Algoritam za automatsku lokalizaciju markera pacijenta u volumetrijskim snimkama

Grgić, Bruno

Master's thesis / Diplomski rad

2017

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Mechanical Engineering and Naval Architecture / Sveučilište u Zagrebu, Fakultet strojarstva i brodogradnje**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:235:055378

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-04-19**

Repository / Repozitorij:

Repository of Faculty of Mechanical Engineering and Naval Architecture University of Zagreb

SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE

DIPLOMSKI RAD

Bruno Grgić

Zagreb, 2017.

SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE

DIPLOMSKI RAD

Mentori:

Prof. dr. sc. Bojan Jerbić, mag.ing.mech.
Filip Šuligoj, mag.ing.mech.

Student:

Bruno Grgić

Zagreb, 2017.

Izjavljujem da sam ovaj rad izradio samostalno koristeći znanja i vještine stečene tijekom studija i navedenu literaturu.

Ovim putem zahvaljujem se prof. dr. sc. Bojanu Jerbiću na iskazanom povjerenju, savjetima i pomoći pri odabiru teme. Isto tako, zahvaljujem asistentu mag. ing. mech. Filipu Šuligoju na pruženoj podršci, potpori i savjetima tijekom izrade i pisanja ovog rada.

Najviše zahvaljujem obitelji i prijateljima koji su mi bili velika podrška tijekom studija i izrade ovog diplomskog rada.

SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE
Središnje povjerenstvo za završne i diplomske ispite
Povjerenstvo za diplomske ispite studija strojarstva za smjerove:
proizvodno inženjerstvo, računalno inženjerstvo, industrijsko inženjerstvo i menadžment, inženjerstvo materijala te mehatronika i robotika

Sveučilište u Zagrebu Fakultet strojarstva i brodogradnje	
Datum	Prilog
Klasa:	
Ur. broj:	

DIPLOMSKI ZADATAK

Student: **Bruno Grgić** Mat. br.: 0035185307

Naslov rada na hrvatskom jeziku: **Algoritam za automatsku lokalizaciju markera pacijenta u volumetrijskim snimkama**

Naslov rada na engleskom jeziku: **Algorithm for Automated Localization of Patient Markers in Volumetric Images**

Opis zadatka:

Precizna i točna registracija pacijenta između volumetrijskih snimaka i fizičkog prostora je važan međukorak u neurokirurškim računalom potpomognutim zahvatima koji uključuju prostorno navođenje kirurških instrumenata tijekom operacije. Registracija pacijenta temelji se na markeru koji se pričvršćuje na glavu pacijenta. Odnos položaja, markera lokaliziranog u volumetrijskim snimkama i planiranih operacijskih točaka, koristi se za povezivanje pripadajućih koordinatnih sustava. U radu se podrazumijeva primjena volumetrijskih snimaka pacijenta dobivenih pomoću uređaja za računalnu tomografiju CT (eng. *computed tomography*). CT generira niz dvodimenzionalnih (2D) snimaka u DICOM formatu (eng. *Digital Imaging and Communications in Medicine*) u sklopu kojih je potrebno automatski identificirati i lokalizirati markerske značajke (retro-reflektivne sfere). Za potrebe diplomskog rada koristi se čvrsti marker koji se sastoji od četiri retro-reflektivne sfere. Za rješavanje problema predlažu se c++ programski alati i pripadajuće knjižnice (VTK ili sl.) za pretvorbu DICOM formata u standardni matrični opis slike. Slike potom mogu biti procesuirane koristeći algoritme strojnog vida koristeći knjižnicu OpenCV ili sl. Pri očitanju DICOM-a potrebno je koristiti meta-podatke koji definiraju parametre potrebne za povezivanje Kartezijevih koordinata u metričkom sustav s pozicijama na snimci. Program mora omogućiti rad s promjenjivim parametrima obzirom na postavke CT uređaja. U radu je potrebno navesti korištenu literaturu i eventualno dobivenu pomoć.

Zadatak zadan: 11. svibnja 2017.

Rok predaje rada: 13. srpnja 2017.

Predviđeni datum obrane: 19., 20. i 21. srpnja 2017.

Zadatak zadao: Prof. dr. sc. Bojan Jerbić

Predsjednica Povjerenstva: Prof. dr. sc. Biserka Runje

SADRŽAJ

SADRŽAJ i
POPIS SLIKA ii
POPIS TABLICA iii
SAŽETAK iv
SUMMARY v
1. UVOD 1
2. TEORIJSKA PODLOGA 3
2.1. Koordinatni sustavi 3
2.2. Matrice transformacije 6
2.3. DICOM (eng. Digital Imaging and Communications in Medicine) 7
2.4. VTK (Visualization ToolKit) 9
2.4.1. Vizualizacijski cjevovod VTK 10
2.4.2. Mehanizam iscrtavanja VTK 13
2.5. Hough Transform 15
3. KORIŠTENI ALATI - SOFTWARE 18
3.1. CMake 18
3.2. Visualization ToolKit 19
3.3. vtkDICOM 20
3.4. OpenCV 21
3.5. Microsoft Visual Studio 21
3.6. MedInria 23
3.7. Grassroots DICOM (GDCM) 23
3.8. Eigen 24
4. RAZVOJ ALGORITMA 25
4.1. Orijentacija snimaka 26
4.2. Windowing 31
4.3. Detekcija kružnica 34
4.4. Filtriranje detekcija kružnica 36
4.4.1. Sekvencijalno filtriranje 36
4.4.2. Ograničena varijabilnost koordinati središta 36
4.4.3. Filtriranje po grumenima 37
4.5. Euklidska udaljenost i metoda glasanja 40
4.6. Estimacija koordinati centra sfera 42
4.7. Verifikacija rezultata 46
5. ZAKLJUČAK 48
LITERATURA 49
PRILOZI 51

POPIS SLIKA

Slika 1. a) Koordinatni sustav svijeta b) Koordinatni sustav anatomskog prostora c) Koordinatni sustav slike [3] 3
Slika 2. Perspektiva radiologa i neurologa [4] 4
Slika 3. Aksijalna, sagitalna i koronalna ravnina lubanje čovjeka [5] 4
Slika 4. Koncept koordinatnog sustava slike 5
Slika 5. Kompozicija DICOM podatkovnog elementa 8
Slika 6. vtkDataSet objekti podržani u VTK [8] 11
Slika 7. Parametri ishodišta, razmaka i dimenzija vtkImageData objekta [8] 12
Slika 8. Vizualizacijski cjevovod 13
Slika 9. Proces snimanja i struktura DICOM snimaka [10] 14
Slika 10. Hough transformacija 16
Slika 11. a) parametrizacija pravca u *xy* ravnini b) normalni zapis u parametarskom prostoru 17
Slika 12. Odabir prevoditelja prilikom izgradnje izvornog koda 18
Slika 13. Odabir modula prilikom gradnje izvornog koda VTK 19
Slika 14. Odabir modula vtkDICOM pri konfiguriranju izvornog koda 20
Slika 15. Integracija VTK knjižnice sa OpenCV knjižnicom 21
Slika 16. Stranice svojstava projekta 22
Slika 17. Grafičko sučelje programa MedInria 23
Slika 18. Slijedni dijagram rada algoritma 25
Slika 19. Snimka aksijalnog pogleda broj 40 a) obrnuti redci piksela b) standardni način zapisivanja piksela 26
Slika 20. Obrnuti redoslijed zapisivanja snimaka 29
Slika 21. Usporedba CT snimke trupe sa različitim parametrima prozora i razine 33
Slika 22. Usporedba CT snimki fantoma lubanje sa različitim parametrima prozora i razine 33
Slika 23. Detekcije svih kružnica u aksijalnom pogledu 35
Slika 24. Neželjene detekcije i oscilacije u njihovim koordinatama 36
Slika 25. Grumeni šumova sa manje od 10 sličnih detekcija 38
Slika 26. a) konstantnost radijusa u cijelom grumenu b) promjenjiv radijus detekcije sfere markera 39
Slika 27. Grumen sfere sa nedovoljnim brojem detekcija vodi pogrešnoj klasifikaciji 39
Slika 28. Grupa detekcija kružnica sa parametrima u mm 40
Slika 29. Konstrukcija sfernog markera pričvršćenog na fantom lubanje 41
Slika 30. Četiri kandidata odabrana glasanjem 41
Slika 31. Grumeni sfera markera nakon filtriranja neželjenih detekcija 42
Slika 32. Usporedba koordinatnog sustava svijeta i anatomskih pogleda 45

POPIS TABLICA

Tablica 1. Preporučeni rasponi vrijednosti parametara razina i prozora……32

Tablica 2. Iznosi grešaka detekcija sfera……47

SAŽETAK

Razvojem robotike i sve većom integracijom robotskih rješenja u svim ljudskim djelatnostima, neizbježno je širenje njihove primjene u raznim područjima medicine. Visoka razina ponovljivosti i preciznosti pruža priliku povećati sigurnost procesa koji se i dalje obavljaju ručno. Eliminacijom repetitivnih postupaka njihovom automatizacijom može se povećati efikasnost procesa. U kirurškim zahvatima, koji su iznimno vremenski ograničeni, uvođenje bržih i standardiziranih automatiziranih procesa je od velike koristi. Jedan od problema automatizacije kirurških zahvata je pronalaženje markera potrebnog za povezivanje koordinata koordinatnih sustava slike sa fizičkim prostorom. Registracija pacijenta neizostavan je korak u neurokirurškim računalom potpomognutim zahvatima poput slikom navođene radijacijske terapije, radiokirurgije, minimalno invazivne terapije, endoskopije i intervencijske radiologije. Osim skraćivanja vremena ciklusa zahvata, prednost je lakše i bolje navođenje tokom intervencije koji vode smanjenoj invazivnosti i povećanoj točnosti. Registracijom pacijenta moguće je dobiti informacije o položajima instrumenata u odnosu na planiranu trajektoriju, okolnih osjetljivih područja i konačnog cilja. Jedan od ključnih čimbenika koji doprinose grešci registracije je ručna lokalizacija značajki. Ovaj rad bavi se razvojem algoritma za automatsku registraciju pacijenata baziranoj na odnosu volumetrijskih snimaka i fizičkog prostora. Volumetrijske snimke dobivene su pomoću uređaja za računalnu tomografiju CT (eng. *computed tomography*). Niz dvodimenzionalnih snimaka CT uređaja omogućava lociranje čvrstog markera pričvršćenog na glavu pacijenta. Snimke su pohranjene u DICOM (eng. *Digital Imaging and Communications in Medicine)* formatu. Za svrhu ovog rada korišten je fantom (imitacija) lubanje na koji je pričvršćen marker sa četiri retro-reflektivne sfere. Algoritam je razvijan u C++ programskom jeziku. Upotrebom VTK knjižnice (eng. *Visualization ToolKit Library*), osim pretvorbe DICOM formata u neki od standardno korištenih formata poput PNG-a, korišteni su parametri definirani u meta-podacima koji sadrže informacije potrebne za povezivanje Kartezijevih koordinata u metričkom sustavu sa pozicijom na snimci. Korištena je OpenCV knjižnica strojnog vida za obradu slika kako bi se detektirale sfere markera, uklonile neželjene detekcije i povećala robusnost algoritma. Usporedbom detekcija sfera markera transformiranih u Kartezijev koordinatni sustav i mjera markera dobivenih u laboratoriju, detekcije su potvrđene i dobivene su procjenjene greške registracije.

Ključne riječi:

Registracija slike, Robotika u medicini, Strojni vid, DICOM, OpenCV

SUMMARY

With the development of robotics and increasing integration of robotic solutions in all human activities, the spread of their application in medicine is unavoidable. High levels of repeatability and precision offer an opportunity to increase the safety of manual procedures. Eliminating repetitive processes by their automation the efficiency of the process can be increased. In surgical procedures, which are extremely time limited, introduction of faster and standardized automated processes can be of great benefit. One of the problems encountered automating surgical procedures is marker localization necessary for alignment of the image coordinates with the physical space. Patient registration is a necessary step in computer aided neurosurgery procedures as image guided radiation therapy, radiosurgery, minimally invasive surgery, endoscopy and interventional radiology. Apart from the reduction of process time intervals, easier and improved navigation during the interventions lead to reduced invasiveness and increased accuracy. By patient registration it is possible to obtain information about the position of surgical tools in relation to the planned trajectory, nearby sensitve surfaces and the final target. One of the key factors leading to the localization error is manual localization of features. This thesis deals with development of the algorithm for automatic patient registration based on the relation of the volumetric images and the physical space. The volumetric images are obtained by a CT scanner. The series of two-dimensional (2D) images allow the localization of the marker fixed to the patient's head. Images are stored in DICOM (Digital Imaging and Communications in Medicine) format. For the purpose of this thesis, the marker with four retro-reflexive spheres was attached to the skull phantom. The algorithm is developed in C++ programming language. Using VTK (Visualization ToolKit Library) library, the medical images were transformed to PNG format and parameters stored in medical images were used to read metadata which contains information necessary for the transformation from the image space to the Cartesian coordinates (metric values). Computer vision library OpenCV was used for image processing and circle detection, removal of false positives and increasing the robustness of the algorithm. Comparing the marker sphere detections transformed into the Cartesian coordinate system and marker measurements obtained from the laboratory, detections were confirmed and estimated registration error was obtained.

Key words:

Image registration, Medical robotics, Computer vision, DICOM, OpenCV

1. UVOD

Razvoj prvih ali i velikog broja modernih robota u neurokirurgiji zasniva se na dizanju stereotaktičkih okvira i razvoju metoda snimanja pacijenta. Povijest razvoja stereotaktičkih okvira počinje 1908. godine, a snimanja pacijenta 1960ih godina razvojem prvih CT i MR skenera. Prekretnica robotike u kirurgiji nastaje 1980ih godina eliminacijom hidrauličkih kompomenti i povećanjem preciznosti izvršnih članova robota, kada su izvedene prve kirurške operacije robotom. Zahtjevi za preciznošću i većom efikasnošću repetitivnih postupaka omogućili su ubrzani razvoj novih tehnologija korištenih u operacijskim salama. Visoki troškovi izvršavanja radova tokom operacija podupiru uvođenje novih tehnologija kojima je cilj smanjiti vrijeme izvođenja registracije pacijenata i navođenja kirurških alata. Potrebe pacijenata, smanjivanje broja grešaka tokom zahvata, uvođenje novih mogućnosti tokom operacija i minimalno invazivnih zahvata neki su od uvjeta koji se pokušavaju zadovoljiti uvođenjem automatiziranih robotskih rješenja. Najpoznatiji primjer je da Vinci kirurški sistem kojim se obavljaju minimalno invazivni zahvati. Sa više od 3800 postavljenih jedinica i 200,000 izvršenih zahvata godišnje, primjer je uspješno implementiranog robotskog sistema u kirurgiji [1]. Unatoč prednostima da Vinci sistema kao što su ublažavanje krvarenja, manji ožiljci i brže zacjeljivanje, određene karakteristike sprječavaju širu primjenu. Sustav nije autonoman, operater upravlja robotskim sistemom putem konzole. Dugo trajanje obuke operatera i visoka cijena sistema čine uređaj nepristupačnim manjim bolnicama, dok zauzimanje velikog prostora ograničava pristup pomoćnom osoblju tokom operacije. Smjer razvoja novih rješenja usmjeren je mobilnim robotima manje mase i dimenzija koje se teži smjestiti na operacijski stol, strop sale ili mobilna kolica.

Za planiranje operacijskih zahvata i prostorno navođenje kirurških instrumenata nužno je obaviti postupak registracije slike. Obradom informacija dobivenih iz volumetrijskih snimaka medicinskih uređaja povezuju se podaci iz različitih snimaka. Uspoređuje se referentni pogled sa ostalim pogledima, snimke snimljene u različitim vremenskim intervalima ili snimke snimljene različitim medicinskim uređajima. Algoritmima računalnog vida moguće je izdvojiti informacije bitne za planiranje liječenja, praćenje stanja pacijenta ili definiranja pozicije i orijentacije markera [2]. Snimke se geometrijski usklađuju prostornom transformacijom.

Najčešće se koriste snimke CT (eng. *Computed Topography)* ili MRI (eng. *Magnetic Resonance Imaging)* uređaja. CT snimke dobivene iz CT uređaja nastaju emitiranjem X-zraka.

Generator X-zraka sastoji se od vakuumske cijevi, katode koja šalje elektrone kroz vakuum i anode koja prima elektrone. Detektori X-zraka registriraju njihov intenzitet koji ovisi o gustoći materijala kroz koji prolaze zrake. Gusti materijali poput kosti propuštaju manje X-zraka nego mekana tkiva, što se očitava kao zasićenost na CT snimci. Bijela boja predstavlja visoku zasićenost, tj. malu propusnost zraka, dok je mala zasićenost i visoka propusnost predstavljena tamnim područjima CT snimke. Nakon tomografske rekonstrukcije, vrijednosti zasićenja snimke predstavljene su intenzitetom piksela veličine 12bita, u rasponu od -1024 do +3071. Budući da svaka snimka ima jednaku debljinu, ona sadrži informaciju o svome položaju u nizu snimaka. Ta treća dimenzija definira dodatnu mjernu jedinicu zvanu voksel. Retro-refleksivne kuglice korištene na markeru u ovom radu imaju nisku propusnost X-zraka pa je moguće raspon intenziteta piksela prilagoditi detekcijama kosti i markera kako bi se eliminirao dio nepoželjnih detekcija. CT snimke najčešće se izrađuju kao niz snimaka iz aksijalnog pogleda, iz kojih je moguće naknadno napraviti rekonstrukciju ostalih pogleda. U ovom radu iz snimaka aksijalnog pogleda, rekonstruiran je niz sagitalnih snimaka putem VTK knjižnice. VTK je besplatan open-source programski paket koji se sastoji od knjižnice za C++ ili interpretiranih sučelja za ostale jezike poput Pythona i Jave, a koristi se za 3D računalnu grafiku, procesiranje slika i vizualizaciju. Nizovi snimaka u aksijalnom i sagitalnom pogledu pretvoreni su iz DICOM formata u PNG format, kako bi mogli biti obrađeni algoritmima strojnog vida putem OpenCV knjižnice. Razlikuje se registracija slike temeljena na intenzitetu i na značajkama. Iz dizajna sfera markera, prevučenih tankim slojem reflektivnog materijala, u ovom radu korištena je metoda obrade slike temeljena na intenzitetu slike [3]. Postavljanjem određenih granica intenziteta, slika se pretvara u binarnu gdje se pikselima vrijednosti intenziteta ispod granice dodjeljuje vrijednost 0, dok onima iznad granice vrijednost 1.

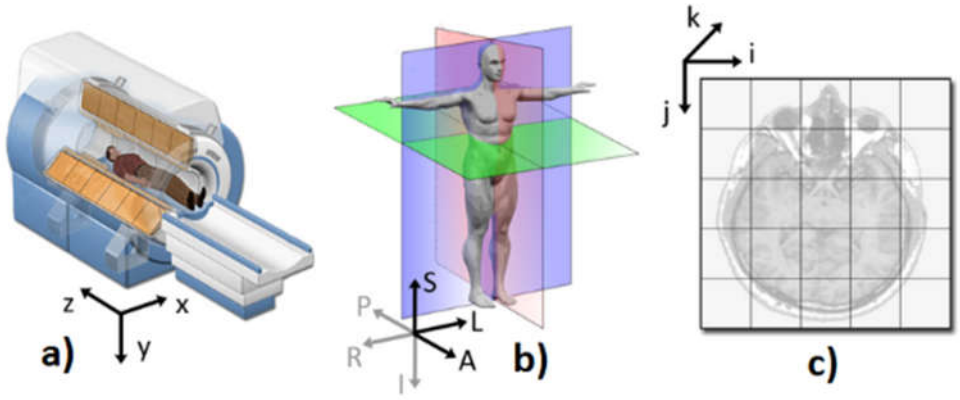
Retro-refleksivne sfere u snimkama detektirane su kao kružnice. Korištena je modifikacija Houghove transformacije za prepoznavanje značajki kružnica. U nizu snimaka nalazi se mnoštvo detekcija kružnica koje nisu sfere markera. Svojstvima sfere poput dimenzija, promjenjivog radijusa i uvjeta Euklidske udaljenosti između sfera poznatih iz dimenzija markera, izdvojene su nepoželjne detekcije od detekcija sfera markera.

2. TEORIJSKA PODLOGA

2.1. Koordinatni sustavi

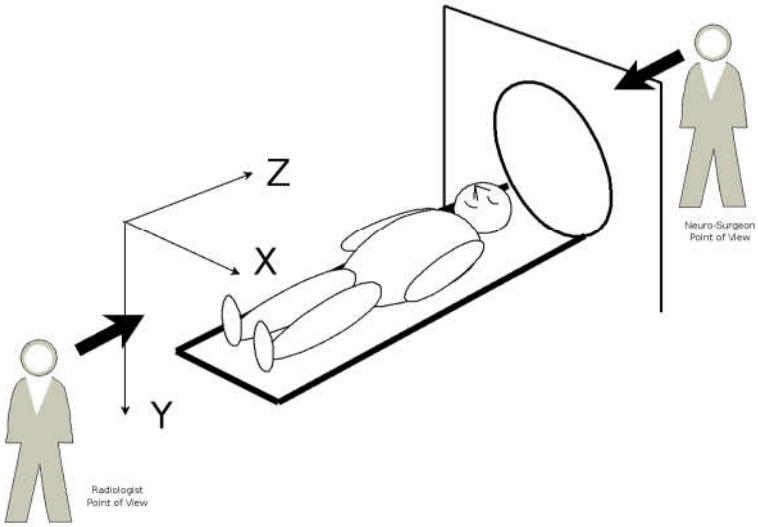
U postupku snimanja i obrade medicinskih snimaka često se koriste tri koordinatna sustava. To su koordinatni sustav svijeta na **Slika 1. a)**, koordinatni sustav anatomskog prostora na **Slika 1. b)** i koordinatni sustav slike, vidljiv na **Slika 1. c)**. Koordinatni sustav svijeta jedini je izražen u stvarnim mjernim jedinicama, dok su ostali koordinatni sustavi konceptualni.

Potrebno ih je pažljivo uskladiti kako bi se koordinate mogle pouzdano transformirati iz konceptualnih koordinatnih sustava u fizički prostor.



Slika 1. a) Koordinatni sustav svijeta b) Koordinatni sustav anatomskog prostora c) Koordinatni sustav slike [4]

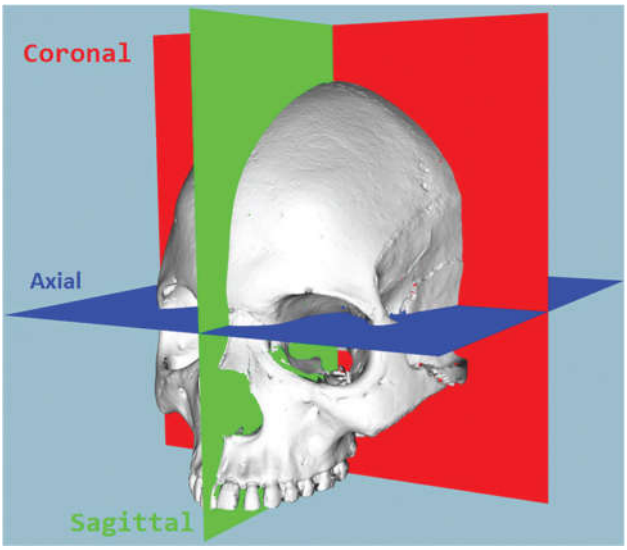
Koordinatni sustav svijeta kartezijski je koordinatni sustav u kojem su smješteni medicinski uređaj ili pacijent. Ovisi o perspektivi promatrača. Razlikuju se perspektiva radiologa, čiji pogled započinje nogama pacijenta i perspektiva neurokirurga, čiji pogled započinje glavom pacijenta, ilustrirano na **Slika 2.** Iz toga se može zaključiti da orijentacija snimaka ovisi o implementaciji. Najčešće se koristi perspektiva radiologa, koja je korištena i u ovom radu.



Slika 2. Perspektiva radiologa i neurologa [5]

Model koordinatnog sustava anatomskog prostora ključan je u procesima obrade medicinskih slika. Sastoji se od tri ravnine koje opisuju standardne anatomske pozicije pacijenta:

- aksijalna ravnina razdvaja noge od glave
- sagitalna ravnina razdvaja lijevu stranu pacijenta od desne
- koronalna ravnina razdvaja prednji dio pacijenta od stražnjega

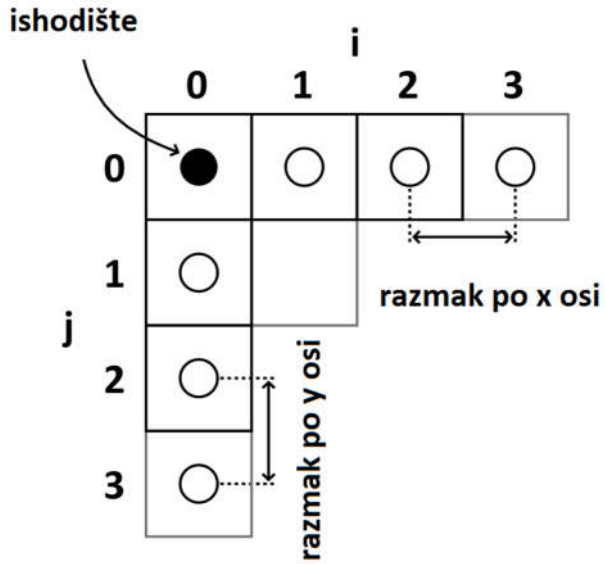


Slika 3. Aksijalna, sagitalna i koronalna ravnina lubanje čovjeka [6]

Anatomski dijelovi tijela definirani su uniformnim nazivima koji se koriste u medicinskim aplikacijama. Tako je dio glave pacijenta nazvan *Superior*, donji dio tijela koji počinje od nogu *Inferior,* lijeva strana tijela *Left,* desna *Right*, te prednji i stražnji dijelovi tijela *Anterior* i *Posterior*.

Anatomski prostor kontinuiran je u tri dimenzije u kojem je slika uzorkovana. Baza tog trodimenzionalnog prostora definirana je anatomskim osima *Inferior-Superior, Left-Right, Anterior-Posterior* baziranim na prethodno opisanim pojmovima. Ovisnost anatomskih osi o dijelovima tijela prikazana je na **Slika 1. b)**. Smjer koordinatnih osi trodimenzionalne baze prilagođen je medicinskim aplikacijama. Kao dio DICOM standarda koristi se LPS sustav u kojem je smjer pozitivnih osi orijentiran u smjeru *Left, Posterior, Superior* osi.

Koordinatni sustav slike ukazuje kako je slika snimljena s obzirom na anatomski prostor. Medicinski uređaji prilikom snimanja slike stvaraju uređene, pravokutne strukture podataka (polja) čije najniže vrijednosti počinju u gornjem lijevom uglu. Os *i* se povećava udesno, os *j* prema dolje i os *k* unazad, tj. pozitivan smjer gleda u ekran. Kao što je navedeno u poglavlju 2.4.1 o cjevovodu VTK knjižnice, objekt podataka *vtkImageData* uz intenzitet svakog piksela, tj. voksela, pohranjuje podatke ishodišta i razmaka između voksela anatomskog prostora, ilustrirano na **Slika 4.**



Slika 4. Koncept koordinatnog sustava slike

2.2. Matrice transformacije

Kako bi se koordinate iz jednog koordinatnog sustava mogle očitati u drugom, potrebno je primijeniti matrice transformacije.

Transformacija vektora prostora koordinatnog sustava slike $\vec{\imath}, \vec{\jmath}, \vec{k}$ u koordinatni sustav anatomskog prostora $\vec{x}$ je afina transformacija. Sastoji se od linearne transformacije, tj. matrice rotacije A i vektora translacije $\vec{t}$:

$$\vec{x} = A(\vec{\imath}, \vec{\jmath}, \vec{k}) + \vec{t} \tag{2.1}$$

Matrica rotacije A je matrica dimenzija 3 x 3 i nosi informacije o orijentaciji osi. Vektor $\vec{t}$, dimenzija 3 x 1, o geometrijskom položaju prvog voksela:

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} \begin{bmatrix} i \\ j \\ k \end{bmatrix} + \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix} \tag{2.2}$$

Linearna transformacija je dakle definirana množenjem matrice rotacije i jediničnih vektora $\vec{\imath}, \vec{\jmath}, \vec{k}$ i zbrajanjem vektora translacije. Matrice rotacije za neki kut rotacije θ glase:

$$A_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\theta & -sin\theta \\ 0 & sin\theta & cos\theta \end{bmatrix}, A_y = \begin{bmatrix} cos\theta & 0 & sin\theta \\ 0 & 1 & 0 \\ -sin\theta & 0 & cos\theta \end{bmatrix}, A_z = \begin{bmatrix} cos\theta & -sin\theta & 0 \\ sin\theta & cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.3}$$

Kako bi se pojednostavnio zapis, matrici rotacije dodaje se redak nula i stupac elemenata vektora translacije, proširen brojem jedan [4].

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ 1 \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} & A_{13} & t_1 \\ A_{21} & A_{22} & A_{23} & t_2 \\ A_{31} & A_{32} & A_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} i \\ j \\ k \\ 1 \end{bmatrix} \tag{2.4}$$

Sfere markera bilo je potrebno detektirati u dva pogleda kako bi se minimizirala greška registracije. Standardna orijentacija korištenog niza CT snimaka koje čine volumen je aksijalna, dok se ostali pogledi mogu rekonstruirati iz volumena. Poznavajući dimenzije snimke, njihov broj i debljinu (razmak između snimaka), koji su nepromjenjivi parametri, lako je izračunati dimenzije kvadra naslaganih snimaka. Cilj je bio pronaći centar tog volumena, te oko njega pomoću matrica transformacije napraviti rekonstrukciju svake snimke zasebno. Za rekonstrukciju svake iduće snimke potrebno je pomicati centar transformacije u smjeru normale na rekonstruiranu snimku. Pretvorbom medicinskih snimaka iz različitih pogleda, detektirani su oblici koji posjeduju geometrijske značajke kružnice na svakoj snimci. Primjenom različitih metoda filtriranja željelo se samo ispravne detekcije svrstati u grumene koji predstavljaju detekcije retro-refleksivnih sfera markera. Pojava grumena šumova je neizbježna. Njihov broj je minimiziran eliminacijom grumena čiji elementi ne posjeduju karakteristike detekcija sfere markera prije primjene posljednje verfikacije metodom Euklidske udaljenosti.

2.3. DICOM (eng. Digital Imaging and Communications in Medicine)

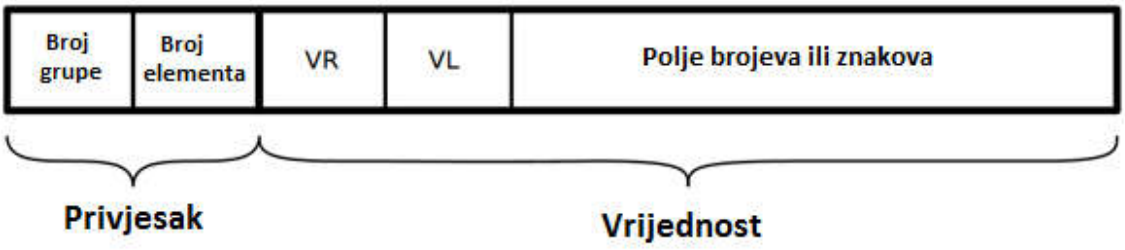
DICOM je međunarodni standard za medicinske snimke i informacije, objavljen 1993. godine. Definira formate za razmjenu medicinskih snimaka, vezanih podataka i kvalitetu za medicinsku upotrebu. Primjenjuje se u gotovo svakom području medicine: radiologiji, kardiologiji, radioterapijskim uređajima poput CT skenera, MR, ultrazvuka ali sve više i u ostalim područjima poput oftamologije i stomatologije. Omogućava digitalnu pohranu i uklanja potrebu za fizičkim medijem za prikaz radioloških nalaza. Koristi se na desecima tisuća uređaja, sa više od milijardu DICOM snimaka u kliničkoj upotrebi [7]. Vijeće DICOM-ovog standarda sastoji se od kompanija koje proizvode uređaje za snimanje u medicini, organizacija koje zastupaju medicinske stručnjake i državnih institucija.

U DICOM standardu definirane su sljedeće značajke:

- Niz protokola koje uređaji trebaju pratiti u mrežnoj komunikaciji
- Sintaksa i semantika naredbi i vezanih informacija izmjenjivanih pri korištenju protokola

- Skup usluga za pohranu medija, format datoteka, struktura medicinskog imenika za pristup snimkama i vezanim informacijama
- Informacije nužne za implementaciju u skladu sa standardom

Iako pojam „meta-podaci“ nije službeno definiran DICOM standardom, koristi se za sve podake unutar DICOM datoteke koji ne predstavljaju piksele. Svi atributi niza DICOM datoteka pohranjeni su kao uređeni niz podatkovnih elemenata. Svi podatkovni elementi sastoje se od 32-bitnog ključa zvanog privjesak (*eng. tag*) sa pripadajućom vrijednošću (*eng. value*) veličine nula ili više bajtova. Svaki atribut DICOM meta-podataka identificiran je 32-bitnim privjeskom zapisanim u heksadecimalnom obliku. Privjesak se sastoji od četveroznamenkastog broja grupe i četveroznamenkastog broja elementa. Svaki privjesak koji koristi parni broj grupe je strogo definiran DICOM standardom, dok neparni brojevi grupa dozvoljavaju proizvođaću opreme da ih sam definira i zadrži privatnim.



Slika 5. Kompozicija DICOM podatkovnog elementa

Svaki DICOM podatkovni element ima VR (*eng. Value Representation)* kod sastavljen od dva slova koji određuje vrstu podataka i format vezane vrijednosti. Vrsta podataka je tekst ili binarni format, a kao razdjelnik više vrijednosti koristi se obrnuta kosa crta.

Vrijednost unutar DICOM podatkovnog elementa može biti polje (*eng. array)* vrijednosti. Dozovljen broj predmeta unutar polja nazvan je mnogostrukost vrijednosti VM (*eng. Value Multiplicity)*. Za većinu vrijednosti atributa definiranih DICOM standardom VM iznosi 1. Klasa *vtkDICOMValue* čitača snimaka unutar vtkDICOM knjižnice omogućava jednostavnu

pretvorbu između različitih nativnih tipova podataka definiranih unutar C++ programskog jezika.

Razvojem standarda rastao je i broj informacija sadržanih u medicinskim snimkama. Budući da su informacije sadržane u podatkovnim elementima, njihov opsežan popis nalazi se u riječniku. Riječnik podataka (*eng. data dictionary*) definiran DICOM standardom je centralizirani registar koji određuje zbir svih DICOM podatkovnih elemenata za predstavljanje informacija.

Za svaki podatkovni element, rječnik podataka definira:

- jedinstveni privjesak, koji se sastoji od grupe i broja elementa
- njegovo ime
- tip vrijednosti (integer, character string)
- količinu vrijednosti (broj vrijednosti po atributu)

Cijeli rječnik dostupan je na web stranicama DICOM standarda, a intuitivan preglednik sa objašnjenjima pojedinih elemenata dostupan je na [8].

2.4. VTK (Visualization ToolKit)

VTK knjižnica je objektno orijentirana. Model bilo kojeg VTK objekta baziran je na „superklasi“ *vtkObject*. Gotovo sve klase izrađuju se na istom principu, te pri njihovom definiranju moraju koristiti New() metodu, te Delete() za njihovo brisanje. Koristeći zajedničku „superklasu“ i standardiziranu metodu definiranja i brisanja objekata, VTK pruža nekoliko objektno orijentiranih operacija:

- Brojanje referenci
- Informacije o vrsti podataka prilikom izvršavanja koda
- Prikaz stanja objekta

Dvije glavne komponente VTK knjižnice su vizualizacijski cjevovod (eng. *Visualization pipeline*) i mehanizam iscrtavanja (eng. *Rendering engine).*

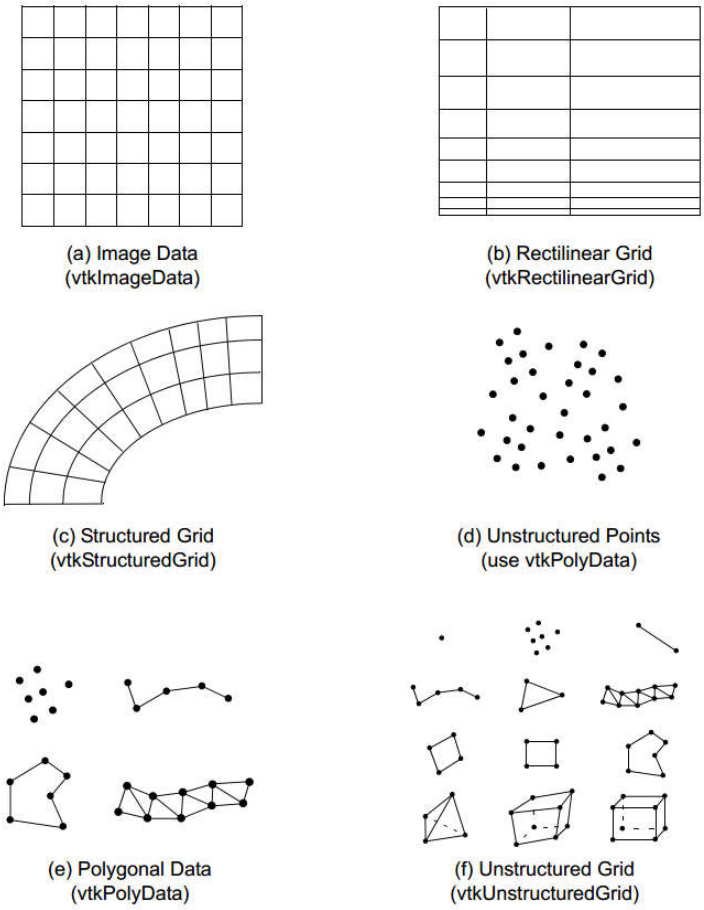
Vizualizacijski cjevovod koristi se za stvaranje i dohvaćanje podataka, obradu tih podataka i spremanje podataka u datoteku ili proslijeđivanje tih podataka u mehanizam iscrtavanja. Mehanizam iscrtavanja zaslužan je za vizualizaciju podataka. Navedene komponente su konceptualne te se ne smatraju dijelom arhitekture VTK knjižnice [9].

2.4.1. Vizualizacijski cjevovod VTK

Korisiti se pri izradi i čitanju podataka, analizu i pretvorbu iz jednog tipa podataka u drugi. U ovom radu, učitane su DICOM snimke, podaci tipa skalara pretvoreni su u „unsigned short“, zatim nakon što je odrađena obrada tih podataka, zapisanu su na tvrdi disk. Kako bi se informacije pretvorile u grafičke reprezentacije, VTK koristi metodu protoka podataka. U tom procesu pojavljuju se dvije osnovne vrste objekata:

- vtkDataObject
- vtkAlgorithm

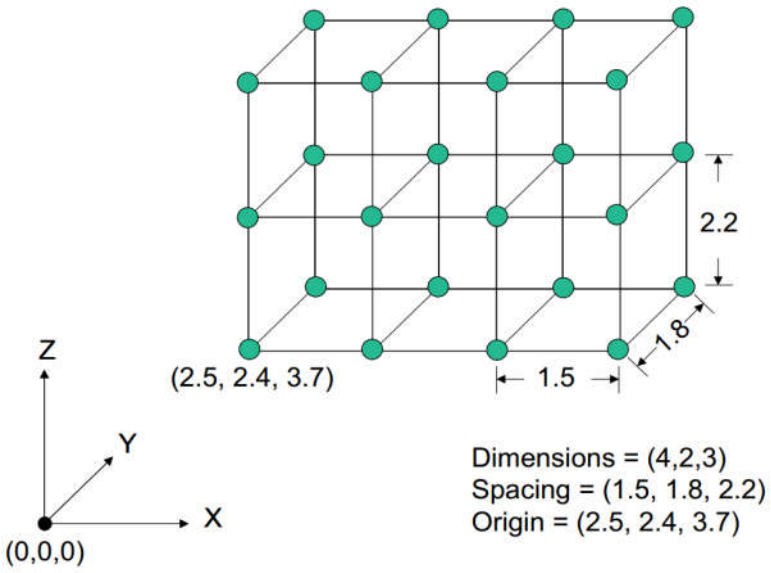
Data objekti predstavljaju podatke različitih tipova. Klasa *vtkDataObject* smatra se generaliziranom skupinom podataka, dok su podaci koji imaju definiranu strukturu predstavljeni *vktDataSet* klasom. Na **Slika 6.** vide se podržani tipovi dataset objekata.



Slika 6. vtkDataSet objekti podržani u VTK [9]

Skupovi podataka definirani u *vktDataSet* objektima imaju geometrijsku i topološku strukturu. Topološka struktura određuje sastoji li se skup podataka od točaka ili ćelija. Na *vktDataSet* objekte se putem točaka ili ćelija također vežu atributivni podaci poput skalara i vektora [10].

Skupovi podataka slika, pa tako i snimaka CT skenera definirani su klasom *vtkImageData*. Taj tip podataka uređene je geometrije i topologije što znači da su pozicije točaka u skupu definirane parametrima ishodišta, razmaka i dimenzija. CT, MR i ultrazvuk skeneri proizvode snimke sa podacima takvog uređenja. Iz **Slika 6. a)** može se vidjeti pravilna raspodjela *vtkPixel* (*vtkVoxel* u 3D) ćelija *vtkImageData* data objekta. Umjesto zapisa u ćelije, koji se koristi više kao konceptualna predodžba, takvi podaci zapisani su u poljima.

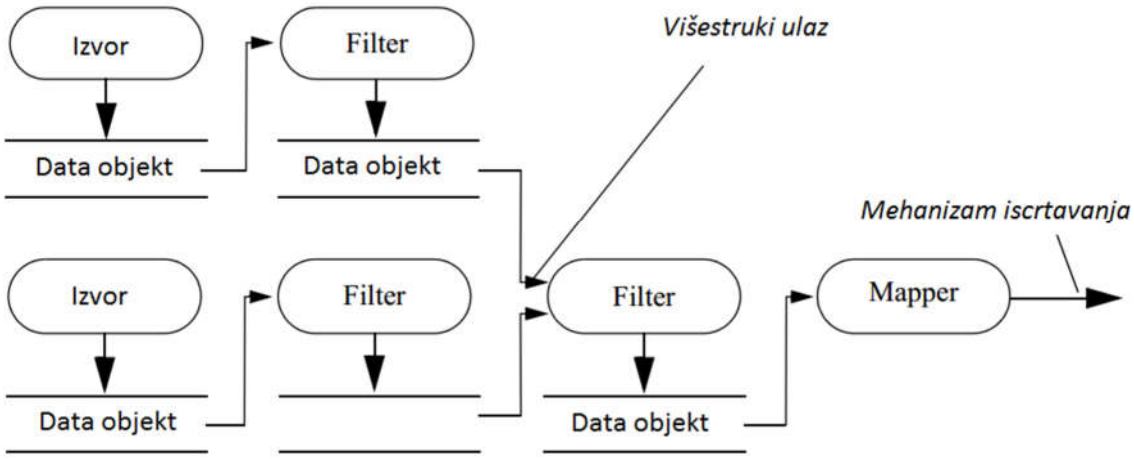


Slika 7. Parametri ishodišta, razmaka i dimenzija vtkImageData objekta [9]

Druga skupina objekata, algoritmi, često zvana i filteri, djelovanjem na data objekte stvaraju nove vrste data objekata. Zajedno sa data objektima i njihovim međudjelovanjem čine vizualizacijski cjevovod vidljiv na **Slika 8.** Algoritmi stvaraju data objekte ili čitaju postojeće, koji su ulaz filterima. Izlaz filtera je jedan ili više novih data objekata. Na kraju cjevovoda nalaze se „*Mapper"* objekti koji data objekte pretvaraju u vizualnu reprezentaciju prikazanu mehanizmom iscrtavanja. „*Writer"* objekt može se smatrati vrstom „*Mapper*" objekta koji zapisuje podatke iz data objekta u datoteku.

Postoji nekoliko značajki bitnih za konstrukciju vizualizacijskog cjevovoda. Prilikom konstrukcije topologije cjevovoda, potrebno je izlaz prethodnog filtera postaviti kao ulaz nadolazećeg filtera, kako bi postojao neprekinuti protok podataka. Neki filteri mogu imati više ulaza i više izlaza. Također, mora postojati mehanizam za kontrolu rada cjevovoda. Cilj je da radi samo onaj dio cvjevovoda potreban da bi se ažurirao izlaz. To se postiže shemom lijene evaluacije koja se izvršava samo u slučaju zahtjeva za podacima, baziranoj na vremenu modifikacije objekta. Cjevovod je sastavljen na taj način da objekti moraju biti kompatibilni da bi mogli biti povezani u cjevovodu. U protivnom, dolazi do grešaka prilikom izvršavanja. Uz to, budući da se u obradi slika i računalnoj grafici često radi o velikim skupovima podataka, važno je planirano raspolagati memorijom. Pri prolasku kroz cjevovod i izvršavanju algoritama, može se odlučiti žele li se podaci zadržati ili poslati u priručnu memoriju. Za

dodatno planiranje raspolaganja resursima, omogućeno je izvršavati dio po dio cjevovoda, te pristupati podacima putem referenci.



Slika 8. Vizualizacijski cjevovod

2.4.2. Mehanizam iscrtavanja VTK

Mehanizam iscrtavanja sadrži se od klasa VTK knjižnice zaslužnih za prikazivanje rezultata vizualizacijskog cjevovoda u prozoru. Sljedeći objekti su najčešće korišteni, a definirani su kao „superklase“, tj. ostale klase temelje se na njima:

- vtkProp
- vtkTransform
- vtkLookupTable
- vtkRenderWindow
- vtkRenderer

Meta-podacima unutar VTK može se pristupiti putem klasa *vtkDICOMReader* i *vtkDICOMParser*. Prethodna čita slikovne podatke i meta-podatke, dok potonja čita samo meta-podatke. Klasa vtkDICOMImageReader, prisutna kao standardni dio VTK knjižnice, ne

može čitati meta-podatke. Za tu svrhu potrebno je koristiti prethodno navedene klase iz vtkDICOM knjižnice.

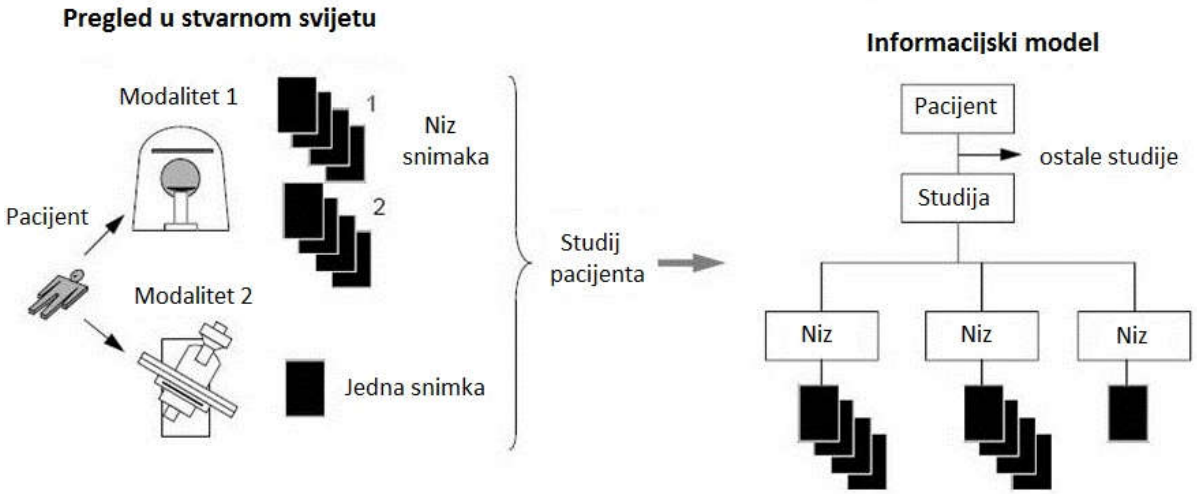
Na početku konceptualnog VTK cjevovoda, postavljen je čitač *vtkDICOMReader* kao izvor informacija.

```
vtkSmartPointer<vtkDICOMReader> reader =
vtkSmartPointer<vtkDICOMReader>::New();
```

Klasa *vtkDICOMDirectory* pretražuje cijelu mapu sa DICOM datotekama. Kao izlaz vraća listu svih snimaka, uređenih prema pacijentu, studiji, seriji snimke i samim slikama.

```
vtkSmartPointer<vtkDICOMDirectory> dicomdir =
      vtkSmartPointer<vtkDICOMDirectory>::New();
dicomdir->SetDirectoryName("C:/vtk/mydata/phantom1");
dicomdir->Update();
```

Svakom modifikacijom podataka u cjevovodu, taj dio cjevovoda je potrebno ažurirati naredbom „Update()“ kako bi podaci bili sinkronizirani.



Slika 9. Proces snimanja i struktura DICOM snimaka

Iz **Slika 9.** može se vidjeti da je moguće više serija sa raznih medicinskih uređaja pohraniti u jedan niz DICOM snimaka. One se tada razlikuju po svojim podatkovnim elementima u meta-

podacima, gdje se može provjeriti kojem uređaju, pacijentu i seriji pripada niz. U ovom slučaju, prisutna je bila jedna serija snimaka sa CT uređaja.

```
vtkStringArray *sortedFiles = dicomdir ->GetFileNamesForSeries(j1);
```

Varijabla *j1* u ovom slučaju predstavlja prvu seriju snimaka.

```
  reader->SetFileNames(sortedFiles);
```

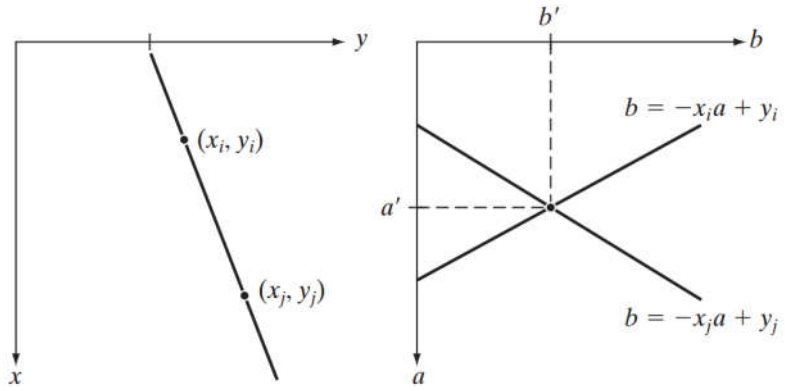
U klasu čitača podataka se postavlja uređena lista snimaka. Sada se na jednostavan način može raditi sa bilo kojim nizom DICOM snimaka, jedino što treba je promijeniti naziv mape.

2.5. Hough Transform

Bez obzira na perspektivu iz koje se promatra volumen snimaka, sfera markera na slici imati će oblik kružnice. Oblike kružnica potrebno je detektirati nekom od metoda strojnog vida. Detekcije kružnica pohranjuju se u obliku koordinata središta i radijusa kružnice [12].

Algoritam korišten za detekciju kružnica je *HoughCircles* baziran na Hough transformaciji. Radi jednostavnosti ilustracije, slijedi opis transformacije koordianta u parametarski prostor pravca. Transformacija je primjenjiva na bilo koju parametarsku jednadžbu.

Ako postoji točka (x_i,y_i) u *xy* ravnini, tada je jednadžba pravca $y_i = ax_i + b$, gdje su *a* nagib, *b* pomak. Kroz točku (x_i,y_i) prolazi beskonačno mnogo linija (pravaca), ali sve zadovoljavaju jedndažbu $y_i = ax_i + b$, za promjenjive parametre *a* i *b*. Ako se jednadžba pravca zapiše kao $b = -ax_i + y_i$ u *ab* ravnini, koja se još zove parametarski prostor, par koordinata (x_i,y_i) postaju parametri pravca (nagib i pomak) sa promjenjivim koordinatama *a* i *b*. Ako se uzme još jedna točka (x_j,y_j) koja u *xy* ravnini leži na istome pravcu kao i prethodna točka, tada će se njen pravac sijeći sa pravcem prethodne točke u parametarskom prostoru. Sjecište ta dva pravca u parametarskom prostoru definira a' i b' parametre pravca koji prolazi kroz točke (x_i,y_i) i (x_j,y_j) u *xy* ravnini. Tim načinom, moguće je sve točke (x_k,y_k) prikazati u parametarskom prostoru kao pravce. Sjecišta tih pravaca određuju nagib, pomak i točke pravca u *xy* ravnini.



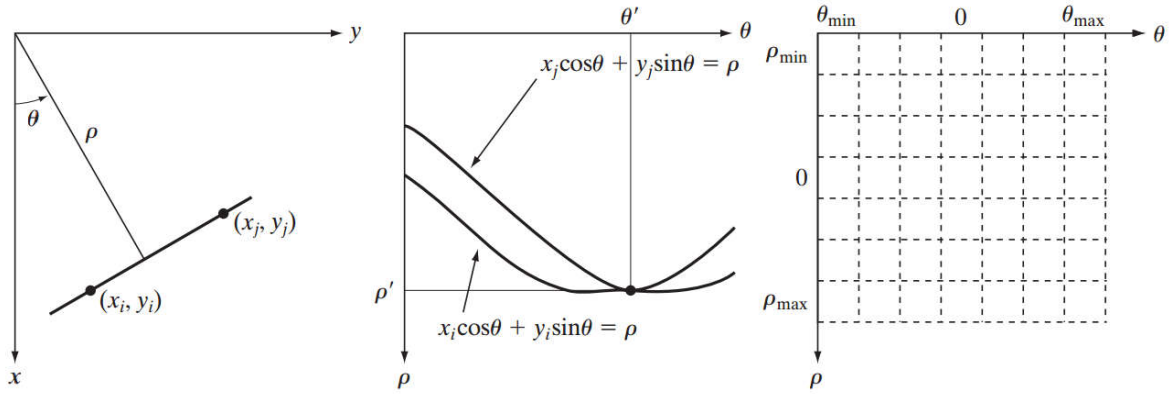
Slika 10. Hough transformacija

Problem nastaje kada je pravac sve više vertikalan, jer parametar nagiba pravca teži beskonačnosti. Način da se to izbjegne je polarni zapis jednadžbe pravca:

$$x \cos \theta + y \sin \theta = \rho \tag{2.5}$$

Na **Slika 11. a)** može se vidjeti geometrijska interpretacija polarnog zapisa pravca u *xy* ravnini. Za slučaj kada je pravac horizontalan, kut $\theta = 0°$, a ρ je jednak pozitivnom djelu *x*-osi. Kada je pravac vertikalan i $\theta = 90°$, ρ je jednak pozitivnim djelu *y*-osi -90°, a ako je $\theta = -90°$, ρ je jednak negativnom djelu *y*-osi. U $\theta\rho$-ravnini, na **Slika 11. b)**, svaka sinusoida predstavlja niz pravaca koji prolaze istom točkom u *xy*-ravnini. Sjecište dvaju sinusoida određuje parametre θ' i ρ' pravca koji prolazi kroz točke (x_i,y_i) i (x_j,y_j) u *xy*-ravnini.

Podjelom parametarskog prostora $\theta\rho$ u ćelije, gdje se svaka ćelija inkrementira za broj točaka koji se nalazi na istom pravcu (uz iste θ i ρ), može se značajno uštedjeti na broju računskih operacija. Za usporedbu uzmimo prvo metodu bez parametrizacije. Postoji *n* točaka, i cilj je pronaći skupine točaka koje leže na istim ravnim linijama. Tražeći sve linije određene parom točaka i skupove koji se nalaze na tim linijama iziskuje pronalaženju $n(n-1)/2 \sim n^2$ linija i $n(n(n-1)/2) \sim n^3$ usporedba svake točke sa svim linijama. Metoda sa parametrizacijom u polarnom koordinatnom sustavu i inkrementiranjem ćelija linearna je s obzirom na n [13].



Slika 11. a) parametrizacija pravca u *xy* ravnini b) normalni zapis u parametarskom prostoru c) podjela parametarskog prostora u akumulatorske ćelije

Za detekciju kružnica umjesto linija, kao parametarsku jednadžbu potrebno je umjesto jednadžbe pravca, koristiti jednadžbu kružnice

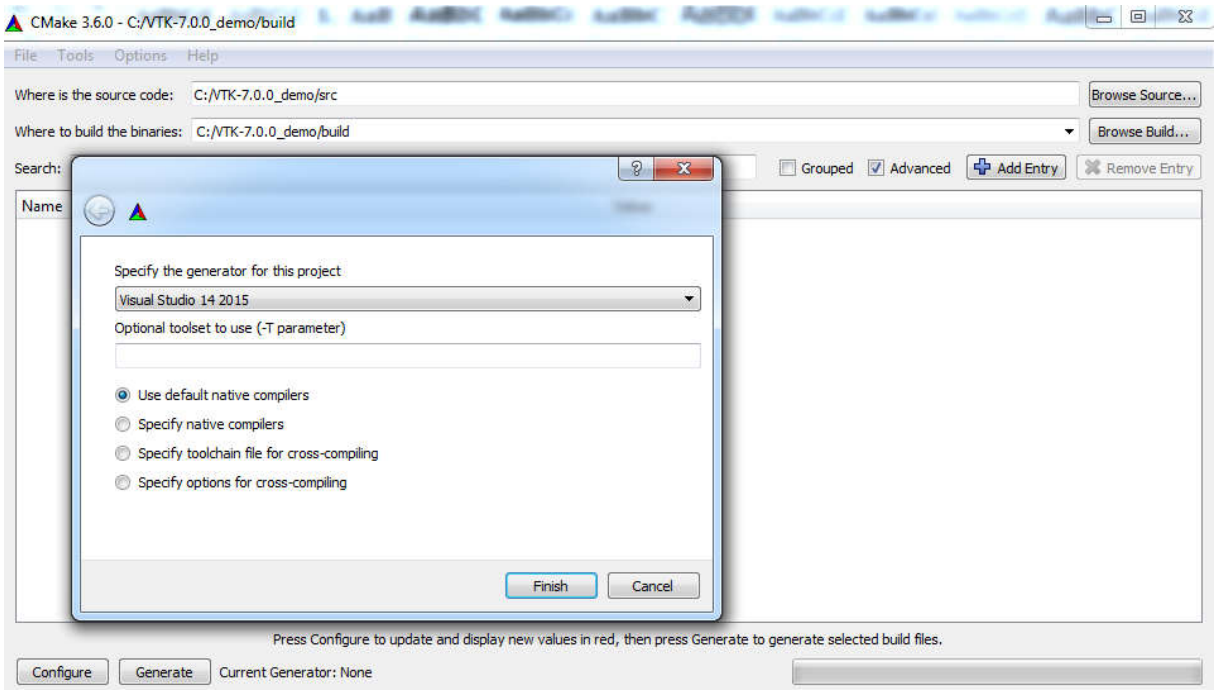
$$(x - c_1) + (y + c_2) = {c_3}^2. \tag{2.6}$$

Razlika je u tome što sada postoje tri parametra, a parametarski prostor je trodimenzionalan. Ćelije više nisu kvadrati, nego kocke.

3. KORIŠTENI ALATI - SOFTWARE

3.1. CMake

CMake je besplatan open-source softver korišten za upravljanje procesom izgradnje softvera neovisan o vrsti prevoditelja. Neovisan je o platformi i koristi na svim vrstama računala. Na Windows platformi koristi se zajedno sa Microsoft Visual Studio razvojnim okruženjem. Podržava hijerarhiju mapa i aplikacije koje ovise o implementaciji više knjižnica. Izvorni kod knjižnica VTK, vtkDICOM i openCV dolazi preveden samo za određene verzije operativnih sustava i razvojnih okruženja. Za ostale verzije Windows operativnog sustava i Visual Studio okruženja potrebno je preuzeti izvorni kod i izgraditi ga pomoću CMake-a. Gradnja knjižnice iz izvornog koda je strogo preporučljiva jer se prilikom izgradnje mogu uključiti dodatne opcije koje nije moguće modificirati u prevedenim knjižnicama. Jedna od njih je vtkDICOM bez koje nije moguće pristupiti meta-podacima snimaka.



Slika 12. Odabir prevoditelja prilikom izgradnje izvornog koda

Nakon preuzimanja zip arhive željene knjižnice ili programskog paketa, izvorni kod pohranjuje se u *src* mapu. Kao destinacija izgrađenog koda standardno se odabire *build* mapa. Sljedeći korak je odabrati generator i prevoditelj. U ovom slučaju to je bio VS 14 2015, a prevoditelje je preporučeno odabrati autotohtone kao na **Slika 12.** Slijedi konfiguracija izvornog koda nakon

koje se pojavljuju opcije za uključivanje raznih modula i opcija. Nakon odabira željenih postavki, izvorni kod se još jednom konfigurira. Generiranjem koda stvorena je .sln (*solution)* datoteka knjižnice ili programskog alata u mapi *build* koju je potrebno otvoriti u okruženju Visual Studio. Posljednji korak u gradnji koda je odabrati ALL_BUILD stavku hijerarhijskog stabla projekta i izvršiti naredbu *Build*. Ovakvim načinom instalacije programskih paketa i knjižnica izgrađeni kod je usko vezan za verziju razvojnog okruženja, tip arhitekture računala i ostale knjižnice integrirane u konačno rješenje. Samom promjenom verzije jedne od knjižnica može doći do nestabilnog ponašanja projekta, pojavljivanja grešaka i neizvršavanja samog koda. Preporuča se na jednom računalu ne imati više od jedne verzije programskih paketa i knjižnica. Rastom broja knjižnica i njihovih verzija, njihovo usklađivanje povećava se rizik od nestabilnog ponašanja projekta.

3.2. Visualization ToolKit

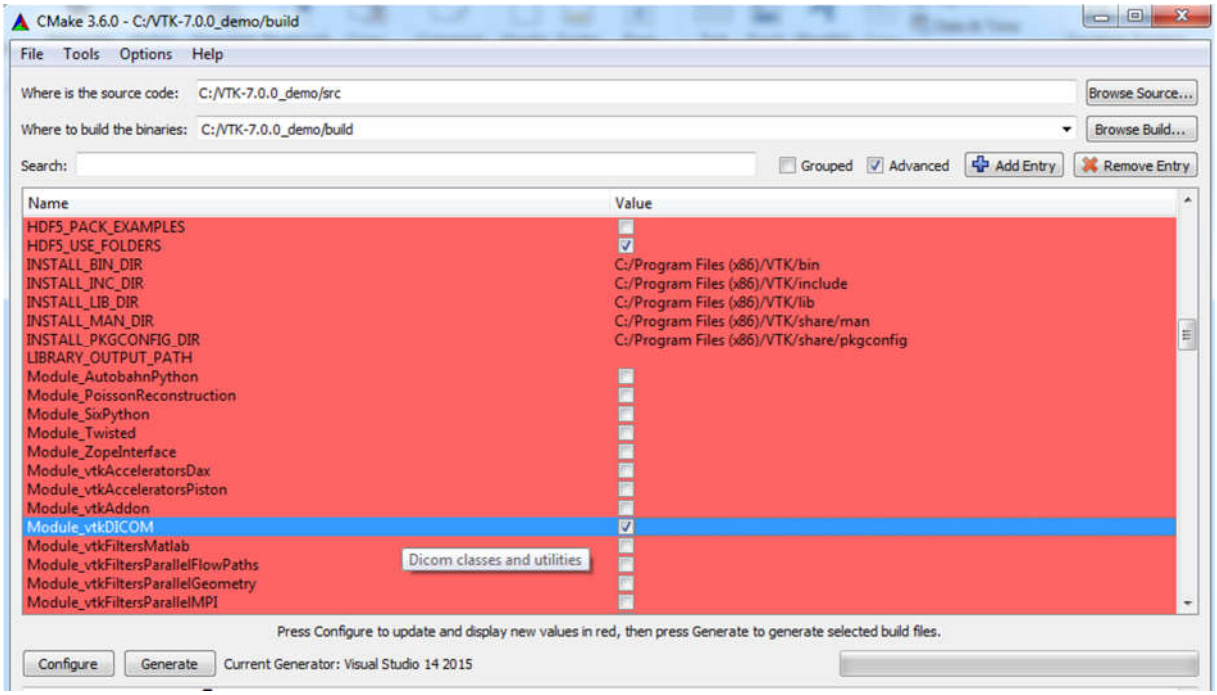
VTK je open-source objektno orijentiran softverski paket za računalnu grafiku, vizualizaciju i obradu slika. Izvorni kod koji nije preveden može se preuzeti sa github-a (https://github.com/Kitware/VTK). U ovom radu korištena je verzija 7.0 VTK knjižnice. Izvorni kod se gradi pomoću CMake-a kao što je opisano u prethodnom poglavlju. Nakon prve konfiguracije potrebno je uključiti opcije *Module_vtkTestingCore* i *Module_vtkTestingRendering*, prikazano na **Slika 13.**

Slika 13. Odabir modula prilikom gradnje izvornog koda VTK

Odabirom GDCM i vtkDICOM modula uključuju se njihove funkcionalnosti u VTK knjižnicu koje su opisane u nadolazećim poglavljima. Dovršavanjem procesa odabira modula i generiranjem koda, formira se *.sln* datoteka alata i u VS okruženju dovršava cijeli proces izgradnje koda.

3.3. vtkDICOM

Iako VTK knjižnica podržava klase za rad sa DICOM snimkama, njihova funkcionalnost je u nekim slučajevima ograničena. Klasa vtkDICOMImageReader može čitati DICOM snimke, ali ne i meta-podatke. vtkDICOM knjižnica dodaje niz klasa koje dodaju nove funkcionalnosti poput čitanja meta-podataka, opsežnijeg rada sa nizom snimaka, pisanja informacija u DICOM snimke i puna podrška za pristup svim informacijama DICOM snimaka unutar programa baziranog na VTK knjižnici. Instalira se kao modul zajedno sa VTK knjižnicom iz izvornog koda pomoću CMake programskog alata. Nakon konfiguracije izvornog koda potrebno je označiti opciju „Module_vtkDICOM“ i ponovo konfigurirati izvorni kod.

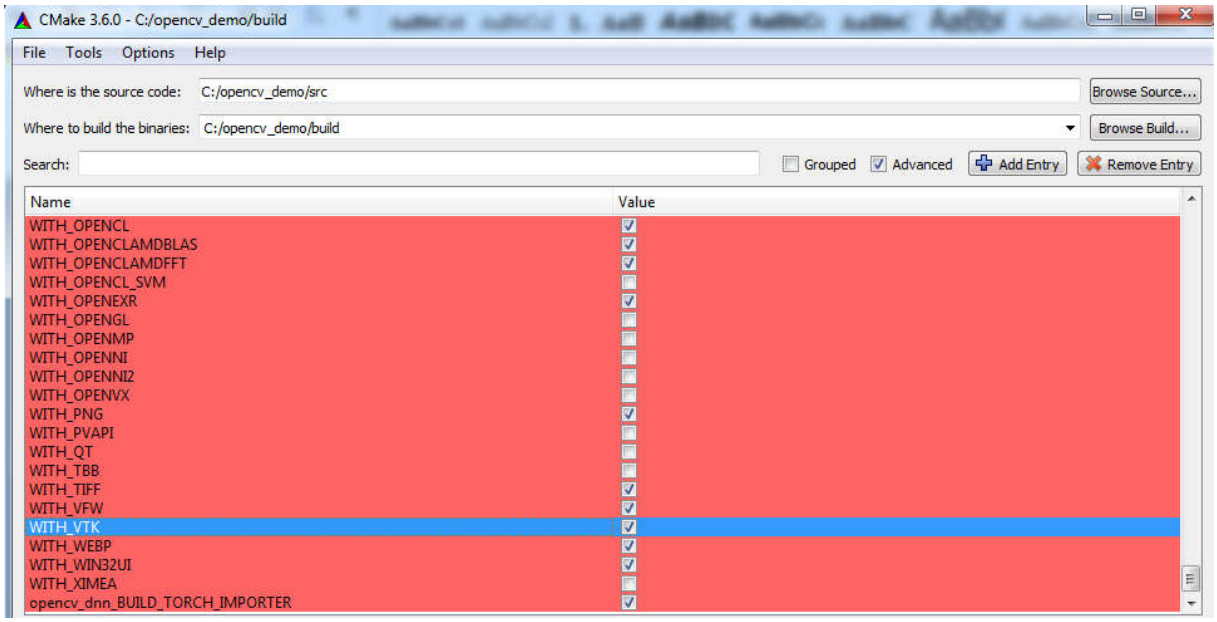


Slika 14. Odabir modula vtkDICOM pri konfiguriranju izvornog koda

Generiranjem koda i dovršavanjem procesa instalacije VTK knjižnice modul vtkDICOM biti će integriran u konačno rješenje.

3.4. OpenCV

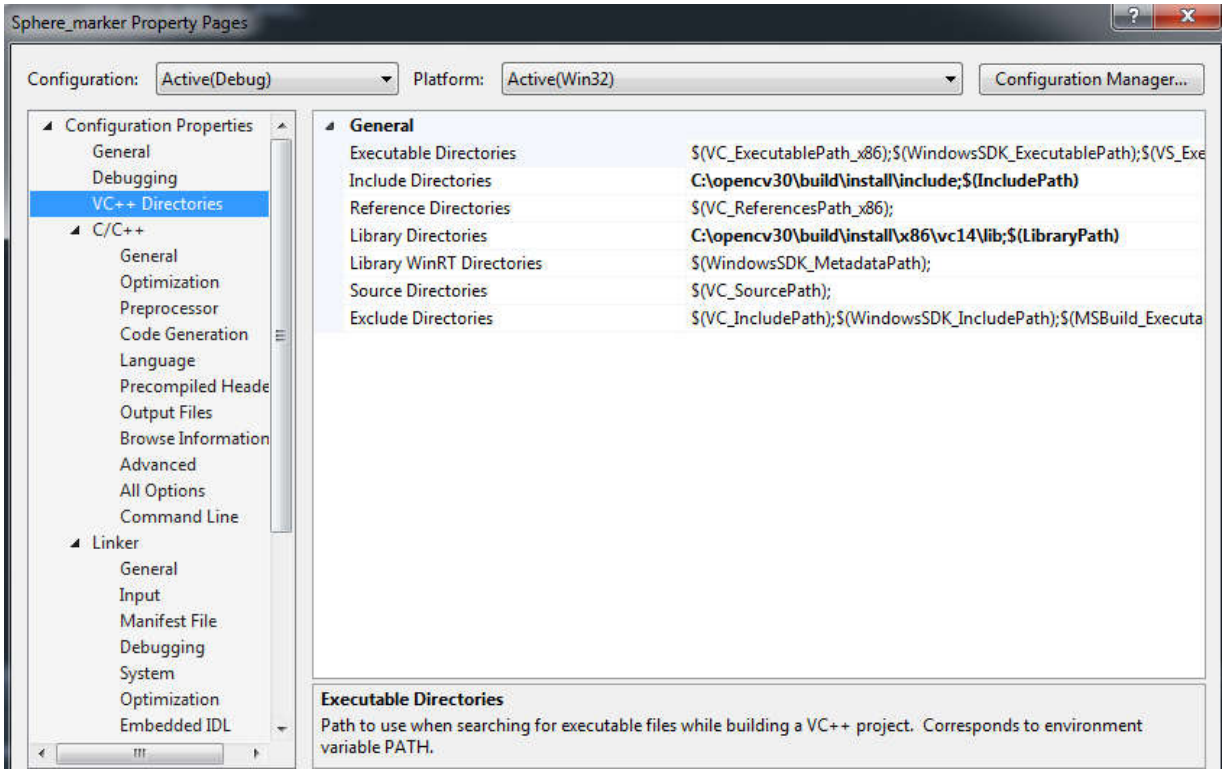
Open Source Computer Vision je knjižnica računalnog vida i strojnog učenja. Sadrži mnoštvo algoritama za detekciju i prepoznavanje objekata, lica i 3D modela. Dostupna na više operativnih sustava, sadrži sučelja za C++, Matlab, Python i Java programske jezike. Korištena je 3.0 verzija, također izgrađena iz izvornog koda kao i ostale knjižnice.



Slika 15. Integracija VTK knjižnice sa OpenCV knjižnicom

3.5. Microsoft Visual Studio

Microsoft Visual Studio je integrirano razvojno okruženje. Sadrži urednik koda, debugger i podržava velik broj programskih jezika. Korištena je verzija Microsoft Visual Studio 2015 Community koja je besplatna za korištenje i podržava većinu opcija. Microsoft Visual C++ prevoditelj je Microsoft-ova implementacija C++ prevoditelja. Prati ANSI C++ specifikacije i podržava mogućnosti C++11 standarda. Izvorni kod izgrađen CMake alatom prilagođen je prevoditelju i samoj verziji razvojnog okruženja. Sav programski kod napisan je u ovom alatu. Osim raznih mogućnosti uređivanja koda, neizostavne su funkcije izgradnje konačnog rješenja *.sln* datoteke koja je izlaz CMake alata. Objedinjuje sve knjižnice i programske alate u stranicama svojstava projekta.



Slika 16. Stranice svojstava projekta

Slijedi popis uvrštenih adresa dijelova knjižnica i programskih alata u stranice svojstava.

VC++ Directories:

- Include directories: …/OpenCV/build/install/include
- Library directories: …/OpenCV/build/install/x86/vc14/lib

C/C++:

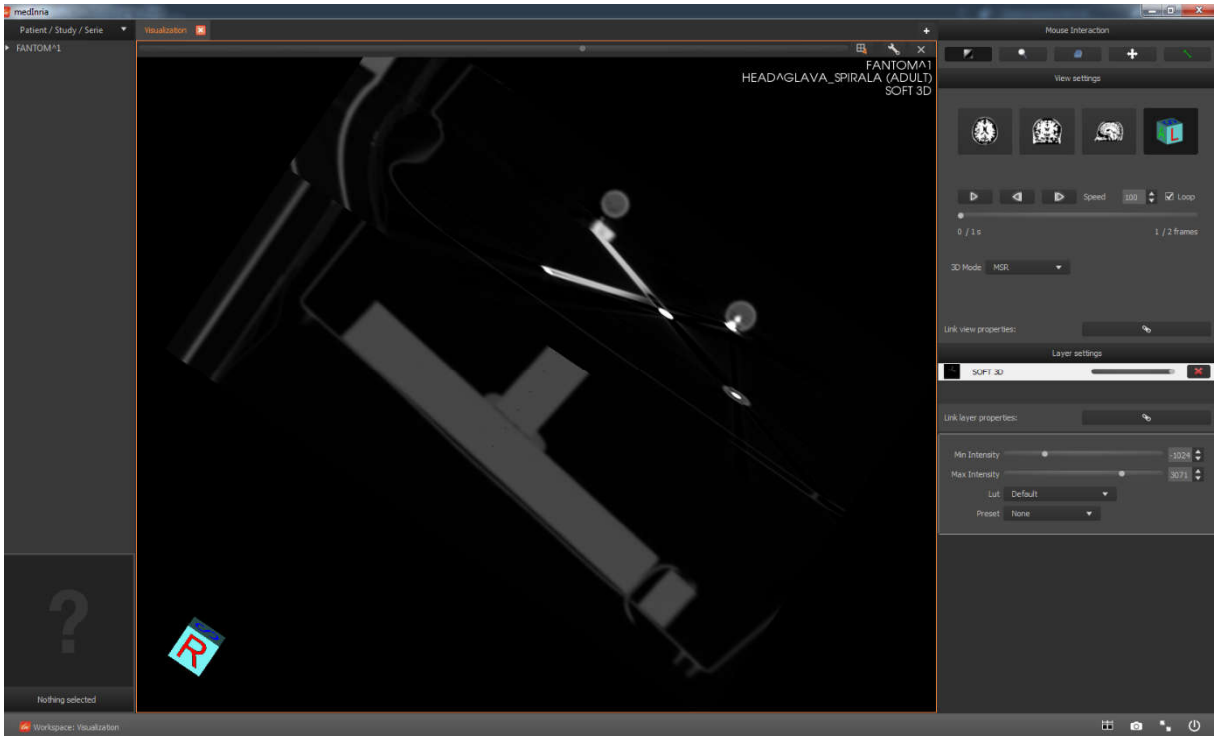
- Additional include directories: …/OpenCV/build/install/include
  …/VTK/build
  …/Eigen_source

Linker:

- Additional library directories: …/OpenCV/build/lib/Debug
- Additional dependencies: popis svih korištenih *.lib* datoteka VTK i OpenCV knjižnice

3.6. MedInria

MedInria je open-source softver za vizualizaciju i obradu medicinskih slika. Besplatan i jednostavan za upotrebu putem grafičkog sučelja, korišten je za provjeru orijentacije volumena snimaka, podešavanja i testiranje „windowing“ procesa i verifikaciju točaka u trodimenzionalnom prostoru.



Slika 17. Grafičko sučelje programa MedInria

Moguće je učitati cijeli volumen DICOM snimaka, odabrati željeni pogled i očitati bilo koju koordinatu.

3.7. Grassroots DICOM (GDCM)

Grassroots DICOM (skraćeno GDCM) je open-source knjižnica napisana u C++ programskom jeziku namijenjena radu sa DICOM snimkama. Omogućava funkcije dekompresije snimaka klasi čitača vtkDICOM knjižnice. Bez GDCM knjižnice moguće je raditi samo sa nekompresiranim snimkama. Kao i za ostale knjižnice preporuča se preuzimanje izvornog koda

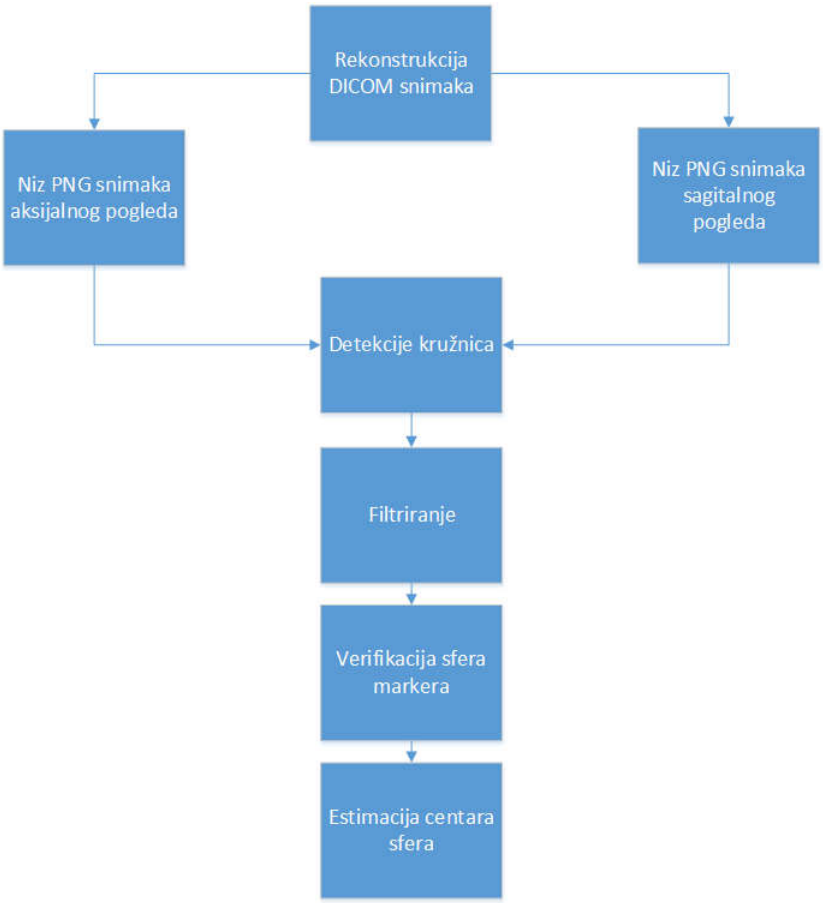
(https://github.com/malaterre/GDCM) i gradnja knjižnice preko CMake alata. Izgrađeni kod potrebno je referencirati *environment* varijablom u operativnom sustavu.

3.8. Eigen

Za rad sa matricama i vektorima korištena je knjižnica Eigen. Osim gotovih implementacija matrica i vektora, omogućava upotrebu velikog broja operacija linearne algebre poput invertiranja i transponiranja matrica, ali i dekompozicije matrica. Od velikog značaja je operacija singularne dekompozicije vrijednosti korištena kako bi se metodom najmanjih kvadrata dobila parametarska jednadžba trodimenzionalnog pravca koji prolazi koordinatama detekcija sfere. Unaprijed su definirane klase matrica i vektora manjih dimenzija, kod kojih je jednostavno inicijalizirati vrijednosti. Prilikom izvršavanja koda, podržani su i dinamički skalirani vektori i matrice što se pokazalo vrlo korisnim kod pohrane većeg broja koordinata. Instalacija Eigen knjižnice je jednostavna. Sve što je potrebno napraviti je preuzeti zadnje izdanje izvornog koda sa službene stranice (http://eigen.tuxfamily.org) i povezati projekt Visual Studia sa zaglavljima Eigen knjižnice. Zaglavlja se povezuju otvaranjem stranice svojstava projekta i navigacijom stabla stavkama *C/C++, General Page, Additional Include Directories* unosi se lokacija mape Eigen knjižnice na računalu.

4. RAZVOJ ALGORITMA

Niz snimaka učitan je klasom čitača vtkDICOM knjižnice. Potrebno je prilagoditi orijentacije snimaka pogledima definiranim DICOM standardom. U poglavlju 4.1 opisani su problemi standardne orijentacije snimaka učitanih VTK knjižnicom. Podešen je intenzitet i svjetlost svake pojedine snimke posutpkom *windowing*. Rekonstruirane DICOM snimke pohranjene su kao PNG slike na kojima su detektirane kružnice upotrebom OpenCV knjižnice.



Slika 18. Slijedni dijagram rada algoritma

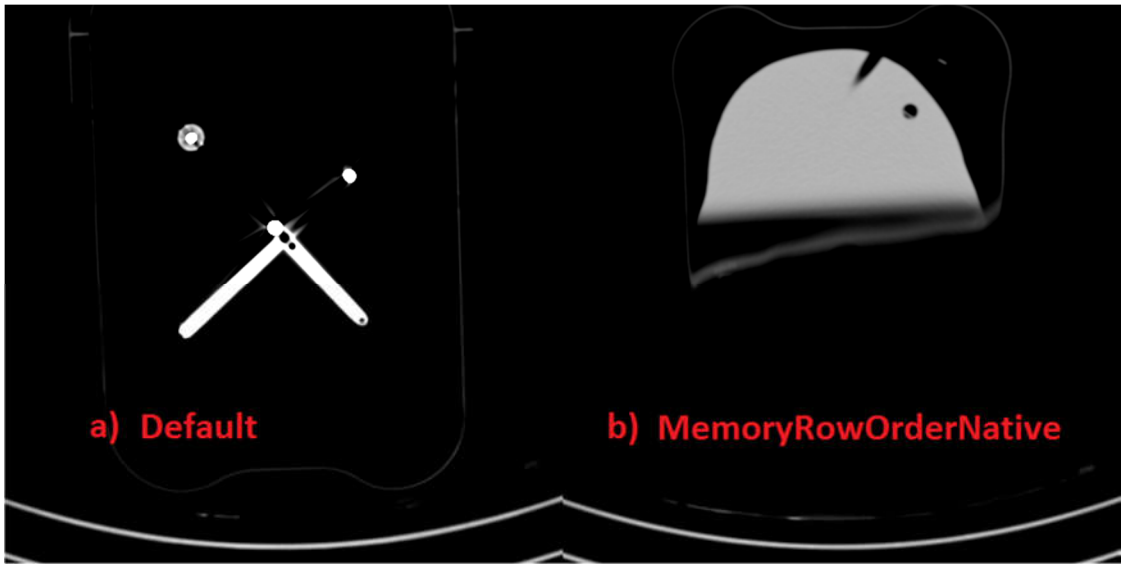
Skup detektiranih kružnica filtriran je metodama temeljenim na karakteristikama sfera markera. Detekcije su verificirane uvjetima euklidske udaljenosti. Iz skupa detekcija slijedi estimacija koordinata centara sfera metodom linearne regresije, **Slika 18.**

4.1. Orijentacija snimaka

Pojam orijentacije snimaka je višeznačan. Može biti riječ o:

- redoslijedu zapisivanja piksela, redova piksela ili samih snimaka u memoriju računala
- orijentaciji pojedinačnih snimaka u koordinatnom sustavu svijeta
- orijentaciji subjekta kada se snimke gledaju sa radne stanice

Budući da je klasa VTK knjižnice za prikazivanje slika *vtkImageViewer* implementirana na način da piksel u donjem lijevom uglu nosi najnižu adresu u memoriji, dolazi do sukoba sa DICOM standardom, u kojem najnižu adresu memorije nosi piksel gornjeg lijevog ugla. Klasa *vtkDICOMReader* knjižnice vtkDICOM za čitanje snimaka radi kompatibilnosti sa VTK knjižnicom standardno okreće sliku tokom učitavanja u memoriju.



Slika 19. Snimka aksijalnog pogleda broj 40 a) obrnuti redci piksela b) standardni način zapisivanja piksela

Da bi se spriječilo obrnuto zapisivanje redaka piksela slike potrebno je uključiti opciju *SetMemoryRowOrderToFileNative()*. U tom slučaju niz snimaka nije zapisan u smjeru *Inferior*, već smjeru *Superior* anatomske osi opisanih u poglavlju **2.1**.

Orijentaciju i položaj koordinatnog sustava svijeta može se dobiti *GetPatientMatrix*() funkcijom koja je dio *vtkDICOMReader* čitača. Izlaz funkcije je *vtkMatrix4x4* objekt koji

opisuje transformaciju koordinata zapisanih u *vtkImageData* objektu slike u koordinatni sustav svijeta definiran DICOM standardom. Sami iznosi koordinata ishodišta koordinatnog sustava svijeta ovise o medicinskom uređaju, tj. postavkama CT ili MR skenera. Matrica izlaza sastoji se od atributa *ImagePositionPatient* i *ImageOrientationPatient* očitanih iz meta-podataka niza DICOM snimaka. Prvi atribut sadrži *x,y,z* koordinate gornjeg lijevog ugla (centra prvog voksela) u milimetrima, dok drugi određuje matricu rotacije koordinatnog sustava. U slučaju da je redoslijed zapisivanja piksela obrnut, rotacijska matrica će radi kompenzacije biti promijenjena.

Sljedećim naredbama se direktno iz klase čitača mogu očitati informacije o dimenzijama snimke, razmaku između voksela i položaju ishodišta koordinatnog sustava niza snimaka. Razmak voksela po *z* koordinati nam ujedno i govori o razmaku između snimaka, koji se može dobiti i iz meta-podataka.

```
reader->GetOutputInformation(0)->Get(vtkStreamingDemandDrivenPipeline::
    WHOLE_EXTENT(), extent);
reader->GetOutput()->GetSpacing(spacing);
reader->GetOutput()->GetOrigin(origin);
```

„extent“ skupa podataka je niz od šest integer brojeva koji nam govore o indeksima prvih i zadnjih piksela za sve tri koordinatne osi. Za niz od 231 snimke 512 x 512 piksela, njihove vrijednosti iznose: 0, 511, 0, 511, 0, 230.

Varijabla „spacing“ vraća nam informaciju o udaljenosti točaka po sve tri koordinatne osi.

U slučaju prvog testnog fantoma ona iznosi; 0.484375mm u smjeru osi *x*, 0.484375mm u smjeru osi *y* i 0.7mm u smjeru osi *z*.

Zadnji parametar potreban za rekonstrukciju pogleda su koordinate ishodišta svih koordinatnih osi. Uobičajene vrijednosti koordinata ishodišta su 0.

Naredbe za proračun centra volumena:

```
center[0] = origin[0] + spacing[0] * 0.5 * (extent[0] + extent[1]);
center[1] = origin[1] + spacing[1] * 0.5 * (extent[2] + extent[3]);
center[2] = origin[2] + spacing[2] * 0.5 * (extent[4] + extent[5]);
```

Sada se pomoću matrica transformacije može rotirati pogled oko centra volumena.

Nakon što su matrice transformacije definirane, radi se kopija matrice transformacije željenog pogleda sa uvrštenim koordinatama centra volumena u četvrtom stupcu koji određuje translaciju.

```
vtkSmartPointer<vtkMatrix4x4> resliceAxes =
    vtkSmartPointer<vtkMatrix4x4>::New();
  resliceAxes->DeepCopy(sagittalElements);
  resliceAxes->SetElement(0, 3, center[0]);
  resliceAxes->SetElement(1, 3, center[1]);
  resliceAxes->SetElement(2, 3, center[2]);
```

Izlaz čitača snimaka u cjevovodu se spaja na ulaz vtkImageReslice filtera. Jedan je od temelja svih vrsta geometrijskog filtriranja VTK knjižnice, nudi efikasno uzorkovanje slika duž proizvoljno odabranog volumena ili slike, a najčešće se koristi za:

- primjenu jednostavnih rotacija, skaliranja i rotacija na snimkama
- usklađivanje uzorkovanja voksela između različitih snimaka, za svrhu uspoređivanja ili spajanja
- izvlačenje snimaka iz volumena, sa mogućnošću definiranja kutova matrice transformacije i dobivanja proizvoljnih pogleda putem klase SetResliceAxesDirecionCosines()

Kao izlaz vtkImageReslice filtera može se dobiti snimka ili volumen. Algoritam je implementiran na način da obrađuje svaku snimku zasebno pa je potrebno postaviti SetOutputDimensionality(2).

```
  vtkSmartPointer<vtkImageReslice> reslice =
    vtkSmartPointer<vtkImageReslice>::New();
  reslice->SetInputConnection(reader->GetOutputPort());
  reslice->SetOutputDimensionality(2);
  reslice->SetResliceAxes(resliceAxes);
  reslice->SetInterpolationModeToLinear();
```

U sljedećem koraku se preko pointera direktno pristupa matrici transformacije, te za svaku snimku pojedinačno mijenja vrijednost koordinate centra u smjeru osi ovisno o odabranom

anatomskom pogledu. U slučaju aksijalne ravnine, centar volumena snimaka pomiče se u smjeru *x* osi.

```
vtkMatrix4x4 *matrix = reslice->GetResliceAxes();
matrix->SetElement(0, 3, i*slice_val);
```

Koordinata centra volumena se ažurira preko elementa prvog retka i četvrtog stupca. Ovisi o broju snimke, pa je njen raspon od prve do zadnje koordinate volumena u smjeru osi *x*. Postavljena vrijednost koordinate ovisi o broju snimke *i* i vrijednosti *slice_val* koja glasi:

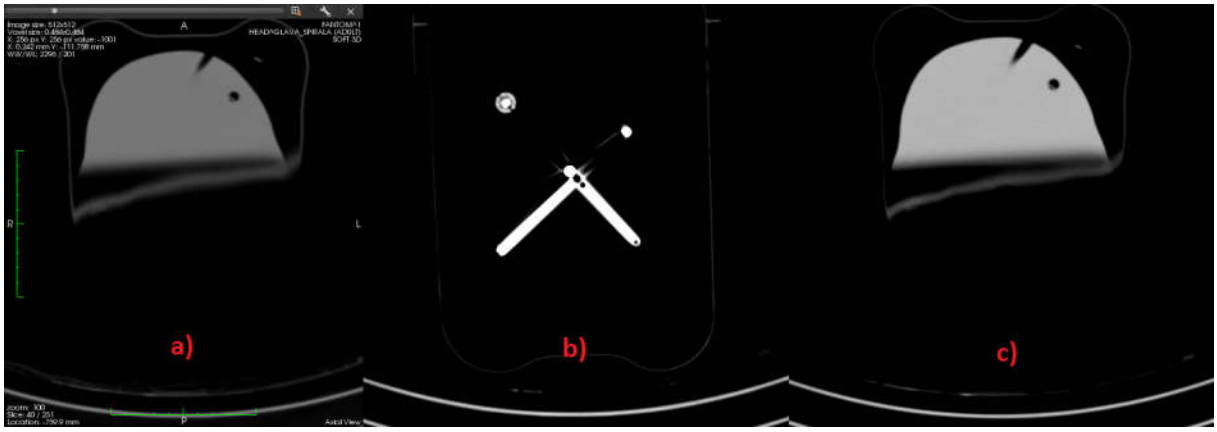
```
double slice_val = (center[0]*2 / num_values );
```

gdje varijabla *num_values* određuje broj snimaka direktno očitan iz čitača.

Provjera i testiranje orijentacije, razine prozora i koordinata odrađena je u programu *medInria*. Standardni pogled niza snimaka je aksijalni, što znači da je matricija transformacije za aksijalni pogled jedinična matrica:

```
static double axialElements[16] = {
           1, 0, 0, 0,
           0, 1, 0, 0,
           0, 0, 1, 0,
           0, 0, 0, 1 };
```

Sa standardnim radom vktDICOMReader klase, dobiva se ispravna orijentacija snimaka, međutim redoslijed očitavanja snimaka započinje vrhom fantoma lubanje, a ne dnom kao što je propisano DICOM standardom. Na **Slika 20. a)** prikazana je snimka rednog broja 40 u aksijalnom pogledu na kojoj se vidi baza fantoma. Na snimci broja 40 VTK knjižnice, **Slika 20. b)**, nalaze se sferni markeri, što ukazuje na prethodno navedeno izvrtanje zapisa DICOM snimaka kao posljedica implementacije *vtkDICOMReader* klase čitača snimaka.



Slika 20. Obrnuti redoslijed zapisivanja snimaka

Kada se uključi opcija nativnog zapisivanja piksela, sa jediničnom matricom transformacije nije moguće dobiti istu orijentaciju snimke u aksijalnom pogledu kao na **Slika 20. c)**, te je matricu transformacije potrebno modificirati. Ona u tom slučaju glasi:

```
static double axialElements[16] = {
        1, 0, 0, 0,
        0,-1, 0, 0,
        0, 0,-1, 0,
        0, 0, 0, 1 };
```

Matrica sagitalnog pogleda je umnožak matrice transformacije aksijalnog pogleda, rotacije oko *x* osi za -90° i rotacije oko *z* osi za 90°:

$$R_z R_x R_{aks} = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{4.1}$$

te sada glasi:

```
static double sagittalElements[16] = {
        0, 0, 1, 0,
        1, 0, 0, 0,
        0, 1, 0, 0,
        0, 0, 0, 1 };
```

dok je matrica koronalnog pogleda:

```
static double coronalElements[16] = {
        1, 0, 0, 0,
        0, 0, 1, 0,
        0,-1, 0, 0,
        0, 0, 0, 1 };
```

4.2. Windowing

Windowing je proces poznat i kao rastezanje kontrasta, modifikacija histograma ili mapiranje sivog nivoa. U CT snimkama sivi tonovi komponenata snimki mogu se manipulirati mijenjanjem parametara prozora (*eng. window)* i razine *(eng. level)* kako bi se naglasile određene strukture na slici. Svjetlost slike podešava se preko parametra prozora, a kontrast putem parametra razine. Oba parametra dio su Hounsfieldove ljestvice (drugi naziv CT brojevi) koja kvantitativno opisuje nemogućnost probijanja materijala elektromagnetskom radijacijom (X-zrakama u ovom slučaju). Hounsfieldova ljestvica linearna je transformacija izmjerenog koeficijenta zasićenja μ_x u vokselu sa koeficijentima zasićenja vode i zraka [14]:

$$HU = \left(\frac{\mu_x - \mu_{vode}}{\mu_{vode} - \mu_{zraka}}\right) \times 1000 \qquad (4.2)$$

Proizvoljno odabrane mjere radiozasićenosti postavljene kao reference su voda sa 0 HU, te zrak sa vrijednosti -1000 HU. Vidljivost internih struktura unutar tijela moguće je dodatno pojačati radiokontrastnim agentima. Intravaskularni agenti su derivati joda, dok su gastro-intestinalni derviati barija.

```
vtkSmartPointer<vtkImageMapToWindowLevelColors> windowlevel =
vtkSmartPointer<vtkImageMapToWindowLevelColors>::New();
windowlevel->SetInputConnection(reader->GetOutputPort());
windowlevel->SetWindow(1500);
windowlevel->SetLevel(300);
windowlevel->Update();
```

Klasa *vtkImageMapToWindowLevelColors* ima ulogu filtera koji sliku bilo kojeg tipa skalara na ulazu preko tablice za pretraživanje (*eng. lookup table)* pretvara u sliku tipa podataka vtk_unsigned_char.

Alternativni način postavljanja vrijednosti raspona tablice za pretraživanje je definiranje klase *vtkLookupTable*, gdje postoji više opcija. Korištena je iz razloga što je metoda fleksibilnija i postoji više dokumentacije o njenoj upotrebi.

```
vtkSmartPointer<vtkLookupTable> table =
  vtkSmartPointer<vtkLookupTable>::New();
table->SetRange(300, 1500); // image intensity range
table->SetValueRange(0.0, 1.0); // from black to white
table->SetSaturationRange(0.0, 0.0); // no color saturation
```

```
table->SetRampToLinear();
table->Build();
```

Klasa `vtkImageMapToColors` je filter koji primjenjuje ranije definiranu tablicu klase *vtkLookupTable*. Kao ulaz filtera postavlja se izlaz ažuriranog filtera *reslice*.

```
vtkSmartPointer<vtkImageMapToColors> color =
vtkSmartPointer<vtkImageMapToColors>::New();
color->SetLookupTable(table);
color->SetInputConnection(reslice->GetOutputPort());
```

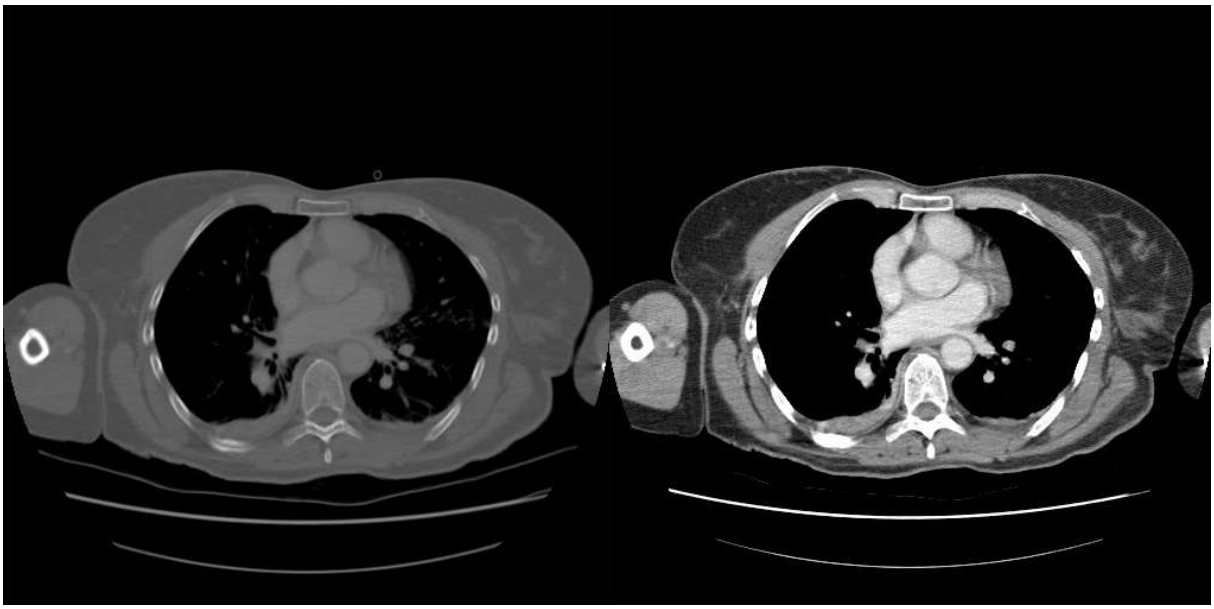
Snimke CT skenera su crno-bijele, pa je raspon vrijednosti od 0 (crne boje) do 1 (bijele boje). Kako nisu korištene ostale boje, zasićenje boja nije potrebno, te je postavljeno na 0. Rampom se može odrediti linearni ili nelinearni način mapiranja vrijednosti iz tipa skalar u tip podataka vtk_unsigned_char.

Postupak „windowing“ često se koristi za bolje raspoznavanje određenih struktura. Postoje određene skupine raspona prozora i razina preporučene za različite pretrage.

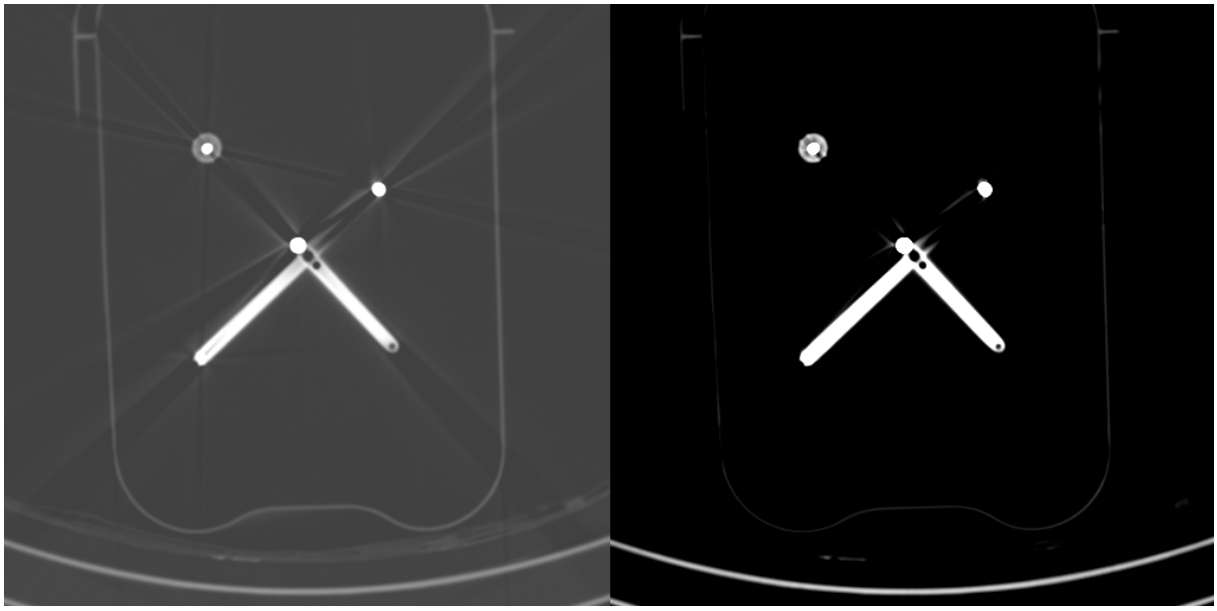
Preset	WL	WW
CT Abdomen	60	400
CT Angio	300	600
CT Bone	300	1500
CT Brain	40	80
CT Chest	40	400
CT Lungs	-400	1500

Tablica 1. Preporučeni rasponi vrijednosti parametara razina i prozora

Za prepoznavanje retro-refleksivnih sfera markera korištene su preporučene vrijednosti razine i prozora za kosti, WL 300, WW 1500.



Slika 21. Usporedba CT snimke trupe sa različitim parametrima prozora i razine a) WL 300 / WW 2000 b) WL 40 / WW 400



Slika 22. Usporedba CT snimki fantoma lubanje sa različitim parametrima prozora i razine a) WL -1000 / WW 3000 b) WL 300 / WW 1500

Nakon primjene matrice transformacije i *windowing* procesa, slika je iz DICOM formata pretvorena u PNG format koji je obradiv openCV knjižnicom računalnog vida.

```
vtkSmartPointer<vtkPNGWriter> writer =
vtkSmartPointer<vtkPNGWriter>::New();


const char * writer_in = filename.c_str();
writer->SetInputConnection(color->GetOutputPort());
writer->SetFileName(writer_in);
writer->Write();
```

4.3. Detekcija kružnica

Ulaz funkcije HoughCircles je crno bijela slika (*grayscale*). Izlaz funkcije, vektor *circles* pohranjuje vektore parametara detektiranih kružnica u obliku (*x,y,r)* tj. x, y koordinata i radiusa. Koordinate *x* i *y* su koordinate lokalnog koordinatnog sustava slike.

```
cv::HoughCircles(img, circles, cv::HOUGH_GRADIENT, 1, img.rows / 6, 255, 14, 8, 14);
```

Gornja granica vrijednosti Canny algoritma za prepoznavanje rubova je postavljena na maksimalnu vrijednost 255. Smanjivanjem ove vrijednosti broj detekcija kružnica se povećava, ali broj detekcija kružnica koje predstavljaju sfere je jedanak što znači da bi se povećao broj lažnih detekcija što je nepoželjno. Slijedi prag za detektiranje broja kružnica. Određena je eksperimentalna vrijednost 14.

Zadnja dva parametra definiraju minimalni i maksimalni radijus kružnice. Promjer retroreflektivne sfere markera je 11.7 mm. Vrijednost radijusa kružnice u funkciji HoughCircles unosi se kao broj piksela, pa je za vrijednost maksimalnog polumjera kružnice radijus sfere podijeljen razmakom između piksela.

Svaka snimka se obrađuje zasebno. Detektirane kružnice iz vektora *circles* izdvojene su i vektori njihovih parametara se proširuju trećom koordinatom. U tom procesu koristi se međukorak gdje je svaka kružnica pohranjena kao skalar sa tri parametra. Nove vektore sa četiri elementa je moguće inicijalizirati sa vrijednostima tog skalara.

```
cv::Vec3i c = circles[n]; // this with conversion to have X,Y,R as int
cv::Scalar pokus = c; // vec3i -> scalar -> vec4f
cv::Vec4f hokus = pokus; // no suitable conversion from vec3i to vec4i exists
```

U idućem koraku se dodjeljuje posljednji parametar, ovisno o pogledu.

```
if (view == "axial")
{
	pokus[3] = slice_thick*i;		}
else if (view == "sagital")
{
	pokus[3] = ((extent[0] + extent[1] +1) / num_values )*i ;
}
```

U aksijalnom pogledu se iz razmaka između snimaka, tj. atributa meta-podataka *SliceThickness* dobiva četvrti parametar, *z* koordinata.

U sagitalnom pogledu to je *x* koordinata izračunata množenjem broja snimke sa omjerom raspona *x* koordinate i ukupnog broja snimaka.

Sve detektirane kružnice sa pripadajućim koordinatama *x,y,z* centra i radijusom pohranjene su u vektor vektora. U primjeru sa **Slika 23.** to je vektor *circles_detected*. Kako se radi o velikom broju detekcija, potrebno je filtrirati kružnice kako bi se pronašle četiri srodne skupine detekcija. Cilj je iz svih detekcija kružnica izdvojiti one detekcije koje pripadaju sferama markera i grupirati ih u grumene. Svaki grumen predstavlja jednu sferu. Cilj je bio iz svakog grumena, nekom od metoda aproksimacije dobiti koordinate središta sfere markera.

```
circles_detected element 300 [153.063, 74.5938, 8, 144.2]
circles_detected element 301 [157.906, 137.563, 10, 144.2]
circles_detected element 302 [80.4063, 142.406, 12, 144.9]
circles_detected element 303 [157.906, 137.563, 11, 144.9]
circles_detected element 304 [153.063, 74.5938, 7, 144.9]
circles_detected element 305 [80.4063, 142.406, 13, 145.6]
circles_detected element 306 [157.906, 137.563, 11, 145.6]
circles_detected element 307 [157.906, 137.563, 12, 146.3]
circles_detected element 308 [80.4063, 142.406, 13, 146.3]
circles_detected element 309 [157.906, 137.563, 12, 147]
circles_detected element 310 [80.4063, 142.406, 13, 147]
circles_detected element 311 [157.906, 137.563, 13, 147.7]
circles_detected element 312 [80.4063, 142.406, 12, 147.7]
circles_detected element 313 [157.906, 137.563, 13, 148.4]
circles_detected element 314 [80.4063, 142.406, 12, 148.4]
circles_detected element 315 [157.906, 137.563, 13, 149.1]
circles_detected element 316 [80.4063, 142.406, 11, 149.8]
circles_detected element 317 [157.906, 137.563, 12, 149.8]
circles_detected element 318 [157.906, 137.563, 11, 150.5]
circles_detected element 319 [79.4375, 142.406, 10, 150.5]
circles_detected element 320 [157.906, 137.563, 11, 151.2]
circles_detected element 321 [80.4063, 142.406, 8, 151.2]
circles_detected element 322 [157.906, 137.563, 10, 151.9]
circles_detected element 323 [80.4063, 142.406, 6, 151.9]
circles_detected element 324 [157.906, 137.563, 9, 152.6]
circles_detected element 325 [157.906, 137.563, 7, 153.3]
```

Slika 23. Detekcije svih kružnica u aksijalnom pogledu

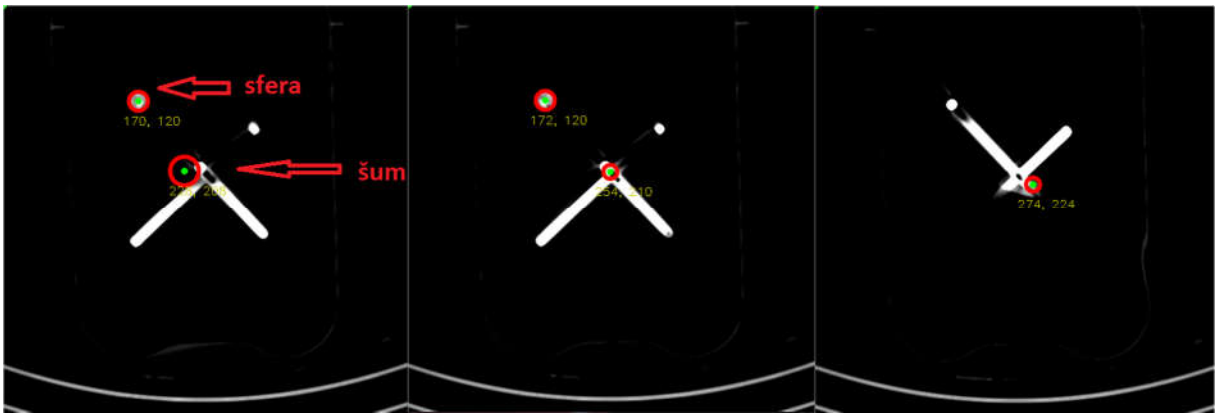
4.4. Filtriranje detekcija kružnica

4.4.1. Sekvencijalno filtriranje

Ako se prolazi kroz volumen snimaka nekim redom, bez obzira o kojem je pogledu riječ, sfera markera očitati će se kao kružnica. Nakon prve detekcije velika je vjerojatnost da će se sfera detektirati u određenom broju nadolazećih snimaka. Iz tog razloga postavljen je prvi uvjet koji eliminira sve detekcije kružnica koje se ne pojavljuju u prethodnih i sljedećih *n* snimaka. Varijabla *n* je testirana sa vrijednosti 3 i 5. Broj detekcija 5 pokazao se nešto boljim za rad algoritma jer se time spriječava gubitak detekcija sfere markera (u slučaju da su detektirane sve četiri kružnice sfera i šumovi).

4.4.2. Ograničena varijabilnost koordinati središta

Idući korak filtriranja bila je eliminacija detekcija kružnica koje se pojavljuju sa oscilacijama obje koordinate većim od 10 piksela. Detekcije sfera često nisu savršene, no razlike u koordinatama njihovih središta su nekoliko piksela, najčešće po samo jednoj kordinati. Provjerom po obje koordinate i postavljanjem navedenog uvjeta spriječilo se dodavanje takvih detekcija u grumene.



Slika 24. Neželjene detekcije i oscilacije u njihovim koordinatama

4.4.3. Filtriranje po grumenima

Slijedi inicijalizacija gnijezda vektora sa vrijednosti nula. U prvoj razini gnijezda nalaze se grumeni, u drugoj elementi grumena i posljednjoj, parametri pojedinih detekcija kružnica. Vektori u C++ programskom jeziku u ovakvim primjenama imaju prednost nad ostalim strukturama podataka jer su dinamički objekti koji se mogu skalirati prilikom izvršavanja programa. Njihovim elementima se može pristupiti preko pointera, koji također služe kao iteratori u petljama.

```
std::vector<std::vector<cv::Vec4f>> clusters_detected(100, std::vector<cv::Vec4f>{0});
```

Iz vektora svih detektiranih kružnica, provjerava se postoji li već slična detekcija u vektoru grumena. Pod slična, misli se na onu koja se ne razlikuje od trenutne za više od 20 piksela u koordinatama lokalnih *x,y* osi. Kao što je prethodno navedeno, eksperimentalno je utvrđeno da detekcije središta sfera ne variraju više od nekoliko piksela. Vrijednost od 20 piksela odabrana je iz razloga što se na taj način šumovi mogu efikasnije grupirati u grumene koji će kasnije biti eliminirani. Ukoliko su detekcije slične, radi se dodatna provjera po trećoj prostornoj koordinati, te ukoliko postoji sličnost i u ovom slučaju, potvrđuje se postojanost grumena za tu detekciju, detekcija dodaje elementima grumena i petlja nastavlja dalje.

```
if (abs(circles_axial_seq[i][0] - cluster_axial[m][0][0]) < 20 // if there exists a
circle like this in a cluster
&& abs(circles_axial_seq[i][1] - cluster_axial[m][0][1]) < 20)
if ((abs(circles_axial_seq[i][3] - cluster_axial[m][n][3]) < 30))
        push_cluster = false;
        break;
```

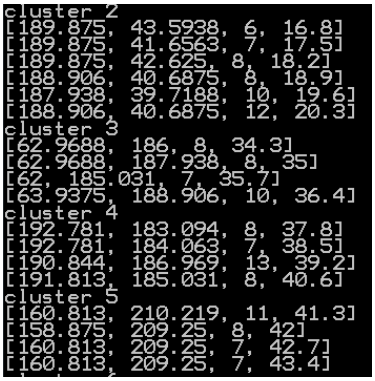
Provjera po trećoj koordinati lokalnog koordinatnog sustava sprječava da se detekcije sličnih koordinata *x,y* iz drugog dijela volumena snimaka ne bi pohranile u isti grumene.

Ako ne postoji slična detekcija u razini gnijezda sa grumenima, inicijalizira se novi grumen kružnica. Tom grumenu dodaju se sve detekcije približno istih koordinata. Idućim prolaskom kroz petlju, detektira se postojanje novog grumena i spriječava dupliciranje istih grumena.

Bitno je inicijalizirati najnižu razinu gnijezda vektora na neku vrijednost, kako bi bilo moguće izvršiti usporedbu koordinata jer C++ programski jezik ne dodjeljuje automatski vrijednost varijablama.

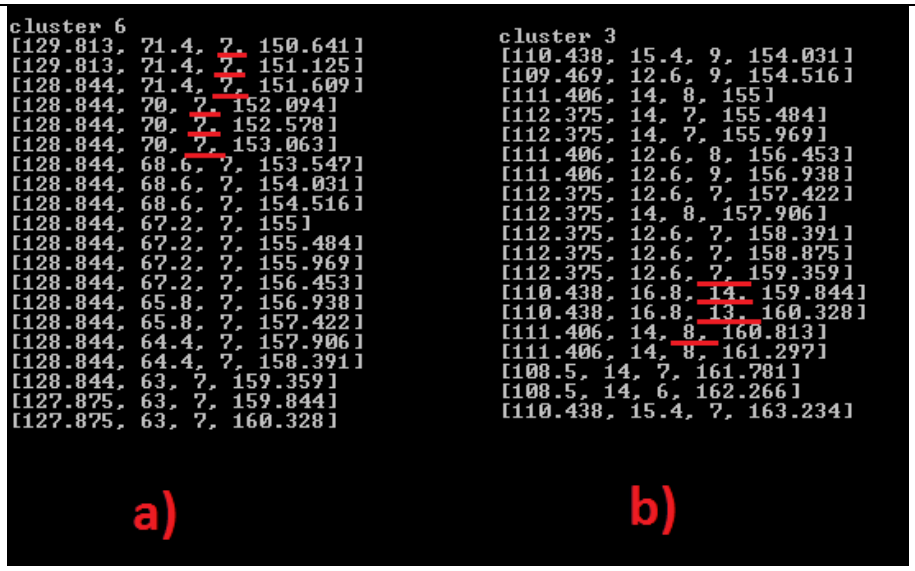
Završavanjem petlje i kreiranjem svih grumena, gnijezdo vektora prilagođava veličinu njihovom broju. Sam broj grumenova znatno varira, ovisno o postavkama parametara funkcije *HoughCircles* za detekciju kružnica.

Promjer retro reflektivne sfere je 11.7mm, dok je razmak između aksijalnih snimaka 0.7mm što rezultira očekivanjem više od 10 detekcija u jednom grumenu kružnica sfere. Ukoliko nije detektirano niti skromnih 10 detekcija svake sfere, potrebno je povećati osjetljivost detektiranja kružnica jer velik broj kružnica sfere markera nije detektiran. Iz toga proizlazi novi korak filtriranja, te su svi grumeni sa manje od 10 detekcija eliminirani iz gnijezda vektora.



Slika 25. Grumeni šumova sa manje od 10 sličnih detekcija

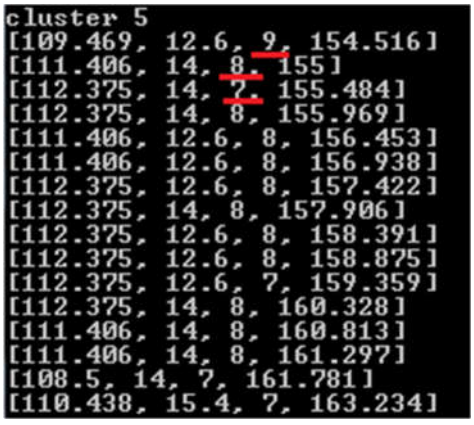
Jedna od karakteristika detektiranih kružnica sfere je što imaju promjenjiv radijus. Nagomilani šumovi mogu biti konstantnog radijusa ili razlike radijusa kroz cijeli grumen iznose tek jedan do dva piksela. Postavljen je dodatni uvjet na grumene kružnica koji provjerava konstantnost radijusa. U slučaju da ne postoji razlika radijusa od 3 piksela između dvije kružnice unutar iste grupe detekcija, grumen je izbrisan.



Slika 26. a) konstantnost radijusa u cijelom grumenu b) promjenjiv radijus detekcije sfere markera

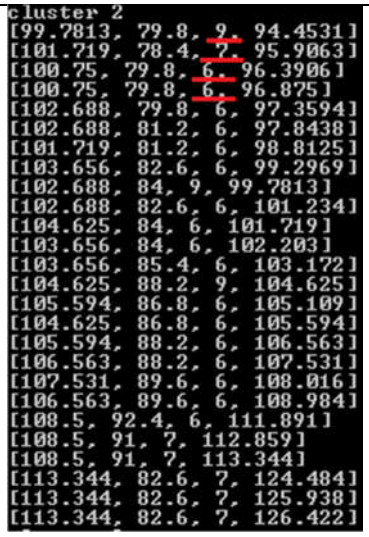
Primjer sa **Slika 26. a)** jasno upućuje da se ne radi o sferi markera uslijed konstantnosti radijusa. Da je uistinu riječ o detekcijama markera, konstantan radijus upućivao bi na velik broj neispravnih detekcija, koje se može minimizirati optimalnim odabirom parametara funkcije *HoughCircles* poput osjetljivosti i kriterija detekcija te kontrasta i svjetlosti.

Neusklađenost navedenih parametara također može dovesti do smanjenog broja detekcija sfere markera. Rezultat toga nije samo smanjena točnost registracije konačnih koordinata središta markera. Primjenom ranije opisanih metoda filtriranja takav grumen pogrešno se klasificira u šum. Eliminacija istog tada vodi do neispravnog rada cijelog algoritma. Kako bi se to izbjeglo, pristup u ovom radu bio je korištenje visoke osjetljivosti detekcija kružnica.



Slika 27. Grumen sfere sa nedovoljnim brojem detekcija vodi pogrešnoj klasifikaciji

Očekivano, porastom broja detekcija kružnica sfera raste i broj neželjenih detekcija. Kod vrlo velikog broja šumova, unatoč filtriranju moguće je njihovo formiranje u grumene, primjer na **Slika 28.** vidljive su relativno niske oscilacije koordinata središta i promjenjiv radijus.



Slika 28. Grupa detekcija kružnica sa parametrima u mm

Informacija koja odaje grupu kao šum je koordinata *z* lokalnog koordinatnog sustava, smještena u četvrtom stupcu. Razlika iznosa prve i posljednje koordinate iznosi više od 30mm, što nikako nije karakteristično za sferu promjera 11.7 mm.

Kao mjera opreza, uvedena je pretpostavka da neki šumovi mogu biti klasificirani kao sfera zajedno sa ispravnim detekcijama. Gornja granica razlike prve i posljednje *z* koordinate je time postavljena na 20mm.

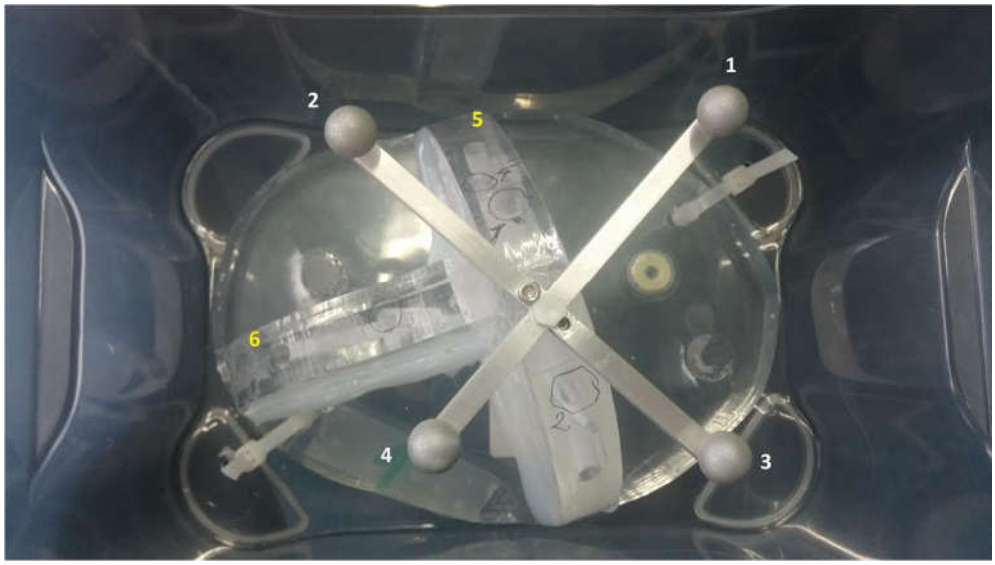
4.5. Euklidska udaljenost i metoda glasanja

Zadnji proces filtriranja temeljen je na euklidskim udaljenostima retro-refleksivnih sfera koje su poznate iz konstrukcije markera sa **Slika 29.** Euklidska udaljenost između dvije točke u trodimenzionalnom Kartezijevom koordinatnom sustavu glasi:

$$d_{i,j} = \sqrt{\left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2 + \left(z_i - z_j\right)^2}, \tag{4.3}$$

gdje su *x,y* i *z* koordinate središta kružnice.

Iz tih vrijednosti, određen je raspon euklidskih udaljenosti temeljen na minimalnoj i maksimalnoj udaljenosti sfera, proširen za nekoliko piksela radi tolerancije nesavršenih detekcija. Raspon je relativno širok, pa postoji mogućnost da i grumen detekcija šumova zadovolji uvjet pripadnosti sa nekim od grumena sfera.



Slika 29. Konstrukcija sfernog markera pričvršćenog na fantom lubanje

Inicijaliziran je vektor glasova povezan sa gnijezdom grumena i zamišljen proces glasanja. Mjeri se udaljenost između grumena koordinata x_i, y_i, z_i i ostalih grumena koordinata x_j, y_j, z_j .

Stvara se novi vektor sa kandidatima za konačne detekcije sfera gdje su smješteni grumen *i* i grumeni *j* koji zadovoljavaju uvjet Euklidske udaljenosti. Sada se unutar tog vektora provodi glasanje između kandidata, te svaki kandidat koji zadovoljava uvjet dobiva glas.

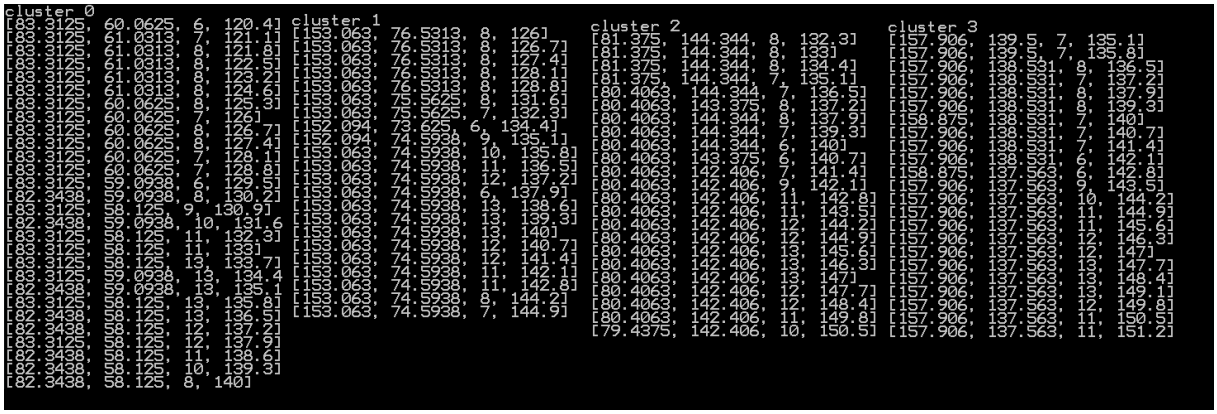
```
# of Euclid_matches: 3

i: 3
vote of 3. element: 3
i: 2
vote of 2. element: 3
i: 1
vote of 1. element: 3
i: 0
vote of 0. element: 3
sphere marker elem 0 :[83.3125, 60.0625, 6, 120.4]
sphere marker elem 1 :[153.063, 76.5313, 8, 126]
sphere marker elem 2 :[81.375, 144.344, 8, 132.3]
sphere marker elem 3 :[157.906, 139.5, 7, 135.1]
FOUND 4 SPHERES!
```

Slika 30. Četiri kandidata odabrana glasanjem

Grumen koji pripada detekcijama sfere dobiti će minimalno tri glasa od ostalih sfera. Grumen sa manje od tri glasa, klasificiran je kao šum i eliminira se iz gnijezda vektora. U slučaju da referentni grumen *i* ne predstavlja sferu, unutar vektora kandidata pojavit će se grumen koji ima manje od tri glasa i biti će izbrisan. Ponavljanjem glasanja bez prethodnog kandidata ostali kandidati će išćeznuti i cijeli vektor kandidata je odbačen.

U slučaju da sve skupine detekcija zadovoljavaju uvjet Euklidske udaljenosti, proces se ponavlja ali sada je kao referenca odabran drugi grumen. Glasanje se smatra uspješnim ako je u novom vektoru kandidata preostalo četiri grumena, od kojih svaki ima po tri glasa, **Slika 30.**



Slika 31. Grumeni sfera markera nakon filtriranja neželjenih detekcija

4.6. Estimacija koordinati centra sfera

Nakon detekcije sve četiri sfere markera potrebno je iz skupa detekcija i koordinata kružnica estimirati centar svake sfere. Odabrana metoda je linearna regresija metodom najmanjih kvadrata. Rješenje metode najmanjih kvadrata dobiveno je dekompozicijom svojstvenih vrijednosti (*Singular Value Decomposition)* matrice kovarijanci koordinata. Cilj je bio pronaći pravac koji zadovoljava kriterij najmanjih kvadrata udaljenosti po svim koordinatama za svaku sferu u aksijalnom i sagitalnom pogledu. Poznavajući pravce iz oba pogleda za svaku sferu kod savršenih detekcija dobilo bi se sjecište koje bi definiralo koordinate središta sfere. Kako detekcije nisu savršene, postojat će određena udaljenost između pravaca koja određuje odstupanje između detekcija sfere iz različitih pogleda. Pravci su predstavljeni vektorskim zapisom, tj. jednom točkom pravca i vektorom smjera.

Parametarska jednadžba trodimenzionalnog vektora je:

$$r = m + tp, \tag{4.4}$$

gdje je :

- *r* vektor koordinata (*x,y,z*)
- *m* 3 x 1 vektor srednjih vrijednosti koordinata
- *p* 3 x 1 svojstveni vektor najveće svojstvene vrijednosti (vektor smjera)
- *t* udaljenost duž pravca, u ovom slučaju udaljenost od točke *m*

Vrijednosti vektora *p* dobivaju se dekompozicijom svojstvenih vrijednosti.

Prema metodi, bilo koja matrica A može se rastaviti u tri matrice:

$$A = U\Sigma V, \tag{4.5}$$

gdje je :

- U ortogonalna matrica, sadrži lijeve svojstvene vektore
- Σ dijagonalna matrica svojstvenih vrijednosti matrice A
- V ortogonalna matrica, sadrži desne svojstvene vektore

Matrice *U* i *V* predstavljaju rotaciju, dok je Σ matrica rastezanja.

Množenjem matrice *A* transponiranom verzijom te matrice:

$$A^T A = \left(V\ \Sigma^{\mathrm{T}} U^T\right)\mathrm{U}\ \Sigma\ \mathrm{V}^{\mathrm{T}} = V\ \left(\Sigma^{\mathrm{T}}\Sigma\right)\mathrm{V}^{\mathrm{T}}, \tag{4.6}$$

dobiva se pozitivna semidefinitna matrica $A^T A$ čiji su svojstveni vektori ortogonalni. Svojstvene vrijednosti dijagonalne matrice $\Sigma^{\mathrm{T}}\,\Sigma$ su pozitivne jer su kvadrati svojstvenih vrijednosti matrice *A*.

Svojstvenim vektorom *p* najveće svojstvene vrijednosti dobiveni su kosinusi smjera koji određuju nagib pravca u odnosu na koordinatne osi Kartezijevog koordinatnog sustava. Kao baza koristi se vektor srednjih vrijednosti *m.*

Najkraća udaljenost između dvaju pravaca računa se kao duljina linije okomite na oba pravca.

Poznavajući svojstvo vektorskog produkta, u vektorskom zapisu linija najkraće udaljenosti dvaju pravaca može se izračunati kao vektorski produkt pozicijskih vektora pravaca.

Ako postoje dva pravca u vektorskom zapisu:

$$r_1 = m_1 + t_1\, p_1, \tag{4.7}$$

$$r_2 = m_2 + t_2\, p_2, \tag{4.8}$$

tada vektor okomit na vektore $r_1\ i\ r_2$ glasi:

$$n = p_1 \times p_2, \tag{4.9}$$

dok je jedinični vektor $\hat{n}$, vektor *n* podijeljen njegovom l^2 normom:

$$\hat{n} = \frac{p_1 \times p_2}{|p_1 \times p_2|}. \tag{4.10}$$

Nazovimo Q neku točku na pravcu r_1, P točku na pravcu r_2.

Vektor QP sada poprima izraz:

$$QP = m_1 - m_2 + t_1\, p_1 - t_2\, p_2. \tag{4.11}$$

Najkraća udaljenost dvaju vektora je projekcija vektora QP na $\hat{n}$, dakle skalarni produkt tih vektora:

$$d = QP\,.\hat{n} \tag{4.12}$$

Orijentacija koordinatnog sustava slike snimaka aksijalnog pogleda poklapa se orijentacijom koordinatnog sustava svijeta, pa se uzima kao referentna. Da bi se mogla izračunati najkraća udaljenost između pravaca koji aproksimiraju detekcije sfera, potrebno je koordinate detekcija iz sagitalnog pogleda geometrijski uskladiti sa koordinatama detekcija iz aksijalnog pogleda. Rastavljanjem matrice transformacije sagitalnog pogleda:

$$R_{sagittal} = R_{a2s}\, R_{axial} \tag{4.13}$$

gdje su:

- $R_{sagittal}$ matrica transformacije iz originalnog pogleda VTK knjižnice u sagitalni
- R_{a2s} matrica transformacije iz aksijalnog pogleda u sagitalni
- R_{axial} matrica transformacije iz originalnog pogleda VTK knjižnice u aksijalni

Matrica R_{a2s}dobiva se kao:

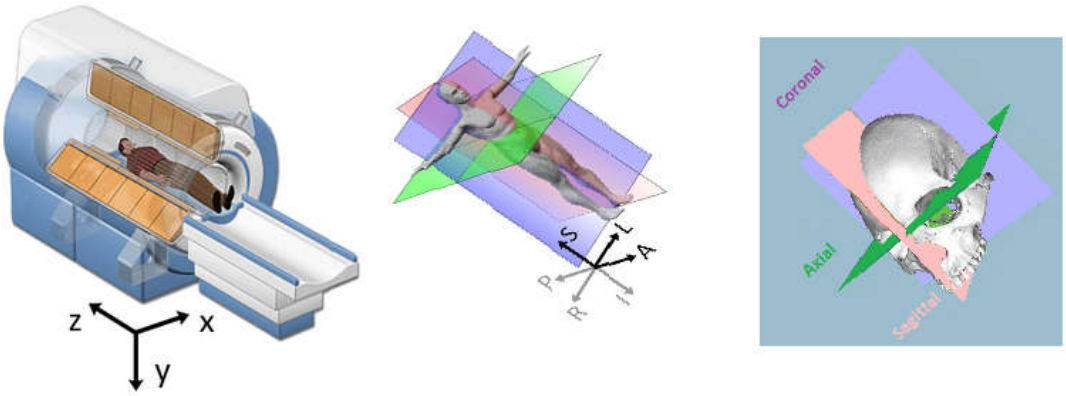
$$R_{a2s} = R_{sagittal}\, {R_{axial}}^{-1}, \tag{4.14}$$

iz čega slijedi izraz za transformaciju koordinata sagitalnog pogleda u koordinatni sustav aksijalnog pogleda:

$${R_{a2s}}^{-1}\, R_{sagittal} = R_{axial}, \tag{4.15}$$

i vektora transformiranih koordinata:

$$r_a = {A_{a2s}}^{-1}\, A_{sagittal} \cdot r_S \tag{4.16}$$



Slika 32. Usporedba koordinatnog sustava svijeta i anatomskih pogleda

Poznavajući najkraću udaljenost detekcija sfera iz različith pogleda i vektor smjera te udaljenosti, točka na središtu pravca najkraće udaljenosti uzima se kao detektirani centar sfere.

Ishodište koordinatnog sustava svijeta poklapa se sa ishodištem definiranim u postavkama CT uređaja. Međutim, kako koordinatni sustavi aksijalnog i sagitalnog pogleda ne postoje u stvarnom svijetu, da bi se dobila stvarna mjera centara detektiranih sfera vrši se translacija koordinati u koordinatni sustav svijeta.

$$S_{w,n} = S_n + t_{world} \tag{4.17}$$

gdje je:

- $S_{w,n}$ detekcija centra sfere *n* u koordinatnom sustavu svijeta
- S_n detekcija centra sfere *n* u aksijalnom koordinatnom sustavu
- t_{world} matrica translacije iz aksijalnog koordinatnog sustava u koordinatni sustav svijeta

4.7. Verifikacija rezultata

Za kvalitativnu procjenu detekcija koordinata centra sfera markera u koordinatnom sustavu svijeta korišten je softverski paket MedInria. Međutim, taj koordinatni sustav nema praktičnu primjenu van procesa lokalizacije markera. Položaj i orijentaciju određene strukture tijela pacijenta ili ciljanu trajektoriju potrebno je odrediti u odnosu na položaj i orijentaciju markera. Iz tog razloga jedna od sfera markera uzima se kao ishodište novog koordinatnog sustava, dok su smjerovi koordinatnih osi usmjereni prema ostalim sferama.

Izvođenje algoritma testirano je na četiri različita niza DICOM snimaka. Svaki od nizova predstavlja isti testni fantom (prikazan na **Slika 29.**) različite orijentacije na koji je učvršćen aluminijski marker sa četiri retro-refleksivne sfere. Nizovi snimaka sastoje se od različitoga broja snimaka, promjenjivih parametara ishodišta koordinatnog sustava i razmaka između piksela.

Algoritam je u uspješno detektirao sve četiri sfere markera na svakome od testnih fantoma.

U **Tablica 2.** uspoređene su procjenjene koordinate sfera sa koordinatama sfera izmjerenih u laboratoriju za precizna mjerenja duljine koordinatnim mjernim strojem. Izmjereni su kvadrati odstupanja svake sfere markera po koordinatma *x,y,z* i prosjek srednje kvadratne pogreške za sve četiri sfere. Minimalni iznos prosjeka srednje kvadratne greške za sve četiri sfere iznosi 0,36mm dok je maksimalni iznos 1,06mm. Motivacija za razvoj algoritma proizlazi iz [15],

gdje je kliničkim ispitivanjima utvrđeno smanjenje greške lokalizacije markera eliminacijom ljudskog čimbenika, postignuta ušteda vremena tokom operativnih zahvata i smanjen stres osoblja u predoperativnim postupcima.

CMM (mjerenje)	X	Y	Z	Eukl.Greška za xyz
Sfera 1	0	0	0	
Sfera 2	84,974	0	0	
Sfera 3	7,014	78,048	0	
Sfera 4	70,796	70,747	0,0157	
Fantom 1				
Sfera 1	0	0	0	
Sfera 2	84,842	0	0	0,1329
Sfera 3	7,2758	77,496	0	0,6101
Sfera 4	70,5265	70,569	-0,081	0,3374
Prosjek greške				**0,3601**
Fantom 2				
Sfera 1	0	0	0	
Sfera 2	84,791	0	0	0,1837
Sfera 3	6,936	78,035	0	0,0793
Sfera 4	70,348	70,575	0,827	0,9427
Prosjek greške				**0,4019**
Fantom 3				
Sfera 1	0	0	0	
Sfera 2	84,375	0	0	0,5991
Sfera 3	7,837	76,899	0	1,4130
Sfera 4	70,743	70,807	-1,154	1,1738
Prosjek greške				**1,0620**
Fantom 4				
Sfera 1	0	0	0	
Sfera 2	84,969	0	0	0,0055
Sfera 3	7,846	78,017	0	0,8323
Sfera 4	71,047	70,847	-0,238	0,3709
Prosjek greške				**0,4029**

Tablica 2. Iznosi grešaka detekcija sfera

5. ZAKLJUČAK

Razvijen je algoritam za automatsku lokalizaciju markera pacijenta u volumetrijskim snimkama CT uređaja u C++ programskom jeziku. Uspješno su integrirane knjižnice obrade medicinskih snimaka DICOM formata i strojnoga vida. Povezivanjem informacija dobivenih iz meta-podataka i geometrijski usklađenih snimaka aksijalnog i sagitalnog pogleda, povezane su Kartezijeve koordinate u metričkom sustavu sa pozicijama na snimci.

Metodama strojnog vida, na temelju intenziteta snimaka detektirane su značajke kružnica i razvijene metode filtriranja smetnji i grupiranja detekcija sfera markera. Iz grumena detekcija različitih pogleda razvijena je metoda estimacije centara sfera markera.

Algoritam je uspješno detektirao sve četiri sfere markera u četiri testna uzorka nizova DICOM snimaka. Procjenjene koordinate centara sfera markera uspoređene su sa laboratorijskim mjerenjem. Na ograničenom testnom uzorku, minimalni iznos prosjeka srednje kvadratne pogreške sve četiri sfere markera iznosi 0,36mm, maksimalni iznos 1,06mm, dok je prosjek za sva četiri mjerenja 0,55mm. Iako su testni uzorci temeljeni na jednom fantomu različitih orijentacija, program se pokazao stabilnim tijekom rada sa promjenjivim parametrima postavki CT uređaja. Rad služi kao polaznica daljnjeg istraživanja u području registracije pacijenta i razvijanja novih metoda automatske lokalizacije markera. Vezano na ovaj rad, planira se dodatno testiranje pouzdanosti i točnosti algoritma, mogućnosti njegove praktične primjene i poboljšanje točnosti estimacije koordinata markera.

LITERATURA

[1] J.A. Smith, J. Jivraj, R. Wong, V. Yang, „30 Years of Neurosurgical Robots- Review and Trends for Manipulators and Associated Navigational Systems“, Ann. Biomed. Eng, Vol. 44, No. 4, pp. 836-846, Apr. 2016.

[2] Chen, Dingguo, Jun Tan, Vipin Chaudhary, and Ishwar K. Sethi. "Automatic fiducial localization in brain images." International Journal of Computer Assisted Radiology and Surgery 1 (2006): 45.

[3] Medha V. Wyawahare, Dr. Pradeep M. Patil, Hemant K. Abhyankar „Image Registration Techniques: An overview“, International Journal of Signal Processing, Image Processing andPattern Recognition, Vol. 2, No.3, September 2009

[4] Coordinate Systems. Dostupno na: https://www.slicer.org/wiki/Coordinate_systems [20. veljačke 2017.]

[5] www.vtk.org Proposals: Orientation. Dostupno na: http://www.vtk.org/Wiki/Proposals:Orientation [10. ožujka 2017.]

[6] P. Dias et al., Craniometric Analysis Application Using 3D Skull Models, *IEEE Computer Graphics and Applications*, vol. 35, no. , pp. 11-17, Nov.-Dec. 2015, doi:10.1109/MCG.2015.136

[7] DICOM – Digital Imaging and Communications in Medicine. Dostupno na: http://dicom.nema.org/Dicom [15.studeni 2016.]

[8] DICOM Browser. Dostupno na: https://dicom.innolitics.com/ciods/ct-image [15. veljače 2017.]

[9] VTKUsersGuide, 11th edition, Kitware, inc. 2010.

[10] VTK/Tutorials/NewPipeline Dostupno na: http://www.vtk.org/Wiki/VTK/Tutorials/New_Pipeline [20.studeni 2016]

[11] A Very Basic DICOM Introduction. Dostupno na: https://dcm4che.atlassian.net/wiki/display/d2/A+Very+Basic+DICOM+Introduction [6. prosinca 2016.]

[12] Gonzalez, R.C., Woods, R.E., *Digital Image Processing*, 3rd edition Prentice Hall, 2008.

[13] R. O. Duda, P. E. Hart, „Use of the Hough Transformation To Detect Lines and Curves in Pictures“, Stanford Research Institute. Dostupno na: https://www.cse.unr.edu/~bebis/CS474/Handouts/HoughTransformPaper.pdf [11. veljače 2017.]

[14] Hounsfield Unit, Dr Yuranga Weerakkody and Kyle Greenway et al. Dostupno na: https://radiopaedia.org/articles/hounsfield-unit [7. siječnja 2017.]

[15] Šuligoj, F., Švaco, M., Jerbić, B., Šekoranja, B., & Vidaković, J. (2017). Automated Marker Localization in the Planning Phase of Robotic Neurosurgery. *IEEE Access*

PRILOZI

I. CD-R disc