^2 + y1^2))*cos(atan(x1/y1) +
current_phi*pi/180);
        dist = round(dist from centre+diagonal/2);
        if dist == 0
            dist = 1;
        end
        if (dist <0 )
            continue;
        end
        measure matrix(round(dist),round(current phi)) =
measure matrix(round(dist),round(current phi))+ lor intenziteti(i,1);
    end
    mean_val = max(measure_matrix(:));
    measure_matrix = (measure_matrix/mean_val)*255;
end
```

Provedba compressive sensing-a

Jednom kad je slika spremna, a u ovom se trenutku u kodu nalazi u varijabli *filteredImg*, pokreće se skripta za provedbu compressive sensing metode.

```
%% Priprema DWT i IDWT matrica te ulaznih podataka
load('filteredImg.mat');
result_matrix = filteredImg;
xlen = size(result_matrix,1);
[ IPHI ] = create_DWT_IDWT( xlen );
xlen = xlen/2;
[ IPHI2 ] = create_DWT_IDWT( xlen );
xlen = xlen*2;
%% pokreni CVX
```

```
if ~exist('cvx___'), % Inicijalizacija optimizacijskog paketa
    cd cvx
    cvx setup
    cd ..
end
cvx begin
    cvx precision
    variable s2(xlen, xlen)
    minimize (norm(s2, 1));
    subject to
        imsr = s2;
        % druga razina
        for k=1:xlen/2
            imsr(k,1:xlen/2) = (IPHI2 * imsr(k,1:xlen/2).').';
        end
        for l=1:xlen/2
            imsr(1:xlen/2,1) = IPHI2 * imsr(1:xlen/2,1);
        end
        % prva razina
        for k=1:xlen
            imsr(k,:) = (IPHI * imsr(k,:).').';
        end
        for l=1:xlen
            imsr(:,1) = IPHI * imsr(:,1);
        end
        result matrix== imsr; % uvjet cvx metode
cvx\_end
%% rekonstrukcija
im recon = s2;
% druga razina
for k=1:xlen/2
    im_recon(k,1:xlen/2) = (IPHI2 * im_recon(k,1:xlen/2).').';
end
for l=1:xlen/2
    im recon(1:xlen/2,1) = IPHI2 * im recon(1:xlen/2,1);
end
```

```
% prva razina
for k=1:xlen
    im_recon(k,:) = (IPHI * im_recon(k,:).').';
end
for l=1:xlen
    im_recon(:,l) = IPHI * im_recon(:,l);
end
```

Rekonstruirana se slika sada nalazi u varijabli *im_recon* te ju je moguće prikazati. Ova skripta još koristi i slijedeću funkciju za izradu DWT i IDWT matrica:

```
function [ IDWT ] = create DWT IDWT( xlen )
% create DWT IDWT Funkcija koja stvara DWT i IDWT matrice za izvedbu DWT
% transformacije putem matrica
9
    Parametri: xlen - duljina manje stranice slike koja će se
00
                       transformirati
   wavelet = 'sym4';
   % DWT
   i=1;
   delta = zeros(xlen,1);
   delta(i) = 1;
   n=1;
    [c, l] = wavedec(delta, n, wavelet);
    DWT = zeros(length(c), xlen);
    DWT(:, 1) = c;
    for i=2:xlen,
        delta = zeros(xlen,1);
        delta(i) = 1;
        DWT(:,i) = wavedec(delta, n, wavelet);
    end
    DWT = wkeep2(DWT, [xlen xlen]);
    % IDWT
    if size(DWT,1) == size(DWT,2),
        IDWT = DWT';
    else
```

```
i=1;
clen = length(c);
delta = zeros(clen,1);
delta(i) = 1;
xr = waverec(delta, 1, wavelet);
IDWT = zeros(length(xr), clen);
IDWT(:,1) = xr;
for i=2:clen,
    delta = zeros(clen,1);
    delta(i) = 1;
    IDWT(:,i) = waverec(delta, 1, wavelet);
    end
end
IDWT = wkeep2(IDWT, [xlen xlen]);
end
```