



Osaamista  
ja oivallusta  
tulevaisuuden  
tekemiseen

Jenni Hautojärvi, Merja Hepoaho

# Lääketieteellisten kuvien syväoppimisen viitekehys

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Hyvinvointi- ja terveysteknologia

Insinöörityö

31.5.2019

Tekijä Otsikko	Jenni Hautojärvi, Merja Hepoaho Lääketieteellisten kuvien syväoppimisen viitekehys
Sivumäärä Aika	76 sivua + 2 liitettä 31.5.2019
Tutkinto	insinööri (AMK)
Tutkinto-ohjelma	Tieto- ja viestintätekniikka
Ammatillinen pääaine	Hyvinvointi- ja terveysteknologia
Ohjaajat	Lehtori Sakari Lukkarinen, Metropolia AMK Liiketoiminnan kehitysjohtaja (Watson Health Imaging & PowerAI), TkT Jukka Remes, IBM
<p>Insinööriyön tavoitteena oli tehdä toteutettavuustutkimus siitä, miten DICOM-muotoista lääketieteellistä kuvadataa voidaan siirtää anonymisoidusti PACS-ympäristöstä tekoälyn koneoppimiseen ilman, että prosessissa menetetään kuvaan liitettyä arvokasta metadataa. Lisäksi haluttiin selvittää, miten Watson OpenScalen avulla voidaan lisätä tekoälyn antamiin tuloksiin läpinäkyvyyttä ja selitettävyyttä. Työ tehtiin yhteistyössä IBM Suomen kanssa, ja se toteutettiin IBM:n datatiedealustoilla käyttäen avoimen lähdekoodin ratkaisuja sekä yrityksen omia sovelluksia.</p> <p>Konseptista luotiin alustava arkkitehtuuri, jota käytettiin pohjana kahdessa eksploraatiivisessa asiantuntijahaastattelussa. Haastatteluista saatiin arvokasta tietoa DICOM-tiedostojen käsittelyyn liittyvistä haasteista ja kyettiin varmistamaan tiedonsiirron toteutusmenetelmän oikeellisuus.</p> <p>PowerAI-alustalle luotiin neuroverkkomallien syväoppimisympäristö ja sen toiminta testattiin Deep Learning Toolkitin omien testausmateriaalien avulla. Työtä varten ladattiin 128 Gt:n kokoinen testidata lääketieteellisiä DICOM-tiedostoja, jotka siirrettiin Cloud Object Storageen. Data sisälsi 1018 tietokonetomografialla tehtyä kuvantamistutkimusta 1012 potilaasta. Näiden joukosta valittiin 131 syöpädiagnoosia sisältävää kuvaa, jotka siirrettiin PowerAI-ympäristöön.</p> <p>Watson OpenScale testattiin kuvadatalalla, joka sisälsi käsinpiirrettyjä numeroita. Kuvia ei kyetty analysoimaan alkuperäisen suunnitelman mukaisesti palvelun kuvien analysointityökalun avulla, koska sen toteutus oli vielä keskeneräinen. Ongelma kierrettiin sillä, että kuvat muutettiin yksitasoisiksi vektoreiksi, jotka voitiin käsitellä ja analysoida OpenScalessa numeerisena/kategorisena datana.</p> <p>Tutkimuksessa todettiin, että konsepti on toteuttamiskelpoinen. Kaikkia osia ei ehditty yhdistämään toisiinsa insinööriyön aikataulujen puitteissa, mutta osien erillinen toiminta on kyetty varmistamaan ja työ siirtyi seuraavaksi toteutusvaiheeseen.</p>	
Avainsanat	Lääketieteellinen kuvantaminen, DICOM, Tekoäly, Syväoppiminen, IBM Cloud

Author Title	Jenni Hautojärvi, Merja Hepoaho The framework for deep learning of medical imaging
Number of Pages Date	76 pages + 2 appendices 31 May 2019
Degree	Bachelor of Engineering
Degree Programme	Information and Communications Technology
Professional Major	Health Technology
Instructors	Sakari Lukkarinen, Senior Lecturer, Metropolia UAS Jukka Remes, Business Development Executive, Dr (Tech.), Watson Health Imaging & PowerAI, IBM
<p>The aim of this Thesis was to make a feasibility study on how DICOM-formed medical imaging data can be transformed anonymously from PACS environment to machine learning of Artificial Intelligence, without losing the valuable metadata attached to the image. In addition, we wanted to find out how Watson OpenScale can add transparency and explainability to the results given by AI. The work was done in co-operation with IBM Finland, and it was carried out using IBM data science platforms using open source solutions and company's own applications.</p> <p>A preliminary architecture was created for the concept, and it was used as the basis for two explorative expert interviews. The interviews provided valuable information about the challenges involved in handling DICOM-files, and we were able to verify the correctness of the data transfer method.</p> <p>Deep learning environment was created in the PowerAI platform for neural network model training, and it was tested using Deep Learning Toolkit's own test materials. A 128 GB test data set of medical DICOM-files were uploaded to Cloud Object Storage. The data included 1018 computed tomography imaging studies from 1012 patients. Of these, 131 images with cancer diagnoses were selected and transferred to the PowerAI environment.</p> <p>Watson OpenScale was tested with image data containing handwritten digits. Pictures couldn't be analyzed according to the original plan using the image analysis tool because its implementation was still incomplete. A workaround to this issue was to convert the images into one-dimensional vectors, after which they could be processed and analyzed in OpenScale as numeric data.</p> <p>During the study it was found out that the concept is feasible. All the parts could not be connected to each other within the work schedule, but the separate operations of different parts were verified, and the project will be transferred to the implementation phase.</p>	
Keywords	Medical imaging, DICOM, Artificial Intelligence, Deep Learning, IBM Cloud

## Sisällys

### Lyhenteet

1	Johdanto	1
2	Lääketieteellinen kuvantaminen ja kuvien käyttö	2
2.1	Lääketieteelliset kuvantamismenetelmät	2
2.2	Lääketieteellisten kuvien tallennus- ja käsittely-ympäristö	6
2.3	Digitaalisten lääketieteellisten kuvien ominaisuudet	9
2.4	Digitaalisten kuvien käsittely lääketieteellisessä kuvantamisessa	14
2.5	Lainsäädäntö ja ohjeistus	15
3	Kuvien syväoppiminen tekoälykehityksessä	18
3.1	Tekoäly, koneoppiminen ja syväoppiminen	18
3.2	Syväoppiminen prosessina	25
3.3	Syväoppimisen alustavaatimukset	29
3.3.1	Laitteisto, ohjelmistot ja kirjastot	30
3.3.2	IBM:n järjestelmät	35
3.4	Opetusdatan laatu ja tulosten vääristyminen	37
4	Tutkimusmenetelmät	39
4.1	Asiantuntijahaastattelu	39
4.2	Toteutettavuustutkimus	41
5	Toteutus ja tulokset	43
5.1	Haastattelujen tulokset	45
5.2	PowerAI-ympäristön pystytys ja testaus	51
5.3	DICOM-tiedostojen siirtäminen ja tarkastelu	53
5.4	Watson OpenScale	57
5.5	Tulokset	61
6	Pohdinnat ja yhteenveto	64
	Lähteet	68

## Liitteet

Liite 1. Radiologisten modalitytien kuvantamisominaisuudet

Liite 2. OpenScale accuracy testitulokset

## Lyhenteet

AI	Artificial Intelligence. Tekoäly. Yleistermi, joka pitää sisällään koneoppimisen ja syväoppimisen.
AI HLEG	European Commission's High-Level Expert Group on Artificial Intelligence. Euroopan komission tekoälyä käsittelevä korkean tason asiantuntijaryhmä.
ANN	Artificial Neural Networks eli keinotekoinen neuroverkko. Kuvaa syväoppimisen rakennetta.
CNN	Convolutional Neural Network. Konvoluutioneuroverkko. Hahmojen tunnistamiseen erikoistunut neuroverkko.
COS	Cloud Object Storage. IBM:n pilvialusta erilaisten datojen säilytykseen.
CPU	Central Processing Unit. Suoritin tai prosessori on tietokoneen osa, joka suorittaa tietokoneohjelman sisältämiä konekielisiä käskyjä.
CUDA	Compute United Device Architecture. Nvidian kehittämä alusta ja ohjelmointirajapinta GPGPU-ohjelmointiin.
DICOM	Digital Imaging and Communications in Medicine. Lääketieteellisten kuvatiedostojen tallennusmuotojen ja tiedonsiirron standardi.
FOV	Field of view. Näkökenttä. Röntgenkentän kuvantamisalue.
GDPR	General Data Protection Regulation. Yleinen tietosuoja-asetus.
GPU	Graphics Processing Unit. Graafinen prosessointiyksikkö.
HIS	Hospital Information Systems. Sairaalan käyttöjärjestelmä.
HSV	Hue-saturation-value. Värimalli, joka määrittää värit sävyn, kylläisyyden sekä kirkkauden perusteella.

HUS	Helsingin ja Uudenmaan sairaanhoitopiiri.
IR	Image reseptor. Kuvareseptori eli ilmaisin.
Kerma	Air kerma, kinetic energy released per unit mass. Säteilyenergian absorboituminen yksikkö per massa.
LSTM	Long short-term memory. Toistuva neuroverkko arkkitehtuuri, jossa on takaisinkytkentä yhteys.
MK	Magneettikuvaus. Englanniksi Magnetic Resonance Imaging (MRI).
NEMA	National Electrical Manufacturers Association.
NLP	Natural Language Processing. Luonnollisen kielen prosessointi. Tietokoneohjelman käyttäminen luonnollisen tekstin ja puheen analysointiin ja tuottamiseen.
OpenCL	Open Computing Language. Avoimen lähdekoodin GPU-ohjelmointirajapintamääritelmä.
OS	OpenScale. IBM:n työkalu, jolla voidaan valvoa ja hallita tuotannossa olevien kone- ja syväoppimismallien sisällön vääristymiä ja tulosten oikeudenmukaisuutta.
PACS	Picture Archiving and Communication Systems. Lääketieteellisten kuvatiedostojen arkistointijärjestelmä.
PET	Positroniemissiotomografia. Isotooppilääketieteen alaan kuuluva kuvantamismenetelmä.
PSP	Photostimulable Phosphor. Röntgensäteilyä tallentava kuvalevy.
RGB	Red-Green-Blue. Värimalli, joka määrittelee värit sen mukaan, kuinka paljon ne sisältävät punaista, sinistä ja vihreää väriä.

RIS	Radiology Information System. Radiologian osaston tietojärjestelmä.
SIMD	Single Instruction, Multiple Data. Komento suoritetaan usealle arvolla samanaikaisesti.
SPECT	Yksifotoniemissiotomografia. PET-kuvauksen tyyppinen kuvantamismenetelmä, jossa käytetään liikkuvaa kameraa.
SSH	Secure Shell. Komentorivitasoinen etäyhteys tietokoneeseen.
THL	Terveyden ja hyvinvoinnin laitos.
TKHJ	Tietokannan hallintajärjestelmä. Ohjelmisto, jonka avulla hallinnoidaan tietokantoja.
TPU	Tensor Processing Unit. Googlen kehittämä tensorikäsitteily-yksikkö.
TT	Tietokonetomografia. Englanniksi Computerised Tomography (CT). Röntgensäteilyyn perustuva radiologinen lääketieteellinen kuvantamismenetelmä.
UÄ	Ultraääni. Ääntä, jonka taajuus on ihmisen korvan kuuloalueen yläpuolella. Lääketieteessä kaikukuvantamiseen perustuva kuvantamismenetelmä.
VPN	Virtual Private Network. Virtuaalinen erillisverkko. Tapa, jolla yksityisiä verkkoja voidaan yhdistää julkisen verkon yli yksityiseksi verkoksi.
WML	Watson Machine Learning. IBM:n yrityksille suunnattu palvelu tekoälyn käyttöönottoon.



## 1 Johdanto

Tekoäly on yksi aikakautemme puhutuimmista teknologioista. Sen mahdollisuuksia pidetään lähestulkoon rajattomina, ja siksi se herättää laajaa kiinnostusta lukemattomilla eri aloilla, myös lääketieteessä ja terveydenhuollossa. Tekoälyn vahvuus on sen kyky käsitellä suuria datamääriä tehokkaasti ja väsymättä, analysoida niiden sisältöä ja tehdä päätelmiä saatujen tulosten perusteella [1]. Toimiessaan se voi esimerkiksi säästää suuret määrät hoitohenkilökunnan aikaa, antaa parempia ja tarkempia tuloksia sairauksien diagnosoinnissa ja tehokkuuden parantuessa tuoda lisäksi taloudellisia säästöjä.

Mahdollisuuksien mukana pinnalle ovat nousseet myös tekoälyn tuomat potentiaaliset uhat. Julkisuudessa on viime vuosina uutisoitu muun muassa Microsoftin rassistiseksi muuttuneesta Tay-chatbotista [2] sekä Amazonin epäonnistuneesta tekoälypohjaisesta rekrytointityökalusta, joka oppi hakemusten miesvoittoisuuden takia syrjimään naisia [3]. Opetusdatan valinnassa täytyy noudattaa tarkkuutta, sillä tekoäly ei ymmärrä mitään käsittelemänsä datan sisällöstä, eikä myöskään arvota sitä eettisten periaatteiden mukaan.

Tekoälyä kokeillaan jo lääketieteessä useilla eri tavoilla. Sillä on saatu lupaavia tuloksia muun muassa erilaisten syöpien ja silmäsairauksien diagnosoinnissa, magneettikuvien analysoinnissa, koomapotilaiden heräämisennusteissa sekä terveyskeskuksen toiminnanohjauksen parantamisessa. Usein tekoälyn ehdottamat diagnoosit ovat olleet tarkempia kuin samoja potilastietoja tarkistelleiden lääkäreiden, ja tulevaisuudessa tekoäly tulee todennäköisesti olemaan entistä useammin lääkärin antamien diagnoosien tukena. [4; 5; 6; 7.]

Tässä insinööriyössä tutkitaan, miten tekoälykehityksen koneoppimisympäristöön saataisiin vietyä DICOM-muodossa olevaa lääketieteellistä kuvadataa anonymisoidusti ilman, että siitä häviäisi prosessin aikana arvokasta metadataa. Samalla halutaan tuoda tekoälyn antamiin vastauksiin läpinäkyvyyttä ja selitettävyyttä. Testaukseen käytetty kuvadata on valittu opetus- ja tutkimuskäyttöön tarkoitetusta kuvapankista, eikä sen käyttöön ole vaadittu erillistä tutkimuslupaa. Työssä ei oteta kantaa siihen, millä tavalla opetettua mallia voitaisiin jatkohyödyntää tai liittää olemassa oleviin järjestelmiin.

Insinööriyö on toteutettu yhteistyössä IBM Suomen kanssa, ja se on toteutettu IBM:n datatiedealustoilla käyttäen sekä avoimen lähdekoodin ratkaisuja että yrityksen omia sovelluksia. IBM on kansainvälinen teknologiayritys, joka on erikoistunut tekoälyä hyödyntävään tietojenkäsittelyyn ja tekoälyn kehittämiseen suunnattuihin teknologia-alustoihin sekä yleisesti että muun muassa terveydenhuoltoon liittyen.

## 2 Lääketieteellinen kuvantaminen ja kuvien käyttö

### 2.1 Lääketieteelliset kuvantamismenetelmät

Lääketieteellinen kuvantaminen on keskeisessä asemassa sairauden ja tapaturman vaurioiden toteamisessa. Lisäksi sitä käytetään myös esimerkiksi hoidon ja raskauden seurannassa. Radiologiset kuvantamismenetelmät eli modaliteetit hyödyntävät kuvantamisessa säteilyä, ääniaaltoja tai magneettikenttää. Radiologia voidaan myös jakaa diagnostiseen ja toimenpideradiologiaan. Diagnostinen radiologia keskittyy nimensä mukaisesti diagnoosien tekemiseen. Toimenpideradiologiaa käytetään kuvantamismenetelmänä kirurgiassa, angiografiassa sekä sädehoidossa. Edellä mainitut radiologiset kuvantamismenetelmät ovat lähes kivuttomia. Poikkeuksena voidaan mainita potilaaseen asennettava kanyyli, jota käytetään esimerkiksi tapauksissa, joissa varjoaine annetaan potilaalle suonensisäisesti. Itse kuvantamistapahtuma on kivutonta. Lisäksi näkyvää valoa käytetään kuvantamismenetelmänä esimerkiksi silmänpohjankuvantamisessa. Liitteeseen 1 on listattu eri radiologisten kuvantamistekniikoiden ominaisuuksia ja lääketieteellisiä käyttökohteita. [8; 9.]

Röntgensäteilyyn perustuvat kuvantamismenetelmät ovat analoginen sekä digitaalinen röntgenkuvaus, tietokonekerroskuvaus eli tietokonetomografia (TT, englanniksi CT) ja angiografia. Röntgenkuvaus perustuu siihen, että röntgensäteet läpäisevät ihmisen kudokset eri tavoin. Tiheät kudokset, kuten luut, läpäisevät vähiten säteitä ja näkyvät siten vaaleana kuvassa. Eniten säteilyä läpäisevät kudokset näkyvät kuvassa harmaana. Vatsan alueelta otetussa kuvassa ihmisen pehmeät elimet eivät erotu juuri lainkaan, sillä röntgensäteet läpäisevät kaikki ihmisen elimet suurin piirtein samalla tavalla. Poikkeuksena on rintakehä, sillä ilmapitoiset keuhkot erottuvat tummina ja niissä olevat poikkeamat, esimerkiksi neste, erottuvat vaaleana tummaa taustaa vasten. [10.] TT-kuvaus

poikkeaa röntgenkuvaksesta siten, että potilas on vaakatasossa ja häntä kiertää säteilevä röntgenputki, jonka vastakkaisella puolella on säteen vastaanottava ilmaisim. Jokaisesta kierroksesta muodostuu poikkileikkaus kuvauskohteesta. [11.] Angiografiassa tutkimus perustuu varjoainekuvaukseen ja sitä käytetään esimerkiksi sydämen sepelvaltimoiden kuvantamiseen. Varjoaineella saadaan röntgenkuvissa näkyviin elimiä, jotka muuten läpäisevät helposti röntgensäteitä. [12.]

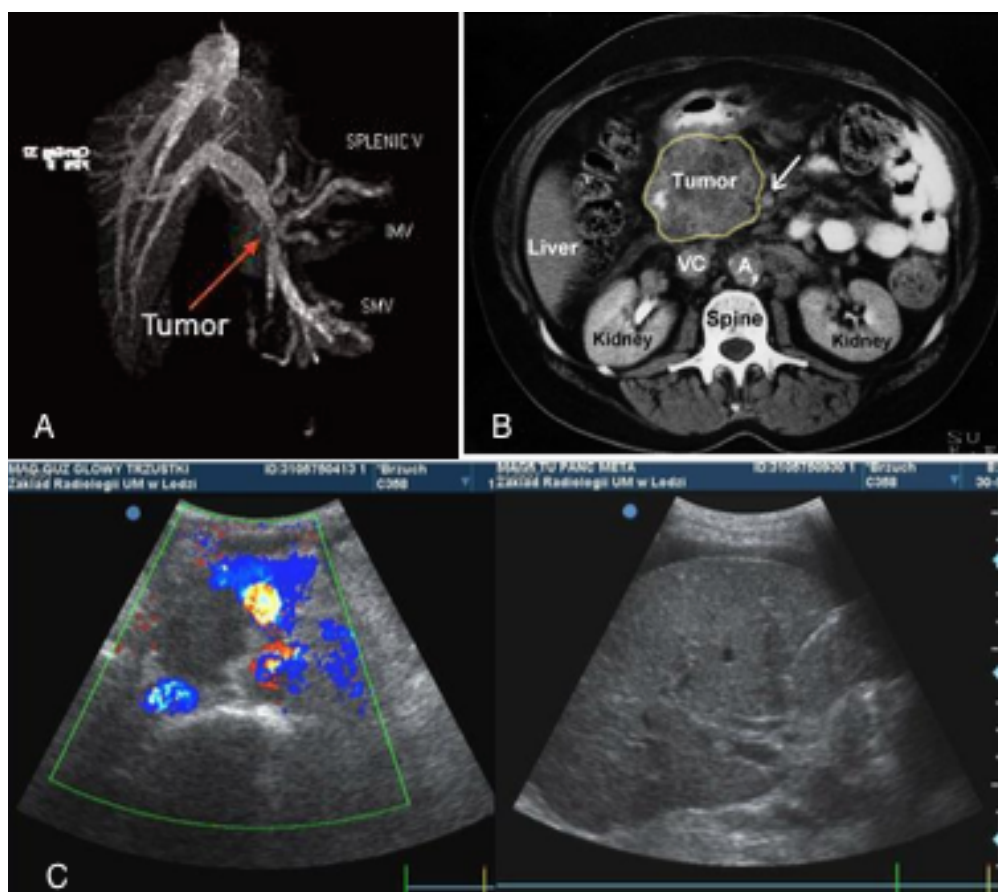
Säteilyä hyödyntävät isotooppilääketieteelliset menetelmät, joihin kuuluvat muun muassa aineenvaihduntaa kuvantavat positroniemissiotomografia (PET) ja yksifotoniemissiotomografia (SPECT). Isotooppilääketieteellisessä menetelmässä potilaalle annetaan yleensä suonensisäisesti isotooppimerkkiainetta, jonka hajoamista kutsutaan radioaktiiviseksi hajoamiseksi. Hajoamisen aikana atomista vapautuu säteilyä, jonka määrä voidaan mitata ja kuvata erityisellä ilmaisimella joko filmille tai digitaalisesti. [13.]

Kaikukuvauksessa, eli ultraäänessä (UÄ) ja doppler-ultraäänessä, kuva muodostetaan lähettämällä tutkittavalle alueelle ääniaaltoja taajuusvälillä 1-20 MHz ja vastaanottamalla kudoksien rajapinnoista heijastuvaa kaikua. Doppler-UÄ:ssä tutkimus perustuu dopplerilmiöön, jossa ääniaallon taajuus muuttuu, kun aaltojen lähde ja havaitsija siirtyvät toisiinsa nähden. Nämä ääniaaltojen vaihtelut voidaan näyttää ruudulla väreinä, jolloin niillä voidaan kuvata esimerkiksi veren virtausta. Tavallisimpia UÄ:llä tutkittavia kohteita ovat verisuonet, sisäelimet, kilpirauhanen, sydän sekä lihakset, jänteet ja nivelet. Menetelmää ei voi käyttää esimerkiksi luuston tai keuhkojen kuvaukseen, koska luu ja ilma vaimentavat ultraäänien värähtelyä voimakkaasti ja signaali heijastuu voimakkaasti takaisin jo sisäänmenorajapinnasta. [14.]

Magneettikuvauksessa (MK, englanniksi MRI) hyödynnetään ihmisessä olevien molekyylien ja atomien luontaisia magneettikenttiä. Vetyatomien ytimen spin-ominaisuudet aikaansaavat pienet magneettikentät, jotka asettuvat magneettitutkimuksessa ulkoisen magneettikentän suuntaiseksi. Suuntausta voidaan muuttaa ulkoisilla radioaalloilla ja vetyatomien virityksen purkautumisesta ylimääräinen energia palautuu takaisin radioaaltoina, muodostaen heikkoja radiosignaaleja. Signaaleja kyetään kuvantamisessa muuntelemaan ja mittaamaan, ja sitä kautta muodostamaan magneettikuvia, jotka voidaan esittää sekä vaaka- että pystyleikkeinä ja joilla voidaan saada kuvattavasta kohteesta monia erityyppisiä näkymiä kuvattavaan kudokseen (mm. anatomian tarkastelu,

nestekertymien havaitseminen, toiminnallinen kuvaus). Magneettikuvaus on käyttötarkoituksiltaan monipuolisempi kuin muut kuvantamistekniikat. [15.]

Tutkittava osa-alue ja tutkimuksen tarkoitus määrittelee kuvantamismenetelmän. UÄ- ja silmänpohjankuvantamisessa kuva on nähtävissä reaaliajassa. Muissa menetelmissä kuva voidaan nähdä vasta kuvantamisen jälkeen. [16.] Kuvassa 1 on esitetty haimasyövän tutkimustapoja. Kuva A on muodostettu MK-tutkimuksella, kuva B on muodostettu TT-kuvantamisella ja kuva C on ultraäänitutkimuksesta, tarkemmin sanottuna doppler-ultraäänitutkimuksesta, missä kaksiulotteisen leikekuvan päälle on väreillä koodattu doppler-menetelmällä saadut virtausnopeudet.



Kuva 1. Haimasyövän tutkiminen eri kuvantamislaitteilla. Kuvantamisessa käytetään harmaasävyjen lisäksi monivärikuvausta, jota hyödynnetään dopplerkuvantamisessa (kuva C). Varjoaine lisää vaimennusta verisuonissa, jotka muuten erottuisivat huonosti muista pehmytosa- ja vesipitoisista kohteista (kuva A). TT-kuva esitetään lähtökohtaisesti viipaalekuvana (kuva B). [17; 18; 19.]

Röntgen oli pitkään ainoa kehon sisäinen kuvantamismenetelmä, mutta tietotekniikan kehittymisen ansiosta nykyisin kyetään muodostamaan monimutkaisilla tietokoneohjelmiilla TT-, UÄ- ja MK-kuvia. Monimutkaiset kuvantamislaitteet sekä niiden käyttö ja ylläpito ovat kalliita. Lisäksi kuvantamista tekevällä henkilökunnalla on oltava niihin erikoiskoulutus. Duodecim Terveyskirjaston sivuilla oli arvioitu TT- ja MK-laitteiden kappalehintojen liikkuvan samoissa hintaluokissa, eli noin 300 000 – 1 000 000 eurossa. Laitteiden lisäksi niiden käyttö edellyttää myös erityisiä tiloja, joissa on huomioitu sekä magneettittä radioaalto suojauksia. Kun kuvaa ei enää hyödynnetä hoidossa, se arkistoidaan pitkäaikaisarkistoon PACS (Picture Archiving and Communication Systems) -järjestelmässä, jonka ylläpito aiheuttaa myös kustannuksia. [8; 20.]

HUS (Helsingin ja Uudenmaan sairaanhoitopiiri) -kuvantamisyksiköissä tehtiin vuonna 2016 radiologisia tutkimuksia 974 700 (taulukko 1), joista 72 100 oli magneettikuvauksia ja 132 300 oli tietokonetomografiakuvauksia. Henkilökunnan määrä HUS-kuvantamisessa 21.9.2017 oli 1 299 henkilöä. Hoitohenkilökuntaan kuului 741 henkilöä, 277 lääkäriä, 37 erityistyöntekijää ja 244 muuta henkilökuntaa. [21.]

Taulukko 1. HUS-Kuvantamisen palveluiden tutkimusmäärät vuonna 2016.

<b>Kuvantamisen palvelut</b>	<b>Yhteensä (kpl)</b>
Radiologiset tutkimukset	974 700
joista magneettikuvauksia	72 100
joista tietokonetomografioita	132 300
Kliinisen fysiologian ja isotooppi -lääketieteen tutkimukset	64 760
joista PET-TT	2 400
Kliinisen neurofysiologian tutkimukset	21 800
joista ENMG	9 700
Lääkintätekniiikan tuntiperusteiset palvelut (omat tunnit)	46 000
Silmänpohjankuvaukset	7 400

HUS-kuvantamisen yksiköissä vuonna 2019 pään TT-kuvaus maksaa 139 euroa ja erittäin laaja pään TT-kuvaus 221 euroa. Magneettitutkimukset pään alueelta maksavat 161 - 476 euroa, riippuen kuvauksen laajuudesta. Hinnat koostuvat pelkästään kuvauksesta eivätkä sisällä radiologin tulkintaa tai lausuntoa. PACS-arkistoinnin hinta on 2,4 euroa

kappaleelta per tutkimus, kun tutkimuksen tekee HUS:n sisäinen yksikkö. Ulkopuolisten yksiköiden tekemien tutkimusten arkistointi on 2,5 euroa kappale. [22.]

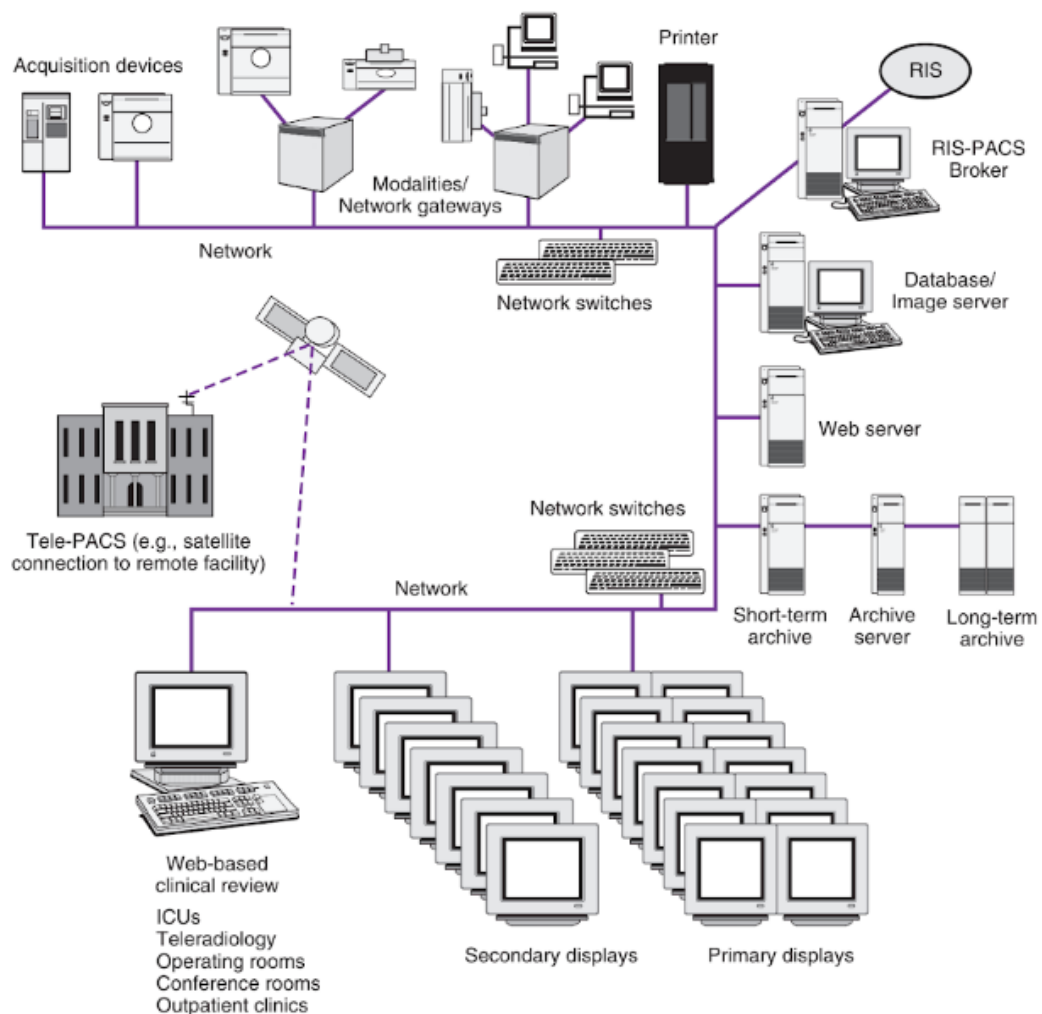
## 2.2 Lääketieteellisten kuvien tallennus- ja käsittely-ympäristö

Potilastiedot ja potilaista otetut lääketieteelliset kuvat ovat laissa suojattua materiaalia, ja niiden käyttö on hyvin säännösteltyä. Tässä työssä keskitytään lääketieteellisiin DICOM (Digital Imaging and Communications in Medicine) kuviin. DICOM:a käytetään lähes kaikissa kuvantamis- ja sädehoitolaitteissa, kuten röntgenkuvaus, tietokonetomografia, magneettikuvaus ja ultraääni, mutta se on yleistynyt myös muiden lääketieteen alojen, kuten silmä- ja hammaslääketieteen, laitteissa. DICOM on kansainvälinen standardi lääketieteellisten kuvantamistietojen lähettämiseen, tallentamiseen, hakuun, tulostamiseen, käsittelyyn sekä näyttämiseen. Standardia kutsutaan myös nimellä *NEMA-standardi PS3* ja *ISO-standardi 12052:2017: ”Terveystietotekniikka – Digitaalinen kuvantaminen ja viestintä lääketieteessä (DICOM) mukaan lukien työnkulku ja tiedonhallinta”*, jonka tekijänoikeudet omistaa National Electrical Manufacturers Association (NEMA). Standardi mahdollistaa tiedostojen vaihdon kahden yksikön välillä, jotka pystyvät lähettämään ja vastaanottamaan kuvia ja potilaan tietoja DICOM-muodossa. DICOM-tiedosto voi sisältää potilastunnuksen, potilaan nimen, sukupuolen ja iän, kuvantamismenetelmän, laitteen tietoja sekä muita tietoja. [23.]

Sairaaloissa käytetään PACS-järjestelmää, joka käsittää muun muassa kuvamateriaalin arkistoinnin, kuvan käsittelyn, siirron tietoverkkoon sekä integraation muihin potilastietojärjestelmiin [24]. Järjestelmän kautta kuviin on pääsy usealle käyttäjälle samanaikaisesti. Esimerkiksi lääkäri voi tarkastella potilaan kuvia samalla, kun radiologi suorittaa niille annotointia. [20.] Annotoinnilla tarkoitetaan aineiston kuvaamista, luokittelua ja jäsentelyä systemaattisella tavalla. Esimerkiksi radiologi voi annotoida kuvaan löydöksiä merkitsemällä ne joko itse kuvaan, tai siihen liitetään järjestelmällisesti nimikkeitä. Annotointia ei ole esimerkiksi kuvan löydösten sanelu potilastekstiin. [25.] Kuvan annotointi tapahtuu ohjelmalla, joka mahdollistaa tekstin tai merkkien lisäämisen digitaalisesti kuvaan.

PACS-järjestelmä myös välittää tietoja, kuten röntgenkuvien lausuntoja ja lähetetietoja, eri potilastietojärjestelmien välillä. PACS koostuu verkottuneista tietokoneista,

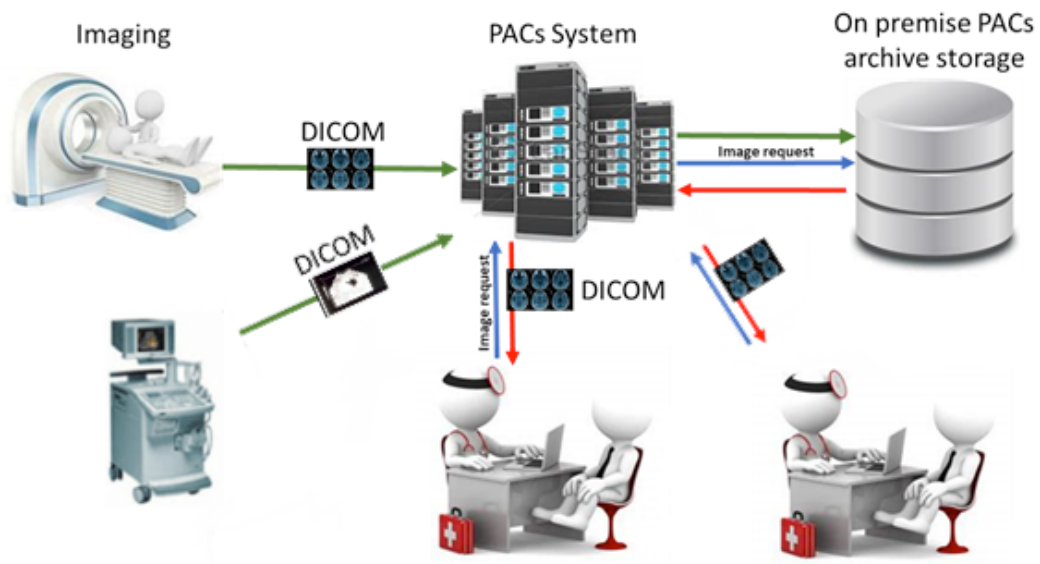
palvelimista ja arkistoista. Kuvassa 2 on esitelty esimerkki PACS-verkosta. Verkko voi koostua monista eri osista, kuten lukupisteistä, radiologien tai lääkärin työasemista, internet-yhteydestä, järjestelmän valvonta- ja hallinta-asemista, arkistointijärjestelmistä sekä monista liitännöistä erilaisiin sairaala- ja radiologisiin järjestelmiin. Radiologian osastoilla on käytössä RIS-tietojärjestelmä (Radiology Information System), joka on osa PACS-järjestelmää. Sitä käytetään osaton tutkimusten tilaamiseen ja tulosten raportointiin. Se voi myös sisältää kuvauslistan ja aikataulun, tutkimuksen suorituskyvyn seurannan sekä laskutuksen. PACS ja RIS ovat integroitavissa suoraan HIS-tietojärjestelmään (Hospital Information Systems), jota käytetään koko sairaalassa. [20.]



Kuva 2. Tyypillinen kuvaus PACS-verkosta. [20, s.10.]



PACS-järjestelmä ottaa vastaan minkä tahansa digitaalisen kuvan, joka on DICOM-muodossa. PACS-arkistoon tallennetut tiedot noudattavat DICOM-standardia, minkä johdosta muiden sairaaloiden PACS-arkistoista, mukaan lukien ulkomailla kuvatut tutkimukset, on mahdollista siirtää ja katsella kuvia esimerkiksi internetverkon välityksellä. PACS-järjestelmällä ja DICOM-standardilla tiedonsiirto onnistuu eri valmistajien potilaskuvantamislaitteiden ja muiden laitteiden välillä, kuten kuvassa 3 on esitetty. Kuvassa potilas on käynyt erilaisissa lääketieteellisissä kuvauksissa, joista on muodostunut DICOM-tiedostoja (vihreä nuoli), jotka on arkistoitu PACS-järjestelmään. Lääkäri tekee pyynnön kuvasta PACS-järjestelmälle (sininen nuoli) ja PACS siirtää sen lääkärille (punainen nuoli). Yhdellä pyynnöllä voidaan hakea yhtä tai useampaa kuvaa samanaikaisesti. Kuvassa tilanne voi olla esimerkiksi se, että potilas on lääkärin vastaanotolla ja samanaikaisesti kuvaa tarkastelee kaksi asiantuntijaa, joita lääkäri konsultoi. [20.]



Kuva 3. DICOM-tiedoston kulku kuvantamislaitteiden, PACS-ympäristön ja lääkärin työasemien välillä.

PACS-järjestelmä on usein räätälöity laitoksen tarpeisiin ja niitä tuottavat eri valmistajat. Esimerkiksi HUS valitsi vuonna 2019 julkisen kilpailutuksen voittajaksi Siemens Healthcare Oy:n, ja Etelä-Pohjanmaan sairaanhoitopiirillä on käytössä Carestreamin PACS-järjestelmä. Etelä-Pohjanmaan sairaanhoitopiirissä on integroitu Effica-järjestelmän PACS ja RIS, jotta kaikkien tarvittavien ohjelmien käyttö onnistuu yhdellä sisäänkirjautumisella [24; 26]. PACS-järjestelmien ohjelmistot ovat yleensä samankaltaisia eri



PACS:issa, mutta komponentit on voitu järjestää eri tavalla. PACS-järjestelmän suunnittelussa huomioidaan erityiset tekijät, kuten potilaiden määrä, alueiden lukumäärä, joissa kuvat tulkitaan, paikat (esimerkiksi sairaala ja terveysasema tai vuodeosastot), joissa muut lääkärit kuin radiologit katsovat kuvia, sekä järjestelmään käytettävä budjetti. [20.]

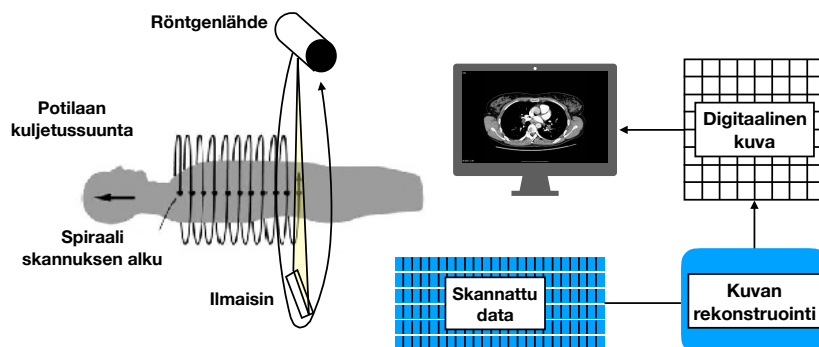
### 2.3 Digitaalisten lääketieteellisten kuvien ominaisuudet

Tässä insinööriyössä käytettävä kuvamateriaali koostuu röntgensäteilyyn perustuvalla TT-kuvantamismenetelmällä otetuista kuvista. Tämän vuoksi työn teoriassa on keskitytty enemmän kyseisen menetelmän kuvantamisprosessiin ja kuvan muodostukseen. Tässä työssä käydään yleisesti läpi kuvantamisen vaiheet ja kuvan muodostukseen vaikuttavia asioita. Koska digitaaliset kuvat ovat yleisempiä, ja tekoälyä varten kuvat täytyy olla digitaalisessa muodossa, keskitytään tässä työssä vain digitaalisen kuvan käsittelyyn.

Digitaalinen röntgenkuvaus perustuu epäsuoraan tai suoraan tekniikkaan. Epäsuorassa digitaalisessa röntgenkuvantamisessa röntgensäteilyn vastaanottaa filmin sijaan digitaalisen kuvan tallentava puolijohdelevy, johon syntyy säteilyn vaikutuksesta atomien viritystiloja. Kuvalevy luetaan optisen helium-neon-laserin ja valomonistinputken sisältävän lukulaitteen avulla, jossa laser purkaa viritystilat piste pisteeltä. Tällä menetelmällä saavutetaan hyvä paikkaerotuskyky. Suoraa digitaalikuvantamista käytetään esimerkiksi TT-kuvauslaitteissa, joissa on hyvän paikkaerotuskyvyn lisäksi hyvä kontrasti. Laitteissa on suuren paikkaresoluution omaava säteilynilmaisinjärjestelmä, joka muuttaa havaitun röntgensäteilyn suoraa sähköiseen muotoon. Potilaan säteilyannoksen seuraamiseksi mitataan kermaa. Kerma (Air kerma, kinetic energy released per unit mass) on mittaus säteilyenergian (joule, J) absorboituminen yksikkö per massa (kg), eli kuinka paljon kohde on imenyt säteilyenergiaa kiloa kohden. Siksi kerman määrä ilmaistaan J/kg tai gray (Gy). [16; 20.]

TT-kuvantamisella otetaan leikekuvia kolmiulotteisesta kohteesta. Leikekuvien etu on, ettei niissä esiinny elinten päällekkäin kuvautumista, kuten perinteisessä röntgentutkimuksessa. TT-laite sisältää röntgengeneraattorin, kuvaustelineen, ilmaisimen, potilaspöydän sekä järjestelmää ohjaavan ja tiedon tallentavan tietokonejärjestelmän. TT-kuvantamisessa on olemassa erilaisia tekniikoita. Kuvassa 4 on esitetty esimerkkinä spiraali-TT-tekniikka, jossa potilaspöytä liukuu kuvantamislaitteessa hitaasti kuvauksen

aikana. Potilaan ympärillä pyörii vastakkain röntgenlähde sekä säteilyn vastaanotin, eli ilmaisimien, joka kerää skannattua signaalia (projektiokuvia ja kudoksen absorptiokertoimia) useista tunnetuista kulmista. Signaali muunnetaan analogia-digitaalimuuntimella (A/D-muunnin) digitaalseksi. Skannatusta datasta absorptiokerroin, eli kudoksen imemän säteilynmäärä, muunnetaan niin sanotulle TT-asteikolle, jossa ilma saa arvon -1000 ja vesi arvon 0. Projektiokuvista muodostetaan leikekuvia laskemalla arvot takaisinprojisoinnilla, jotta vältetään pistemäisen kohteen muuttuminen tähtimäiseksi. Skannatun datan rekonstruktiossa lasketaan digitaalinen numerotaulu (kuvamatriisi). [16.]

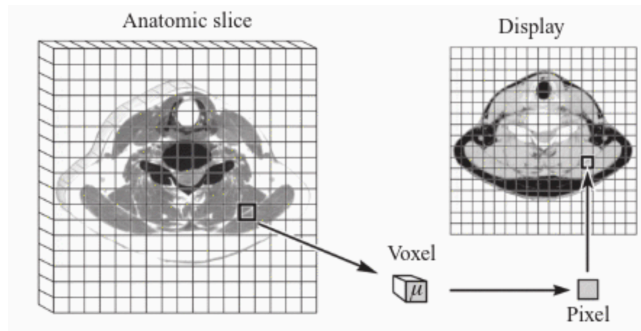


Kuva 4. Kuvassa on kuvattu TT-laitteen toimintaperiaate. Vasemmalla on esitetty kuvantamistapahtuma. Skannatusta datasta muodostetaan rekonstruoinnin kautta digitaalinen kuva. Digitaalinen kuva koostuu pikseleistä, joiden informaation määrää A/D-muuntimen erottelukyky eli monellako bitillä analoginen signaali muunnetaan digitaalseksi arvoksi, biteiksi.

TT-kuvantamisessa potilaan ympärillä sijaitsevat röntgenlähde ja ilmaisimien pyörähtävät yhden kierroksen jokaista viipalekuvaa varten. Pitch-parametrilla määritetään viipalepaksuus, joka kertoo, kuinka pitkän matkan pöytä siirtyy aina yhden kuvantamiskierroksen aikana. TT-kuvissa on vain vähän sironnutta säteilyä ja siksi siinä on suuri kontrasti. Kuvantamisessa säteilyä on rajoitettu sekä röntgenlähteen että ilmaisimien puolella sijaitsevilla kollimaattoreilla, jotka vähentävät kuvassa esiintyvää säteilyn sirontaa. Ilmaisimien voi koostua yhdestä tai monirivisestä ilmaisimijärjestelmästä, jolla voi kuvantaa usean (4-64) leikkeen samanaikaisesti. Ilmaisimissa oleva amorfinen seleeni- tai piilevy ionisoituu säteilystä, josta muodostunut sähköinen signaali muunnetaan digitaalseksi ja tallennetaan vastaavan kuva-alkion intensiteettitiedoksi. [16.]



Yksittäistä kaksidimensionaalista kuva-alkiota kutsutaan pikseliksi. Se on kuvan pienin elementti ja sisältää digitaalisen kuvan tiedon neljän muotoisena (kuva 5). Se muodostuu

kuvantamisessa muodostuneista intensiteetti-arvosta ja sijoittuu kuvamatriisissa paikka-kohtaisen (spatiaalisesta) sijainnin, eli kuvantamisen kohteen sijainnin mukaan. Pikselin koko on suoraan kytköksissä kuvan paikkakohtaiseen tarkkuuteen ja yksityiskohtien määrään. Pikselin koko vaihtelee kuvantamismenetelmien välillä, koska niissä käytetään eri kuvakenttä- (FOV, Field of view) ja matriisikokoa. Leikekuvien kuva-alkiot ovat kolmi-dimensionaalisia tilavuusalkioita, vokseleita, joissa otetaan huomioon leikekuvan pak-suus. [16; 20; 27.]



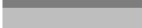



Kuva 5. Kuvassa on 17x17 kokoiset matriisit. Matriisin alkio vastaa tiettyä sijaintia kuvassa (paikkakohtainen sijainti) ja sillä on potilaan kudoksen tiheyttä, intensiteettiä, vastaava arvo bitteinä. [27, s.194.]

Digitaalisessa kuvantamisessa analogiset signaalit muunnetaan ja tallennetaan binäärikoodatuiksi luvuiksi. Jokainen pikseli sisältää informaatiota bitteinä tai kokonaislukuina, joka voidaan esittää binäärilukuna. Pikselin sisältämä binääriluku kertoo kuvan kohdan intensiteetin, josta määritetään kuvan syvyys ja laajuus. Syvyys kertoo, kuinka monta eri arvoa kuva-alkioissa voi esiintyä. Harmaasävytasot ilmaistaan kuvissa binääriluvuilla. Jos käytössä on vain 1 bitti luku 0 vastaa mustaa ja 1 valkoista (kuva 6). Pikselin bittisyvyyden ollessa 8, tuotettujen harmaiden sävyjen määrä on 2 potenssiin bittisyvyys, eli  $2^8$  vastaa 256 harmaasävytasoa. Radiologisissa kuvantamislaitteissa käytetään bittisyvyyttä 8 – 16. Kuvissa on kuitenkin mahdollista käyttää myös värejä, jolloin harmaasävyn kirkkaustasot korvataan keinoväreillä. Värien käytön etuna on, että silmä kykenee erottamaan herkemmin värisävyjen eroja kuin pelkästään harmaasävyjen eroja. [16; 20.]

Binääriluku	10-järjestelmä	Harmaasävy
0	0	
1	1	

Binääriluku	10-järjestelmä	Harmaasävy
0 0	0	
0 1	1	
1 0	2	
1 1	3	

Kuva 6. Neljä eri harmaasävytasoa voidaan esittää käyttämällä 2 bittiä (00, 01, 10, 11), jolloin pikselit saavat arvot 0-3 välillä. [28, s.33.]

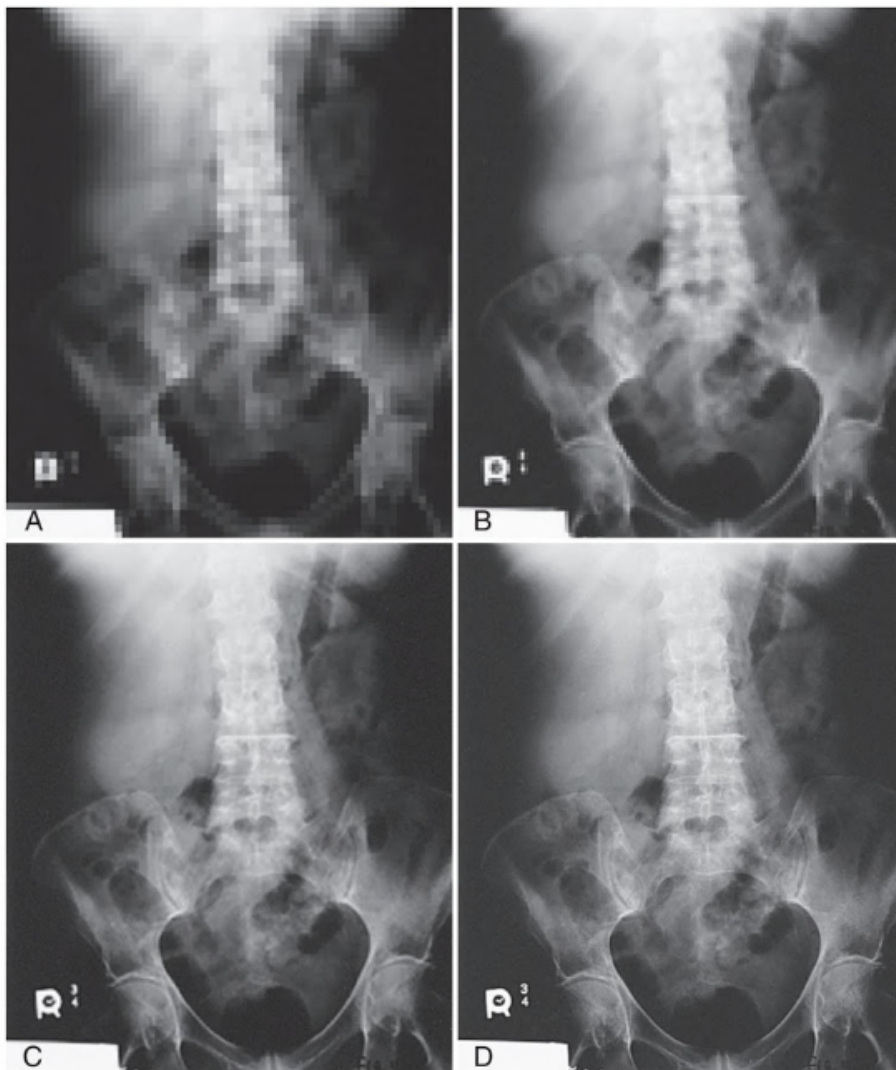
Harmaasävytasojen määrä, eli dynamiikka, on yksi kuvan kontrastin resoluutiota määrittävä tekijä. Kuvan kontrasti viittaa kykyyn näyttää hienovaraisia muutoksia harmaasävyissä, ja se liittyy suoraan kuvassa olevien pikselien bittisyvyyteen. Röntgensäteilyyn perustuvassa kuvantamismenetelmässä jokaisen elimen ja kudostyyppin fyysiset ominaisuudet ja paksuus tuottaa erilaisen arvon riippuen sen kyvystä päästää säteilyä läpi. Kontrastiin vaikuttaa näin ollen potilaan kudosten paksuus ja tiheys, sekä niiden kemiallinen koostumus. Lisäksi siihen vaikuttaa säteilyn muodostavat komponentit, säteilyn määrä ja sen sironta, kuvantamisetäisyys sekä ilmaisimen ominaisuudet. [16; 20.]

Kyky esittää kohteen pieniä yksityiskohtia kutsutaan paikkaerotuskyvyksi. Arvo kertoo, kuinka pieniä yksityiskohtia kuvasta voidaan erottaa. Paikkaerotuskyky resoluutio on riippuvainen pikselikoosta. TT-kuvassa viipaleiden paksuus vaikuttaa paikkaerotuskykyyn. Paksumpi leike sisältää enemmän säteilyä imevää kudosta, jolloin esiintyy vähemmän kohinaa. Tämä johtaa kuitenkin huonoon paikkaerotuskykyyn, koska kaikki kudokset saavat saman harmaasävynarvon, jolloin esimerkiksi luusta on vaikea erottaa hiusmurtumaa. Ohuissa viipalekuvauksissa saavutetaan tarkempi paikkaerotuskyky, mutta saatetaan menettää osa signaalista. [16; 20.]

Sädekeilan ja ilmaisimen kohtaaman alueen koko määrää FOV:n, eli kuvatun alueen koon. Jos FOV:n sivun pituus on 300 mm ja pikseleitä, eli matriisin alkioita, on rivissä 3000 kpl. Tällöin pikselikoko on 300 mm/3000, eli 0,1 mm, jolloin kuvan paikkaerotuskyky on 0,2 lp/mm (line pair per mm). Matriisien koot kuitenkin riippuvat kuvantamistekniikasta. Esimerkiksi TT-kuvantamisessa käytetään tyypillisesti 256x256- ja 512x512-kokoisia matriiseja. Matriisin koko ja tallennetut bitit määräävät kuvan tiedostokoon. Mitä suurempi matriisi ja tallennettujen bittien määrä, sitä suurempi tiedostokoko. Tämä tieto

on tärkeä kuvien arkiston kokoa suunnitella. [16; 20.] Esimerkiksi tuhat 512x512x256-kokoista TT-kuvapakkaa vie pakkaamattomana noin 268 Gt [29].

Kuvassa 7 on nähtävissä sama röntgenkuva saman kokoisella FOV:illa esitettyä eri kokoisilla kuvamatriiseilla ja siitä voi nähdä kuva-alkioiden määrän ja koon vaikutuksen kuvan tarkkuuteen. Kuvassa A:n matriisin koko on 64x64, B:n 215x215, C:n 1024x1024 ja D:n 2048x2048. Kuvan paikkaresoluution kasvattaminen parantaa kuvan erottelukykä. A-matriisista voidaan nähdä, että pienellä matriisilla isolta alueelta otettu kuva jättää helpommin yksityiskohdat piiloon.



Kuva 7. Paikkaresoluution vaikutus kuvan laatuun ja erottelukykäyn samankokoisella FOV-avulla. 64x64-matriisia (A) käytetään isotooppikuvauksissa. 215x215-matriisia (B)

käytetään TT-kuvauksissa. 1024x1024- (C) ja 2048x2048-matriisia (D) käytetään 2D-röntgenissä. [20, s.20.]

Ilmaisimen vastaanottama säteilyn ajallinen määrä ilmoitetaan valotusindeksillä. Liian suuri valotusindeksi voi johtaa tarpeettoman suureen säteilyn määrään. Säteilyn ollessa suuri ihmisen keho päästää enemmän säteilyä lävitse, jolloin ilmaisin on ylivalottunut, eli kuvan tummuus lisääntyy, jolloin anatomisten alueiden rajoja ei kyetä erottamaan. Sama on päinvastoin. Valotusindeksin ollessa liian pieni muodostuu liian vähän säteilyä kehon läpäisemiseksi, mikä johtaa ilmaisimen alivalottumiseen. Valotusajan tulee olla mahdollisimman lyhyt, jotta vältetään liikkeen aiheuttamat häiriöt, kuten rintakehää kuvatessa hengityksestä johtuvat liikeartefaktat. [16; 20.]

#### 2.4 Digitaalisten kuvien käsittely lääketieteellisessä kuvantamisessa

Digitaalisessa röntgenkuvausjärjestelmässä kuvat muutetaan kuvantamislaitteessa digitaaliseksi signaaliksi näytteistykseen avulla. Muunnoksen jälkeen kuvat ovat käytettävissä käsittelyä ja manipulointia varten. Tässä työssä esitellään vain muutama yleisin TT-kuvan käsittelymenetelmä.

Radiologiset kuvat esitetään tavallisesti xy-paikkatasossa, mutta ne voidaan myös muuntaa matemaattisesti paikkatasosta taajuustasoon. Kuvassa olevat tasasävyiset alueet esitetään matalataajuisina ja epätasaisuus, reunat ja viivat esitetään korkeina taajuuksina. Paikkatason kuvankäsittelyssä alkuperäiset kuvamatriisiarvot (pikseli) muokataan matemaattisilla menetelmillä. Käsittelyn tavoitteena on korostaa kuvan ominaisuuksia tai poistaa yksityiskohtia, kuten kuvauslaitteiston tai kuvan rekonstruoinnin tuottamia häiriöitä (artefakteja), joita esimerkiksi ovat kohina tai tähtiartefaktit. Vaihtoehtoisesti voi myös korostaa haluttuja piirteitä kuvasta, kuten korostaa anatomisia reunoja tai poistaa taustan vaihtelua. Esimerkiksi keskiarvosuodatinta käyttäessä pikselin arvo korvataan pikseliarvon ympäristön keskiarvolla. Tämän käyttö johtaa kuitenkin reunojen häviämiseen kuvasta. Reunojen häviäminen voidaan välttää käyttämällä mediaanisuuodatinta, jossa pikseliarvo korvataan sen ympäristön mediaaniarvolla. Mediaanisuuodatuksessa maskilla, esimerkiksi 3x3-matriisilla, poimitaan pikseliä ympäröivien alkiodien arvot. Maskin keskipisteen arvo korvataan pikselin ympäristön harmaasävyarvojen keskimmäiseksi

suurimmalla arvolla, keskiluvulla. Suodattimen käyttö pakottaa huomattavasti poikkeavat pikseliarvot vastaamaan ympäristöään. [16.]

Paikkatason kuva muunnetaan taajuustasoon käyttäen kaksiulotteista Fourier-muunnosta. Kyseinen tapa on tehokas hahmontunnistuksessa ja kuvan terävöittämisessä. Ali-päästösuodattimella matalat taajuudet läpäisevät suodattimen, jolloin korkeat taajuudet suodatetaan pois. Kuvasta tulee pehmennyt, eikä siinä esiinny teräviä yksityiskohtia. Korkeapäästösuodatuksella suodatetaan matalat taajuudet pois, jolloin kuvasta korostuu yksityiskohdat. Siirtofunktiolla kerrotaan taajuustasossa kuvaan kohdistuvan suodatus. [16.]

Yksi tärkeimmistä kuva-analyysin vaiheista on segmentointi, jossa mielenkiintoiset alueet irrotetaan kuvasta jatkokäsittelyyn. Segmentointialgoritmit etsivät esimerkiksi harmaansävyn epäjatkuvuuksia tai sävyiltään yhtenäisiä alueita. Algoritmit käyttävät pääasiassa spatiaalitason maskeja. Yksi tärkeimpiä segmentointitapoja on kynnystäminen, jossa intensiteetiltään samanlaiset alueet rajataan yhtenäiseksi alueeksi.

Kuvafuusiossa yhdistetään kahdella kuvantamistekniikalla tuotetut kuvat lisäinformaation etsimiseksi. Tämän onnistumiseksi kuvien kuva-alueiden tulee olla identtiset ja kuvien paikkatietojen on vastattava toisiaan. Kuvat skaalataan ja asemoidaan kohdakkain näkyvien anatomisten maamerkkien tai ulkoisten merkintöjen avulla. [16.]

Kuvien annotointi, eli merkintä, on tapa lisätä kuvaan lisätietoja tavallisen tunnistuksen lisäksi. Kuvaan lisätään merkintätoiminnolla (annotointifunktio) tietoa esiasetettujen termien ja/tai manuaalisen tekstinsyötön valinnalla. Annotointikerros peittää kuvan bittikarttakuvana. Esimerkiksi jos kuvaan rajataan alue värillä, sitä ei tallenneta kuvan pikseli tietoihin, vaan erillisenä kuvamatriisina. [29.]

## 2.5 Lainsäädäntö ja ohjeistus

Tekoälyn kehitys ja käyttöönotto kliinisessä ympäristössä on Suomessa haastavaa lainsäädännön vuoksi. 25.5.2018 astui voimaan EU:n tietosuojasetuksen päivitys (EU 2016/679), eli GDPR (General Data Protection Regulation), jonka pohjalta säädettiin kansallinen tietosuojalaki (1050/2018). Lailla säädetään yleisesti henkilötietojen

keräämisestä ja käsittelystä. EU:n tietosuoja-asetuksen 9. artikla, erityisiä henkilötietoryhmiä koskeva käsittely, määrää, että:

Sellaisten henkilötietojen käsittely, joista ilmenee rotu tai etninen alkuperä, poliittisia mielipiteitä, uskonnollinen tai filosofinen vakaumus tai ammattiliiton jäsenyys sekä geneettisten tai biometrinen tietojen käsittely henkilön yksiselitteistä tunnistamista varten tai terveyttä koskevien tietojen taikka luonnollisen henkilön seksuaalista käyttäytymistä ja suuntautumista koskevien tietojen käsittely on kiellettyä.

Artiklassa on määritelty poikkeustapaukset, jotka mahdollistavat terveyttä koskevien tietojen käsittelemisen ja sitä myötä mm. sairaanhoidon järjestämisen ja toteuttamisen [30]. Potilastietoihin sovelletaan lisäksi potilaan asemaa ja oikeuksia käsittelevän lain (785/1992) 13 § (653/2000), jossa määrätään potilasasiakirjoihin sisältyvien tietojen salassapidosta [31]. Asiakkaalle tai potilaalle on aina kerrottava, mihin hänen antamiaan tietoja käytetään, mistä muualta tietoja mahdollisesti hankitaan, mihin tiedot tallennetaan ja millä ehdoin niitä luovutetaan ja kenelle. Tallennettuihin tietoihin tutustuminen kuuluu asiakkaan tai potilaan oikeuksiin ja hän voi aina pyytää virheellisten tietojen korjaamista sellaisia löytäessään.

Potilastietojen käsittelyyn vaikuttaa lisäksi laki sosiaali- ja terveydenhuollon asiakirjojen sähköisestä käsittelystä (159/2007) [32]. Sen tarkoituksena on taata potilastietojen turvallinen säilyttäminen, sisällön asianmukainen suojaaminen sekä varmistaa potilasasiakirjojen käyttö ja luovuttaminen. Potilastietojen luovutus perustuu aina lakiin tai potilaan itsensä antamaan suostumukseen. Pääsääntöisesti tietoja luovutetaan ainoastaan terveydenhuollon eri toimijoiden kesken potilaan terveydenhoitoon liittyvissä asioissa. Vuonna 2011 voimaan tulleen terveydenhuoltolain (1326/2010) [33] myötä kunnallisen perusterveydenhuollon ja erikoissairanhoidon potilasasiakirjat muodostavat terveydenhuollon yhteisen potilastietorekisterin, josta potilasta hoitava henkilökunta voi käyttää hoidon kannalta tarpeellisia tietoja ilman potilaan suostumusta. [34.]

Tietojen luovutusta tieteelliseen käyttöön on ohjannut laki viranomaisen toiminnan julkisuudesta (621/1999) [35] ja se sallii 28 § mukaisesti yksittäistapauksissa tietojen luovutuksen tieteellistä tutkimusta, tilastointia sekä viranomaisten suunnittelu- tai selvitystyötä varten. Tieteellistä tutkimuslupaa varten tarvitaan hyvin perusteltu tutkimuslupa, jonka saaminen voi kestää usean vuoden. Maaliskuussa 2019 eduskunta hyväksyi täysistunossaan toisiolain, eli lain sosiaali- ja terveystietojen toissijaisesta käytöstä [36].



Toisilain tavoitteena on purkaa lupakäsittelyyn liittyvää päällekkäistä hallinnollista työtä, nopeuttaa lupakäsittelyä ja sujuvoittaa tietojen yhdistelyä eri rekistereiden välillä. Sote-aineistot ovat uuden lain myötä helpommin tieteellisen tutkimuksen, tilastoinnin, kehittämis- ja innovaatiotoiminnan, opetuksen, tietojohdamisen ja erilaisten viranomaisten vaatimien toimien käytettävissä. Lupien myöntämisestä vastaa tietolupaviranomainen, joka toimii Terveyden ja hyvinvoinnin laitoksen (THL) yhteydessä, mutta kuitenkin erillään sen muusta toiminnasta.

Tekoälyä varten ei ole olemassa erillistä lainsäädäntöä, vaan siihen sovelletaan voimassa olevia lakipykälää tapauskohtaisesti. EU-tasolla on alettu kuitenkin työstää eettistä ohjenuoraa luotettavan tekoälyn rakentamisen avuksi. Sen tarkoitus on edistää ihmiskesteistä tekoälykehitystä, jossa teknologian luotettavuus on keskeisessä roolissa. Luotettavuuden lisäämiseksi raportti painottaa erityisesti tulosten selitettävyyttä, ja ottaa myös kantaa opetusdatan puolueettomuuteen. Dokumentin luonnos julkaistiin asiantuntijoiden arvioitavaksi joulukuussa 2018 ja tämän osion kirjoittamisen aikaan palaute oli vielä analysoitavana, eikä sen tuomia mahdollisia muutoksia ei ole otettu huomioon. [37.]

Ensimmäisessä luonnoksessa Euroopan komission tekoälyä käsittelevä korkean tason asiantuntijaryhmä (European Commission's High-Level Expert Group on Artificial Intelligence eli AI HLEG) on määritellyt luotettavalle tekoälylle kaksi osa-aluetta:

1. "Eettisen tarkoituksperän" varmistaminen perusoikeuksia, sovellettavaa lainsäädäntöä sekä keskeisiä periaatteita ja arvoja kunnioittaen.
2. Tekninen luotettavuus.

Asiakirjan tarkoituksena on ohjeistaa, miten tekoälyn keskeisiä arvoja ja periaatteita saadaan otettua käyttöön tekoälyjärjestelmissä ja siitä, miten niitä kannattaa soveltaa kehityksen eri vaiheissa. Se ottaa kantaa tekoälyn eettisen tarkoituksperän varmistamiseen ja ohjeistaa tekoälyn käyttöönottoa siten, että siinä otetaan huomioon tekoälyn vaatimukset sekä tekniset ja muut ominaisuudet. [37, s.2.]

Suunnittelun tueksi on laadittu kymmenenkohtainen lista, jonka tarkoitus on auttaa huomioimaan paremmin luotettavaa tekoälyä koskevat vaatimukset. Se pitää sisällään seuraavat kohdat:

- vastuuvollisuus
- datanhallinta
- kaikkia palveleva suunnittelu
- tekoälyn autonomian hallinta (ihmisen suorittama valvonta)
- syrjimättömyys
- ihmisen itsemääräämisoikeuden kunnioittaminen
- yksityisyyden kunnioittaminen
- luotettavuus
- turvallisuus
- läpinäkyvyys. [37, s.14.]

Luetteloa on tarkoitus mukauttaa jokaiseen eri käyttötarkoitukseen sopivaksi, eikä sitä ole tarkoitus käyttää tarkistuslistana sovellusta luodessa. Eettiset näkökulmat tulisi kuitenkin ottaa huomioon jokaista tekoälysovellusta kehittäessä.

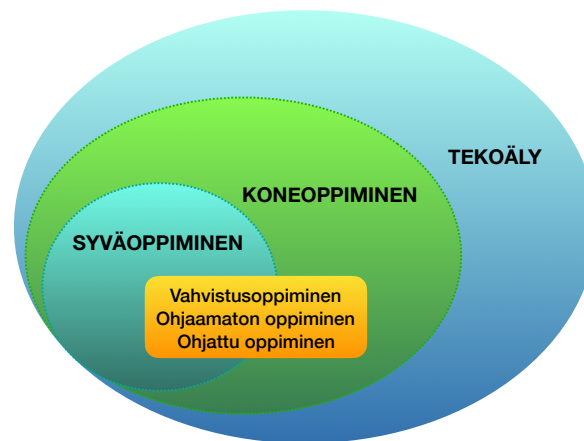
### **3 Kuvien syväoppiminen tekoälykehityksessä**

#### **3.1 Tekoäly, koneoppiminen ja syväoppiminen**

Ajatus tekoälyn rakentamisesta syntyi toisen maailmansodan jälkeen, ja se sai nykyisen nimensä vuonna 1956. Koneoppimisen kehitys lähti käyntiin 1960-luvun alkupuolella, mutta ala koki takaiskun 70-luvun puolivälissä, kun tietokoneiden suorituskyky ei ollut riittävä käsittelemään teknologian vaatimia suuria datamääriä. Valtiot ja rahoittajat menettivät uskonsa tekoälyyn, kunnes 1980-luvulla kehitys lähti uuteen nousuun neuroverkkojen siivittämänä. Into laantui vielä hetkellisesti 90-luvulla, kunnes 2000-lukua lähestyessä ala lähti kolmannen kerran nousuun. Suurina virstanpylväinä voidaan pitää

esimerkiksi IBM:n Deep Bluen voittoa shakin maailmanmestaria Garry Kasparovia vastaan 1997, IBM Watsonin voittoa amerikkalaisessa tietokilpailu Jeopardyssa 2011 sekä sitä, kun Google Brain oppi tunnistamaan kissan valokuvasta 2012. [38; 39.]

Tekoäly (Artificial Intelligence eli AI) on terminä suurimmalle osalle tuttu, mutta sen määrittely ei ole aivan yksiselitteistä. MIT:n (Massachusetts Institute of Technology) professori Thomas Malosen määrittelyn mukaan sillä tarkoitetaan koneita, jotka käyttäytyvät älykkäästi vaikuttavalla tavalla [40]. Yleisesti termi kattaa kokonaisen tieteenalan, johon kuuluvat useat tekoälyn kehittämismenetelmät (kuva 8), kuten koneoppiminen (machine learning), sen osa-alueena neuroverkkoja hyödyntävä syväoppiminen (deep learning) sekä niihin liittyen omina koneoppimistapoinaan ohjattu oppiminen (supervised learning), ohjaamaton oppiminen (unsupervised learning) ja vahvistusoppiminen (reinforcement learning).



Kuva 8. Tekoäly pitää sisällään koneoppimisen ja syväoppimisen sekä niiden erilaiset oppimismenetelmät.

Karkeasti tekoäly voidaan jakaa kahteen kategoriaan – heikkoon ja vahvaan. Heikko tekoäly kykenee suorittamaan rajallisesti yksinkertaisia tehtäviä, kuten pelaamaan shakkia, tunnistamaan puhetta tai analysoimaan kuvia. Kaikki nykyiset tekoälysovellukset kuuluvat tähän kategoriaan. Vahvaa tekoälyä ei ole vielä onnistuttu rakentamaan, eivätkä asiantuntijat ole yksimielisiä edes siitä, onko sellainen ylipäänsä mahdollista. Vahvan tekoälyn tulisi olla aidosti älykäs ja sillä pitäisi olla jonkintasoinen tietoisuus, eli sen pitäisi toimia kuten ihminen. [41; 42.]

Vuonna 1950 brittiläinen matemaatikko Alan Turing kehitti testin, jotta pystyttäisiin arvioimaan koneiden älykkyyttä. Kokeessa ihminen käy tekstipohjaisesti keskustelua sekä koneen että ihmisen kanssa. Mikäli testaaja ei kykene erottamaan, onko vastakkaisella puolella ihminen vai kone, voidaan kone luokitella älykkääksi. Vaikka testillä on omat puutteensa ja se on saanut osakseen paljon kritiikkiä, se ottaa kattavasti huomioon tekoälyn eri osa-alueet, jotka koneen tulisi hallita läpäistäkseen testin. Näitä ovat muun muassa luontaisen kielen prosessointi (NLP, Natural Language Processing) ja koneoppiminen (ML, Machine Learning). [43.]

Nykyiset tekoälysovellukset perustuvat suurimmalta osin koneoppimiseen. Se tarkoittaa sitä, että kone oppii tekemään päätöksiä käsittelemänsä datan perusteella, eikä jokaista sääntöä tarvitse kirjoittaa käsin. Kuten aiemmin todettiin, koneoppiminen voidaan jakaa karkeasti kolmeen alakategoriaan, jotka ovat ohjattu oppiminen, ohjaamaton oppiminen ja vahvistusoppiminen. [44.]

Ohjattu oppiminen on menetelmä, jossa koneelle syötetään valmiiksi luokiteltua tai muuten luonnehdittua opetusdataa. Tämän perusteella koneoppimisella opetettu tekoäly osaa antaa syötedatalle luokittelun (esim. ”kuvassa on kissa”) tai muun luonnehdinnan (esim. ”alueella x olevan talon myyntihinta tulee olemaan y euroa”). Jos tekoälyn halutaan oppivan tunnistamaan kuvista kissoja ja koiria täytyy kummallekin kategorialle manuaalisesti kerätä riittävän suuri ja monipuolinen otanta kuvia, joilla mallin voi opettaa. Mitä enemmän opetusluokat sisältävät eri ikäisiä ja rotuisia yksilöitä sekä eri kuvakulmista ja tilanteista otettuja kuvia, sen paremmaksi malli muuttuu ja mitä useampiin kategorioihin kuvia jakaa, sen tarkemmin malli pystyy kuvassa olevan kohteen tunnistamaan.

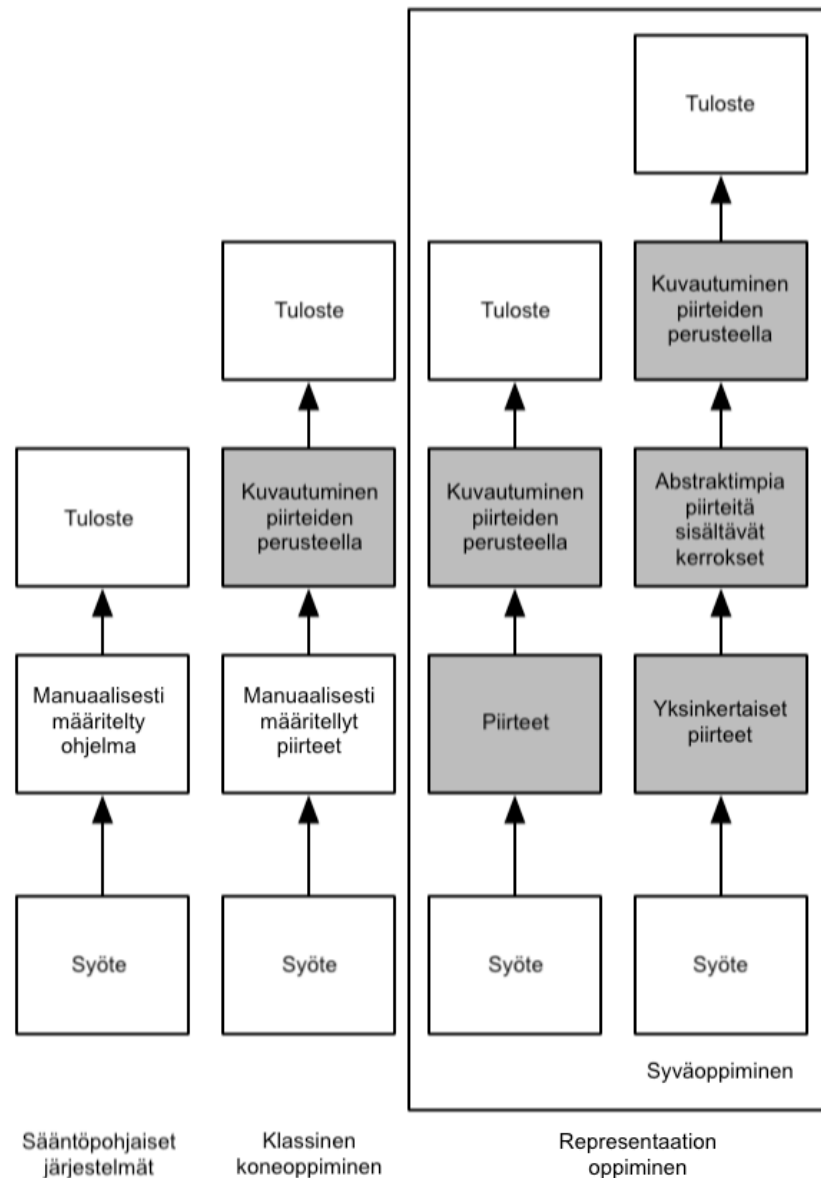
Ohjaamattomassa oppimisessä koneelle ei anneta valmiita vastauksia, vaan se päätelee asioita datasta löytyvien säännönmukaisuuksien perusteella. Tätä ominaisuutta käyttävät esimerkiksi verkkokaupat sekä elokuva- ja musiikkipalvelut, jotka suosittelevat käyttäjilleen seuraavaa sopivaa tuotetta aiempaan ostohistoriaan perustuen. Tällä tavoin palveluntarjoajat pyrkivät parantamaan käyttökokemusta ja ohjaamaan käyttäjän toimintaa.

Vahvistusoppimisella tarkoitetaan sitä, että kone oppii sille määrätyssä ympäristössä tekemistään havainnoista ja muokkaa toimintatapojaan saamansa palautteen perusteella. Se

pyrkii lisäämään toimintaa, josta saatu palaute on ollut positiivista ja välttelee niitä, josta palaute on ollut negatiivista. Tunnetuimpana esimerkkinä voidaan pitää itseajavia autoja.

Jokaiseen oppimistyyliin on valittavissa useita eri algoritmeja. Valinta riippuu muun muassa siitä, kuinka paljon dataa on käytettävissä, minkä kokoista ja tyyppistä se on, tallennustilan tarpeesta ja ratkaisulle asetetuista vaatimuksista. Algoritmillä tarkoitetaan koneoppimisen taustalla toimivaa logiikkaa ja matemaattisia laskutoimenpiteitä, jotka sisältävät usein esimerkiksi todennäköisyyslaskentaa. [45.]

Koneoppimisen keskeinen ongelma on se, miten data käsitellään sille parhaiten soveltuvalla tavalla. Representaatioita oppiva malli kykenee etsimään parhaimman käsittelytavan suoraan datasta löytyvien piirteiden perusteella, eikä se vaadi sääntöpohjaisten järjestelmien tapaan jokaisen piirteen opettamista manuaalisesti (kuva 9). Representaatiolla tarkoitetaan oppimismalleja, eli tapaa, jolla tieto jäsentyy muistiin. Mikäli käsittelytapoja on useampia, muodostaa jokainen niistä oman representaatiokerroksen. Esimerkiksi valokuvaa voidaan tarkastella RGB- tai HSV-formaatissa. RGB on värimalli, jonka nimi on lyhenne sanoista red-green-blue, ja se määrittelee värit sen mukaan, kuinka paljon ne sisältävät punaista, sinistä ja vihreää väriä. HSV on lyhenne sanoista hue-saturation-value, ja se puolestaan määrittää värit sävyn, kylläisyyden sekä kirkkauden perusteella. Jos kukkaketoa esittävästä kuvasta halutaan löytää kaikki punaiset unikat, on kuvaa parempi käsitellä RGB-formaatilla. Kylläisyyden korjaaminen kuvasta sen sijaan onnistuu paremmin HSV:n avulla. [46.]

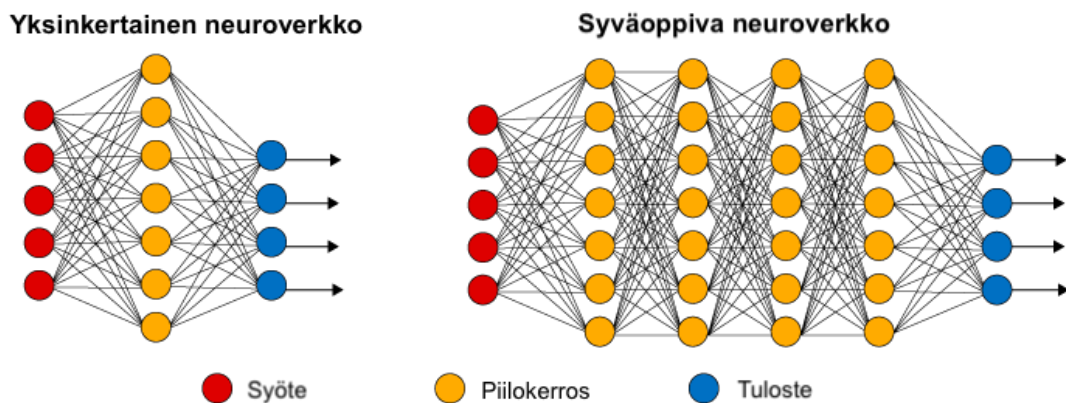


Kuva 9. Eri koneoppimismallien rakenteita. Harmaalla värjätyt alueet ovat komponentteja, jotka kykenevät oppimaan asioita suoraan käsiteltävästä datasta. [47, s. 10]

Opitut representaatiot johtavat usein parempaan suorituskyykyyn kuin mitä voi saavuttaa manuaalisesti määrittelyillä representaatioilla. Ne mahdollistavat tekoälyn mukautumisen uusiin tehtäviin ilman, että ihmisen tarvitsee juuri puuttua asiaan. Representaation oppiva algoritmi voi löytää yksinkertaisesta tehtävästä kattavan joukon piirteitä minuu-teissa, monimutkaisemmassa tehtävässä se voi viedä tunneista kuukausiin. Piirteiden manuaalinen suunnittelu monimutkaiseen tehtävään vaatii paljon aikaa ja vaivaa. Se voi viedä kokonaiselta tutkijayhteisöltä vuosikymmeniä. [47.]

Syväoppiminen on erityinen koneoppimisen osa-alue. Siinä missä koneoppimisen muut mallit käyttävät usein yhtä tai kahta esityskerrosta, syväoppiva verkko voi sisältää kymmeniä tai jopa satoja peräkkäisiä esityskerroksia. Mallin syvyyden määrittää esityskerrosten lukumäärä. Lähes kaikki syväoppiminen tehdään neuroverkoksi kutsutun mallin avulla, jossa esityskerrokset rakennetaan kerroksittain toistensa päälle. [46.] Syväoppimisen prosessi esitellään yksityiskohtaisemmin seuraavassa kappaleessa.

Keinotekoiset neuroverkot (Artificial Neural Networks eli ANN) rakentuvat syötekerroksesta, piilokerroksista ja tulostekerroksesta. Ne sisältävät suuren määrän toisiinsa kytkettyjä prosessointielementtejä, neuroneita, jotka voivat vastaanottaa yksinkertaista syöttödataa, suorittaa sille yksinkertaisia operaatioita ja välittää tulokset eteenpäin toisille neuroneille. [48.] Täysin yhdistetyssä (fully connected) neuroverkossa jokainen piilokerroksessa oleva neuron on kytköksissä kaikkiin viereisten kerrosten neuroneihin (kuva 10).



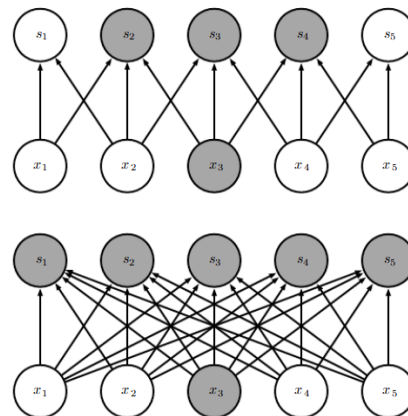
Kuva 10. Täysin yhdistetty yksinkertainen ja syväoppiva neuroverkko. [49]

Neuroverkkoja on kahta tyyppiä. Eteenpäin syöttävät (Feed Forward) verkot kuljettavat informaatiota ainoastaan yhteen suuntaan, ja se soveltuu mallien generointiin, tunnistamiseen ja luokitteluun. Takaisinkytketyssä (Feedback) neuroverkossa on paluusi-lmukka, joka kykenee syöttämään informaatiota myös takaisinpäin verkossa ja se soveltuu käytettäväksi muun muassa ennustaviin malleihin. [48.]

Kuvien käsittelyssä käytetään yleisimmin eteenpäin syöttäviä konvoluutioneuroverkkoja (Convolutional Neural Network eli CNN), sillä ne soveltuvat erityisen hyvin visuaalisen datan luokitteluun ja analysointiin. Matemaattisesti konvoluutiolla tarkoitetaan

integraalia, joka ilmaisee funktion  $g$  päällekkäisyyden määrän, kun se siirtyy toisen funktion  $f$  yli ja muodostaa niistä siten yhdistetyn funktion. Konvoluution avulla voidaan toteuttaa suodattimia (kerneleitä), joiden avulla kuvista voidaan etsiä esimerkiksi tiettyjä muotoja, reunoja, tekstuureja tai värejä. [50.]

Suurin ero täysin yhdistetyn kerroksen ja konvoluutiokerroksen välillä (kuva 11) on se, että konvoluutiokerros oppii paikallisia hahmoja sen sijaan, että se kävisi läpi kuvan kaikki pikselit ja käsittelisi sitä yhtenä hahmona. Menetelmän merkittävimpiä etuja on se, että opittuaan jonkin hahmon, CNN kykenee tunnistamaan sen mistä tahansa kohdasta kuvassa. Täysin yhdistetty kerros oppii sen sijaan hahmon piirteet kokonaan uudestaan, mikäli kuvassa tapahtuu muutoksia. Tämän vuoksi konvoluutioverkot tarvitsevat vähemmän opetusdataa. Lisäksi konvoluutiokerrokset voivat oppia hahmojen alueellisia hierarkioita. Ensimmäinen kerros voi oppia esimerkiksi reunoja, toinen kerros suurempia kokonaisuuksia ensimmäisen kerroksen ominaisuuksien perusteella ja niin edelleen. Tämän ansiosta konvoluutioverkko oppii tehokkaasti monimutkaisia ja abstrakteja visuaalisia kokonaisuuksia. [46.]

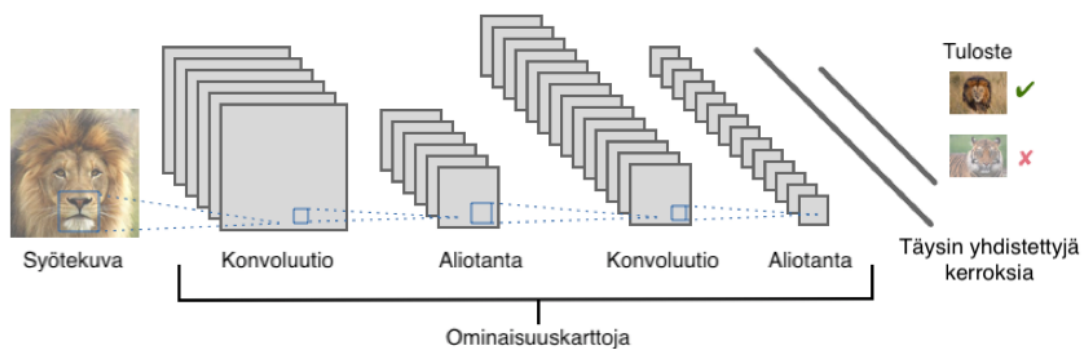


Kuva 11. Täysin yhdistetyn kerroksen (alhaalla) ja konvoluutiokerroksen (ylhäällä) neuronit yhdistyvät toisiinsa eri tavoilla. [47, s. 331]

Neuroverkot koostuvat konvoluutio-, aliotanta- (subsampling layer) sekä yhdestä tai useammasta täysin yhdistetystä kerroksesta (kuva 12). Konvoluutiokerros koostuu ominaisuskartoista, jotka sisältävät jonkin kuvion sijainnin kuvassa. Resoluutiota pienennetään aliotantakerroksessa maksimikeräysmenetelmällä (max pooling), mikä vähentää laskentaa myöhemmissä kerroksissa. Tästä huolimatta konvoluutioneuroverkot



tarvitsevat huomattavan suuria määriä laskentatehoa ja muistia ja siksi niitä opetetaan tietokonejärjestelmissä, joissa on paljon rinnakkaisia laskentayksiköitä (computational core). Verrattuna perinteiseen keskusprosessorilaskentaan (CPU, Central Processing Unit) rinnakkaisia laskentayksiköitä on erityisesti grafiikkaprosessoreissa (GPU, Graphics Processing Unit). Riittävän laskentatehon saavuttamiseksi neuroverkko-opetusta tehdään myös tietokonejärjestelmien klustereissa. Klusterilla tarkoitetaan toisiinsa kytköksissä olevien tietokoneiden joukkoa, joista yksi toimii palvelimena, joka jakaa tehtäviä muiden verkossa olevien koneiden kesken, jolloin suurtakin laskentatehoa vaativat prosessit voidaan suorittaa lyhyemmässä ajassa. [51; 52.]



Kuva 12. Konvoluutioneuroverkon ominaisuuskartat koostuvat konvoluutio- ja aliotantakerroksista, joiden jälkeen on yksi tai useampi täysin yhdistetty kerros [53]

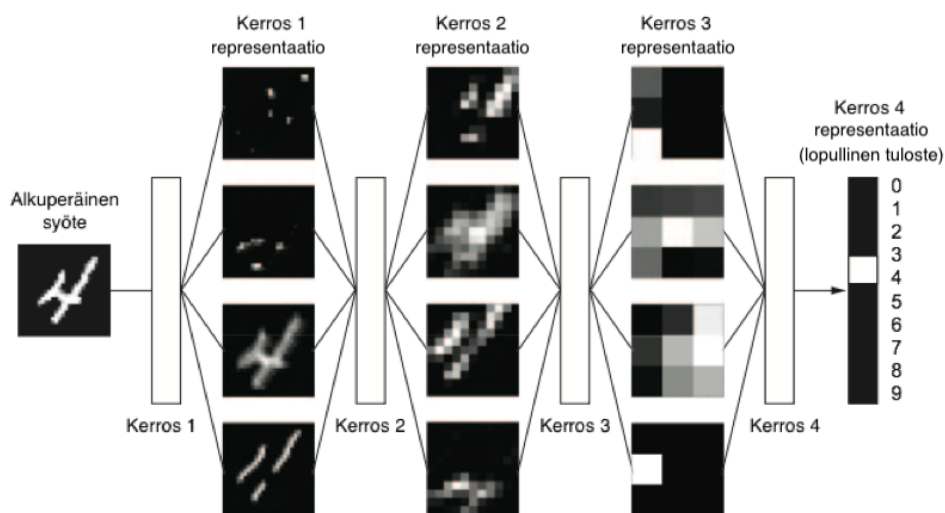
Tässä työssä kuvien käsittelyä varten luodaan eteenpäin syöttävä konvoluutioneuroverkko, jossa käytetään opetusmenetelmänä ohjattua oppimista. Seuraavassa luvussa kuvataan tarkemmin syväoppimisen prosessia.

### 3.2 Syväoppiminen prosessina

Syväoppimisella tarkoitetaan matemaattista kehystä tietojen esitysten oppimiselle. Syväoppiminen ei viittaa oppimisessa minkäänlaiseen syvempään ymmärrykseen, vaan tarkoittaa ajatusta peräkkäisistä/päällekkäisistä esityskerroksista. Esityskerrokset yhdessä muodostavat neuroverkkomallin, jonka kerrosten määrä kertoo sen syvyydestä. Kuitenkin yksi neuroverkkomalli voi koostua monesta eri neuroverkosta. Neuroverkko on terminä viittaus neurobiologiaan, josta neuroverkkoalgoritmien kehittäminen sai alkunsa

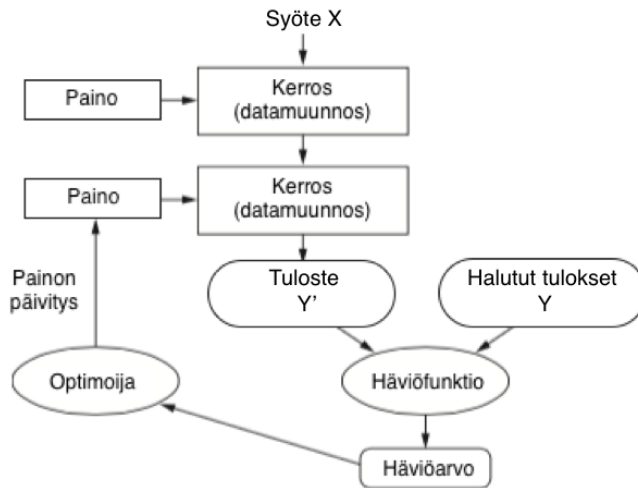
ja myös myöhemmin innoitusta, mutta neuroverkkomallit eivät ole aivojen malleja eivätkä myöskään toteuta läheskään samoja oppimismekanismeja kuin aivot. [46.]

Kuvassa 13 on esitetty esimerkkinä numeroluokitus-neuroverkkomalli, jossa on neljä representaatiokerrosta. Neuroverkko suorittaa datamuunnoksen numerokuvasta poimituille hahmoille jokaisessa representaatiokerroksessa. Muunnoksien myötä representaatiot eroava entistä enemmän alkuperäisestä kuvasta, mutta ne antavat lopulta hyvin tarkkaa tietoa tuloksen saamiseksi. Chollet Francois kuvaa kirjassaan *”Deep learning with Python”* prosessia monivaiheiseksi tiedon tislusoperaatioksi, jossa tiedot kulkevat peräkkäisten suodattimien läpi ja tulevat siten yhä enemmän puhdistetuksi, eli hyödyllisiksi jonkin tehtävän suhteen. [46, s.9.]



Kuva 13. Numeroluokitusmallin representaatiokerroksissa malli etsii kuvasta hahmoja löytääkseen sille oikean luokittelutuloksen. [46, s.9]

Kuvassa 14 on esitetty takaisinkytketyn neuroverkon harjoitussilmukka. Oppimisen alussa verkon kerrosten painot on määritetty esimerkiksi satunnaisarvoiksi. Painoarvot määrittävät, mitä kerros tekee syötetylle datalle. Opetuksen aluksi verkon tuottama ennuste on kaukana halutusta. Neuroverkko toimii pitkälti matemaattisen funktion tavoin. Häviöfunktio (Loss funktion) muodostuu vertaamalla ennustetta ja oikeita arvoja, josta lasketaan häviöarvo (Loss score). Harjoitussilmukkaa ajetaan häviöarvon minimointiin saakka. Neuroverkon ennustuksen parantamisen vuoksi sen toimivuutta on kyettävä tarkkailemaan.



Kuva 14. Neuroverkon päivittyminen painon avulla. [46, s.58.]

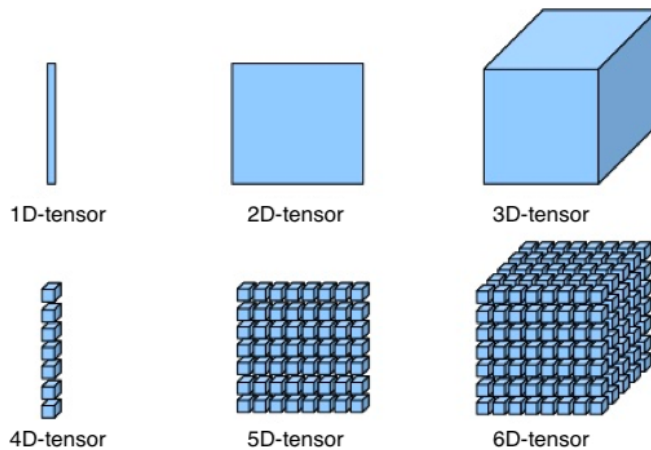
Neuroverkon opetuksen alussa määritetään haluttujen kerrosten määrä ja häviöfunktio, joka määrittelee oppimiseen käytetyn palautesignaalin sekä optimoija, joka määrittää, miten oppiminen etenee. Lisäksi valitaan opetusmateriaali tarkoituksen mukaisesti, esimerkiksi kuvaluokittelua varten tarvitaan kuvadatan lisäksi niitä vastaavat luokittelutiedot. Syötetiedot  $X$  muodostuvat neuroverkon harjoitustietojen joukosta, esimerkiksi harjoituskuville ja niiden luokittelutiedoista sekä validointi ja/tai testitietojen joukosta, joilla testataan opetetun verkon lopullista kyvykkyyttä opetuksessa käyttämättömällä datalla (riippumaton testaus ja verkon yleistämiskyky). Opetuksessa neuroverkko antaa harjoituskuville  $X$  tulosteen  $Y'$ , esimerkiksi luokittelun, jota verrataan harjoituskuvien oikeaan luokittelu tulokseen  $Y$ . Häviöfunktio laskee eron halutun ja neuroverkon antaman tulosteen välille ja eroa hyödynnetään verkon painojen päivityksessä.

Syväoppimisen tärkein osa on käyttää häviöfunktioista saatua häviöarvoa takaisinkytkentäsignaalina. Optimoija hyödyntää tätä tietoa neuroverkon painojen arvon säätämiseksi ja pyrkii minimoimaan saatua häviöarvoa. Tarkemmin sanottuna optimoija määrittää tarkan menetelmän, jossa häviöfunktion avulla määritetään gradientti, eli mihin suuntaan funktion täytyy liikkua häviöarvon minimoimiseksi. Painoja päivitetään jokaisen annetun syötön  $X$  mukaan. Päivityksestä muodostuu harjoitussilmukka, jota toistamalla riittävän monta kertaa tuottaa painoarvot, jotka minimoivat häviöfunktion. Harjoitussilmukassa muodostuu suuri määrä laskusuorituksia, joten on tehokkainta ajaa niitä rinnakkaisesti. Tämä voidaan toteuttaa GPU:n tuhansilla rinnakkaisilla laskentayksiköillä.

Käytännön toteutuksissa neuroverkon jokainen kerros on tietojenkäsittelymoduuli, joka ottaa syötteen yhden tai useamman tietoryhmän, jota kutsutaan tensoriksi, ja antaa ulostulona yhden tai useamman tensorin. Tensoriin tieto on tallennettu moniulotteisiksi numerojoukoiksi. Esimerkiksi matriisit ovat 2D-tensoreja. Tensorit ovat matriisien yleistäminen mielivaltaiseen määrään ulottuvuuksia. Tietokone ei käsittele kuvia kuvana vaan muuntaa ne numerojoukoksi. Tensori määritellään kolmella ominaisuudella:

- Akselien lukumäärä, esimerkiksi 2D-tensorissa on kaksi akselia ja 3D-tensorissa on kolme akselia (kuva 15).
- Muoto on kokonaisluku, joka kuvaa kuinka monta ulottuvuutta on kullakin akselilla, esimerkiksi matriisi, joka on muotoa 2x4, 2 riviä ja 4 saraketta on muotoa (2,4).
- Tietotyyppi kertoo tensorissa olevien tietojen tyyppin, joka voi esimerkiksi olla liukuluku (float32, float64) tai kokonaisluku (uint8). [46.]

Jotkin neuroverkon kerrokset ovat tilattomia, mutta useammin kerroksilla on tila eli kerrospaino, joka on muodostettu yhdestä tai useammasta tensorista stokastisen gradienttimenetelmän mukaan. Stokastisessa gradienttimenetelmässä parametreja korjataan jokaisen syötteen jälkeen. Tilat yhdessä sisältävät verkon tietämyksen. Syy kerrosten erilaisuuteen on, että ne sopivat erilaisille tensorimuodoille ja erilaisille tietojenkäsittelylle. Esimerkiksi 2D-tensori (2 Dimensiota, 2 akselia, matriisin rivi ja sarake) sisältää vektoritietoja, kuten näytteiden lukumäärän ja niiden ominaisuudet, jotka käsitellään usein tiheästi yhdistetyillä kerroksilla. Sekvenssitiedot tai aikakausien tiedot tallennetaan 3D-tensoreihin, joka sisältää näytteiden lukumäärän ja niiden aikaleiman sekä ominaisuudet. Ne käsitellään tyypillisesti toistuvilla kerroksilla, LSTM-kerroksella (Long short-term memory). Kuvat tallentuvat 4D-tensoreihin, jotka käsitellään yleensä 2D-konvoluutiokerroksilla (Conv2D). Ne sisältävät tiedon näytteiden lukumäärästä, kuvan korkeuden ja leveyden sekä värikanavien määrän (järjestys voi vaihdella). On olemassa myös 1D-tensori, joka muodostuu yhdestä vektorista eli numeroiden joukosta, joten sillä on vain yksi akseli. Syväoppimisessa käsitellään yleensä 0D-4D-tensoreita. 5D-tensoreita käytetään, jos käsitellään videotiedostoja. Niihin tallentuvat kuvantiedot: näytteiden (kuvien) lukumäärä, kehykset, kuvan korkeus ja leveys sekä värikanavien määrä (järjestys voi vaihdella). Esimerkiksi pakkaamalla 2D-tensoreita saadaan 3D-tensoreita, jolloin matriiseista muodostuu kuutioita ja toisinpäin. [46.]



Kuva 15. 1D-tensori muodostuu vektorista, 2D-tensori on matriisi ja 3D-tensori on monesta matriisistä koostuva kuutio. 4D-tensorissa kuutioita on vektorissa, 5D-tensorissa kuutiot ovat matriisi muodossa ja 6D-tensorissa kuutiot ovat kuution muodossa. [54.]

Harmaasävyisiä kuvia on mahdollista tallentaa 2D-tensoreihin, sillä ne sisältävät vain yhden värikanavan. Kuitenkin kuvatensorit ovat aina 3D-tensorimuotoa ja niissä on yksilotteinen värikanava harmaasävyisiä kuvia varten. Käyttäessä TensorFlow'n koneoppimisen puitteita 128 harmaasävykuvaan, joiden koko on 256x256, tallennetaan muotoon (näytteiden lukumäärä, kuvan korkeus, kuvan leveys ja värisyvyys) eli (128, 256, 256, 1). Jos kyseessä olisi värikuva käytössä olisi kolme värikanavaa, jolloin vaihdettaisiin 1 tilalle numero 3. [46.]

### 3.3 Syväoppimisen alustavaatimukset

Tekoälyn hyödynnettävyyden kannalta on tärkeä valita oikeanlainen teknologia, joka on yhdistettävissä tarvittaviin syväoppimisen viitekehyksiin. On olemassa erilaisia syväoppimiseen luotuja ohjelmia, jotka palvelevat eri tarkoituksia. Oikean ohjelman löytäminen määrää, kuinka nopeasti ja vaivattomasti tekoälyn kehitys sekä käyttöönotto tapahtuu. Valintaa helpottaa tieto, mihin haasteeseen haluaa tuoda tekoälyä, ja vertaamalla eri ratkaisujen etuja ja haittoja. Samalla on tarkistettava niiden rajoitukset ja tutustuttava kunkin ratkaisun parhaaseen käyttöön. Suurin osa syväoppimisen ohjelmista toimii Linux-, macOS- ja Windows-alustoilla. [55; 56.]

Python on tehokas, monipuolinen ja tulkettava matalan kynnyksen ohjelmointikieli. Siinä on tehokkaita korkean tason tietorakenteita ja yksinkertainen, mutta tehokas lähestymistapa olio-ohjelmointiin. Sitä sovelletaan paljon koneoppimisessa, koska se käyttää useita kirjastoja, joita hyödynnetään syväoppimisessa ja datan käsittelyssä. Python-koodia on helppo ymmärtää, ja siksi se on suosittu kieli koneoppimismallien rakentamiseen vasta-alkajista ammattitason ohjelmoijiin. Tekoälyn ja koneoppimisen algoritmien toteuttaminen voi olla haastavaa ja vaatii paljon aikaa. On erittäin tärkeä, että opetusympäristö on hyvin pystytetty ja testattu. Pythonille on olemassa ohjelmistokirjastoja, jotka sisältävät valmiiksi kirjoitettua koodia. Kirjastojen käyttäminen vähentää koodin kirjoittamisen tarvetta, mikä nopeuttaa koodin ajoa ohjelmassa sekä vähentää tallennustilan määrää. [57; 58.]

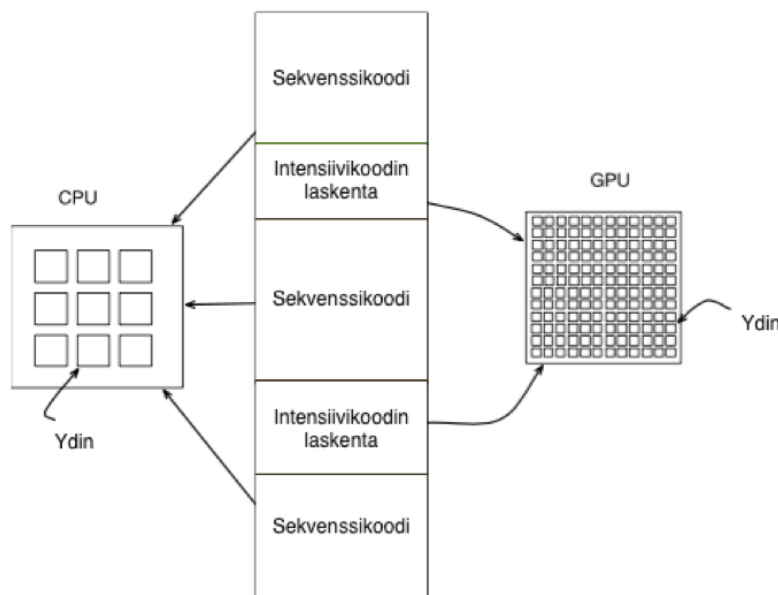
### 3.3.1 Laitteisto, ohjelmistot ja kirjastot

Tietosarjojen ja parametrien kokojen kasvaessa GPU:t ovat tällä hetkellä parhaiten so-piva laskenta-alusta. Ne ovat erikoistuneita prosessointiyksiköitä, jotka on suunniteltu pääasiassa kuvien ja videoiden käsittelyyn. GPU-pohjainen laskenta on tärkeä osa syväoppimisen kehityksessä. Ne ovat CPU-yksikköihin verrattuna yksinkertaisempia prosessointiyksiköitä, mutta niihin voi järjestää paljon suuremman määrän ytimiä. Tämän vuoksi ne sopivat sovelluksiin, joissa tietoa käsitellään rinnakkain, kuten kuvien ja videoiden pikseleitä. CPU:n suurin etu on, että se on helposti ohjelmoitavissa ja tukee kaikkia ohjelmointikehyksiä kuten C/C++:aa, Scalaa, Javaa ja Pythonia. Koneoppimisessa se kuitenkin soveltuu vain yksinkertaisiin neuroverkkomalleihin, joiden koulutukset eivät vie paljon aikaa, sekä pieniin opetusaineistomääriin. On myös olemassa TPU-yksiköitä (Tensor Processing Unit), jotka ovat 10 kertaa nopeampia ja energiatehokkaampia kuin GPU:t. Nykyisin myös GPU:ssa on tensor-ytimiä [60]. Ne ovat erittäin nopeita tiheiden vektori- ja matriisilaskelmien suorittamiseen ja ovat erikoistuneet TensorFlow'hun perustuvan ohjelman suorittamiseen. Niitä on järkevä käyttää vain TensorFlow-pohjaisiin malleihin, joissa ei ole omaa TensorFlow-toimintaa neuroverkon harjoitussilmukan sisällä. [46; 59; 61.]

Tyypillinen syväoppiminen käyttää korkeantason kirjastoja, kuten Kerasta tai Theanoa, jotka automaattisesti muuntavat suoritettavan laskennan saumattomasti CPU:lla ja/tai GPU:lla suoritettavaksi kokonaisuudeksi. Tämän johdosta useimmissa tapauksissa

syväoppimisen harjoittajan ei tarvitse ymmärtää GPU:n sisäistä toimintaa käyttäkseen sitä sovelluksissaan. GPU-pohjaisen laskennan ydin on käsite SIMD (Single Instruction, Multiple Data), jossa sama laskenta suoritetaan rinnakkain datan useista kohdista. Laskenta-alusta sopii hyvin raskaiden lineaaristen algebraoperaatioiden (vektorit ja matriisit) laskemiseen. Syväoppivien neuroverkkojen koulutukseen liittyvä ydinlaskenta on gradienttien laskeminen ja näiden gradienttien mukaisten parametrien päivittäminen. Tämän ytimessä on lineaariset algebriset toiminnot, kuten pistetulo sekä vektori- ja matriisikertoimet. Tästä syystä GPU-pohjainen laskenta sopii neuroverkon koulutukseen ja ennusteiden tekemiseen. [59.]

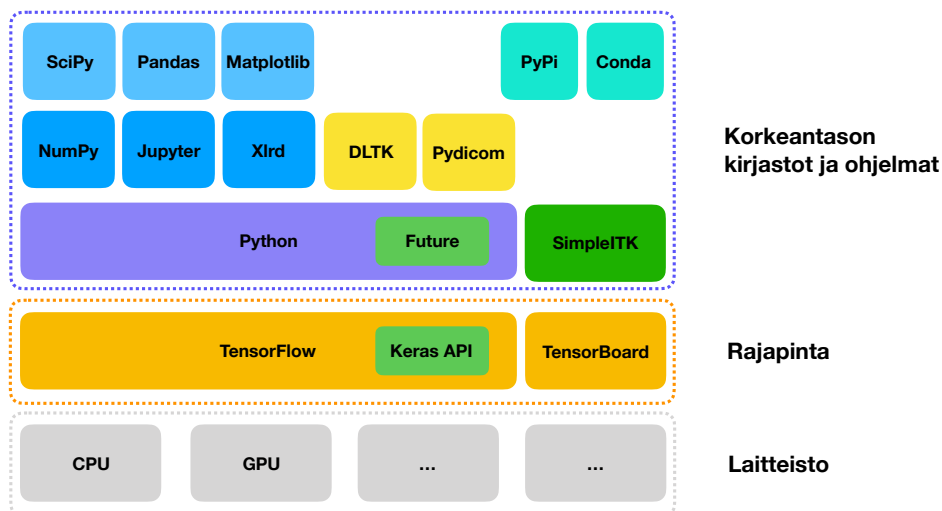
GPU-pohjaisessa laskennassa seuraavat kohdat on huomioitava: Syväoppimiseen liittyvä laskenta sisältää jonkin verran peräkkäin suoritettavaa koodia, ja rinnakkain ajettavaa intensiivikoodia (kuva 16). CPU käsittelee sekvenssikoodin, joka sisältää esimerkiksi tiedon lataamisen levyiltä. Koska laskennallisesti raskas intensiivikoodi sisältää gradienttien laskemisen ja parametrien päivittämisen siirtyy laskennan tiedot ensin GPU-muistiin ja laskenta tapahtuu sen jälkeen GPU:ssa. Laskennan jälkeen tulokset tuodaan takaisin päämuistiin jatkokäsittelyä varten. Tällaisia laskennallisesti raskasta intensiivikoodia voi olla useita lohkoja, jotka on lomitettu sekvenssikodeilla. [59.]



Kuva 16. Syväoppimiseen liittyvän laskennan jaottelu CPU:n ja GPU:n välillä. [59, s. 148.]

GPU:iden ympärille on rakennettu kaksi rajapintaa, joista yksi on Nvidian kehittämä CUDA (Compute Unified Device Architecture), joka toimii ainoastaan Nvidian GPU:issa. Toinen on OpenCL (Open Computing Language), joka on avoimen lähdekoodin GPU-ohjelmointirajapintamääritelmä. CUDA on sekä vanhempi että suositumpi verrattuna OpenCL:iin. GPU-laskenta on nopeuttanut syväoppimismallien opetusta. Aiemmin mallien opettaminen on vienyt CPU-pohjaisissa alustoissa päiviä, tai jopa viikkoja. GPU-pohjainen laskenta vie nyt tunneista päiviin riippuen tiedon määrästä. Opetuksen kesto on kuitenkin riippuvainen opetusaineistosta ja neuroverkkoarkkitehtuurista. Opetukseen tarvitaan merkittävä määrä GPU-yksiköitä, jotta opetusaika voidaan saada mahdollisimman lyhyeksi. [46; 59; 60.]

Kuvaan 17 on koottu ohjelmien ja kirjastojen välinen hierarkia niiden kuvausten perusteella. Katkoviivojen sisälle, esimerkiksi Pythonin päälle, on koottu ohjelmat ja kirjastot, jotka ovat siitä riippuvaisia. PyPi ja Conda ovat pakettienhallintajärjestelmiä, minkä johdosta ne on sijoitettu muista erilleen oikeaan ylänurkkaan. Pääosin kaikki ohjelmat ja kirjastot ovat avointa lähdekoodia.



Kuva 17. Numeerisen koneoppimisen alustan hierarkia

Conda on paketinhallinta- ja ympäristöjärjestelmä. Sen avulla asennetaan ohjelmistopakettien useita eri versioita ja tarvittavista kirjastoista ne, jotka sopivat tietokonealustalle. Condalla voi ylläpitää erilaisia rinnakkaisia Python-ympäristöjä, jotka voivat sisältää eri



versioita Pythonista ja Python-kirjastoista, riippuen käyttötarkoituksesta. Se on luotu Python-ohjelmille, mutta voi pakata ja jakaa ohjelmistoja kaikilla ohjelmointikielillä. [62.]

PyPi, Python Package Index, on Python-ohjelmointikielen ohjelmistojen arkisto. Pip-komento on työkalu Python-pakettien asentamiseen ja hallintaan. [63.]

TensorFlow on ohjelmistokirjasto Pythonille. Se on tarkoitettu korkean suorituskyvyn numeerisia laskemia varten, joiden avulla voidaan luoda kehittyneitä syväoppimisen ja koneoppimisen sovelluksia. Se sisältää laajan ja monipuolisen työkalujen, kirjastojen ja yhteisöresurssien ekosysteemin. CPU:lla suoritettuna TensorFlow käyttää Eigen-nimistä matalan tason kirjastoa tensoritoiminnoille. GPU:ssa TensorFlow käyttää NVIDIA Cuda Deep Neural Network (cuDNN) optimoitua syväoppimisen kirjastoa. TensorFlow sisältää TensorBoardin, mikä on tiedon visualisointityökalu TensorFlow'n lokitiedolle. Sillä voi visualisoida esimerkiksi näytteitä opetusdatasta, verkon suorituskyvyn muutoksista läpi opetuksen etenemiseen sekä laskentaa kuvaavia graafisia esityksiä. [46; 64.]

Keras on korkean tason neuroverkkojen API, jota voi käyttää Python-ohjelmissa ja ajaa TensorFlowin, CNTK:n ja Theanon päällä. Kerasilla voi rakentaa helpon ja nopean prototyypin neuroverkosta, koska se hyödyntää useimpia tähän asti kehitettyjä syväoppimismalleja. Se tukee konvoluutioverkkoja, takaisinkytkettyjä neuroverkkoja sekä näiden yhdistelmiä. Kerasia voi ajaa saumattomasti CPU:n ja GPU:n kanssa. [46; 65.]

DLTK, deep learning toolkit, on kirjasto, joka on luotu helpottamaan lääketieteellisten kuvien käyttämistä syväoppimisessa. Se on kirjoitettu Pythonilla ja ohjelmoitu TensorFlow-järjestelmään, mikä nopeuttaa prototyyppien valmistamista. [66.]

Pydicom on Python-paketti, joka on suunniteltu DICOM-tiedostojen kanssa työskenteleeseen ja myös niiden siirtämiseen (pynetdicom-osio). Sillä voi lukea DICOM-tiedostoista lääketieteellisiä kuvia ja raportteja. Pydicom kääntää tiedostot luonnollisiksi python-tietorakenteiksi, jolloin niitä on helpompi muokata. Muutetut tietokokonaisuudet voidaan kirjoittaa uudelleen DICOM-muotoisiin tiedostoihin. Pydicom ei ole DICOM-palvelin eikä sitä ole suunniteltu ensisijaisesti kuvien katseluun vaan DICOM-tiedostojen dataelementtien muokkaukseen Python-koodilla sekä DICOM-tiedostojen siirtämisen DICOM-tiedonsiirtoprotokollia tukevien järjestelmien kanssa. [67.]

SimpleITK on monikielinen kuvien analysointityökalu, jossa on ohjelmointirajapinta avoimen lähdekoodin ITK-ohjelmaan. ITK (Insight Toolkit) on tehokas kuvien analysointia varten suunniteltu työkalu. ITK:n ydin on toteutettu C++:lla mutta siinä on WrapITK-moduuli esimerkiksi Pythonin ja Javan käyttöön. SimpleITK:lla on helpompi rakentaa prototyyppi ja hyödyntää ITK:n algoritmeja. Tämä kirjasto tukee 2D-, 3D- ja 4D-kuvia sekä valittua pikselityyppiä. SimpleITK tukee DICOM-kuvatiedostojen lukemista. [68.]

Future on Pythonille kirjoitettu kirjasto. Python-future on yhteensopivuuskerros Python 2:n ja Python 3:n välille. Sen avulla voi käyttää Python 3.x -yhteensopivaa koodia, joka tukee sekä Python 2:n ja Python 3:n minimoiden päällekkäisyydet. [69.]

Pandas on BSD-lisensioitu (Berkeley Software Distribution) kirjasto. Se sisältää korkean suorituskyvyn, helppokäyttöisiä tietorakenteita ja tiedon analysointityökaluja Python-ohjelmointikielelle. Se on luotu täydentämään tietojenkäsittelyä Pythonissa. [70.]

Matplotlib on Pythonin 2D-kirjoituskirjasto, jota voi käyttää Python-skripteissä, Python- ja IPython-shells, Jupyter Notebookeissa ja web-sovelluspalvelimissa. Sen avulla voi esimerkiksi luoda tulosteita, histogrammeja, tehospektrejä, viivakaavioita ja virhekarttoja. Kirjastoon on olemassa lisätyökalu 3D-kuvaajan luomisen ja tulostamisen. [71.]

Xlrd on python-kirjasto, joka tarjoaa mahdollisuuden poimia tietoja Microsoft Excel -laskentataulukotiedoista. Paketti itsessään on pelkästään Pythonia, jolla ei ole riippuvuutta moduuleista tai muista paketeista Python-standardin jakelun ulkopuolella. [72.]

NumPy on tieteellisen laskennan Python-kirjasto. Se on kirjoitettu C-kielellä, mikä tekee siitä paljon nopeamman verrattuna muihin Python-kirjastoihin. Kirjastot, jotka on kehitetty tietojenkäsittelytilastoja varten ja yleensä matemaattisia laskelmia varten on kirjoitettu NumPyn päälle. [73.]

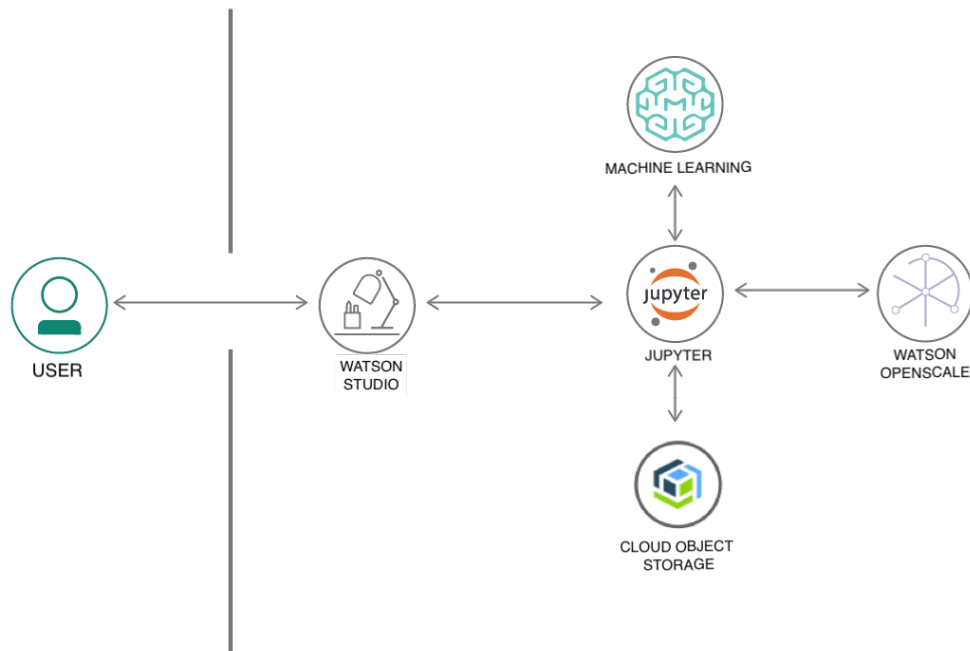
SciPy, scientific Python, on Python-kirjasto, mikä on kirjoitettu NumPyn päälle ja on riippuvainen siitä. SciPy sisältää tilastoinnin, optimoinnin, integroinnin, lineaarisen algebran, Fourier-muunnoksen, signaalin ja kuvankäsittelyn, ODE-ratkaisijan ja useita muita ominaisuuksia. [74.]

### 3.3.2 IBM:n järjestelmät

IBM PowerAI on ammattikäyttöön tarkoitettu laskenta-alusta, joka yhdistää ja tukee suosittuja avoimen lähdekoodin syväoppimiskehyksiä ja -kirjastoja, joita ovat esimerkiksi Tensorflow, PyTorch ja Caffe. Tässä projektissa käytetty PowerAI 1.5.4 -ympäristö käyttää prosessorina NVIDIA Tesla P100 GPU:ta. [75.]

PowerAI-alustalla voidaan esimerkiksi hyödyntää Docker-konttitekniologiaa. Kontilla (engl. Container) tarkoitetaan standardoitua ympäristöä, jossa ohjelmia voidaan ajaa, ja ne mahdollistavat ohjelmien siirtämisen helposti eri ympäristöjen välillä ilman ohjelmien muokkausta. Samassa koneessa voidaan ajaa useita kontteja samanaikaisesti ilman, että ne vaikuttavat toistensa toimintaan. Konttien etu on se, että ne käyttävät samaa käyttöjärjestelmää ja vievät siksi vähemmän tilaa. Lisäksi vioittunut kontti voidaan tarvittaessa korvata välittömästi toisella kontilla, jonka ansiosta toimintavarmuus paranee. [76.]

IBM Cloud on kokoelma pilvipohjaisia tietojenkäsittelypalveluita, ja se sisältää yli 170 tuotetta ja palvelua, joihin kuuluu esimerkiksi data, kontit, tekoäly, IoT ja blockchain. Cloud-palvelut sisältävät infrastruktuuria, ohjelmistoja sekä data-alustoja, jotka toimivat julkisessa ja yksityisessä pilvipalvelussa sekä hybridipilvessä. [77.] Kuvassa 18 on esitely tässä insinööriyössä käytettävien Cloud-palveluiden arkkitehtuuri.



Kuva 18. IBM Cloudin palveluita voidaan yhdistellä erilaisiksi kokonaisuuksiksi käyttötärpeen mukaan.

Watson Studio on IBM:n pilviohjelmistojen (IBM Cloud) datatiedeympäristö monenlaisen datankäsittelyyn tukien ohjelmistokehitystä mm. Python- ja R-kielillä. Se sisältää työkaluja tietojen analysoimiseen ja visualisoimiseen, tietojen puhdistamiseen ja muokkaamiseen, datavirran käsittelyyn sekä koneoppivien mallien luomiseen ja opettamiseen. Alusta tukee myös avoimen lähdekoodin ohjelmia, kuten Jupyter Notebookia ja RStudiota ja on käytettävissä sekä IBM:n pilvipalveluissa että erillisasennuksena yksittäisissä tietokoneympäristöissä tai esimerkiksi muiden palveluntarjoajien pilvialustojen päällä (Amazon, Azure, Google Cloud). [78.]

Cloud Object Storage (COS) on joustava ja skaalautuva pilvialusta erilaisten datojen säilytykseen, kuten kuva, ääni, teksti, taulukko jne. Palveluun integroidulla IBM Asperalla datansiirto on nopeaa palvelujen välillä ja sen avulla voidaan suorittaa data-analyysiä suoraan tallennetusta datasta. [79.]

Watson Machine Learning (WML) on palvelu, joka on tehty helpottamaan tekoälyn käyttöönottoa yritysten erilaisiin käyttötarkoituksiin. WML tarjoaa helpon, nopean ja hallitun tavan tekoälymallien opettamiseen ja testaamiseen sekä mallien ajamiseen ja

käyttämiseen mikropalveluina ohjelmointi- ja verkkorajapinnan kautta. Mallin voi sijoittaa halutulle pilvialustalle (public, private, hybrid, multicloud). [80.]

Watson OpenScale (OS) on työkalu, jolla voidaan valvoa ja hallita tuotannossa olevia kone- ja syväoppimismalleja. OpenScale havaitsee sisällöstä vääristymiä ja auttaa pitämään huolta järjestelmän suorittamien toimenpiteiden oikeudenmukaisuudesta. Työkalu antaa kirjallisen selityksen mallista löydetyistä vääristymistä ja auttaa ymmärtämään niiden vaikutuksia yrityksen toimintaan. Watson OpenScale voi automaattisesti korjata vääristyneitä tuloksia, ja se antaa vertailutiedot sekä vääristyneen että korjatun mallin tuloksista, joihin datatieteilijät ja yritysten johto voivat pohjata päätöksiään. OpenScalella voi seurata muun muassa malleja, joita ajetaan Watson Machine Learning -ympäristössä mutta myös esim. Amazon-, Azure-ympäristöissä sekä räätälöidyissä ympäristöissä.

OpenScale mahdollistaa yksittäisten tapahtumien tarkastelun ja kykenee antamaan selityksen algoritmien antamiin tuloksiin vaikuttaneista tekijöistä ja niiden painoarvosta päätöksen syntymiseen. Tämä luo malliin läpinäkyvyyttä ja selitettävyyttä, jota tarvitaan esimerkiksi terveydenhuollon sovelluksissa. [81.]

### 3.4 Opetusdatan laatu ja tulosten vääristyminen

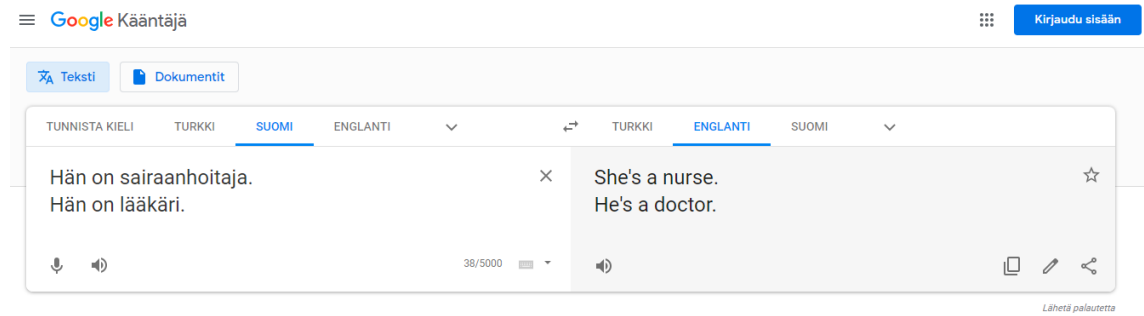
Tietotekniikan tutkija Harri Valpola totesi Wired-lehden haastattelussa, että tekoälyn kehityksen ensimmäisessä aallossa tarvittiin ohjelmoijia ja toisessa aallossa datatieteilijöitä. Nyt kolmannessa aallossa keskiöön on noussut moraalisuus eli käsitys siitä, mikä on oikein ja mikä väärin [82; 83]. Luottamuksen saavuttamiseksi tekoälyn tekemien ratkaisujen täytyy olla eettisesti kestäväällä pohjalla ja annettuja tuloksia täytyy kyetä arvioimaan. Ihmisten on voitava luottaa siihen, että päätöksiä puolestamme tekevä kone antaa faktoihin perustuvia, luotettavia tuloksia, eikä se tahallisesti tai tahattomasti syrji mitään käyttäjäryhmiä tai vaaranna hyvinvointiamme.

Koneoppimisella luotu tekoäly ei kykene luomaan kokonaan uutta, vaan tällä hetkellä kaikki sen toiminta perustuu sille syötettyyn opetusdataan. Sovellus oppii oikeiden tai toivottujen ominaisuuksien ohella aivan yhtä helposti virheelliset toimintatavat ja väärät ajatusmallit, joten opetusdatan laatuun on kiinnitettävä erityistä huomiota.

Ihmiset ovat luonnostaan ennakkoluuloisia ja ihmismieli pyrkii automaattisesti järjestelmään monimutkaiset asiat helposti ymmärrettäviin kokonaisuuksiin. Tämä johtaa siihen, että jokaiselle syntyy ennakkokäsityksiä, jotka voivat olla esimerkiksi tiettyjä ihmisryhmiä suosivia tai syrjiviä. Muiden ihmisten luokittelu voi perustua esimerkiksi ikään, sukupuoleen tai etniseen alkuperään ja niiden taustalla voi olla henkilökohtaisia kokemuksia, opittuja ajatusmalleja tai ympäristöstämme kumpuavia stereotyyppioita. [84.]

Suomessa on jo saatu ensimmäinen ennakkotapaus, jossa luottoyhtiön käyttämä tekoälysovellus oli evännyt hakijalta lainan ikään, sukupuoleen, kieleen ja asuinpaikkaan perustuvan pisteytysjärjestelmän perusteella. Mies oli valittanut saamastaan päätöksestä yhdenvertaisuus- ja tasa-arvolautakunnalle, joka yritti tuloksetta sovitella asiaa yli vuoden ajan. Tilanne päättyi siihen, että yhdenvertaisuusvaltuutettu kielsi uhkasakon voimalla luottoyhtiötä käyttämästä syrjivää menetelmää, joka perustui yksilöllisen arvioinnin sijasta tilastolliseen profilointiin ja oli sitä myöten yhdenvertaisuuslain vastainen. [85.]

Jotta tekoäly voi toimia eettisesti (esim. tasapuolisuus), sen opetukseen käytetyn datan on oltava tarkoitukseen soveltuvaa ja tasapuolisesti jakautunutta (balanced). Monimuotoisuuden takaamiseksi opetusdataa pitää olla riittävästi, eikä sen sisältö saa olla liian geneeristä, jotta tulokset eivät vinoudu ja anna virheellisiä tuloksia. Algoritmien antamat tulokset voivat helposti vääristyä, mikäli opetusmateriaalissa on suuri määrä sisältöä vain tietyltä osa-alueelta. Tämä johtaa tekoälymallin ylioppimiseen, jolloin se ehdottaa vastauksiksi tuloksia, joihin sen oma opetusdata on painottunut. Esimerkiksi Google-kääntäjän antamat vastaukset ovat edelleen joiltain osin sukupuolittuneet (kuva 19), koska taustalla toimiva tekoäly on oppinut tiettyjen ammattiryhmien edustajien olevan tiettyä sukupuolta, ja tämä vääristymä on sisällytetty myös kääntäjän tuloksiin. Esimerkiksi, kun kääntää suomesta englanniksi lauseet ”Hän on lääkäri. Hän on sairaanhoitaja” saadaan tulokseksi ”He’s a doctor. She’s a nurse”.



Kuva 19. Google-kääntäjän antama sukupuolivääristynyt tekstikäännös. Käännös tehty 28.4.2019.

Monissa tapauksissa tekoälyn oppimat vääristymät voidaan korjata lisäämällä opetusdataa niiltä osa-alueilta, joilta sitä on liian vähän tai puuttuu kokonaan. Vääristymien hallinta muuttuu kuitenkin haastavammaksi, mikäli tekoäly oppii esimerkiksi verkkolähteistä, joiden vinoumien löytäminen ja torjuminen on huomattavasti monimutkaisempaa. Tekoälyn syväoppimisen algoritmien kehityksessä onkin alettu painottamaan, miten niiden suorittama päätöksenteko saataisiin helpommin tarkasteltavaksi tai itseään selittäväksi. Tällä tavoin pyritään pääsemään eroon niin sanotusta mustan laatikon ongelmasta, eli tilanteesta, jossa algoritmit ovat niin monimutkaisia, että niiden päätöksenteon prosessia on hyvin vaikea jäljittää tai selittää. [86.]

## 4 Tutkimusmenetelmät

### 4.1 Asiantuntijahaastattelu

Haastattelu on vuorovaikutustilanne, jossa haastattelija pyrkii kysymysten avulla keräämään tietoa määrätystä aihealueesta ja se on erittäin hyödyllinen tapa kerätä tutkimustietoa. Haastattelua suunniteltaessa on tärkeää perehtyä erilaisiin haastattelutekniikoihin, sillä sama lähestymistapa ei toimi jokaisessa tilanteessa. Tekniikoita tulee muokata sen mukaan, onko kyseessä ryhmä- vai yksilöhaastattelu ja toteutetaanko haastattelu kasvotusten vai etänä esimerkiksi sähköpostitse, puhelimitse tai videoyhteyden välityksellä. Haastattelu sopii erityisen hyvin kvalitatiivisen (laadullisen) tiedon keräämiseen. [87.]

Tämän insinööriyön tueksi tehtiin kaksi asiantuntijahaastattelua. Asiantuntijahaastattelu oli rakenteeltaan puolistrukturoitu haastattelu, jossa kysymyksiä ei ole esitetty tarkassa muodossa tai järjestyksessä. Menetelmän tärkein tavoite on päästä käsiksi siihen tietoon, joka asiantuntijalla oletetaan olevan tutkittavasta aiheesta. Asiantuntijuudelle ei ole olemassa yleistä määritelmää, mutta haastateltavalla on yleisesti ammatistaan tai asemastaan johtuen sellaista tietoa, jota maallikolta ei löydy. Asiantuntijuus on aina tilanteeseen ja ihmisen toimintaan sidottu käsite, eikä sitä pidetä henkilökohtaisena kykyinä tai pysyvänä ominaisuutena.

Asiantuntijuus voi esiintyä hyvin kapea-alaisena ja instituutioihin kiinnittyvään asiantuntijuuteen liittyy usein tietynlainen kielenkäyttö, kuten ammattitermien ja lyhenteiden käyttö. Nykyisin ongelmien ratkominen vaatii kuitenkin monenlaista osaamista ja eri osa-alueilta löytyvän osaamisen yhdistämistä. Tarvitaan monialaista kokemusasiantuntijuutta, eli pitkälle kehittyneitä tietotaitoa eri aihealueisiin liittyen.

Asiantuntijahaastattelua voidaan käyttää kolmella eri tavalla. Eksploraatiivinen, eli kartoitettava asiantuntijahaastattelu, sopii parhaiten sellaisten aihealueiden tutkimiseen, jotka ovat uusia tai niistä on saatavilla vain vähän tietoa. Se on myös nopea keino saada lisää tietoa tutkittavasta aiheesta. Systematisoivan haastattelun tavoitteena on saada kokonaisvaltaisemmin selvitettyä asiantuntijan omaama objektiivinen tieto tietystä aihepiiristä. Teoriaa luova asiantuntijahaastattelu pyrkii edellä mainitun lisäksi selvittämään myös tiedostamattomampaa tietoa asiantuntijan käyttäytymisestä sekä ajatuksista aiheeseen liittyen.

Koska asiantuntijahaastatteluille ei ole olemassa määrättyä rakennetta, siihen voidaan soveltaa monenlaisia haastattelutyyppejä tutkittavasta aiheesta, tutkimuskysymyksistä ja tutkimusasetelmasta riippuen. Yksi mahdollisuus on käyttää avointa haastattelua, jossa haastattelu toteutetaan kokonaisuudessaan vain ennalta määrättyjen aihepiirien pohjalta.

Huolellinen valmistautuminen on erittäin tärkeää, ja haastattelijalla olisi syytä olla mahdollisimman selvä näkemys tutkimuksen tavoitteista. Usein kysymyksien edellytetään olevan haastattelua varten räätälöityjä. Kysymykset voivat olla tyyliltään vapaamuotoisia, eli kerronnallisia, tai ne voivat olla tarkasti tiettyyn tilanteeseen tai toimintatapaan



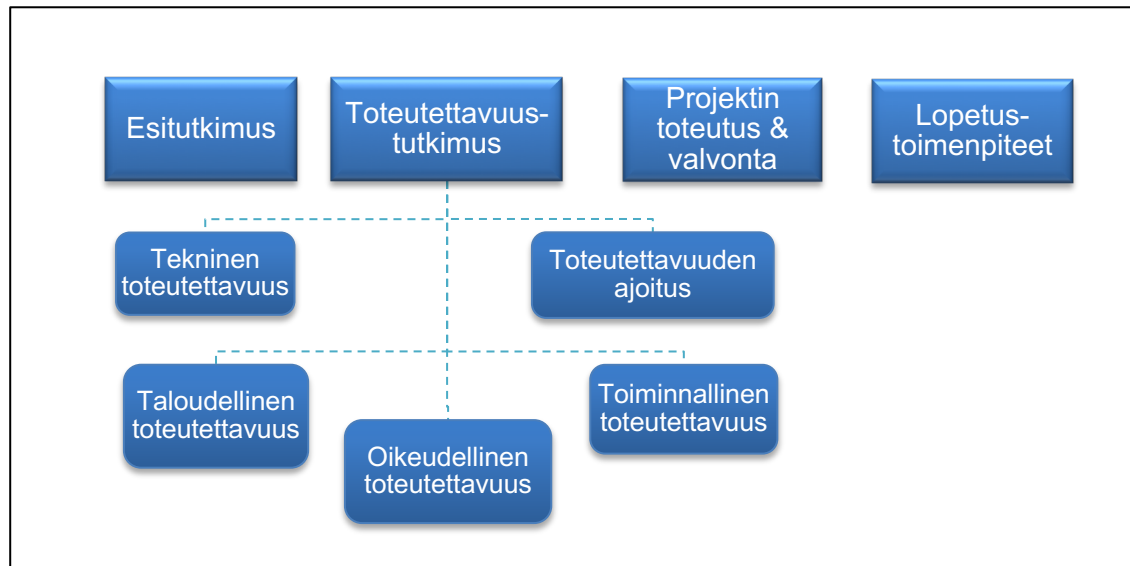
kohdennettuja. Sisällön monipuolisuuden takaamiseksi on myös hyvä esittää kysymyksiä poikkeuksellisista tilanteista. Haastatteluun voi olla hyödyllistä tuoda mukana materiaalia, joka toimii keskustelun inspiraationa. Näissä haastatteluissa käytettiin apuna konseptin arkkitehtuurikuvaa.

Joissain tapauksissa valmistautuminen saattaa jäädä puutteelliseksi, mikäli aiheeseen löytyvää tietoa ei joko löydy tai haastattelu järjestyy niin nopealla aikataululla, että kaikkea aineistoa ei ehdi perehtymään ennen tapaamista. Haastattelun aloittamista ei kuitenkaan ole syytä näistä syistä pelätä tai pitkittää, sillä yllätyksellisyys kuuluu tutkimustyöhön. Loppujen lopuksi asiantuntijoiden haastattelu on aina ennakoimatonta ja tilanesidonnaista. Haastatteluun on useimmiten mahdollisuus palata vielä jälkikäteen ja pyytää tarkennuksia tai korjauksia joko puhelimitse tai sähköpostin välityksellä. [88.]

#### 4.2 Toteutettavuustutkimus

Feasibility study -menetelmä suomennetaan seuraavasti: kannattavuus selvitys, esiselvitys toteuttamiskelpoisuudesta, toteuttamiskelpoisuus selvitys, toteutettavuustutkimus, esitutkimus, toteuttamiskelpoisuutta koskeva selvitys [89]. Suomensos riippuu tutkimuksen kohteen aihealueesta sekä sen tarkoituksesta. Tässä työssä käytetään nimitystä toteutettavuustutkimus.

Toteutettavuustutkimus toteutetaan osana projektin elinkaarta, joka on esitetty kuvassa 20. Sen pohjalta käy ilmi projektin jatkamisen kannattavuus, toteutuvuus sekä varmistetaan yrityksen resurssien hyödyntäminen parhaiten. Projektin alussa tehdään esitutkimus selvittämällä tarve tai tilaisuus tuotteelle, laitokselle tai palvelulle. Toteutettavuustutkimus tehdään toisessa vaiheessa. Kolmannessa vaiheessa toteutetaan projekti toteutettavuustutkimuksen mukaan muodostetun peruslinjan mukaisesti. Neljännessä ja viimeisessä vaiheessa projekti on toteutettu ja päättyy. [90; 91.] Tässä työssä keskitytään toteutettavuustutkimuksen alaosiin tekninen toteutettavuus.



Kuva 20. Projektin elinkaari

Technical feasibility (technical feasibility study) -termiä ei ole suomennettu ja vapaa suomennos siitä on teknillinen toteutettavuustutkimus. Menetelmä on yksi toteutettavuustutkimuksen alaotsikoista, ja se on prosessi, jossa validoidaan tuotteen tai hankkeen tekniset oletukset, arkkitehtuuri ja suunnittelu. Tutkimuksen tarkoitus on osoittaa, onko hanke tai tuote teknillisesti toteutettavissa. Sen päätarkoitus on auttaa organisaatiota määrittämään, täyttävätkö tekniset resurssit kapasiteetin ja onko tekninen tiimi kykenevä muuttamaan ideoita työjärjestelmiksi. Tekninen toteutettavuus edellyttää myös ehdotetun järjestelmän laitteisto-, ohjelmisto- ja muiden teknologiavaatimusten arviointia [91]. On olemassa 14 erilaista teknillisen toteutettavuustutkimuksen aluetta, jotka ovat:

- **Konsepti.** Proof of concept. Suoritetaan todiste idean ja lähestymistavan testaamiseksi.
- **Infrastrukturi.** Tutkitaan infrastruktuurin kapasiteetti, suorituskkyominaisuudet ja toimivuus. Hanke voi esimerkiksi vahvistaa sen käyttökelpoisuuden.
- **Laitteisto.** Vahvistetaan esimerkiksi tietokeskuksen kaltaiset tilat vastamaan projektivaatimuksia.
- **Arkkitehtuuri ja suunnittelu.** Hankkeen arkkitehtuurin ja suunnittelun validointi toiminnallisia ja ei-toiminnallisia vaatimuksia vastaan.
- **Data** vaatimusten tarkistaminen. Arvioida esimerkiksi vaadittujen tietojen datan laatua.

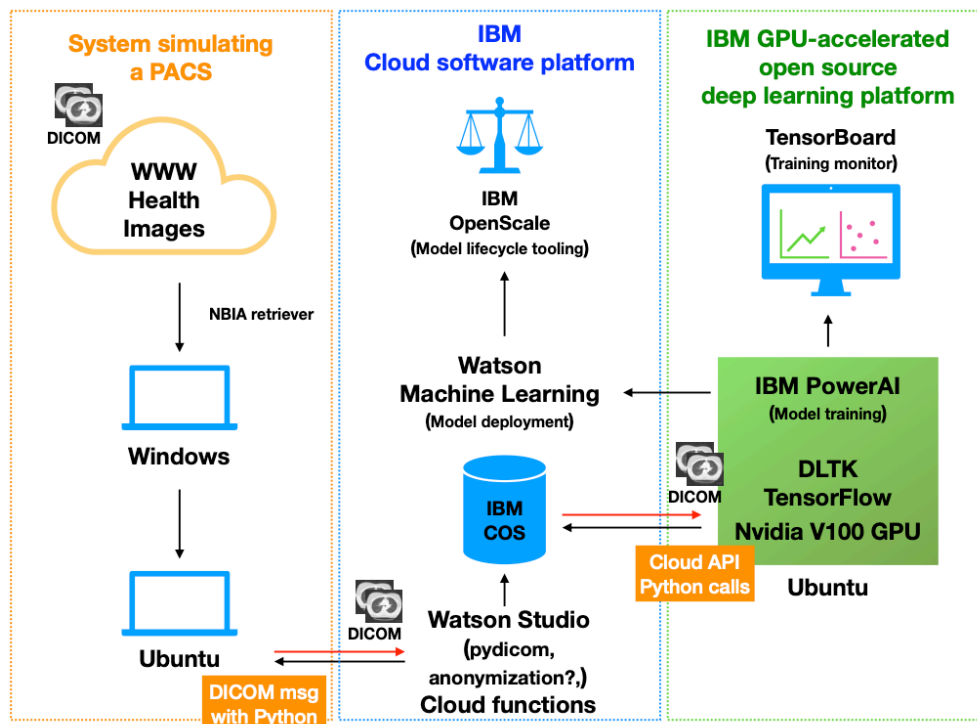
- **Noudattaminen.** Teknisten standardien ja määräysten noudattaminen.
- **Alustat ja sovellusliittymät,** niiden arvioiminen esimerkiksi toiminallisuuden ja luotettavuuden aloilla.
- **Komponentit.** Komponenttien ja materiaalien testit ja prototyypit.
- **Työkalut.** Teknologioiden, kuten järjestelmien ja sovellusten validointi esimerkiksi vahvistamaan, että sovellus voidaan räätälöidä vastaamaan käyttöliittymän vaatimuksia.
- **Integraatio.** Tarkastellaan, miten prosessit, järjestelmä, sovellukset ja tiedot toimivat yhdessä.
- **Tietoturva.** Arvioidaan infrastruktuurin, arkkitehtuurin, mallien, tuotteiden ja komponenttien turvallisuutta.
- **Laitteet.** Tutkitaan laitteiden, esimerkiksi robotiikan, ominaisuuksia tai suorituskykyä.
- **Hankinta.** Tarkastetaan, että hankkeen/projektin tarvittavat tuotteet voidaan hankkia ja vahvistetaan niiden laatu, toimivuus ja kaupalliset ehdot, kuten hinta.
- **Toiminnot.** Hankkeen toteuttamis-, hallinto- ja käyttökelpoisuus huomioiden toimintariskit. [92.]

Tässä työssä toteutettavuustutkimuksessa keskitytään erityisesti konseptin luomiseen, infrastruktuuriin, arkkitehtuuriin ja suunnitteluun, integraatioon, määräysten noudattamiseen sekä alustojen ja sovellusliittymien toiminnallisuuden arvioimiseen. Tärkeimpänä kohtana on tutkia projektin aloittamisen kannattavuutta järjestelmien toimivuuden ja yhteensopivuuden näkökulmasta.

## 5 Toteutus ja tulokset

Toteutettavuustutkimuksen päätarkoituksena oli tarkastella, miten lääketieteellisiä kuvia voidaan käsitellä DICOM-tiedostoina läpi syväoppimiseen liittyvän datatiedeprosessin työnkulun käyttäen avoimen lähdekoodin ratkaisuja IBM:n syväoppimiseen suunnattujen kehitys- ja laskentaympäristöissä (IBM Cloud software ja IBM PowerAI). Projektissa on käytetty syväoppimisen menetelmänä ohjattua oppimista.

Toteutusympäristöstä piirrettiin arkkitehtuurikuva (kuva 21), johon on kuvattu datan liikuminen eri ympäristöjen välillä. Työssä perehdyttiin myös siihen, kuinka DICOM-tiedostoja liikutetaan PACS:ssa sekä tarkasteltiin, miten niiden anonymisointi tapahtuu ennen tiedostojen siirtymistä PACS:n ulkopuolelle. Työssä tutkittiin myös, miten uusilla työvälineillä voidaan seurata opittujen ja käyttöön vietyjen tekoälymallien suoriutumiskykyä liittyen muun muassa niiden antamien tulosten tasapuolisuuteen ja ymmärrettävyyteen. Lisäksi tarkasteltiin prosessin kulun eri vaiheiden järjestystä sekä huomioonotettavia seikkoja.



Kuva 21. Toteutettavuustutkimuksen osa-alueet

Projekti suoritettiin kannettavilla tietokoneilla Web-selaimessa IBM Cloudissa luotujen tunnusten avulla. Yhteydet ympäristöihin otettiin terminaalien kautta hyödyntäen VPN- (Virtual Private Network) sekä SSH-yhteyttä (Secure Shell). Tutkittavan toteutuksen eri vaiheet ja osa-alueet luotiin eri järjestyksessä kuin kuvassa on esitetty. Ne ovat toisistaan riippumattomia pilvialustoja, joiden välille luotiin yhteys kunkin osan valmistumisen jälkeen.

## 5.1 Haastattelujen tulokset

Insinööriyötä varten tehtiin kaksi eksploratiivista asiantuntijahaastattelua. Ensimmäiseksi haastateltiin Neagenin toimitusjohtajaa Lasse Jyrkistä. Neagen Oy on Oulussa vuonna 2002 perustettu terveysteknologian yritys, joka suunnittelee ja valmistaa lääketieteellisen kuvantamisen ohjelmistoja sekä julkiselle sektorille että yksityisille terveydenhuollon toimijoille. Haastattelu suoritettiin 26.4.2019 Menuetto Business Centerissä Helsingin Munkkivuoressa. Haastattelun pohjana käytettiin järjestelmän arkkitehtuuripiirosta (kuva17). Keskeisiksi aiheiksi muodostuivat tietojen siirto, tietoturva, lainsäädäntö sekä tekoälymalleihin käytettävän opetusdatan laatu. [94.]

Arkkitehtuurin perusteella voitiin todeta, että DICOM-tiedostojen siirto PACS-ympäristöstä on toteutettavissa DICOM toolkitin avulla, joten sen toteutettavuus voitiin näin ollen varmistaa. DICOM-tiedostoja etsitään halutuilla kriteereillä PACS-järjestelmästä C-Find-komennolla. Tiedostojen reitittäminen kuva-analyysiin onnistuu kahdella eri tavalla joko käyttämällä C-Move- tai C-Get-komentoa. C-Moven etu on siinä, että sitä varten joudutaan avaamaan rajapinnasta ainoastaan yksi portti, jonka kautta tieto liikkuu molempiin suuntiin. Jälkimmäinen menetelmä vaatii sen, että ensin Cloud-ympäristöstä lähetetään C-Get-pyyntö PACS:iin, johon järjestelmä vastaa C-Storella. Tämä vaatii rajapinnan avaamisen erikseen kummallekin toiminnolle. C-Get poistettiin välillä standardista, mutta ne palautettiin myöhemmin. Osa PACS-toimijoista siirtyi takaisin käyttämään C-Getiä, mutta kaikki järjestelmät eivät välttämättä enää tue komentoa.

Kun tekoälymalli opetetaan DICOM-kuvilla, on huomioitava kuvien metadatan sisältö, jotta malli ei opi yhdistämään epäoleellista tietoa luokitteluihin. Ääriesimerkkinä tämä voisi olla tietyn sukunimen yhdistäminen sen kanssa usein esiintyneisiin diagnooseihin. Mikäli anonymisoitava tieto korvataan metatietoihin toisenlaisella sisällöllä, on oltava varmoja, että se ei johda mallin vääristymiseen. Opetusta tehdessä on määriteltävä tarkasti, mitkä tiedot ovat mallin opetuksen kannalta oleellisia.

Haastattelussa korostui, että opetusdatan valinnassa on lisäksi tärkeä huomioida, että siinä ei korostu liikaa mikään yksittäinen ominaisuus, kuten tietyt diagnoosit, kuvantamislaitte, ikä tai sukupuoli. On huomioitava, että myös geneettinen alkuperä voi vaikuttaa mallin tuloksiin ja siksi esimerkiksi suomalaisen väestön kuvien pohjalta opetettu malli ei

ole automaattisesti sovellettavissa myös muissa maissa. Opetusdatan on oltava mahdollisimman monipuolinen, jotta siihen ei synny vääristymiä. Siksi on hyvin oleellista, että malleja rakentaessa päästäisiin käyttämään mahdollisimman laajaa aineistoa.

Käsittelyssä olevan toisilain on tarkoitus helpottaa lääketieteellisen datan käyttömahdollisuuksia myös tieteellisessä tutkimuksessa, ja tämä tuo järjestelmien käyttöön omanlaisensa haasteet. Tietojen haku aiheuttaa aina järjestelmälle kuormitusta ja siksi on syytä miettiä, miten ja missä datan käsittely tapahtuu ja minkä verran sen prosessointi vaatii järjestelmältä. Esimerkiksi tietojen louhiminen potilastietojen seasta voi aiheuttaa ongelmia. Mikäli asiaa ei oteta etukäteen huomioon ja varmisteta prosessointitehon priorisointia sairaalan omaan käyttöön, on olemassa riski, että se vaarantaa järjestelmän toiminnan ja näin ollen myös potilasturvallisuuden.

Haastattelussa pohdittiin potilasta suojaavia lakeja ja niiden vaikutusta järjestelmän kehitykseen sen eri vaiheissa. Potilaan oikeuksiin kuuluu saada tietää, milloin hänen tietojensa on käsitelty ja kenen toimesta. Nämä tiedot tallentuvat potilastietojärjestelmään, josta ne on toimitettava potilaalle hänen niin pyytäessään. Olisi hyvä, että potilastietoihin tallentuisi myös tieto siitä, mikäli hänen tietojensa on käytetty tutkimustarkoituksiin. On myös relevanttia pohtia, tarvitaanko potilailta tulevaisuudessa erillistä lupaa siihen, saako heidän potilastietojensa käyttää tekoälymallien rakentamiseen. On vielä epäselvää, voidaanko tekoälymalli joissain tapauksissa tulkita rekisteriksi tai rekisterin osaksi ja miten siinä tapauksessa tulee hoitaa GDPR:n asettamat vaatimukset.

Anonymisoinnista huolimatta on olemassa pieni mahdollisuus, että kuva identifioi potilaan. On tiedossa tapauksia, joissa potilas on tunnistanut omat hammaskarttakuvansa luennolla, ja kirurgi oli kyennyt tunnistamaan potilaskuvassa olevista ruuveista, että leikkaus oli tehty heidän sairaalassaan. Jos potilas on tunnistettavissa, on olemassa riski, että Cloud Object Storagesta muodostuisi laitton rekisteri.

Haastateltavan kanssa pohdittiin myös tietoturva-asioita eri näkökulmista. Koska PACS-arkistot sisältävät lailla suojattua potilasdataa, niillä on oltava lain vaatima suojaus tietoturvahukien varalta. Eri sairaanhoitopiirit valitsevat itse arkistojen toimittajan, joten yhden suurarkiston sijaan on olemassa useita, pienempiä järjestelmiä. Tällä on etunsa tietojen suojaamisen kannalta, koska rikollisen saavuttamat potentiaaliset hyödyt yksittäiseen

arkistoon murtautumisesta jäisivät pieniksi. Tulevaisuudessa ollaan kuitenkin siirtymässä valtakunnalliseen terveydenhuollon kuva-aineiston arkistoon, Kvarkkiin, johon on jatkossa tarkoitus siirtää ainakin röntgentutkimusten, varjoainekuvausten, tietokonekerroskuvausten, ultraäänikuvausten, magneettikuvausten, isotooppikuvausten ja hybridikuvantamisen kuva-aineistot sekä valokuvia ja videotallenteita. Lisäksi kuva-arkistoon siirretään EKG-, EEG-, KTG- ja FAG-tutkimusten tulokset. [95.] Tämä saattaa lisätä rikollisten kiinnostusta arkistoa kohtaan ja tämä on hyvä ottaa huomioon tietoja suojaessa.

Haastateltava totesi, että heikoin lenkki PACS-järjestelmän suojauksessa on kuitenkin ihminen. Mikäli järjestelmän toimintaa halutaan häiritä, niin helpoin tapa sen tekemiseen on ylikuormittaa järjestelmä niin, että se kaatuu. Pahimmassa tapauksessa järjestelmään kohdistettu hyökkäys saattaisi lukita koko rekisterin ja sekoittaa sairaalan toiminnan. Cyber-hyökkäystä todennäköisempää on, että järjestelmään murtaudutaan sairaalan omien työpisteiden kautta, ja tämän takia tietoturvallisten käyttäytymismallien omaksuminen sairaalaympäristössä on ensisijaisen tärkeää.

Tietoturvasta puhuessa täytyy kuitenkin muistaa, että mikään tapa tiedon turvaamiseen ei ole 100-prosenttisen varma. Kaikki kryptaukset ovat teoriassa murrettavissa ja laitteisiin on mahdollista asentaa ennen käyttöönottoa tai käytön aikana erilaisia haittaohjelmia ja ulkoisia komponentteja, jotka voivat aiheuttaa häiriötä tai lähettää suojattua tietoa ohi järjestelmän. Uhkiin on varauduttava riittävällä vakavuudella, kuitenkin estämättä kehityksen kulkua ja uusien innovaatioiden syntymistä.

Toinen haastateltava oli lääketieteellisten kuvien käsittelyn parissa tutkijana työskentelevä Antonios Thanellas. Hän viimeistelee paraikaa tohtorintutkintoon johtavia maisteriopintojaan Aalto-yliopistossa lääketieteellisten kuvankäsittelyn parissa, jossa hänen osaamisaluettaan ovat olleet erityisesti magneettikuvat. HUS:lla hän on työskennellyt vajaan kahden vuoden ajan TT-kuvien parissa. Haastattelu tehtiin HUS:n tiloissa Gradus-talossa Helsingissä 29.4.2019.

Haastateltavalla on kokemusta tämän insinööriyön kaltaisesta projektista, jossa aivojen TT-kuvista pyritään rakentamaan poikkeamia havaitseva neuroverkkomalli ja haastateltavan tiimissä on jo ehditty tehdä pidemmän aikaa työtä datan anonymisoinnin parissa.

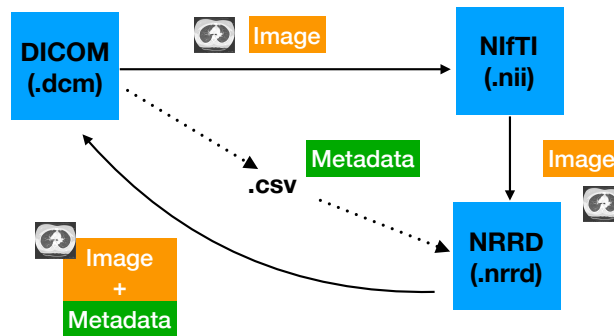
Haastattelu aloitettiin esittelemällä insinööriyön arkkitehtuuripiirros, jonka pohjalta keskustelua pääasiassa käytiin. Koska Antonioksen osaaminen kohdistuu erityisesti kuvien käsittelyyn, keskityttiin pääasiassa DICOM-kuvien anonymisointiin ja käsittelyyn liittyviin aihealueisiin.

Haastattelusta selvisi, että DICOM-tiedostojen metadatassa on paljon informaatiota sisältäviä tageja, ja normaalissa tilanteessa kuva voi sisältää jopa 2000 erilaista merkintää. Tämän lisäksi tiedostosta voi löytyä valmistajan omia, yksityisiä tageja, jotka eivät välttämättä näy muille kuin valmistajalle itselleen. Nämä voivat aiheuttaa ongelmia anonymisoinnissa, sillä vaikka kuvasta poistaisi tai korvaisi kaikki poistettavaksi valitut potilastunnisteet, piilotageihin voi jäädä sensitiivistä tietoa, jonka ei olisi olettanut olevan siellä alunperinkään.

Yksi keino asian ratkaisemiseksi on poistaa tiedot kaikista muista tageista paitsi niistä, joiden sisällön on pystynyt todentamaan. Tämä menetelmä toimii tiettyyn pisteeseen asti, mutta joissain tapauksissa toimenpide on johtanut koko tiedoston rikkoutumiseen, eikä sitä ole pystytty enää lukemaan. Jos tiedostot halutaan pitää lukukelpoisina, niitä joutuu pahimmassa tapauksessa tarkastamaan manuaalisesti, jotta voidaan varmistaa tietojen oikeellisuus. Jos kuvia on tuhansia, tähän menee huomattava määrä aikaa.

Haastateltavan mukaan paras ratkaisu ongelmaan on yhdistää tiedot siten, että otetaan ensin DICOM-tiedoston tärkeimmät metadatat talteen erilliselle CSV-tiedostolle. Tämän jälkeen muutetaan ensin DICOM-tiedosto NIFTI-muotoikseksi ja sen jälkeen NRRD:ksi, jolloin tiedostoon saadaan lisättyä CSV-tiedostosta mukaan tarvittava metadata (kuva 22). Tämä on kohtalainen helppo tapa varmistaa tietojen mahdollisen turvallinen käsittely siten, että mukana voidaan pitää kohtalainen määrä informaatiota.





Kuva 22. NIFTI:n ja NRRD:n avulla DICOM-tiedostojen piilotetut tagit voidaan turvallisesti poistaa.

Ikinä ei kuitenkaan voida olla täysin varmoja, että tiedostosta on saatu poistettua kaikki ylimääräinen tieto, sillä tietoa voi löytyä metadatan lisäksi itse kuvasta. Näidenkin poistamiseen on olemassa erillisiä ohjelmia, jotka etsivät kuvasta sinne kuulumatonta tietoa, mutta suurin osa niistä on tehty NIFTI-tiedostoille. Joistain kuvista, kuten kallon kuvista, ympäröivään tilaan kirjatut annotoinnit on kohtalaisen helppo poistaa, koska ne erottuvat selkeästi taustasta. Kallokuvaan on myös olemassa valmiita ohjelmia, jotka poistavat niistä kasvojen muotoja.

Haastateltu ei ollut erityisen huolissaan siitä, että potilaita voitaisiin tunnistaa kuvista uniikkien tunnisteiden, kuten implanttien tai esimerkiksi hammaskartan perusteella. On mahdollista, että tunnistamista tapahtuu, mutta sen todennäköisyys on äärimmäisen pieni. Lisäksi TT-kuvissa esimerkiksi useat hammasimplantit ikään kuin peittävät itse itsensä, sillä metalli näkyy häiriönä kuvissa. Mikäli keskittyy etsimään täydellistä tapaa anonymisoida kaikki mahdolliset potentiaalisetkin virheet, voi olla, ettei koskaan tule löytämään ratkaisua. Täytyy vaan toimia oman parhaan osaamisensa mukaan.

Liiallinen ominaisuuksien tai reunojen poistaminen voi pahimmillaan johtaa koneoppimisessa siihen, että neuroverkko oppii väärin, kun se ei näe koko kuvaa. Toisaalta taas ylimääräisen informaation opettaminen ei ole tarkoituksenmukaista, joten mallin rakentamisessa täytyy ensimmäisenä selvittää, mikä on juuri kyseiselle mallille ominaista tietoa oppia.

Haastateltu totesi, että tutkimustyössä tehdään usein neuroverkkomalleja, jotka ovat niin tarkkaan valikoidulla datalla opetettuja ja testattuja, että niiden vastaukseksi antamat

todennäköisyydet esimerkiksi sairauksien tunnistamiseen ovat todella korkeat. Tärkeämpää olisi opettaa malli mahdollisimman monimuotoisella aineistolla, jotta se oppisi tekemään tunnistuksia myös laadultaan vaihtelevasta aineistosta, jota sairaalaympäristössä todellisuudessa esiintyy, sillä moni asia voi aiheuttaa kuviin häiriötä tai poikkeamia. TT-kuvissa häiriötä voi aiheuttaa esimerkiksi kehosta löytyvä metalli, potilaan liikkuminen kuvaa otettaessa sekä lääkärin valitsema rekonstruktiosuodatin. Jokaisella valmistajalla on lisäksi valittavana erilaiset käyttöjärjestelmän asetukset, jotka voivat poiketa paljonkin toisistaan.

Kerroskuvauksessa tarkkuuteen vaikuttaa hyvin paljon myös kuvauskerrosten paksuus. Toisinaan lääkärit voivat haluta ottaa ohuempia kerroksia sellaisista kohdista, joissa tiedetään olevan löydöksiä ja paksumpia kohtia kyseisen alueen ulkopuolelta. Toinen lääkäri saattaa sen sijaan valita aivan toisenlaiset asetukset, jolloin samankaltaisesta tilanteesta syntyy melko eri näköisiä kuvia. Kuvien uudelleenrakentamisessa tapahtuukin enemmän muutoksia kuin kuvaa otettaessa. Jotkut saattavat ottaa kuvia myös normaalista poikkeavissa kulmissa, jolloin kuva näyttää hyvin erilaiselta kuin vaakatasossa otettu kuva. Lisäksi on hyvä osata ottaa huomioon biologiset eroavaisuudet. Esimerkiksi aivoissa voi olla rakenteellisia muutoksia esimerkiksi turvotuksen tai muunlaisen aivoaurion vuoksi.

Lopuksi todettiin, että hyvän datajoukon valitseminen vaatisi moniosaavan tiimin, joka osaisi ottaa huomioon aiheeseen liittyviä erilaisia osa-alueita. Tämä vaatisi myös sitä, että käytössä olisi riittävä määrä dataa, jotta siitä olisi varaa valita. Täytyy myös olla tietoinen niistä asioista, joita kuvista on otettava huomioon, kuten esimerkiksi kolmiulotteiset kuvien koordinaattitiedoista. 3D-kuvien alkuun on tallennettu erillinen koordinaatit sisältävä viipale (scout slice), jonka perusteella kuvat kasataan. Mikäli tiedot poistetaan kuvasta, on sitä enää hyvin vaikea saada koottua oikein. Lisäksi tietyissä lääketieteen ammattilaisten käyttämissä tietokoneohjelmissa segmentointi voi tapahtua sellaisella tavalla, joka sotkee kuvaan annotoitujen tietojen koordinaatit, mikäli ne siirretään johonkin toiseen järjestelmään. Tämä voi olla riski, jos kuvatietoja aletaan keräämään yhteen paikkaan, esimerkiksi datajärveen (datalake). Datajärvi on keskitetty tallennustila, johon dataa tallennetaan raakamuodossa.

Haastatteluiden lopputuloksena toteutettavuustutkimukseen saatiin tärkeää tietoa sellaisista lääketieteellisen datankäsittelyn osa-alueista, joiden parissa työskentely olisi vaatinut tutkimusluvan hankkimista ja ne olisivat siten jääneet kokonaan tämän työn ulkopuolelle. Haastatteluista saatiin ammattilaisten kokemuksiin perustuvia suosituksia DICOM-tiedostojen siirtämisestä ja niiden anonymisoinnista. Näiden tietojen avulla pystyttiin varmentamaan tiedonsiirtomenetelmät, joilla DICOM-tiedostoja voi siirtää PACS-ympäristön ulkopuolelle. Molemmissa haastatteluissa korostui hyvälaatuisen opetusdatan merkitys mallia opettaessa sekä tietosuojan huomioiminen dataa käsitellessä. Nämä asiat tullaan huomioimaan projektin jatkokehityksen seuraavissa vaiheissa.

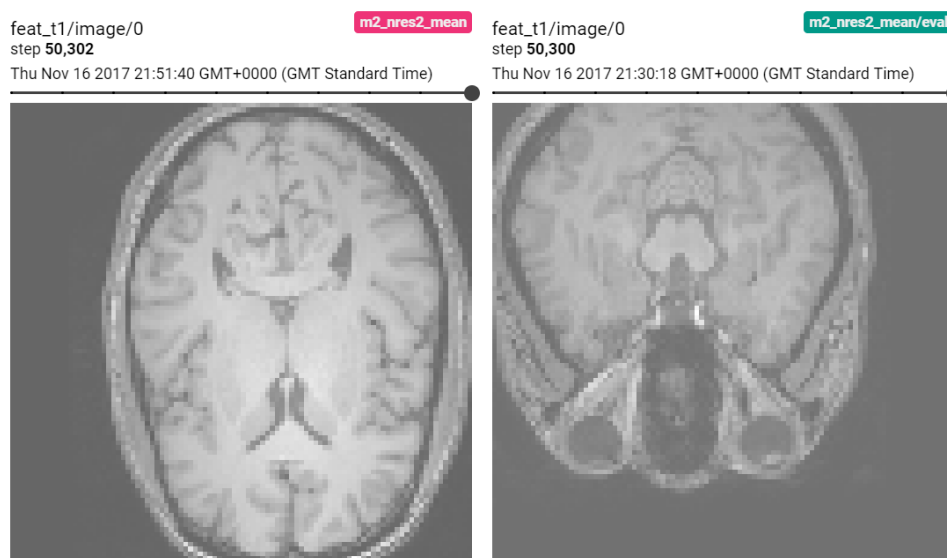
## 5.2 PowerAI-ympäristön pystytys ja testaus

Projektissa käytetyn PowerAI:n käyttöjärjestelmänä toimi Linux-pohjainen Ubuntu 18.04.1 LTS. PowerAI on käytettävissä myös paikallasennuksina soveltuvalla GPU-palvelinlaitteistolla (esim. IBM Power9-palvelimet), mutta tässä työssä käytetty PowerAI-ympäristö sijaitsi Pedabin (IBM-yhteistyökumppani) konesalissa. Yhteys PowerAI-pilvialustalle muodostettiin VPN-yhteydellä. Ympäristöön kirjauduttiin terminaalien kautta SSH-yhteydellä. SSH, luo suojatun yhteyden kahden koneen välille myös epäturvallisissa verkoissa.

Ympäristön pystytys PowerAI:ssa aloitettiin luomalla Condalla oma työskentely-ympäristö, johon kaikki tarvittavat kirjastot ja muut tarvittavat ohjelmat asennettiin. Ympäristö pystytettiin osaksi laajempaan kokonaisuutta, joten luomalla oma dedikoitu ympäristö vältetään muiden samassa ympäristössä ajettujen ohjelmien aiheuttamilta häiriöiltä ja mahdollistetaan omassa projektissa tarvittavien ohjelmaversioiden käyttö. Esimerkiksi jotkin kirjastot saattavat toimia vain Python 2.x- tai Python 3.x- versioilla, joten on tärkeää varmistaa eri komponenttien yhteensopivuus.

Ympäristön luonnissa ja aktivoinnissa päivitettiin Conda-versioon 4.6.7 ja aloitettiin tarvittavien kirjastojen asentaminen. PyPin avulla asennettiin DLTK-, Scikit-build- ja SimpleITK-kirjastot. Condan avulla asennettiin Future Pandas, Scikit-image sekä Cmake. Samalla päivitettiin Conda Freetype ja PyPi. Kirjastojen asennuksessa ilmeni ongelmia ja syyksi ilmeni, että ympäristössä oli aktivoitava TensorFlow erikseen. Kyseisen PowerAI-ympäristön päivitysten myötä TensorFlow aktivoitui jatkossa sisäänkirjautumisen alussa.

Ympäristön testaus tehtiin DLTK – deep learning tool kit -artikkelin [29] mukaan, koska tarkoitus oli testata asennettuja kirjastoja. Artikkelin github-repositoryn [66] sisältö kloonattiin ympäristöön ohjeiden mukaan, mikä loi kansion nimeltä DLTK. Githubin ohjeiden mukaan ajettiin download\_IXI\_HH.py-tiedosto, mikä koostui Python-skriptistä datan lataamista varten. Python-skriptiin oli määriteltynä, miten ja mistä data ladataan ja tallennetaan. Lataamisessa aiheutui virheviesti ja selvisi, että Excel-tiedostoja käsittelevä Xlrd-kirjasto ei ollut ajan tasalla. Kyseinen kirjasto päivitettiin versioon xlrd-1.2.0 ja Python-skripti ajettiin uudelleen, jolloin puuttuneet tiedostot latautuivat. Kokonaisuudessaan datan lataaminen kesti noin 50 minuuttia. Kuvassa 23 on esitelty esimerkki ladatusta kuvamateriaalista. Materiaali sisälsi 178 kuvaa ja niitä vastaavat demografiset tiedot CSV-tiedostona. [29; 66.]



Kuva 23. Pään MK-kuvia NIfTI-muodossa. [66]

Githubista ladattu repository sisälsi 6 erilaista mahdollisuutta testaukseen, jotka löytyivät Examples-kansion alla olevassa alakansiossa nimeltä applications. Valitsimme IXI\_HH\_sex\_classification\_resnet-testauksen, jossa tutkittiin sukupuoliluokittelua aivo-kuvista. Saman nimisestä kansioista löytyi tiedosto nimeltä train.py. Kyseinen tiedosto sisälsi Python-skriptiä, jossa luotiin ohjattua oppimista soveltava luokittelu neuroverkko-malli. Python-skriptiä ei kuitenkaan onnistuttu ajamaan, sillä se ei löytänyt ”module dlTK”, koska se yritti etsiä sitä samasta kansioista, missä kyseinen Python-skripti sijaitti. Virhe saatiin korjattua lisäämällä train.py-tiedoston alkuun polku DLTK-moduuliin.

Train.py-tiedosto sisälsi tiedon, mistä voi poimia kuvat opetukseen, mutta CSV-tiedoston (kuva 24), johon oli määritelty miehet vastaamaan numeroa 1 ja naiset numeroa 2, sijainti oli määriteltävä. Sukupuoli oli esitettävä numerona sillä kyseinen neuroverkko ei kykene tekemään luokittelua luonnollisesta kielestä. Koska neuroverkko ei kykene käsittelemään kuvia kuvina, käsitellään ne numerojoukkoina, jotka muodostuivat kuvan pikselidatasta. Materiaalista 150 datasettiä käytettiin opetukseen ja loput validointiin. Neuroverkon opetus ympäristössä kesti noin 6 tuntia. [66.]

```
IXI_ID, "SEX_ID (1=m,
2=f)", HEIGHT, WEIGHT, ETHNIC_ID, MARITAL_ID, OCCUPATION_ID, QUALIFICATION_ID, DOB,
DATE_AVAILABLE, STUDY_DATE, AGE
IXI012, 1, 175, 70, 1, 2, 1, 5, 1966-08-20, 1, 2005-06-01, 38.7816563997
IXI013, 1, 182, 70, 1, 2, 1, 5, 1958-09-15, 1, 2005-06-01, 46.7104722793
...
```

Kuva 24. Esimerkki CSV-tiedostossa olevasta metadatatista [66]

Deploy.py-tiedostoon tehtiin sama DLTK-moduulin polun lisäys kuin train.py-tiedostoon. Ennen validointiajua opetetun neuroverkkomallin polku konfiguroitiin ja tulokseksi saatiin sama tarkkuus kuin esitetyissä testituloksissa, eli 0.964 [66]. Tällä testauksella saatiin varmistettua, että halutut paketit pystyttiin asentamaan kyseiseen ympäristöön, ja ne myös toimivat halutulla tavalla. Samankaltainen ympäristö pystytettiin myöhemmin Watson Studiossa Notebookien avulla.

### 5.3 DICOM-tiedostojen siirtäminen ja tarkastelu

Työhön valittiin Cancer Imaging Archive -sivustolta Lung Image Database Consortium -kuvakokoelma (LIDC-IDRI), joka koostui diagnostisista ja keuhkosityövän rintakehän tietokonetomografioista, joihin on annotoitu leesiot. Tietokannan luomiseen oli osallistunut seitsemän akateemista keskusta ja kahdeksan lääketieteellistä kuvantamisyhtiötä. Tietokanta koostui 1018 tutkimustapauksesta ja oli kooltaan 124 Gt. Sivuston mukaan potilaiden määrä oli 1010, ja kuvat muodostuivat 1308 tutkimuksesta sisältäen 1018 TT- ja 290 CR/DX-kuvaa. [96.]

Kun haluttu DICOM-data oli valittu luotiin IBM:n analytiikan arkkitehdin Jukka Ruosen kanssa virtuaalikone Linux pohjaisella Ubuntu-alustalla tiedostojen lataamista varten.

Käyttöjärjestelmän versio oli 18.04.2 LTS ja datan latausta varten varattiin 500 G levytilaa. Virtuaalikonetta hallinoitiin terminaalin kautta SSH-yhteydellä. Cancer imaging archive -sivustolta tiedostojen lataaminen tapahtuu NBIA retriever -ohjelmalla, mikä ladattiin koneelle terminaalin kautta sivuston ohjeiden mukaan. Ohjelma ladattiin koneelle yhdessä manifest-tiedoston kanssa, joka sisälsi latausohjeet halutuista tiedostoista ohjelmalle. NBIA retriever -ohjelma ei suostunut käynnistymään terminaalin kautta virtuaalikoneessa. Ohjelmaa varten ladattiin virtuaalikoneelle Java-komennolla default-jre, koska ohjelma ei onnistunut löytämään manifest-tiedostoa. NBIA retriever -ohjelmaa ei onnistuttu käyttämään javan latauksesta huolimatta.

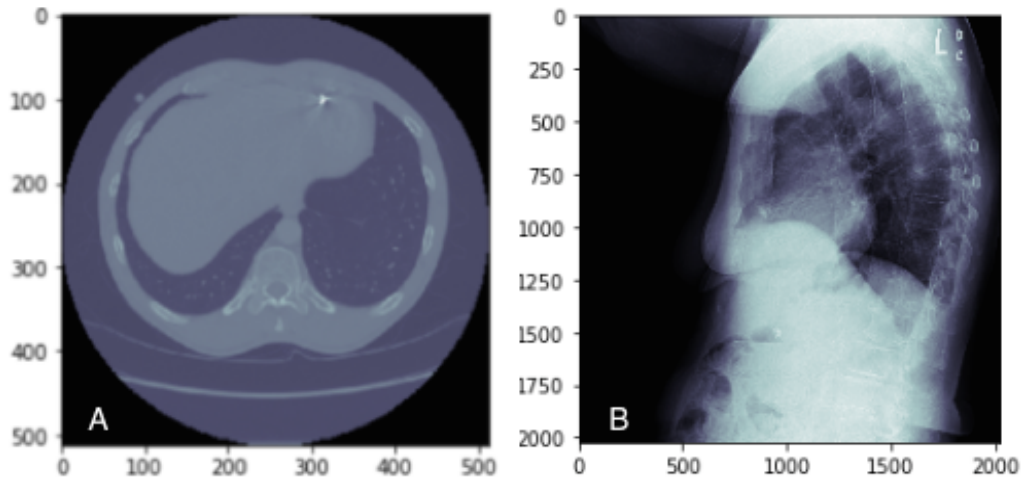
Tiedostojen latausongelma ratkaistiin ottamalla Windows-käyttöjärjestelmää käyttävä kone käyttöön etäyhteydellä. Etäyhteydellä saatiin näkyviin ruudulle Windows-koneen graafinen käyttöliittymä. Windows-koneeseen mountattiin, eli liitettiin hakemistopuuhun, virtuaalikoneessa oleva dataa varten varattu levytila. Mounttaamisella virtuaalikoneen datalevy näkyi Windows-koneessa omana levytilana, jolloin Windows-koneesta ei tarvinnut varata samaa määrää tallennustilaa tiedostojen latausta varten. Näin varmistettiin, että tiedostot siirtyvät suoraan virtuaalikoneeseen, jolloin poistui kaksi vaihetta työlistalta: tiedostojen uudelleensiirto Windows-koneesta virtuaalikoneeseen ja tiedostojen poisto Windows-koneesta. NBIA retriever -ohjelma ladattiin yhdessä manifest-tiedoston kanssa Windows-koneelle. Ohjelma saatiin ladattua ja käynnistettyä graafisen käyttöliittymän kautta. Ohjelman käynnistymisen jälkeen halutut DICOM-tiedostot laitettiin latautumaan virtuaalikoneeseen.

IBM Cloud Object Storage (COS) luotiin bucket DICOM-tiedostoja varten. COS:iin luotiin HMAC-tunnukset, jotka mahdollistavat yhteyden luonnin COS:in ja halutun koneen välille. Tässä työssä luotiin yhteys virtuaalikoneen ja COS:in sekä PowerAI:n ja COS:in välille. Yhteys luotiin käyttämällä avoimen lähdekoodin s3cmd-komentorivin apuohjelmaa, joka tarvitsee avuksensa Python-ohjelman sekä sen dateutil- ja setuptools-moduulit. GitHubista ladattiin s3cmd varten esimerkki konfiguraatitiedosta, johon syötettiin COS-tunnukset (HMAC) yhteyden muodostusta varten. Lisäksi lisättiin s3cmd-polku kotivalikkoon, jotta sen käyttö olisi helpompaa. Yhteyden tarkistamista varten käytettiin s3cmd ls -komentoa, jolloin komentoriville tuli näkyviin COS:issa olevat tallennusalueet ("bucketit"). Virtuaalikoneesta ladattiin COS:iin DICOM-tiedostot "objekteiksi". Virtuaalikoneeseen ei mountattu COS:ia näkymään omana levynä sillä tarkoitus on myöhemmin

tutkia, kuinka DICOM toolkitilla on mahdollista siirtää DICOM-tiedostoja PACS-järjestelmästä COS:iin. COS:in mounttaaminen on kuitenkin hyvä pitää mielessä, jos PACS:in ja tietokoneen X välille luodaan suojattu yhteys datalouhinnan tai DICOM-tiedostojen pyyntöjä varten. Näin DICOM-tiedostot saadaan suoraan COS:iin ilman ylimääräisiä latauksia, jolloin myös säästetään muistia tietokoneesta X.

PowerAI:hin yritettiin mountata COS näkymään levynä, jolloin tiedostojen siirtäminen/hyödyntäminen PowerAI:ssa olisi ollut luultavasti nopeampaa ja helpompaa. Mounttaaminen ei onnistunut ja asiasta konsultoitii IBM:n asiantuntijaa Jukka Ruposta. Kävi ilmi, että tässä projektissa käytetty PowerAI-palvelin oli suljettu ympäristö ja tästä syystä COS:in mounttaaminen ei onnistunut. On kuitenkin olemassa työkaluja ja ohjelmia, jolla mounttaaminen olisi voitu toteuttaa. Se olisi kuitenkin vaatinut kernelin uudelleenkäynnistystä. Saaduilla tunnuksilla oli mahdollista päästä järjestelmään, joka ohjasi varattuun konttiin PowerAI:ssa. Näin ollen koko PowerAI:n uudelleen käynnistämiseen ei ollut oikeuksia. Konttiin pystyi lataamaan tiedostoja vain https-protokollalla, joten käytetyssä PowerAI:ssa päädyttiin hyödyntämään s3cmd-apuohjelmaa.

Tiedostojen lataamisen jälkeen COS:n haluttiin varmistaa DICOM-tiedostojen muoto ja niitä yritettiin lukea Pydicom-kirjastolla käyttäen Notebookia Watson Studiossa. Ongelmaksi muodostui se, että DICOM-tiedostojen lukeminen suoraan COS:sta Pydicom-kirjaston avulla tuotti Notebookille puskuroidun "streaming\_body" -objektin, joka ei sisällä DICOM-tiedoston metatietoja. Pydicom-kirjasto tarvitsee näitä metatietoja DICOM-kuvien käsittelemiseksi. Ongelmasta konsultoitii Jukka Ruposta, ja hän ehdotti, että käsiteltävä DICOM-tiedosto kannattaa ladata ensin Watson Studion Notebook-ympäristön paikalliselle levyille ja lukea sitten sieltä tiedostona (ei "objektina") Pydicom-kirjaston avulla, jolloin myös metatiedot säilyvät. Tarkemman ohjeistuksen löytää Ruosen githubista nimellä notebook-dicom-from-cos. Yhteys Notebookin ja COS:in välillä luotiin samoilla HMAC-tunnuksilla, joista kerrottiin aikaisemmin. Osa DICOM-kuvista ladattiin ja luotiin esitys (kuva 25) matplotlib-kirjaston avulla. Samalla haluttiin tutkia, sisältääkö kuvat mahdollisesti piilometadatta vai onko kuvien anonymisoinnissa onnistuttu. Kuvan TT-kuvaan on lisätty mustaa aluetta kuten myös DX-kuvalle. Kuitenkin DX-kuvaan on jäänyt metadatta oikeaan yläkulmaan.



Kuva 25. Watson Studiossa tulostetut saman potilaan TT- ja DX-kuvat.

Pydicomilla onnistuttiin lukemaan DICOM-tiedoston metadata anonymisoinnin tarkistamiseksi (kuva 26). Kuva koostuu vain osasta tageistä anonymisoinnin demonstroimiseksi. Tulostetun metadatan pixel\_array on tulostettuna kuvan 20 DX-kuvana. Potilaan ID on muutettu tunnistetiedoksi, jota käytetään kansionimenä.

```

(0008, 0060) Modality                CS: 'DX'
(0008, 0070) Manufacturer             LO: 'GE MEDICAL SYSTEMS'
(0008, 0090) Referring Physician's Name PN: ''
(0008, 1090) Manufacturer's Model Name LO: 'Revolution XQi ADS_28.2'
(0008, 1155) Referenced SOP Instance UID UI: 1.3.6.1.4.1.14519.5.2.1.6279.6001.165159452785720376135411192086
(0010, 0010) Patient's Name           PN: ''
(0010, 0020) Patient ID               LO: 'LIDC-IDRI-0001'
(0010, 0030) Patient's Birth Date     DA: ''
(0010, 0040) Patient's Sex            CS: ''
(0010, 1010) Patient's Age            AS: ''
(0010, 21d0) Last Menstrual Date      DA: '20000101'
(0012, 0062) Patient Identity Removed CS: 'YES'
(0012, 0063) De-identification Method LO: 'DCM:113100/113105/113107/113108/113109/113111'
(0013, 0010) Private Creator          LO: 'CTP'
(0013, 1010) Private tag data         LO: 'LIDC-IDRI'
(0013, 1013) Private tag data         LO: '62796001'
(0018, 0015) Body Part Examined       CS: 'CHEST'
(0018, 0020) Body Part Examined       CS: 'CHEST'

```

Kuva 26. Anonymisoinnissa osa tiedoista on päätetty jättää tyhjiksi.

Cancer imaging archive –sivustolta [96] ladattiin 2 erillistä CSV-tiedostoa, jotka oli muodostettu käyttäen potilas-ID:tä referenssinä. Toisessa tiedostossa oli modality- ja patient- tiedot ja toisessa diagnooseista tietoa. Ladatun datan muodostuminen tarkistettiin ja huomattiin, että se muodostui 1 018 TT-, 273 DX- (Digital X-Ray), 90 SEG- (TT-Segmentointi) ja 53 CR (Computed Radiography) -tutkimuksesta 1 012 potilaan kuvantamistutkimuksista DICOM-muodossa.



Koska kaikille potilaille ei löytynyt diagnoosia CSV-tiedostosta, päädyttiin valitsemaan PowerAI:hin siirrettäviksi tiedostoiksi vain TT-kuvat, joissa oli keuhkosityövän diagnoosi numero 1-3. Kävi kuitenkin ilmi, että kansiot muodostuivat siten, että yläkansion nimenä oli potilas-ID ja alakansion nimi oli tutkimusnumero, joka sisälsi kuvaussarja-UID-nimillä eri radiologisilla modalityteilla tehdyt kuvantamiset. CSV-tiedostoissa ei ollut tietoa tutkimusnumerosta vaan pelkästään kuvaussarja-UID-tieto muiden tietojen lisäksi. Ratkaisuksi luotiin Watson Studiossa Notebook, johon ladattiin molemmat CSV-tiedostot. CSV-tiedostoista poimittiin pandas-kirjaston avulla taulukkoon potilas-ID:t, joilla oli keuhkosityöpädiagnoosi. Taulukko muodostui potilas-ID:stä ja diagnoosi numerosta. Tämä taulukko yhdistettiin toisesta CSV-tiedostosta muodostettuun taulukkoon, mikä sisälsi eri modalityteettien tiedot. Yhdistetystä taulukosta valittiin seuraavaan taulukkoon vain TT-kuvantamisella tehdyt tiedot. Seuraavaksi luotiin taulukko, joka koostui potilas-ID:stä ja sitä vastaava kuvaussarja-UID:stä.

Watson Studiossa luotiin yhteys COS:issa olevaan ”buckettiin”, jossa DICOM-tiedostot sijaitsivat. Tulostettiin yksi tiedosto, jotta nähtiin, miten tiedostopolku muodostuu. Taulukosta, joka koostui potilas-ID:stä ja kuvaussarja-UID:stä luotiin lista, jonka avulla luotiin silmukka, joka keräsi uuteen listaan tiedostopolut, joka vastasi potilas-ID:tä sekä päättyi kuvaussarja-UID:hen. Kyseisestä tiedostopolku-listasta luotiin silmukan avulla tekstitiedosto. Tekstitiedostoon luotiin tieto silmukalla erillisille riveille komento, jossa ensin luotiin kansio potilas-ID:lle ja seuraavalle riville komento, joka latsi COS:ista halutun TT-kuvia sisältävän tiedoston luotuun kansioon. Kolme ensimmäistä kansion luonti ja kansion lataus -komentoa ajettiin PowerAI:ssa testinä ja huomattiin, että on käytettävä `s3cmd get -recursive` -komentoa latauksessa. Lisäys tehtiin Notebookissa silmukan avulla ja luotiin uusi tekstitiedosto komennoista. PowerAI:hin luotiin tyhjä tekstitiedosto, johon kopioitiin Notebookissa luotu tekstitiedoston sisältö. Tekstitiedosto muutettiin ajettavaan muotoon ja ajettiin PowerAI:ssa. Kansiot latautuivat hyvin PowerAI:hin. Tiedostoja ei poistettu COS:sta.

#### 5.4 Watson OpenScale

Opetettujen koneoppimisen mallien toimintaa voidaan arvioida kolmen IBM Cloudin palvelun avulla. Watson OpenScalea käytetään mallin tarkkuuden arviointiin, Watson

Machine Learningia mallien tallentamiseen ja suorittamiseen sekä Cloud Object Storagea mallin kehityksen tietovarastona.

Työkalujen toimivuutta lähdettiin ensin testaamaan OpenScalen oman tutorialin avulla, joka tarjoaa step by step -ohjeistuksen kaikista ympäristön kokoamiseen tarvittavista vaiheista. Alkupään asennukset onnistuivat suunnitellusti, mutta eteneminen pysähtyi Monitoring-vaiheessa. Korjausyrityksistä huolimatta mallia ei saatu työstettyä tästä pisteestä eteenpäin.

Toinen testaus yritettiin tehdä German Credit Risk -mallilla ja -datalla, jossa arvioidaan lainanhakijan luottokelpoisuutta 20 muuttujan avulla. Malli ajettiin Watson Studioon Jupyter Notebookin avulla [97]. Asennuksessa oli hieman ongelmia yhteensopimattomien ohjelmaversioiden takia, mutta ongelmat saatiin korjattua vaihtamalla skriptiin oikeat versiot ja ajamalla komennot läpi uudestaan. Malli kuitenkin pysähtyi samaan kohtaan kuin IBM Cloudin kautta tehtävä demo. Työtilat asennettiin varmuuden vuoksi uudestaan siltä varalta, että niiden yhdistämisessä olisi tapahtunut virhe.

Toinen ohjelmistotestaus saatiin suoritettua paremmalla menestyksellä. OpenScale tutorial oli päivitetty uuteen versioon, eikä siinä enää esiintynyt etenemistä pysäyttäviä virheitä. Mallin luominen, tarkisteltavien parametrien määrittäminen ja tulosten analysointi saatiin suoritettua onnistuneesti. Ohjelmalla oli aluksi hankaluuksia muodostaa raportteja yksittäisten vastausten tuloksista, mutta tilanne korjaantui itsestään noin puolen tunnin kuluttua.

Seuraava vaihe oli kokeilla OpenScale-testauksen soveltamista kuvia (images) luokittelevaan malliin. Esimerkkimallia alettiin aluksi työstämään käyttämällä Scikit-learn työkalua, mutta kun tätä mallia yritettiin lukea OpenScaleen, ei sen määrittämissä ollut valittavissa "image"-vaihtoehtoa. Avuksi pyydettiin jälleen Jukka Ruponen, joka auttoi selvittämään, mistä ongelmat aiheutuivat.

Syitä tutkiessa ilmeni, että OpenScalen image-mallien arviointi tukee tällä hetkellä vain Kerasilla rakennettuja malleja [98]. Tästä johtuen kyseinen ominaisuus ei ollut käytettävissä Scikit-learnilla rakennetulle esimerkkimallille. Ruponen teki aiemmin

julkaisemaansa Keras-Notebookiin tarvittavat lisäykset mallien yhdistämistä ja datan käsittelyä varten. Tämän jälkeen sitä voitiin käyttää palvelun testaamiseksi.

Seuraavaksi testattiin Kerasilla toteutettua neuroverkkoa ajamalla Watson Studion data-alustalla läpi *Tutorials to Artificial Neural Networks with Watson Data Platform* [99]. Harjoitus koostui kahdesta Jupyter Notebookista: *Introduction to Neural Networks with Keras.ipynb* sekä *Detect handwritten digits with Keras.ipynb*:stä. Koska *Introduction to Keras* on ainoastaan pohjustava harjoitus, ei tässä käydä sen sisältöä tarkemmin läpi.

*Detect handwritten digits with Keras Notebook* avustaa rakentaamaan neuroverkon ja luomaan mallin käsinkirjoitettujen numeroiden tunnistamiseen. Aluksi neuroverkon luomista varten ladattiin tarvittavat *numpy*- ja *matplotlib*-kirjastot sekä Kerasin dataseteistä *MNIST*-datasetti, joka sisältää 60 000 kappaletta 28x28 pikselin kokoisia mustavalkoisia kuvia 0-9 välillä sekä luokkatiedon kuvassa olevasta numerosta. Lisäksi datasettiin kuuluu 10 000 kuvan testisetti, jolla opetetun neuroverkon toimivuus voidaan varmistaa. Mallin treenausta varten kuvat formatoitiin 784-kokoiseksi yksiulotteiseksi vektoriksi, jolloin kuvista muodostui 60000x784-kokoinen matriisi. Luokkatiedot muutettiin *One-Hot*-formaattiin, jolloin niistä syntyi 10x10-kokoinen yksikkömatriisi.

Mallin opettamista varten luotiin neuroverkko, joka koostuu 784 pikseliä leveästä syötekerroksesta, kahdesta 512 neuronin piilokerroksesta sekä 10 neuronin tulostekerroksesta. Tämän neuroverkon häviöfunktiona käytettiin *Categorical Cross-Entropy*ä, optimointialgoritmina *Adama* ja verkon tarkoituksiksi määriteltiin tarkkuuden arviointi. Määrittelyjen jälkeen verkko päästiin kouluttamaan aiemmin ladatulla datalla, jonka jälkeen mallin suorituskyky voitiin arvioida. Testien tulokseksi saatiin arvo 0.062 ja tarkkuudeksi 0.980.

Koska mallia haluttiin arvioida *OpenScalen* avulla, opetettu malli oli saatava tallennettua ja julkaistua *Watson Machine Learning* -palvelussa. Ensimmäiseksi opetettu malli tallennettiin *Watson Machine Learning* -palveluun siten, että Notebookiin lisättiin *WML client*, joka yhdistettiin *IBM Cloud*issa jo aiemmin luotuun *WML*-palveluun ja lisättiin siihen *WML*:n vaatimat tunnistetiedot (metadata). Tämän jälkeen opetettu malli voitiin tallentaa ja julkaista *WML*-alustalla ja varmistaa sen toiminta pisteytystestillä sekä varmistaa, että tulokset vastaavat niitä, jotka saatiin ajamalla malli suoraan Notebookilla.

Seuraavaksi malli voitiin lisätä Watson OpenScaleen sen selainpohjaisen käyttöliittymän kautta. OpenScale "Configure" -välilehdeltä valittiin ensin käyttöön Watson Machine Learning -palvelu ja valittiin sitten koulutettu ja julkaistu malli. Monitoroitavaksi valittiin edellä mainittu malli, ja OpenScalen vaatimana tietovarastona käytettiin palvelun tarjoamaa ilmaista Lite-versiota.

Tämän jälkeen voitiin Notebookin kautta luoda yhteydet OpenScaleen lataamalla OS client sekä yhdistämällä palvelut. Tämä yhteys on luotava, jotta OS:lle voidaan lähettää tietosisältö WML:n käyttämästä datasta, koska ilman sitä OpenScale ei tiedä, mitä parametreja sen kuuluu valvoa. Näiden määrittelemiseksi OS:lle lähetettiin WML:n kautta syötetiedot, johon määriteltiin ensin kentät ("fields"), joka pitää sisällään jokaisen lukujoonossa olevan pikselin ("p1", "p2", ... "p784") sekä arvot ("values"), joka sisältää kaikkien pisteiden väriarvon. Vastaustiedoiksi kenttään ('fields') määriteltiin ennuste ('prediction'), ennusteluokat ('prediction\_classes') sekä todennäköisyys ('probability') ja arvoiksi ('value') tieto kuviossa olevasta numerosta.

Kun OpenScale oli saanut tarvitsemansa tiedot, voitiin seurantatiedot määritellä loppuun selaimen kautta. Implementations-välilehdeltä valittiin käyttöön otettava malli ja siirryttiin sen konfigurointiin. Koska Notebookin kautta OS oli saanut jo tarvittavat syötteet, määrittely aloitettiin valitsemalla syötteen datatyyppiä 'Image' ja algoritmityyppiä 'Multi-class classification'. Ulostulodatasta valittiin ennusteen ominaisuudeksi 'prediction\_classes' ja ennusteen sisältäväksi tiedoksi 'probability'. Tämän jälkeen konfigurointi oli valmis.

Oli alun perin tiedossa, että kuvista ei pystytä arvioimaan oikeudenmukaisuutta (fairness), mutta paikkansapitävyys (accuracy) piti olla kuville tuettu ominaisuus. Kenttä ei kuitenkaan ollut valittavissa, mikä tarkoittaa sitä, että ominaisuus ei ollut käytettävissä, eikä kuvia voitu näillä OpenScalen asetuksella käsitellä, ainakaan tällä hetkellä.

Päätettiin, että mallia yritetään arvioida vaihtamalla datatyyppiä 'Numeric/categorical'. Tämä ei kuitenkaan onnistunut käyttämällä suoraan aiempaan Notebookiin määriteltyjä tietoja, vaan malliin piti tehdä muutoksia. Algoritmina pidettiin sama 'Multi-class classification'.

Numeric/categorical-tyyppisen mallin arviointi vaatii myös käytetyn treenausdatan antamisen OpenScalelle, jolloin OS pitää yhdistää tietovarastoon, josta kyseinen datasetti löytyy. Tätä varten mallin treenausdata siirrettiin Cloud Object Storageen, josta saatiin osoitettua OpenScalelle sen tarvitsema treenausdata.

Tämän jälkeen päästiin määrittelemään seurantaan vaadittuja tietoja. Ensin valittiin treenausdatasta vastauksia sisältäväksi sarakkeeksi 'class'. Ulostulodatasta ennusteen sisältäväksi ominaisuudeksi valittiin 'prediction\_classes' ja tekoälyn luomaksi ennusteeksi 'probability'. Mallin syötetiedoksi määriteltiin kaikki pikselikentät (784 kpl). Mallin antamien vastausten tekstiselitteitä kuvaava kategoria jätettiin tyhjäksi, koska kyseistä ominaisuutta ei tarvita tämän mallin arvioinnissa. Valintojen jälkeen asetukset tallennettiin ja seuraavaksi päästiin määrittelemään tarkkuuden arviointia.

Ensin täytyi asettaa halutut arvot, joiden mukaan OpenScale seuraa mallin tarkkuutta. Ensimmäiseksi määritellään vaadittu tarkkuus prosentteina, johon valittiin 90-prosenttia (Good). Seuraavaksi malliin määriteltiin arviointiin vaadittujen näytteiden määrä, johon valittiin alarajaksi 10 ja ylärajaksi 10 000. Ennen arviointia malli vaati vielä palautedatan (additional feedback data), joka pitää syöttää joko suoraan koneelta tiedostona tai tietokannan kautta. Lopputuloksena mallista saatiin analysoitua tarkkuustieto, jonka tuloksia on esitetty liitteessä 2.

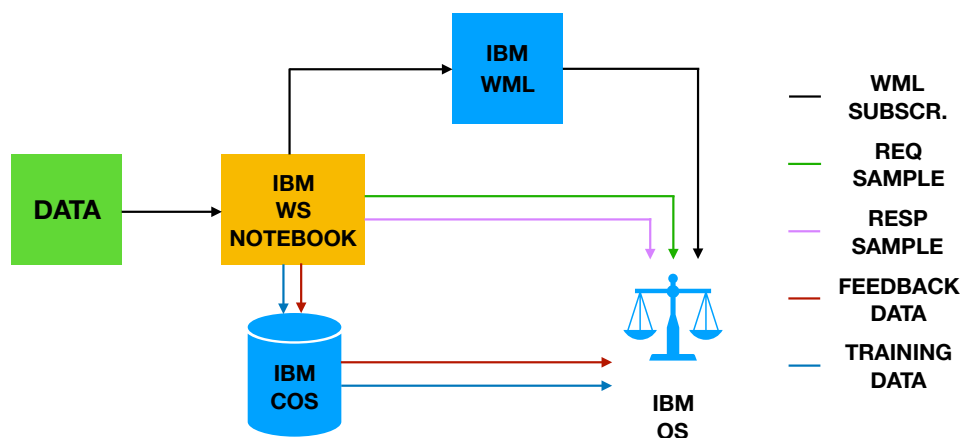
## 5.5 Tulokset

Asiantuntijahaastatteluiden tulosten perusteella voidaan todeta, että DICOM-tiedostoja pystytään anonymisoida riittävällä tarkkuudella, jotta niitä voidaan siirtää PACS-järjestelmän ulkopuolelle. Voidaan myös todeta, että IBM COS soveltuu DICOM-tiedostojen säilytykseen ja Watson Studioon liitettyllä Notebookilla tiedostoja voi avata ja tarkastella Pydicom-kirjaston avulla. Näin ollen IBM:n palveluiden resurssit ovat riittävät suurten tiedostojen lataamiseen, vastaanottamiseen, käsittelemiseen ja siirtämiseen. PowerAI-alustaan on mahdollista luoda oma oppimisympäristö mallinopettamista varten. Lisäksi IBM:n palveluista on mahdollista siirtää tiedostoja PowerAI-alustan ympäristöön. Jos ympäristöä ei olla suljettu täysin, on mahdollista mountata COS haluttuun ympäristöön tiedostojen hyödyntämistä varten.

Lääketieteellisiin kuviin voi jäädä metadataa, kuten DX-kuvaan oli jäänyt tieto kuvantamissuunnasta. Tiedon tallennuskohta on joko valmistajan määrittelemä tai kuvantamisyksikön asettama. Lisätty tieto tulostui kuvan päälle minkä johdosta sen poistaminen voi johtaa tiedon menetykseen. Kuitenkin DX-kuvaan on jäänyt metadataa oikeaan yläkulmaan. DICOM-tiedostot olivat valmiiksi anonymisoitu, joten sen testausta ei päästy kokeilemaan tällä datasetillä.

Watson OpenScalen tarvittavia ominaisuuksia ei ollut vielä otettu laajemmin käyttöön, vaikka dokumentaation mukaan kuvien tarkkuuden analysoinnin piti jo olla palvelun avulla mahdollista. Accuracy-kenttä oli kuitenkin inaktiivinen, joten ominaisuuden toimivuutta ei kyetty tällä tavoin testaamaan.

Sama data voitiin kuitenkin analysoida numeerisena/kategorisena mallina, joka vaatii datan muokkaamista sille sopivaan muotoon. Kyseinen menetelmä vaatii jo perusteellisempaa osaamista datatieteistä sekä OpenScalen toiminnallisuuksien ymmärtämistä. Proessin vaiheiden kuvaamiseen luotiin kaavio (kuva 27), jonka tarkoitus on selventää datan kulkemisen eri vaiheita ja helpottaa OpenScalen toimintaperiaatteiden ymmärtämistä tulevassa kehitystyössä.



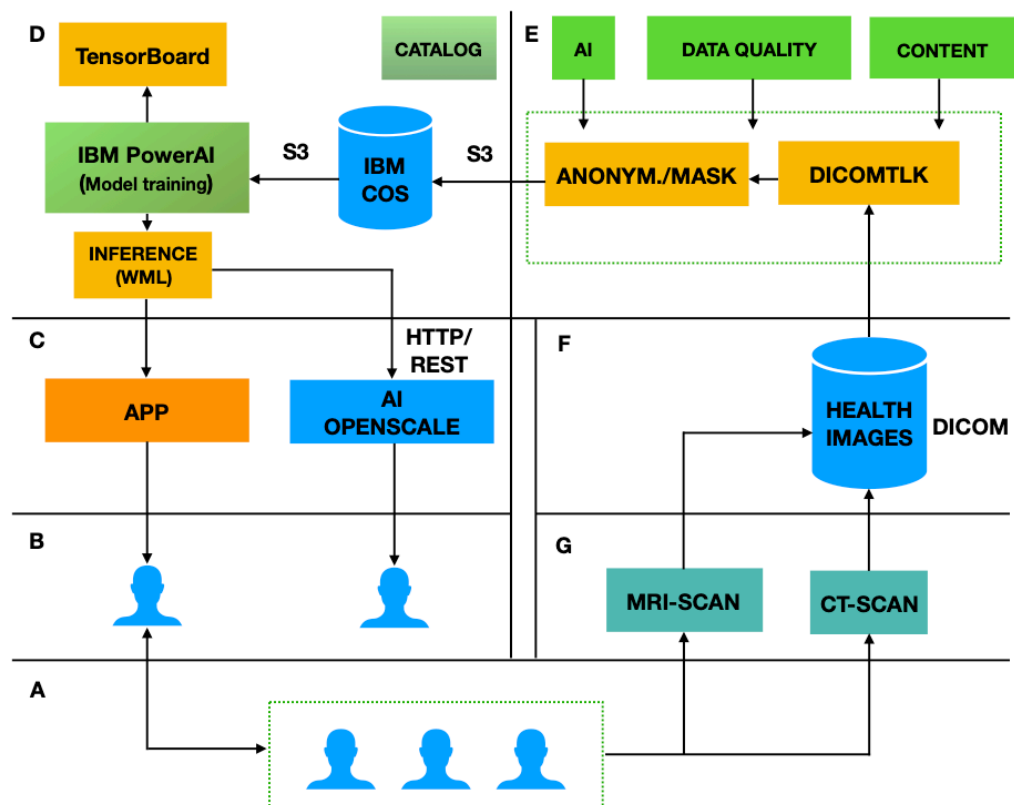
Kuva 27. Datan liikkuminen OpenScalen ja muiden palveluiden välillä.

Kuvassa 27 Watson OpenScale yhdistyy Watson Machine Learningiin (sininen laatikko) ja Cloud Object Storageen (sininen säiliö) Watson Studioon (keltainen laatikko) kautta. Määrittelyvaiheessa malli kehitetään ja opetetaan Notebookilla (keltainen laatikko)

opetusdatan avulla (vihreä laatikko). Osana OpenScale-määrittelyä sille kerrotaan mm. mallin edellyttämän input-datan muoto ("Request Sample"), mallin palauttaman vastauksen muoto ("Response Sample") sekä käytetty treenidata ("Training Data"). Määrittelyn jälkeen OpenScale kuuntelee Watson Machine Learning -palvelua monitoroitavan mallin osalta (tässä tapauksessa Keras-malli). Malliin lähetetyt pyynnöt (request) ja näihin palautetut vastaukset (response) tallentuvat myös OpenScale-tietokantaan ("payload logging") monitorointia varten. Jotta OpenScale voi arvioida mallin tarkkuutta, sille on annettava myös testausdataa ("Feedback Data"), jota vasten tarkkuuden arviointi OpenScalella tapahtuu.

Jukka Ruosen kanssa käydyssä keskustelussa arkkitehtuurikuvan parantamisesta muodostui logical deployment zones -arkkitehtuuripiirros (kuva 28). Piirros sisältää osaluokkia, jotka rajattiin tämän työn ulkopuolelle mutta otettiin mukaan työn jatkoa ajatellen. A-lohko koostuu potilaista. B-lohko koostuu hoitajasta tai lääkäreistä. Terveystieteiden ammattilaiset ovat yhteydessä potilaisiin. Lääkärit ja hoitajat käyttävät työssään C-lohkon tekoälyä hyödyntävää sovellusta. Ratkaisun toimintaa ja luotettavuutta arvioiva henkilö käyttää C-lohkon testausvälinettä (OpenScale).

D-lohko koostuu mallin kehittämiseen ja opettamiseen tarvittavista palveluista. Tässä lohossa TensorBoardilla pystytään graafisesti seuraamaan mallin oppimista. Mallin opetusdata siirtyy PowerAI:hin COS:ista. COS:ssa oleva data voitaisiin luetteloida "katalogissa" (catalog), helpottamaan datan valintaa. Lohko E on kliinisen potilasdatan toisiokäytön ympäristö, joka sijaitsee sairaalaympäristössä kliinisten järjestelmien ulkopuolella. E-lohossa tapahtuu DICOM-tiedostojen nouto sekä anonymisointi. Tässä lohossa voisi hyödyntää tekoälyä esimerkiksi varmistamaan datan laatu sekä anonymisoinnin. Lohkot F ja G sijaitsevat PACS-järjestelmässä. F-lohko koostuu PACS-arkistosta, johon siirtyy G-lohkon kuvantamismenetelmistä ja radiologian osastolta DICOM-tiedostoja. A-lohkon henkilöt (potilaat) käyvät G-lohkon tutkimuksissa.



Kuva 28. Logical Deployment Zones

Logical Deployment Zones kuvaa käytetään jatkossa konseptin jatkojalostamiseen. Tämän työn ulkopuolelle jääneistä lohkoista ja niiden välisistä yhteyksistä tehdään lisätutkimusta.

## 6 Pohdinnat ja yhteenveto

Tulosten perusteella voidaan todeta, että suunniteltu konsepti on toteutettavissa, vaikka sitä ei kaikilta osin ehditty toteuttaa. Jokainen komponentti saatiin toimimaan omina kokonaisuuksinaan, vaikka OpenScale vaati ohjelmiston kehityksen keskeneräisyyden vuoksi alkuperäisestä suunnitelmasta poikkeavaa lähestymistapaa. Tämä toteutettavuustutkimus on herättänyt suurta kiinnostusta, ja projektin tuloksia tullaan mahdollisesti esittelemään IBM:n tulevissa tilaisuuksissa.



Jatkokehityksessä tulee huomioida, että lääketieteelliseen käyttöön tarkoitetun tekoälyn rakentaminen vaatii suuren määrän hyvälaatuista opetusdataa. Kuvadatan valinnassa on huomioitava, miten kuva on muodostettu ja minkälaista metadataa tiedosto sisältää, koska eri toimipisteiden kuvantamisessa käytetyt asetukset ja annotointi voi poiketa merkittävästi toisistaan. Tähän vaikuttavat useat eri seikat, kuten laite, siihen tehdyt ja valmistajan asettamat säädöt, laitteelle tehdyt huoltotoimenpiteet sekä käyttäjien henkilökohtaiset mieltymykset asetusten valinnassa.

Kuville tehdään monesti jälkikäsitteilyä, koska kuvissa esiintyy usein kohinaa ja muita häiriöitä. Kuvien muokkauksessa on mahdollista menettää arvokasta tietoa, jos sen tekee ei-ammattilainen tai huonoilla välineillä, esimerkiksi näytön resoluutio on heikentynyt. Kuvadatan harmaasävytasoja on enemmän, kuin silmällä on mahdollista nähdä. Vaikka segmentoinnilla pystytään poimimaan kuvista poikkeavaa tietoa, on mahdollista, että tekoäly löytäisi sieltä enemmän informaatiota.

Jos kuvantamisessa käytetään liian pientä tai suurta säteilyannostusta, kuvasta voi olla mahdotonta havaita tarvittavia asioita, eikä niitä voida enää korjata edes kuvankäsittelyn keinoin. Siksi on tärkeä pyrkiä lähtökohtaisesti laadukkaasti otettuihin kuviin, jotka vastaavat niille asetettuja vaatimuksia. On syytä pohtia, onko kuvien muokkaus tekoälykehityksessä hyödyksi vai haitaksi, sillä muokkaukset on suunniteltu palvelemaan ihmistä, ei konetta. Mikäli kone hyödyntää kuva-asetuksiin liittyvää metadataa osana oppimista, se voisi mahdollisesti suodattaa kuvia useammalla eri tavalla ja siten kerätä siitä enemmän informaatiota kuin ihminen. Koneoppiva malli voisi tulevaisuudessa olla jopa ihmistä parempi määrittämään optimaalisia asetuksia ja siten nopeuttaa kuvanmuokkausprosesseja.

Tietojen anonymisointi kuvista voi olla haastavaa, sillä tiedostot saattavat sisältää piiloteuttua dataa esimerkiksi kuvan pikselitiedoissa tai metadataan tallennetuissa piiloteuttuissa tageissa. Oman lisähaasteensa tuo se, että esimerkiksi tallennustapa sekä valikon kieli vaihtelee valmistajasta riippuen ja voi johtaa tietojen virheelliseen tallentamiseen. Koska DICOM-tiedoston mukana siirtyy kaikki sen sisältämä metadata, jota ei ole anonymisoitu, voi tietojen mukana siirtyä vahingossa sellaisia potilastietoja, jotka voivat johtaa potilaan tunnistamiseen ja näin ollen rikkoa lakia. DICOM-tiedostojen muuntaminen NIFTI-tiedostoiksi on yksi vaihtoehto anonymisointiin, mutta samalla menetetään paljon

tietoja, jotka voivat olla tärkeitä tekoälymallin opettamisessa. Laitteen tallentamiin tietoihin pitäisi perehtyä perusteellisesti, jotta sieltä kyettäisiin valitsemaan vain tarpeellinen ja mallin opetuksen kannalta relevantti tieto. Loput tiedot tulisi poistaa tai korvata siten, että tiedosto säilyy luettavassa muodossa.

Molemmissa asiantuntijahaastatteluissa tuli ilmi, että DICOM-tiedoston mukana siirtyvä piilodata voi sisältää potilaan henkilökohtaista tietoa, joko metadatan piilotagien tai pikselitietojen seassa. Pahimmassa tapauksessa tekoälymalli saattaisi oppia yhdistämään sensitiivistä tietoa esimerkiksi diagnooseihin ja vääristää tuloksia. Tämä vaatisi kuitenkin sen, että malli olisi ylioppinut yksipuolisesta datasta ja todennäköisyys tähän on olemattoman pieni, mutta silti huomioitava lääketieteellisessä syväoppimisen kehityksessä.

Joissain neuroverkkomalleissa on olemassa sisäänrakennettuja mekanismeja, joilla mallin toimintaa voidaan tarkkailla, mutta ulkopuolisia mekanismeja ei juurikaan ole. Lääketieteelliset sovellukset kuitenkin vaativat läpinäkyvyyttä ja tekoälyä hyödyntävien diagnoosin tekoa avustavien järjestelmien tarkkuutta on kyettävä tarkkailemaan ja muokkaamaan tarvittaessa. Tämä vaatii, että tuloksia pystytään jäljittämään riittävällä tarkkuudella ja tulokset on ilmaistu yksiselitteisesti.

Tarkasti opetettu neuroverkkomalli mahdollistaa sen hyödyntämisen erilaisissa käyttötarkoituksissa, kuten hoidon tukena, tutkimustyössä sekä osana ammatillista koulutusta. Internetistä on mahdollista löytää lääketieteellistä dataa, mutta sitä ei voida käyttää luotettavien mallien rakentamiseen, koska sen sisältöä on vaikea todentaa määrältään ja laadultaan luotettavaksi. Vääristynyt data johtaa koko neuroverkkomallin hylkäämiseen, koska sen antamalla väärillä tuloksilla voisi olla vakavia seurauksia. Toisaalta verkosta löytyvän datan avulla kuka tahansa voi tehdä alustavia kokeiluja ja neuroverkkomallien opetuksia. Lääketieteellisen datan oikeellisuuden ja mallien laadukkaan opettamisen varmistamiseksi on paras ratkaisu saada dataa suoraan sairaalan PACS-järjestelmästä tekoälylle. PACS-järjestelmän suorituskyky tulee aina priorisoida sairaalan käyttöön, eikä datan louhinta saa haitata sen toimintaa. Se, kuinka datan louhinta tulisi tapahtua, vaatii vielä lisätutkimusta. Tulevaisuudessa voisi miettiä, voisiko DICOM-tiedostoon lisätä tagin, joka ilmaisee sen soveltuvuuden tekoälykehitykseen. Tämä voisi tehdä louhintaprosessista helpompaa ja nopeampaa.

Tällä hetkellä HUS:lla on olemassa tutkimuksia varten oleva data-allas, joka on kliinisten järjestelmien ulkopuolella. Tämä tietosäilö on kuitenkin käytössä vain HUS:n omiin tutkimuksiin, ja data koostuu vain sen omissa kuvantamisyksiköissä tehdyistä tutkimuksista. Helsingin ja Uudenmaan sairaanhoitopiirin arkistoihin tallennetun datan määrä ja monipuolisuus on todennäköisesti riittävän suuri koneoppia hyödyntävien mallien rakentamiseen ja mallit voisivat olla käytettävissä maanlaajuisesti. Pienemmille sairaanhoitopiireille materiaalin vähyyys voi koitua ongelmaksi ja siten heikentää niiden kykyjä kehittää omia älykkäitä ratkaisuja. Tästä syystä olisi hyvä pohtia, olisiko kannattavaa perustaa koko Suomen kattava tutkimusarkisto, joka tukisi lääketieteellistä tekoälykehitystä ja auttaisi Suomea pysymään myös kansainvälisesti tekoälykehityksen kärkimaana.

## Lähteet

- 1 Tekoäly, koneoppiminen ja data yritysten liiketoiminnassa. 2018. Verkkoaineisto. Kauppalehti Gaala 2018. <<https://optiongaala.kauppalehti.fi/elisa/tekoaly-koneoppiminen-ja-data-yritysten-liiketoiminnassa>>. Luettu 3.4.2019.
- 2 Pitkänen, Manu. 2016. Microsoftin chatbotista tuli PR-katastrofi: Nettikeskustelijat opettivat sen rasistiksi. Verkkoaineisto. Afterdawn. <<https://fin.afterdawn.com/uutiset/artikkeli.cfm/2016/03/24/microsoftin-chatbotista-tuli-pr-katastrofi-nettikeskustelijat-opettivat-sen-rasistiksi>>. Luettu 3.4.2019.
- 3 Karkimo, Ari. 2018. Amazon yritti hyödyntää tekoälyä rekrytoinnissa – hyvällä ajatuksella oli ohrainen loppu. Verkkoaineisto. Markkinointi&Mainonta. <<https://www.marmai.fi/uutiset/amazon-yritti-hyodyntaa-tekoalya-rekrytoinnissa-hyvalla-ajatuksella-oli-ohrainen-loppu-6744454>>. Luettu 3.4.2019.
- 4 Käsälä, Samuli. 2018. Tutkimus: Tekoäly haistoi syövän lääkäreitä paremmin. Verkkoaineisto. Kauppalehti. <<https://www.kauppalehti.fi/uutiset/tutkimus-tekoaly-haistoi-syovan-laakareita-tarkemmin/dda2c9b5-6b07-35fd-ba7b-7b0b2bb1399e>>. Luettu 7.4.2019.
- 5 Ihme, Tiina. 2018. Tekoälystä uusia mahdollisuuksia lääketieteen kuvantamiseen. Verkkoaineisto. Oulun yliopisto. <<https://www oulu.fi/yliopisto/node/55327>>. Luettu 7.4.2019.
- 6 Lääkärit ja tekoäly olivat eri mieltä koomapotilaista – arvaatko kumpi osui oikeaan?. 2018. Verkkoaineisto. Tivi. <<https://www.tivi.fi/uutiset/laakarit-ja-tekoaly-olivat-eri-mieltä-koomapotilaista-arvaatko-kumpi-osui-oikeaan/926780c8-75c0-3f8f-ab26-3a5141f14cdd>>. Luettu 7.4.2019.
- 7 Lampela, Rosa. 2019. Tutkimus: Tekoäly toi 14% säästöt terveystieteiden potilaskustannuksiin. Tekniikka&talous. <<https://www.tekniikkatalous.fi/tekniikka/tutkimus-tekoaly-toi-14-saastot-terveyskeskuksen-potilaskustannuksiin-6758096>>. Luettu 7.4.2019.
- 8 Mustajoki, Pertti; Kaukua, Jarmo. 2008. Kuvantamismenetelmät. Verkkoaineisto. Kustannus Oy Duodecim. <[https://www.terveysportti.fi/terveyskirjasto/tk.koti?p\\_artikkeli=trg00002&p\\_hakusana=kuvantamismenetelm%C3%A4](https://www.terveysportti.fi/terveyskirjasto/tk.koti?p_artikkeli=trg00002&p_hakusana=kuvantamismenetelm%C3%A4)>. Luettu 20.4.2019.
- 9 Toikkanen, Ulla. 2016. Radiologien tarve kasvaa. Verkkoaineisto. Lääkärilehti. <<https://www.laakarilehti.fi/ajassa/ajankohtaista/radiologien-tarve-8232-kasvaa/>>. Luettu 15.5.2019.

- 10 Mustajoki, Pertti; Kaukua, Jarmo. 2008. Röntgenkuvaus. Verkkoaineisto. Kustannus Oy Duodecim. <[https://www.terveyskirjasto.fi/terveyskirjasto/tk.koti?p\\_artikkeli=trg00003](https://www.terveyskirjasto.fi/terveyskirjasto/tk.koti?p_artikkeli=trg00003)>. Luettu 15.5.2019.
- 11 Mustajoki, Pertti; Kaukua, Jarmo. 2008. Tietokonekerroskuvaus. Verkkoaineisto. Kustannus Oy Duodecim. <[https://www.terveyskirjasto.fi/terveyskirjasto/tk.koti?p\\_artikkeli=trg00004](https://www.terveyskirjasto.fi/terveyskirjasto/tk.koti?p_artikkeli=trg00004)>. Luettu 15.5.2019.
- 12 Mustajoki, Pertti; Kaukua, Jarmo. 2008. Varjoainekuvaukset. Verkkoaineisto. Kustannus Oy Duodecim. <[https://www.terveyskirjasto.fi/terveyskirjasto/tk.koti?p\\_artikkeli=trg00007](https://www.terveyskirjasto.fi/terveyskirjasto/tk.koti?p_artikkeli=trg00007)>. Luettu 15.5.2019.
- 13 Mustajoki, Pertti; Kaukua, Jarmo. 2008. Isotooppitutkimukset. Verkkoaineisto. Kustannus Oy Duodecim. <[https://www.terveyskirjasto.fi/terveyskirjasto/tk.koti?p\\_artikkeli=trg00008](https://www.terveyskirjasto.fi/terveyskirjasto/tk.koti?p_artikkeli=trg00008)>. Luettu 15.5.2019.
- 14 Mustajoki, Pertti; Kaukua, Jarmo. 2008. Kaikukuvaus. Verkkoaineisto. Kustannus Oy Duodecim. <[https://www.terveyskirjasto.fi/terveyskirjasto/tk.koti?p\\_artikkeli=trg00006](https://www.terveyskirjasto.fi/terveyskirjasto/tk.koti?p_artikkeli=trg00006)>. Luettu 13.5.2019.
- 15 Mustajoki, Pertti; Kaukua, Jarmo. 2008. Magneettikuvaus. Verkkoaineisto. Kustannus Oy Duodecim. <[https://www.terveyskirjasto.fi/terveyskirjasto/tk.koti?p\\_artikkeli=trg00005](https://www.terveyskirjasto.fi/terveyskirjasto/tk.koti?p_artikkeli=trg00005)>. Luettu 15.5.2019.
- 16 Soimakallio, Seppo; Kivisaari, Leena; Manninen, Hannu; Svedström Erkki; Tervonen, Osmo. 2005. Radiologia. WSOY.
- 17 Pancrias. Verkkoaineisto. Picturesso. <<https://www.picturesso.com/pics/pancreatic-duct-pancreas-mri-liver-trauma-55.html>>. Luettu 12.5.2019.
- 18 CAT Scan (computerized axial tomography). Verkkoaineisto. Johns Hopkins University. <<http://www.pathology.jhu.edu/pc/slides/ctscan.html>>. Luettu 12.5.2019.
- 19 Pancreatic Adenocarcinoma Ultrasound. Verkkoaineisto. Kidskunst. <<https://www.kidskunst.info/2/13082-pancreatic-adenocarcinoma-ultrasound.htm>>. Luettu 12.5.2019.
- 20 Carter, Christi; Veale, Beth. 2018. Digital radiography and PACS. Third edition. Elsevier.
- 21 HUS-Kuvantaminen. Verkkoaineisto. Helsingin ja Uudenmaan Sairaanhoidopiiri. <<https://www.hus.fi/hus-tietoa/materiaalipankki/esitysmateriaalit/Koottu%20tiedostokirjasto%20%20kaikki%20tapahtumat/05%20Mykkanen%20-%20Perusterveydenhuollon%20ja%20erikoissairanhoidon%20integraatio.pdf>>. Luettu 21.4.2019.

- 22 HUS Kuvantaminen – Hinnasto 2019 Sairaanhoidoalueet ja kunnat. Verkkoaineisto. Helsingin ja Uudenmaan Sairaanhoidopiiri. <<https://www.hus.fi/ammattilaiselle/hus-kuvantaminen/Hinnastot/HUS%20-alueen%20hinnasto%202019.pdf>>. Luettu 5.5.2019.
- 23 DICOM. 2019. Verkkoaineisto. National Electrical Manufacturers Association. <<https://www.dicomstandard.org/>>. Luettu 3.4.2019.
- 24 Kuvankatselujärjestelmät ja sairauskertomus sekä radiologisen informaation hallinta. 2019. Verkkoaineisto. Etelä-Pohjanmaansairaanhoidopiiri. <[http://www.epshp.fi/yksikoiden\\_sivut/sairaanhoidolliset\\_palvelut/radiologia/toiminnan\\_esittely/pacs\\_ris](http://www.epshp.fi/yksikoiden_sivut/sairaanhoidolliset_palvelut/radiologia/toiminnan_esittely/pacs_ris)>. Luettu 3.4.2019.
- 25 Lennes, Mietta. 2005. Puheen annotaatio. Verkkoaineisto. Lennes.github.io. <<https://lennes.github.io/puheen-annotaatio/node3.html>>. Luettu 5.5.2019.
- 26 HUS vahvistaa diagnostisten palveluiden johtamista. 2019. Verkkoaineisto. HUS. <<https://www.hus.fi/hus-tietoa/ uutishuone/Sivut/HUS-vahvistaa-diagnostisten-palveluiden-johtamista.aspx>>. Luettu 3.4.2019.
- 27 Wolbarst, Anthony; Capasso, Patrizio; Wyant, Andrew. 2013. Medical Imaging: Essentials for Physicians. Kanada: Wiley-Blackwell.
- 28 Jauhiainen, Jukka. 2007. Röntgenkuvaus, digitaalinen kuvaus ja tietokonetomografia. <<http://www.oamk.fi/~jjauhai/opetus/mittalaitteet/mittalaitteet07-v1.1.pdf>>. Luettu 12.5.2019.
- 29 Rajchl, Martin. 2018. An Introduction to Biomedical Image Analysis with TensorFlow and DLTK. Verkkoaineisto. Medium. <<https://medium.com/tensorflow/an-introduction-to-biomedical-image-analysis-with-tensorflow-and-dltk-2c25304e7c13>>. Luettu 4.3.2019.
- 30 Yleinen tietosuoja-asetus. 2016. Verkkoaineisto. EUR-lex. <<https://eur-lex.europa.eu/legal-content/FI/TXT/?uri=CELEX%3A32016R0679>>. Luettu 19.4.2019.
- 31 Laki potilaan asemasta ja oikeuksista. 1992. Verkkoaineisto. Finlex. <<https://www.finlex.fi/fi/laki/ajantasa/1992/19920785>>. Luettu 19.4.2019.
- 32 Laki sosiaali- ja terveydenhuollon asiakastietojen sähköisestä käsittelystä. 2007. Verkkoaineisto. Finlex. <<https://www.finlex.fi/fi/laki/ajantasa/2007/20070159>>. Luettu 20.4.2019.
- 33 Terveydenhuoltolaki. 2010. Verkkoaineisto. Finlex. <<http://finlex.fi/fi/laki/ajantasa/2010/20101326?search%5btype%5d=pika&search%5bpika%5d=terveydenhuoltolaki%20-%20L1P9>>. Luettu 19.4.2019.

- 34 Potilastietojen käyttö ja salassapito. Verkkoaineisto. HUS. <[https://www.hus.fi/potilaalle/potilasasiakirjat\\_tietojen\\_salassapito/potilastietojen-kaytto-salassapito/Sivut/default.aspx](https://www.hus.fi/potilaalle/potilasasiakirjat_tietojen_salassapito/potilastietojen-kaytto-salassapito/Sivut/default.aspx)>. Luettu 19.4.2019.
- 35 Laki viranomaisten toiminnan julkisuudesta. 1999. Verkkoaineisto. Finlex. <<https://www.finlex.fi/fi/laki/ajantasa/1999/19990621>>. Luettu 20.4.2019.
- 36 Toisiolaki. Verkkoaineisto. Sosiaali- ja terveysministeriö. <<https://stm.fi/sote-tiedon-hyodyntaminen>>. Luettu 20.4.2019.
- 37 Draft Ethics guidelines for trustworthy AI. 2018. Verkkoaineisto. European Commission. <<https://ec.europa.eu/digital-single-market/en/news/draft-ethics-guidelines-trustworthy-ai>>. Luettu 19.4.2019.
- 38 Ide, Harry. 2018. What causes AI boom and bust: a personal view of Artificial Intelligence history. Verkkoaineisto. Innovation observatory. <<http://www.innovationobservatory.com/node/243>>. Luettu 6.4.2019.
- 39 Ray, Shaan. 2018. History of AI. Verkkoaineisto. Toward Data Science. <<https://towardsdatascience.com/history-of-ai-484a86fc16ef>>. Luettu 6.4.2019.
- 40 Ala-Outinen Matti. 2018. Juttusarja: Tekoälyn perusteet AI, ML, Deep Learning, Neural Network. Verkkoaineisto. LinkedIn. <<https://www.linkedin.com/pulse/juttusarja-teko%C3%A4lyn-perusteet-ai-ml-deep-learning-matti-ala-outinen/>>. Luettu 6.4.2019.
- 41 Tekoäly. Verkkoaineisto. Itewiki. <<https://www.itewiki.fi/opas/tekoaly/>>. Luettu 4.4.2019.
- 42 Tekoälyn filosofiaa. Verkkoaineisto. Elements of AI. <<https://course.elementsofai.com/fi/1/3>>. Luettu 4.4.2019.
- 43 Russell, Stuart; Norvig, Peter. 2010. Artificial Intelligence, A Modern Approach. Third Edition. Pearson.
- 44 Koneoppiminen. Verkkoaineisto. Itewiki <<https://www.itewiki.fi/opas/koneoppiminen/>>. Luettu 7.4.2019.
- 45 Lehto, Martti; Neittaanmäki, Pekka; Nyrhinen, Riku; Ojalainen, Anniina; Pölönen, Ilkka; Rautiainen, Ilkka; Ruohonen, Toni; Tuominen, Heli; Vähäkainu, Petri; Äyrämö, Sami; Äyrämö, Sanna-Mari. 2018. Tekoälyn perusteita ja sovelluksia. E-kirja. <<https://helituominen.files.wordpress.com/2018/09/kirja1809.pdf>>. Luettu 7.4.2019.

- 46 Chollet, Francois. 2018. Deep Learning with Python. United State of America: Manning Publications Co.
- 47 Goodfellow, Ian; Bengio, Yoshua; Courville, Aaron. 2016. Deep Learning. E-kirja. MIT Press. <[deeplearningbook.org](http://deeplearningbook.org)>. Luettu 8.4.2019.
- 48 Vähäkainu, Petri; Neittaanmäki, Pekka. 2018. Verkkoaineisto. Tekoäly terveydenhuollossa. Verkkoaineisto. Jyväskylän yliopisto. <<https://www.jyu.fi/it/fi/tutkimus/julkaisut/tekes-raportteja/tekoaly-terveydenhuollossa.pdf>>. Luettu 28.4.2019.
- 49 Vázquez, Favio. 2017. Deep Learning made easy with Deep Cognition. Verkkoaineisto. Becoming Human. <<https://becominghuman.ai/deep-learning-made-easy-with-deep-cognition-403fbe445351>>. Luettu 4.5.2019.
- 50 Weisstein, Eric W. Convolution. Verkkoaineisto. MathWorld – A Wolfram Web Resource. <<http://mathworld.wolfram.com/Convolution.html>>. Luettu 24.5.2019.
- 51 Kiesiläinen, Jarno. 2016. Koneoppimisen hyödyntäminen konenäössä. Verkkoaineisto. Jyväskylän yliopisto. <<https://jyx.jyu.fi/bitstream/handle/123456789/52738/1/URN%3ANBN%3Afi%3Aju-201701161164.pdf>>. Luettu 6.5.2019.
- 52 Vähäkainu, Petri; Neittaanmäki, Pekka. 2018. Terveysthuollon alustat ja tekoäly. Verkkoaineisto. Jyväskylän yliopisto. <[https://www.jyu.fi/it/fi/tutkimus/julkaisut/tekes-raportteja/terveydenhuollon\\_alustat\\_ja\\_tekoaly.pdf](https://www.jyu.fi/it/fi/tutkimus/julkaisut/tekes-raportteja/terveydenhuollon_alustat_ja_tekoaly.pdf)> Luettu 6.5.2019.
- 53 Tejani, Shafeen. 2016. Machines that can see: Convolutional Neural Networks. Verkkoaineisto. From Bits to Brains. <<https://shafeentejani.github.io/2016-12-20/convolutional-neural-nets/>>. Luettu 6.5.2019.
- 54 Zhang, Alina. 2018. From Scalar to Tensor: Fundamental Mathematics for Machine Learning with Intuitive Examples Part 1/3. Verkkoaineisto. Medium. <<https://medium.com/datadriveninvestor/from-scalar-to-tensor-fundamental-mathematics-for-machine-learning-with-intuitive-examples-part-163727dfea8d>>. Luettu 24.5.2019.
- 55 Deep Learning Frameworks Comparison – Tensorflow, PyTorch, Keras, MXNet, The Microsoft Cognitive Toolkit, Caffe, Deeplearning4j, Chainer. Verkkoaineisto. Netguru S.A. <<https://www.netguru.com/blog/deep-learning-frameworks-comparison>>. Luettu 22.4.2019.
- 56 Comparison of deep-learning software. Verkkoaineisto. Wikipedia. <[https://en.wikipedia.org/wiki/Comparison\\_of\\_deep-learning\\_software](https://en.wikipedia.org/wiki/Comparison_of_deep-learning_software)>. Luettu 22.4.2019.



- 57 Python. Verkkoaineisto. Python Software Foundation. <<https://www.python.org/>>. Luettu 22.4.2019.
- 58 What makes Python the best programming language for machine learning and the best programming language for AI?. Verkkoaineisto. Steel KiWi. <<https://steelkiwi.com/blog/python-for-ai-and-machine-learning/>>. Luettu 8.5.2019.
- 59 Ketkar, Nikhil. 2017. Deep Learning with Python. Apress.
- 60 Deep Learning. Verkkoaineisto. NVIDIA Corporation. <<https://developer.nvidia.com/deep-learning>>. Luettu 21.4.2019.
- 61 CPU, GPU, FPGA or TPU: Which one to choose for my Machine Learning training?. Verkkoaineisto. InAccel. <<https://www.inaccel.com/cpu-gpu-fpga-or-tpu-which-one-to-choose-for-my-machine-learning-training/>>. Luettu 21.4.2019.
- 62 Conda. Verkkoaineisto. Anaconda. <<https://docs.conda.io/en/latest/>>. Luettu 21.4.2019.
- 63 Pypi. Verkkoaineisto. Python Software Foundation. <<https://pypi.org/>>. Luettu 11.5.2019.
- 64 TensorFlow. Verkkoaineisto. GitHub. <<https://github.com/tensorflow/tensorflow>>. Luettu 21.4.2019.
- 65 Keras. Verkkoaineisto. Keras Documentation. <<https://keras.io/>>. Luettu 21.4.2019.
- 66 DLTK. Verkkoaineisto. Github. <<https://github.com/DLTK/DLTK>>. Luettu 4.3.2019.
- 67 Pydicom. Verkkoaineisto. Darcy Mason and pydicom contributors. <[https://pydicom.github.io/pydicom/stable/getting\\_started.html](https://pydicom.github.io/pydicom/stable/getting_started.html)>. Luettu 14.5.2019.
- 68 SimpleITK. Verkkoaineisto. Kitware. <<http://www.simpleitk.org/>>. Luettu 21.4.2019.
- 69 Future. Verkkoaineisto. Python Software Foundation. <<https://pypi.org/project/future/>>. Luettu 6.4.2019.
- 70 Pandas. Verkkoaineisto. NumFOCUS. <<https://pandas.pydata.org/>>. Luettu 21.4.2019.

- 71 Matplotlib. Verkkoaineisto. The Matplotlib development team. <<https://matplotlib.org/>>. Luettu 14.5.2019.
- 72 Xlrd. Verkkoaineisto. GitHub. <<https://github.com/python-excel/xlrd>>. Luettu 21.4.2019.
- 73 Numpy. Verkkoaineisto. NumPy developers. <<https://www.numpy.org/>>. Luettu 21.4.2019.
- 74 SciPy. Verkkoaineisto. GitHub. <<https://github.com/scipy/scipy>>. Luettu 21.4.2019.
- 75 Containerize PowerAI with nvidia-docker. Julkaistu 2017, päivitetty 2018. Verkkoaineisto. IBM. <<https://developer.ibm.com/linuxonpower/tutorials/powerai-docker-images/>>. Luettu 7.5.2019.
- 76 What is a container? Verkkoaineisto. Docker. <<https://www.docker.com/resources/what-container>>. Luettu 7.5.2019.
- 77 IBM Cloud is the cloud for smarter business. Verkkoaineisto. IBM. <<https://www.ibm.com/cloud>>. Luettu 16.5.2019.
- 78 IBM Watson Studio. Verkkoaineisto. IBM. <<https://www.ibm.com/cloud/watson-studio>>. Luettu 11.05.2019.
- 79 IBM Cloud Object Storage. Verkkoaineisto. IBM. <<https://www.ibm.com/cloud/object-storage>>. Luettu 13.5.2019.
- 80 Data science and machine learning. Verkkoaineisto. IBM. <<https://www.ibm.com/analytics/machine-learning>>. Luettu 13.5.2019
- 81 Vaidyanathan, Rohan. 2018. Watson OpenScale. Verkkolähde. IBM. <<https://www.ibm.com/blogs/bluemix/2018/12/automate-and-operationalize-your-ai-with-ai-openscale/>>. Luettu 14.5.2019.
- 82 Manthorpe, Rowland. 2017. Harri Valpola dreams of an internet of beautiful AI minds. Verkkoaineisto. Wired. <<https://www.wired.co.uk/article/harri-valpola-curious-ai-artificial-intelligence-third-wave>>. Luettu 2.4.2019.
- 83 Rantanen, Kalevi. 2019. Se on kuin tiikeri häkissä - Tekoäly on pakotettava tekemään kolme asiaa: oppimaan, omaksumaan ja säilyttämään inhimilliset tavoitteet. Verkkoaineisto. Tekniikka&talous. <<https://www.tekniikkatalous.fi/tekoaly/se-on-kuin-tiikeri-hakissa-tekoaly-on-pakotettava-tekemaan-kolme-asiaa-oppimaan-omaksumaan-ja-sailyttamaan-inhimilliset-tavoitteet-6754907>>. Luettu 2.4.2019.

- 84 Ennakkoluulot ja stereotypiat. 2018. Verkkoaineisto. Peda.net. <<https://peda.net/kol/foorumi/m/mkk/t2kk/ejs/ejs>>. Luettu 22.4.2019.
- 85 Keinoäly syrji sukupuolen, kielen ja asuinalueen perusteella – Yhdenvertaisuusvaltuutettu älähti lainanhakijan kohtelusta. 2018. Verkkoaineisto. Yle. <<https://yle.fi/uutiset/3-10176934>>. Luettu 22.4.2019.
- 86 Dickson, Ben. 2018. What is algorithmic bias? Verkkoaineisto. TechTalks. <<https://bdtechtalks.com/2018/03/26/racist-sexist-ai-deep-learning-algorithms/>> Luettu 28.4.2019.
- 87 Kysely- ja haastattelumenetelmät. Verkkoaineisto. Tampereen yliopisto. <<https://www.tut.fi/verne/tutkimusmenetelmat/kysely-ja-haastattelumenetelmat/>>. Luettu 14.5.2019.
- 88 Hyvärinen, Matti; Nikander, Pirjo; Ruusuvoori, Johanna. 2017. Tutkimushaastattelun käsikirja. Vastapaino.
- 89 Feasibility study. Verkkoaineisto. Sanastokeskus TSK. <<http://www.tsk.fi/tepa/fi/haku/feasibility%20study>>. Luettu 6.4.2019.
- 90 Burke, Rory. 2003. Fourth Edition. Project Management: Planning and Control Techniques. United Kingdom: Burke Publishing.
- 91 What is a Feasibility Study?. 2009. Verkkoaineisto. Iowa State University. <<https://www.extension.iastate.edu/agdm/wholefarm/pdf/c5-65.pdf>>. Luettu 25.10.2018.
- 92 Feasibility study. Verkkoaineisto. Simplilearn Solutions. <<https://www.simplilearn.com/feasibility-study-article>>. Luettu 6.4.2019.
- 93 Technical Feasibility. Verkkoaineisto. Simplicable. <<https://simplicable.com/new/technical-feasibility>>. Luettu 6.4.2019.
- 94 Ylä-Outinen, Julia. 2017. Tekoäly on hyvä perämies terveystarkentamisessa - kontrolli pysyy lääkäriä. Verkkoaineisto. Kauppalehti. <<https://www.kauppalehti.fi/uutiset/tekoaly-on-hyva-peramies-terveystarkentamisessa-kontrolli-pysyy-laakarilla/5ccaab19-8d41-3729-b5de-313ef751c86a>>. Luettu 30.4.2019.
- 95 Kuvantamistutkimusten ja potilastietojen säilyttäminen. 2018. Verkkoaineisto. ePressi. <<https://www.epressi.com/tiedotteet/tiede-ja-tutkimus/kuvantamistutkimusten-ja-potilastietojen-sailyttaminen.html>>. Luettu 2.5.2019.

- 96 LIDC-IDRI. Verkkoaineisto. Cancer Imaging Archive. <<https://wiki.cancerimagingarchive.net/display/Public/LIDC-IDRI#632c66e790ee4276b61fa62139c10c79>>. Luettu 28.3.2019.
- 97 Ruponen, Jukka. 2019. IBM Watson OpenScale Lab: Validating a German Credit Risk Model for Fairness and Accuracy. Verkkoaineisto. Github. <<https://github.com/jruponen/aio>>. Luettu 7.4.2019.
- 98 Watson OpenScale for ICP for Data. Verkkoaineisto. IBM Cloud. <<https://cloud.ibm.com/docs/services/ai-openscale-icp?topic=ai-openscale-icp-abt-about>>. Luettu 9.5.2019.
- 99 Ruponen, Jukka. 2018. Tutorials to Artificial Neural Networks with Watson Data Platform. Verkkoaineisto. Github. <<https://github.com/jruponen/ann-with-wdp>>. Luettu 2.5.2019.

## Radiologisten modaliteettien kuvantamisominaisuudet

Taulukkoon on koottu radiologisten kuvantamistekniikoiden ominaisuuksia ja lääketieteellisiä käyttökohteita. Taulukko pääasiassa koostuu Radiologian kirjasta s. 12.

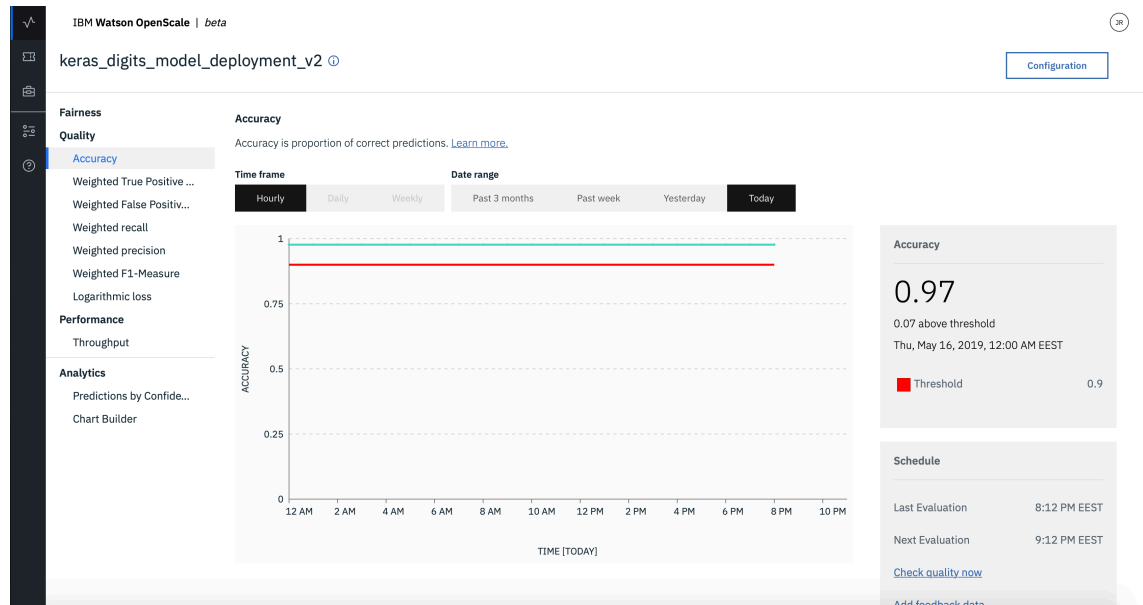
	Ultraääni	Röntgen 2D (digitaalinen)	Magneettikuva	Tietokone-tomografia	Isotooppi-kuvantaminen
Harmaasävytaso (Kontrasti)	8–10 bittiä, eli 256-1024 tasoa	10–14 bittiä, eli 1024 – 16384 tasoa	12–16 bittiä, eli 4 096-65 536 tasoa	12 bittiä, eli 4096 tasoa	Riippuu tilanteesta ja valmistajasta
Paikkaerotuskyky (mm)	0,1–2	0,1	0,2–1	< 1	5-10 (SPECT) ~3 (PET)
Kuvasignaalin alkuperä	Äänen heijastuskerroin, taajuusmuutos (doppler)	Jarrutussäteily + Karakteristinen röntgensäteily	Ydinmagneettinen resonanssi	Jarrutussäteily + Karakteristinen röntgensäteily	Radioaktiivinen hajonta (SPECT) Annihilaatio (PET)
Keskeisiä käyttökohteita	Sikiö, sydän, rakko, veren virtaus (doppler)	Anatomiset rakenteet, tiheät kudokset esim. luut	Anatomiset rakenteet, pehmytkudokset, virtaus, liike, diffuusio	Anatomiset rakenteet, tiheät kudokset esim. luut	Aivot, sydän, luusto (SPECT) Metabolia, syöpä, aivot, epilepsia, Alzheimerin tauti (PET)
Energia	Ultraääni (1-39MHz)	≤140 keV	Radiotaajuussäteily (RF) ≤130 MHz	≤140 keV	Gammasäde esim. 140 keV (SPECT) 2 x 511 keV (PET)
Signaali-intensiteettiin vaikuttavat seikat	Akustisen impedanssin erot, absorptio, sironta, virtausnopeus (doppler)	Absorptio, Sironta	Kuvausparametrit, kohteen protonitiheys	Absorptio, Sironta	Absorptio, Sironta

Lähteet:

- Soimakallio, Seppo; Kivisaari, Leena; Manninen, Hannu; Svedström Erkki; Teronen, Osmo. 2005. Radiologia. WSOY.
- Hoffmann, Klaus-Peter; Kramme, Rüdiger. 2011. Springer Handbook of Medical Technology. E-kirja. Springer Link.

## OpenScale accuracy testitulokset

Kuvakaappaus OpenScalen antamasta Accuracy tuloksesta ja tulosten esitys konfuusiomatriisilla valitulta ajan hetkeltä.



IBM Watson OpenScale | beta

keras\_digits\_model\_deployment\_v2 : Accuracy

Metric	Value
Accuracy	0.97
Weighted True Positive Rate (wTPR)	0.97
Weighted False Positive Rate (wFPR)	0
Weighted recall	0.97
Weighted precision	0.97
Weighted F1-Measure	0.97
Logarithmic loss	0.07

Actual \ Prediction	0	1	2	3	4	5	6	7	8	9
0	85	0	0	0	0	0	0	0	0	0
1	0	125	0	0	0	1	0	0	0	0
2	0	0	114	0	0	0	0	1	0	1
3	0	0	0	104	0	2	0	1	0	0
4	0	0	1	0	106	0	0	0	0	3
5	0	0	0	0	1	85	0	0	1	0
6	3	0	0	0	0	1	83	0	0	0
7	0	0	0	0	0	0	0	99	0	0
8	0	0	3	0	1	0	1	1	82	1
9	0	0	0	0	0	0	0	0	0	94

Records Evaluated

Total 1000