

Positroniemissiotomografian Monte Carlo
-simulaatio ja kuvien OSEM-rekonstruktio
OMEGA-ohjelmistolla

Pro Gradu
Turun yliopisto
Fysiikka
2026
LuK Aleksi Varho
Tarkastajat:
Dos. Kimmo Luoma
Dos. Jarmo Teuho

Turun yliopiston laatujaarjestelmän mukaisesti tämän julkaisun alkuperäisyys on tarkastettu Turnitin OriginalityCheck-järjestelmällä

Varho, Aleks Positroniemissiotomografian Monte Carlo -simulaatio ja kuvien
OSEM-rekonstruktio OMEGA-ohjelmistolla

Pro Gradu, 49 s.
Fysiikka
Kesäkuu 2026

OMEGA on Wettenhovin *et al.* positroniemissiotomografiaa (PET) varten kehittämä kuvarekonstruktio-ohjelmisto, jolla voidaan muodostaa PET-kuvia nopeasti MATLABin tai GNU Octaven käyttöliittymän kautta. Se sisältää laajan valikoiman iteratiivisia rekonstruktioalgoritmeja ja tukee myös uusien algoritmien kehittämistä. OMEGAan on mahdollista tuoda myös GATE-ohjelmistossa Monte Carlo -simuloitua PET-dataa. Monte Carlo -simulaatio on tärkeä työkalu uusien PET-kuvantamismenetelmien ja -laitteistojen kehittämisessä, ja GATEssa voidaan mallintaa geometrialtaan monimutkaisiakin kameroita ja fantomeita.

Tässä opinnäytetyössä toteutettiin GATE-simulaatioita GE Discovery-690 -laitteen kameralla, jolla kuvannettiin sylinteri-, Derenzo- ja NEMA-fantomia sekä Hoffman-aivofantomia, ja rekonstruoidtiin kuvat OMEGAssa MATLABin kautta. Jokaista fantomia kohden kerättiin noin 2 000 000 gammafotoniparia eli koinssidenssia. Simuloidulle datalle tehtiin OMEGAssa satunnais-, sironta- ja normalisaatiokorjaukset, ja valmiit kuvat muodostettiin OSEM-algoritmilla. Kullekin fantomille muodostettiin neljä kuvaa eri kuvako'oilla ja näkökentillä (engl. *field of view*, FOV) ja vertailtiin näiden vaikutusta kuvanlaatuun. Sylinterin kuvista tarkasteltiin rekonstruoidun aktiivisuusjakauman tasaisuutta, Derenzon kuvista saavutettavaa resoluutiota ja NEMAn kuvista kontrastia; Hoffmanilla selvitettiin, miten hyvin OMEGA muodostaa monimutkaista ja realistista elimen aktiivisuusjakamaa esittävän kuvan. GATE-dataan tallennetuilla positroniemissiopisteillä piirrettiin myös lähdekuvat, joihin rekonstruoituja kuvia verrattiin.

Kaikissa kuvissa rekonstruoitu aktiivisuusjakauma noudattaa melko hyvin lähdekuvia, mutta valmiit kuvat ovat todella kohinaisia koinssidenssien liian pienen lukumäärän vuoksi. Sylinterin rekonstruoidut aktiivisuusjakaumat ovat etenkin suuremmalla kuvakoolla ja FOV:llä selvästi epätasaisempia kuin simuloitu jakauma. Derenzon kuvissa voidaan tunnistaa yksittäiset lähdepalloryhmät etenkin pienemmällä FOV:llä, mutta vain isommat pallot näkyvät erillisinä ja vain juuri ja juuri fantomin kattavalla FOV:llä. NEMAn kuvissa näkyy aktiivisin lähdepallo erityisesti pienemmällä kuvakoolla, kun taas muut eivät erotu taustasäteilystä. Hoffmanin kuvista on mahdollista tunnistaa harmaan aineen alueet suurin piirtein. OMEGAa oli helppo käyttää ja se muodosti kuvat nopeasti, joten sitä kannattaa käyttää jatkossakin GATEssa simuloitun datan käsittelyyn, mutta paremman kuvanlaadun takaamiseksi tarvittaisiin kymmeniä kertoja enemmän koinssidensseja.

Asiasanat: positroniemissiotomografia, Monte Carlo -simulaatio, GATE, kuvarekonstruktio, OMEGA, OSEM, Discovery-690, Derenzo, NEMA, Hoffman

Sisällys

Johdanto	2
1 Teoreettinen tausta	3
1.1 Positroniemissiotomografian fysiikkaa	3
1.1.1 Positroniemissio ja elektroni–positroni-annihilaatio	3
1.1.2 Gammasäteilyn ja aineen väliset vuorovaikutukset	5
1.1.3 Kvantamislaitteen rakenne ja toiminta	8
1.2 Monte Carlo -simulaatio	10
1.3 Lääketieteellinen kuvarekonstruktio	12
1.3.1 Kohteen ja havaintojen matemaattiset esitykset	12
1.3.2 Iteratiiviset rekonstruktioalgoritmit	14
1.3.3 Havaintodatan korjaukset	16
2 Ohjelmistot	19
2.1 GATE	19
2.2 OMEGA	22
3 Menetelmät	25
3.1 Simulaatiot	26
3.1.1 Kamera ja digitoija	26
3.1.2 Fantomit	28
3.2 Kuvarekonstruktiot	31
4 Tulokset	32
4.1 Simuloidut havainnot	32
4.2 Rekonstruoidut kuvat	36
5 Pohdinta	39

6 Yhteenveto ja johtopäätökset

44

Johdanto

Positroniemissiotomografia eli PET-kuvantaminen on isotooppilääketieteen toiminnallinen kuvantamismenetelmä, jolla tutkitaan elinten toimintaa ja aineenvaihduntaa. Sen tärkeimpiä sovelluskohteita ovat syövän diagnosointi, sen levinneisyyden tutkiminen ja sädehoidon suunnittelu, jotka kattavat 60–70 % kliinisestä PET-kuvantamisesta [1]; sillä on tutkittu myös esimerkiksi sydämen, aivojen ja umpirauhasten toimintaa sekä tulehduksia [2]. Menetelmän käyttö on lisääntynyt huomattavasti viime vuosikymmeninä: vuosina 2003–2024 Suomessa tehtyjen PET-tutkimusten lukumäärä on kasvanut 1 930:stä 17 376:een [3]. Tämä on luonut yhä suurempaa tarvetta uusien kuvantamislaitteiden ja -tekniikoiden, simulaatioiden sekä kuvarekonstruktioalgoritmien suunnittelulle ja kehittämiselle [4, 5].

PET-kuva esittää positronilähteen jakaumaa elimistössä, ja se muodostetaan annihilatiopisteistä emittoituvien gamma (γ)-fotoniparien avulla. Perinteisesti PET-kuvia on rekonstruoitu analyyttisillä menetelmillä, kuten suodatetulla takaisinprojektiolla [6], joissa aktiivisuusjakauma lasketaan havaittujen fotoniparien lukumäärää vastaavien viivaintegraalien Fourier-muunnosten avulla [7, 8]. Analyyttiset menetelmät ovat nopeita ja helppoja toteuttaa, mutta niillä ei voida mallintaa kaikkia PET-kuvantamiselle luontaisia virhelähteitä, kuten emittoidun positronin kantamaa, fotonien sirontaa kohteessa tai kamerassa taikka ilmaisinkiteiden vaihtelevaa herkkyyttä [9].

Nykyään kliinisessä PET-kuvantamisessa käytetään lähinnä iteratiivisia rekonstruktio menetelmiä [10]. Niissä muodostetaan kaikki havaintoihin vaikuttavat fyysiset ilmiöt ja virhelähteet huomioiva mallikuvaus lähteestä kameralle ja tehdään aktiivisuusjakaumalle alkuarvaus, josta kehitetään valmis kuva askelittain [11]. Näin rekonstruoitujen kuvien laatu on parempi kuin analyyttisillä menetelmillä, mutta niiden muodostaminen on monimutkaisempaa [9]. Suurimmassa osassa kliinisessä kuvantamisessa käytetyistä algoritmeista kuva rekonstruoidaan suurimman uskotta-

vuuden perusteella [5]; näihin lukeutuvat erilaiset odotusarvonmaksimointialgoritmit, kuten MLEM [12] ja OSEM [13]. Toisilla algoritmeilla, kuten OSL:llä [14] ja BSREM:llä [15], rekonstruktio perustuu suurimman jälkitodennäköisyyden laskemiseen [5].

PET-kuvantamiselle on olemassa useita rekonstruktio-ohjelmistoja: esimerkiksi NiftyRec [16], STIR [17] ja CASToR [18]. Wettehovichin *et al.* kehittämä OMEGA (*open-source MATLAB emission tomography software*) [10] hyödyntää MATLABin tai GNU Octaven käyttöliittymää ja OpenCL-tukea. Se on suunniteltu rekonstruoidaan niin staattisia kuin dynaamisia PET-kuvia helposti ja nopeasti, sisältää enemmän rekonstruktioalgoritmeja kuin sitä edeltävät ohjelmistot sekä tarjoaa alustan uusien rekonstruktioalgoritmien suunnittelulle ja kehittämiselle [10].

Monte Carlo -simulaatio on yksi tärkeimmistä välineistä uusien kuvantamislaitteiden suunnittelussa sekä kuvantamisen tehokkuuden ja laadun parantamisessa [4]. OMEGAan on mahdollista tuoda GATE-ohjelmistossa (*Geant4 Application for Tomographic Emission*) [19] simuloitua dataa ja rekonstruoida kuvia sen avulla. GATE soveltaa ydin- ja hiukkasfysiikan simulaatioita varten kehitetyn Geant4-kirjaston [20] työkaluja lääketieteellisen kuvantamisen ja sädehoidon mallintamiseen. Sillä voidaan simuloida monimutkaisia PET-kameran ja kuvannuskohteen geometrioita sekä fyysisiä vuorovaikutuksia ilman sen pohjana toimivan C++-kielen tuntemusta [19].

Tässä opinnäytetyössä suunniteltiin ja ajettiin Monte Carlo -simulaatioita GATEssa ja tuotiin simuloitu data OMEGAan rekonstruoidavaksi. GATEssa mallinnettiin GE Discovery-690 -hybridikuvantamislaitteen [21] PET-kameraa ja neljää erilaista fantomia: sylinterifantomia, Derenzo- [22], NEMA- [23] ja Hoffman-fantomia [24]. Simulaatiodatalle tehtiin satunnais-, sironta- ja normalisaatiokorjaukset OMEGAssa, ja lopulliset kuvat rekonstruoidtiin OSEM-algoritmeilla.

1 Teoreettinen tausta

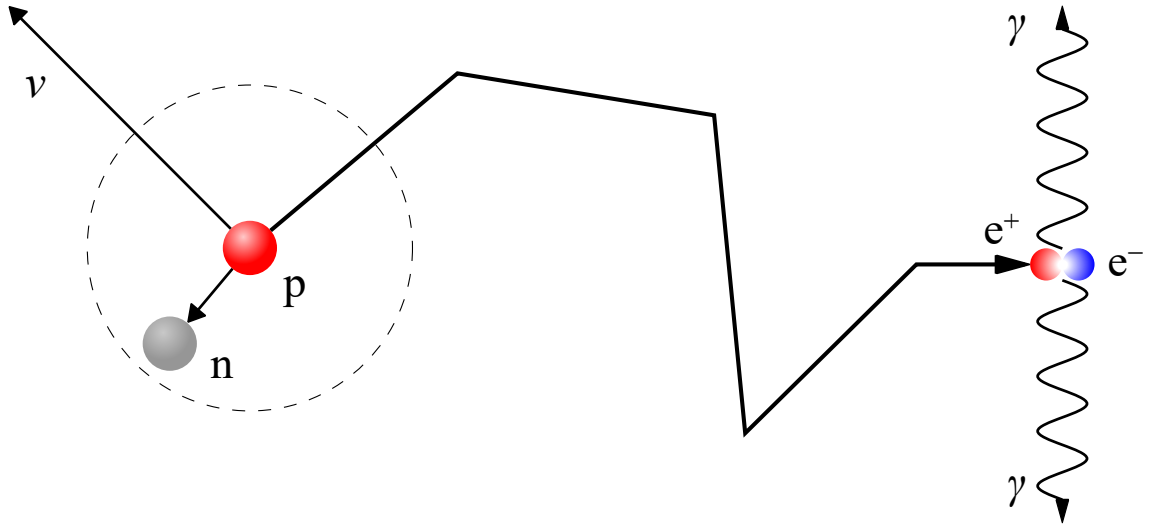
Tämän luvun ensimmäisessä alaluvussa esitellään PET-kuvantamisen kannalta olennaista fysiikkaa: positronien ja γ -fotonien käyttäytymistä sekä kuvantamislaitteen toimintaa. Toisessa alaluvussa kuvaillaan lyhyesti Monte Carlo -simulaation maattista perustaa. Viimeisessä alaluvussa selostetaan, miten kohde ja havainnot esitetään kuvarekonstruktiossa sekä miten iteratiivinen rekonstruktio toteutetaan.

1.1 Positroniemissiotomografian fysiikkaa

PET-kuvantamisessa hyödynnetään merkkiaineena käytettävän isotoopin positiivisia beeta (β^+) -hajoamisia, positroni–elektroni-annihilaatioissa syntyvää γ -säteilyä ja γ -fotonien vuorovaikutuksia PET-kameran ilmaisimien kanssa. PET-merkkiaineet sisältävät protonirikkaita, lyhytikäisiä nuklideja, jotka emittoivat hajotessaan positroneja. Positroni on elektronin antihiukkanen: sillä on elektronin massa mutta positiivinen sähkövaraus. Edettyään muutamia millimetrejä kuvannettavassa kohteessa ja menetettyään kaiken liike-energiansa se annihiloituu elektronin kanssa. Annihilaatiopisteestä emittoituu kaksi vastakkaisiin suuntiin etenevää fotonia, jotka PET-kamera on suunniteltu havaitsemaan lähes yhtäaikaaisesti. [25, 26]

1.1.1 Positroniemissio ja elektroni–positroni-annihilaatio

PET-kuvantamisessa käytettävillä radioisotoopeilla on massalukuunsa nähden liian vähän neutroneja, ja ne ovat protonien välisten sähköisten poistovoimien takia epävakaita. Protonirikas ydin vakautuu β^+ -hajoamisessa, jossa heikko vuorovaikutus muuttaa yhden sen protoneista neutroniksi, positroniksi ja elektronin neutriinoksi. Emoytimen menettämä energia jakautuu satunnaisesti positronin ja neutriinon kesken, joten positroniemissiolla on jatkuva energiaspektri, jonka yläraja vastaa hajonneen nuklidin ja hajoamistuotteiden massaeroa ja jonka keskiarvo on noin kolmannes sen maksimiarvosta. [25]



Kuva 1. Positronin (e^+) elinkaari. Protoni (p) hajoaa neutroniksi (n), positroniksi ja elektronin neutriinoksi (ν). Positroni poistuu emoatomista, törmäilee ympäristön atomeihin ja annihiloituu elektronin (e^-) kanssa, jolloin syntyy kaksi γ -fotonia.

Emittoidun positronin koko elinkaari on esitetty kuvassa 1. Se ei annihiloitu emoatominsa sisällä vaan luovuttaa ensin liike-energiaa ympäristöönsä [27]. Positroni menettää energiansa pääsääntöisesti kimmottomissa törmäyksissä atomien elektroneihin, mikä aiheuttaa atomien virittymisiä ja ionisaatioita [25]. Se voi myös sivuta ytimiä ja emittoida jarrutussäteilyä taikka sirota kimmoisasti elektroneista tai ytimistä luovuttamatta energiaa [28]. Jokaisessa vuorovaikutuksessa positronin energia ja kulkusuunta muuttuvat satunnaisesti, joten sen kulkeman matkan kokonaispituus ei yleensä vastaa sen etäisyyttä emoytimeistä annihilaatiohetkellä eikä matkan keskimääräistä pituutta ole mielekäästä laskea [29]. Kolmelle käytetyimmälle isotoopille – ^{11}C , ^{13}N ja ^{18}F [30] – kokeellisesti määritetyt keskimääräiset etäisyydet vedessä ovat 1,1 mm, 1,5 mm ja 0,6 mm; vastaavat maksimietäisyydet ovat 4,1 mm, 5,1 mm ja 2,4 mm [28].

Positronin ja elektronin oletetaan olevan levossa annihilaation tapahtuessa [26]. Tällöin kummankin annihilaatiopisteestä emittoituvan γ -fotonin energia on identtinen elektronin lepoenergian $m_e c^2 = 511 \text{ keV}$ kanssa, missä m_e on elektronin lepo-massa ja c valonnopeus, ja ne etenevät vastakkaisiin suuntiin. Käytännössä positroni

ja elektroni eivät ole täysin levossa PET-kameraan nähden, joten fotonien lentoradat eivät ole täysin vastakkaissuuntaiset: niiden keskimääräinen kulmapoikkeama on noin $\pm 0,28^\circ$ [31].

1.1.2 Gammasäteilyn ja aineen väliset vuorovaikutukset

Gammasäteily vaimentuu edetessään väliaineessa, kun osa γ -fotoneista menettää energiaansa atomeille ja joko muuttaa etenemissuuntaansa tai katoaa täysin. Jos isotrooppisesti säteilevä pistelähde sijaitsee tasaisesti vaimentavassa kappaleessa syvyydessä x , niin ulkoiselle ilmaisimelle osuvien fotonien tiheys ajan suhteen on

$$C = C_0 e^{-\mu x} = C_0 e^{-\rho_N \sigma x}, \quad (1)$$

missä C_0 on havaintotiheys ilman vaimennusta, μ lineaarinen vaimenemiskerroin, ρ_N kappaleen atomitiheys ja σ kokonaisvaikutusala, joka kuvaa fotonin todennäköisyyttä vuorovaikuttaa yksittäisen atomin kanssa. Yhden annihilaatiofotonin edetessä matkan x kappaleessa sen pari etenee matkan $L - x$, missä L on kappaleen leveys fotonien kulkusuunnassa; ilmaisiparin havaitsemien fotoniparien tiheys on siten

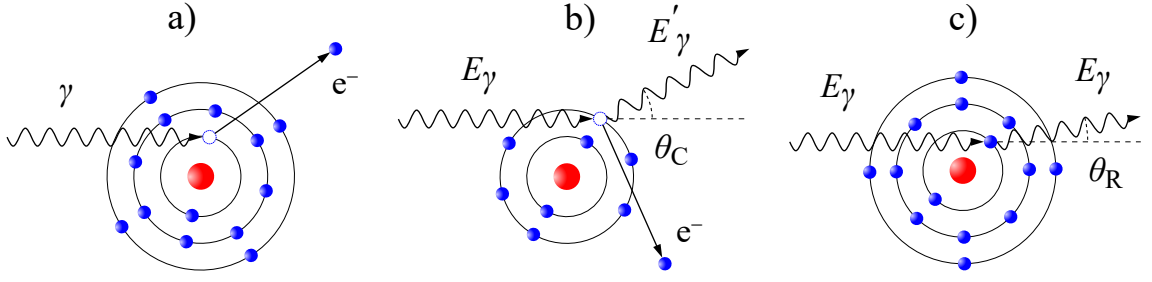
$$C = C_0 e^{-\mu x} e^{-\mu(L-x)} = C_0 e^{-\mu L}, \quad (2)$$

eli se ei riipu lähteen tarkasta sijainnista kappaleessa. Annihilaatiofotonit voivat vuorovaikuttaa ympäristönsä kanssa kolmella tavalla, jotka on havainnollistettu kuvassa 2: valosähköisen ilmiön, Comptonin sironnan ja Rayleigh'n sironnan kautta. [26, 28]

Valosähköisessä ilmiössä atomi absorboi fotonin, joka luovuttaa koko energiansa yhdelle sen elektroneista. Osa fotonin energiasta muuttuu elektronin liike-energiaksi E_k yhtälön

$$E_k = E_\gamma - E_b \quad (3)$$

mukaisesti, missä E_γ on fotonin energia ja E_b elektronin sidosenergia [26]. Vapaa elektroni ei voi absorboida fotonia, vaan liikemäärän säilyminen edellyttää ytimen vastaanottavan osan fotonin liikemäärästä. Yli 80 % valosähköisestä absorptiosta



Kuva 2. Annihilaatofotonille mahdolliset vuorovaikutukset: a) valosähköinen ilmiö, b) Comptonin sironta ja c) Rayleigh'n sironta.

tapahtuukin sisimmällä eli K-elektronikuorella, jolla E_b on suurin [29], ja absorptio on sitä todennäköisempää, mitä lähempänä E_γ ja E_b ovat toisiaan [26]. Valosähköisen ilmiön vaikutusala K-kuorella ja yli 200 keV:n energioilla voidaan arvioida empiirisellä kaavalla

$$\sigma_{\text{vs.,K}} \approx Z^5 \sum_{n=1}^4 \frac{a_n + b_n Z}{1 + c_n Z} E_\gamma^{-p_n} \quad (4)$$

ja kokonaisvaikutusala kaavalla

$$\sigma_{\text{vs.}} \approx (1 + 0,01481 \ln^2 Z - 0,000788 \ln^3 Z) \sigma_{\text{vs.,K}}, \quad (5)$$

missä Z on atomin järjestysluku ja vakiot a_n , b_n , c_n ja p_n ovat sovituspärametreja [32]; vakiot on esitetty taulukossa I. Annihilaatofotonit vuorovaikuttavat valosähköisesti lähinnä raskaissa alkuaineissa, joita on esimerkiksi PET-kameran ilmaisimissa [26].

Comptonin sironnassa fotonit törmää heikosti sidottuun elektroniin, luovuttaa sille osan energiastaan ja muuttaa etenemissuuntaansa. Jos elektroni on aluksi levossa, fotonin energia ja liikemäärä muuttuvat yhtälöparin

$$\begin{cases} E'_\gamma = E_\gamma - E_k \\ \frac{E'_\gamma}{c} \cos \theta_C = \frac{E_\gamma}{c} - p_e \cos \theta_e \end{cases} \quad (6)$$

mukaisesti, missä E'_γ on sironneen fotonin energia, p_e elektronin liikemäärä, θ_C fotonin sirontakulma ja θ_e elektronin kulkusuunnan kulma verrattuna fotonin alkuperäiseen kulkusuuntaan [29]. Sironneen fotonin energia voidaan ilmaista sen alkuperäisen

Taulukko I. Kaavan (4) parametrien sovitusarvot [32].

n	a_n	b_n	c_n	p_n
1	$1,6268 \times 10^{-9}$	$-2,683 \times 10^{-12}$	$4,173 \times 10^{-2}$	1
2	$1,5274 \times 10^{-9}$	$-5,110 \times 10^{-13}$	$1,027 \times 10^{-2}$	2
3	$1,1330 \times 10^{-9}$	$-2,177 \times 10^{-12}$	$2,013 \times 10^{-2}$	3,5
4	$-9,1200 \times 10^{-11}$	0	0	4

energian ja sirontakulman funktiona [26]

$$E'_\gamma = \frac{E_\gamma}{1 + (E_\gamma/m_e c^2)(1 - \cos \theta_C)}. \quad (7)$$

Sironnan todennäköisyys kulman funktiona määritetään Kleinin–Nishinan yhtälöllä

$$\frac{d\sigma_C}{d\Omega} = \frac{Zr_e^2}{2} \left(\frac{E'_\gamma}{E_\gamma} \right)^2 \left(\frac{E'_\gamma}{E_\gamma} + \frac{E_\gamma}{E'_\gamma} - \sin^2 \theta_C \right), \quad (8)$$

missä $d\sigma_C/d\Omega$ on differentiaallinen sirontavaikutusala ja r_e klassinen elektronin säde [28, 29]. Annihilaatofotoni menettää eniten energiaa takaisinsironnassa ($\theta_C = 180^\circ$) mutta siroaa todennäköisimmin eteenpäin [28]. Comptonin sironta on vallitseva vuorovaikutus kevyitä alkuaineita sisältävissä materiaaleissa, kuten vedessä ja kudoksissa [26].

Rayleigh'n sironnassa fotonin kulkee atomin elektronipilven läpi ja vuorovaikuttaa vahvasti sidottujen elektronien kanssa. Toisin kuin Comptonin sironnassa, fotonin ei menetä energiaa eikä viritä tai ionisoi atomia mutta muuttaa etenemissuuntaansa. Rayleigh'n sironnan vaikutusala on yhtälön

$$\frac{d\sigma_R}{d\Omega} = \phi^2(\theta_R, Z) \frac{r_e^2}{2} (1 + \cos^2 \theta_R) \quad (9)$$

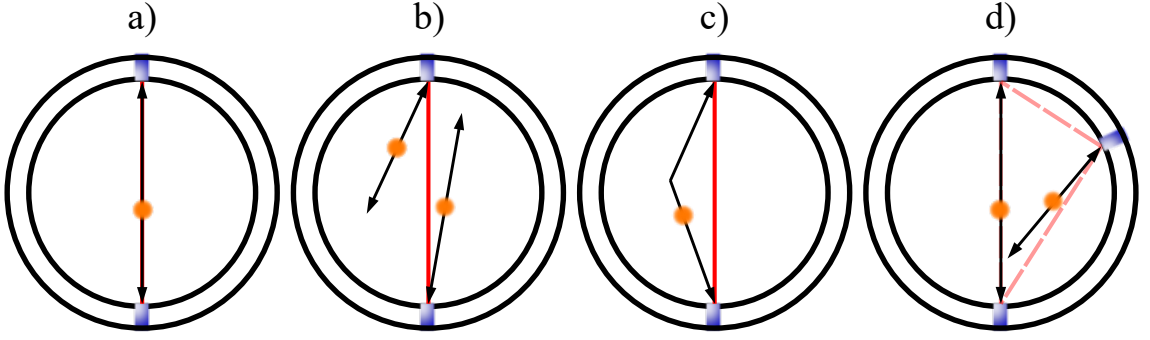
mukainen, missä $\phi(\theta_R, Z)$ on atomin muototekijä, jolla huomioidaan elektronitiheyden jakauma atomissa. Comptonin sironnan tavoin Rayleigh'n sironta rajoittuu annihilaatofotoneilla pienille, alle 20° kulmille; valosähköisen ilmiön tavoin se on tavallisempaa raskaissa kuin kevyissä alkuaineissa, mutta se on paljon muita vuorovaikutuksia harvinaisempi. [29, 33]

1.1.3 Kuvantamislaitteen rakenne ja toiminta

Tyypillinen PET-kuvantamislaitte koostuu kameran sisältävästä kanturista (engl. *gantry*), liikutettavasta pöydästä, signaalinkäsittelyelektroniikasta ja koinsidenssi-piiristä sekä raakadatan tallentamiseen ja valmiin kuvan muodostamiseen ja näyttämiseen käytettävistä tietokoneista [25, 27]. PET-kameran pitää voida havaita kaksi vastakkaisiin suuntiin etenevää, 511 keV -energiaista fotonia lähes yhtäaikaaisesti ja antaa tarkkaa tietoa niiden osumasijainnista ja -ajasta sekä energiasta, jotta niiden välinen vastesuora (engl. *line of response*, LOR) voidaan määrittää ilman geometrista kollimaatiota [34]. Nykyaikaisen PET-laitteen kanturissa on rengasmainen, ilmaisinlohkoihin jaettu kamera, jonka annihilaatiofotoneja havaitsevinä elementteinä toimivat valomonistininputkiin tai puolijohdevalodiodeihin kytketyt tuikekiteet: yhdessä ilmaisinelohkossa voi olla esimerkiksi 8×8 tuikekideä ja neljä valomonistinputkea [34].

Tuikeilmaisina säteilee näkyvää valoa korkeaenergiaisen fotonin osuessa siihen. Tuikekide on tavallisesti raskas alkalimetallihalidikide, johon on lisätty seostamalla epäpuhtauksia: yleisiä kideaineita ovat esimerkiksi NaI:Tl, BGO ja LYSO:Ce [34]. Kun γ -fotoni osuu tuikekiteeseen, se luovuttaa energiansa elektronille ja virittää sen valenssivyöltä johtavuusvyölle. Epäpuhtauksien takia virittynyt elektroni voi siirtyä valenssi- ja johtavuusvyön väliselle, normaalisti kielletylle energiatilalle, josta se palaa valenssivyölle emittoimalla näkyvän valon fotonin [28].

Valomonistinputkessa tuikekiteen emittoima valofotoni ohjataan katodille, jonka vapauttama elektroni vapauttaa edelleen sekundaarielektroneja putken sisäisistä dynodeista. Yksi fotoni voi näin synnyttää miljoonien elektronien muodostaman sähkövirran. Vaihtoehtoisesti fotoni voidaan ohjata tuikekiteeltä puolijohdevalodiodille, jossa fotoni muodostaa elektroni-aukkoparin. Vyöryvalodiodissa fotoelektronia kiihdytetään estojännitteellä siten, että puolijohteessa muodostuu lisää elektroni-aukkopareja. Valomonistinputkeen verrattuna vyöryvalodiodin kvanttihyötysuhde



Kuva 3. Erilaisia PET-kameran havaitsemia koinvidenssitapahtumia: a) todellinen, b) satunnainen, c) sirounut ja d) moninkertainen tapahtuma. Mustat nuolet esittävät fotonien lentoratoja ja punaiset janat vastesuoria; moninkertaisen tapahtuman mahdolliset vastesuorat on esitetty katkoviivalla.

on parempi mutta signaalin vahvistus ja ajallinen erotuskyky ovat huonommat. [28, 34]

PET-kuvantamisessa hyödynnetään sähköistä kollimaatiota annihilaatiofotoniparien tunnistamiseksi [25]. Jos kaksi γ -fotonia osuu vastakkaisille ilmaisimille ennalta määrätyn ajan eli koinvidenssi-ikkunan τ sisällä, ne yhdistetään yhdeksi koinvidenssipariksi ja tallennetaan jatkokäsittelyä varten. Kaikista yksittäistapahtumista, joissa yksi ilmaisimista havaitsee yhden γ -fotonin, vain 1–10 % rekisteröidään koinvidenssitapahtumina [35].

Todellisten koinvidenssitapahtumien lisäksi PET-dataan lukeutuu myös satunnaisia, sirouneita ja moninkertaisia koinvidenssitapahtumia, jotka on havainnollistettu kuvassa 3. Satunnaisessa tapahtumassa kaksi eri annihilaatioissa syntynyttä fotonia osuu kameraan koinvidenssi-ikkunan sisällä ja tulkitaan virheellisesti koinvidenssipariksi. Yksittäisen ilmaisimen fotonihavaintojen tiheyden ollessa C_i ilmaisiparin satunnaishavaintojen tiheys on

$$C_{\text{sat.},ij} = 2\tau C_i C_j. \quad (10)$$

Sironneessa tapahtumassa ainakin yksi havaituista fotoneista on Compton-sironnut ja muuttanut etenemissuuntaansa; vaikka fotoni menettää energiaa Comptonin siroonnassa, suurimmalla osalla PET-kameroista on liian huono erotuskyky sirounei-

den fotonien tunnistamiseen. Sironneiden koincidenssien osuus riippuu kuvannettavan kohteen rakenteesta ja aktiivisuusjakaumasta. Moninkertaisessa tapahtumassa ilmaisimet havaitsevat enemmän kuin kaksi fotonia yhden koincidenssi-ikkunan sisällä. Satunnaisille ja sironneille tapahtumille määritetyt vastesuorat eivät lävistä todellisia annihilaatiopisteitä, joten ne antavat väärää tietoa kohteen aktiivisuusjakaumasta; moninkertaisille tapahtumille ei puolestaan voida määrittää vastesuoraa yksiselitteisesti, joten ne yleensä hylätään. [35]

1.2 Monte Carlo -simulaatio

Monte Carlo -menetelmissä tarkasteltavan ongelman ratkaisu esitetään tunnettua todennäköisyysjakaumaa noudattavan satunnaislukujoukon funktiona, jonka arvo selvitetään kerätyn satunnaislukuotoksen avulla. Itse ongelma tai sen ratkaisu voi olla luonteeltaan tilastollinen tai täysin deterministinen [36]. Monte Carlo -simulaatiossa näillä menetelmillä tutkitaan todellista järjestelmää ja sen ominaisuuksia: järjestelmälle ja sen sisältämille olioille muodostetaan matemaattinen malli, joka alustetaan satunnaislukuarvoilla, joilla lasketaan tutkittavien parametrien arvot. Kun näitä laskuja toistetaan jopa miljardeja kertoja eri alkuarvoilla, simuloituilla tuloksilla voidaan arvioida mallinnettavan järjestelmän ominaisuuksia [37].

Monte Carlo -menetelmät voidaan ainakin muodollisesti samaistaa integroimiseen: kun tutkittavan ongelman ratkaisua arvioidaan tasajakaumasta $U(0, 1)$ kerättyjen satunnaislukujen funktiona $f(U_1, U_2, \dots, U_n)$, sen odotusarvona voidaan pitää määrättyä integraalia

$$\langle f \rangle = \int_0^1 \int_0^1 \dots \int_0^1 f(u_1, u_2, \dots, u_n) du_1 du_2 \dots du_n. \quad (11)$$

Käytännössä funktiolle f lasketaan hyvin monta arvoa, joiden keskiarvo on suurten lukujen lain nojalla

$$\bar{f} = \frac{1}{N_f} \sum_{i=1}^{N_f} f(U_{i,1}, U_{i,2}, \dots, U_{i,n}) \approx \langle f \rangle, \quad (12)$$

missä N_f on toistojen lukumäärä; keskeisen raja-arvolauseen nojalla tämä keskiarvo on normaalisti jakautunut. [36, 38]

Monte Carlo -menetelmän käyttö edellyttää hyvää näennäissatunnaislukugeneraattoria (PRNG), jonka muodostamaa lukujonoa ei käytännössä voida erottaa riippumattomien ja identtisesti jakautuneiden satunnaislukujen jonosta. PRNG:n jakson täytyy olla niin pitkä, etteivät sen tuottamat lukuarvot toistu simulaation aikana, mutta sen pitää tuottaa täsmälleen sama lukujono joka kerta, kun sille syötetään tietty alkuarvo eli siemen. Sen pitäisi myös olla nopea, muistitehokas, halpa sekä helppo asentaa ja käyttää. [39]

Tasajakauman $U(0, 1)$ avulla on mahdollista simuloida muitakin todennäköisyysjakaumia. Jos satunnaismuuttujan W kertymäfunktio F_W on helposti ratkaistava käänteisfunktio

$$F_W^{-1}(u) = \inf\{w : F_W(w) \geq u\}, \quad (13)$$

muuttujan W arvo voidaan määrittää käänteismuunnosmenetelmällä, jossa poimitaan U jakaumasta $U(0, 1)$ ja lasketaan

$$W = F_W^{-1}(U). \quad (14)$$

Yleisesti muunnosmenetelmillä voidaan simuloida sellaisia jakaumia, jota noudattavia satunnaismuuttujia on helppo konstruoida kaavalla (14) laskettavien muuttujien avulla. Joitakin jakaumia on vaikeaa tai mahdotonta simuloida tällä tavalla. Jos kuitenkin muuttujan W todennäköisyystiheysfunktio (PDF) f_W ja helposti simuloitavasta PDF:sta g tiedetään, että yleisesti $f_W(w) \leq Qg(w)$ jollakin vakiolla Q , jakauman f_W mallintamiseen voidaan käyttää hyväksyntä-hylkäysmenetelmää. Siinä poimitaan U jakaumasta $U(0, 1)$ ja W jakaumasta g ja hyväksytään W vain, jos

$$U \leq \frac{f_W(W)}{Qg(W)}; \quad (15)$$

muulloin W hylätään ja poimitaan uudet satunnaisluvut. [38–40]

Joissakin malleissa satunnaismuuttujien arvot eivät ole täysin toisistaan riippumattomia. Markovin ketju on satunnaismuuttujien sarja $W_1, W_2, \dots$, jossa muuttujan W_1 arvo poimitaan jostakin alkujakaumasta ja muuttujan W_{n+1} arvon todennäköisyysjakauma riippuu sitä edeltävästä muuttujasta W_n mutta ei muista [41]. Markovin ketju Monte Carlo -menetelmässä tarkasteltavan järjestelmän tilaa käsitellään Markovin ketjuna, joka alustetaan jollakin satunnaisluvulla W_1 ja päivitetään arvoon W_{n+1} arvosta W_n riippuvalla todennäköisyydellä [41]. Hiukkasen kuljettamista voidaan mallintaa Markovin ketjulla: jos hiukkasen tila jollakin ajanhetkellä määrittyy sen sijainnilla $\mathbf{r}_n$, energialla E_n ja etenemissuunnalla $\mathbf{d}_n$, niin sen seuraava tila $\mathbf{r}_{n+1}$, E_{n+1} , $\mathbf{d}_{n+1}$ riippuu edeltävästä tilasta, hiukkasen kulkemasta matkasta, matkan päättävästä vuorovaikutuksesta sekä siinä mahdollisesti menetetyistä energiasta ja sirontakulmasta [42].

1.3 Lääketieteellinen kuvarekonstruktio

Kuvarekonstruktio on matemaattinen operaatio, jossa kuvantamislaitteen havainnot muunnetaan kuvannettua kohdetta tai sen sisältämää merkkiainetta esittäväksi kuvaksi. Tätä varten on kehitetty lukuisia analyyttisiä ja iteratiivisia laskenta-algoritmeja, joista jälkimmäisiä käytetään nykyään enemmän PET-kuvantamisessa [10].

1.3.1 Kohteen ja havaintojen matemaattiset esitykset

PET-merkkiaineen jakaumaa voidaan teoriassa käsitellä jatkuvana funktiona $C(\mathbf{r}, t)$, joka kertoo aktiivisuustiheyden paikassa $\mathbf{r}$ ajanhetkellä t . Sellaisen mittaaminen ja sitä esittävän kuvan rekonstruoiminen vaatisi kuitenkin äärettömän monta koinssidenssihavaintoa äärettömän monelta vastesuoralta. Onkin käytännöllisempää muodostaa merkkiaineen jakaumalle diskreetti esitys: staattisessa kuvantamisessa aktiiv-

visuusjakauma voidaan kirjoittaa J kantafunktion $\alpha_j(\mathbf{r})$ avulla muodossa

$$C(\mathbf{r}) \approx \sum_{j=1}^J c_j \alpha_j(\mathbf{r}). \quad (16)$$

Tavallisesti kantafunktiot vastaavat yksittäisiä vokseleita (vx) ja ovat paikan suhteen täysin erilliset, jolloin kukin kerroin c_j on yksinkertaisesti vokselin j aktiivisuustiheys ja voidaan esittää integraalina

$$c_j = \int C(\mathbf{r}) \alpha_j(\mathbf{r}) d\mathbf{r}. \quad (17)$$

Iteratiivisessa rekonstruktiossa voidaan kuitenkin käyttää myös osittain päällekkäisiä kantafunktioita. [43]

PET-kuvantamisdata voidaan esittää listamuotoisena, jolloin jokainen koinssidenssitapahtuma, sen havainnut ilmaisipari sekä esimerkiksi fotonien energiat ja havaintojen välinen aikaero on luetteloitu erikseen. Mahdollisten vastesuorien ja havaintojen suuren lukumäärän takia on nykyään tavallisempaa muodostaa datasta sinogrammeja, joissa yksittäisten koinssidenssien sijaan ilmoitetaan kaikilla vastesuoran kulmilla φ ja etäisyyksillä s näkökentän (engl. *field of view*, FOV) keskipisteestä tehtyjen havaintojen lukumäärät. Jokaiselle rengasparille muodostetaan oma sinogrammi, mutta dataa voidaan myös tiivistää yhdistämällä vierekkäisiltä vastesuorilta tehdyt havainnot yhdeksi datapisteeksi. [10]

Kohinaista PET-dataa sisältävälle vektorille $\mathbf{m}$, jonka alkio m_i sisältää ilmaisiparin i yhdistävällä vastesuoralla havaittujen koinssidenssien lukumäärän, voidaan arvioida kohinatonta odotusarvovektori $\mathbf{q}$ kaavalla

$$q_i \equiv \langle m_i \rangle = \int C(\mathbf{r}) \zeta_i(\mathbf{r}, \mu) d\mathbf{r}, \quad (18)$$

missä $\zeta_i(\mathbf{r}, \mu)$ on ilmaisiparin i herkkyys lineaarisen vaimenemiskertoimen ollessa $\mu(\mathbf{r})$. Käytännössä $\zeta_i(\mathbf{r}, \mu)$ kertoo todennäköisyyden, jolla paikassa $\mathbf{r}$ tapahtunut positroniemissio on aiheuttanut vastesuoralla i havaitun koinssidenssin. Todelliset aktiivisuusarvot sisältävä kuvavektori $\mathbf{c}$ voidaan yhdistää vektoriin $\mathbf{q}$ lineaarikuvauksella

$$\mathbf{q} = \mathbf{A}\mathbf{c}, \quad (19)$$

missä $\mathbf{A}$ on PET-kuvantamissysteemiä mallintava matriisi eli systeemimatriisi. Sen yksittäinen alkio

$$A_{ij} = \int \zeta_i(\mathbf{r}, \mu) \alpha_j(\mathbf{r}) d\mathbf{r} \quad (20)$$

on siten painotuskerroin, joka määrittää kantafunktiota j vastaavan aktiivisuuden osuuden datapisteen i arvosta. [43]

1.3.2 Iteratiiviset rekonstruktioalgoritmit

PET-kuvantamisen rekonstruktio menetelmät voidaan luokitella analyyttisiin, suoriin ja iteratiivisiin menetelmiin [10]. Analyyttisessä rekonstruktiossa käytetään projektionmuotoista dataa, joka samaistetaan jatkuvan aktiivisuustiheysfunktion $C(\mathbf{r})$ viivaintegraaleihin [43]. Nämä menetelmät ovat nopeita ja helppoja toteuttaa, mutta niissä data oletetaan kohinattomaksi eikä esimerkiksi ilmaisimien havaintokyvyn vaihtelua huomioida [9]. Suorissa rekonstruktio menetelmissä pyritään laskemaan diskreetti kuva $\mathbf{A}^{-1}\mathbf{m}$ joko suoraan tai likimääräisesti arvioiden esimerkiksi systeemimatriisin $\mathbf{A}$ hajotelman avulla [43]. Käänteismatriisia $\mathbf{A}^{-1}$ ei kuitenkaan yleensä ole olemassa, koska $\mathbf{A}$ on harvoin neliömatriisi, ja pienikin määrä kohinaa voi johtaa isoihin muutoksiin kuvan $\mathbf{c}$ estimaatissa tai tehdä sen selvittämisestä suoraan mahdotonta [11, 43]. Likimääräisenkin ratkaisun muodostaminen on usein epäkäytännöllistä, koska systeemimatriisissa voi olla jopa $\sim 10^{16}$ alkioita [10, 43].

Iteratiivisessa rekonstruktiossa aktiivisuusjakaumalle muodostetaan alkuarvaus $\mathbf{c}^{(0)}$, jolle lasketaan projektio $\mathbf{q}^{(0)}$ yhtälön (19) mukaisesti. Jokaisella iteraatiokerralla k laskettua projektiota $\mathbf{q}^{(k)}$ verrataan datavektoriin $\mathbf{m}$ ja arvausta $\mathbf{c}^{(k)}$ korjataan ennalta määrätyn kriteerin pohjalta [9]. Tätä kriteeriä esitetään kohdefunktiolla, joka mittaa vektorien $\mathbf{m}$ ja $\mathbf{q}^{(k)}$ välistä etäisyyttä: useimmin käytettyjä kohdefunktioita ovat suurin uskottavuus (engl. *maximum likelihood*, ML), suurin jälkitodennäköisyys (engl. *maximum a posteriori*) ja pienin neliösumma (engl. *least squares*) [10, 43].

ML-algoritmit etsivät suurimman arvon uskottavuusfunktiolle $P(\mathbf{m}|\mathbf{c})$ muodos-

taakseen parhaan kuvan $\mathbf{c}$. PET-kuvantamisessa fotonihavainnot noudattavat Poissonin jakaumaa, joten uskottavuusfunktio on muotoa

$$P(\mathbf{m}|\mathbf{c}) = \prod_{i=1}^{N_{\text{LOR}}} \frac{q_i^{m_i}}{m_i!} e^{-q_i}, \quad (21)$$

missä N_{LOR} on vastesuorien lukumäärä. Laskujen helpottamiseksi käytetään tavallisesti uskottavuusfunktion logaritmia

$$\ln P(\mathbf{m}|\mathbf{c}) = \sum_{i=1}^{N_{\text{LOR}}} (m_i \ln q_i - \ln m_i! - q_i). \quad (22)$$

Suurimmassa osassa ML-algoritmeista iteraatiokerran k arvausta $\mathbf{c}^{(k)}$ korjataan uskottavuusfunktion paikallisen gradientin avulla:

$$c_j^{(k+1)} = c_j^{(k)} + \lambda^{(k)} \frac{\partial}{\partial c_j} \ln P(\mathbf{m}|\mathbf{c}) = c_j^{(k)} + \lambda^{(k)} \left(\sum_{i=1}^{N_{\text{LOR}}} A_{ij} \frac{m_i}{q_i^{(k)}} - \sum_{i=1}^{N_{\text{LOR}}} A_{ij} \right), \quad (23)$$

missä $\lambda^{(k)}$ on askelkoko; gradientin ensimmäinen termi on korjauskertoimien $m_i/q_i^{(k)}$ takaisinprojektio ja toinen termi herkkyyskuva (engl. *sensitivity image*). [11, 43]

MLEM-algoritmissa (*maximum likelihood expectation maximization* eli suurimman uskottavuuden odotusarvon maksimointi) kaavan (23) askelkooksi valitaan

$$\lambda^{(k)} = \frac{c_j^{(k)}}{\sum_{i=1}^{N_{\text{LOR}}} A_{ij}}. \quad (24)$$

Iteraatiokaava on siten

$$c_j^{(k+1)} = \frac{c_j^{(k)}}{\sum_{i=1}^{N_{\text{LOR}}} A_{ij}} \sum_{i=1}^{N_{\text{LOR}}} A_{ij} \frac{m_i}{\sum_j^{N_{\text{vx}}} A_{ij} c_j^{(k)}}, \quad (25)$$

missä N_{vx} on vokselien lukumäärä, tai matriisimuodossa

$$\mathbf{c}^{(k+1)} = \frac{\mathbf{c}^{(k)} \mathbf{A}^T \mathbf{m}}{\mathbf{A}^T \mathbf{1} \mathbf{A} \mathbf{c}^{(k)}}, \quad (26)$$

missä $\mathbf{1}$ on ykkösmatriisi [43]. MLEM suppenee johdonmukaisesti ja ennustettavasti mutta hitaasti ja yleensä kohti kohinaista ratkaisua [9]. Useimmin käytetty muokattu MLEM-algoritmi on OSEM (*ordered subsets expectation maximization* eli järjestettyjen osajoukkojen odotusarvon maksimointi), jossa kuvantamisdata jaetaan N_{OS}

osajoukkoon ja arvauksen $\mathbf{c}^{(k)}$ parantamiseen käytetään yhtä osajoukkoa kerrallaan:

$$c_j^{(k,n)} = \frac{c_j^{(k,n-1)}}{\sum_{i \in S_n} A_{ij}} \sum_{i \in S_n} A_{ij} \frac{m_i}{\sum_j^{N_{\text{vx}}} A_{ij} c_j^{(k,n-1)}}, \quad (27)$$

missä S_n on n :s vastesuorien osajoukko [10]. OSEM on noin N_{OS} kertaa nopeampi kuin MLEM, mutta se ei välttämättä suppene ML-arvoon [9].

1.3.3 Havaintodatan korjaukset

PET-kuvantamisessa useat virhelähteet vaikeuttavat β^+ -hajoamisten havaitsemista tai lisäävät havaintodataan kohinaa. Todellisten annihilaatiofotoniparien lisäksi PET-data sisältää myös satunnaisia ja sironneita koinssidensseja, joten lineaarikuvausta (19) tarkempi malli datalle on muotoa

$$\mathbf{q} = \mathbf{A}\mathbf{c} + \mathbf{q}_{\text{sat.}} + \mathbf{q}_{\text{sir.}}, \quad (28)$$

missä $\mathbf{q}_{\text{sat.}}$ ja $\mathbf{q}_{\text{sir.}}$ ovat arvioidut satunnais- ja sirontavektorit. Todellisetkaan koinssidenssit eivät sisällä kaikkia kuvantamisen aikana syntyneitä γ -fotonipareja vaimenemisen sekä kohteen ja kameran geometrian takia. [10]

Monet virhelähteistä voidaan sisällyttää systeemimatriisiin, joka voidaan hajottaa yksittäisiä kuvanlaatuun vaikuttavia ilmiöitä esittäviin tekijöihin esimerkiksi seuraavasti:

$$\mathbf{A} = \delta \mathbf{N} \mathbf{R} \mathbf{M} \mathbf{G} \mathbf{B}, \quad (29)$$

missä δ on kaikille vastesuorille yhteinen korjauskerroin, $\mathbf{N}$ normalisaatiomatriisi, $\mathbf{R}$ projektiosumennusmatriisi, $\mathbf{M}$ vaimennusmatriisi, $\mathbf{G}$ geometrinen projektiomatriisi ja $\mathbf{B}$ kuvasumennusmatriisi. Kuvasumennusmatriisin alkio $B_{jj'}$ kuvaa todennäköisyyttä, että vokselissa j havaittu hajoaminen on oikeasti tapahtunut vokselissa j' , ja mallintaa esimerkiksi positronin kantamaa ja tuikeilmaisimen erotuskykyä. Geometrisen projektiomatriisin alkio G_{ij} kertoo todennäköisyyden havaita vokselista j emittoitunut fotonipari vastesuoralla i systeemin geometrian ja mahdollisesti foto-

nien lentoajan perusteella. Lävistäjämatrissi $\mathbf{M}$ sisältää kaikkien vastesuorien kokonaisvaimenemiskertoimet:

$$M_{ii} = \exp \left(- \sum_{j=1}^{N_{\text{vx},i}} l_{ij} \mu_j \right), \quad (30)$$

missä $N_{\text{vx},i}$ on vastesuoran i leikkaamien vokselien lukumäärä ja l_{ij} vokselia j leikkaavan janan pituus. Projektiosumennusmatrissin alkio $R_{ii'}$ vastaa todennäköisyyttä, että vastesuoralla i havaittu fotonipari on oikeasti peräisin vastesuoralta i' , mihin vaikuttavat esimerkiksi kiteiden välinen sironta ja parallaksivirhe. Lävistäjämatrissi $\mathbf{N}$ sisältää kunkin ilmaisiparin suhteelliset herkkyudet annihilaatiofotoneille. Skaalaari δ voi kattaa esimerkiksi hajoamis- ja kuollut aika -korjaukset. Systemimatriisiin voidaan sisällyttää myös sirontamatrissi, joka sisältää sironnan todennäköisyydet jokaiselle vastesuoralle. [10]

Satunnaisille koinsidensseille on kaksi yleistä korjaustapaa. Yksittäistapahtumien lukumäärä kullakin vastesuoralla voidaan arvioida, minkä jälkeen satunnaisten tapahtumien osuus voidaan laskea yhtälön (10) avulla ja vähentää jokaisesta vastesuorasta. Toinen tapa on kahden koinsidenssi-ikkunan käyttö: toisessa koinsidenssipiirissä yhdeltä ilmaisimelta saapuvaa pulssia viivästetään niin pitkään, että se ei voi muodostaa todellista koinsidenssiparia toiselta ilmaisimelta saapuvan pulssin kanssa. Viivästetyt havainnot tai niillä muodostetut sinogrammit vähennetään toisen koinsidenssipiirin käsittelemistä havainnoista. [44]

Sirontakorjausta varten on kehitetty monenlaisia tekniikoita. Sironneiden koinsidenssien osuus voidaan arvioida ja vähentää kaikista havainnoista sovittamalla analyyttinen funktio kuvannuskohteen ulkopuolisiin vastesuoriin tai konvolvoimalla havainnot kokeellisesti arvioidulla sirontavastefunktiolla. Monienergiaikkunatekniikoissa (engl. *multiple energy window technique*) mitataan erillisillä energiaikkunoilla sellaisten koinsidenssien lukumäärät, joiden energia on yli tai huomattavasti alle 511 keV; edellisten oletetaan sisältävän pelkästään siroamattomia fotoneita, kun taas jälkimmäiset sisältävät suurimman osan sironneista fotoneista. Sironneiden koinsidens-

sien jakaumaa voidaan myös mallintaa analyttisesti tai Monte Carlo -menetelmillä. [44]

Koska vaimennus ei riipu säteilylähteen sijainnista, se voidaan määrittää kuvannuskohteen ulkopuolisella lähteellä. Tällöin kuvannuskohteen läpäisevän säteilyn intensiteettiä verrataan ilman kohdetta mitattuun intensiteettiin. Yhdistelmäkameroi- den tultua käyttöön vaimennuskorjauksia on tehty myös PET-datan kanssa samaan aikaan saadulla tietokonetomografia- (TT) tai magneettikuvausdatalla. Molemmissa tapauksissa säteilyn energia poikkeaa yleensä 511 keV:sta, joten vaimennuskertoimet täytyy skaalata jälkikäteen. [10, 44]

Normalisaatiolla korjataan ilmaisiparien herkkyyksien vaihtelut. Komponentteihin perustuvassa normalisaatiossa (engl. *component-based normalization*) normalisaatiomatriisi $\mathbf{N}$ jaetaan erilaisia herkkyyteen vaikuttavia osatekijöitä mallintaviin komponentteihin. Jos $N_{hih'i'}$ on alkio, jota vastaava vastesuora yhdistää PET-kameran renkaalla h sijaitsevan ilmaisinkiteen i kiteeseen i' renkaalla h' , niin se voidaan esittää tulona

$$N_{hih'i'} = \epsilon_{hi}\epsilon_{h'i'}b_{hi}^{\text{tr.}}b_{h'i'}^{\text{tr.}}b_h^{\text{ax.}}b_{h'}^{\text{ax.}}\tau_{hih'i'}g_{hih'i'}^{\text{tr.}}g_{hh'}^{\text{ax.}}\kappa_{hih'i'}. \quad (31)$$

Ilmaisimien sisäiset tehokkuustekijät (engl. *intrinsic detector efficiency factor*) ϵ esittävät kiteiden rakenteesta ja valomonistinputkien toiminnasta riippuvaa ilmaisimen tehokkuuden satunnaista vaihtelua. Transaksiaaliset lohkoprofilitekijät (engl. *block profile factor*) $b^{\text{tr.}}$ kuvaavat kunkin kiteen tehokkuuden riippuvuutta sijainnista ilmaisinelohkolla ja renkaalla, kun taas aksiaaliset lohkoprofilitekijät $b^{\text{ax.}}$ ovat kokonaisten renkaiden suhteellisia tehokkuuksia. Transaksiaaliset geometriset tekijät $g^{\text{tr.}}$ esittävät kideparien herkkyyksien riippuvuutta esimerkiksi fotonin tulokulmasta sekä kiteiden välisestä häirinnästä lohkon sisällä ja aksiaaliset geometriset tekijät $g^{\text{ax.}}$ puolestaan rengasparien suhteellisia herkkyyksiä. Aikaikkunoiden kohdistustekijät (engl. *time-window alignment factor*) τ kuvaavat kideparien ajastinsignaalien synkronointivirheitä. Rakenteelliset kohdistusvirhetekijät (engl. *structural misalignment*

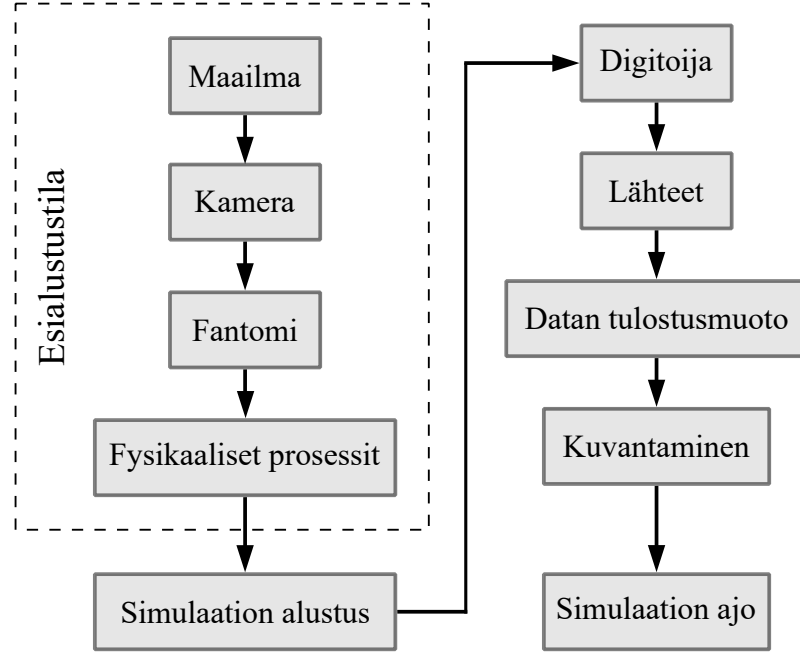
factor) κ esittävät kideparien virheellisestä kohdistamisesta johtuvaa herkkyyden vaihtelua. Tekijät ϵ , b^{tr} , b^{ax} ja g^{ax} voidaan laskea sylinterilähteellä pienellä aktiivisuudella ja pitkällä kuvantamisajalla tehdyistä mittauksista; muiden laskemiseen käytetään tavallisesti kohdetta kiertävää säteilylähdettä, taso- tai viivalähdettä. [44]

2 Ohjelmistot

Tässä luvussa esitellään työssä käytetyt tietokoneohjelmistot. Simulaatiot toteutettiin GATE V9.3:lla CSC:n cPouta-pilvipalvelussa virtuaalikoneella (kuusi ydintä, 16 Gt muistia, 80 Gt tallennustilaa). Se ajettiin Qt-moodissa, mikä mahdollistaa kameran ja fantomien visualisoinnin. Visualisointiin käytettiin Geant4 Visualization System -järjestelmää [45] OpenGL-katselutilassa. Datatiedostot siirrettiin SSH-yhteyden avulla henkilökohtaiselle tietokoneelle, jossa kuvat rekonstruoidtiin OMEGA 1.2.1:llä MATLABissa (R2024a).

2.1 GATE

GATE on rakennettu siten, että sitä on mahdollista käyttää monenlaisten isotooppilääketieteellisten tekniikoiden mallintamiseen ja helppoa kehittää uusia sovelluksia varten. Sen ydin sisältää C++-kielellä ohjelmoituja Geant4:n perusluokkia ja työkaluja, joita käytetään esimerkiksi geometrioiden, lähteiden ja fysikaalisten prosessien määrittämiseen, hiukkasten kuljettamiseen ja simulaation visualisoimiseen, sekä GATE:ssa määritellyjä perusluokkia kuten virtuaalisen kellon. Perusluokista johdetut luokat, jotka edustavat erityisesti lääketieteellisen kuvantamisen tai sädehoidon mallintamiseen tarvittavia olioita ja prosesseja, muodostavat GATE:n sovelluskerroksen. Mallinnettavan systeemin kanssa vuorovaikuttaminen ei vaadi C++-kielen hallitsemista, vaan se tehdään käyttäjäkerroksesta käsin yksittäisten komentojen tai komentosarjojen avulla. [19, 46]



Kuva 4. GATE-simulaation vaiheet. Suoritusjärjestys on muuten vapaa, paitsi geometriat ja fysiikka täytyy määrittää ennen simulaation alustamista. [47]

GATE-simulaation rakenne on esitetty kuvassa 4. Aluksi täytyy määrittellä maailma, jonka sisällä koko simulaatio tapahtuu, ja käytettävissä olevat materiaalit. Kun maailman sisältämät oliot – PET-kamera ja fantomi – on kuvailtu ja on valittu fysiikan malli, johon eri vuorovaikutusten todennäköisyydet perustuvat, simulaatio alustetaan: tässä vaiheessa vaikutusalat lasketaan sekä geometriat ja fysiikka lukitaan. Alustuksen jälkeen määritetään signaalinkäsittely, radioaktiiviset lähteet, havaintojen tulostusmuoto, kuvantamisaika ja satunnaislukugeneraattorin siemen. Lopuksi käynnistetään kuvantaminen, jonka aikana hiukkasia luodaan ja kuljetetaan ja kameran tekemät havainnot tallennetaan. [47]

Radioaktiivinen lähde alustetaan tietyllä sijainnilla, muodolla ja aktiivisuudella, emittoitujen hiukkasten lajilla sekä emission suunnalla ja energiaspektrillä. Kahden emissiotapahtuman välinen aika δt poimitaan eksponentiaalisesta todennäköisyysjakaumasta

$$p(\delta t) = A(t)e^{-A(t)\delta t}, \quad (32)$$

missä

$$A(t) = A_0 e^{\frac{t}{T}} \quad (33)$$

on lähteen aktiivisuus simulaatiohetkellä t , T lähteen elinaika ja A_0 aktiivisuus simulaation alussa [19].

Simuloitavia hiukkasia kuljetetaan askel kerrallaan. Fysikaaliseen vuorovaikutukseen päättyvää askelta luonnehtii hiukkasen keskimääräinen vapaa matka

$$\lambda = \frac{1}{\rho_M} \left(\sum_i \frac{\xi_i \sigma_i}{M_i} \right)^{-1}, \quad (34)$$

missä ρ_M on väliaineen tiheys ja M_i , ξ_i , ja σ_i ovat sen sisältämän isotoopin i massa, massaosuus ja kyseisen vuorovaikutuksen vaikutusala. Hiukkasen todennäköisyys edetä matka l vuorovaikuttamatta on

$$P(l) = e^{-n_\lambda}, \quad (35)$$

missä

$$n_\lambda = \int_0^l \frac{dl}{\lambda}. \quad (36)$$

Kun GATE on luonut hiukkasen, se valitsee jokaiselle mahdolliselle vuorovaikutukselle satunnaisluvun u jakaumasta $U(0, 1)$, laskee eksponentin n_λ arvolla $P(l) = u$ ja määrittää näin kyseiseen vuorovaikutukseen päättyvän askelen pituuden senhetkessä ympäristössä. Varatun hiukkasen, joka menettää energiaa jatkuvasti, askel voi päättyä myös energian loppumiseen. Lisäksi hiukkanen ei saa kulkea kahden aineen rajapinnan läpi askelen aikana, vaan rajapinnalla askelen täytyy katketa. Käyttäjä voi myös asettaa askelelle maksimipituuden. Askelen pituudeksi valitaan aina pienin näistä vaihtoehdoista. [20]

GATE mallintaa γ -fotonien realistista havaitsemista ja signaalinkäsittelyä digitoijaketjulla. Kun fotonin askel päättyy ”herkäsi ilmaisimeksi” määritetyssä kappaleessa, se tallennetaan osumana. Osuma sisältää esimerkiksi askelen päättävän vuorovaikutuksen, sen sijainnin ja ajan sekä fotonin luovuttaman energian. Digitoi-

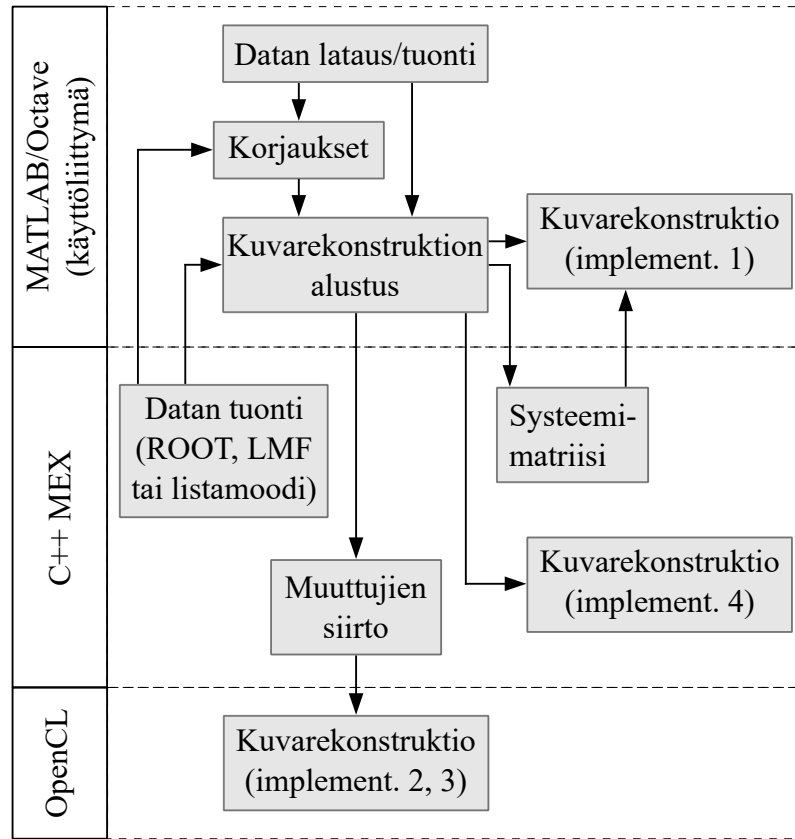
ja muuntaa yhden fotonin osumat lopulliseksi lukemaksi eli yksittäistapahtumaksi. Ensimmäinen komponentti eli summain (engl. *adder*) yhdistää osumat yhdeksi pulssiksi, ja lukija (engl. *readout*) yhdistää tietyn ilmaisinyhmän pulssit yhdeksi summapulssiksi. Lukijaa seuraavat vaste- eli resoluutiomoduurit lisäävät kohinaa summapulssin energiaan sekä fotonin havaintoaikaan ja -paikkaan. Oletusarvoisesti energiavastemoduuli lisää Gauss-sumennuksen, jonka puoliarvoleveys (FWHM) lasketaan kaavalla

$$R_E = R_0 \frac{\sqrt{E_0}}{\sqrt{E_\gamma}}, \quad (37)$$

missä R_0 on FWHM energian viitearvolla E_0 . Aika- ja paikkavastemoduulit lisäävät vastaavasti Gauss-sumennuksen FWHM-arvoilla R_t , R_x , R_y ja R_z . Energiaikkunamoduuli hylkää sellaiset pulssit, joiden energia on liian pieni tai suuri. Digitoijalla voidaan myös mallintaa esimerkiksi ilmaisinosien hyötysuhdetta, kuollutta aikaa ja sisäistä kohinaa. Digitoijan viimeinen osa on koinsidenssilajittelija, joka yhdistää kaksi yksittäistapahtumaa koinsidenssipariksi, jos niiden havaintoaikojen erotus on valittua aikaikkunaa pienempi. [19, 47]

2.2 OMEGA

OMEGA koostuu kolmesta eri ohjelmointikielillä ja alustoilla toimivasta kerroksesta, jotka on esitetty kuvassa 5. Suurin osa OMEGAN toiminnoista suoritetaan päällimmäisessä kerroksessa MATLABissa tai GNU Octavessa, jossa komentoja voidaan muokata vapaasti. Kaikki tehtävät havaintojen lataamisesta ja korjaamisesta kuvan rekonstruoimiseen on mahdollista suorittaa tässä kerroksessa, mutta se voi olla hidasta ja vaatia paljon muistia. Prosessia voidaan nopeuttaa ja tehostaa hyödyntämällä keskimmäisen kerroksen C++ MEX -tiedostoja tai pohjimmaisen kerroksen OpenCL-tiedostoja, jotka mahdollistavat esimerkiksi ROOT-tiedostoille tai listamoodissa tallennettujen havaintojen lataamisen, datan muuntamisen sinogrammeiksi, systeemimatriisin muodostamisen sekä kuvan rekonstruoimisen myös ilman



Kuva 5. OMEGAn rakenne ja sen eri kerrosten ydinfunktioita [10].

systeemimatriisia. [10]

Rekonstruktio voidaan toteuttaa OMEGAssa neljällä tavalla eli implementaatiolla. Implementaatioissa 1 systeemimatriisi $\mathbf{A}$ lasketaan C++ MEX -tiedostossa mutta muutoin rekonstruktio toteutetaan täysin MATLAB-komentojen avulla; tämä rekonstruktio on hitain ja vie paljon muistia. Implementaatioissa 2, 3 ja 4 projektiio $\mathbf{A}\mathbf{c}$ ja takaisinprojektiio $\mathbf{A}^T\mathbf{m}$ lasketaan muodostamatta systeemimatriisia. Implementaatio 2 toteutetaan OpenCL-tiedostossa, hyödyntää ArrayFire-kirjaston funktioita ja on nopein ja monipuolisin. Implementaatio 3 toteutetaan myös OpenCL-tiedostossa ja hyödyntää rinnakkaislaskentaa useassa GPU:ssa taikka CPU:ssa ja GPU:ssa, mutta sillä voidaan laskea vain MLEM- tai OSEM-rekonstruktio. Implementaatio 4 toteutetaan MEX-tiedostossa OpenMP:n avulla, ja siinä laskut suoritetaan rinnakkain CPU:ssa, mutta se ei ole yhtä nopea kuin implementaatiot 2 ja

3. Implementaatioissa 1 ja 4 laskennassa käytetään kaksoistarkkuuden liukulukuja, kun taas implementaatioissa 2 ja 3 käytetään perustarkkuuden liukulukuja, mutta erot lopullisissa tuloksissa ovat yleensä hyvin pieniä. [10, 48]

OMEGA ei muodosta systeemimatriisin tekijöitä erikseen, mutta sitä voidaan tarkastella kaavassa (29) esitettynä hajotelmana. Kuvasumennusmatriisi $\mathbf{B}$ toteutetaan Gaussin pistevastefunktiolla; projektiosumennusta $\mathbf{R}$ ei ole mallinnettu [10]. Geometriset projektiot eli matriisin $\mathbf{G}$ alkiot voidaan laskea OMEGAssa parannelulla Siddonin algoritmilla [49], ortogonaaliseen etäisyyteen perustuvalla säteenjäljittimellä (engl. *orthogonal distance-based ray tracer*, ODRT) [50] tai tilavuusperusteisella säteenjäljittimellä (engl. *volume-based ray tracer*, VRT) [51]. Siddonin algoritmissa seurataan vastesuoraa PET-kameran FOV:n sisällä, lasketaan sen läpäisemille vokseleille vastaavien leikkausjanojen pituudet ja normalisoidaan ne koko FOV:n leikkaavan janan pituudella [49]. ODRT-algoritmissa käytetään leikkausjanan pituuden sijaan vastesuoran ja vokselin keskipisteen välistä kohtisuoraa etäisyyttä, joka jaetaan paikan FWHM:llä [50]. VRT-algoritmissa vastesuora korvataan kolmiulotteisella, sylinterimäisellä vasteputkella (engl. *tube of response*, TOR), vokseleita approksimoidaan palloina ja kullekin vasteputkelle ja vokselille lasketaan suhde

$$G_{ij} = \frac{V_{ij}}{V_{\text{TOR},i}}, \quad (38)$$

missä $V(i, j)$ on vasteputken i ja vokselin j leikkaustilavuus ja $V_{\text{TOR},i}$ vasteputken tilavuus [51].

Datan korjaukset voidaan suorittaa OMEGAssa joko ennen rekonstruktiota tai sen aikana. Satunnaiskorjauksia varten ohjelmaan täytyy syöttää viivästetyt havainnot sisältävä tiedosto tai niistä muodostettu sinogrammi. Sirontakorjauksiin tarvitaan GATE-simulaatiosta poimittua sirontadataa, joka joko vähennetään suoraan havainnoista tai sinogrammista, käsitellään rekonstruktion aikana taikka sisällytetään systeemimatriisiin lävistäjämatriisina. Rekonstruktion aikana tapahtuvat satunnais- ja sirontakorjaukset toteutetaan ML-menetelmissä sisällyttämällä siron-

tadata vektoreina $\mathbf{q}_{\text{sat.}}$ ja $\mathbf{q}_{\text{sir.}}$ iteraatiokaavaan: esimerkiksi OSEM-algoritmissa

$$c_j^{(k,n)} = \frac{c_j^{(k,n-1)}}{\sum_{i \in S_n} A_{ij}} \sum_{i \in S_n} A_{ij} \frac{m_i}{\sum_j^{N_{\text{vx}}} A_{ij} c_j^{(k,n-1)} + q_{\text{sat.},i}^{(k,n-1)} + q_{\text{sir.},i}^{(k,n-1)}}. \quad (39)$$

Vaimennuskorjaukset voidaan toteuttaa TT-kuvista tai ^{68}Ge -lähteellä suoritetuista mittauksista saadulla datalla. Normalisaatiokertoimet voidaan laskea OMEGAssa erillisen normalisaatiomittausdatan avulla lähteessä [52] esitetyllä tavalla; kertoimista vain sisäiset tehokkuudet ϵ , lohkoprofililit $b^{\text{tr.}}$ ja $b^{\text{ax.}}$ sekä geometriset kertoimet $g^{\text{tr.}}$ ja $g^{\text{ax.}}$ on laskettavissa. Sisäiset tehokkuudet määritetään viuhkasumma-algoritmilla (engl. *fan-sum algorithm*, jossa lasketaan useilla renkaan h kiteeseen i päättyvillä vastesuorilla tehtyjen havaintojen summa m_{hi} ; tällöin

$$\epsilon_{hi} = \frac{\overline{m_{hi}}}{m_{hi}}, \quad (40)$$

missä $\overline{m_{hi}}$ on kaikkien renkaan h viuhkasummien keskiarvo. Lohkoprofilitekijät sisällytetään viuhkasummamenetelmällä laskettuihin sisäisiin tehokkuuksiin, mutta ne voidaan tarvittaessa määrittää myös jakamalla kunkin lohkon tai renkaan sinogrammit kaikkien lohkojen tai renkaiden sinogrammien keskiarvolla ja kääntämällä kuten yhtälössä (40). Vastaavasti aksiaaliset geometriset tekijät saadaan vertaamalla kunkin rengasparin normalisoituja sinogrammeja kaikkien parien keskiarvoon ja transaksiaaliset geometriset tekijät vertaamalla kunkin kideparin havaintoja niiltä vastesuorilta saatuihin havaintoihin, joiden etäisyys s keskeltä FOV:ia on yhtä suuri. OMEGA sisältää myös kaarikorjausfunktion, joka korjaa vierekkäisten, samansuuntaisten vastesuorien väliset kohtisuorat etäisyydet, mutta sitä suositellaan vain Inveon-laitteiden datalle. [10, 48]

3 Menetelmät

Tässä luvussa käydään läpi simulaatioiden toteutuksen ja kuvarekonstruktion vaiheet. Lisäksi simuloitava PET-laite ja fantomit esitellään yksityiskohtaisesti.

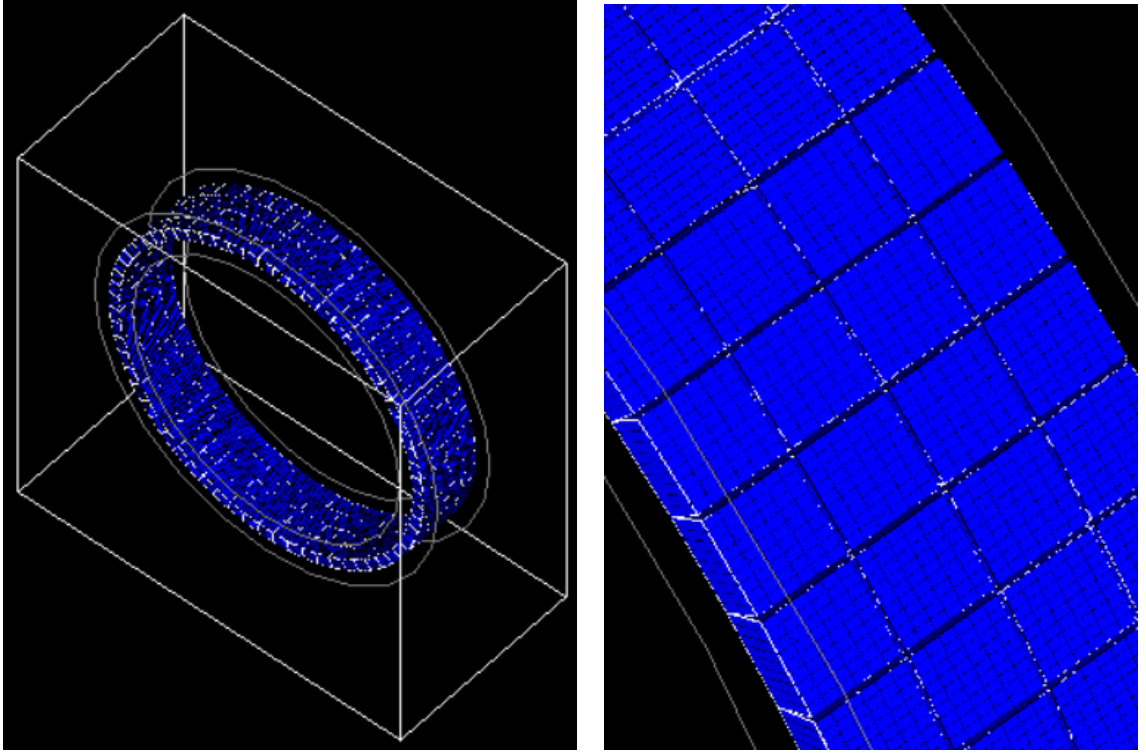
3.1 Simulaatiot

Simuloitavaksi kameraksi valittiin PET/TT-laite GE Discovery-690 (D-690). Koh-teiksi valittiin sylinterifantomi, Derenzo-fantomi, NEMA 2-2001 IQ -kehofantomi ja Hoffman-aivofantomi. Kameran ja fantomien geometriat, materiaalit ja aktiivisuus-jakaumat sekä digitoijan rakenne on selostettu seuraavissa alaluvuissa. Simulaation vaiheet suoritettiin kuvassa 4 esitetystä järjestyksessä. Fysikaaliset prosessit asetet-tiin Geant4:n *empenelope*-listanrakentajalla ja perustuivat nimenomaisesti Monte Carlo -simulaatioita varten suunniteltuun Penelope-malliin [42]. Malli sisältää ku-vaukset positronien kokemille suunnan- ja energianmuutoksille kimmoisten ja kim-mottomien törmäysten sekä jarrutussäteilyn takia. Valosähköisen ilmiön vaikutusa-lat määritetään mallissa interpoloimalla LLNL Evaluated Photon Data Libraryn [53] taulukosta, Comptonin siirron vaikutusala muokatulla Kleinin–Nishinan yh-tälöllä (8) ja Rayleigh’n siirron vaikutusala yhtälöllä (9); siirontakulmat valitaan hyväksyntä-hylkäysmenetelmällä [42]. Simulaatioiden tulokset tallennettiin raakada-tana ASCII- ja ROOT-tiedostoihin, joista havainnot visualisoitiin ROOT-ohjelmalla.

3.1.1 Kamera ja digitoija

D-690:ssä on 24-renkainen PET-kamera, joka sisältää 13 824 LYSO-kidettä, joiden koko on $4,2 \times 6,3 \times 25 \text{ mm}^3$. Yksi ilmaisinyksikkö koostuu 54 (9×6) LYSO-kiteestä, jotka on kytketty nelianodiseen valomonistinputkeen. Kameran aksiaalinen FOV on 157 mm ja transaksiaalinen FOV 70 cm. Laitteen energiakynnys on 425 keV ja koinsidenssi-ikkunan leveys 4,9 ns. [21]

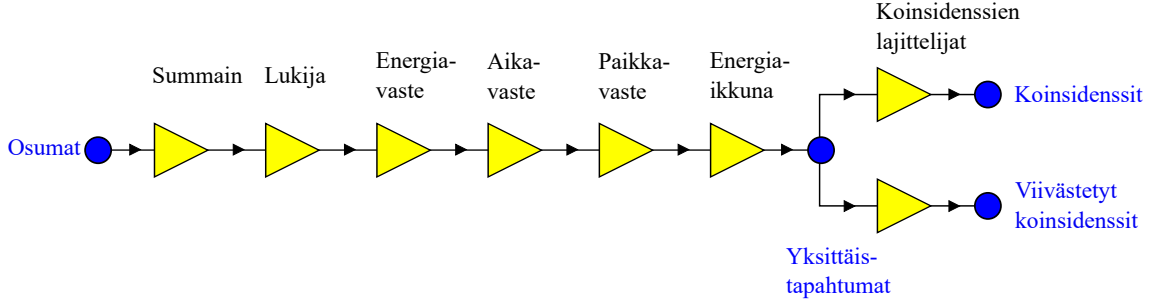
Simuloitu PET-kamera on esitetty kuvassa 6. Koko simulaation sisältävän maail-man mitoiksi valittiin $X = 100 \text{ cm}$, $Y = 100 \text{ cm}$ ja $Z = 50 \text{ cm}$. Laitteen kanturia esit-ti ontto sylinteri, jonka ulkosäde oli $r_{\text{D-690,max}} = 450 \text{ mm}$, sisäsäde $r_{\text{D-690,min}} = 350 \text{ mm}$ ja leveys $z_{\text{D-690}} = 157 \text{ mm}$; kanturin keskiakseli oli maailman Z -akselin suun-tainen. Kamera rakennettiin 64 rengaslohkosta, joiden pituus kameran säteen suun-



Kuva 6. Simuloitu PET-kamera. Lähikuvassa erottuvat yksittäiset rengaslohkot, moduulit ja kiteet.

nassa oli $x_{\text{lohko}} = 25$ mm, kehän suunnassa $y_{\text{lohko}} = 37,8$ mm ja akselin suunnassa $z_{\text{lohko}} = 157$ mm; niiden keskipisteiden etäisyydeksi keskiakselista asetettiin 397,3 mm, jotta vierekkäiset lohkot eivät leikkaisi toisiaan. Kukin lohko koostui neljästä 54 LYSO-kidettä sisältävästä moduulista, joille $x_{\text{mod.}} = x_{\text{lohko}}$ ja $y_{\text{mod.}} = z_{\text{mod.}} = y_{\text{lohko}}$. Kiteiden mitat olivat $x_{\text{kide}} = x_{\text{lohko}}$, $y_{\text{kide}} = 4,2$ mm ja $z_{\text{kide}} = 6,3$ mm. Vierekkäisten moduulien väliin jätettiin 1,9 mm tyhjää tilaa, jotta simuloitu aksiaalinen FOV ja kameran leveys olisivat yhtä suuret.

Simulaatiossa käytetyn digitoijan rakenne on esitetty kuvassa 7. Digitoija koostui summaimesta, lukijasta, energia-, aika- ja paikkavastemoduuleista, energiaikkunasta sekä koinvidenssien ja viivästettyjen koinvidenssien lajittelijoista. Lukija simuloi valonistinputken toimintaa: se kokosi yhdestä kameran moduulista saapuvat signaalit yhdeksi pulssiksi. Vastemoduulien arvot valittiin siten, että ne vastaavat D-690:n käyttämää energia- ja koinvidenssi-ikkunaa sekä kokeellisesti määritettyä erotusky-



Kuva 7. Simuloitu digitoija. Se muuntaa ilmaisimien rekisteröimät osumat yksittäis-tapahtumiksi, joista osa karsitaan ja loput yhdistetään koinisidenssipareiksi.

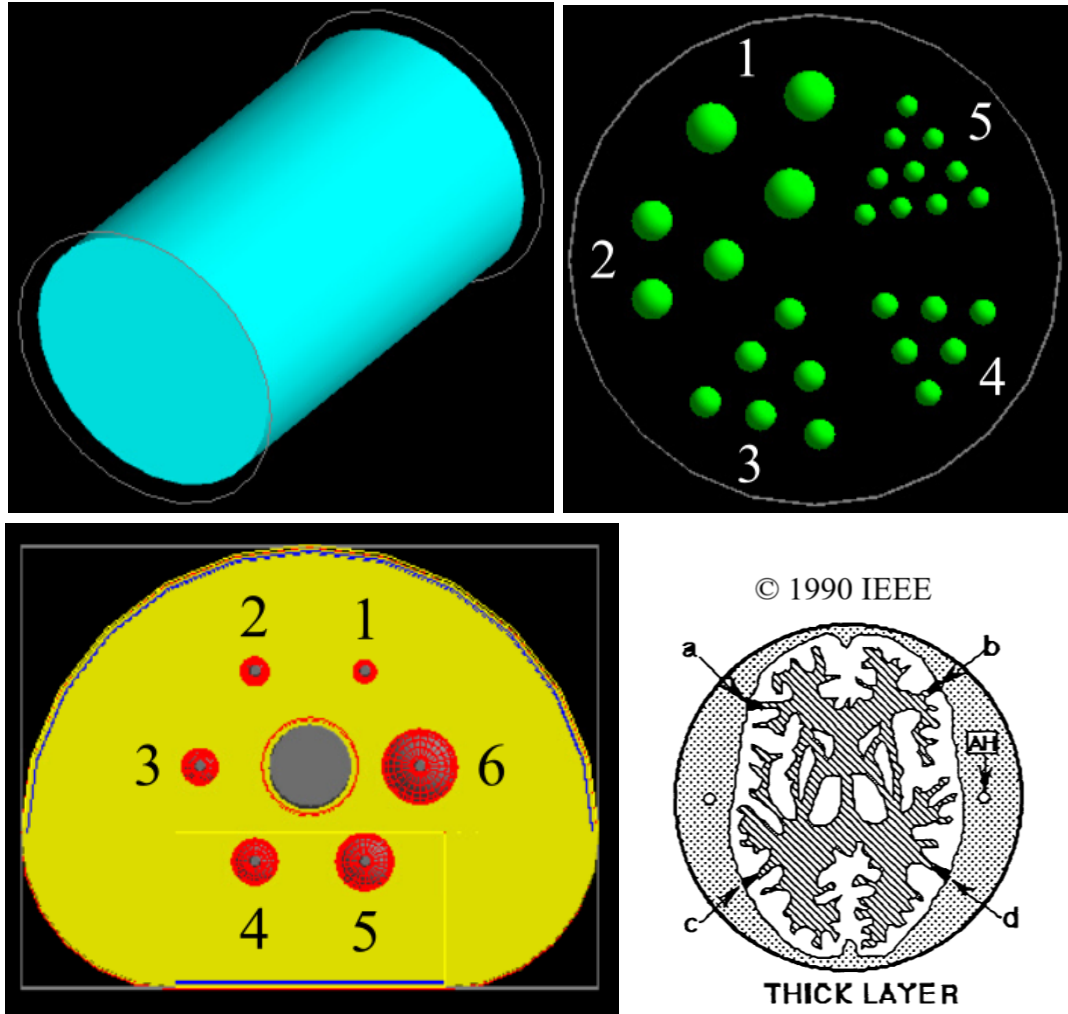
kyä [21]: energian FWHM:ksi asetettiin $R_0 = 12,4 \%$ viitearvolla $E_0 = 511$ keV, ajan FWHM:ksi $R_t = 384,9$ ps sekä paikan FWHM:ksi $R_z = 4,74$ mm ja $R_x = R_y = 4,70$ mm. Energiaikkunaksi asetettiin 425–650 keV ja koinisidenssi-ikkunoiden leveydeksi $\tau = 4,9$ ns. Viivästettyjen koinisidenssien lajittelija määritettiin hyväksymään 100 ns:n viiveellä havaitut satunnaistapahtumat.

3.1.2 Fantomit

Derenzo-, NEMA- ja Hoffman-fantomien geometriat ja aktiivisuusjakaumat perustuivat OpenGATE Collaborationin GitHubissa [54] jakamiin esimerkkifantomeihin. Fantomien geometriat on esitetty kuvassa 8. Ne kaikki sisälsivät ^{18}F :a säteilylähteenä.

Sylinterifantomi oli vedellä täytetty ja muovikuorinen. Sen sisempi pituus oli $z_{\text{syl.}} = 160$ mm ja säde $r_{\text{syl.}} = 50$ mm; muovikuoren paksuus oli 5 mm. Sen aktiivisuus oli $A_{\text{syl.}} = 4$ MBq.

Derenzo-fantomi oli muovisylinteri ($z_{\text{Dzo}} = 10$ mm ja $r_{\text{Dzo}} = 25$ mm), joka sisälsi 28 vedellä täytettyä palloa. Pallot jaettiin viiteen ryhmään siten, että ryhmän i sisällä jokaisen pallon säde oli $r_{\text{Dzo},i}$ ja kahden pallon keskipisteiden välinen etäisyys oli noin $4r_{\text{Dzo},i}$. Pallojen säteet olivat $r_{\text{Dzo},1} = 2,5$ mm, $r_{\text{Dzo},2} = 2,0$ mm, $r_{\text{Dzo},3} = 1,5$ mm, $r_{\text{Dzo},4} = 1,25$ mm ja $r_{\text{Dzo},5} = 1,0$ mm. Niiden vastaaviksi aktiivisuuksiksi asetettiin $A_{\text{Dzo},1} = 250$ kBq, $A_{\text{Dzo},2} = 200$ kBq, $A_{\text{Dzo},3} = 150$ kBq, $A_{\text{Dzo},4} = 120$ kBq



Kuva 8. Simuloidut fantomit: (ylhäältä vasemmalta alkaen myötäpäivään) sylinteri-fantomi, Derenzo, Hoffman ja NEMA. GATE V9.3 ei voi näyttää vokseloituja fantomeita, joten Hoffmanin viitekuva on poimittu artikkelista [24]; rakenteita a, b, c ja d sekä AH ei ole simuloidussa fantomissa.

ja $A_{Dzo,5} = 100$ kBq; fantomin kokonaisaktiivisuus oli siten $A_{Dzo} = 3,97$ MBq.

NEMA-kehofantomin äärimitat olivat $x_{NEMA,max} = 30$ cm, $y_{NEMA,max} = 23$ cm ja $z_{NEMA,max} = 22$ cm. Se rakennettiin puolisyylinteristä, jonka säde oli $r_{NEMA,yl} = 14,7$ cm; kahdesta neljännessylinteristä, joiden säde oli $r_{NEMA,al} = 7,7$ cm; ja suorakulmaisesta särmiöstä, jonka mitat olivat $x_{NEMA,al} = 14,0$ cm ja $y_{NEMA,al} = 7,7$ cm. Kaikkien osien pituus oli $z_{NEMA,min} = 21,4$ cm, ja ne oli täytetty vedellä. Fantomia ympäröi 3 mm paksu muovikuori. Se sisälsi kuusi ilmatäytteistä palloa eli ”leesiota”, joiden säteet olivat $r_{NEMA,1} = 5$ mm, $r_{NEMA,2} = 6,5$ mm, $r_{NEMA,3} = 8,5$ mm,

$r_{\text{NEMA},4} = 11$ mm, $r_{\text{NEMA},5} = 14$ mm ja $r_{\text{NEMA},6} = 18,5$ mm ja joiden ympärillä oli 1 mm paksu muovikuori. Kaikkien leesioiden keskipisteiden etäisyys fantomin keskiakselista oli 5,72 cm ja sijainti z -suunnassa 3,5 cm; lisäksi jokaisella leesiolla oli täyttöputki, jonka säde oli 3,5 mm ja pituus 5,05–6,4 cm. Keskellä fantomia oli ontto, ilmalla täytetty sylinteri eli ”keuhko”, jonka mitat olivat $r_{\text{NEMA},k.} = 25$ mm ja pituus $z_{\text{NEMA},k.} = 21,4$ cm. Julkaisun [23] pohjalta veden aktiivisuuspitoisuudeksi valittiin 5,3 kBq/mL ($A_{\text{NEMA},bg.} = 59$ MBq), ja neljän pienemmän, ”kuuman” leesion aktiivisuuspitoisuudet asetettiin nelinkertaisiksi taustaan verrattuna: $A_{\text{NEMA},1} = 11$ kBq, $A_{\text{NEMA},2} = 24$ kBq, $A_{\text{NEMA},3} = 55$ kBq ja $A_{\text{NEMA},4} = 118$ kBq; yhteensä $A_{\text{NEMA}} = 59,208$ MBq. Isoimmat leesiot, täyttöputket ja keuhko olivat ”kylmiä” eli epäaktiivisia.

Hoffman-aivofantomi rakennettiin vokseloidun fantomin ja lähteen sisältävien Interfile-tiedostojen avulla. Se muistutti rakenteeltaan ihmisaivoja ja esitti aktiivisuuden jakautumista harmaan ja valkean aineen välillä siten, että harmaan aineen aktiivisuuspitoisuus oli noin viisinkertainen valkeaan verrattuna. Simuloidun Hoffmanin ”harmaan aineen” ja ”valkean aineen” aktiivisuudet asetettiin siten, että sen kokonaisaktiivisuus oli $A_{\text{Hoff.}} = 15,6$ MBq.

Virtuaalikoneen käytettävissä olevan rajallisen muistin takia yhden simulaatioajon aikana pystyttiin mallintamaan vain vähän yli neljä miljoonaa β^+ -hajoamista. Siksi kullakin fantomilla ajettiin monta simulaatiota; kullekin fantomille valittiin yksi PRNG-siemen, jota käytettiin jokaisessa simulaatiossa, ja uuden simulaation aloitusajaksi asetettiin aina edellisen päättymisaika. Näin varmistettiin, että jokaisen yksittäisen simulaation alkutila oli sama kuin edellisen lopputila. Sylinterifantomille ja Derenzolle valittiin yhden simulaation kuvantamisajaksi 1 s, NEMAlle 0,068 s ja Hoffmanille 0,25 s. Simulaatioita suoritettiin niin monta kertaa, että kullekin fantomille saatiin yhteensä noin kaksi miljoonaa havaintoa: sylinterillä kokonaiskuvantamisaika oli 25 s, Derenzolla 10 s, NEMAlla 3,128 s ja Hoffmanilla 8,25 s.

3.2 Kuvarekonstruktiot

PET-kuvien rekonstruoimiseen käytettiin OMEGAN implementaatiota 2, jota varten OMEGAN tueksi asennettiin OpenCL SDK ja ArrayFire 3.9. Kuvat muodostettiin `gate_main.m`-tiedoston sisältämällä komentosarjalla. Rekonstruktiossa käytettiin kaikkia ASCII-tiedostoilta tallennettuja koinsidensseja. Rekonstruktioalgoritmiksi valittiin OSEM: rekonstruktiot toteutettiin neljässä iteraatiossa ja havainnot jaettiin kahdeksaan osajoukkoon. Projektoriksi valittiin VRT, jota suositellaan parhaan kuvanlaadun takaamiseksi [48].

Jokaisesta fantomista tehtiin neljä kuvaa, joille asetettiin eri resoluutiot säätämällä transaksiaalista FOV:ä tai kuvakokoa. Oletusarvoinen FOV ja kuvakoko D-690:lle oli $700 \times 700 \times 157 \text{ mm}^3$ ja $128 \times 128 \times 47 \text{ vx}$. Jokaiselle fantomille muodostettiin kaksi kuvaa pienemmällä FOV:llä, joka kattoi juuri ja juuri koko fantomin x - ja y -suunnissa: sylinterin pienempi FOV oli $110 \times 110 \text{ mm}^2$, Derenzon $50 \times 50 \text{ mm}^2$, NEMAn $300 \times 300 \text{ mm}^2$ ja Hoffmanin $270 \times 270 \text{ mm}^2$. Kaksi kuvaa muodostettiin suurimmalla koolla, mitä OMEGA pystyi rekonstruoimaan annetusta simulaatio-datasta: $225 \times 225 \times 47 \text{ vx}$. Sinogrammien lukumäärä kutakin kuvaa kohti laskettiin OMEGAssa valmiiksi annetulla komennolla, jota varten vinojen vastesuorien kompression määrittäväksi *span*-tekijäksi asetettiin 3, radiaalisten etäisyyksien s lukumääräksi 381 FOV:llä $700 \times 700 \text{ mm}^2$ ja kulmien ϕ lukumääräksi 288; etäisyyksien s lukumääräksi muille FOV:ille asetettiin `ndist_max`-funktion antamat suositukset (suuruusjärjestyksessä: 23, 53, 131 ja 147).

Kuville tehtiin satunnais-, sironta- ja normalisaatiokorjaukset rekonstruktion aikana. Satunnaiskorjauksissa käytettiin simulaatioista saatuja viivästettyjä koinsidensseja. Sirontakorjauksia varten kullakin fantomilla tehtiin toiset simulaatiot, joista tallennetuista sironneista koinsidensseista muodostettiin sirontasinogrammit. GATE:n tulostamiin ASCII-tiedostoihin on kirjoitettu jokaisen fotonin siroamisten lukumäärät, joten datasta oli mahdollista eritellä Compton- tai Rayleigh-sironneet

sekä fantomissa tai ilmaisimissa sironneet koinssidenssit; tässä sironneiksi koinssidensseiksi lasketaan vain fantomissa Compton-sironneet havainnot, koska ilmaisimissa tapahtuva sironta ei merkittävästi heikennä kuvanlaatua [10] ja Rayleigh'n sironta muuttaa fotonin kulkusuuntaa hyvin vähän [33]. Normalisaatiokorjauksia varten kuvannettiin sylinterifantomia, jonka sisäsäde oli 100 mm, sisustan pituus 160 mm ja kuoren paksuus 5 mm, ja sillä kerättiin noin kolme miljoonaa havaintoa. Näistä poimittiin todelliset koinssidenssit, joilla laskettiin normalisaatiomatriisin komponentit ϵ , $b^{\text{tr.}}$, $b^{\text{ax.}}$, $g^{\text{tr.}}$ ja $g^{\text{ax.}}$; tekijöitä $b^{\text{tr.}}$ ja $b^{\text{ax.}}$ ei määritetty erikseen vaan ne saatiin laskemalla tehokkuuksien ϵ keskiarvot kullekin kiteen sijainnille lohossa.

4 Tulokset

Tässä luvussa esitellään simulaatioiden tulokset ja rekonstruoidut kuvat. GATE tulosti simuloidut havainnot ASCII- ja ROOT-tiedostoille, joista edelliset syötettiin OMEGAan rekonstruktiota varten ja jälkimmäisiä käytettiin simuloitujen aktiivisuusjakaumien näyttämiseen. OMEGAssa rekonstruoitujen kuvien vokseliarvot muunnettiin havaintojen lukumäärästä vokselia kohti (lkm/vx) aktiivisuustiheysarvoiksi (Bq/mL).

4.1 Simuloidut havainnot

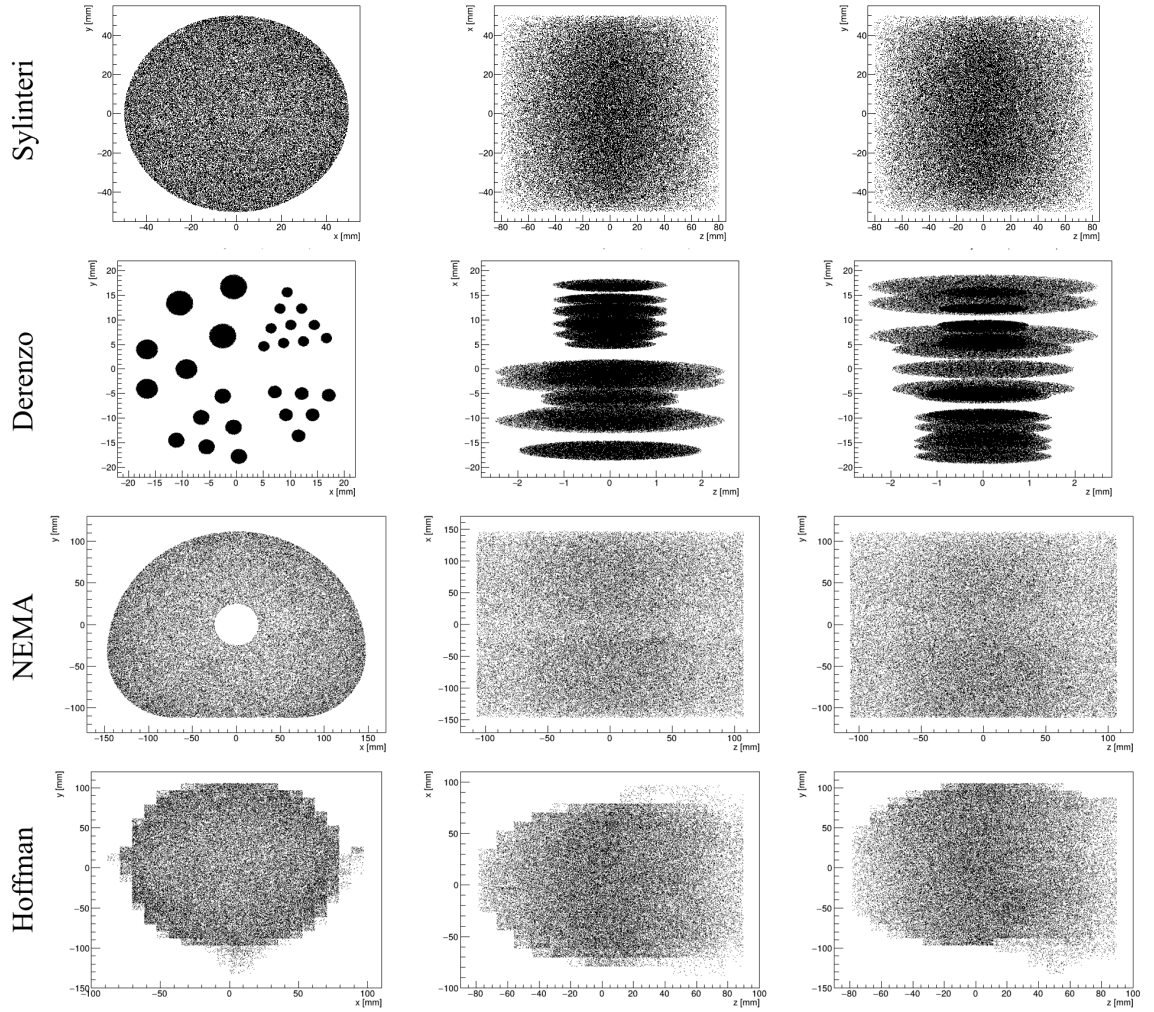
GATE-simulaatioissa tuotettiin sylinterifantomilla yhteensä 99 777 058, Derenzolla 39 669 094, NEMAlla 184 149 301 ja Hoffmanilla 129 154 285 β_+ -hajoamista. Sylinterillä havaittiin yhteensä 2 035 386, Derenzolla 2 198 600, NEMAlla 2 037 718 ja Hoffmanilla 2 042 948 koinssidenssiparia. Todellisten, satunnaisten ja sironneiden koinssidenssien suhteelliset osuudet on esitetty taulukossa II, jossa sironneet koinssidenssit sisältävät sekä Compton- että Rayleigh-sironneet parit riippumatta siitä, missä sironta tapahtui. Jokaisella fantomilla sironneita koinssidensseja saatiin

Taulukko II. Simulaatioissa havaittujen erilaatuisten koinvidenssiparien osuudet. Jokainen koinvidenssi on luokiteltu joko todelliseksi, satunnaiseksi tai sironneeksi. Monta kertaa sironnut pari voi kuulua useampaan kuin yhteen sirontaluokkaan.

Koinvidenssit	Sylinteri	Derenzo	NEMA	Hoffman
Todelliset [%]	15,10	14,52	5,64	11,15
Satunnaiset [%]	2,26	0,57	40,96	9,95
Sironneet [%]	82,64	84,91	53,40	78,90
Compton-sironneet fantomissa [%]	3,14	5,04	20,96	19,51
Compton-sironneet ilmaisimissa [%]	81,02	82,91	49,04	74,54
Rayleigh-sironneet fantomissa [%]	0,04	0,07	0,24	0,31
Rayleigh-sironneet ilmaisimissa [%]	12,17	12,45	7,45	11,43

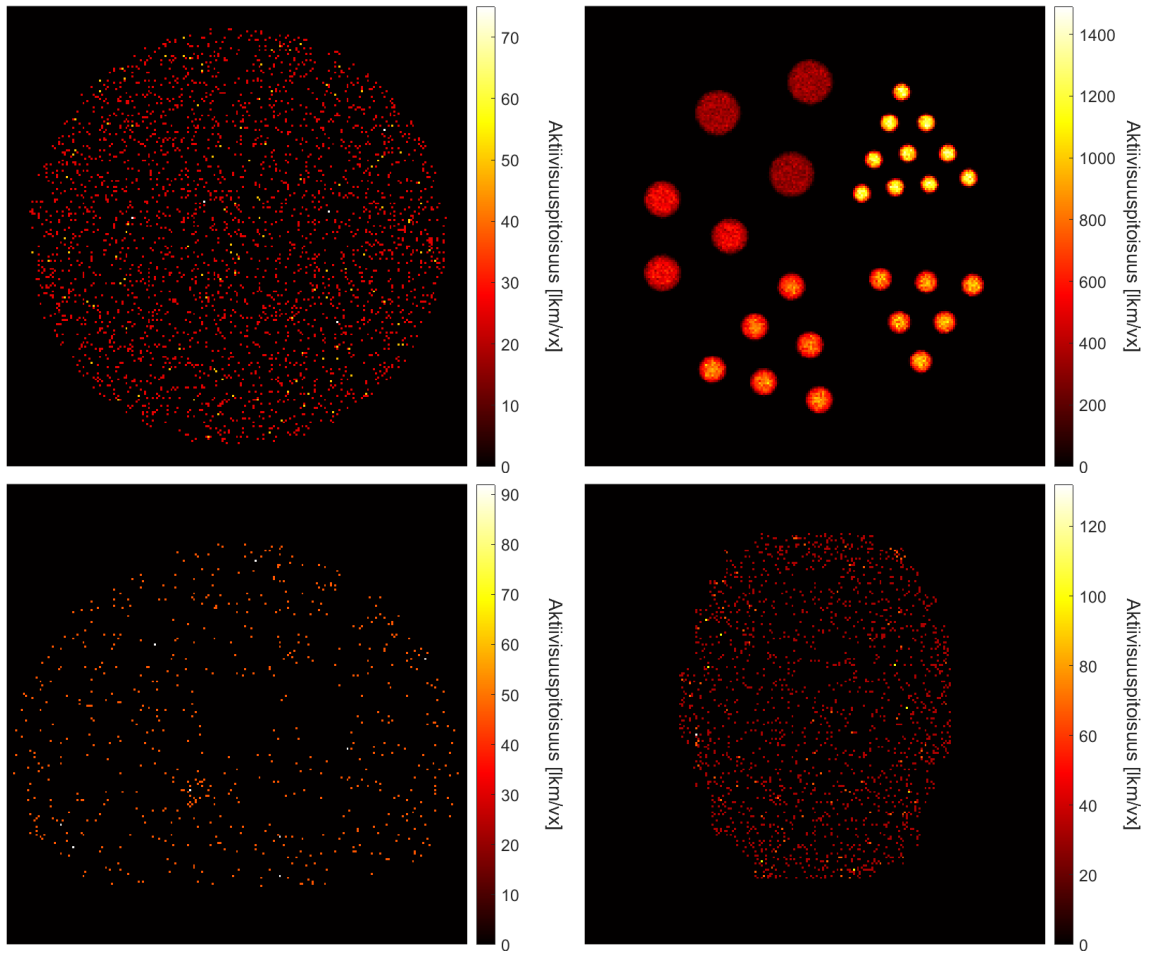
enemmän kuin todellisia ja satunnaisia yhteensä, mutta niistä lähes kaikki sirosivat kamerassa. Rayleigh'n sirontaa tapahtui paljon vähemmän kuin Comptonin sirontaa. Sylinterillä ja Derenzolla satunnaisia koinvidensseja saatiin hyvin vähän, mutta Hoffmanilla niitä havaittiin lähes yhtä paljon ja NEMAlla paljon enemmän kuin todellisia koinvidensseja. Tämä oli odotettavissa kaavan (10) perusteella, sillä Hoffmanilla ja NEMAlla oli paljon toisia suurempi kokonaisaktiivisuus. NEMAssa ja Hoffmanissa tapahtui odotetusti enemmän sirontaa kuin pienemmissä sylinterissä ja Derenzossa, mutta Derenzossa tapahtui paljon enemmän sirontaa kuin sylinterissä mahdollisesti siksi, että sen tilavuudesta suurin osa on muovia, jonka tiheys on simulaatioissa 18 % suurempi kuin veden [54].

Kuvassa 9 on esitetty ROOT-sovelluksessa tehdyt 2D-hajontakuviot, joissa näkyvät kaikkia paitsi viivästettyjä koinvidenssipareja vastaavat hajoamispisteet jokaisessa fantomissa. Vertaamalla hajontakuvioita kuvaan 8 nähdään, että havaitut aktiivisuusjakaumat noudattavat hyvin fantomien geometriaa. Sylinterin havaittu jakauma on xy -koordinaatistossa tasainen, mutta läheltä fantomin päitä saatiin vähemmän havaintoja kuin sen keskeltä; lähellä PET-kameran reunoja syntyvien an-



Kuva 9. Fantomien GATE-simulaatioissa havaittujen aktiivisuuskajaukien hajontakuviot. Kussakin Root-kuvaajassa on esitetty xy -, xz - tai yz -koordinaatistossa kaikki ne hajoamispisteet, joista alkunsa saaneet fotonit on havaittu ja rekisteröity osana koincidenssiparia.

nihilaatiofotonien todennäköisyys osua kameraan ja muodostaa koincidenssipari on luonnollisesti pienempi kuin FOV:n keskellä syntyvien. Derenzo oli niin pieni kameraan verrattuna, että sen havaitut aktiivisuuskajaukset ovat täysin tasaiset. NEMAn jakaumista erottuvat hyvin tausta-aktiivisuus ja keuhko, jossa ei tapahtunut hajoamisia ollenkaan, mutta leesioita ei näy kuvaajissa. Toisaalta leesiöt olivat melko pieniä NEMAn kokonaistilavuuteen nähden; toisaalta NEMasta saatiin hyvin paljon satunnaisia ja sironneita koincidensseja, jotka voivat peittää leesiosta tulleet todelliset havainnot. Hoffmanissa satunnaisten ja fantomissa sironneiden koincidenssien



Kuva 10. Fantomien OMEGAssa muodostetut lähdekuvat. Kuvien pisteet esittävät kaikkia havaittuja lähteitä paitsi niitä, joista on saatu satunnaisia tai fantomissa Compton-sironneita koincidenssipareja.

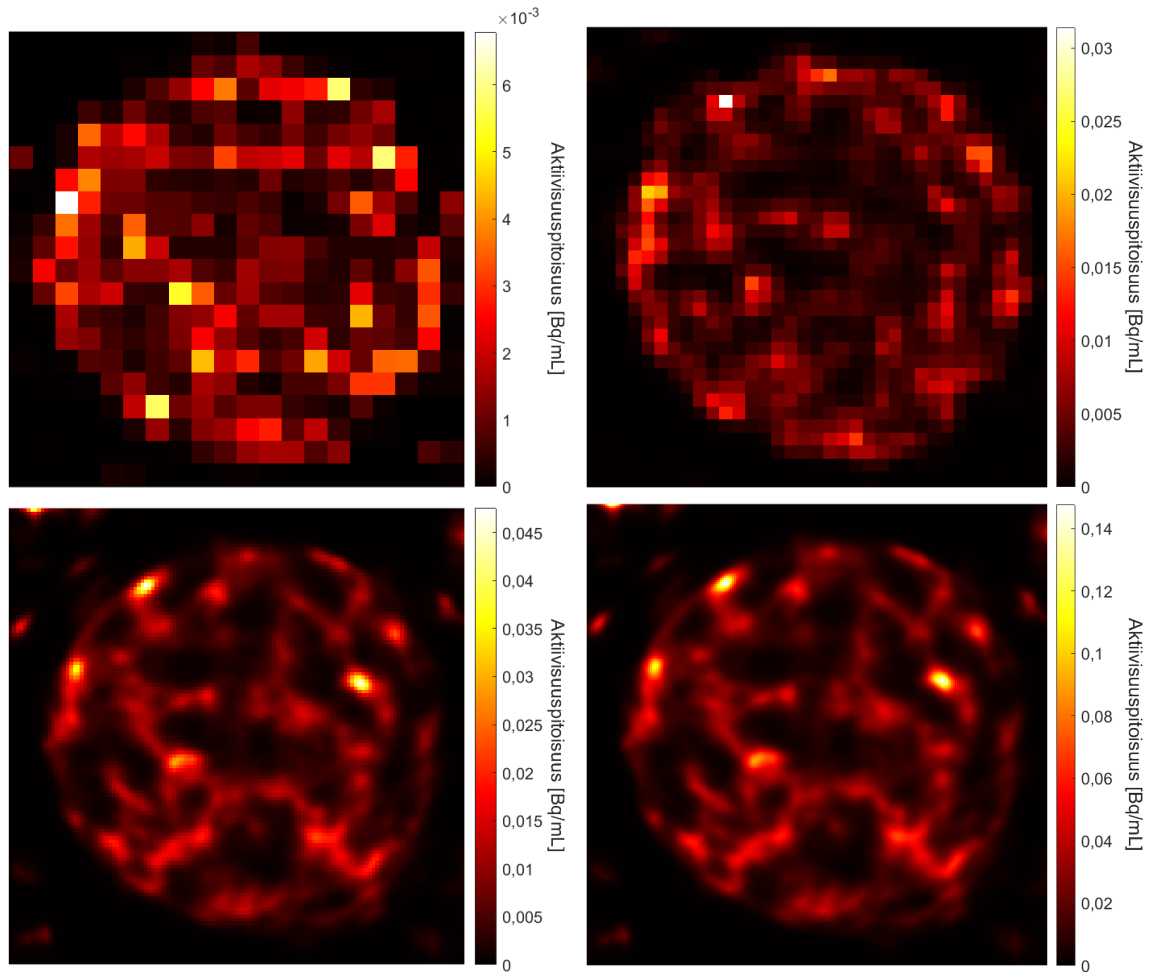
osuudet eivät ole yhtä suuria kuin NEMAssa, mutta ne voivat osaltaan selittää harmaan ja valkean aineen jakaumien huonon erottumisen kuvaajissa. Hajontakuviot näyttävät kuitenkin noudattavan suurin piirtein ihmisaivojen muotoja.

Hajoamispisteiden pohjalta muodostettiin myös OMEGAssa lähdekuvat, jotka on esitetty kuvassa 10. Ne ovat leikekuvia, jotka sisältävät kaikkien todellisten, Rayleigh-sironneiden tai ilmaisimissa Compton-sironneiden koincidenssiparien lähteet pienessä osassa fantomia. Kuvien koko ja FOV on valittu siten, että erotuskyky on mahdollisimman hyvä: sylinterillä FOV:n leveys on 110 mm, Derenzolla 50 mm, NEMAlla 300 mm ja Hoffmanilla 270 mm; kaikilla fantomeilla kuvakoko on 225×225 px ja leikkeen paksuus 3,3 mm. NEMAn kuvaksi valittiin leesi sisältävä

leike, kun taas muiden fantomien kuviksi valittiin keskimmäiset leikkeet. Sylinterin ja Derenzon lähdekuvat ovat odotusten mukaiset: edellisessä havaittu aktiivisuusjakauma on samanlainen kaikkialla, kun taas jälkimmäisessä lähdepisteet ovat rajoittuneet palloihin niille asetettujen aktiivisuuksien mukaisesti. NEMAn lähdekuvassa näkyy melko vähän hajoamispisteitä suhteessa fantomin kokoon, koska niin iso osa koinsidensseista oli satunnaisia tai Compton-sironneita. Kuitenkin kuuma leesio 4, kylmät leesiot ja keuhko erottuvat melko hyvin. Hoffmanin lähdekuvassa fantomin reunat näkyvät hieman sisäosia aktiivisempina. Sisäosat eivät näytä tasaisesti aktiivisilta, mikä johtuu fantomin rakenteesta mutta myös havaintojen melko pienestä lukumäärästä.

4.2 Rekonstruoidut kuvat

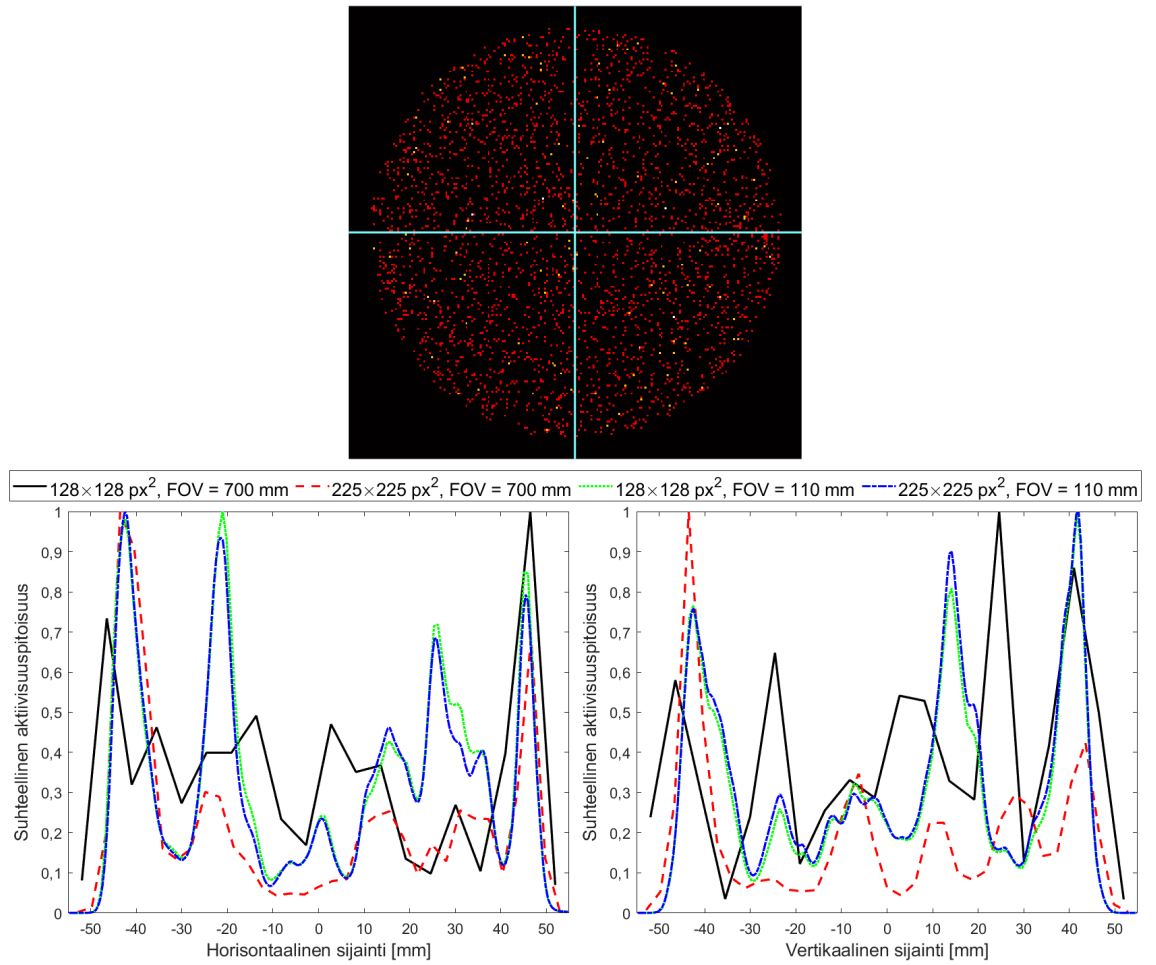
Sylinterifantomien rekonstruktiot on esitetty kuvassa 11. Kaikille rekonstruktioille piirrettiin viivaprofilit, jotka näyttävät suhteellisen aktiivisuustiheyden vaihtelun fantomin keskeltä halkaisevilla horisontaalisella (x) ja vertikaalisella (y) suoralla; ne on näytetty kuvassa 12. Sylinterin rekonstruoitu aktiivisuusjakauma ei ole yhtä tasainen kuin simuloitu, vaan fantomin reunat ovat selvästi aktiivisemmat kuin keskusta. Toisaalta aktiivisuus on hyvin rajautunut sylinterin sisälle, kun taas kuori ja fantomin ulkopuolinen FOV ovat joitakin kohinaisia pisteitä lukuun ottamatta epäaktiivisia. Eri suuntien viivaprofilien aktiivisuusarvot ovat jakautuneet hyvin samanlaisesti. Pienemmällä kuvakoolla ja suuremmalla FOV:llä aktiivisuustiheyden variaatiokertoimeksi (keskihajonta/keskiarvo) saatiin x -suunnassa 60,2 % ja y -suunnassa 61,7 %; pienemmällä FOV:llä saatiin vastaavasti 69,0 % ja 71,0 %. Suuremmalla kuvakoolla ja pienemmällä FOV:llä aktiivisuustiheyden variaatiokerroin on molemmissa suunnissa 69,5 %. Suuremmalla kuvakoolla ja FOV:llä aktiivisuusarvot ovat selvästi hajanaisemmat: x -suunnan variaatiokerroin on 99,8 % ja y -suunnan 96,5 %. Pienemmällä kuvakoolla ja suuremmalla FOV:llä havaitaan vähän aktiivisuutta



Kuva 11. Sylinterifantomin rekonstruoidut ja korjatut kuvat eri FOV:illä (ylh. 700 mm, alh. 110 mm) ja kuvako'oilla (vas. 128×128 px, oik. 225×225 px).

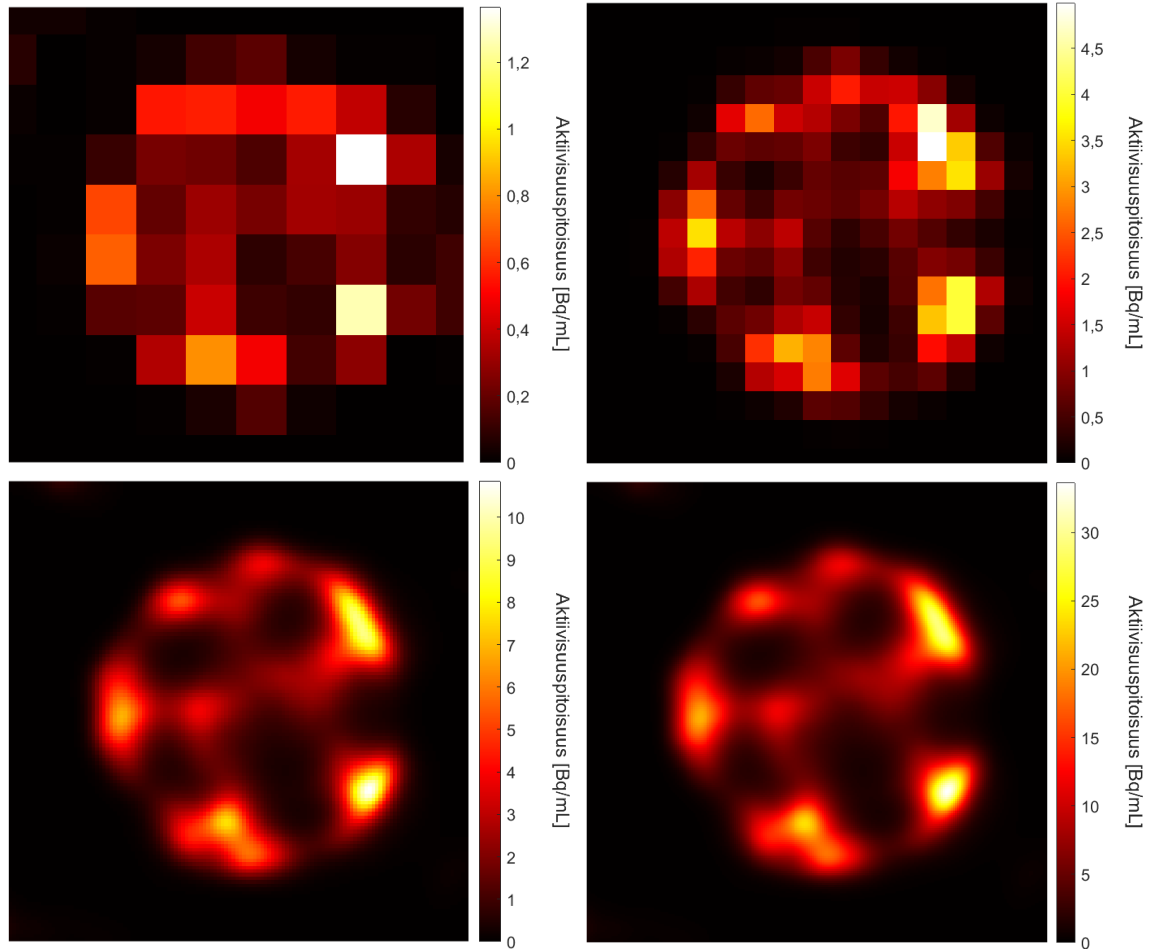
myös kuorella luultavimmin huonon resoluution takia.

Kuvassa 13 esitetyt Derenzon rekonstruktiot näyttävät, miten pieniä yksityiskohtia D-690:llä simuloituista kuvista voi erottaa riippuen kuvakoosta ja FOV:stä. Yhden pikselin leveys on suurimmillaan 5,47 mm, mikä riittää kattamaan minkä tahansa Derenzon palloista, ja pienimmillään 0,22 mm eli noin viidennes ryhmän 5 pallojen säteestä. Suuremmalla FOV:llä yksittäiset pallot eivät näy kuvissa, kun taas pienemmällä FOV:llä osa ryhmien 1 ja 2 palloista erottuu juuri ja juuri ympäristöstään. Eri palloryhmät on mahdollista tunnistaa jokaisessa kuvassa, joskin ryhmät 1 ja 5 näyttävät päällekkäisiltä pienemmällä kuvakoolla. Pienemmällä FOV:llä kuvakoon kasvattaminen ei merkittävästi paranna kuvanlaatua.



Kuva 12. Sylinterifantomien viivaprofiilit. Aktiivisuuspitoisuusarvot on skaalattu vertailukelpoisiksi jakamalla profiilin huippuarvolla.

NEMAn rekonstruktiot on esitetty kuvassa 14. Ne ovat selvästi huonolaatuisempia kuin muiden fantomien, koska koinssidenssihavaintoja saatiin vähän fantomin kokoon nähden ja niistä suurin osa oli joko satunnaisia tai Compton-sironneita. Kaikissa kuvissa yhtä (128×128 px, 700 mm) lukuun ottamatta kirkkaimmat pikselit ovat leesioiden tai koko fantomin ulkopuolella; kuvien väriasteikot on rajattu siten, että niiden huippuarvo löytyy leesiosta 4, jolloin myös tausta-aktiivisuus näkyy paremmin. Pienemmällä FOV:llä tai suuremmalla kuvakolla myös leesio 3 voidaan tunnistaa kuvasta. NEMAlle piirrettiin leesioiden läpi viivaprofiilit, jotka on esitetty kuvassa 15. Kaikilla kuvako'oilla ja FOV:illä leesio 4 näkyy selvänä piikkinä noin -25 mm kohdalta, mutta muut eivät erotu kohinasta.

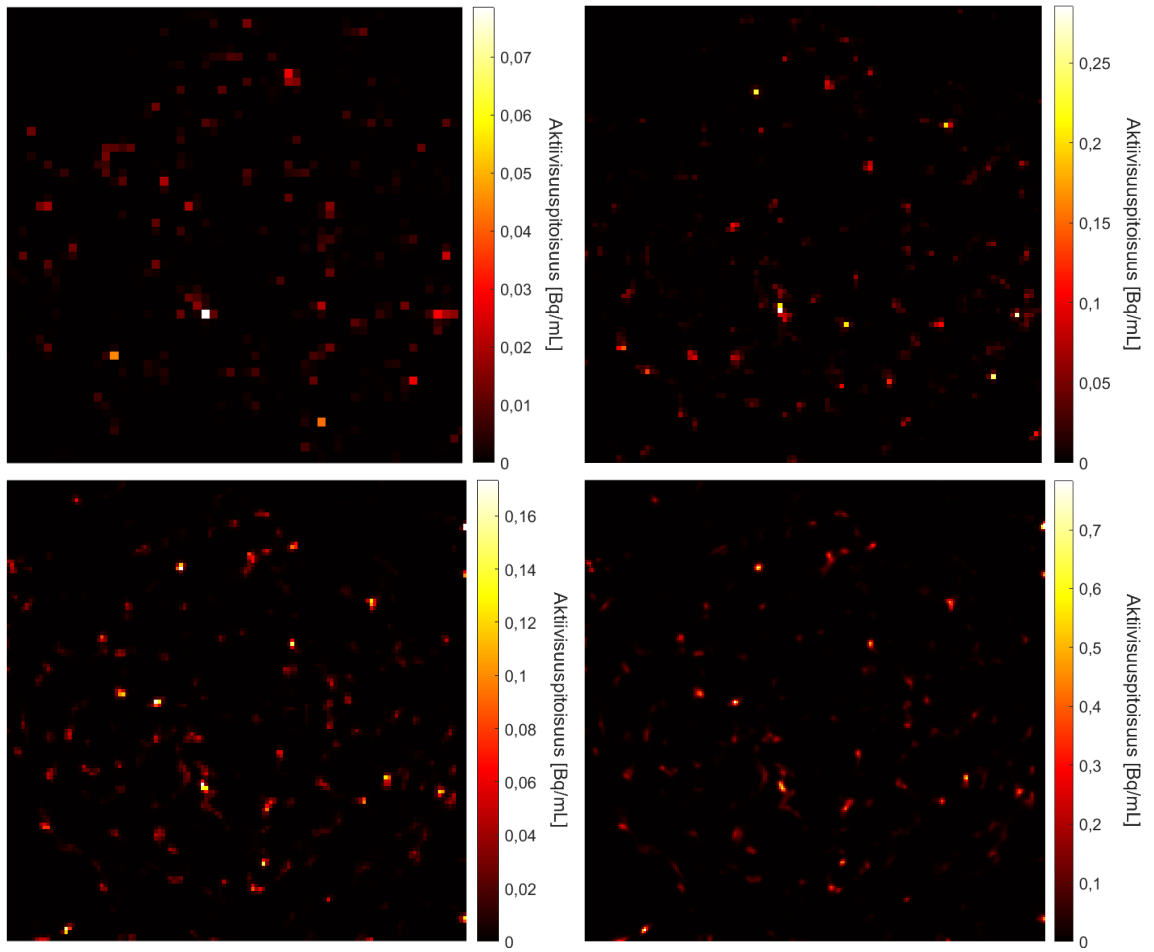


Kuva 13. Derenzo-fantomin rekonstruoidut ja korjatut kuvat eri FOV:illä (ylh. 700 mm, alh. 50 mm) ja kuvako'oilla (vas. 128×128 px, oik. 225×225 px).

Hoffmanin rekonstruktiot on esitetty kuvassa 16. Lähdekuvaan 10 verrattuna reunojen harmaa aine on vielä paljon kirkkaampi ja selvemmin erottuva kuin valkea aine. Myös sisäosien harmaan aineen jakauma erottuu mutta huonommin kuin reunojen. Hoffmanin ulkopuolella on joitakin kohinaisia pikseleitä, jotka heikentävät kuvanlaatua.

5 Pohdinta

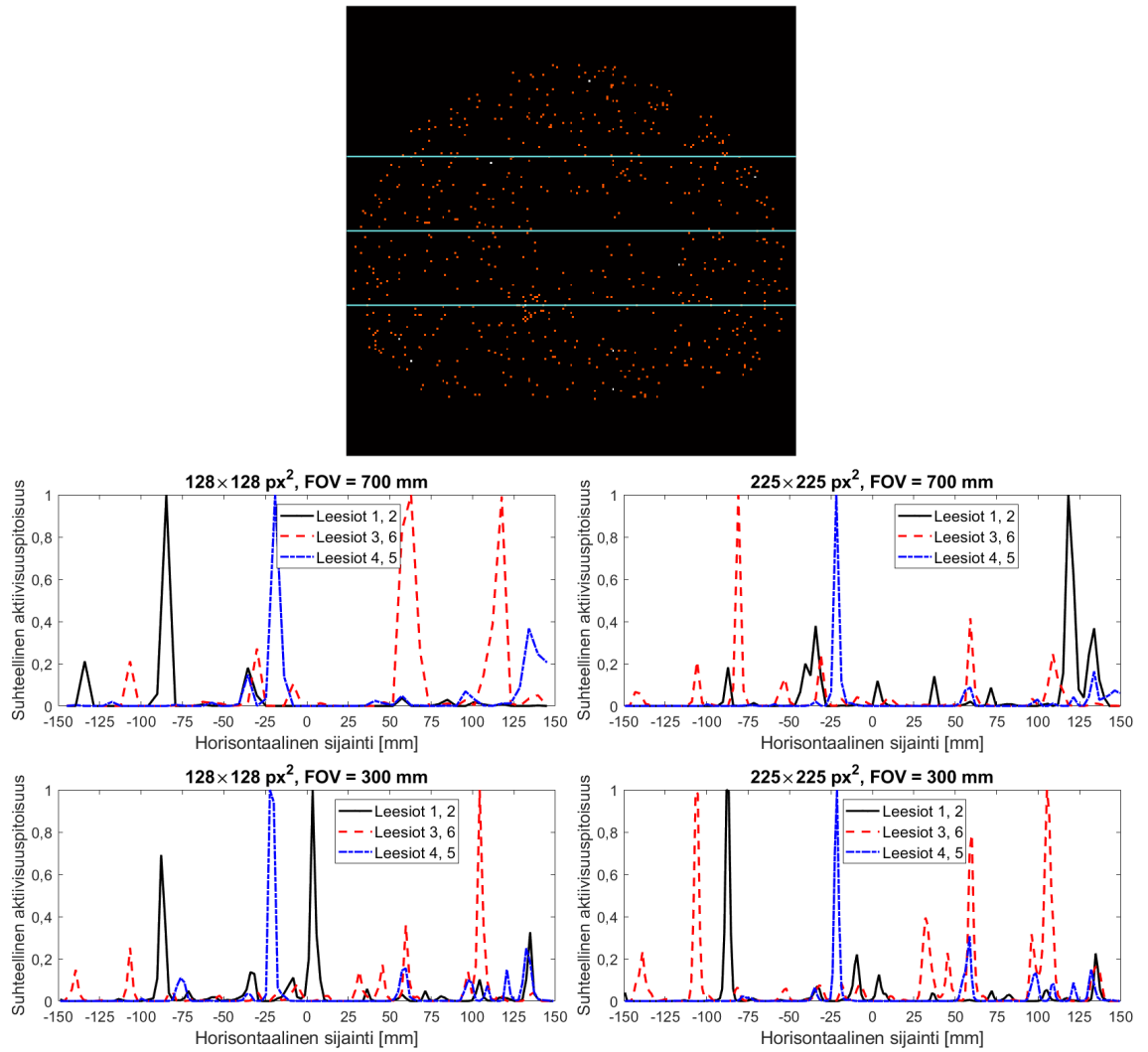
OMEGA todettiin monilta osin käyttäjäystävälliseksi ohjelmaksi. GATE-datan rekonstruktioon tarkoitettussa `gate_main`-koodissa suurin osa muuttujista on nimetty



Kuva 14. NEMA-fantomien rekonstruoidut ja korjatut kuvat eri FOV:illä (ylh. 700 mm, alh. 300 mm) ja kuvako'illa (vas. 128×128 px, oik. 225×225 px).

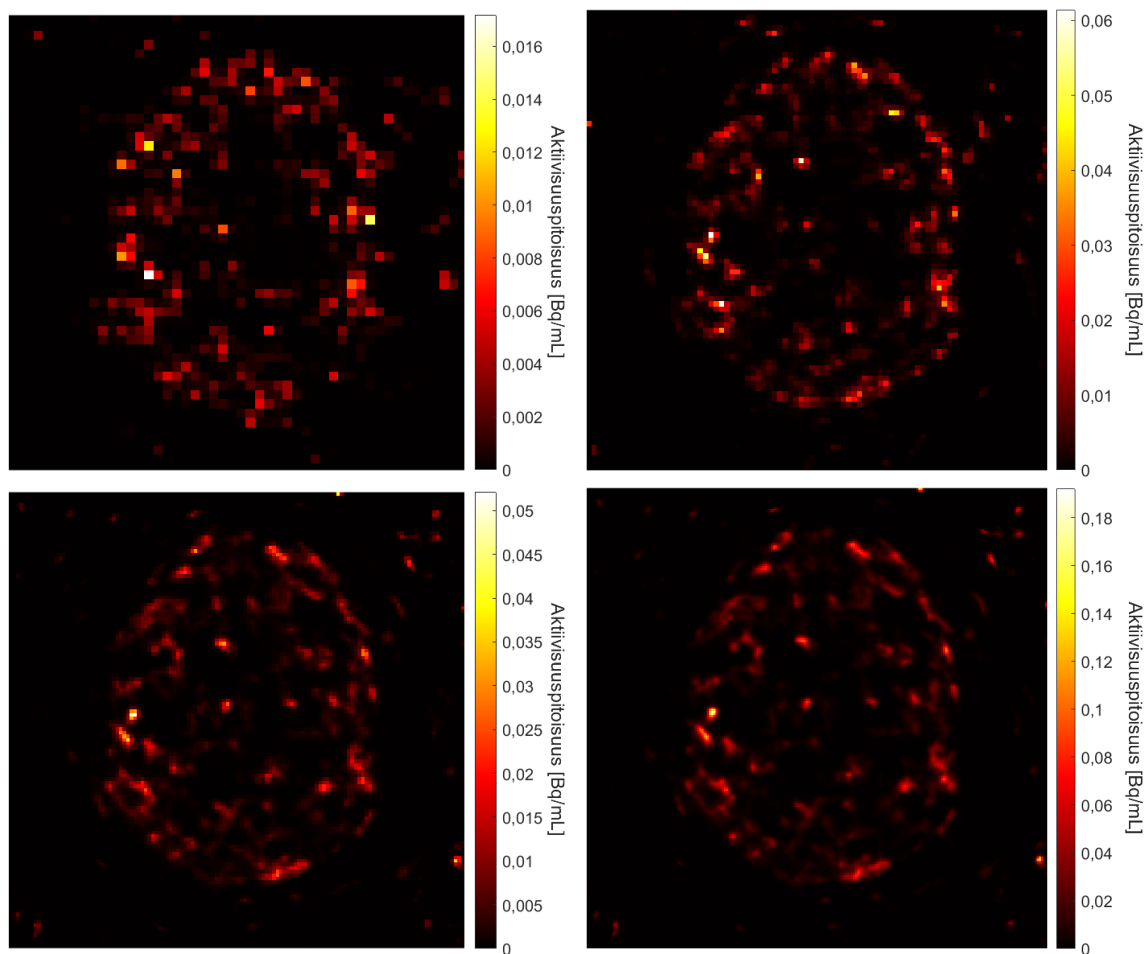
ja selitetty hyvin sekä itse tiedostossa että Githubin OMEGA-wikissä [48]. Wikissä on myös annettu hyvin suosituksia sille, mitä funktioita ja parametriarvoja koodia ajettaessa kannattaa käyttää. Tästä huolimatta parametrien säätäminen oikein oli välillä hankalaa, päätiedoston ulkopuolella kommentointi on vaillinaista ja muutujat välillä epäselvästi nimettyjä, ja OpenCL:ssä tapahtuvia virheitä oli hyvin vaikea selvittää ja korjata tuntematta C++-kieltä.

Rekonstruktioon suositellaan implementaatiota 2 sen nopeuden ja tuettujen metodien laajan valikoiman takia [48], vaikka se tarvitseekin tuekseen OpenCL- ja ArrayFire-kirjastot. Se todettiin toimivaksi ja hyväksi käyttää jatkossakin etenkin, jos halutaan kokeilla muita rekonstruktio menetelmiä kuin OSEM. VRT-projektorilla



Kuva 15. NEMA-fantomin viivaprofiilit. Suorat on valittu siten, että jokainen suora lävistää kaksi palloa.

suositellaan sen laadun takia [48], vaikka se on hitaampi kuin ODRT tai paranneltu Siddonin algoritmi. Siinä kuitenkin oletetaan käytettävän kuutiomaisia vokseleita, joita approksimoidaan kuvaa muodostettaessa täydellisinä palloina, ja kiteitä, joiden poikkileikkaus on neliö [51]. D-690:n oletuskuvakoolla ja -FOVilla vokselit eivät ole kuutiomaisia – yhden vokselin koko on $5,47 \times 5,47 \times 3,34 \text{ mm}^3$ – eivätkä sen ilmaisinkiteet ole poikkileikkaukseltaan neliön muotoisia. Ehkä tämän lähtöoletuksen ja vokselien vaihtelevan muodon takia erikokoisten kuvien aktiivisuustiheysarvot eivät ole vertailukelpoisia. Leikekoolla $225 \times 225 \text{ px}$ ja oletus-FOVilla vokselit ovat



Kuva 16. Hoffman-aivoantomin rekonstruoidut ja korjatut kuvat eri FOV:illä (ylh. 700 mm, alh. 270 mm) ja kuvako'illa (vas. 128×128 px, oik. 225×225 px).

kuutiomaisempia (leveys $3,11 \times 3,11 \times 3,34$ mm³), joten näillä saadut kuvat voivat olla luotettavampia.

Jokaisella fantomilla suoritettiin GATEssa tarpeeksi monta simulaatiota noin 2 000 000 koinssidenssiparin tallentamiseksi rekonstruktiota varten ja yhtä monta simulaatiota, joista poimituilla sironneilla koinssidensseilla tehtiin sirontakorjaukset. Käytetyn virtuaalikoneen muisti ja aika rajoittivat kumpikin simulaatioiden kokoa. Derenzolla ja sylinterillä saatiin tarpeeksi koinssidensseja melko hyvälaatuisten kuvien rekonstruoimiseen, ja Hoffmanillekin muodostetuissa kuvissa voidaan erottaa, miten harmaa ja valkea aine ovat suunnilleen jakautuneet fantomissa. Pienten kohteiden, kuten yksittäisten elinten, kuvantamiseen työssä käytetyt simulaatiokoot voi-

vat siten olla riittäviä. NEMA-fantomilla olisi kuitenkin pitänyt simuloida kymmeniä kertoja enemmän hajoamisia, jotta leesiot olisi saatu kunnolla erottumaan taustasta. Jotta voitaisiin tehdä simulaatioita pidemmällä kuvannusajoilla ja isommilla fantomeilla, GATEa varten pitäisi varata enemmän muistia.

Aktiivisuusjakaumat näyttävät suunnilleen oikeanmuotoisilta kaikilla kuvakooilla ja FOVeilla, vaikka oikeiden aktiivisuustiheysarvojen saamiseksi vokselien koko ja muoto pitääkin säätää sopivaksi. Derenzolla huomattiin, että pienimmällä koko fantomin kattavalla FOVilla kuvakoon kasvattaminen ei paranna resoluutiota merkittävästi. Yhden pikselin leveys on silloin alle 0,4 mm eli pienempi kuin ^{18}F :n emittoiman positronin kantamasta vedessä [28] ja paljon pienempi kuin D-690:n erotuskyky [21].

Kuvarekonstruktiossa ei käytetty lentoaikadataa (TOF), koska se oli tuettu vain Siddonin algoritmilla. GATE-tiedostoihin oli kuitenkin tallennettu kunkin fotonin havaintoaika ja digitoija simuloi ajallista kohinaa. TOF mahdollistaisi annihilaatiopisteen tarkemman paikantamisen vastesuoralla, mikä olisi voinut parantaa kuvien resoluutiota ja vähentänyt kohinaa etenkin isommilla fantomeilla [55, 56]. Voisi olla hyödyllistä verrata TOF-rekonstruktiota parannellulla Siddonin algoritmilla VRT-rekonstruktioon.

Työssä ei myöskään mallinnettu TT-kuvantamista, vaikka TT-tutkimuksissa saatavia anatomisia, yksityiskohtaisia kuvia käytetään usein PET-kuvantamisen tukena [52, 57]. GATE on alun perin suunniteltu γ -kuvaussimulaatioiden toteuttamista varten [19] ja sisältänyt vain vähän työkaluja yksinkertaiselle TT- ja optiselle kuvantamiselle [47]. Tuoreemmissa tutkimuksissa [58, 59] GATEa on käytetty TT-kameran simuloimiseen, ja siten on saatu realistisia tuloksia. Näin ollen TT- ja PET-kuvantamisen simuloiminen samoilla fantomeilla rakenteellisen tiedon keräämistä ja PET-kuvien vaimennuskorjauksia varten voisi olla lupaava tutkimuskohde.

6 Yhteenveto ja johtopäätökset

Tässä opinnäytetyössä toteutettiin tehtäväsarja PET-kuvantamisen Monte Carlo -simuloinnille ja kuvarekonstruktiolle OMEGAssa. Tätä varten kirjoitettiin GATE-komentosarjat D-690:n ja neljän erilaisen fantomin mallintamiseksi sekä muokattiin valmista MATLAB-koodia kuvien muodostamiseksi simuloidun datan pohjalta. Sylinterifantomilla voitiin tarkastella rekonstruoidun aktiivisuusjakauman tasaisuutta, Derenzolla saavutettavissa olevaa kuvan resoluutiota, NEMalla kontrastia, ja Hoffmanilla mahdollisuutta kuvantaa monimutkaista, realistista kohdetta.

Kaikki simulaatiot ajettiin onnistuneesti jokaisella fantomilla. Sylinterillä, Derenzolla ja Hoffmanilla saatiin rekonstruoitua melko hyvälaatuisia kuvia, mutta NEMalla havaintoja saatiin selvästi liian vähän ja kohinaa oli liikaa. Tulevia simulaatioita ajatellen GATE kannattaisikin asentaa virtuaalikoneelle, jossa on enemmän käytettävää muistia ja jonne voisi mahdollisesti asentaa myös MATLABin tehtäväketjun sujuvoittamiseksi. Kuvanlaatua voitaisiin myös yrittää parantaa TT-simulaatioilla tai TOF-rekonstruktiolla. Lisäksi erilaisten fantomien tai kameran simuloimista ja muiden kuin OSEM-algoritmin testaamista olisi hyvä pohtia jatkossa.

Tekoälyn käyttö tutkielmassa

Tämän tutkielman teossa ei ole käytetty tekoälyä.

Viitteet

- [1] J. Kemppainen, K. L. Timonen ja H. Mussalo, *Duodecim* **136**, 1068–1075 (2020).
- [2] J. Knuuti, *Duodecim* **136**, 1059–1061 (2020).
- [3] I. Kuurne, valvontaraportti STUK-B 348, Säteilyturvakeskus, Vantaa, 2026.
- [4] D. Sarrut *et al.*, *Phys. Med. Biol.* **67**, 184001 (2022).
- [5] V.-V. Wettenhovi, väitöskirja, Itä-Suomen yliopisto, Kuopio, 2021.
- [6] J. G. Rogers, R. Harrop ja P. E. Kinahan, *IEEE Trans. Med. Imaging* **6**, 239–243 (1987).
- [7] M. Defrise, P. E. Kinahan ja C. J. Michel, kirjassa *Positron Emission Tomography: Basic Sciences*, toimittanut D. L. Bailey, D. W. Townsend, P. E. Valk ja M. N. Maisey (Springer-Verlag, Lontoo, 2005), p. 63–92.
- [8] G. L. Zeng, *Image Reconstruction: Applications in Medical Sciences*, 1. painos (de Gruyter, Berliini, 2017).
- [9] S. Tong, A. M. Alessio ja P. E. Kinahan, *Imaging Med.* **2**, 529–545 (2010).
- [10] V.-V. Wettenhovi, M. Vauhkonen ja V. Kolehmainen, *Phys. Med. Biol.* **66**, 065010 (2021).
- [11] J. Qi ja R. M. Leahy, *Phys. Med. Biol.* **51**, R541–R578 (2006).
- [12] A. P. Dempster, N. M. Laird ja D. B. Rubin, *J. R. Stat. Soc. B* **39**, 1–38 (1977).
- [13] H. M. Hudson ja R. S. Larkin, *IEEE Trans. Med. Imaging* **13**, 601–609 (1994).
- [14] P. J. Green, *IEEE Trans. Med. Imaging* **9**, 84–93 (1990).
- [15] A. R. de Pierro ja M. E. B. Yamagishi, *IEEE Trans. Med. Imaging* **20**, 280–288 (2001).
- [16] S. Pedemonte *et al.*, kirjassa *IEEE Nuclear Science Symposium & Medical Imaging Conference* (IEEE, Knoxville, TN, 2010), p. 2657–2661.
- [17] K. Thielemans *et al.*, *Phys. Med. Biol.* **57**, 867–883 (2012).
- [18] T. Merlin *et al.*, *Phys. Med. Biol.* **63**, 185005 (2018).
- [19] S. Jan *et al.*, *Phys. Med. Biol.* **49**, 4543–4561 (2004).
- [20] S. Agostinelli *et al.*, *Nucl. Instrum. Methods Phys. Res. A* **506**, 250–303 (2003).
- [21] V. Bettinardi *et al.*, *Med. Phys.* **38**, 5395–5411 (2011).

- [22] S. E. Derenzo *et al.*, IEEE Trans. Nucl. Sci. **24**, 544–558 (1977).
- [23] National Electrical Manufacturers Association, *NEMA Standards Publication NU 2-2001: Performance Measurements of Positron Emission Tomographs*, National Electrical Manufacturers Association, Rosslyn, VA, 2001.
- [24] E. J. Hoffman, P. D. Cutler, W. M. Digby ja J. C. Mazziotta, IEEE Trans. Nucl. Sci. **37**, 616–620 (1990).
- [25] S. Vallabhajosula, *Molecular Imaging: Radiopharmaceuticals for PET and SPECT* (Springer-Verlag, Berliini, 2009).
- [26] J. S. Lilley, *Nuclear Physics: Principles and Applications* (Wiley, Chichester, 2001).
- [27] A. Granov, L. Tiutin ja T. Schwarz, kirjassa *Positron Emission Tomography*, toimittanut A. Granov, L. Tiutin ja T. Schwarz (Springer-Verlag, Berliini, 2013), p. 3–24.
- [28] D. L. Bailey, J. S. Karp ja S. Surti, kirjassa *Positron Emission Tomography: Basic Sciences*, toimittanut D. L. Bailey, D. W. Townsend, P. E. Valk ja M. N. Maisey (Springer-Verlag, Lontoo, 2005), p. 13–40.
- [29] C. Leroy ja P.-G. Rancoita, *Principles of Radiation Interaction in Matter and Detection*, 2. painos (World Scientific, Singapore, 2009).
- [30] V. Pichler *et al.*, J. Nucl. Med. **59**, 1350–1354 (2018).
- [31] K. Shibuya *et al.*, Phys. Med. Biol. **52**, 5249–5261 (2007).
- [32] J. H. Hubbell, tekninen julkaisu NSRDS-NBS 29, National Bureau of Standards, Washington, D.C., 1969.
- [33] E. B. Podgorsak, A. L. Kesner ja P. S. Soni, kirjassa *Nuclear Medicine Physics: A Handbook for Teachers and Students*, toimittanut D. L. Bailey, J. L. Humm, A. Todd-Pokropek ja A. van Aswegen (IAEA, Wien, 2014), p. 1–48.
- [34] A. Zatcepin ja S. I. Ziebler, Z. Med. Phys. **33**, 4–12 (2023).
- [35] D. L. Bailey, kirjassa *Positron Emission Tomography: Basic Sciences*, toimittanut D. L. Bailey, D. W. Townsend, P. E. Valk ja M. N. Maisey (Springer-Verlag, Lontoo, 2005), p. 41–62.
- [36] F. James, Rep. Prog. Phys. **43**, 1145–1189 (1980).
- [37] A. Stevens, *Monte-Carlo Simulation: An Introduction for Engineers and Scientists*, 1. painos (CRC Press, Boca Raton, FL, 2023).
- [38] S. M. Ross, *Simulation*, 4. painos (Elsevier, Amsterdam, 2006).

- [39] D. P. Kroese, T. Taimre ja Z. I. Botev, *Handbook of Monte Carlo Methods*, 1. painos (John Wiley & Sons, Hoboken, 2011).
- [40] C. P. Robert ja G. Casella, *Monte Carlo Statistical Methods*, 1. painos (Springer New York, New York, NY, 1999).
- [41] C. J. Geyer, kirjassa *Handbook of Markov Chain Monte Carlo*, 1. painos, toimittanut S. Brooks, A. Gelman, G. L. Jones ja X.-L. Meng (Chapman and Hall/CRC, UK, 2011), p. 3–48.
- [42] F. Salvat, J. M. Fernández-Varea, E. Acosta ja J. Sempau, kirjassa *Workshop Proceedings* (OECD NEA, Issy-les-Moulineaux, 2001).
- [43] A. J. Reader ja H. Zaidi, *PET Clin.* **2**, 173–190 (2007).
- [44] M. A. Lodge ja E. C. Frey, kirjassa *Nuclear Medicine Physics: A Handbook for Teachers and Students*, toimittanut D. L. Bailey, J. L. Humm, A. Todd-Pokropek ja A. van Aswegen (IAEA, Wien, 2014), p. 312–397.
- [45] J. Allison, L. Garnier, A. Kimura ja J. Perl, *Int. J. Model. Simul. Sci. Comput.* **4**, 1340001 (2013).
- [46] S. Jan *et al.*, *Phys. Med. Biol.* **56**, 881–901 (2011).
- [47] OpenGATE Collaboration, *GATE Documentation (V9.3)*, 2023.
- [48] V.-V. Wettenhovi, OMEGA wiki, <https://github.com/villekf/OMEGA/wiki>, luettu 18.11.2025.
- [49] F. Jacobs *et al.*, *J. Comput. Inf. Technol.* **6**, 89–94 (1998).
- [50] P. Aguiar *et al.*, *Med. Phys.* **37**, 5691–5702 (2010).
- [51] A. Lougovski *et al.*, *Phys. Med. Biol.* **59**, 561–577 (2014).
- [52] S. R. Meikle ja R. D. Badawi, kirjassa *Positron Emission Tomography: Basic Sciences*, toimittanut D. L. Bailey, D. W. Townsend, P. E. Valk ja M. N. Maisey (Springer-Verlag, Lontoo, 2005), p. 93–126.
- [53] D. E. Cullen, J. H. Hubbell ja L. Kissel, raportti UCRL-50400 6, Rev. 5, Lawrence Livermore National Laboratory, Livermore, CA, 1997.
- [54] OpenGATE Collaboration, Gaten julkinen tietovarasto, <https://github.com/OpenGATE/Gate>, luettu 13.11.2023.
- [55] S. Vandenberghe *et al.*, *EJNMMI Phys.* **3**, 3 (2016).
- [56] J. Nuyts, M. Defrise, C. Morel ja P. Lecoq, *Phys. Med. Biol.* **70**, 245001 (2015).
- [57] T. Janatuinen ja J. Kemppainen, *Duodecim* **136**, 1062–1067 (2020).

- [58] A. W. Aldahouk, M. Sezdi ja M. Demir, Radiat. Phys. Chem. **239**, 113364 (2026).
- [59] H. Sekkat, A. Khallouqi, O. El Rhazouani ja A. Halimi, Radiat. Meas. **191**, 107597 (2026).