

Diplomamunka

**Időben változó folyamat parallel vetítésű SPECT leképezés
3D képrekonstrukciós módszerének kidolgozása klinikai
alkalmazások számára**

Barna Zsófia

Témavezető:

Dr. Kári Béla

Tudományos munkatárs

Semmelweis Egyetem, Általános Orvostudományi Kar
Radiológiai és Onkoterápiás Klinika
Nukleáris Medicina Tanszék

Hesz Gábor

Tudományos segédmunkatárs

BME Irányítástechnika és Informatika Tanszék

Belső konzulens:

Dr. Légrády Dávid

Egyetemi docens

BME Nukleáris Technika Intézet

2016

Alulírott **Barna Zsófia**, a Budapesti Műszaki és Gazdaságtudományi Egyetem fizikus MSc szakos hallgatója kijelentem, hogy ezt a diplomamunkát meg nem engedett segédeszközök nélkül, önállóan, a témavezető irányításával készítettem, és csak a megadott forrásokat használtam fel. Minden olyan részt, melyet szó szerint, vagy azonos értelemben, de átfogalmazva más forrásból vettem, a forrás megadásával jelöltem.

Budapest, 2016. június 6.

.....

Barna Zsófia

Fizika MSc :: Diplomamunka :: Kiírás

Konzulens:

Neve: Légrády Dávid

Tanszéke: NTI

E-mail címe: legrady@reak.bme.hu

Telefonszáma: 1254

Azonosító: DM-2014-26	
Diplomamunka címe:	Időben változó folyamat parallel vetítésű SPECT leképezés 3D képrekonstrukciós módszerének kidolgozása a klinikai alkalmazások számára.
Melyik szakiránynak ajánlott?	"Orvosi fizika"
A jelentkezővel szemben támasztott elvárások:	Nukleáris Medicina ismerete, orvosi képző eljárások, jelfeldolgozás, villamosmérnöki ismeretek
Témavezető neve:	Dr. Kári Béla tudományos munkatárs
Témavezető Intézménye:	SE ÁOK, Radiológiai és Onkoterápiás Klinika / Nukleáris Medicina Tanszék
Témavezető e-mail címe:	kari.bela@med.semmelweis-univ.hu
Leírása:	<p>A SPECT (Single Photon Emission Tomography) leképezési eljárások a mai funkcionális klinikai diagnosztikában olyan bio-kémiai állapotban valósulnak meg, amikor az aktivitás eloszlás a vizsgált szervben nem változik mérhetően. Biológia jellel (EKG) szinkronizált vizsgálatok esetén a szív vértartalma (kapuzott blood pool vizsgálat), valamint a szív térfogat változása és a kamrafal mozgások (kapuzott szívizom szcintigráfia) dinamikája nyomon követhető az idő függvényében. További fontos információt hordoz a bio-kémiai folyamatok élő szervezetben történő dinamikája szempontjából a farmakon beépülés és kiürülés mechanizmusa is. Ezen folyamatok szimultán térbeli-időbeli változásának vizsgálata képezi a szakdolgozat témáját, melyben fel kell állítani azon kritérium rendszert, mely alapján parallel vetítési technikával a folyamatok időbeli és térbeli tulajdonságai egyidejűleg vizsgálhatók (a 4D rekonstrukció feltételrendszerei). A hallgatónak a szakdolgozat keretében a következő feladatokat kell elvégeznie: - Tanulmányozza az élő szervezetben lejátszódó bio-kémiai folyamatok dinamikáját, Tipikusan: szív, vese, máj, agy. - Készítsen 4D dinamikus matematikai fantomot, mely segítségével szimulálható az egyes szervek időbeli-térbeli farmakon eloszlás változása. - Különböző parallel leképező geometriák esetén szimuláció alapján határozza meg a dinamikus folyamatok leképezésének és rekonstrukcióinak kivitelezhetőségét és kritérium rendszerét (képfelvételi technika, képfeldolgozás, jel/zaj analízis, 4D rekonstrukciós eljárás). - A szimulációval kapott eredményeket vesse össze a jelen parallel technológiával rendelkező SPECT rendszerekkel és határozza meg a klinikai alkalmazhatóság módszerét. - Optimalizálja a kialakított képrekonstrukciós módszert mind képminőség, mind futási idő tekintetében, hogy alkalmas legyen a meghatározott dinamikus folyamatok térbeli-időbeli analízisére rutin klinikai körülmények között is.</p>

Tartalomjegyzék

1. Bevezetés.....	6
2. Kapuzott szívizom SPECT vizsgálatok.....	8
2.1. SPECT képalkotás	8
2.2. Kapuzott szívizomvizsgálatok menete	8
2.3. Kapuzott szívizomvizsgálatok kiértékelése.....	11
3. Néhány általánosság.....	14
3.1. Szívizomvizsgálatokat szimuláló és feldolgozó programkód felépítése	14
3.2. DICOM fájlkezelés és koordinátarendszer	15
3.3. Környezet.....	18
4. Szív mozgását közelítő matematikai fantom.....	19
4.1 Saját matematikai fantom	19
4.2. NCAT szívphantom.....	24
5. Projekciós képek	26
5.1. Leképezés fizikája	26
5.2. Valós projekciós képek.....	29
5.3 Előrevetítés Monte Carlo alapú szimulációval	29
5.4 Előrevetítő függvény modell alapú korrekciókkal	29
6. Képrekonstrukció és korrekciók háromdimenziós esetben.....	35
6.1 ML-EM (Maximum Likelihood – Expectation Maximization) algoritmus	35
6.2. OS-EM (Ordered Subset - Expectation Maximization) algoritmus	43
6.3. MAP-EM (Maximum A Priori – Expectation Maximization) algoritmus	44
6.4 A rendszermátrix elemeinek meghatározása, visszavetítő függvény	47
6.5 A prior eloszlás becslése, TV (total variation)-norma.....	50
6.6 A háromdimenziós rekonstrukció megvalósítása	53
7. Időbeli korrekciók	56
7.1 Lehetőségek	56
7.2 Büntetőfüggvény kiterjesztése idő dimenzióra.....	56
7.3 Szűrés Fourier-térben	57
8. Rekonstrukció és korrekciók kiértékelése.....	59

8.1. Kiértékelés menete, mérőszámok	59
8.2. ML-EM/OS-EM algoritmus tesztelése időbeli korrekciók nélkül	60
8.3. OS-EM algoritmus paramétereinek megválasztása időbeli korrekciók nélkül	62
8.4. Időbeli regularizáció hatása a rekonstruált kép minőségére	64
8.5. Fourier-térbeli szűrés hatása a rekonstruált kép minőségére	69
8.6. Rekonstrukció tesztelése más matematikai fantomokon	73
8.7. Eredmények összefoglalása, hibalehetőségek	76
9. Összefoglalás.....	77
10. Akiknek köszönöm.....	78
11. Hivatkozások.....	79
12. Ábrák jegyzéke.....	82

1. Bevezetés

A SPECT (Single-Photon Emmission Computed Tomography) képalkotó eljárás során a vizsgált szervbe radioaktív anyagot juttatunk, az ebből származó sugárzást detektáljuk különböző irányokból. Az így kapott vetületi képek alapján becsüljük a betegben kialakult izotópeloszlást, majd ebből következtetünk a szerv megfelelő vagy rendellenes működésére. A klinikai vizsgálatok egy része időben változó folyamatot képez le, ilyenek a kapuzott perfúziós szívizomvizsgálatok is. A felvételeknél kihasználjuk az időbeli változások kváziperiodikus mivoltát, a szív ciklust általában 8 vagy 16 időszeletre bontjuk, ezek mindegyikéhez külön vetületi képeket veszünk fel. Ezekből rekonstruáljuk a szív aktuális állapotát minden egyes időszeletre, a klinikai gyakorlatban az időszelleteket általában egymástól függetlenül kezeljük. Az egyes időszelletek valójában nem függetlenek egymástól, a rekonstruált kép minőségét javíthatjuk, ha egy adott képtérvoxel adott időszeletszámú értékének meghatározása során figyelembe vesszük a többi időszelethez tartozó vetületi képeket is.

Diplomamunkám során a kapuzott szívizomvizsgálatok rekonstrukciójának lehetőségeivel foglalkoztam. Néhány, a klinikai gyakorlatban használt rekonstrukciós algoritmus elméleti alapjával és a szakirodalomban előforduló időbeli korrekciók egy részével ismerkedtem meg, illetve egy SPECT képrekonstrukciót megvalósító, C++ nyelvű programkódot állítottam össze, amely kapuzott szívizomvizsgálatokat dolgoz fel. Első lépésként egyszerű felépítésű matematikai fantomot hozhatunk létre, amely a szív bal kamrájának alakját, mozgását közelíti, ezután a fantomról projekciós képeket készíthetünk (vagyis a vizsgálatot szimulálhatjuk). A vetületi képek alapján a matematikai fantomot különböző korrekciókat alkalmazva rekonstruálhatjuk, végül a rekonstruált kép minőségét értékelhetjük ki.

Egy képrekonstrukció során gyakran alkalmazott statisztikai iteratív módszer a MAP-EM (maximum a priori – expectation maximization) algoritmus, ezt valósítottam meg. A statisztikai módszerek előnye, hogy a vizsgált fantom/személy, illetve leképező rendszer strukturális felépítése alapján a leképezést meghatározó jelenségekkel számolhatunk (sugárzás csillapodása, detektor felbontásának távolságfüggése), hatékonyabban, mint a hagyományos, klinikai gyakorlatban elterjedt analitikus módszer (szűrt visszavetítés) esetében, így jobb minőségű képeket kaphatunk. A rekonstruált kép várható tulajdonságairól is előfeltételezéseket tehetünk, ezeket matematikai formába öntve (ez jelenti a prior – előzetes –információt) az eredményt szintén tovább pontosíthatjuk, például térbeli regularizáció alkalmazásával. Az

aktivitásértékek időbeli változását többféleképpen figyelembe vehetjük. Két lehetőséget választottam ki, ezekkel bővíttem a rekonstrukciós algoritmust: a prior információ alkalmazása idő dimenzióra kiterjeszhető, illetve a szív mozgásának periodikus mivoltát kihasználva Fourier-térben való szűréssel is megpróbálhatunk javítani a képminőségen. Diplomamunkámban azt vizsgáltam, hogy érdemes lehet-e részletesebben foglalkozni ezekkel a módszerekkel.

A MAP-EM algoritmus és a térbeli korrekciók (sugárzás csillapodásának, illetve leképezés távolságfüggésének figyelembevétele, térbeli regularizáció), amelyeket én is alkalmaztam, korábbi tapasztalatok alapján hatékonyak bizonyultak, klinikai bevezetés alatt állnak (a Mediso Kft. által fejlesztett InterView XP szoftver [1] részeként). Az időbeli összefüggések figyelembevétele egyelőre nem része a rekonstrukciós algoritmusnak, ezzel a továbbiakban a tanszéken foglalkozni szeretnénk.

Diplomamunkámban a felhasznált algoritmusokat és korrekciókat, illetve az időbeli változások figyelembevételének hatását foglalom össze.

2. Kapuzott szívizom SPECT vizsgálatok

2.1. SPECT képalkotás

SPECT vizsgálatok esetében a betegnek farmakont adunk, amelyet gamma-sugárzó izotóppal jelölünk, leggyakrabban technéciummal. Ez megkötődik a vizsgálni kívánt szervben, az alkalmazott farmakon határozza meg, hogy melyikben. A beteg körül gamma-kamera forog, ezzel detektálni tudjuk a szervezetbe jutott radiofarmakonból érkező fotonok pozícióját, energiáját és intenzitását.

Így elő tudjuk állítani az egyes szögállásokban a betegben kialakult aktivitáseloszlás vetületi képét, vagyis meg tudjuk határozni, hogy az egyes detektorpixelekbe hány foton érkezett a detektor síkjára merőleges irányból (parallel vetítés esetében). A képek általában step-and-shoot technikával készülnek, azaz a kamera az egyes pozíciókban megáll, meghatározott ideig adatokat gyűjt, majd innen halad tovább a következő szögpozícióba (a rutin klinikai vizsgálatok többségében 360°-os szögtartományban mintavételezünk).

A felvett projekciós képek alapján becsülni tudjuk a betegben kialakult aktivitáseloszlást. Az így rekonstruált kép funkcionális információt ad a szervről, azt tudjuk megállapítani, hogy a szerv mely területei mennyi farmakont vettek fel, ezzel arányos az adott területről érkező sugárzás intenzitása. Például daganatokat lokalizálhatunk így (ezek jóval többet/kevesebbet kötnek meg, mint a környező egészséges szövetek), elhalt vagy nem kielégítő vérellátású szöveteket határozhatunk meg (ide kevesebb farmakon épül be). A kapott kép elsősorban a vizsgált szerv működéséről, és nem morfológiájáról ad információt, bár a kettő gyakran korrelációt mutat. Egyéb képalkotó eljárásokkal – CT, röntgen, MRI – felvett képek alapján határozhatjuk meg a rendellenesség morfológiáját.

2.2. Kapuzott szívizomvizsgálatok menete

Kapuzott szívizomvizsgálatok segítségével a szívizom vérellátásáról, illetve a szív pumpafunkciójának megfelelőségéről kaphatunk információt. Ez a klinikai gyakorlatban az egyik leggyakrabban végzett izotópdiagnosztikai vizsgálat: például iszkémiás betegség (a vérellátás csökkenését vagy megszűnését iszkémiának nevezzük) helyét, súlyosságát, infarktus

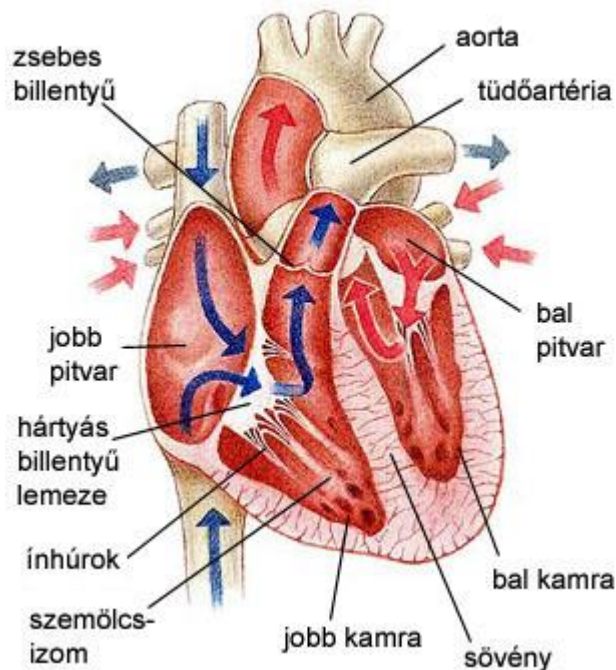
utáni károsodás mértékét, szívbetegség kockázatát lehet így meghatározni, bypass műtétek és utókezelések elengedhetetlen eszköze.

A vizsgálat során a beteg intravénásan sugárzó izotóppal jelölt farmakont kap, ez a véráramba kerül, majd a szívfal izomsejtjeiben megkötődik a szívizom vérellátottságával arányos mennyiségben. Egészséges szív működés esetében a szívfalban homogén eloszlást kellene tapasztalnunk.

Miután a radiofarmakon a szívben megkötődött, a beteget a SPECT képalkotó készülékbe ültetik/fektetik. A vizsgálat során felvételeket csak 180°-os szögterületben (a szív felőli oldalon) készítenek, mivel az alany jobb oldalán már akkora a detektor és a szív közé eső szövetek csillapítása, hogy az innen készített vetületi képek nem javítanak jelentősen a rekonstruált kép minőségén, viszont a vizsgálat idejét feleslegesen növelnék. A szív felőli oldalon az izmok, illetve nők esetében az emlő csillapítása ronthatja leginkább a felvételek statisztikáját. A szív ciklust időszelletekre osztjuk, a SPECT felvételt EKG jellel szinkronizáljuk, ez alapján határozzuk meg az időszeltek hosszát. A szív ciklus kezdetének az EKG jel R-hullámát számítjuk, ekkor jönnek ingerületbe a kamrai szívizomrostok (az EKG-n a kamraizomzat nagy tömege miatt ez produkálja a legnagyobb jelet). A vizsgálat előtt az EKG jel R-R távolsága alapján mérjük az egyes szív ciklusok hosszát, ezekből átlagot számolunk, majd ezt az időtartamot osztjuk annyi egyenlő időtartamra, ahány időszeltek felvenni szeretnénk. A SPECT adatgyűjtés során csak azokat a szív ciklusokat fogadjuk el, amelyek időtartama előre definiált határokon belül megegyezik a mért átlagos időtartammal (szabálytalanabb szív működés esetében emiatt megnőhet a vizsgálat hossza). A beérkező beütéseket a megfelelő időszeltekhez regisztráljuk. A klinikai gyakorlatban nyugalmi és terheléses (fizikai/gyógyszeres) vizsgálatot végeznek. Terhelés hatására a szívizom munkája, ezzel együtt az oxigénigénye is nő. A szöveti perfúzióknak emiatt szintén a terheléssel arányosan kellene nőnie, ebben az esetben azt vizsgáljuk, hogy a nyugalmi állapothoz képest megfelelő mértékben változik-e a szívfal egyes területeinek vérellátása. A szív ciklust általában 8, 16, 20 vagy 24 időszeltekre bontjuk, minden időszeltekhez előállítjuk az aktuális izotópeloszlás rekonstruált képét, mind a nyugalmi, mind a terheléses állapotban.

A szív négy részre: jobb és bal pitvarra, illetve kamrára osztható. A bal kamra pumpálja az oxigénben dús vért a szervekhez, ez végzi a legnagyobb munkát. Ennek a térfogata a legnagyobb, a nagyobb terhelés miatt fala lényegesen vastagabb, mint a jobb kamráé. A

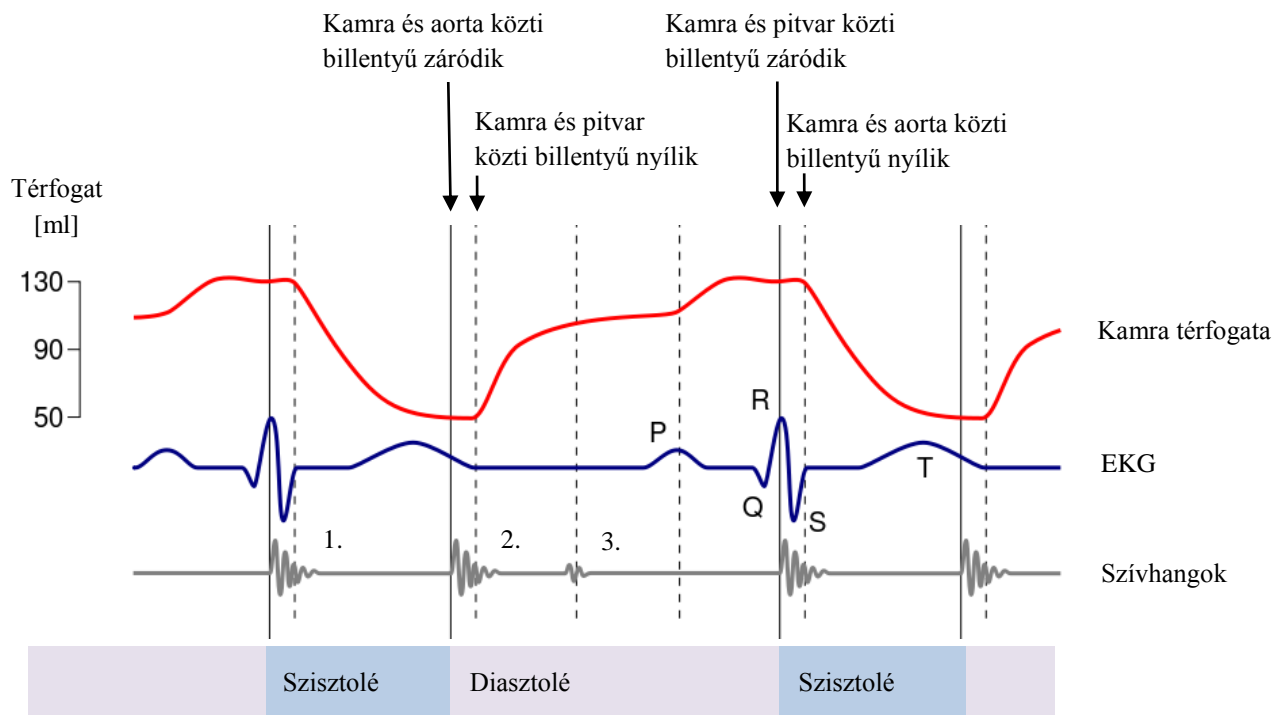
vizsgálatok kiértékelése során (amennyiben nem a szív egy specifikus részét szeretnénk vizsgálni, hanem az általános jellemzőit) a bal kamra változásait elemezzük.



2.2.1. ábra – A szív üregei [K1]

A bal kamrát a bal pitvartól, illetve az aortától szívbillentyűk választják el. Ezek a nyomásviszonyoknak megfelelően nyílnak, illetve záródnak, meghatározzák, hogy mikor áramlik a vér a pitvarból a kamrába, illetve a kamrából az aortába. A bal kamrát tekintve a szívciklust két szakaszra bonthatjuk. Az ürülés szakaszát kamrai **szisztolénak** nevezzük. Ezalatt a kamraizomzat összehúzódik, ennek következtében az oxigénben dús vér az aortába pumpálódik. Ezután a kamra ismét telítődni kezd (térfogata ennek megfelelően nő), ezt a szakaszt kamrai **diasztolénak** nevezzük. Amennyiben a ciklus elejének az EKG jel R hullámát tekintjük, a kamrák legnagyobb térfogatukat a ciklus elején (a diasztolés szakasz végén) érik el, ezt nevezzük **EDV**-nek (end diastole volume – diasztolé végi térfogat), legkisebb térfogatukat pedig körülbelül a ciklus harmadánál (a szisztolés szakasz végén), ez az **ESV** (end systole volume – szisztolé végi térfogat).

Mind a szisztolé, mind a diasztolé izovolumetriás szakasszal kezdődik: ekkor minden billentyű zárt, a kamrában levő vér térfogata állandó marad. Az izovolumetriás szakaszt követi a kamra telítődése, illetve ürülése, ezek kezdetben gyors, majd lassuló térfogatváltozást eredményeznek. A kamratérfogat változását grafikonon ábrázolhatjuk, a 2.2.2. ábra a bal kamrai szív ciklus eseményeit foglalja össze.



2.2.2. ábra – EKG jel és a bal kamra térfogatának változása a szív ciklus során [K2]

2.3. Kapuzott szívizomvizsgálatok kiértékelése

A rekonstruált képen láthatjuk, hogy mely területeknek nem kielégítő a vérrellátása (ide nem/kevésbé jut el a véráramba került sugárzó izotóp, kevés épül be). Fontos információt jelent a bal kamra térfogatváltozásának mértéke is, hiszen ez adja meg, hogy mekkora vértérfogatot képes megmozgatni a szív egy ciklus során. A szívfal mozgását, vastagodását elemezve szintén működési rendellenességekre következtethetünk, pl. az elhalt szövet csak passzívan követi a körülötte normálisan összehúzódó rostokat, fáziskiesés, paradox mozgás.

A bal kamra falát gyakran kétdimenziós körlapra képezzük le (bull's eye kép). [27] Ezen a kamra tengelyére merőleges metszeti képek körgyűrűkként jelennek meg: a szívesűcs a körlap

közepére kerül, majd ahogy távolodunk, a metszeti képeket egyre nagyobb sugarú koncentrikus körgyűrűkként ábrázoljuk.

Pumpafunkció, térfogatváltozás

Az egyik legfontosabb paraméter a kamra térfogatváltozásának mértéke. Ez egyénekenként változó, viszonylag tág határok között mozog. A szív pumpafunkcióját az **ejekciós frakcióval** (EF) jellemezhetjük, amely a bal kamra térfogatváltozásának (a megmozgatott vér térfogatának) a legnagyobb térfogathoz viszonyított arányát adja meg.

$$EF = \frac{EDV-ESV}{EDV} \quad (2.3.1)$$

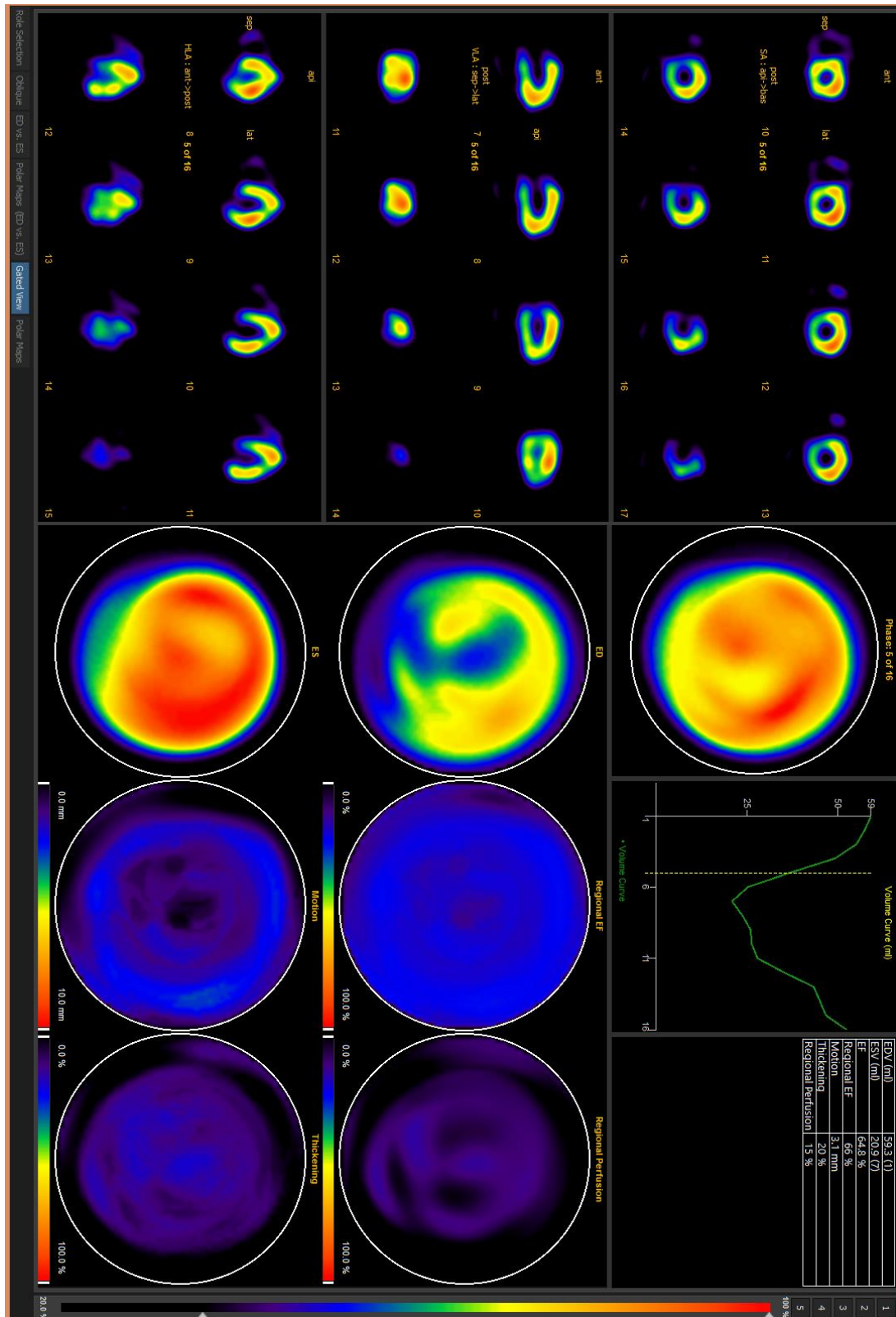
Ez egy egészséges szívnél 55 és 75% közötti érték. 40 és 55% között ugyan az átlagosnál kisebb a megmozgatott vértérfogat, általában ez még nem okoz komolyabb problémát. 40% alatti ejekciós frakció szívelégtelenséget jelent. Alacsonyabb ejekciós frakciót a szív megnagyobbodással képes kompenzálni.

Falvastagság változása, szívfal elmozdulása

Egy másik fontos paraméter a szívfal megvastagodásának és elmozdulásának mértéke a szívfal egyes részein. Normális esetben a szívfal kicsit megvastagodik, amíg a kamra térfogata lecsökken és eléri a szisztolé végi állapotot. A vastagodás mértéke helyenként eltérő, pl. a szívcsúcs közelében nagyobb mértékű, mint a bázis közelében. Rendellenes esetben a vastagodás elmaradhat, vagy akár vékonyodhat is a fal. Az átlagos falvastagodást és elmozdulást bull's eye képen ábrázolhatjuk.

Vérellátottság

A szívfal egyes részeinek vérellátottságára a megkötődött izotópmennyiségből következtethetünk. A falvastagodáshoz hasonlóan ábrázolhatjuk.



2.3.1. ábra – Kapuzott szívizomvizsgálat kiértékelése – a kép jobb oldalán a bal kamra falában megkötődött sugárzó izotóp mennyiségét látjuk minden időszelethez (ez a vérellátottsággal arányos). A bal kamra térfogatváltozását görbe mutatja, a regionális ejekciós frakciót, a szívfall vérellátottságát, megvastagodását és elmozdulását bull's eye képen láthatjuk (a kép jobb oldalán).

3. Néhány általánosság

3.1. Szívizomvizsgálatokat szimuláló és feldolgozó programkód felépítése

Diplomamunkám során egy kapuzott szívizomvizsgálatokat szimuláló, a rekonstrukció és különböző korrekciók tesztelésére alkalmas C++ programkódot írtam. A program négy fő részből áll, ezeket a 4-7. fejezetekben mutatom be részletesebben:

1. Szív mozgását közelítő matematikai fantom generálása

Fontos, hogy a rekonstrukciót és a korrekciók hatását ismert felépítésű matematikai fantomokon teszteljük. Ezért első lépésként egyszerű, a szív bal kamrájának alakját és mozgását közelítő matematikai fantomot hozhatunk létre különböző méretekkel, illetve defektusokkal (csökkent aktivitású területekkel). Fantom helyett egyéb (megfelelő formájú) DICOM (Digital Imaging and Communications in Medicine) [2] formátumú fájlt is betölthetünk további feldolgozás céljából.

2. Projekciós képek készítése

A generált fantomról vagy betöltött fájlról projekciós képek készülnek megadott detektorparaméterek alapján, vagyis a vizsgálatot szimulálhatjuk.

3. Képrekonstrukció

A harmadik lépés a képrekonstrukció, vagyis az eredeti aktivitáseloszlást állíthatjuk vissza a felvett projekciós képek alapján. Itt megadhatjuk, hogy milyen rekonstrukciós módszert szeretnénk használni, illetve milyen korrekciókat szeretnénk alkalmazni. A képrekonstrukciót végezhetjük az előző lépésben generált projekciós képek alapján, vagy ezek helyett használhatunk tetszőleges (megfelelő formájú), DICOM fájlból betöltött valós vizsgálatokhoz tartozó, vagy szimulációval (pl. GATE [6]) készült projekciós képeket is.

4. Rekonstruált kép kiértékelése

A rekonstruált képet az eredeti aktivitáseloszlással hasonlíthatjuk össze, így következtethetünk arra, hogy mennyire voltak hatékonyak a rekonstrukció során alkalmazott korrekciók.

A program futásához szükséges bemenő adatokat egy ini fájl tartalmazza, innen olvashatjuk be a fantom és a projekciós képek generálásához, a rekonstrukcióhoz szükséges paramétereit, illetve azt, hogy melyik funkciót szeretnénk használni.

Kimenetként DICOM, illetve txt formátumú fájlok jönnek létre: DICOM fájlként a generált fantomot, a projekciós képeket, illetve a rekonstruált képet, txt formátumban pedig a rekonstruált kép számszerű kiértékelést kaphatjuk meg.

3.2. DICOM fájlkezelés és koordinátarendszer

A DICOM (Digital Imaging and Communications in Medicine) [2] szabvány definiálja, hogy orvosi képeket milyen formában tudunk tárolni. Mivel diplomamunkám során nagyrészt DICOM fájlokkal dolgoztam, célszerű volt a DICOM szabvány által definiált koordinátarendszerhez igazodni, illetve a kapuzott szívizom SPECT vizsgálatokat leíró paraméterekkel dolgozni a fantom létrehozása, a projekciós képek generálása és a rekonstrukció során.

A DICOM fájlok egy fejléc (header), illetve egy adat (data) állományból állnak.

A **fejléc** tartalmazza azokat a paramétereit, amelyek alapján a képfeldolgozó szoftver az adatokat értelmezni, illetve a képeket térben elhelyezni tudja. A fejléc néhány kötelezően kitöltendő (pl. páciens neve, különböző azonosítók) általános paramétert és rengeteg opcionálisan kitöltendő, vizsgálat típusokra specifikus elemet (tag-et) tartalmaz.

Háromdimenziós képek esetében a legfontosabb, a kép térbeli elhelyezkedését leíró paraméterek a voxeltömb méretei, voxelek oldalhosszúsága, voxeltömb abszolút pozíciója és orientációja.

Projekciós képek DICOM fájlként való mentésekor fontos megadni a projekciós képek számát, a projekciós képek méreteit, a detektorpixelek méreteit, a detektor kezdőpozícióját, pályájának sugarát, forgásának irányát, illetve azt, hogy hány fokos szögterületben készültek a projekciós képek.

Ezenkívül esetünkben kapuzott szívizom SPECT vizsgálatokra specifikus paraméterek szerepelnek a DICOM fájlok fejlécében, a képek helyes megjelenítéséhez legfontosabb paraméter számomra az időszelekciók száma volt. A többi paramétert (alkalmazott farmakon,

időszeletek időtartama, szívfrekvencia stb.) diplomamunkám során nem használtam fel, ezeket a fájlok kiírása során átlagos, szívizomvizsgálatok során előforduló értékekkel töltöttem fel.

Az **adat** részbe kerülnek a képek (a matematikai fantom, a projekciós képek vagy a rekonstruált kép) pixel/voxelértékei.

A **DICOM koordinátarendszer** a páciens vizsgálat közbeni helyzetéhez igazodik. A 3.2.1. ábra a koordinátarendszer tengelyeit ábrázolja, ebben vesszük fel a leképezni kívánt fantomot, majd ebben mozgunk a projekciós képek készítése és a rekonstrukció során. A páciens a hátán fekszik, fejfelé előre toljuk a SPECT kamerába (head-first pozíció). Ekkor a z tengely a páciens talpa felől a feje felé (inferior-superior irány), az x tengely a jobb oldala felől a bal oldala felé (right-left irány), az y tengely pedig a páciens hasi oldala felől a háta felé (anterior-posterior irány) mutat.

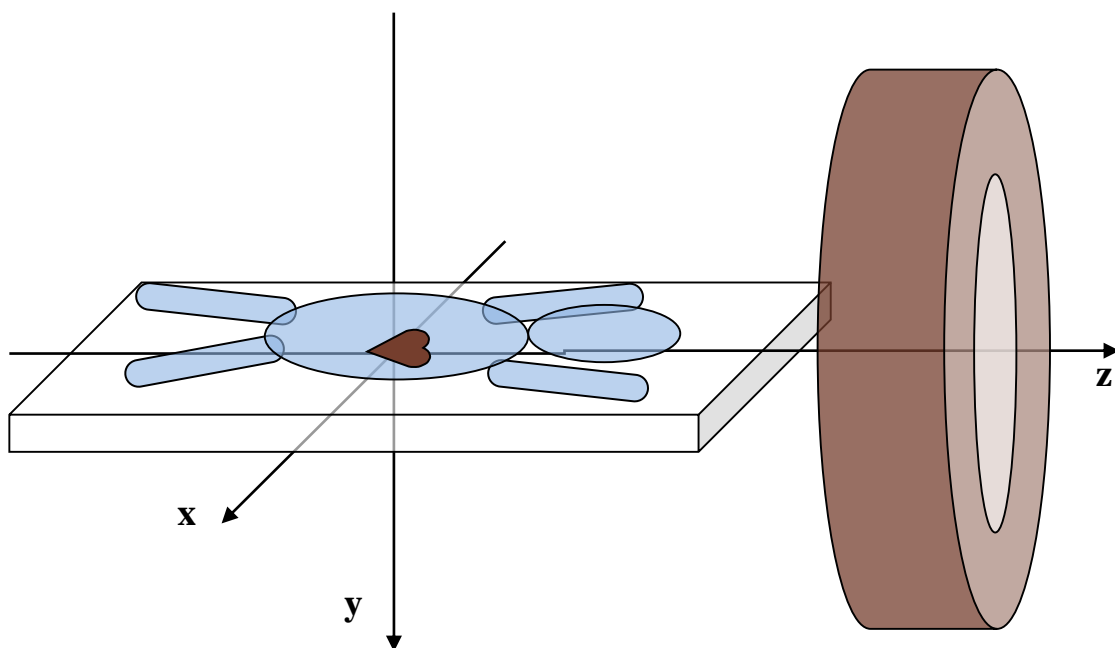
A leképezni kívánt aktivitáseloszlást minden időszeletre egy-egy háromdimenziós voxeltömbként vehetjük fel. Programkódomban a voxelek méreteit mindhárom irányban azonosnak tekintettem, a koordinátarendszer origójának a voxeltömb középpontját választottam, egységnyiinek tekintettem a voxelek oldalhosszúságát (a fantom koordinátarendszerében mozogtam). A voxelek oldalhosszára a klinikai gyakorlatban elterjedt érték $64 \times 64 \times 64$ -es tömb esetében a 6 mm, $128 \times 128 \times 128$ -es tömb esetében a 3 mm. Ezek a méretek a szívfall vastagságához (10-15 mm) képest elég nagyok. 4 és 2 mm-es oldalhosszal dolgoztam, a rekonstrukció során alkalmazott korrekciók hatása így pontosabban kiértékelhető. A szívnek megfelelő matematikai fantomot a koordinátarendszer origójába definiáltam.

A vizsgálat során a detektor a z tengely körül forog, megadható, hogy az óramutató járásával megegyező (CW – clockwise) vagy ellentétes (CC – counterclockwise) irányba, forgásközéppontja szintén az origó, a körpálya sugarát kell még megadnunk (RoR – Radius of Rotation). A DICOM szabvány szerint alapértelmezett esetben a z tengely körüli forgatáskor 0° -os szögnek a $(0,1,0)$ egységvektorral megadható irány számít. Ezt az irányt használtam a detektor kezdőpozíciójának, a szimulált vizsgálatok során általában 360° -os szögtartományban készült 64 vagy 32 vetületi kép. (A klinikai gyakorlatban 180° -os szögtartomány a jellemző, mivel a páciens jobb oldalán nagyon sokat csillapodik a sugárzás, az innen készült projekciós képek statisztikája olyan alacsony, hogy nem javítanak a másik oldalon felvett adatok alapján végzett rekonstrukció minőségén. Én olyan matematikai fantommal számoltam, ahol az egyes irányokban nem voltak csillapítás szempontjából nagy különbségek, ezért nem volt túl nagy

jelentősége annak, hogy 360 vagy 180°-os szögtartományt jár be a detektor.) A detektort 64×64 vagy 128×128 pixelből állónak vettem, a pixelek oldalhosszúsága szintén 4 mm, illetve 2 mm (mint a fantomot/pácienszt reprezentáló voxeltömb esetében).

A detektort a leképezni kívánt fantom koordinátarendszerében vettem fel. A detektorpixelek helyzete egy adott pozícióban leírható a detektor középpontjának helyével, a detektor méreteivel, illetve a kép első sorának és első oszlopának orientációjával. Ezek kiszámolhatóak minden detektorszögre az előző bekezdésben felsorolt adatokból, illetve ezek alapján már szimulálni tudjuk a leképezést (ezt a lépést az 5. fejezetben mutatom be részletesebben).

A továbbiakban a leírt, 3.2.1. ábrán látható koordinátarendszerben számoltam.



3.2.1. ábra – DICOM koordinátarendszer

3.3. Környezet

A programkódot C++ nyelven írtam, Microsoft Visual Studio fejlesztőkörnyezetben. A geometriai műveletekhez, képek megjelenítéséhez az OpenCV (Open Source Computer Vision) [3] könyvtár néhány függvényét és osztályát használtam kiegészítésként.

A DICOM fájlkezelést a DCMTK (DICOM Toolkit) [4] nyílt forráskódú DICOM fájlkezelő könyvtár függvényei segítségével valósítottam meg.

Amennyiben a statisztikai iteratív algoritmust használunk, a rekonstrukció időigényes ($64 \times 64 \times 64$ vagy $128 \times 128 \times 128$ voxelből álló képterünk van). Napjainkban elterjedt a grafikus kártyák (GPU – Graphic Processor Unit) használata, ezek párhuzamosan akár több ezer szálon tudják futtatni ugyanazt a műveletet. A rekonstrukciós algoritmus két lépésében (előrevetítés és visszavetítés, lásd 5. és 6. fejezet) egymástól függetlenül számoljuk ki a képtér voxeleinek, illetve a detektorpixelek értékeit, ezek könnyen párhuzamosítható műveletek. Az előrevetítést és visszavetítést megvalósító függvényeket grafikus kártyára is megírtam OpenCL (Open Computing Language) [5] nyelven a futási idő csökkentése érdekében. A képrekonstrukciós algoritmus CPU-n fut, a benne levő előrevetítés és visszavetítés függvények CPU-n és GPU-n is futtathatók. A CPU-n futó kódrészletekkel többet foglalkoztam (ezeket írtam meg először, majd ez alapján a GPU-n futót, a hibakeresés egyszerűbb, mint GPU-s kód esetében), a rekonstrukció tesztelése során inkább ezeket használtam. A GPU-s verzióval inkább a módszerrel való ismerkedés volt a célom (a grafikus kártyák használata elég általános képrekonstrukció esetében).

A generált DICOM fájlok – matematikai fantom, projekciós képek, rekonstruált képek orvosi képeket feldolgozó szoftver segítségével megtekinthetők és vizuálisan kiértékelhetők. Erre a Mediso Kft. által fejlesztett InterView XP –t [1] használtam.

4. Szív mozgását közelítő matematikai fantom

4.1 Saját matematikai fantom

A lehető legegyszerűbb, a szív bal kamrájának alakját és mozgását valamennyire követő modell egy hengerhøjra illesztett félgömbhøjből áll. Ilyen például a Biodex fizikai szívphantom [26], ebből indultunk ki. Egy háromdimenziós fantomot generáló programkódot felhasználva olyan kódot írtam, amely a kamra méreteinek időbeli változásait is modellezi (az időbeli változások mértéke különböző paraméterekkel állítható).

A valóságban a bal kamra alakja nem pont ilyen (inkább mintha a henger végéből egy darabot átlósan levágtak volna), a mozgása is elég összetett. A leghangsúlyosabb változás a kamra összehúzódása. A szív ciklus során a szívfal belső (endokardiális) sugara jelentősen, a külső (epikardiális) sugár kisebb mértékben csökken, ez a szívfal megvastagodását eredményezi. A bal kamra hosszában is megrövidül kissé. A szívfal helyenként más mértékben húzódik össze, attól függően, hogy milyen területről van szó (pl. a csúcs környékén jobban, a bázishoz közel kevésbé vastagodik meg), csavarodó mozgást is végez, valamint a légzőmozgások is hatással vannak rá. Egy egészséges szívfalrészlet teljesen máshogy mozog, mint egy elégtelen vérellátású.

A változások pontosabb leírására ez az egyszerű geometriájú modell nem alkalmas, csak a szívfal radiális irányú összehúzódását és a bal kamra hosszirányú megrövidülését vettem figyelembe. Diplomamunkám során a fantomgenerálás-leképezés-rekonstrukció folyamat egészét próbáltam megismerni, illetve a rekonstrukció minősége durvább közelítéseket tartalmazó modellen is tesztelhető, így nem is volt cél ennél valóságosabb szívphantom tervezése.

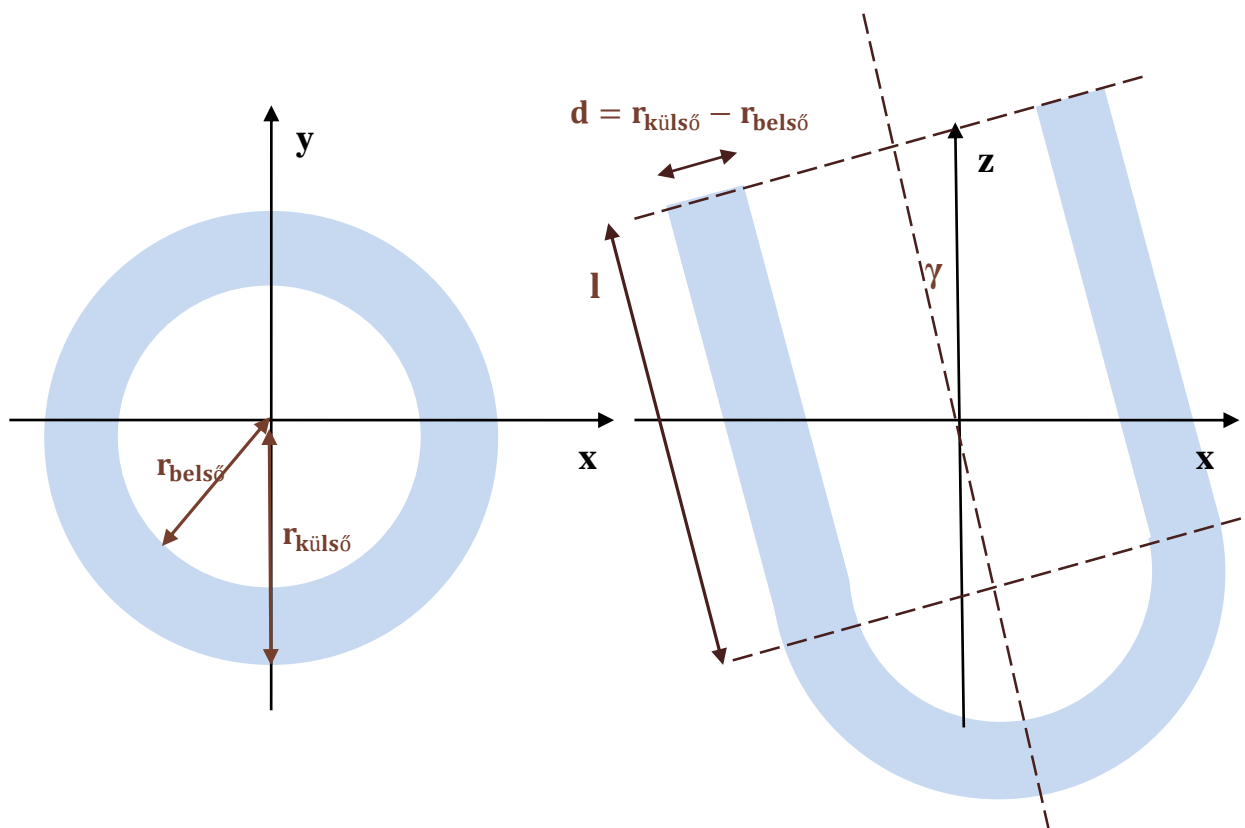
A fantom térfogatát időben úgy változtattam, hogy az körülbelül egy tipikus térfogat-idő görbe alakját (2.2.1. ábra) adja vissza. Mivel a fantom amúgy is lényeges egyszerűsítéseket tartalmaz, ennél pontosabb közelítéssel élni valószínűleg nem érdemes.

A fantom szívfalnak megfelelő részében csökkent aktivitású területek definiálhatók. Ezek szintén nem túl pontosan közelítik a szívben fellépő defektusokat (pl. a valóságban máshogy mozognának, mint az egészséges területek), de a rekonstrukció és a különböző korrekciók hatásának tesztelésére alkalmasak.

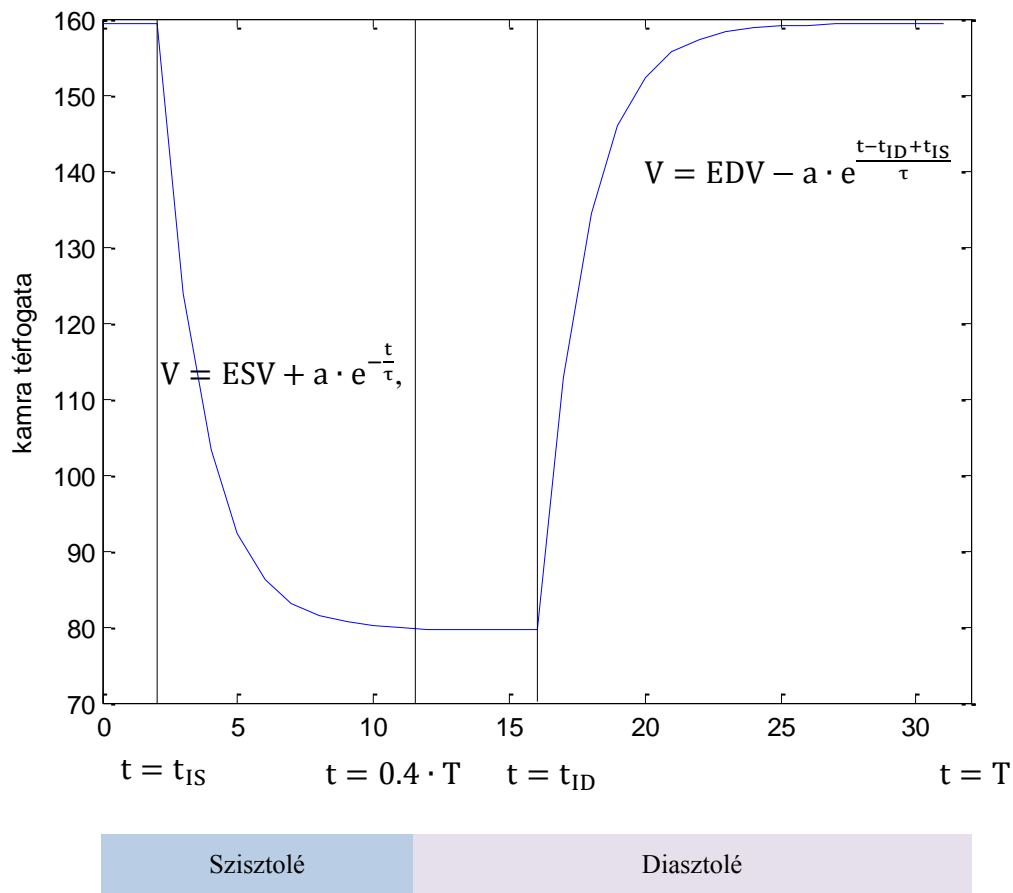
A szívizomvizsgálatok során a szívet nem önmagában vizsgáljuk, a környező szövetekkel és azok sugárzást csillapító hatásával is számolnunk kell. Leegyszerűsített modellünkben a fantomot egy vízhenger belsejébe vehetjük fel. (A lágy testszövetek vízekvivalensnek tekinthetők radioaktív sugárzás elnyelése szempontjából.) A vízhenger méretei és pozíciója a képtérben megadhatóak.

Összefoglalva a matematikai fantom az alábbiaknak tesz eleget:

- közelíti a bal kamra **radiális irányú összehúzódását** és a **szívfal megvastagodását**, az összehúzódás és a szívfal megvastagodásának mértéke megadható (ezek az egész fantomra állandók)
- modellezi a bal kamra **hosszirányú rövidülését**, ennek mértéke szintén megadható
- **a térfogat időbeli változását** egy tipikus térfogat-idő görbe alakja alapján közelíti
- **defektusok** (csökkent aktivitású területek) definiálhatók
- a páciens testét (csillapító közeget) **vízhenger** helyettesíti



4.1.1. ábra – Szívfantom paraméterezése (EDV állapot)



4.1.2. ábra – Szívfantom időbeli változásainak paraméterezése

A szívfantom hengeres részének középpontját a koordináta-rendszer origójába vettem fel. A bal kamra kicsit döntött helyzetű, ezt a henger középpontjából a félgömb középpontjába mutató vektor y tengely körüli 15° -os elforgatásával adtam meg. Diasztolé végi (a legnagyobb térfogatú) állapotban a kamra belső sugarát 24 mm-nek, külső sugarát 36 mm-nek, a hengeres rész hosszát 72 mm-nek választottam (az értékeket a Segars NCAT matematikai fantom [8] méretei alapján választottuk). Ezeket az értékeket használtam minden, a diplomamunkámban szereplő fantom esetében.

A szívvizsgálatok kiértékelése során a két legfontosabb paraméter a bal kamra térfogatváltozásának (ez alatt a belső üreget értjük) és a szívfal megvastagodásának mértéke. Ennek megfelelően én is a térfogatváltozást jellemző ejekciós frakcióval, illetve a fal vastagodásának mértékével paramétereztem a fantom térfogatváltozásának mértékét. Egy harmadik paraméterrel állítható, hogy hosszirányban mennyire húzódik össze a fantom (vagyis a térfogatváltozásért mekkora részben feleljen a sugár, és mekkora részben a hossz változása). Egy átlagos, 50% ejekciós frakcióval rendelkező fantomot generáltam, a

rekonstrukció tesztelése során legtöbbször ezt használtam. A szisztolé végi állapotban a falvastagság és a henger rész hossza a diasztolé végi érték 120%-a, illetve 90%-a volt.

A méretek időbeli változásait az alábbi módon határoztam meg:

Megadható a fantom szívfrekvenciája, ezt 75 szívverés/perc-nek vettem, ebből számolható a szív ciklus hossza: $T = \frac{60\text{ s}}{75} = 800\text{ ms}$. Állítható paraméter, hogy a ciklus hányadrészenél érje el a kamra a minimális, szisztolé végi térfogatát (ESV – end systolic volume), ez a szív nyugalmi állapotában általában $0.3 \cdot T - 0.4 \cdot T$, én $t_{ES} = 0.4 \cdot T$ -vel számoltam. A szisztolés és a diasztolés szakasz első ötödét izovolumetriásnak számítottam. A további két szakaszon a kamra térfogatát exponenciális függvény szerint változtattam: szisztolében $V = \text{ESV} + a \cdot e^{-\frac{t}{\tau}}$, diasztolében $V = \text{EDV} - a \cdot e^{\frac{t-t_{ID}+t_{IS}}{\tau}}$ függvény szerint (t_{IS} a szisztolé, t_{ID} pedig a diasztolé izovolumetriás szakaszának végét jelenti). A τ időállandó megadható, én 40 ms értéket használtam (egy tipikus idő-kamrai nyomást bemutató ábra [7] alapján, térfogatváltozásra nem találtam ilyet), az a paramétert függvényillesztéssel kapjuk meg. A kamra hosszirányú rövidülését és csökkenését a nem izovolumetriás szakaszokon, valamint a szívfall vastagodását és vékonyodását a szisztolé és diasztolé idején egyenletesnek vettem. A leírt módszer nem egészen korrekt: a térfogat változását a kamrában levő vér, és nem az üreg térfogatához illeszttem, illetve az egyenletesnek vett változások a valóságban valószínűleg nem azok. Diplomamunkám keretein belül a modellt nem pontosítottam.

A szív ciklust 8, illetve 16 időszeletre bontottam. (A rekonstrukciót leginkább a 8 időszeléből álló fantomon teszteltem a futási idő rövidítése miatt.) A fenti paraméterekből meghatározható a fantom belső és külső sugarának mérete, illetve a hengeres rész hossza mindegyik időszeletre. Ebből már megadható minden időszelletben a képtér minden voxelére, hogy része-e a szívfallnak, amennyiben igen, az adott voxel értéke a szétosztott aktivitásmennyiségnek megfelelő, egyébként 0.

A fantomot az alábbi paraméterekkel írhatjuk le (a DICOM szabvány által megadott koordinátarendszerben vesszük fel):

1. időszelletek száma
2. belső sugár mérete diasztolé végi állapotban (mm)
3. külső sugár mérete diasztolé végi állapotban (mm)
4. fantom hengeres részének hossza diasztolé végi állapotban (mm)

5. fantom abszolút helyzete a leképezendő objektumot megadó voxeltömbben (a henger középpontjának koordinátái, ez a szívciklus során állandó marad)
6. fantom orientációja (a koordináta-rendszer tengelyei körüli forgatás szöge)
7. teljes aktivitás, amelyet szétosztunk a fantomban (egyenletesen)
8. ejekciós frakció (térfogatváltozás legnagyobb térfogathoz viszonyított mértéke)
9. hosszirányú megrövidülés mértéke (legkisebb hossz legnagyobb hosszhoz viszonyított aránya)
10. fal megvastagodásának mértéke
11. a szívciklus hányadrésznél érje el a fantom a legkisebb térfogatú (szisztolé végi) állapotot
12. szívfrekvencia (szívverés/perc)
13. szisztolé/diasztolé alatti térfogatváltozás időállandója (ms)

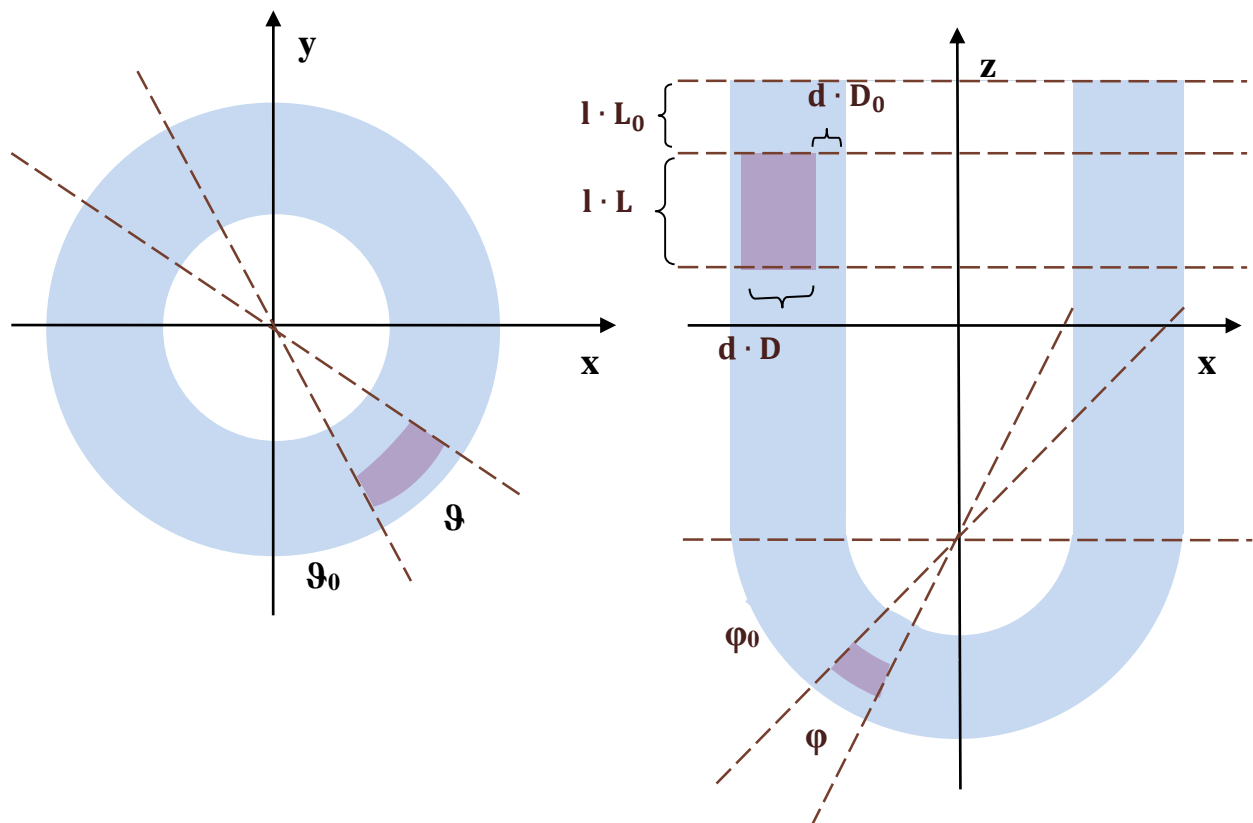
A fantomot kiindulásként a koordináta-rendszer z tengelye mentén vettem fel, a henger középpontja az origóba került (innen forgattam, illetve toltam el végleges helyére). A defektusokat ebben a pozícióban definiálhatjuk.

Megadhatjuk azt a szögtartományt, illetve amennyiben a modell henger részében vesszük fel, a henger hosszirányú tengelye mentén azt a tartományt, ahol csökkent aktivitású területet szeretnénk létrehozni.

Defektusok paraméterei:

1. a defektus típusa (csökkent aktivitású szívfalrészlet a henger vagy a félgömb részben található)
2. a defektus aktivitása hányadrésze az egészséges területek aktivitásának
3. a csökkent aktivitású terület vastagsága hányszorosa a szívfal vastagságának (D)
4. a csökkent aktivitású terület a szívfal hányadrésznél kezdődik (D_0)
5. defektust megadó szögtartomány mérete a z tengely körül (ϑ)
6. z tengely körüli szögtartomány kezdete (ϑ_0)
7. defektust megadó szögtartomány mérete a félgömb részbe eső defektus esetében, xy síkkal bezárt szög alapján (φ)
8. xy síkkal bezárt szögtartomány kezdete, csak a félgömb részbe eső defektus esetében (φ_0)

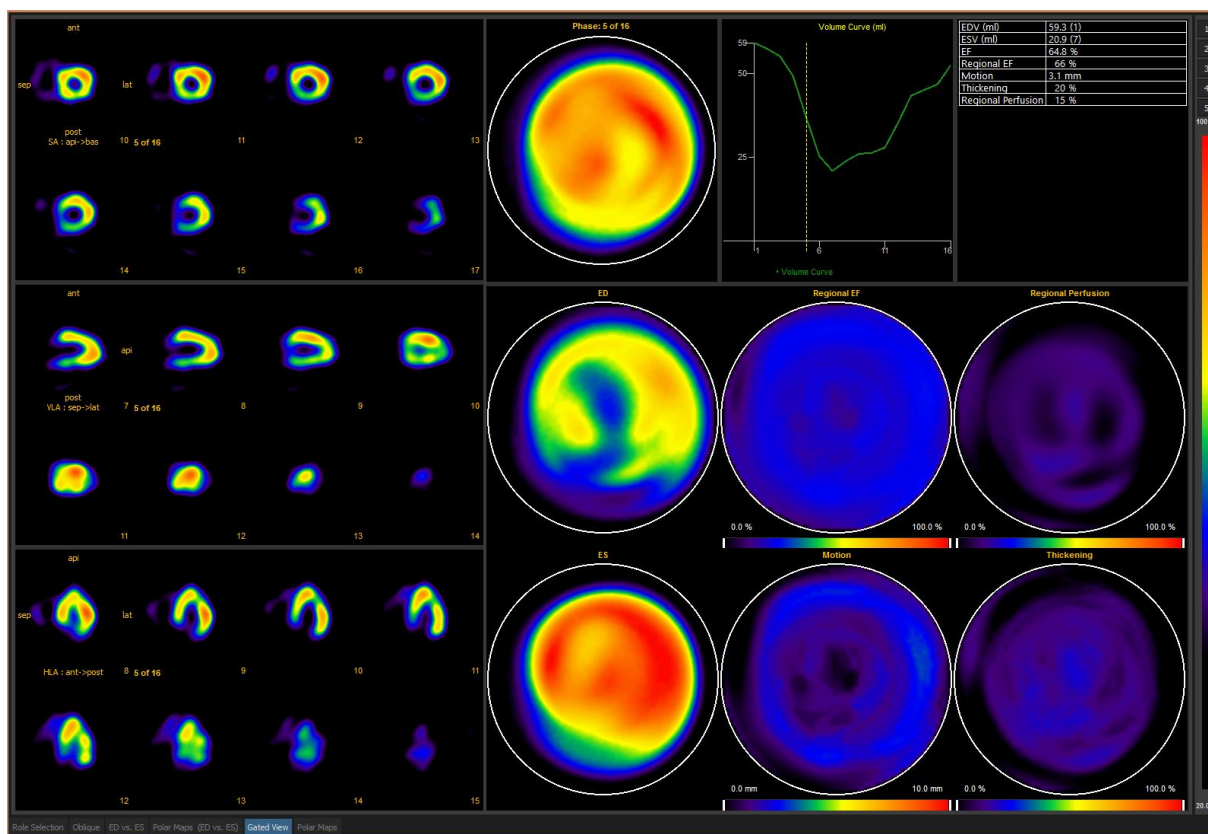
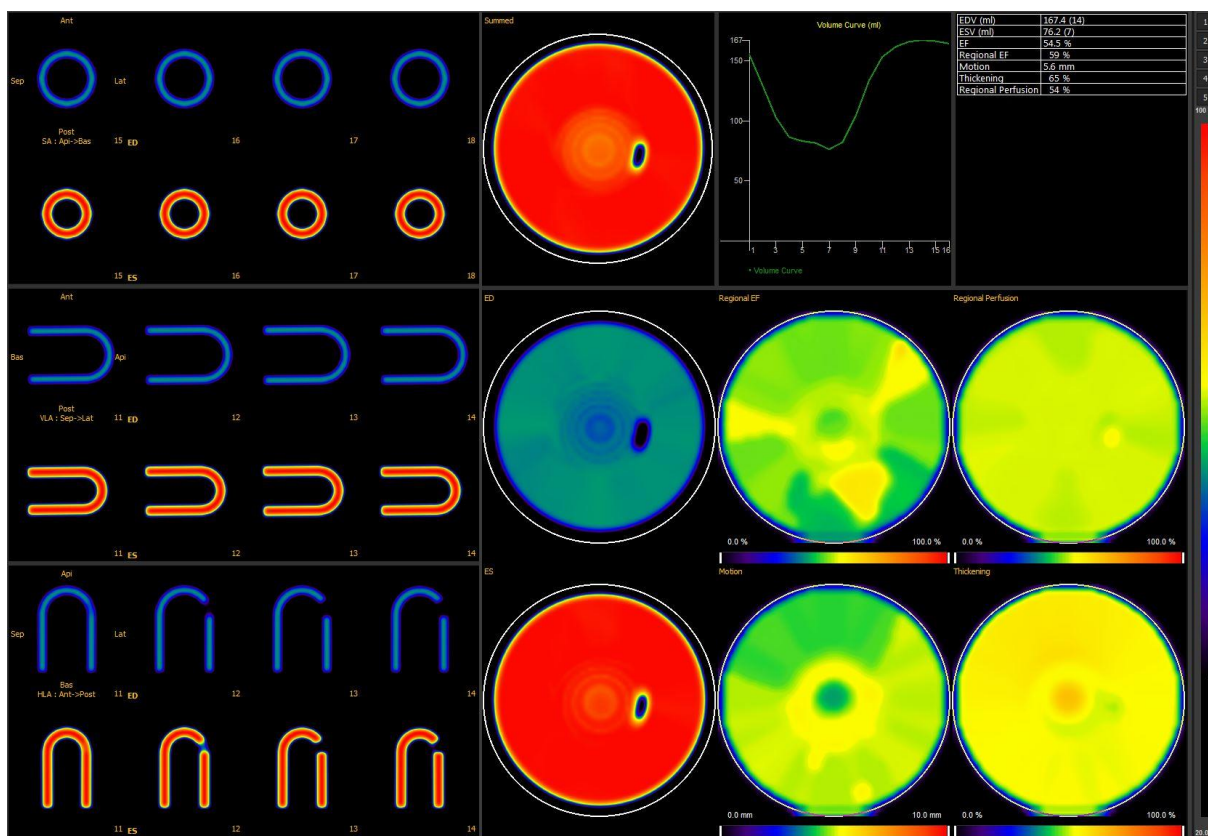
9. a defektus hossza hányszorosa a henger hosszának, csak a hengeres részbe eső defektus esetében (L)
10. a defektus kezdete a hengerben, csak a hengeres részbe eső defektus esetében (L_0)



4.1.2. ábra – Defektusok paraméterezése

4.2. NCAT szívfantom

Léteznek olyan matematikai szívfantomok, amelyek valósághűebben közelítik a szív anatómiai felépítését és mozgását, ilyen a Segars NCAT [8] fantom. A saját szívfantom mellett ilyenén is célszerű kipróbálni a rekonstrukciós algoritmust.



4.2.1. ábra – Saját matematikai fantom defektussal és valós szívizomvizsgálat

5. Projekciós képek

5.1. Leképezés fizikája

Miután létrejött a matematikai fantom, a következő lépés a vizsgálat szimulációja, vagyis meg kell határoznunk, hogy az egyes detektorpozíciókban hány foton érkezne be a használni kívánt fantomból (azaz egy adott aktivitáseloszlás alapján) az egyes detektorpixelekre. Ezt a lépést **előrevetítésnek** nevezzük.

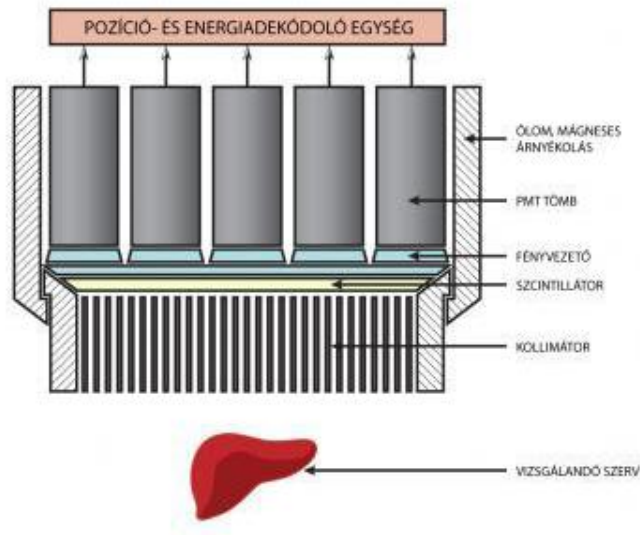
A klinikai gyakorlatban az Anger-rendszerű gamma-kamerák elterjedtek, ezekkel detektáljuk a páciensből érkező gamma-sugárzást. Három fő részből állnak:

1. A **szcintillációs detektorban** (általában NaI kristály) nyelődik el a beteg testében levő radioaktív izotópból származó fotonok egy része, ezekkel számolunk. A beérkező gamma-fotonok a szcintillációs kristályban energiájukkal arányos számú fény fotont keltenek, ezek a kristályhoz csatlakozó fotoelektródával ütköznek, amelyből fotoeffektus következtében elektronok lépnek ki.

2. Ezeket az elektronokat **fotoelektron-sokszorozókkal** (PMT – photomultiplier tube) sokszorozzuk, az így kapott jel amplitúdója a beérkező a sugárzás energiájával arányos és már elég nagy további feldolgozás számára.

3. Parallel vetítés esetén csak azokat a fotonokat szeretnénk detektálni, amelyek a detektor síkjára merőlegesen érkeztek. Erről a szcintillációs kristály elé helyezett **kollimátor** gondoskodik, ez egy lyukas ólomlemez. A detektorsíkra nem merőlegesen érkező fotonok a kollimátor falában elnyelődnek, így már nem érik el a kristályt (nem detektáljuk ezeket).

Különböző pozícióbecslő algoritmusokkal meg tudjuk becsülni, hogy a detektor felszínén hova csapódtak be az érzékelt fotonok. Így összességében azt mérjük, hogy a detektorsíkra merőleges egyenesek (továbbiakban: **válaszegyenesek**) mentén hány olyan foton keletkezett, amely a detektor síkjára merőlegesen haladt tovább (minden egyes detektorszög esetében). Mivel a gamma-sugárzás izotropnak tekinthető, ez arányos a válaszegyenes mentén keletkezett fotonok számával.



5.1.1. ábra – SPECT leképező rendszer vázlata [K3]

A valóságban egy detektorpixelben nem pontosan a hozzá tartozó válaszegyenes mentén keletkezett fotonokat detektáljuk, valamint a válaszegyenes különböző pontjain keletkezett fotonok nem ugyanakkora hányada éri el a detektort. Több fizikai hatás is közrejátszik, amelyeket a projekciós képek készítésének szimulációja során figyelembe kellene vennünk.

Sugárzás csillapodása

Annak a valószínűsége, hogy egy kibocsátott foton eljut a detektorig, függ attól, hogy milyen szöveteken kell keresztülhaladnia, illetve hogy a test felületétől mekkora távolságra keletkezett. Egy testfelszínhez közel kibocsátott foton kevesebb csillapító szöveten halad keresztül, ezért nagyobb valószínűséggel éri el a detektort, mint egy, a test belsőbb részeiből érkező foton. Ezenkívül a különböző szövetek különböző mértékben nyelik el a fotonokat, pl. a csontszövet csillapítása nagyobb, mint a légyszöveteké, vagy a levegőt tartalmazó tüdőé.

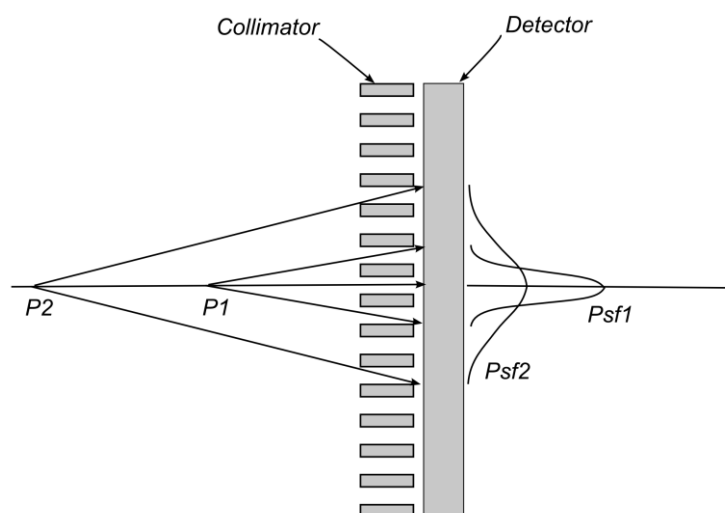
A Beer-Lambert törvény értelmében egy egyenes mentén haladó fotonnyaláb $I(x=0)$ intenzitása d távolság megtétele után

$$I(x = d) = I(x = 0) \cdot e^{-\int_{x=0}^d \mu(x) dx} \quad (5.1.1)$$

intenzitásúra gyengül, ahol x a foton egyenes menti pozícióját, $\mu(x)$ az x pontra jellemző csillapítási együtthatót jelenti.

Detektor felbontásának távolságfüggése

Egy pontforrás képe a detektoron nem pontként, hanem elmosódott foltként jelenik meg, mivel a kibocsátott fotonok nem csak a forrásból induló, a detektorsíkra pontosan merőleges egyenes mentén található kollimátorlyukakon haladhatnak keresztül és érhetik el a detektort, hanem kisebb valószínűséggel a környező lyukakba is beeshetnek. Minél távolabbról érkezik egy foton, annál elmosódottabb a képe, az elmosódás mértéke a kollimátor méreteitől is függ. A fotonok kis része a kollimátor falán is átjuthat, kisebb energiákon ez a hatás szintén elhanyagolható.



5.1.2. ábra – A detektortól különböző távolságra elhelyezkedő pontforrások képe [K4]

Szóródás

A fotonok a testen áthaladva, valamint a kollimátor falában szóródhatnak, így olyan detektorpixelben okozhatnak beütést, amely nem a foton keletkezési helyéhez tartozó válaszegyenes mentén található. Ezzel a hatással diplomamunkámban nem foglalkoztam.

Többféleképpen is létrehozhatunk a felsorolt hatásokat (illetve azok egy részét) figyelembe vevő projekciós képeket, a következő pontokban néhány lehetőséget nézünk erre.

5.2. Valós projekciós képek

Diplomamunkám során nem használtam valós, SPECT leképező rendszerrel felvett projekciós képeket, mivel egy rekonstrukciós módszert első lépésként ismert aktivitáseloszlás mellett érdemes tesztelni. Dinamikus fizikai szívphantomok léteznek, egy lehetőség lenne ilyenekről felvételeket készíteni és rekonstruálni (nekünk nem áll rendelkezésre ilyen), de első lépésként matematikai fantomokkal érdemes foglalkozni a korrekciók hatásának feltérképezése céljából.

5.3 Előrevetítés Monte Carlo alapú szimulációval

Matematikai fantomok feldolgozása esetében egy lehetőség, hogy Monte Carlo alapú szoftver (pl. [6]) segítségével készítjük a vetületi képeket. A szimuláció során megadhatjuk a leképező rendszer geometriáját, vagyis a detektor méreteit, tulajdonságait, a körülvevő szövetek tulajdonságait, a használt radioaktív anyagot. A képtér egyes pontjaiból a leképezendő fantomnak megfelelő számú fotont indíthatunk, a program ezek útját szimulálja: nagy vonalakban a leképező rendszer és a használt sugárzó izotóp tulajdonságainak megfelelő valószínűségértékeket használva a fotonoknak szabad úthosszt (ekkora távolságot tesznek meg egyenes vonalban, kölcsönhatás nélkül), majd kölcsönhatást (fotoeffektus, szóródás, párkeltés) sorsolunk. Ezt a két lépést ismétljük addig, míg a foton vagy a figyelembe vett térrészen kívülre kerül, vagy elveszti energiáját. Amennyiben a foton a szcintillációs kristályon belül veszti el energiáját, detektáljuk, a megfelelő detektorpixelhez regisztráljuk.

A módszer előnye, hogy viszonylag valósághű projekciós képeket kaphatunk, amennyiben a fellépő fizikai hatásokat jól modelleztük. Hátránya, hogy időigényes, egy szimuláció napokig tarthat.

5.4 Előrevetítő függvény modell alapú korrekciókkal

Ideális esetben (a detektor minden pontjában csak a hozzátartozó válaszegyenes mentén keletkező fotonokat detektáljuk, nincs csillapodás) egy projekciós kép adott pontbeli értékét úgy kapjuk, hogy a hozzátartozó L válaszegyenes mentén integráljuk a képtér aktivitáskoncentrációját (diszkrét esetben L mentén adott lépésközönként mintavételezzük a képteret, majd az így kapott értékeket összeadjuk), azaz kiszámoljuk L egyenes mentén a leképezni kívánt aktivitáseloszlás sugár-transzformáltját. Ha ebből a megközelítésből indulunk

ki, majd az 5.1. fejezetben bemutatott hatásokra korrekciót alkalmazunk, viszonylag gyorsan generálhatunk megfelelő minőségű projekciós képeket.

Diplomamunkám során ilyen előrevetítő függvényt írtam, ebben az 5.1. fejezetben bemutatott hatások közül a sugárzás csillapodását és a detektor felbontásának távolságfüggését vettem figyelembe.

Csillapítás figyelembevétele

Tekintsük ismertnek a leképezni kívánt fantom/páciens strukturális felépítését (hol milyen szövet található)! Ekkor meghatározható a használt izotóp fotonenergiájára a képtér minden egyes voxeléhez tartozó csillapítási együttható, a továbbiakban ezt **csillapítástérkép**nek nevezzük. Valós vizsgálatok esetében általában CT kép alapján azonosítjuk a leképezésben részt vevő struktúrákat. Programkódomban a matematikai fantomot vízhenger belsejében vettem fel, azaz olyan csillapítástérképet generáltam, ahol a hengeren belül található voxelek értéke a víz csillapítási együtthatójával egyezik meg ($\mu_{\text{víz}} = 0.15 \frac{1}{\text{cm}}$), az ezen kívül esők helyére levegőt ($\mu_{\text{levegő}} = 0 \frac{1}{\text{cm}}$) képzelünk. Amennyiben rendelkezésünkre áll CT alapján generált csillapítástérkép, vízhenger helyett ezt is használhatjuk. A továbbiakban a csillapítástérkép értékeit fel fogjuk használni a projekciós képek generálása közben.

Detektor felbontásának távolságfüggése

Az, hogy mennyire mosódik el a detektoron kapott kép, a detektor paramétereitől, a sugárzó izotóp energiájától, illetve a forrás és a detektor távolságától függ. Amennyiben egy pontforrás a detektortól d távolságra helyezkedik el, képe a detektoron kiterjedt foltként jelenik meg. A folt alakját és méretét a kollimátor geometriai tulajdonságai határozzák meg, jó közelítéssel Gauss eloszlással számolhatunk [9]. A pontforráshoz tartozó válaszegyenes és a detektor síkjának p_0 metszéspontjától t távolságra, a detektor síkján fekvő p_t pontban a pontforrás várhatóan

$$f(p_t) = f(p_0) \cdot \frac{1}{\sqrt{2\pi\sigma_d^2}} \cdot e^{-\frac{t^2}{2\sigma_d^2}} \quad (5.4.1)$$

értéket ad.

A d forrás-detektor távolsághoz tartozó gaussos elmosás σ_d paramétere jól becsülhető az alábbi formulával:

$$\sigma_d = \sqrt{\frac{\sigma_{intrinsic}^2 + (psf_a + d \cdot psf_b)^2}{2}}, \quad (5.4.2)$$

ahol psf_a és psf_b a kollimátor hatását jellemzi, $\sigma_{intrinsic}$ a kollimátor nélküli szcintillációs detektor elmosását. Ezek az értékek kalibrációs méréssel és a mért értékekre való függvényillesztéssel meghatározhatóak az egyes detektorokra és energiákra. Amennyiben ezt a modellt használjuk, a detektor geometriájával a továbbiakban nem kell foglalkoznunk, a fenti három paraméter ezeket magában foglalja.

Előrevetítő függvény megvalósítása

Programkódban az előrevetítő függvény azt határozza meg, hogy adott aktivitáseloszlás, csillapítástérkép, psf_a , psf_b és $\sigma_{intrinsic}$ paraméterek mellett egy adott detektorpozícióban mekkora aktivitásértéket mérünk az egyes detektorpixelekben, az alábbi módszerrel:

A leképezni kívánt aktivitáseloszlásra háromdimenziós voxeltömbként tekinthetünk. Az egyes voxelek középpontjának helye meghatározható, rendeljük a megfelelő voxelértékeket ezekhez a pontokhoz. Rendeljük a csillapítástérkép megfelelő értékeit is ezekhez a pontokhoz.

A detektort egy síkon fekvő pontok halmazának tekinthetjük, minden pont egy-egy detektorpixelt reprezentál, ezek egyenletes lépésközönként helyezkednek el egymástól. Vegyünk fel egy mátrixot, amelyben az egyes detektorpixelekben mért értékeket gyűjtjük!

Készítsünk el egy vetületi képet adott detektorpozíció mellett, azaz határozzuk meg a detektort képviselő mátrix elemeit!

Ideális esetben ezt úgy tehetjük meg, hogy a detektorpixelet reprezentáló pontokat a detektor síkjára merőleges irányban adott távolsággal eltoljuk, majd az így kapott pontokban az aktivitáseloszlást leíró voxeltömböt mintavételezzük – ezt lehet például trilineáris interpolációval. Az így kapott aktivitásértékkel növeljük a megfelelő detektorpixelhez tartozó

mátrixelemet, majd az eltolást és mintavételezést ismételjük addig, amíg az egész voxeltömbön át nem érünk.

Mivel a sugárzás csillapodik, egy, az adott detektorpixeltől a hozzá tartozó válaszegyenes mentén $i \cdot l$ távolságra levő pontban interpolációval kapott a_i aktivitásérték helyett csak $a_i \cdot e^{-\sum_{k=0}^i \mu_k \cdot l}$ aktivitást mérünk (az (5.1.1) Beer-Lambert törvény értelmében). Itt l a mintavételezés lépésközét, μ_k pedig a csillapítástérkép megfelelő mintavételi pontbeli értékét jelenti (a csillapítástérképet ugyanúgy mintavételezhetjük interpolációval, mint az aktivitáseloszlást megadó voxeltömböt). Ez alapján a sugárzás csillapodását úgy vehetjük figyelembe, hogy a detektorpixeleknek megfelelő mátrixelemeket minden lépésben az így módosított értékkel növeljük.

A detektor felbontásának távolságfüggésével úgy számolhatunk, hogy miután a mintavételezés a detektor eredeti pozíciójától egy adott d távolságra megtörtént, az interpolációval kapott értékekre alkalmazzuk a Beer-Lambert törvényt, majd az így kapott kétdimenziós képen (5.4.2) alapján számolt σ_d paraméterű gaussos elmosást alkalmazunk (ehhez az OpenCV könyvtár megfelelő függvényét használtam).

Megadható paraméter, hogy mekkora beütésszámmal szeretnénk projekciós képeket készíteni, a kapott pixelértékeket ennek megfelelően átskáláztam. A vizsgálatok során a képekhez random zaj is adódik, ezért végül a projekciós képekhez Poisson eloszlású zajt adtam, a zaj nélkül kapott pixelértékeknek megfelelő várható értékkel.

A bemutatott módszerrel minden detektorpozícióban, minden időszelre meghatároztam a vetületi képeket, az így kapott projekciós képsorozatot DICOM fájlként mentettem el.

A projekciós képek generálása során kihasználhatjuk, hogy az egyes detektorpixel értékek egymástól függetlenül számolhatók. Annyi módosításra van szükség, hogy az elmosást nem úgy valósítjuk meg, hogy az egyes távolságokra felvett kétdimenziós képekre alkalmazzuk (ebben az esetben a pixelértékeket nem egymástól függetlenül számítanánk). Ehelyett a képteret mintavételezzük az eltoló detektorsíkon a számolni kívánt pont adott környezetében (a pont köré rajzolható 3σ sugarú körrel számoltam) is, majd ezeket az értékeket (5.4.1) szerint súlyozva hozzáadjuk a számolt pixelértékhez.

Így az előrevetítést futtathatjuk grafikus kártyán. Olyan függvényt (kernelt) írhatunk, amely egy detektorpixel értéket számít ki a detektorpixel helyzete, a mintavételezni kívánt kép, a

csillapítástérkép, valamint a detektor elmosását megadó paraméterek alapján. A grafikus kártyán a kernelt a detektorpixelekre párhuzamosan futtatjuk (több ezer szál futhat egymás mellett), így a program jóval gyorsabban generálja a vetületi képeket.

A projekciós képek generálásához az alábbi paramétereket kell megadnunk:

Detektor elmosásának paraméterei:

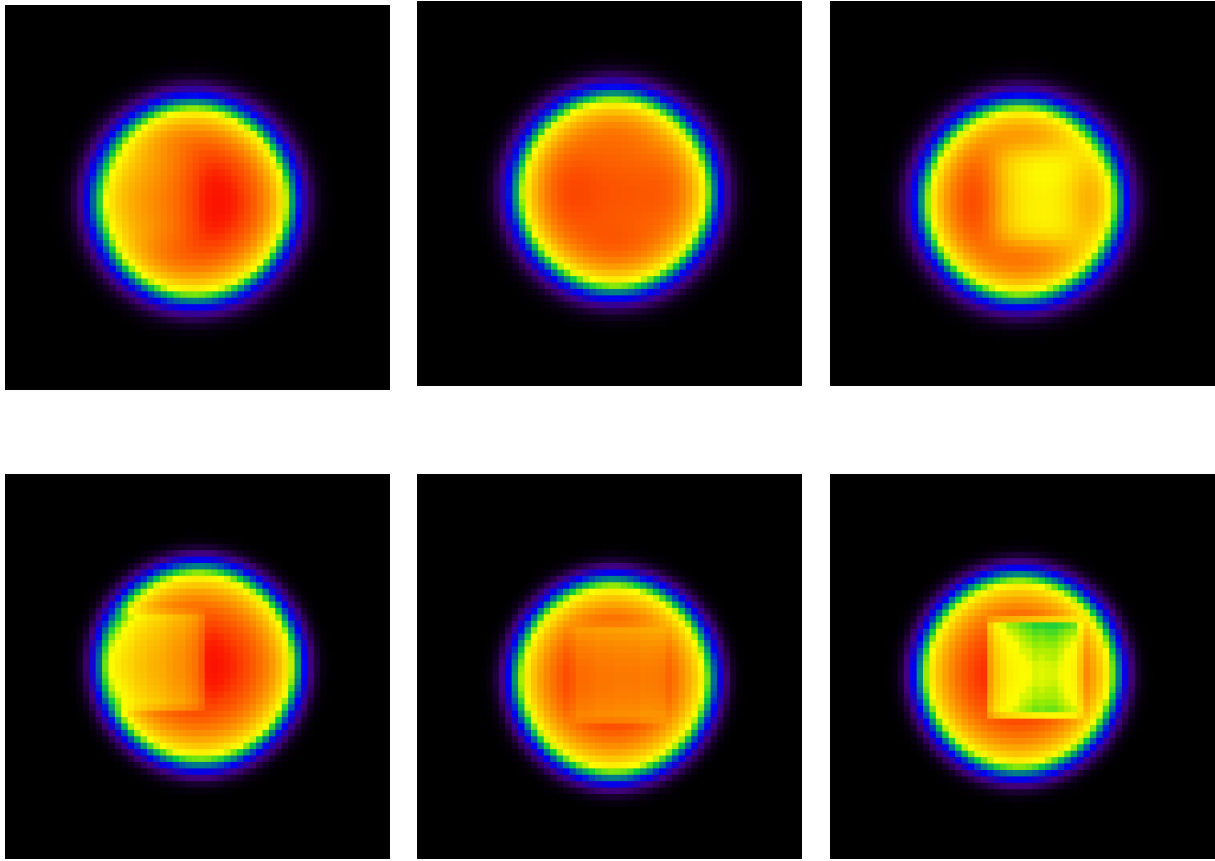
1. psf_a [mm]
2. psf_b [mm]
3. $\sigma_{\text{intrinsic}}$ [mm]

Detektor méretei, pozíciói

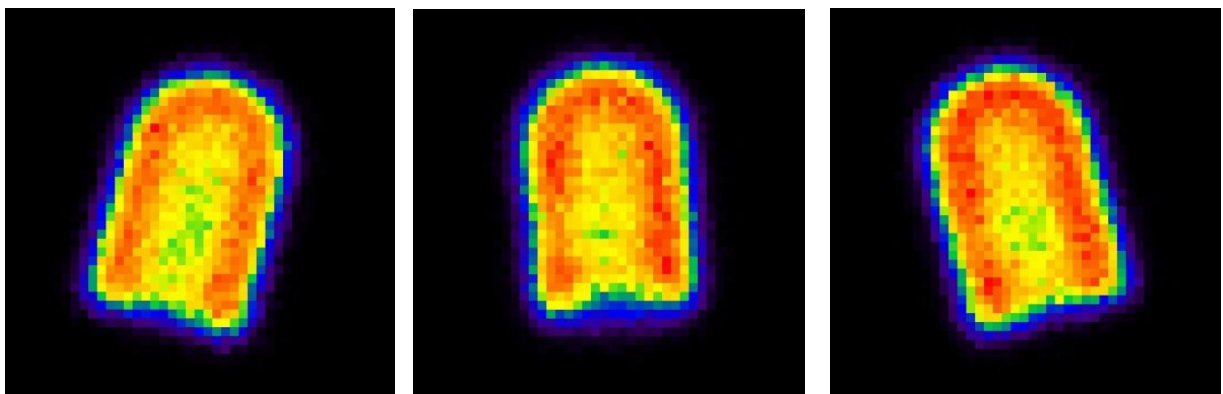
4. Detektor forgásközéppontjának koordinátái [mm]
5. Detektor forgástengelyének irányvektora (a forgás iránya jobbkézszabály szerinti)
6. Detektor sorainak és oszlopainak száma
7. Detektorpixelek oldalhossza [mm] (négyzetes pixeleket feltételezünk)
8. Szögterület, amelyben a projekciós képeket készítjük
9. Detektor kezdőpozíciójának szöge
10. Detektorpálya sugara
11. Felvenni kívánt projekciós képek száma

Projekciós képek felvételekor figyelembe vett hatások

12. amennyiben létező csillapítástérképet szeretnénk használni, a betölteni kívánt fájl neve, illetve amennyiben vízhangert szeretnénk felvenni, ennek a pozíciója és méretei
13. projekciós képek statisztikája (egy időszelethez tartozó beütések száma)



5.4.2. ábra – Henger csillapító közeget tartalmazó gömbfantomról készített vetületi képek (saját előrevezető függvénnyel, CPU-n (felül) és GPU-n (alul) futó kóddal)



5.4.3. ábra – A szívfantomról készített vetületi képek (saját előrevezető függvénnyel, CPU-n futó kóddal), Poisson statisztikájú zaj hozzáadásával

6. Képrekonstrukció és korrekciók háromdimenziós esetben

6.1 ML-EM (Maximum Likelihood – Expectation Maximization) algoritmus [10], [24]

Képrekonstrukció során az előállított vetületi képekből próbáljuk minél valóságosabban visszaállítani az azokat eredményező aktivitáseloszlást. A klinikai gyakorlatban erre a legelterjedtebb módszer a szűrt visszavetítés [11], amely analitikus megoldása az ideális esetben (csak a detektor síkjára pontosan merőleges egyenesek mentén érkező fotonokat detektáljuk minden egyes pixelben, egy beütés a detektorban azonos valószínűséggel származik a hozzá tartozó válaszegyenes bármely pontjából) kapott projekciós képeken alapuló rekonstrukciónak. Háromdimenziós esetben a leképező eszközök felépítéséből adódóan a szűrt visszavetítés megvalósítása nehézkes, egy lehetőség, hogy a háromdimenziós képet kétdimenziós szeletekből állítjuk elő, ezekhez rendeljük a detektorpixelben mért értékeket és kétdimenziós szűrt visszavetítést alkalmazunk. Viszont az egyes szeletek a valóságban nem függetlenek egymástól (más szelethez tartozó detektorpixelben is beütéseket adnak), részben az előző fejezetben bemutatott hatások miatt, ezt jó lenne figyelembe venni. Ahelyett, hogy a szűrt visszavetítésből indulunk ki, és erre próbálunk korrekciókat alkalmazni, célszerű lehet más megközelítéssel élni, például a rekonstrukciót statisztikai problémaként is kezelhetjük:

Tegyük fel, hogy ismerjük a leképező rendszer tulajdonságait, és ez alapján meg tudjuk határozni, hogy mekkora a valószínűsége annak, hogy a képtér egy adott voxeléből (jelöljük m -mel) induló foton egy adott detektorpixelben (jelöljük i -vel) beütést ad. Jelöljük ezt a valószínűségértéket a_{im} -ként. Határozzuk meg a_{im} -et minden egyes voxel-detektorpixel párra (minden detektorszög esetében), majd eredményeinket foglaljuk össze mátrixos formában, így egy, a leképező rendszert jellemző mátrixot kapunk, jelöljük ezt \bar{A} -val! A képrekonstrukció során a mért \bar{Y} projekciós képek adóttak, az ezeket eredményező \bar{X} aktivitáseloszlást keressük.

A képtér egyes voxeleinek aktivitásértékeit egymástól független valószínűségi változóknak tekinthetjük. Ekkor a leképezést a következő egyenlet írja le: $\bar{A} \cdot \bar{X} = E(\bar{Y})$, vagyis egy adott \bar{X} aktivitáseloszlás mellett az \bar{Y} vetületi képek várható értékét az \bar{A} rendszer mátrixszal való szorzás adja meg, amennyiben a képtérvoxelek x_m , illetve a detektorpixel y_i intenzitásértékeit egy-egy (\bar{X} és \bar{Y}) oszlopvektorba rendezzük. (Amennyiben a képtér M voxelből áll, valamint I detektorpixel pozíciónk van, az \bar{A} mátrix M oszlopból és I sorból áll.)

Egy lehetőség, hogy **maximum likelihood becslést** végzünk, vagyis azt az aktivitáseloszlást keressük, amely a legnagyobb valószínűséggel eredményezte a kapott projekciós képeket. Ekkor a $P(\bar{Y}|\bar{X})$ feltételes valószínűséget szeretnénk az \bar{X} paramétervektorra maximalizálni, azaz a keresett értékek: $\bar{X}_{\text{keresett}} = \underset{\bar{X}}{\operatorname{argmax}}(P(\bar{Y}|\bar{X}))$.

Vegyük figyelembe, hogy az egyes detektorpixelekbe érkezett beütések száma a radioaktív bomlás természetéből adódóan Poisson-statisztikát követ. Jelöljük az i . detektorpixelbe érkező beütések számát y_i -vel, ekkor annak a valószínűsége, hogy az adott detektorpixelbe y_i beütés érkezik:

$$P(y_i) = \frac{E(y_i)^{y_i} \cdot e^{-E(y_i)}}{y_i!} = \frac{(\sum_m(x_m \cdot a_{im}))^{y_i} \cdot e^{-\sum_m(x_m \cdot a_{im})}}{y_i!} \quad (6.1.1)$$

Az y_i beütésszám várható értékét kifejezhetjük az \bar{A} rendszermátrix és az \bar{X} paramétervektor elemei segítségével (az m . voxelben kibocsátott fotonok a_{im} része érkezik az i . detektorpixelbe, ezeket összegezve kapjuk meg, hogy várhatóan hány foton érkezik ide), ezt helyettesítettük be a 6.1.1. egyenletben:

$$E(y_i) = \sum_m(x_m \cdot a_{im}) \quad (6.1.2)$$

Mivel a voxelértékeket egymástól független valószínűségi változóknak tekinthetjük, együttes valószínűsége-sűrűség-függvényüket szorzatként írhatjuk fel. Ezt kihasználva, valamint (6.1.2) kifejezést (6.1.1)-be helyettesítve a következő alakra hozhatjuk az egyenletet, amely megadja, hogy a kapott \bar{Y} vetületi képeket mekkora valószínűséggel eredményezi egy \bar{X} aktivitáseloszlás:

$$P(\bar{Y}|\bar{X}) = \prod_i \frac{(\sum_m(x_m \cdot a_{im}))^{y_i} \cdot e^{-\sum_m(x_m \cdot a_{im})}}{y_i!} \quad (6.1.3)$$

Ezzel ekvivalens, ha a függvény logaritmusára teszünk maximum likelihood becslést, utóbbi esetben jobban kezelhető formulához jutunk:

$$\ln(P(\bar{Y}|\bar{X})) = \sum_i(y_i \cdot \ln(\sum_m(x_m \cdot a_{im})) - \sum_m(x_m \cdot a_{im}) - \ln(y_i!)) \quad (6.1.4)$$

Amennyiben hagyományos maximum likelihood becslést szeretnénk tenni (a kifejezés egyes x_k változók szerinti deriváltjait tesszük 0-val egyenlővé), egy elég összetett egyenletrendszert kapunk, az

$$\frac{\partial \ln(P(\bar{Y}|\bar{X}))}{\partial x_k} = \sum_i \left(y_i \cdot \frac{1}{\sum_m (x_m \cdot a_{im})} \cdot a_{ik} - a_{ik} \right) = 0 \quad (6.1.5)$$

egyenleteket kellene megoldanunk. Az egyenletrendszernek nincs egyértelmű megoldása (valószínűségi folyamatról van szó, nem pont a várható beütésszámokat mérjük), illetve probléma, hogy minden egyenletben változóként szerepel az összes többi voxelérték is.

Némi módosítással jobban használható valószínűségrűség-függvényhez juthatunk. Az **ML-EM (Maximum Likelihood – Expectation Maximization)** algoritmus esetében iteratív megoldást kaphatunk a problémára, amennyiben az alábbi lépéseket hajtjuk végre:

Keressünk olyan \bar{S} változókat, amelyek értékét, ha ismernénk (azaz ezeket is mérni tudnánk \bar{Y} mellett), a problémát olyan részproblémák együtteseként kezelhetnénk, amelyeknek maximum likelihood becslés esetében már egyértelmű megoldásuk van! Esetünkben a (6.1.5) egyenlet bonyolultságát az adja, hogy az $E(y_i) = \sum_m x_m \cdot a_{im}$ várható beütésszámok a hozzájuk tartozó válaszgyenesek mentén keletkezett fotonokat összesítik, ezért a $P(y_i|\bar{X})$ valószínűségek rengeteg voxelértéktől (ezek mindegyike ismeretlen értékű valószínűségi változó) függenek. Jóval könnyebb dolgunk lenne, ha azt is mérni tudnánk, hogy az y_i beütésszámok közül melyik melyik voxelből származik. Bontsuk fel a mért y_i értékeket, és jelöljük s_{im} -ként az i . detektorpixelbe az m . voxelből érkező fotonok számát és inkább ezekre tegyünk maximum-likelihood becslést (ekkor az egyenletből az m -re való szummázás eltűnik)!

$$P(s_{im}|\bar{X}) = \frac{(x_m \cdot a_{im})^{s_{im}} \cdot e^{-x_m \cdot a_{im}}}{s_{im}!} \quad (6.1.6)$$

Az így kapott valószínűségrűség-függvény már csak egy valószínűségi változót (x_m) tartalmaz ismeretlen paraméterként. Az s_{im} változókat szintén egymástól függetleneknek tekinthetjük, így a rendszert leíró $P(\bar{S}|\bar{X})$ valószínűségrűség-függvény is szorzat alakban áll elő:

$$P(\bar{S}|\bar{X}) = \prod_i \prod_m \frac{(x_m \cdot a_{im})^{s_{im}} \cdot e^{-x_m \cdot a_{im}}}{s_{im}!} \quad (6.1.7)$$

(Kihasználtuk, hogy $E(s_{im}) = a_{im} \cdot x_m$.)

Az így kapott egyszerűbb $P(\bar{S}|\bar{X})$ valószínűségi sűrűség-függvényre tehetünk maximum-likelihood becslést az összetettebb $P(\bar{Y}|\bar{X})$ helyett, az alábbi gondolatmenetet követve:

Az $\ln(P(\bar{Y}|\bar{X}))$ valószínűségi sűrűség-függvényt felírhatjuk az \bar{S} változók segítségével, itt \bar{S}_k az \bar{S} valószínűségi változók egy lehetséges megvalósulását jelenti. A képletet bővíthetjük egy $Q(\bar{S}_k)$ taggal. Amennyiben $Q(\bar{S}_k)$ az \bar{S}_k állapot megvalósulásának valószínűségét jelenti, a kifejezés felírható várható értékes formában:

$$\ln(P(\bar{Y}|\bar{X})) = \ln \sum_k P(\bar{Y}, \bar{S}_k|\bar{X}) = \ln \sum_k Q(\bar{S}_k) \cdot \frac{P(\bar{Y}, \bar{S}_k|\bar{X})}{Q(\bar{S}_k)} = \ln \left(E \left(\frac{P(\bar{Y}, \bar{S}|\bar{X})}{Q(\bar{S})} \right) \right) \quad (6.1.8)$$

A Jensen-egyenlőtlenség értelmében konkáv f függvény esetében (a továbbiakban használt valószínűségi sűrűség-függvények ilyenek)

$$f(E(X)) \geq E(f(X)). \quad (6.1.9)$$

Ezért az eredeti ML becsléssel ekvivalens, ha az alábbi függvényre teszünk maximum likelihood becslést:

$$F(Q(\bar{S}), \bar{X}) = \sum_k Q(\bar{S}_k) \cdot \ln \left(\frac{P(\bar{Y}, \bar{S}_k|\bar{X})}{Q(\bar{S}_k)} \right). \quad (6.1.10)$$

A képletben ismeretlenek az \bar{X} paramétervektor elemei, illetve a $Q(\bar{S}_k)$ valószínűségértékek. Határozzuk meg ezeket iteratívan, azaz feltételezzünk egy kezdeti \bar{X} eloszlást, majd ismételgessük a következő lépéseket (a $t+1$. iterációs lépés során az előző, t . iterációs lépésben kapott értékeket jelöljük \bar{X}^t -vel, illetve $Q(\bar{S}_k)^t$ -vel):

1. Feltételezzük \bar{X}^t aktivitáseloszlást ismertnek, végezzünk maximum likelihood becslést a $Q(\bar{S}_k)^t$ valószínűségértékek meghatározására!

$$Q(\bar{S}_k)^t = \operatorname{argmax}_{Q(\bar{S}_k)} \left(\sum_k Q(\bar{S}_k) \cdot \ln \left(\frac{P(\bar{Y}, \bar{S}_k | \bar{X}^t)}{Q(\bar{S}_k)} \right) \right). \quad (6.1.11)$$

2. Ezután feltételezzük $Q(\bar{S}_k)^t$ valószínűségértékeket ismertnek, majd végezzünk maximum likelihood becslést \bar{X} következő iterációs lépésbeli értékének (\bar{X}^{t+1}) meghatározására!

$$\bar{X}^{t+1} = \operatorname{argmax}_{\bar{X}^{t+1}} \left(\sum_k Q(\bar{S}_k)^t \cdot \ln \left(\frac{P(\bar{Y}, \bar{S}_k | \bar{X}^{t+1})}{Q(\bar{S}_k)^t} \right) \right) \quad (6.1.12)$$

Az 1. lépés eredményeként levezethető (pl. [24]), hogy $Q(\bar{S}_k)^t$ legvalószínűbb értékét akkor kapjuk, ha

$$Q(\bar{S}_k)^t = P(\bar{S}_k | \bar{X}^t, \bar{Y}). \quad (6.1.13)$$

Ezt az eredményt felhasználhatjuk a 2. lépésben:

$$\begin{aligned} F(Q(\bar{S})^t, \bar{X}^{t+1}) &= \sum_i Q(\bar{S}_k)^t \cdot \ln \left(\frac{P(\bar{Y}, \bar{S}_k | \bar{X}^{t+1})}{Q(\bar{S}_k)^t} \right) = \sum_k P(\bar{S}_k | \bar{X}^t, \bar{Y}) \cdot \ln \left(\frac{P(\bar{Y}, \bar{S}_k | \bar{X}^{t+1})}{P(\bar{S}_k | \bar{X}^t, \bar{Y})} \right) = \\ &= \sum_k \left(P(\bar{S}_k | \bar{X}^t, \bar{Y}) \cdot \ln(P(\bar{Y}, \bar{S}_k | \bar{X}^{t+1})) - P(\bar{S}_k | \bar{X}^t, \bar{Y}) \cdot \ln(P(\bar{S}_k | \bar{X}^t, \bar{Y})) \right) \end{aligned} \quad (6.1.14)$$

Az \bar{X}^{t+1} -re tett maximum likelihood becslésben a negatív előjellel szereplő tag nem játszik szerepet, illetve az első tag az $\ln(P(\bar{Y}, \bar{S}_k | \bar{X}^{t+1}))$ függvény várható értékét adja meg, amennyiben az \bar{S}_k valószínűségét $P(\bar{S}_k | \bar{X}^t, \bar{Y})$ -ként határozzuk meg. Ezeket felhasználva a (6.1.12)-ben felírt maximum likelihood becsléssel ekvivalens, ha az alábbi maximum likelihood becslést tesszük:

$$\bar{X}^{t+1} = \underset{\bar{X}^{t+1}}{\operatorname{argmax}}(E\{\ln[P(\bar{Y}, \bar{S}_k | \bar{X}^{t+1})]\}), \text{ amennyiben } P(\bar{S}_k) = P(\bar{S}_k | \bar{X}^t, \bar{Y}). \quad (6.1.15)$$

Vagyis a fent bemutatott két lépést átrendezhetjük, a továbbiakban ezt tekinthetjük az ML-EM algoritmus definíciójának:

ML-EM algoritmus

Adjunk az \bar{X} változóknak kezdeti (pozitív, nem 0) értéket, majd az alábbi két lépést ismételjük:

1. Várható érték képzés (expectation) lépés

Határozzuk meg az $E\{\ln[P(\bar{Y}, \bar{S}_k | \bar{X}^{t+1})]\}$ kifejezést, feltéve, hogy $P(\bar{S}_k) = P(\bar{S}_k | \bar{X}^t, \bar{Y})!$

2. Maximalizálás (maximization) lépés

Végezzünk az 1. lépésben meghatározott függvényre maximum-likelihood becslést \bar{X}^{t+1} paramétervektor elemei szerint!

Amennyiben az iteráció során pozitív értéktartományban maradunk (ez teljesül, ha \bar{X} elemeinek pozitív kezdeti értékeket adunk), valamint a felírt valószínűsűrség-függvényünk konkáv, az ML-EM algoritmus konvergenciája bizonyítható, az iterációs lépések számának növelésével ugyan nem biztos, hogy javítunk előző becslésünkön, de biztosan nem rontunk.

Határozzuk meg a két lépést emissziós tomográfia esetére! [10], [25]

1. Várható érték képzés („expectation”) lépés

Feltételezhetjük, hogy $P(s_{im} | x_m, y_i)$ binomiális eloszlást követ (egy, az x_m voxelben kibocsátott foton állandó valószínűséggel jut el az i . detektorpixelbe), ezért

$$P(s_{im} | y_i, \bar{X}^t) = \binom{y_i}{s_{im}} p_{im}^{s_{im}} \cdot (1 - p_{im})^{y_i - s_{im}} \quad (6.1.16)$$

Az $E\{\ln[P(\bar{Y}, \bar{S}_k | \bar{X}^t)]\}$ várható érték meghatározásával ekvivalens, ha felírjuk az $\ln[P(\bar{Y}, \bar{S}_k | \bar{X})]$ valószínűsűrség-függvényt, és $s_{im} = E(s_{im} | \bar{X}^t, \bar{Y})$ behelyettesítéssel élünk.

A binomiális eloszlás várható értéke $p_{im} \cdot y_i$. Ismeretlen a p_{im} paraméter, vagyis azt kell meghatároznunk, hogy egy adott detektorpixelben mért beütés mekkora valószínűséggel

származik a képtér egyes voxeleiből. Ezt becsülhetjük a feltételezett \bar{X}^t aktivitáseloszlás és az \bar{A} rendszermátrix alapján. Egy, az i . detektorpixelben beütést okozó foton $p_{im} = \frac{a_{im} \cdot x_m^t}{\sum_m a_{im} \cdot x_m^t}$ valószínűséggel származik az m . voxelből.

Így az alábbi kifejezést kapjuk:

$$E(s_{im} | \bar{X}^t, \bar{Y}) = \frac{a_{im} \cdot x_m^t}{\sum_m a_{im} \cdot x_m^t} \cdot y_i \quad (6.1.17)$$

$P(\bar{S}, \bar{Y} | \bar{X}^{t+1})$ meghatározása

Az s_{im} változók várható értékét paraméterezhetjük a rendszermátrix elemeivel és a keresett aktivitáseloszlás értékekkel, majd ezt behelyettesíthetjük a fenti egyenletbe:

$$E(s_{im}) = a_{im} \cdot x_m^{t+1} \quad (6.1.18)$$

(várhatóan ennyi foton érkezik az m . voxelből az i . detektorpixelbe).

Mivel \bar{S} elemeit az y_i valószínűségi változókhoz hasonlóan egymástól független, Poisson eloszlású valószínűségi változóknak tekinthetjük, az alábbi feltételes valószínűséghez jutunk:

$$P(\bar{S}, \bar{Y} | \bar{X}^{t+1}) = \prod_i \prod_m \frac{E(s_{im})^{s_{im}} \cdot e^{-E(s_{im})}}{s_{im}!} = \prod_i \prod_m \frac{(a_{im} \cdot x_m^{t+1})^{s_{im}} \cdot e^{-a_{im} \cdot x_m^{t+1}}}{s_{im}!} \quad (6.1.19)$$

Logaritmikusan formában:

$$\ln(P(\bar{S}, \bar{Y} | \bar{X}^{t+1})) = \sum_i \sum_m (s_{im} \cdot \ln(a_{im} \cdot x_m^{t+1}) - a_{im} \cdot x_m^{t+1} - \ln(s_{im}!)) \quad (6.1.20)$$

Helyettesítsük be a (6.1.17)-ben kapott eredményt a (6.1.20)-ba, így megkapjuk a maximalizálandó függvényt:

$$E \left\{ \ln \left[P(\bar{S}, \bar{Y} | \bar{X}^{t+1}) \right] \right\} =$$

$$= \sum_i \sum_m \left(\frac{a_{im} \cdot x_m^t}{\sum_m a_{im} \cdot x_m^t} \cdot y_i \cdot \ln(a_{im} \cdot x_m^{t+1}) - a_{im} \cdot x_m^{t+1} - \ln\left(\frac{a_{im} \cdot x_m^t}{\sum_m a_{im} \cdot x_m^t} \cdot y_i!\right) \right) \quad (6.1.21)$$

2. Maximalizálás („Maximization” lépés)

Ebben a lépésben a (6.1.21)-re végzünk hagyományos ML-beclést, vagyis a kifejezés egyes x_k^{t+1} paraméterek szerinti deriváltját tesszük nullával egyenlővé, azaz a képtér minden egyes k voxelére

$$\frac{\partial}{\partial x_k^{t+1}} \sum_i \sum_m \left(\frac{a_{im} \cdot x_m^t}{\sum_m a_{im} \cdot x_m^t} \cdot y_i \cdot \ln(a_{im} \cdot x_m^{t+1}) - a_{im} \cdot x_m^{t+1} - \ln\left(\frac{a_{im} \cdot x_m^t}{\sum_m a_{im} \cdot x_m^t} \cdot y_i!\right) \right) = 0 \quad (6.1.22)$$

A deriválást elvégezve:

$$\sum_i \left(\frac{a_{ik} \cdot x_k^t}{\sum_m (a_{im} \cdot x_m^t)} \cdot y_i \cdot \frac{a_{ik}}{a_{ik} \cdot x_k^{t+1}} - a_{ik} \right) = 0 \quad (6.1.23)$$

Némi egyenletrendezést követően az egyes voxelértékekre iteratív képletet kapunk, amely csak a rendszermátrix elemeit, a mért projekciós képeket és az előző lépésben becsült voxelértéket tartalmazza paraméterként:

$$x_k^{t+1} = \frac{x_k^t}{\sum_i a_{ik}} \cdot \sum_i \left(a_{ik} \cdot \frac{y_i}{\sum_m (a_{im} \cdot x_m^t)} \right) \quad (6.1.24)$$

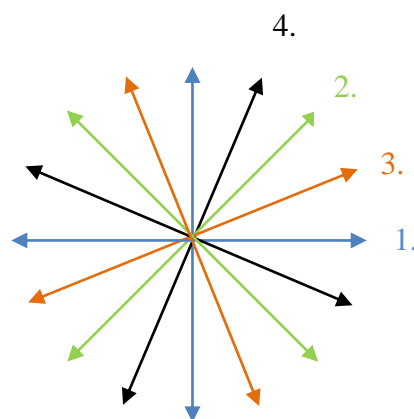
Vagyis az ML-EM algoritmus alkalmazása során feltételezünk egy kezdeti aktivitáseloszlást, majd (6.1.24) alapján frissítjük minden iterációs lépésben a voxelértékeket.

6.2. OS-EM (Ordered Subset - Expectation Maximization) algoritmus [12]

Amennyiben az ML-EM algoritmust alkalmazva egy iterációs lépés során minden projekciós képet felhasználunk és így frissítjük a feltételezett aktivitáseloszlást, az algoritmus lassan konvergál, SPECT alkalmazások esetében körülbelül 100 iterációs lépés eredményez megfelelő minőségű rekonstruált képet. A konvergenciát gyorsíthatjuk, ha a projekciós képeket kisebb részhalmazokra (subset) osztjuk, majd csak egy-egy részhalmazt felhasználva szubiterációs lépéseket végzünk

Az OS-EM algoritmus alkalmazása során a részhalmazokat úgy határozzuk meg, hogy a bennük szereplő projekciós képekhez tartozó detektorszögek egyenletesen töltsék ki a detektorpálya által meghatározott 360° -ot (vagy 180° -ot). Egy szubiterációs lépés során csak az adott részhalmazhoz tartozó projekciós képek alapján frissítjük a képet. Egy iterációs lépésnek számít, amikor az összes részhalmazon végigérünk, a részhalmazokon is úgy haladunk végig, hogy az egymás után következők egymással a lehető legnagyobb szöget zárják be. Ekkor például 4 részhalmaz esetében már 20 iterációs lépés is elég jó képet eredményez, időben pedig nem tart sokkal tovább egy iterációs lépés, mint ML-EM becslés esetében.

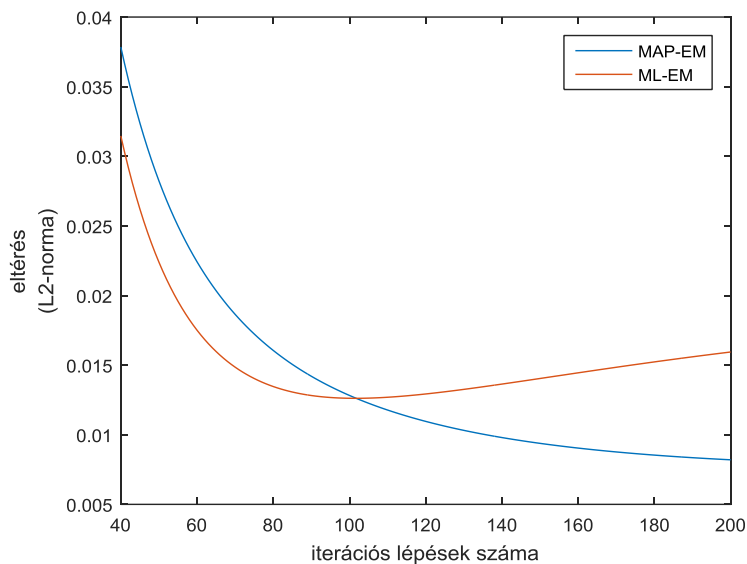
A továbbiakban nem fogok külön kitérni rá, de ahol az ML-EM algoritmust használom, alkalmazható helyette az OS-EM algoritmus is.



6.2.1. ábra – Projekciós képek rendezése (16 projekciós kép, 4 subset)

6.3. MAP-EM (Maximum A Priori – Expectation Maximization) algoritmus

Az ML-EM algoritmus problémája, hogy mivel a projekciós képek zajosak, valós vizsgálatok esetében nem a valójában keresett aktivitáseloszláshoz konvergál. Nehéz meghatározni, hogy a módszer hány iterációs lépés után adja a valósághoz legközelebb álló megoldást, ez vizsgálatonként eltérhet (pl. jel-zaj viszonytól függően). A 6.3.1. ábrán egy ML-EM algoritmussal rekonstruált kép eredeti képtől való eltéréseinek mértéke látható az iterációs lépések számának függvényében (L2-normát – lásd később – használva az eltérés mértékeként), látható, hogy körülbelül a 80. iterációs lépésig csökken az eltérés, majd növekedni kezd:



6.3.1. ábra – ML-EM és MAP-EM algoritmus konvergenciája (saját rekonstrukció)

Javíthatunk a rekonstrukció hatékonyságán (6.3.1. ábra, MAP-EM algoritmus konvergenciagörbéje), ha modellünket prior információval bővítjük: a keresett képünk ugyan ismeretlen, de tehetünk róla előrejelzéseket, pl. pontszerűen kiugró aktivitásértékek nem valószínűek, CT kép alapján az élek helyzetére tehetünk feltételezéseket (a funkcionális és strukturális képek között sok esetben összhang van), szívvizsgálatok esetében a szívfal két szomszédos pontja közel azonos irányba mozog. Ezek alapján azt próbálhatjuk megbecsülni, hogy mennyire reálisak az egyes iterációs lépések után kialakult aktivitásértékek. Ezt felhasználva olyan \bar{X} aktivitáseloszlást kereshetünk, amely nem csak nagy valószínűséggel eredményezte az \bar{Y} projekciós képeket, hanem annak a $P(\bar{X})$ valószínűsége is nagy, hogy ez egy valóságban előforduló aktivitáseloszlás.

A **MAP-EM algoritmus** [13] hasonlít az ML-EM algoritmushoz, de itt a $P(\bar{X}, \bar{Y})$ együttes sűrűségfüggvényre végzünk maximumkeresést.

A Bayes-tétel alapján $P(\bar{X}, \bar{Y}) = P(\bar{Y}|\bar{X}) \cdot P(\bar{X})$, valamint maximum likelihood becslés esetében ezzel egyenértékű, ha a kifejezés logaritmusával számolunk, így a MAP-EM módszer esetében az alábbi problémát oldjuk meg:

$$\operatorname{argmax}_{\bar{X}} (\ln(P(\bar{Y}|\bar{X})) + \ln P(\bar{X})) \quad (6.3.1)$$

Az $\ln(P(\bar{Y}|\bar{X}))$ tagra alkalmazzuk az ML-EM algoritmusnál ismertetett várható érték képzést (6.1.19) szerint! (Amennyiben az OS-EM módszert használjuk, az algoritmust MOS-EM néven szokták emlegetni.)

Szükséges még a $P(\bar{X})$ valószínűsűrűség-függvény meghatározása. Képfeldolgozás során gyakran alkalmazott modell, hogy a képre olyan valószínűségi változók halmazaként tekintünk, amelyek közül az egymással szomszédosak értéke között kapcsolat áll fenn. Esetünkben: egy voxel aktivitásértékére a szűk környezetében levő voxelek értéke hatással van, viszont a távolabb eső voxelek értéke nincs, így a kép lokális tulajdonságai alapján határozhatjuk meg egy-egy voxelérték valószínűségét. Az ilyen valószínűségi eloszlást Markov-mezőnek nevezzük.

A Hammersley-Clifford tétel alapján egy Markov-mező változóinak valószínűsűrűség-függvénye

$$P(\bar{X}) = \frac{1}{Z(\beta)} e^{-\beta \cdot V(\bar{X})} \quad (6.3.2)$$

alakú. Az ilyen valószínűsűrűség-függvénnyel leírt valószínűségi eloszlást Gibbs-eloszlásnak (vagy Boltzmann-eloszlásnak) nevezzük.

A $Z(\beta)$ tag normálás miatt szükséges (így a valószínűsűrűség-függvény integrálja az összes lehetséges állapotra 1-et ad). Mivel ez egy konstans érték (pontos értékét nem ismerjük), a maximum likelihood becslésnél nincs jelentősége, a továbbiakban elhagyhatjuk.

Helyettesítsük be $P(\bar{Y}|\bar{X})$ és $P(\bar{X})$ valószínűségi sűrűség-függvényeket (6.3.1)-be, előbbit (6.1.21), utóbbit (6.3.2) alapján! A probléma megoldása MAP-EM módszerrel annyiban változik az ML-EM algoritmushoz képest, hogy a maximalizálás lépést a (6.1.19) kifejezés helyett az alábbi valószínűségi sűrűség-függvényre végezzük el:

$$\begin{aligned} & E\{\ln(P(\bar{S}|\bar{X}^{t+1})|\bar{Y}, \bar{X}^t)\} + \ln(P(\bar{X}^{t+1})) \\ &= \sum_i \sum_m \left(\frac{a_{im} \cdot x_m^t}{\sum_m (a_{im} \cdot x_m^t)} \cdot y_i \cdot \ln(a_{im} \cdot x_m^{t+1}) - a_{im} \cdot x_m^{t+1} - \ln\left(\frac{a_{im} \cdot x_m^t}{\sum_m a_{im} \cdot x_m^t} \cdot y_i!\right) \right) - \beta \cdot V(\bar{X}^{t+1}) \end{aligned} \quad (6.3.3)$$

Ha elvégezzük a maximum likelihood becslést, majd átrendezzük az így kapott egyenletet, az alábbi kifejezést kapjuk, eszerint fogjuk frissíteni az iterációs lépésekben az egyes voxelek értékét:

$$x_k^{t+1} = \frac{x_k^t}{\sum_i a_{ik} - \beta \cdot \frac{\partial V(\bar{X}^{t+1})}{\partial x_k^{t+1}}} \cdot \sum_i \frac{a_{ik} \cdot y_i}{\sum_m (a_{im} \cdot x_m^t)} \quad (6.3.4)$$

Látható, hogy az ML-EM algoritmushoz képest az iterációs képlet a nevezőben $-\beta \cdot \frac{\partial V(\bar{X}^{t+1})}{\partial x_k^{t+1}}$ taggal bővült.

Ha a deriválást a (6.3.4)-ben a $V(\bar{X}^{t+1})$ tagra elvégeznénk, kezelhetetlen képletet kapnánk (hiszen $V(\bar{X}^{t+1})$ szintén több ismeretlen voxelértéket tartalmaz változóként). Egy gyakorlatban jól működő közelítés, hogy ezt a tagot OSL (one-step-late) módszerrel határozzuk meg [13], vagyis az előző iterációs lépésben kapott értékeket helyettesítjük be a tagfüggvény deriváltjába, így az iterációs képlet meghatározásakor ezt a tagot is konstansként kezelhetjük. Így a végleges formula, amely alapján az iterációt végezzük, az alábbi:

$$x_k^{t+1} = \frac{x_k^t}{\sum_i a_{ik} - \beta \cdot \left. \frac{\partial V(\bar{X})}{\partial x_k} \right|_{x^t}} \cdot \sum_i \frac{a_{ik} \cdot y_i}{\sum_m (a_{im} \cdot x_m^t)} \quad (6.3.5)$$

A rekonstrukció ugyanazon a módon történik, mint az ML-EM algoritmus esetében: feltételezünk egy kezdeti (nem 0) aktivitáseloszlást a rekonstruálni kívánt képtérre, majd az egyes voxelek aktivitásértékeit (6.3.5) szerint finomítjuk.

Egyelőre ismeretlenek a rendszermátrix a_{im} elemei, illetve a képünk voxelértékeinek valószínűségét meghatározó $P(\bar{X})$ valószínűségrűség-függvény, amely a $-\beta \cdot \left. \frac{\partial V(\bar{X})}{\partial x_k} \right|_{x^t}$ tagot adja.

A képletből látszik, hogy a rekonstrukciónak két kritikus pontja van:

1. A leképező rendszerünk tulajdonságait jól kell modelleznünk, hiszen ez határozza meg a rendszermátrix a_{im} elemeit.
2. Jó modellt kell találnunk az aktivitáseloszlást leíró prior valószínűségeloszlásra is – vagyis a $\beta \cdot V(\bar{X})$ tagot kellene ügyesen megválasztanunk.

6.4 A rendszermátrix elemeinek meghatározása, visszavetítő függvény

Értelmezzük a (6.3.5) képletet!

Definiáljuk az alábbi módon az előrevetítés, illetve visszavetítés műveleteket:

$$\text{Előrevetítés: } y_i = \sum_m (a_{im} \cdot x_m) \quad (6.4.1)$$

$$\text{Visszavetítés: } x_m = \sum_i (a_{im} \cdot y_i) \quad (6.4.2)$$

A (6.3.5) egyenletet felírhatjuk ezek segítségével:

A $\sum_m (a_{im} \cdot x_m^t)$ tag a becsült kép előrevetítését jelenti, az $\sum_i \frac{a_{ik} \cdot y_i}{\sum_m (a_{im} \cdot x_m^t)}$ kifejezés a projekciós kép és az előrevetített becsült kép pixelenkénti hányadosának visszavetítését. A $\sum_i a_{ik}$ tagot megkapjuk, ha csupa 1-eseket tartalmazó projekciós képeket vetítünk vissza.

Azaz az iterációt végezhetjük úgy is, hogy meghatározzuk az a_{im} mátrixelemeket és ezeket helyettesítjük a (6.3.5)-be, vagy ezzel ekvivalens, ha írunk egy előrevetítést (jelölése a továbbiakban: FP) és visszavetítést megvalósító függvényt (a továbbiakban: BP), és ezeket használjuk (ekkor nem kell kiszámolnunk a mátrixelemek konkrét értékeit). Programkódomban az utóbbi szerepel, ebben a formában az iterációs képlet így néz ki:

$$x_k^{t+1} = \frac{x_k^t}{\text{BP}(\bar{Y}_1)_k - \beta \cdot \left. \frac{\partial V(\bar{X})}{\partial x_k} \right|_{x^t}} \cdot \text{BP} \left(\frac{\bar{Y}}{\text{FP}(\bar{X}^t)} \right)_k \quad (6.4.3)$$

Itt \bar{Y}_1 a csupa 1-et tartalmazó projekciós képet jelenti.

Az előrevetítő függvényt már bemutattam az 5.4. fejezetben, a képrekonstrukció során is ugyanezt használtam.

Az előrevetítő függvényből meghatározhatóak az a_{im} mátrixelemek: ha az m . voxelben 1, egyébként csupa 0 képet vetítjük előre, az i . detektorpixelben mért érték adja az a_{im} mátrixelemet, kissé „matematikusabb” megfogalmazással élve:

Legyen az m . voxel középpontja a p_m pont, δ a mintavételezés lépésköze, \underline{n} pedig a detektor síkjának normálvektora, d a detektorsík és az m . voxel távolsága, t az i . detektorpixel középpontjának és az m . voxelhez tartozó válaszegyenes és a detektorsík metszéspontjának távolsága! $A(p)$ -vel jelöljük a p pontbeli csillapítástérkép értéket. Ekkor

$$a_{im} = 1 \cdot e^{-\sum_k A(p_m + k \cdot \underline{n} \cdot \delta) \cdot \delta} \cdot \frac{1}{\sqrt{2\pi\sigma_d^2}} \cdot e^{-\frac{t^2}{2\sigma_d^2}} \quad (6.4.3)$$

Definiáljuk a visszavetítő függvényt az előbb felírt mátrixelemeket felhasználva!

$$x_m = \sum_i a_{im} \cdot y_i = \sum_i e^{-\sum_k A(p_m + k \cdot \underline{n} \cdot \delta) \cdot \delta} \cdot \frac{1}{\sqrt{2\pi\sigma_d^2}} \cdot e^{-\frac{t^2}{2\sigma_d^2}} \cdot y_i \quad (6.4.4)$$

Ez kicsit szemléletesebben megfogalmazva azt jelenti, hogy amennyiben egy projekciós képet szeretnénk visszavetíteni, az alábbi lépéseket kellene elvégeznünk:

Határozzuk meg a projekciós képhez tartozó detektorpozícióhoz a detektorpixel középpontjait megadó pontokat, ezeket toljuk el egy lépésköznyi távolsággal a detektor síkjára merőleges irányban! Adjunk az eredeti projekciós képhez az eltolt és az eredeti detektorsík távolságának megfelelő paraméterű gaussos elmosást, majd szorozzuk meg az így kapott pixelértékeket a megfelelő válaszegyenesek mentén, az eredeti és az eltolt detektorpixel közötti úton számolható $e^{-\sum_k A(p_m + k \cdot \underline{n} \cdot \delta) \cdot \delta}$ taggal, amely a sugárzás csillapodását írja le. Az így kapott

eltolt pontbeli értéket osszuk szét a ponthoz legközelebb eső voxelek között, a pont és ezen voxelek középpontjának távolságával fordított arányban (interpoláció inverze)! Ismételjük ezt addig, amíg az egész rekonstruálni kívánt képtéren át nem toltuk a detektort!

A visszavetítő függvényt az előrevetítéshez hasonlóan szintén megírhatjuk grafikus kártyára is a program gyorsítása céljából. Ekkor a projekciós képek, a csillapítástérkép és a detektor elmosását jellemző psf_a , psf_b és $\sigma_{intrinsic}$ paraméterek adottak, olyan kernelt írunk, amely a képtér egy adott pozíciójú voxelének értékét határozza meg ezek alapján. Egy képtérvoxel értékéhez több detektorpixel is járulékot ad. Amennyiben az egyes detektorpixelreket rendelünk egy-egy szálat, előfordulhat, hogy a párhuzamosan futó szálak ugyanazt a memóriaterületet (voxelértéket) egyszerre módosítják, emiatt adatot veszthetünk. Ezért inkább a képtér voxeleihez rendeltem válaszegyeneseket (ez kis eltérés az előrevetítő függvényhez képest), azaz az alábbi lépéseket végeztem:

A számolt voxel középpontjából, a detektorsíkra merőleges egyenes mentén mintavételezzük a teret adott lépésközönként, illetve az adott mintavételi pont ($3\sigma_d$ méretű, σ_d értékét (5.4.2)-ből számolhatjuk) környezetében. Ezeket az értékeket súlyozzuk (5.4.1) szerint, majd az így kapott értéket szorozzuk a sugárzás csillapodásának megfelelő $e^{-\sum_k A(p_m+k\cdot n)\cdot\delta}$ taggal.

Diplomamunkám során inkább a CPU-s verziót használtam, grafikus kártyán próbaként futtattam néhány rekonstrukciót.

A visszavetítő függvényemet a saját előrevetítő függvényem alapján írtam (ugyanazokat a leképezés során szerepet játszó fizikai hatásokat vettem figyelembe, illetve hanyagoltam el – kérdéses, hogy mennyire jogosan és pontosan). Diplomamunkámban a saját előrevetítő függvényemmel készített projekciós képeken teszteltem a rekonstrukciót (azaz feltételeztem, hogy a leképező rendszer fizikáját helyesen modelleztük), mivel az időbeli korrekciók hatásának vizsgálata, és nem a modell helyességének ellenőrzése volt a fő cél.

6.5 A prior eloszlás becslése, TV (total variation)-norma

A rekonstrukció minőségét meghatározza, hogy hogyan választjuk meg a (6.3.5) iterációs képlet $\sum_i a_{ik} - \beta \cdot \frac{\partial V(\bar{X})}{\partial x_k} \Big|_{x^t}$ nevezőjében szereplő $\beta \cdot \frac{\partial V(X)}{\partial x_k}$ tagot. Ezt az egyes k voxelértékek valószínűségét meghatározó $P(x_k) = e^{-\beta \cdot V(x_k)}$ tagok adják.

Zárt rendszer részecskéinek állapotát meghatározó valószínűsűrség-függvény esetében $P(E) \sim e^{-\frac{E}{k \cdot T}}$ függvény adja meg egy adott E energiájú állapotban tartózkodás valószínűségét, ezzel analóg a képről állított modell [14]. A szomszédos részecskék között párkölcsönhatások jönnek létre, ezek az egyes részecskék energiájához járulékot adnak, egymástól távol levő részecskék esetében a kölcsönhatás elhanyagolható. A párkölcsönhatások járulékát egy-egy k részecske energiaállapotához a $V(X_k)$ **energiafüggvény** összesíti. Azt, hogy a párkölcsönhatások mennyire érvényesülnek (pl. a rendszerben esetlegesen fellépő többi hatáshoz képest), az $\frac{1}{k \cdot T}$ tag – vagyis a hőmérséklet – szabja meg.

A rekonstruált képről hasonlóan gondolkodhatunk: a szomszédos voxelértékek összefüggenek, minél közelebb találhatók egymáshoz, annál erősebben. A β paraméter (ez az $\frac{1}{k \cdot T}$ tagnak felel meg) esetünkben azt határozza meg, hogy a rekonstruált képet mennyire határozzák meg a voxelek között feltételezett összefüggések (azaz a kép feltételezett lokális tulajdonságai) a projekciós képekhez (ez felel meg a részecskékre ható külső erőknek) képest. A β paraméter szabadon választható, amennyiben 0, az ML-EM algoritmust kapjuk vissza. Fontos, hogy olyan β értéket válasszunk, amely mellett elsősorban a projekciós képek határozzák meg az iterációs képletet (a mért értékek biztosabb kiindulópontot jelentenek, mint a képről állított feltételezéseink), optimális értéke vizsgálatonként eltérhet. Egy másik szempont, hogy ha β túl nagy, az iterációs képlet nevezője negatívvá válhat, emiatt az iterációs lépés eredményeként negatív voxelértéket kaphatunk. Ez nem szerencsés, mivel az ML-EM és MAP-EM algoritmusok konvergenciájának feltétele, hogy az iteráció során végig a pozitív tartományban maradjunk. Az OSL kiértékelés β értékére korlátozással jár (hiszen az iterációs képlet nevezőjében emiatt egy pozitív számot vonunk le, ez [15] szerint például az alacsony statisztikájú szívvizsgálatok esetében problémát jelenthet), a [15] és [16] cikkekben az OSL kiértékelés módosítását javasolják. (Diplomamunkám során ezt nem próbáltam ki, a rekonstrukció során olyan statisztikájú projekciós képeket, büntetőfüggvényt és β értékeket használtam, amelyek mellett nem jelentek meg negatív értékek.)

A β paraméter optimális értékének megválasztása mellett egy alkalmas $V(\bar{X})$ energiafüggvényt kellene még találnunk, amely alacsony értéket ad, amennyiben egy adott voxelérték a képről állított előfeltételeinknek megfelel (ekkor $P(x_k) = e^{-\beta \cdot V(x_k)}$ értéke nagy lesz), illetve magas értéket ad, ha a voxel túlzottan eltér környezetétől. Ezt a lépést regularizációnak nevezzük.

Az egyik legegyszerűbb feltevés, hogy a képre, mint éles határokkal elválasztott elmosódottabb területekre tekintünk. A szomszédos voxelek értéke az elmosódott (SPECT képalkotás esetében hasonló működésű) területeken nem különbözik nagyon, az egymással szomszédos területek (pl. elhalt és egészséges szívizom) aktivitásértéke között elég nagy eltérés lehet. A képen pontszerűen kiugró értékek nem valószínűek. Ezért alkalmas lehet egy olyan energiafüggvény, amely pontszerűen kiugró értékek esetében magas, egyébként alacsony értéket ad.

A $V(x_k)$ energiafüggvény az $U(x_k, x_s)$ párpotenciálok súlyozott összegeként áll elő:

$$V(x_k) = \sum_{s \in S} w_s \cdot U(x_k, x_s) \quad (6.5.1)$$

Itt w_s súlyfaktor: térben minél közelebb van egymáshoz két voxel, értékük annál szorosabban függ össze. Súlyfaktorként használhatjuk a voxelek középpontjai közti euklideszi távolság reciprokát. S a k . voxel környezetét jelenti, vagyis azokat a voxeleket, amelyek értéke x_k -ra hatással van.

$U(x_k, x_s)$ meghatározása esetében kiindulhatunk abból, hogy mekkora a szomszédos voxelértékek különbsége, jelöljük ezt r -rel, ennek függvényeként írjunk fel párpotenciálokat. Két voxelérték között kis (egy tartományon belül található pontok) és nagy (tartományok határán levő pontok) eltérések valószínűek, közepes eltérések jó eséllyel zajból származnak. $U(r)$ -nek a zajra jellemző r értékek esetében kellene felvennie maximális értékét. Emellett $U(r)$ -nek néhány általánosabb feltételnek kell eleget tennie: pl. párosnak kell lennie, illetve a $P(\bar{X}, \bar{Y}) = P(\bar{Y}|\bar{X}) \cdot P(\bar{X})$ valószínűségi-sűrűség-függvénynek, amelyre maximum likelihood becslést végzünk, konkávnak kell maradnia (ekkor lesz a rekonstrukciós problémának globális megoldása). Például a [17] cikk ilyen büntetőfüggvényekre is hoz példákat. Problémát jelent, hogy a büntetőfüggvény alakjának, maximumának, a csúcs kiszélesedésének optimális értéke vizsgálatonként változó. Választhatunk olyan $U(r)$ függvényt, amelynek alakja széles határok között mozog (így sok vizsgálatra jól illeszthető) [18], azonban itt az jelenthet problémát, hogy több szabadon választható paraméterünk van, amelyek optimális megválasztása körülményes.

Egy másik, jól használható lehetőség, hogy nem a voxelértékek különbsége, hanem az adott pontbeli, különböző irányú gradiensek alapján határozzuk meg az $U(x_k, x_s)$ párpotenciálokat. A [17], illetve [19] cikkekben bemutatott **TV-norma (total variation norm)** ezt írja le. Előnye, hogy itt nincsenek plusz paraméterek, amelyeket optimalizálni kellene.

TV-norma

Számítsuk ki a képtér pontjaiban több irányba a gradienst, majd adjuk össze ezeket! Pontszerűen kiemelkedő aktivitásértékek esetében a gradiens minden irányban nagy, ezért az összeg nagyobb, mint élek voxelei esetében (itt csak néhány irányban nagyok az aktivitáskülönbségek).

Ebben az esetben két voxel között a következő párpotenciált definiálhatjuk

$$U(x_k, x_s) = \frac{x_s - x_k}{d(x_s, x_k)} \quad (6.5.2)$$

Ezeket összegezzük az egyes x_k voxelekre:

$$TV(x_k) = \sqrt{\sum_{s \in S} (U(x_k, x_s))^2} \quad (6.5.3)$$

Itt $d(x_s, x_k)$ az x_s és x_k voxelek középpontjának euklideszi távolságát jelenti, amennyiben az oldalak hosszát 1-nek vesszük:

$$d(x_{i,j,k}, x_{i+d_i, j+d_j, k+d_k}) = \sqrt{(d_i)^2 + (d_j)^2 + (d_k)^2} \quad (6.5.4)$$

S a k . voxel azon környezetét jelenti, amelybe tartozó voxelek alapján a k . voxelhez gradienst számolunk.

A $V(x)$ energiafüggvény (amelyet az egész képre írunk fel) a $TV(x_k)$ függvényekből áll elő:

$$V(x) = \sum_k TV(x_k) \quad (6.5.5)$$

A k . voxelre (x_k) felírt iterációs képletben szereplő $\frac{\partial V(X)}{\partial x_k}$ tagot kell ezek alapján meghatározni. $V(X)$ energiafüggvényből a deriválás után a k . és annak S szomszédságába tartozó x_s voxelekre (ezek szomszédsága legyen T) felírt $TV(x_k)$ és $TV(x_s)$ tagok maradnak meg, $s \in S, k \in T$.

$$\begin{aligned} \frac{\partial V(x)}{\partial x_k} &= \frac{\partial}{\partial x_k} \left(TV(x_k) + \sum_{s \in S} TV(x_s) \right) = \frac{\partial}{\partial x_k} \left(\sqrt{\sum_{s \in S} (U(x_k, x_s))^2} + \sum_{s \in S} \sqrt{\sum_{t \in T} (U(x_s, x_t))^2} \right) = \\ &= \frac{\partial}{\partial x_k} \left(\sqrt{\sum_{s \in S} \left(\left(\frac{x_s - x_k}{d(x_s, x_k)} \right)^2 \right)} + \sum_{s \in S} \sqrt{\sum_{t \in T} \left(\left(\frac{x_t - x_s}{d(x_t, x_s)} \right)^2 \right)} \right) = \\ &= \frac{-1}{\sqrt{\sum_{s \in S} \left(\left(\frac{x_s - x_k}{d(x_s, x_k)} \right)^2 \right)}} \cdot \sum_{s \in S} \left(\frac{x_s - x_k}{d^2(x_s, x_k)} \right) + \sum_{s \in S} \left(\frac{1}{\sqrt{\sum_{t \in T} \left(\left(\frac{x_t - x_s}{d(x_t, x_s)} \right)^2 \right)}} \cdot \frac{(x_k - x_s)}{d^2(x_s, x_k)} \right) \end{aligned} \quad (6.5.6)$$

Vagyis a (6.5.3) definíciót felhasználva

$$\frac{\partial V(x)}{\partial x_k} = \sum_{s \in S} \left(\frac{x_k - x_s}{TV(x_s) \cdot d^2(x_s, x_k)} \right) - \frac{1}{TV(x_k)} \cdot \sum_{s \in S} \left(\frac{x_s - x_k}{d^2(x_s, x_k)} \right). \quad (6.5.7)$$

A TV-normát tartalmazó tagot kiértékelhetjük úgy, hogy választunk egy β paramétert (értéke tetszőleges, optimalizálás szükséges), minden iterációs lépés után meghatározzuk $TV(x_k)$ értékét a képtér minden k voxelére, majd ezekből meghatározzuk a $\beta \cdot \frac{\partial V(X)}{\partial x_k}$ tagot, szintén minden egyes voxelre. Ezt helyettesítjük be a (6.3.5) iterációs képletbe.

6.6 A háromdimenziós rekonstrukció megvalósítása

A 6.1-6.5. fejezetekben azt mutattam be, hogy hogyan lehet MAP-EM (vagy MOS-EM) algoritmussal képrekonstrukciót felépíteni, időbeli korrekciók nélkül. Diplomamunkám során

először ezt programoztam le, minden egyes időszelést külön rekonstruáltam. Az így kapott rekonstruált kép szolgált referenciául az időbeli korrekciók tesztelése során.

Az előző fejezetekben a rekonstrukció elméleti alapjait próbáltam levezetni, ezek programozás szintjén végül tulajdonképpen két, viszonylag egyszerű képlet alkalmazására vezetnek: a (6.3.5) iterációs képletre és a kép feltételezett tulajdonságait magában foglaló (6.5.7) képletre lesz szükségünk.

Foglaljuk össze röviden, milyen lépéseket végzünk a háromdimenziós rekonstrukció során!

Az alábbi adatok állnak rendelkezésünkre:

1. Y projekciós képek (ezeket mértük, illetve szimulációval kaptuk) és az, hogy ezek milyen detektorpozíció mellett készültek. Általában 64 db 64×64 pixelből álló képet használtam.

2. Projekciós képek felvételekor használt detektor paraméterei:

- a) psf_a , psf_b , $\sigma_{intrinsic}$ paraméterek, amelyek azt adják meg, hogy mennyire elmosódottak a felvett képek: (lásd 5.4. fejezet), ezek kalibrációval meghatározhatóak.
- b) projekciós képek száma
- c) detektor kezdeti pozíciója
- d) detektor által bejárt szögtartomány

3. Csillapítástérkép, amely megadja, hogy a képtér egyes voxeleiben mekkora a csillapítási együttható, a vizsgált személy/fantom strukturális felépítése alapján. Ez valós vizsgálatok alapján pl. CT kép alapján számolható, matematikai fantom esetében generálható.

4. Kezdeti becsült kép, amelyből kiindulhatunk az iteratív rekonstrukció során. Én csupa 1-es értékeket tartalmazó képet használtam.

5. Rekonstrukció beállításai: milyen rekonstrukciós algoritmust és korrekciókat szeretnénk használni:

- a) csillapításkorrekció alkalmazása
- b) detektorfelbontás távolságfüggésének kompenzálása
- c) TV-norma β paramétere, amely megadja, hogy mennyire vegyük figyelembe a képről tett előfeltételezésünket: a kép elmosódott területekből áll, amelyeket élek választanak el egymástól, pontszerűen kiugró értékek nem valószínűek.

- d) iterációs lépések száma
- e) amennyiben OS-EM algoritmust használunk, a subsetek száma

6. A 6.4. fejezetben megmutattam, hogy amennyiben definiálunk egy előrevetítő (FP) és visszavetítő (BP) operátort, az iterációs képletet ezekkel is felírhatjuk:

$$x_k^{t+1} = \frac{x_k^t}{BP(Y_1)_k - \beta \cdot \left. \frac{\partial V(X)}{\partial x_k} \right|_{x^t}} \cdot BP\left(\frac{Y}{FP(x^t)}\right)_k \quad (6.6.1)$$

Az iterációs képlet számításához az alábbi főbb függvényeket írtam meg:

1. Előrevetítő függvény, amellyel egy háromdimenziós képről adott számú (kétdimenziós) vetületi képet tudunk készíteni, a detektor paramétereit, illetve a csillapítástérkép alapján (5.4. fejezet).
2. Visszavetítő függvény, amely adott vetületi képeket vetít vissza a háromdimenziós képtérbe, szintén a detektorparaméterek és a csillapítástérkép alapján (6.4. fejezet).
3. TV-normát számoló függvény, amely egy adott 3D kép minden voxeléhez kiszámítja $\left. \frac{\partial V(X)}{\partial x_k} \right|_{x^t}$ értékét az alábbi képlet alapján, a környező voxelek értékeit felhasználva, ennek mikéntjét a 6.5 fejezetben mutattam be:

$$\frac{\partial V(x)}{\partial x_k} = \sum_{s \in S} \left(\frac{x_k - x_s}{TV(x_s) \cdot d^2(x_s, x_k)} \right) - \frac{1}{TV(x_k)} \cdot \sum_{s \in S} \left(\frac{x_s - x_k}{d^2(x_s, x_k)} \right) \quad (6.6.2)$$

Az alábbi lépéseket végezzük a rekonstrukció során:

Adott egy becslésünk a rekonstruált képre. Erre alkalmazzuk a TV-normát számoló függvényt.

A következő becslt képet úgy kapjuk meg, hogy alkalmazzuk a 6.4.1. iterációs képletet, és ebbe behelyettesítjük a kapott $\left. \frac{\partial V(X)}{\partial x_k} \right|_{x^t}$ tagot.

Ezt minden időszeletre elvégezzük, majd annyiszor megismételjük, ahány iterációs lépést végezni szeretnénk.

7. Időbeli korrekciók

7.1 Lehetőségek

A 6. fejezetben bemutatott rekonstrukciós algoritmust bővíthetjük olyan korrekciókkal, amelyek figyelembe veszik azt, hogy az különböző időszeltek voxelértékei nem függetlenek egymástól. Irodalomkutatás során rengeteg cikket találok, amelyek szívizom SPECT vizsgálatok rekonstrukciójával foglalkoztak, ezek elég sokféle módszert írtak le, néhány példa, amelyek számomra érdekesnek tűntek:

Több esetben a büntetőfüggvényt idő dimenzióra is kiterjesztették, gyakran prior információként a szív mozgását becsülték, pl. [20]. A [21] cikkben a szív ciklus periodikus mivoltát kihasználva Fourier-térbeli szűréssel próbálták a rekonstrukció minőségén javítani. A [22] cikkben olyan transzformációt alkalmaztak, amely a projekciós képeket egymástól időben független részekre bontja, az így kapott projekciós képeket a már bemutatott iteratív algoritmussal rekonstruálták, majd az így kapott képekre inverz-transzformációt alkalmaztak. A [23] cikkben Fourier-komponensekre bontották fel a képeket, és hasonlóképp jártak el, mint [22] esetében.

Két módszert választottam ki és építettem be a programkódba:

1. Az egyik legkézenfekvőbb megoldásnak az tűnt, hogy az előző fejezetben bemutatott TV-normát terjesszük ki idő dimenzióra is, hiszen azt várjuk, hogy a voxelek értéke időben inkább folyamatosan, mint ugrásszerűen változik (olyan SPECT rekonstrukcióról, amely időbeli regularizációként is a TV-normát használja, nem találtam cikket).

2. A [21] cikkben az iterációs lépések között végzett Fourier-térbeli szűréssel próbálták javítani a rekonstruált képen, ez szintén könnyen beépíthető a rekonstrukciós algoritmusba.

7.2 Büntetőfüggvény kiterjesztése idő dimenzióra

A prior eloszlás becslése során az egymással szomszédos időszeltek voxelei között is összefüggéseket feltételezhetünk. A 6.5. fejezetben bemutatott TV-normát könnyen kiterjeszthetjük idő dimenzióra is, annyi változtatásra van szükség, hogy egy voxel szomszédságához hozzávesszük a szomszédos időszeltek megfelelő voxeleit is. Arra

számíthatunk, hogy a rekonstruált kép időben máshogy viselkedik, mint térben, két azonos időszelvényben található voxel között nem ugyanolyan erős az összefüggés, mint két különböző időszelvényhez tartozó voxel között. Ezt úgy vettem figyelembe, hogy a t . időszelvény k . és a $t+1$. időszelvény k . voxelének távolsága nem 1, hanem egy szabadon megválasztható értékű paraméter, ezt jelöljük a továbbiakban δ -val.

A 6.3. fejezetben leírtakhoz képest annyival módosul a TV-norma számítása, hogy két voxel távolságát az alábbi módon számítjuk:

$$d(x_{t,i,j,k}, x_{t+dt,i+di,j+dj,k+dk}) = \sqrt{(\delta \cdot dt)^2 + (di)^2 + (dj)^2 + (dk)^2}. \quad (7.2.1)$$

Az iterációs képletben szereplő $\beta \cdot \frac{\partial V(x)}{\partial x_k}$ tagot ugyanúgy számolhatjuk, mint az időbeli korrekciót nem tartalmazó esetben (6.3.3).

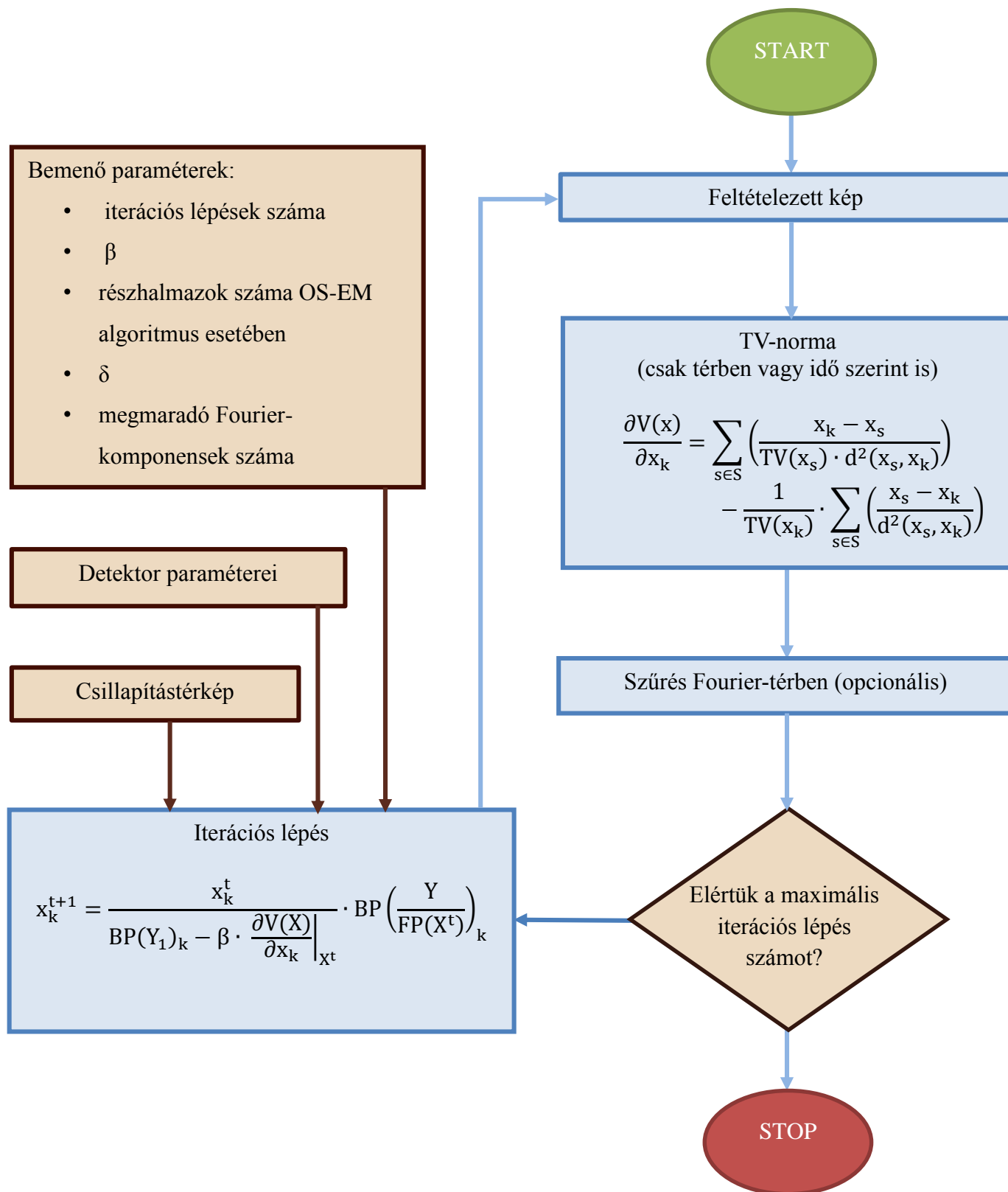
β paraméter továbbra is azt adja meg, hogy a rekonstrukciót mennyiben határozza meg a prior információ, és mennyiben a mért projekciós képek, δ pedig azt, hogy az időbeli szomszédosságot hányszor erősebben/gyengébben vegyük figyelembe, mint a térbeli szomszédosságot. (Minél nagyobb δ értéke, annál kevésbé függenek össze a különböző időszelvényekben található voxelek.)

β és δ értéke szabadon választható pozitív számok, optimális értékük vizsgálatonként eltérhet, függ az időszelvények számától, jel/zaj viszonytól stb., diplomamunkám során olyan β és δ értékeket próbáltam találni, amelyek mellett a képminőség javul.

7.3 Szűrés Fourier-térben

A [21] cikkben azt használták ki, hogy a szív mozgása periodikus. A becült kép voxelértékeit idő szerint Fourier-sorba fejtették, a magasabb rendű együtthatókat nullázták, mivel ezeket nagy részben zaj adja. Ezután inverz Fourier-transzformációt alkalmaztak, így a becült képre tulajdonképpen idő dimenzióbeli zajszűrést alkalmaztak. A Fourier-térbeli zajszűrést többféleképpen végezték: az iterációs lépések megkezdése előtt, csak az utolsó iterációs lépés után, illetve minden iterációs lépés után. 8 időszelvénytel dolgoztak, a legjobb minőségű képet akkor kapták, amikor a Fourier-térbeli szűrést minden iterációs lépés után alkalmazták.

Diplomamunkám során azt vizsgáltam, hogy mennyiben változik a rekonstruált kép minősége a Fourier-térbeli szűrés során kinullázott komponensek számától függően.



7.3.1. ábra – Képrekonstrukció időbeli korrekciókkal

8. Rekonstrukció és korrekciók kiértékelése

8.1. Kiértékelés menete, mérőszámok

A rekonstruált képek számszerű kiértékeléséhez két mérőszámot definiáltam:

1. **L2 normát** használtam, amely azt méri, hogy összességében mennyire térnek el a kapott kép voxelértékei az eredeti kép voxelértékeitől, minél kisebb ez az érték, a rekonstruált kép ($X_{\text{rekonstruált}}$) annál jobban közelíti az eredetit (X_{eredeti}):

$$L2(X_{\text{eredeti}}, X_{\text{rekonstruált}}) = \frac{\sum_{m=0}^M \left(\frac{X_{\text{eredeti},m} - X_{\text{rekonstruált},m}}{\text{norm}} \right)^2}{\sum_{m=0}^M X_{\text{eredeti},m}^2} \quad (8.1.1)$$

$$\text{norm} = \frac{\sum_{m=0}^M X_{\text{rekonstruált},m}}{\sum_{m=0}^M X_{\text{eredeti},m}} \quad (8.1.2)$$

2. Fontos, hogy a **szívfal vastagságát** helyesen adja vissza a kép (hiszen a valós vizsgálatok során az orvos számára ez fontos információt hordoz), így ezt is meghatároztam mind az eredeti, mind a rekonstruált képen. Ezt félértékszélesség alapján állapítottam meg: egy, a szívphantom hossz tengelyére merőleges irány mentén megkerestem a maximális voxelértéket, majd azokat a voxeleket számítottam a szívfalhoz tartozónak, amelyek értéke ennek a felénél nagyobb volt. Ezek közül a két szélső távolsága adja a szívfal vastagságát, a továbbiakban ennek jelölése WT (wall thickness) lesz.

A rekonstrukciós programkód egyes részletei nincsenek optimalizálva, ezenkívül sok szabad paraméterünk van, például: β , δ , OS-EM algoritmus esetében a subsetek száma. Ezek optimális értéke nem független egymástól, illetve függ a rekonstruálni kívánt projekciós képektől is (ezek statisztikájától, projekciós képek számától, időszeltek számától, képtér és detektorpixelek méretétől stb.), ezen kívül a matematikai fantom paramétereinek függvényében is lehetne például vizsgálni a rekonstrukció hatékonyságát. Ez rengeteg tesztet jelentene. Diplomamunkám során leginkább az időbeli korrekciók hatásának vizsgálata volt a cél, ezért az ehhez nem kapcsolódó paramétereknek körülbelül optimális értéket kerestem, majd az időbeli korrekciókhoz kapcsolódó paramétereket változtatva vizsgáltam a rekonstruált képet. Egy 50% ejekciós frakcióval rendelkező saját matematikai fantomon teszteltem leginkább az

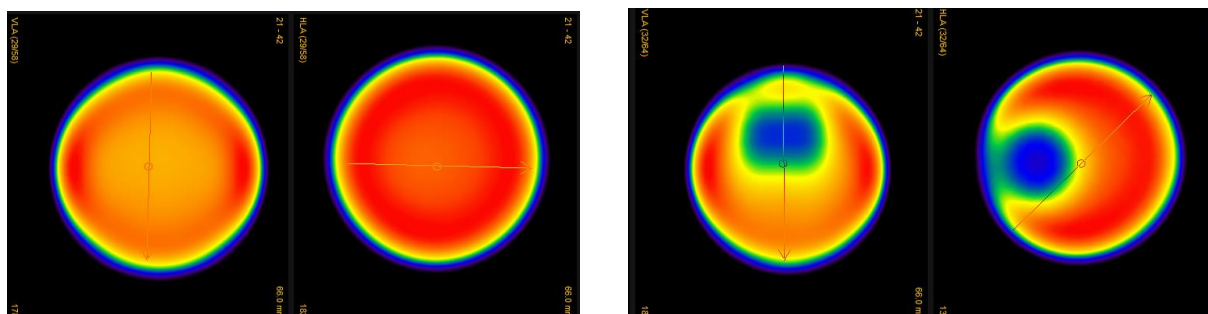
algoritmust, négyféle statisztikájú projekciós képsorozaton végeztem rekonstrukciót: egy időszelethez 800000, 1600000, 3200000, illetve 6400000 beütés tartozott, 64 projekciós képet vettem fel (a klinikai gyakorlatban a 6400000 körüli beütésszám reálisnak számít). A tesztelés lépéseit az 50% ejekciós frakciójú fantomon mutatom be, ezen kívül egy 70% és egy 30% ejekciós frakcióval rendelkező saját fantomot rekonstruáltam még 6400000 időszeletenkénti beütésszám mellett, ezeket az eredményeket csak röviden ismertetem.

8.2. ML-EM/OS-EM algoritmus tesztelése időbeli korrekciók nélkül

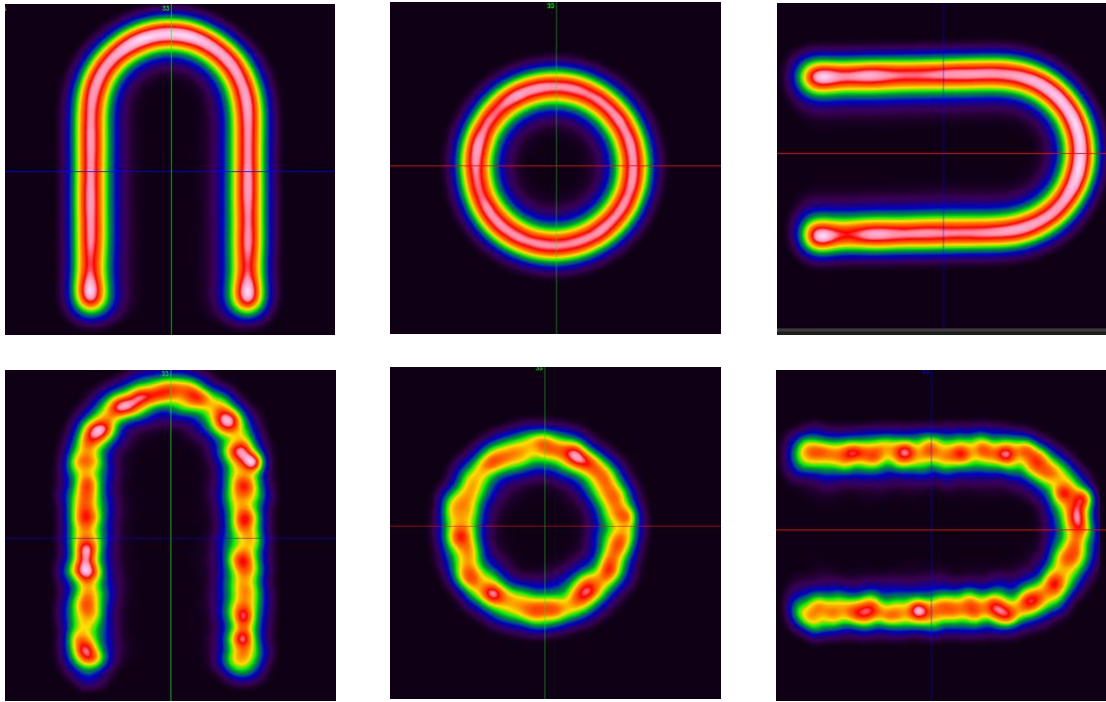
A rekonstrukció tesztelése során saját előrevezető függvénnyel készült projekciós képeket használtam, tehát azt feltételeztem, hogy a leképező rendszerről alkotott modell helyes, csak a radioaktív bomlás bizonytalanságából származó Poisson-zaj jelenik meg a projekciós képeken. A továbbiakban tesztelni kellene azt is, hogy a modell mennyire jól közelíti a valóságot, hogy teljesít más rekonstrukciós algoritmusokhoz képest, diplomamunkám keretein belül ezt nem tettem meg.

Egy defektust nem tartalmazó, 1 időszeletes szívphantomot (paraméterezése a 4. fejezetben) és gömbfantomot generáltam, majd ezekről különböző statisztikájú projekciós képeket készítettem. Az így kapott vetületi képek alapján végeztem rekonstrukciót.

A térbeli korrekciók (regularizáció, sugárzás csillapodásának figyelembevétele, detektor felbontóképességének távolságfüggése) működését teszteltem, ezek alkalmazásával a rekonstruált kép láthatóan javult:

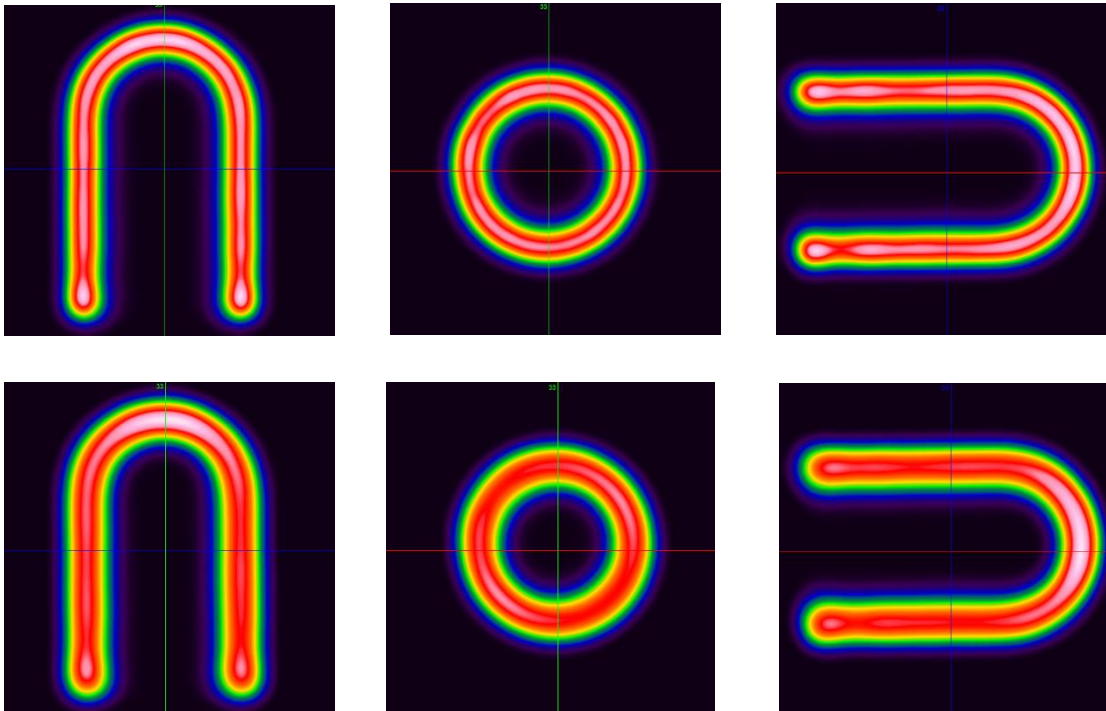


8.2.2. ábra – A belsejében henger alakú csillapító közeget tartalmazó gömbfantom rekonstruált képe csillapításkorrekció alkalmazásával (bal oldalon) és anélkül (jobb oldalon)



8.2.3. ábra – Rekonstruált kép a detektor felbontóképességének távolságfüggését kompenzálva (felül), illetve anélkül (alul)

A TV-norma alkalmazásának hatása inkább a rekonstruált kép számszerű kiértékelésnél látszik, (képkértékeléshez kevésbé szokott) szemmel jól látható hatása a nagyon rosszul beállított β paraméternek van.



8.2.4. ábra – Térbeli regularizáció hatása β körülbelül optimális (felül) és ettől erősen eltérő (alul) értéke mellett, utóbbi elmosódottabb szívfalat ad a valósnál

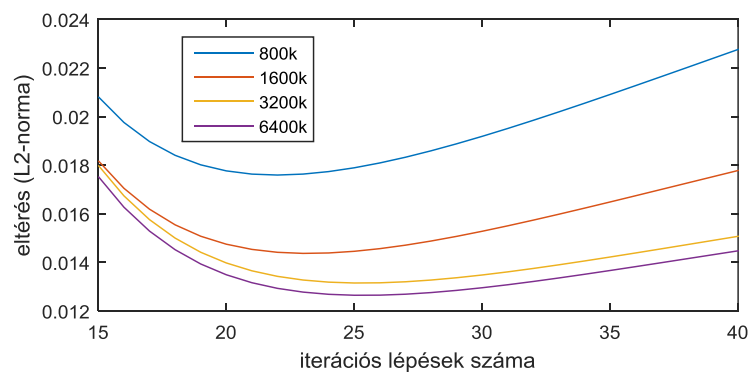
8.3. OS-EM algoritmus paramétereinek megválasztása időbeli korrekciók nélkül

A rekonstrukció során az alábbi paramétereket kell megadnunk:

- **subsetek száma** (OS-EM algoritmus esetében, a rekonstrukció gyorsítható így)
- **iterációs lépések száma**
- β – TV-norma alkalmazása esetében mekkora súllyal számítsen a prior információ a rekonstrukció során
- δ – időbeli TV-norma alkalmazása során mennyire szoros kapcsolat álljon fenn a szomszédos időszakok között (minél nagyobb δ , annál kevésbé függenek össze)
- **megmaradó Fourier komponensek száma**, amennyiben Fourier-térbeli szűrést szeretnénk alkalmazni idő szerint

Az időbeli korrekciók hatásának tesztelése során szükség lesz referenciául szolgáló, időbeli korrekciókat nem tartalmazó rekonstrukcióra. A subsetek számának, az iterációs lépések számának, illetve a büntetőfüggvény hatását megadó β paraméternek kerestem körülbelül optimális értékeket többféle statisztikájú projekciós képsorozat mellett, egy időszakból álló (statikus) fantomot használtam 12 mm-es falvastagsággal. A subsetek számát 4-nek választottam, korábbi tapasztalatok alapján.

Ezután $\beta=0$ mellett (térbeli regularizáció nélkül) futtattam a rekonstrukciós algoritmust egy statikus szívphantomra, minden iterációs lépés után megnéztem, hogy L2-norma szerint mennyire közelíti a rekonstruált kép az eredeti képet. Ebből megállapítottam az iterációs lépések optimális számát. A projekciós képek statisztikájától csak kicsit függött az iterációs lépések optimális száma, mind a négy esetben körülbelül 20-at kaptam.



8.3.1. ábra – OS-EM algoritmus konvergenciája az iterációs lépések számának függvényében

statisztika	iteráció	18	19	20	21	22	23	24	25
800k	L2	0.018	0.0178	0.0176	0.0176	0.0176	0.0178	0.0179	0.0181
	WT	11.6	11.6	11.2	11.2	11.2	11.2	11.2	11.2
1600k	L2	0.0151	0.0147	0.0145	0.0141	0.0144	0.0144	0.0145	0.0146
	WT	11.6	11.6	11.6	11.6	11.6	11.2	11.2	11.2
3200k	L2	0.0144	0.014	0.0136	0.0134	0.0133	0.0132	0.0131	0.0132
	WT	12	12	12	11.6	11.6	11.6	11.6	11.6
6400k	L2	0.0139	0.0135	0.0131	0.0129	0.0128	0.0127	0.0126	0.0126
	WT	12	11.6	11.6	11.6	11.2	11.2	11.2	11.2

8.3.1. táblázat – Rekonstruált kép minősége az iterációs lépések függvényében

Ezután különböző β értékek mellett végeztem rekonstrukciót (itt már 4 subsetet és 20 iterációs lépést használtam), majd néztem, hogy milyen β paraméter mellett kapjuk a legpontosabb rekonstruált képet. β optimális értéke a projekciós képek statisztikájától függően szintén csak kicsit változott, a vizsgált tartományban nem tapasztaltam nagy eltéréseket (minél magasabb statisztikával rendelkeztek a képek, annál kevésbé volt hatása a térbeli regularizációnak). β értékét 0.05-nek választottam, statisztikától függetlenül.

statisztika	β	0	0.01	0.02	0.03	0.04	0.05	0.06	0.07
800k	L2	0.0176	0.0173	0.017	0.0169	0.0168	0.0167	0.0167	0.0168
	WT	11.2	11.6	11.6	12	12	12.4	12.4	12.4
1600k	L2	0.0145	0.0144	0.0142	0.0141	0.0139	0.0139	0.0139	0.014
	WT	11.6	11.6	11.6	11.6	11.6	12	12	12
3200k	L2	0.0136	0.0137	0.0136	0.0136	0.0135	0.0136	0.0136	0.0137
	WT	12	12	12	12	12	12	12	12.4
6400k	L2	0.0131	0.0133	0.0132	0.0132	0.0131	0.0131	0.0131	0.0131
	WT	11.6	12	12	12	12	12	12	12

8.3.2. táblázat – Rekonstruált kép minősége β függvényében (20 iterációs lépés)

8.4. Időbeli regularizáció hatása a rekonstruált kép minőségére

Időbeli regularizáció alkalmazása esetében a δ paraméternek adhatunk értéket (ezzel kisebb/nagyobb súllyal figyelembe vesszük az egymással szomszédos időszeletek voxelértékei közti összefüggéseket is). Azt vizsgáltam, hogy ez javít-e a rekonstruált kép minőségén, illetve ha igen, mekkora δ érték mellett kapjuk a legjobb minőségű képet. A 8.3. fejezetben megállapított paramétereken nem változtattam.

Ehhez egy 8 időszeletes szívphantomot generáltam (szintén a 4. fejezetben leírt paraméterezéssel), erről készítettem különböző statisztikájú vetületi képeket a saját előrevetítő függvényvel, majd ezek alapján rekonstruáltam különböző δ mellett az eredeti aktivitáseloszlást.

statisztika	δ	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7
800k	L2	0.0362	0.0439	0.0278	0.0256	0.026	0.0269	0.0276	0.0283
1600k	L2	0.0244	0.0406	0.0224	0.0193	0.019	0.0192	0.0196	0.0199
3200k	L2	0.0196	0.0406	0.0207	0.0173	0.0167	0.0167	0.0168	0.0169
6400k	L2	0.0172	0.035	0.0178	0.0154	0.015	0.0151	0.0152	0.0153

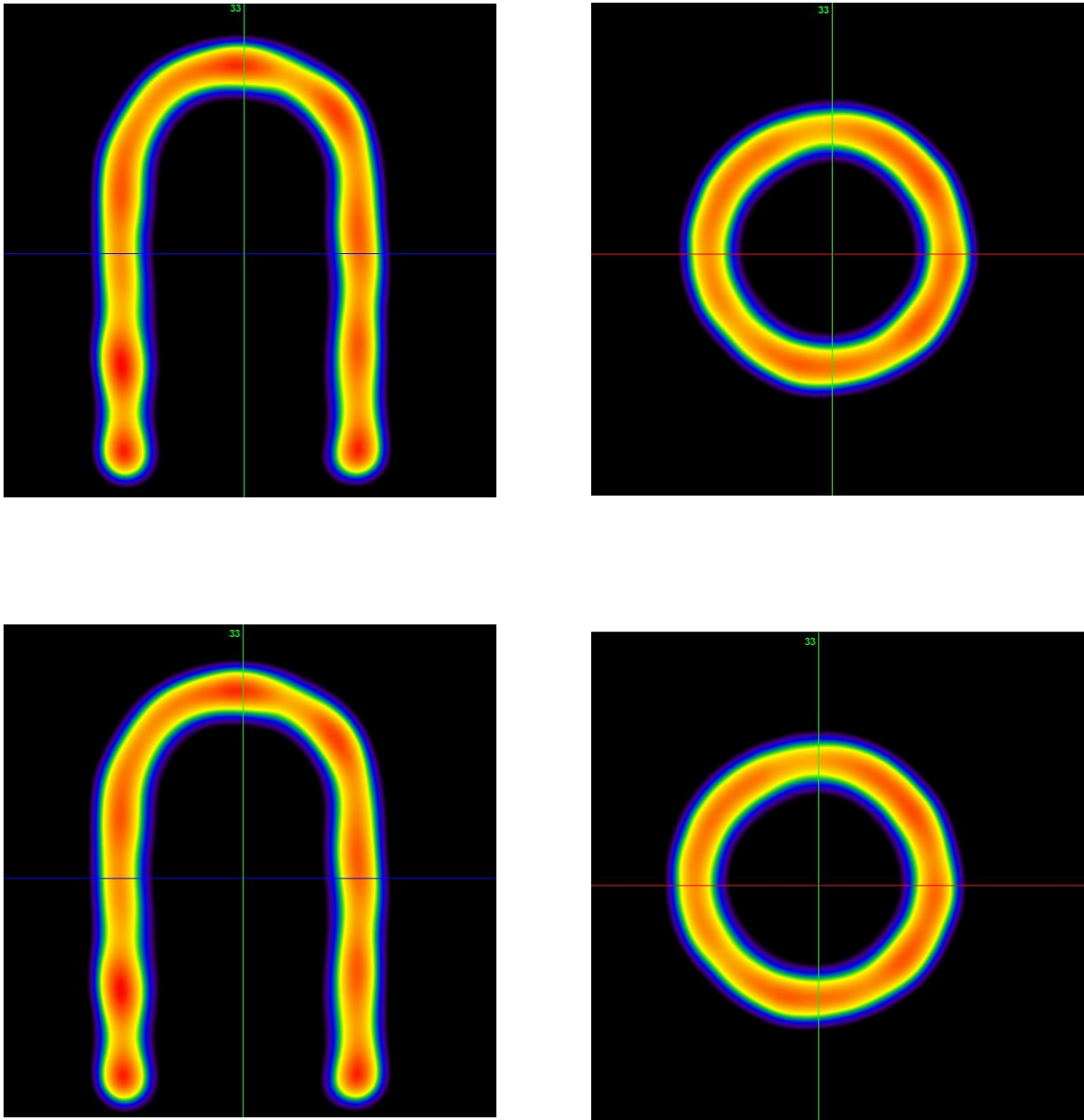
8.4.1. táblázat – Időbeli regularizáció hatása a rekonstruált kép minőségére (L2-norma)

A kapott eredmények azt mutatják, hogy az időbeli regularizáció alkalmas δ paraméter mellett L2-norma szerint javíthat a rekonstruált kép minőségén, 800k időszeletenkénti beütésszámnál $\delta=0.3$, 1600k és 3200k és 6400k-nál $\delta=0.4$ eredményezte L2-norma szerint a legjobb minőségű képet. A falvastagságot pontatlanul kaptam vissza, ez valószínűleg elég erősen függ attól, hogy milyen mozgást feltételezünk a szívnek (például korábban futtattam rekonstrukciót állandó falvastagság mellett is, ekkor mind az időbeli korrekciók nélküli, mind az időbeli regularizáció mellett rekonstruált kép elég pontosan adta vissza az eredetit).

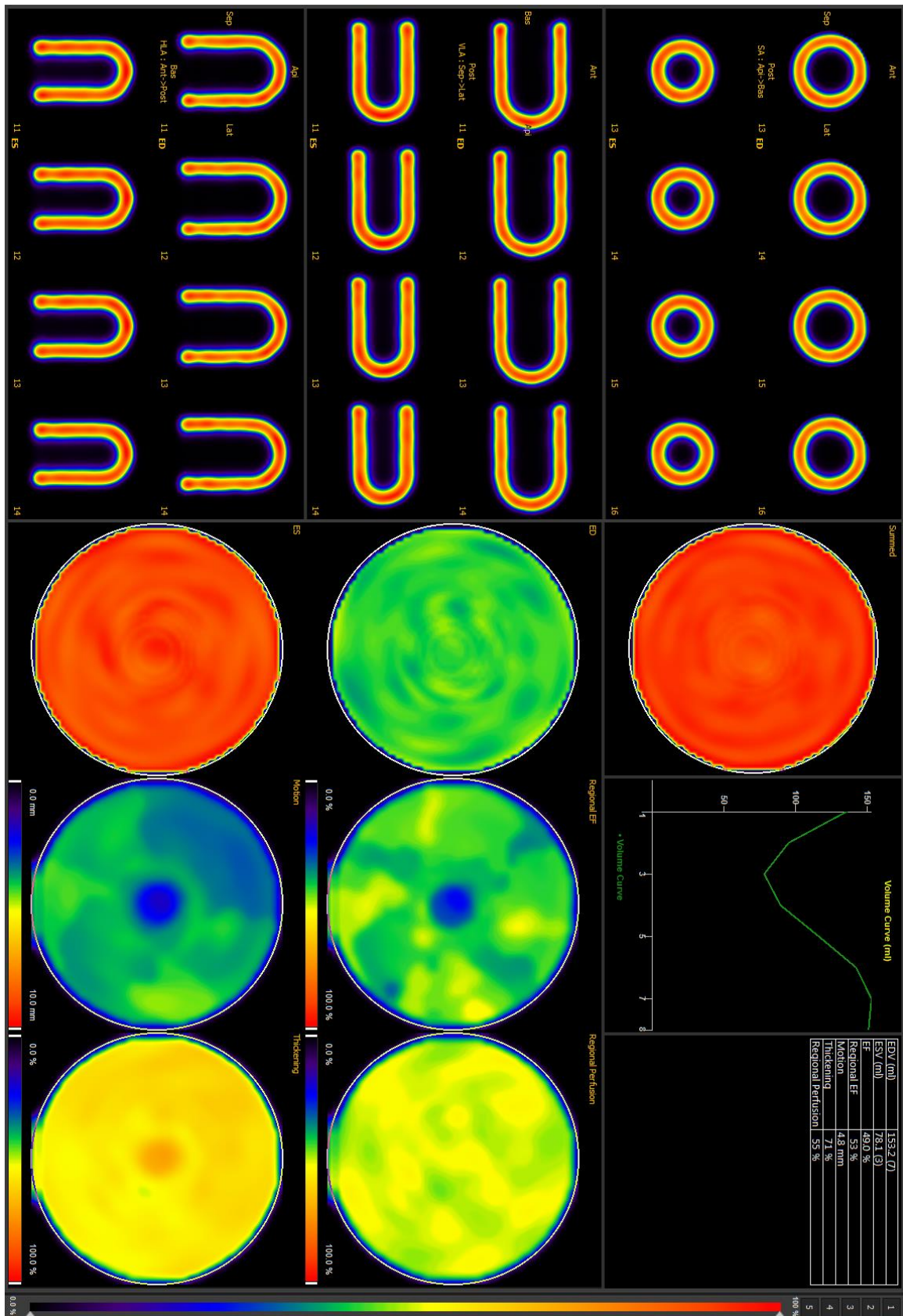
A δ paraméter optimális értékét az összes időszeletre számolt L2-norma alapján kerestem, de meghatároztam időszeletenként is (8.4.3. táblázat). Az eredményekből az látszik, hogy az egyes időszeletekben külön-külön is körülbelül ugyanazok a δ értékek javítanak a legtöbbet L2-norma szerint a képminőségen, mint amikor az időszeleteket együttesen vettem figyelembe.

A rekonstruált képeken a szívfal nem volt teljesen egyenletes, „csomósodás” műterméket tapasztaltam mind az időbeli korrekciók nélkül, mind az idő szerinti regularizáció mellett

rekonstruált képeken. A gyakorlatban szűrőket alkalmazunk rekonstrukció előtt/közben, így egyenletesebb képet kaphatunk. A „legjobb” képet L2-norma alapján választottam ki, de a valóságban nem egyedül ez jellemzi a kép minőségét, kevesebb iterációs lépést alkalmazva valószínűleg kevésbé csomós képet kaptunk volna.



8.4.1. ábra – Szívfantom (EF = 50%) rekonstruált képe időbeli regularizáció mellett (felül) és anélkül (alul), statisztika: 6400k



8.4.2. ábra – Szívfantom (EF = 50%) rekonstruált képe időbeli regularizáció mellett, statisztika: 6400k

statisztika	δ	eredeti	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7
800k	1.	12.375	12	12.8	13.6	13.6	13.2	13.2	12.8	12.8
	2.	13.125	12.4	16	15.2	14.4	14	14	14	13.2
	3.	13.875	12.8	16.4	14.8	14	13.6	13.2	13.2	13.2
	4.	14.25	13.2	16.4	15.2	14.4	14	13.6	13.6	13.2
	5.	13.75	12.4	13.2	13.6	13.2	12.8	12.8	12.4	12.4
	6.	13.25	12.4	12.8	12.8	12.4	12.4	12.4	12.4	12.4
	7.	12.75	12	12	12.4	12	12	12	12	12
	8.	12.25	12.8	12.8	13.2	12.8	12.8	12.8	12.8	12.8
1600k	1.	12.375	11.6	12.4	13.2	13.2	12.8	12.8	12.4	12.4
	2.	13.125	13.2	16.4	15.2	14.8	14.4	14	14	14
	3.	13.875	13.6	16.8	15.6	14.4	14.4	14.4	14	14
	4.	14.25	14	17.2	16.4	15.6	14.8	14.8	14.8	14.8
	5.	13.75	12.4	13.6	14	13.6	13.2	13.2	12.8	12.8
	6.	13.25	12.8	12.8	12.8	12.4	12.4	12.8	12.8	12.8
	7.	12.75	13.2	12.8	13.6	13.6	13.6	13.2	13.2	13.2
	8.	12.25	13.2	13.2	13.2	13.2	12.8	12.8	12.8	12.8
3200k	1.	12.375	11.6	12.4	12.8	12.8	12.4	12.4	12	12
	2.	13.125	12.8	16	15.2	14.8	14	14	13.6	13.6
	3.	13.875	13.6	17.2	15.6	14.8	14.4	14.4	14.4	14.4
	4.	14.25	13.2	16.4	15.2	14.8	14	14	14	14
	5.	13.75	12.4	13.6	13.6	13.2	12.8	12.8	12.8	12.8
	6.	13.25	12.8	12.8	12.8	12.8	12.4	12.4	12.4	12.8
	7.	12.75	12.8	12.8	12.8	13.2	13.2	13.2	13.2	13.2
	8.	12.25	12.8	12.8	12.8	12.8	12.8	12.8	12.8	12.8
6400k	1.	12.375	12.4	12.8	13.2	13.6	13.2	13.2	12.8	12.8
	2.	13.125	12.8	15.6	14.4	14	13.6	13.6	13.6	13.6
	3.	13.875	13.2	16	14.4	13.6	13.6	13.6	13.6	13.6
	4.	14.25	12.8	16	14.8	14.4	14	13.6	13.6	13.6
	5.	13.75	12.4	13.2	13.2	12.8	12.8	12.8	12.8	12.8
	6.	13.25	12.8	12.8	12.8	13.2	12.8	12.8	12.8	12.8
	7.	12.75	12	12.8	12.8	12.8	12.8	12.8	12.8	12.8
	8.	12.25	12.8	12.8	12.8	12.8	12.8	12.8	12.8	12.8

8.4.2. táblázat – Szívfall vastagsága időszeletenként

statisztika	δ	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7
800k	1.	0.0497	0.0784	0.0513	0.0448	0.0434	0.0434	0.0438	0.0442
	2.	0.0336	0.0486	0.0317	0.0278	0.0267	0.0268	0.0271	0.0275
	3.	0.0288	0.0384	0.0217	0.0191	0.0197	0.0205	0.0213	0.0219
	4.	0.0291	0.0419	0.0243	0.0206	0.0203	0.0207	0.0212	0.0217
	5.	0.0421	0.0329	0.0265	0.0264	0.0277	0.0291	0.0302	0.0312
	6.	0.0434	0.0476	0.0292	0.0294	0.031	0.0324	0.0335	0.0344
	7.	0.0485	0.0306	0.0258	0.0298	0.0328	0.0349	0.0364	0.0376
	8.	0.0485	0.0329	0.0264	0.03	0.0329	0.035	0.0365	0.0377
1600k	1.	0.0277	0.0707	0.039	0.0302	0.0275	0.0266	0.0263	0.0262
	2.	0.0208	0.0458	0.0247	0.0192	0.0178	0.0174	0.0174	0.0175
	3.	0.0228	0.0352	0.0175	0.0153	0.0159	0.0166	0.0172	0.0177
	4.	0.0221	0.0417	0.0219	0.0177	0.0169	0.0169	0.0171	0.0173
	5.	0.0254	0.03	0.021	0.0193	0.0194	0.0198	0.0202	0.0206
	6.	0.0292	0.039	0.0222	0.0215	0.0222	0.023	0.0236	0.024
	7.	0.0329	0.0251	0.0195	0.0221	0.0239	0.0252	0.026	0.0267
	8.	0.029	0.0263	0.0188	0.0203	0.0216	0.0226	0.0233	0.0238
3200k	1.	0.0232	0.0697	0.0371	0.0279	0.0249	0.0238	0.0233	0.023
	2.	0.0181	0.047	0.0236	0.0181	0.0166	0.0161	0.016	0.016
	3.	0.0179	0.0365	0.016	0.0138	0.0141	0.0145	0.0148	0.0151
	4.	0.0184	0.0412	0.0198	0.0158	0.0149	0.0148	0.0148	0.015
	5.	0.0217	0.0303	0.0202	0.0178	0.0175	0.0176	0.0178	0.0181
	6.	0.0218	0.0375	0.0197	0.0182	0.0183	0.0186	0.0189	0.0191
	7.	0.0211	0.0229	0.0163	0.0173	0.018	0.0184	0.0187	0.019
	8.	0.0231	0.0238	0.0165	0.0174	0.0183	0.019	0.0194	0.0198
6400k	1.	0.0173	0.0614	0.0298	0.0217	0.0192	0.0182	0.0178	0.0175
	2.	0.0154	0.0418	0.0204	0.016	0.0148	0.0144	0.0143	0.0142
	3.	0.0171	0.031	0.0139	0.0133	0.0138	0.0142	0.0145	0.0148
	4.	0.0176	0.0354	0.0171	0.0144	0.014	0.0141	0.0143	0.0144
	5.	0.0184	0.0274	0.0179	0.0158	0.0155	0.0156	0.0158	0.0159
	6.	0.019	0.0311	0.0174	0.0166	0.0167	0.0169	0.0171	0.0173
	7.	0.0176	0.0181	0.0143	0.0151	0.0156	0.0159	0.0161	0.0162
	8.	0.0169	0.0194	0.0141	0.0146	0.0151	0.0154	0.0156	0.0158

8.4.3. táblázat – L2-norma időszakonként

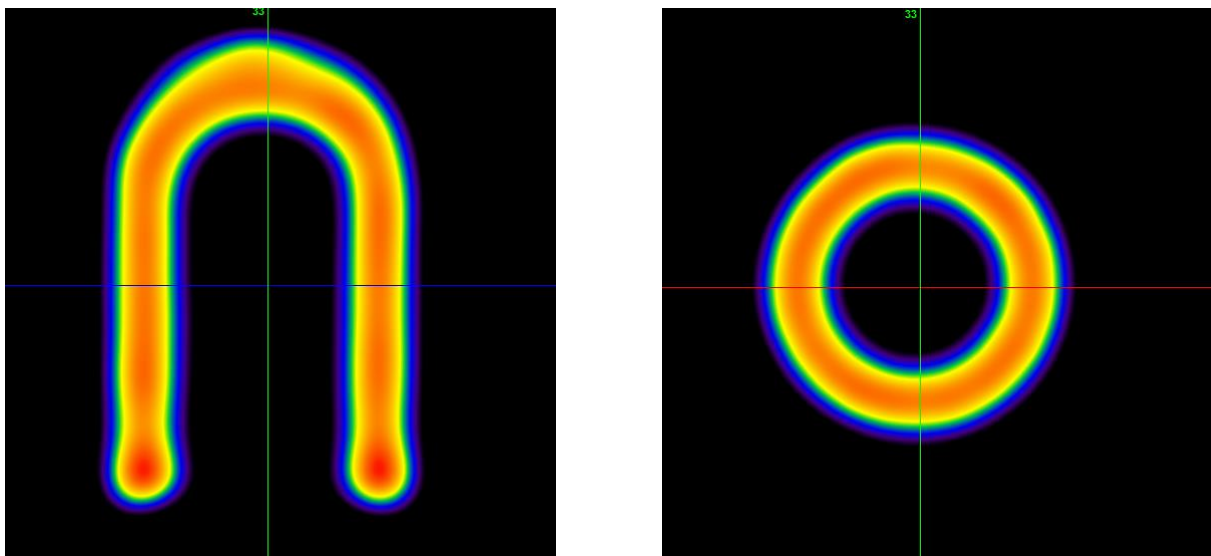
8.5. Fourier-térbeli szűrés hatása a rekonstruált kép minőségére

A rekonstrukciót elvégeztem a 7.3. fejezetben bemutatott módszerrel is, különböző számú Fourier-komponens elhagyása mellett, különböző statisztikájú vetületi képekre.

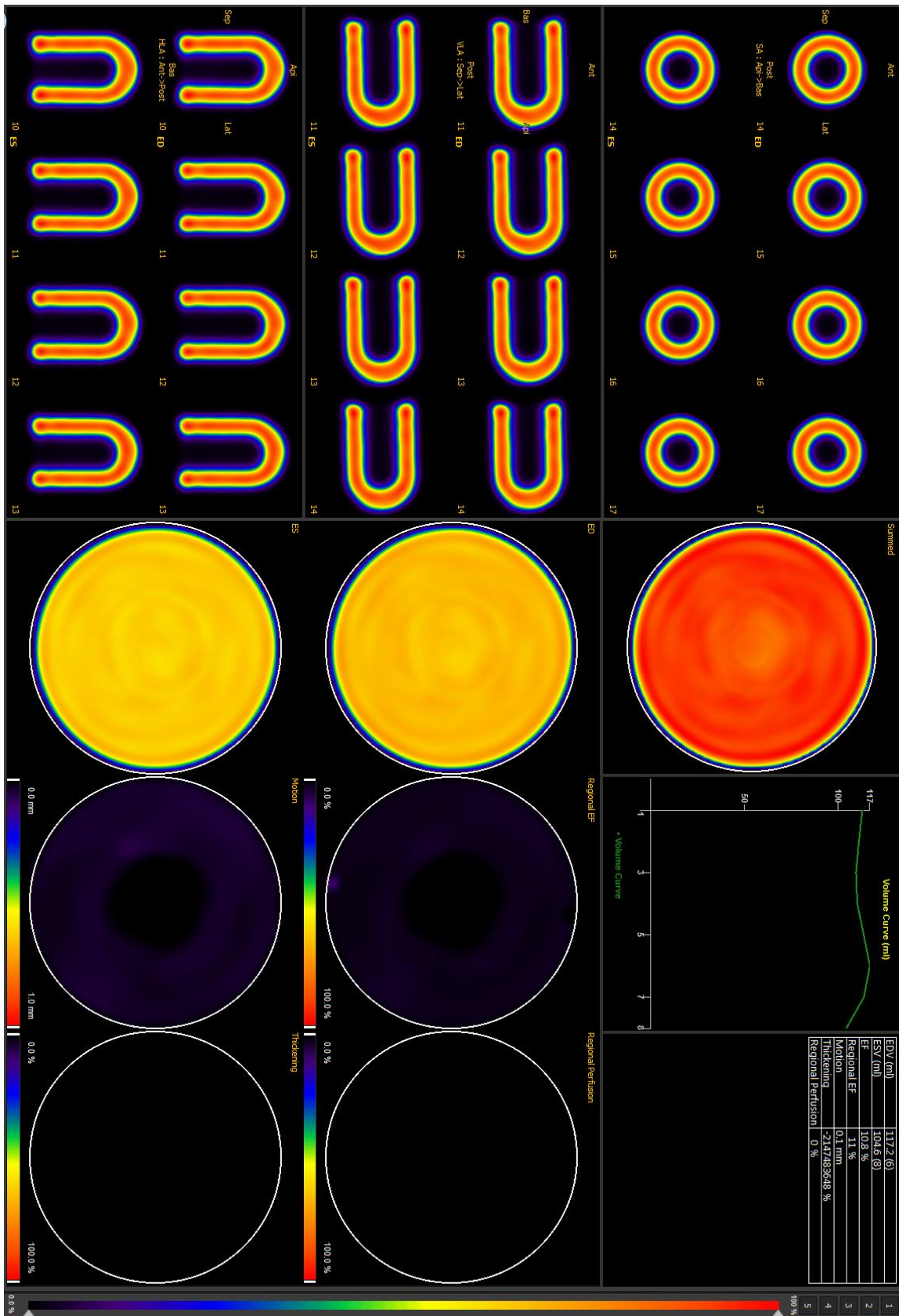
A kapott eredmények azt mutatják, hogy a korrekció L2-norma szerint rontott (a legtöbb esetben akár egy nagyságrendnyit) a rekonstruált kép pontosságán az időbeli korrekciókat nem tartalmazó algoritmushoz képest. A falvastagság megnőtt, ugyan egyenletesebb (a szívfalban tapasztalható „csomósodás” eltűnt), de jóval elkentebb képet kaptam.

statisztika	δ	nincs	1	2	3	4	5	6	7
800k	L2	0.0362	0.2	0.1494	0.1441	0.1439	0.1412	0.3414	0.4201
1600k	L2	0.0244	0.199	0.147	0.1418	0.1415	0.1383	0.3282	0.3967
3200k	L2	0.0196	0.1987	0.1473	0.1422	0.142	0.1387	0.3164	0.3733
6400k	L2	0.0172	0.1983	0.1465	0.1412	0.1409	0.1376	0.3124	0.3734

8.5.1. táblázat – Fourier-térbeli szűrés hatása a rekonstruált kép minőségére (L2-norma)



8.5.1. ábra – Szívfantom rekonstruált képe Fourier-térbeli szűrés mellett, statisztika: 6400k



8.5.1. ábra – Szívphantom rekonstruált képe Fourier-térbeli szűrés mellett, statisztika: 6400k

statisztika	F	eredeti	nincs	1	2	3	4	5	6	7
800k	1.	12.375	12	13.2	14.4	14.8	14.8	14.8	11.2	14.4
	2.	13.125	12.4	13.2	14.8	14.8	14.8	14.8	10	9.6
	3.	13.875	12.8	13.2	14.8	15.2	14.8	15.2	9.6	9.6
	4.	14.25	13.2	13.2	14.8	14.8	14.8	14.8	8.4	9.6
	5.	13.75	12.4	13.2	14.8	14.8	14.8	15.2	10.8	17.2
	6.	13.25	12.4	13.2	14.4	14.8	14.8	14.4	15.2	12
	7.	12.75	12	13.2	14.4	14.8	14.8	14.8	17.2	12.8
	8.	12.25	12.8	13.2	14.4	14.8	14.8	14.8	12.8	13.2
1600k	1.	12.375	11.6	14	15.2	15.6	15.2	15.6	12.4	15.6
	2.	13.125	13.2	14	15.2	15.6	15.2	15.6	10	12.8
	3.	13.875	13.6	14	15.2	15.6	15.2	16	8.8	8.8
	4.	14.25	14	14	15.2	15.6	15.2	15.2	9.2	9.6
	5.	13.75	12.4	14	15.2	15.2	15.2	16	10.4	10
	6.	13.25	12.8	14	15.2	15.2	15.2	15.2	12.8	13.2
	7.	12.75	13.2	14	15.8	15.2	15.2	15.2	14.8	14.8
	8.	12.25	13.2	14	15.2	15.2	15.2	15.6	13.6	14.8
3200k	1.	12.375	11.6	13.6	14.8	14.8	14.8	15.6	12.4	14.8
	2.	13.125	12.8	13.6	14.4	14.8	14.8	14.8	9.2	10.4
	3.	13.875	13.6	13.6	14.4	15.2	14.8	15.2	8.8	9.6
	4.	14.25	13.2	13.6	14.8	14.8	14.8	14.8	8.8	8.8
	5.	13.75	12.4	13.6	14.8	14.8	14.8	15.2	10.8	19.2
	6.	13.25	12.8	13.6	14.8	14.8	14.8	14.8	11.6	15.6
	7.	12.75	12.8	13.6	14.4	14.8	14.8	15.2	12	14.8
	8.	12.25	12.8	13.6	14.8	14.8	14.8	14.8	12.8	14.4
6400k	1.	12.375	12.4	13.6	14.8	15.2	15.2	15.2	14.4	12.8
	2.	13.125	12.8	13.6	14.8	15.2	14.8	14.4	9.6	13.2
	3.	13.875	13.2	13.6	14.8	14.8	14.8	15.6	9.2	8.8
	4.	14.25	12.8	13.6	14.8	15.2	14.8	14.4	9.2	10.4
	5.	13.75	12.4	13.6	14.8	15.2	14.8	15.6	11.2	14.8
	6.	13.25	12.8	13.6	14.8	14.8	14.8	14.8	12	12.8
	7.	12.75	12	13.6	14.8	14.8	14.8	14.8	13.2	12
	8.	12.25	12.8	13.6	14.8	14.8	14.8	14.8	14.8	12

8.5.2. táblázat – Szívfall vastagsága időszakonként

statisztika	F	nincs	1	2	3	4	5	6	7
800k	1.	0.0497	0.1711	0.2404	0.2305	0.2289	0.1938	0.2606	0.2281
	2.	0.0336	0.136	0.0867	0.0868	0.0829	0.0668	0.2085	0.1972
	3.	0.0288	0.1513	0.0832	0.0751	0.0771	0.0986	0.5428	0.733
	4.	0.0291	0.1286	0.0707	0.0668	0.0661	0.0507	0.3532	0.5247
	5.	0.0421	0.051	0.1018	0.1026	0.1062	0.0905	0.2849	0.2264
	6.	0.0434	0.2043	0.2429	0.2399	0.2318	0.2844	0.3865	0.4651
	7.	0.0485	0.1864	0.1985	0.2246	0.2336	0.1931	0.3516	0.4434
	8.	0.0485	0.1571	0.1884	0.199	0.1978	0.2444	0.3013	0.3731
1600k	1.	0.0277	0.1656	0.2341	0.2241	0.2229	0.187	0.2415	0.2107
	2.	0.0208	0.1367	0.0856	0.0857	0.0819	0.0642	0.2011	0.1765
	3.	0.0228	0.1529	0.0827	0.0744	0.0766	0.0984	0.5348	0.7041
	4.	0.0221	0.13	0.07	0.0659	0.0653	0.0489	0.3417	0.4955
	5.	0.0254	0.0474	0.0967	0.0979	0.1013	0.0863	0.2536	0.1832
	6.	0.0292	0.1993	0.2368	0.2336	0.2256	0.2778	0.368	0.4456
	7.	0.0329	0.1813	0.1929	0.2186	0.227	0.1861	0.3436	0.4284
	8.	0.029	0.1518	0.1827	0.1935	0.1915	0.2374	0.2804	0.3595
3200k	1.	0.0232	0.1682	0.237	0.228	0.2269	0.1907	0.2207	0.205
	2.	0.0181	0.134	0.0835	0.0836	0.0802	0.0635	0.1999	0.1664
	3.	0.0179	0.1502	0.0807	0.0731	0.0752	0.0963	0.5169	0.6631
	4.	0.0184	0.1277	0.0682	0.0644	0.0639	0.0482	0.337	0.4602
	5.	0.0217	0.0488	0.0988	0.1	0.1035	0.0877	0.2124	0.1608
	6.	0.0218	0.202	0.24	0.237	0.2287	0.2799	0.3602	0.4199
	7.	0.0211	0.1838	0.1957	0.2204	0.2289	0.188	0.3452	0.4078
	8.	0.0231	0.154	0.1852	0.1957	0.1937	0.2385	0.2602	0.3469
6400k	1.	0.0173	0.1669	0.2356	0.2263	0.2251	0.1874	0.2075	0.2051
	2.	0.0154	0.1342	0.0832	0.0829	0.0792	0.0625	0.212	0.1606
	3.	0.0171	0.1505	0.0807	0.0726	0.0748	0.0963	0.4807	0.6704
	4.	0.0176	0.1278	0.0681	0.064	0.0633	0.0473	0.3558	0.4659
	5.	0.0184	0.0478	0.097	0.0981	0.1015	0.085	0.183	0.1444
	6.	0.019	0.2011	0.239	0.236	0.2278	0.2799	0.3653	0.4237
	7.	0.0176	0.1828	0.1948	0.2199	0.2287	0.1863	0.3529	0.402
	8.	0.0169	0.1529	0.184	0.1948	0.1929	0.239	0.2572	0.3475

8.5.3. táblázat – L2-norma időszakonként

8.6. Rekonstrukció tesztelése más matematikai fantomokon

Az időbeli regularizáció hatását teszteltem egy 70% ejekciós frakcióval rendelkező matematikai fantomon is. L2-norma szerint itt is javult kicsit a képminőség, δ optimális értéke eltért az 50% ejekciós frakciójú fantom esetében kapott értéktől, 0.7-et kaptam.

statisztika	δ	0	0.2	0.7	1.2	1.7	2.2	2.7	3.2
6400k	L2	0.0169	0.0291	0.0153	0.0156	0.016	0.0162	0.0164	0.0166

8.6.1. táblázat – Időbeli regularizáció hatása (L2-norma), EF=70%

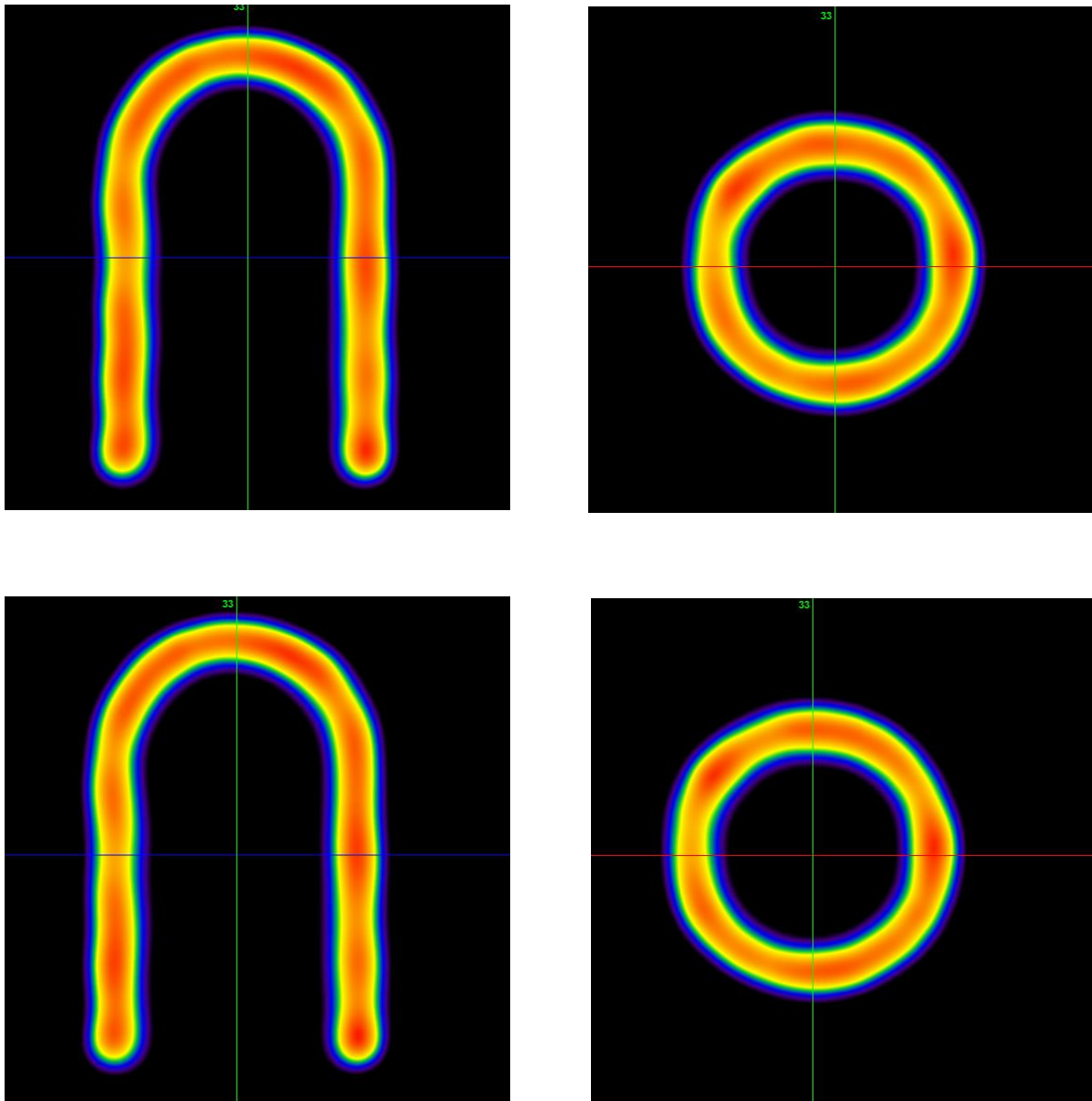
statisztika	δ	eredeti	0	0.2	0.7	1.2	1.7	2.2	2.7	3.2
6400k	1.	12.375	12	14	12.8	12.8	12.4	12.4	12.4	12
	2.	13.125	12.8	15.6	13.2	13.2	13.2	12.8	12.8	12.8
	3.	13.875	12.8	16.4	14	13.2	13.2	13.2	13.2	13.2
	4.	14.25	12.8	14.4	12.8	12.8	12.8	12.8	12.4	12.4
	5.	13.75	12.4	13.2	12.4	12.4	12.4	12.4	12.4	12.4
	6.	13.25	12	12.4	12.8	12.8	12.8	12.8	12.8	12.8
	7.	12.75	12	12.4	12.4	12.4	12.4	12.4	12.4	12.4
	8.	12.25	12.4	12.4	12	12	11.6	11.6	11.6	11.6

8.6.2. táblázat – Időbeli regularizáció hatása a falvastagságra, EF=70%

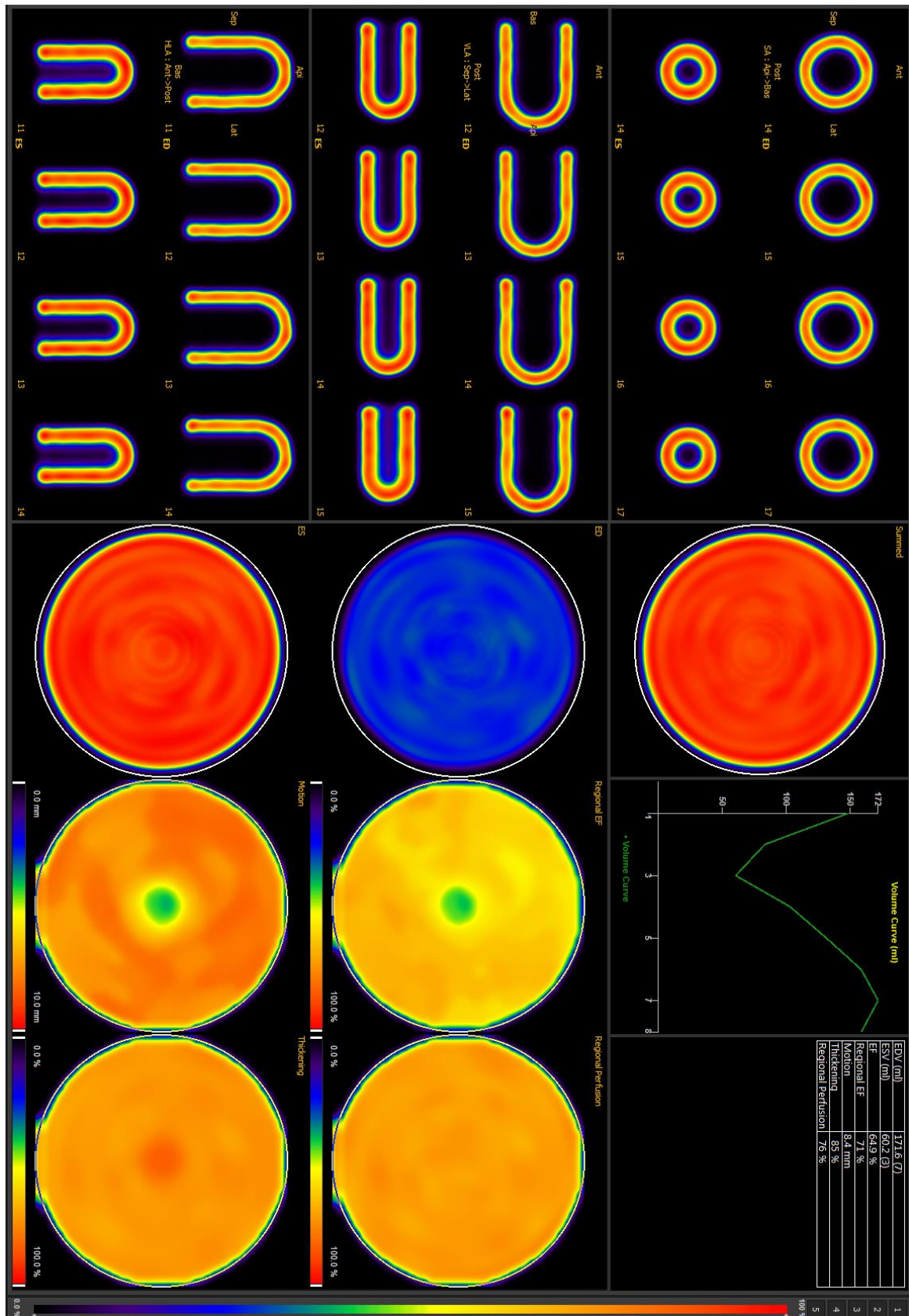
statisztika	δ	0	0.2	0.7	1.2	1.7	2.2	2.7	3.2
6400k	1.	0.0174	0.0406	0.0179	0.0176	0.0175	0.0175	0.0175	0.0175
	2.	0.0145	0.0288	0.0137	0.0138	0.014	0.0142	0.0143	0.0144
	3.	0.0159	0.0347	0.0141	0.0144	0.0149	0.0154	0.0156	0.0159
	4.	0.021	0.0293	0.0179	0.0186	0.0193	0.0198	0.0201	0.0203
	5.	0.0191	0.0151	0.0167	0.0172	0.0175	0.0178	0.018	0.0181
	6.	0.0195	0.0183	0.0177	0.0181	0.0183	0.0185	0.0186	0.0187
	7.	0.0187	0.0157	0.0166	0.0169	0.0171	0.0172	0.0173	0.0173
	8.	0.0183	0.0155	0.0169	0.0171	0.0173	0.0174	0.0175	0.0175

8.6.3. táblázat – Időbeli regularizáció hatása időszeletenként (L2-norma), EF=70%

A 70%-os fantom esetében a falvastagodás és a hossz változása ugyanúgy 120% és 90% volt, mint az 50%-os fantom esetében, abban, hogy eltérő eredményeket kaptam, szerepe lehet a térfogatváltozás, a falvastagodás és a hosszváltozás egymáshoz képesti arányának is. A szívfal vastagságát ebben az esetekben is elég változó pontossággal kaptam vissza, nagyobb eltéréseket is tapasztaltam, mind az időbeli korrekció nélküli, mind az időbeli korrekció mellett rekonstruált képeken.



8.6.1. ábra – Szívfantom (EF = 70%) rekonstruált képe időbeli regularizáció mellett (felül) és anélkül (alul), statisztika: 6400k



8.6.2. ábra – Szívphantom (EF = 70%) rekonstruált képe időbeli regularizáció mellett, statisztika: 6400k

8.7. Eredmények összefoglalása, hibalehetőségek

Az előző pontokban bemutatott eredmények szerint a TV-norma idő dimenzióra való kiterjesztése L2-norma szerint javíthat a rekonstruált kép minőségén. A szívfal vastagságát a használt fantom méreteitől, mozgásától függően elég változatos pontossággal kaptam vissza, vizuálisan nem tapasztaltam lényeges eltéréseket az időbeli regularizáció mellett és anélkül rekonstruált képek között. A Fourier-térbeli szűrés a képet túlzottan elkente, L2-norma szerint sokat rontott a képminőségen, a szívfal elfogadhatatlanul megvastagodott.

A rekonstrukció során szabadon megadható paraméterek optimális értéke nem független egymástól, a projekciós képek statisztikájától, illetve a rekonstruálni kívánt fantom méreteitől, mozgásától. A gyakorlatban különböző, vizsgálatra specifikus szűrőket is használunk a jobb képminőség eléréséhez. Sok szabad paraméterünk van, ezeket összehangolni időigényes, diplomamunkám keretein belül inkább csak egy képet próbáltam kapni arról, hogy a megírt rekonstrukciós algoritmus azt adja-e, amire előismereteink alapján számítunk, illetve azt próbáltam feltérképezni, hogy az alkalmazott időbeli korrekciók javíthatnak-e a képminőségen.

Az iterációs lépések számát és a prior információ hatását megadó β paraméter értékét a hosszú futási idő miatt 1 időszeletes fantomra optimalizáltam, a 8 időszeletre kapott L2-norma és falvastagság értékek alapján valószínűleg lehetne ezeknek jobb értéket is találni. Az optimális érték keresésekor a falvastagságot nem vettem figyelembe (ezt az 1 időszeletes fantom esetében elég pontosan kaptam vissza az L2-norma szerint optimális iterációs lépés számnál és β értéknél), több időszelet esetében azonban elég nagy eltéréseket is tapasztaltam, nem érdemes egyedül L2-norma alapján optimalizálni. β és δ paraméterek optimális értéke is összefügghet egymással, érdemes lenne ezt is tesztelni, illetve TV-norma helyett más időbeli regularizációval is lehetne kísérletezni. A matematikai fantom mozgása szintén csak elnagyoltan követi a szív mozgását, nem biztos, hogy a fantom esetében jól működő paraméterek valós szívvizsgálatok esetében is alkalmazhatóak. A 4.2. fejezetben bemutatott NCAT szívfantom jóval realiztikusabb, ezen is tesztelni kellene az algoritmust.

Összességében elképzelhető, hogy a TV-norma idő dimenzióra való kiterjesztésével javítani lehet a rekonstruált képen, ennek megállapításához a paraméterek közti összefüggések további vizsgálata és optimalizálás szükséges. A Fourier-térbeli szűrés nekem nem adott jó eredményeket, de itt is meg lehetne nézni, hogy mi történik precízebb paraméterbeállítás és kiértékelés mellett.

9. Összefoglalás

Diplomamunkámban kapuzott szívizom SPECT vizsgálatok kapcsán vizsgáltam, hogy mennyire változik a kapott kép minősége, amennyiben a rekonstrukció során a különböző időszeltek voxelértékei közti összefüggéseket is figyelembe vesszük. Ehhez olyan C++ programkódot írtam, amely kapuzott szívizomvizsgálatokat rekonstruál MAP-EM/MOS-EM statisztikai iteratív algoritmussal, időbeli korrekciók alkalmazásával, illetve anélkül. Az alábbi lépéseket végezhetjük:

1. Szív mozgását közelítő matematikai fantom generálása különböző paraméterekkel
2. Projekciós képek készítése a generált fantomról, vagy egyéb, DICOM fájlból betöltött aktivitáseloszlásról
3. Rekonstrukció a projekciós képek alapján térbeli és időbeli korrekciókkal
4. Rekonstruált kép minőségének kiértékelése

Programkódomban az egyes lépések pontosításra szorulnak, céloom inkább a folyamat egészének megértése és megvalósítása volt, mint az egyes részletek optimalizálása. Olyan ismereteket próbáltam elsajátítani, amelyek kiindulópontot jelenthetnek, amennyiben a jövőben orvosi képalkotással szeretnék foglalkozni: a használt rekonstrukciós algoritmusok matematikai alapjait szerettem volna megérteni; megtanultam, hogy hogyan lehet SPECT rekonstrukciót megvalósító programkódot összeállítani; a GPU programozás és DICOM fájlkezelés szintén kikerülhetetlen, amennyiben klinikai gyakorlatban is alkalmazható rekonstrukciós algoritmust szeretnénk írni.

A programkóddal két időbeli korrekció hatását vizsgáltam: a korábbi tapasztalatok alapján térbeli regularizációként jól működő TV-normát terjesztettem ki idő dimenzióra is, illetve az iterációs lépések között Fourier-térben való szűrést alkalmaztam. Az idő dimenzióbeli TV-norma alkalmazásával L2-norma szerint jobb, bár „csomós”, Fourier-térben való szűréssel ugyan egyenletesebb, de L2-norma szerint jóval pontatlanabb, illetve jóval elkentebb (vastagabb szívfal) képet kaptam. A rekonstruált kép minőségét sok szabadon választható értékű paraméter határozza meg, ezek nem függetlenek egymástól, illetve a projekciós képek statisztikájától, a leképezett fantom/szerv méreteitől, mozgásától. A rekonstruált kép minőségét és a korrekciók hatását érdemes lenne a továbbiakban más paraméterbeállítások mellett is vizsgálni, más rekonstrukciós algoritmusokkal összehasonlítani.

10. Akiknek köszönöm

Gábornak, aki annyit segített, hogy annyi süti nincs, amennyivel viszonzni tudom,

Béla bácsinak, a témám végtelen vékony síkokkal harcoló szülőatyjának,

Családomnak, akik mellettem álltak, bármilyen butaságot csináltam,

Beának és Tiqulónak, akik ebből a felsorolásból kihagyhatatlanok,

Dávidnak a képköltés okosításokért, Andrisnak a magfizikáért, Ákosnak és az IIT tanszéknek a segítségükért és mindenkinek, aki hiányolja, hogy kimaradt.

11. Hivatkozások

- [1] InterView XP Users Manual, Mediso Ltd., <http://www.mediso.hu>
- [2] DICOM (Digital Imaging and Communications in Medicine), <http://dicom.nema.org>
- [3] OpenCV (Open Source Computer Vision), Version 2.4.9, <http://opencv.org>
- [4] DCMTK (DICOM Toolkit), Version 3.6.0, <http://dicom.offis.de>
- [5] OpenCL (Open Computing Language), <https://www.khronos.org/opencv/>
- [6] GATE User's Guide, <http://www.opengatecollaboration.org>
- [7] Joseph D. Bronzino: The Biomedical Engineering Handbook
28.3. Cardiac Pump Function
- [8] W. P. Segars, G. Sturgeon, S. Mendoca, Jason Grimes, B. M. W. Tsui: 4D XCAT phantom for multimodality imaging research, Med. Phys., 2010 September
- [9] Habib Zaidi: Quantitative Analysis in Nuclear Medicine Imaging
5. E. C. Frey, B. M. W. Tsui: Collimator-Detector Response Compensation in SPECT
- [10] Orvosi képzés tantárgy, Budapesti Műszaki és Gazdaságtudományi Egyetem (Dr. Légrády Dávid)
- [11] Orvosi fizika tankönyv, <http://oftankonyv.reak.bme.hu>
2.3.3 A szűrt visszavetítés
- [12] Miles N. Wernick, Ph.D., John N. Aarsvold, Ph.D.: Emission Tomography – The Fundamentals of PET and SPECT
V.F. Subset-Based Reconstruction Algorithms
- [13] Peter J. Green: Bayesian Reconstructions From Emission Computed Tomography Data Using a Modified EM Algorithm, Medical Imaging, March 1990

- [14] Stuart Geman & Donald Geman: Stochastic relaxation, Gibbs Distributions, and the Bayesian Restoration of Images, *Pattern Analysis and Machine Intelligence*, November 1984
- [15] B. A. Mair, Jeff Zahnzen: A generalization of Green's one-step-late algorithm for penalized ML reconstruction of PET images, *IEEE Nuclear Science Symposium Conference Record*, 2006
- [16] Kenneth Lange: Convergence of EM Image Reconstruction Algorithms with Gibbs Smoothing, *Medical Imaging*, December 1990
- [17] V. Y. Panin, G. L. Zeng, G. T. Gullberg: Total Variation Regulated EM Algorithm, *Nuclear Science Symposium*, 1998
- [18] David S. Lalush, Benjamin M. W. Tsui: A generalized Gibbs prior for maximum a posteriori reconstruction in SPECT, *Physics in Medicine and Biology*, 1993
- [19] Mikael Persson, Dianna Bone, Hakan Elmqvist: Three-dimensional total variation norm for SPECT reconstruction, *Nuclear Instruments and Methods in Physics*, September 2001
- [20] David S. Lalush, Lin Cui, Benjamin M. W. Tsui: A Priori Motion Models for Four-Dimensional Reconstruction in Gated Cardiac SPECT, *Nuclear Science Symposium*, 1996. *Conference Record*
- [21] Christian Vanhove, Philippe R. Franken, Michel Defrise, Frank Deconinck, Axel Bossuyt: Reconstruction of Gated Myocardial Perfusion SPET Incorporating Temporal Information During Iterative Reconstruction, *European Journal of Nuclear Medicine and Molecular Imaging*, April 2002
- [22] Manoj V. Narayanan, Michael A. King, Miles N. Wernick, Charles L. Byrne, Edward J. Soares, P. Hendrik Pretorius: Improved Image Quality and Computation Reduction in 4-D Reconstruction of Cardiac-Gated SPECT Images, *Medical Imaging*, May 2000
- [23] Xiaofeng Niu, Yongyi Yang: Tomographic Reconstruction of Gated Data Acquisition Using DFT Basis Functions, *Image Processing*, 19. July 2010
- [24] <http://cs229.stanford.edu/notes/cs229-notes8.pdf>

- [25] Kenneth Lange, Richard Carson: EM Reconstruction Algorithms for Emission and Transmission Tomography, Journal of Computer Assisted Tomography, 1984
- [26] <http://www.biodex.com/nuclear-medicine/products/phantoms/cardiac-insert>
- [27] Orvosi fizika tankönyv, <http://oftankonyv.reak.bme.hu>

4.4.8 Transzformációk

12. Ábrák jegyzéke

[K1] http://elsosegely.blog.hu/2012/07/05/a_sziv_felepitese_es_mukodese

[K2] https://en.wikipedia.org/wiki/Wiggers_diagram

[K3] Orvosi fizika tankönyv, <http://oftankonyv.reak.bme.hu>

3.3.2 Radioaktív gamma-sugárzó anyag (2D) képszerű leképezése Anger-elv alapján

1. ábra

[K4] Ákos Szlávecz, Gábor Hesz, Tamás Bükki, Béla Kári, Balázs Benyó: GPU-based acceleration of the MLEM algorithm for SPECT parallel imaging with attenuation correction and compensation for detector response, 18th IFAC World Congress, Milano, 2011

Fig. 2. The gamma camera produce a distance dependent point spread function resulting in a Gaussian blurring