



**UNIVERSITY
OF OULU**

TIETO- JA SÄHKÖTEKNIIKAN TIEDEKUNTA

**Matti Luukkonen
Rasmus Kurtti
Juho Hummasti**

**TOISTUVIEN KOHTAAMISTEN VAIKUTUS
MESSUESITTELYROBOTIN TOIMINTAAN -
KASVOJEN TUNNISTAMINEN JA
MUISTAMINEN**

Kandidaatintyö
Tietotekniikan tutkinto-ohjelma
Toukokuu 2024

Luukkonen M., Kurtti R., Hummasti J. (2024) Toistuvien kohtaamisten vaikutus messuesittelyrobotin toimintaan - kasvojen tunnistaminen ja muistaminen. Oulun yliopisto, Tietotekniikan tutkinto-ohjelma, 50 s.

TIIVISTELMÄ

Konenäöllä on monia sovelluksia nyky-yhteiskunnassa. Kasvojentunnistus on yksi sen sovelluksista, ja sitä käytetään laajalti eri tarkoituksiin ihmisen ja robotin välisessä vuorovaikutuksessa. Robotit yleistyvät ihmisten keskuudessa ja erilaiset ympäristöt tuovat haasteita kasvojentunnistukselle. Esimerkiksi messuympäristössä suuri ihmismäärä hankaloittaa robotin kasvojentunnistuksen toimintaa.

Jotta humanoidirobotit voitaisiin hyväksyä osaksi ihmisten jokapäiväistä elämää, tulee ihmisten kyetä luonnolliseen vuorovaikutukseen niiden kanssa. Kun robotti pystyy tunnistamaan ihmisten kasvot ja mahdollisesti jopa niiden ilmeet, se voi reagoida paremmin ihmisten läsnäoloon ja tarpeisiin. Tämä tekee robotista helpommin lähestyttävän ja vuorovaikutustilanteesta turvallisemman. Kun robotille toteutetaan kasvojentunnistuksen lisäksi muisti, se pystyy kohtelemaan ihmisiä personoidusti.

Kasvojentunnistusmalleja on paljon, ja niiden välillä on merkittäviä eroja niin nopeudessa kuin tarkkuudessaakin. On tärkeää tutkia näitä eroja, jotta voidaan valita kuhunkin tilanteeseen parhaiten soveltuva malli.

Opiskelijoiden kehittämän messuesittelyrobotin kasvojentunnistusjärjestelmään lisättiin kyky kasvojen muistamiselle. Järjestelmää muokattiin niin, että kasvojen löytämis- ja tunnistamismalleja on helppo vaihtaa. Järjestelmälle valittiin käyttötarkoitukseen sopivat mallit vertailemalla useiden tunnettujen mallien suorituskykyä ja tarkkuutta.

Kasvojen muistaminen toteutettiin klusterointialgoritmin avulla. Kasvojen muistamista testattiin ajamalla videomateriaalia kasvojentunnistusjärjestelmän läpi. Kehitetty kasvojentunnistusjärjestelmä pystyy tunnistamaan, onko messuesittelyrobotin näköpiirissä oleva henkilö robotille tuttu vai tuntematon.

Avainsanat: konenäkö, kasvojentunnistus, kasvojen klusterointi, ihmisen ja robotin vuorovaikutus, luonnollinen vuorovaikutus, humanoidirobotit

Luukkonen M., Kurtti R., Hummasti J. (2024) The Impact of Repeated Encounters on Trade Show Presentation Robot Operation - Facial Recognition and Memory. University of Oulu, Degree Programme in Computer Science and Engineering, 50 p.

ABSTRACT

Machine vision has many applications in society. Facial recognition is one of the applications widely used for various purposes in human-robot interaction. Robots are becoming more common among humans, and different environments pose challenges for facial recognition. For instance, in a crowded exhibition environment, the large number of people complicates facial recognition for robots.

For humanoid robots to be accepted as part of everyday life, humans must be able to interact with them naturally. When a robot can identify human faces and possibly even their expressions, it can respond better to human presence and needs. This makes the robot more approachable and the interaction situation safer. With memory integrated alongside facial recognition, a robot can treat people in a personalized manner.

There are many facial recognition models with significant differences in speed and accuracy. It is important to study these variances to choose the most suitable model for each situation.

Face recognition system developed by students for a trade show presentation robot was improved by adding the ability to remember faces. The system was modified to allow easy switching between different face detection and recognition models. Suitable models were selected for the system by comparing the performance and accuracy of several well-known models.

The face memory feature was implemented using a clustering algorithm. Clustering was tested by running video material through the face recognition system. The developed face recognition system can determine whether a person within the robot's field of view is familiar or unknown.

Keywords: machine vision, facial recognition, human-robot interaction, natural interaction, humanoid robots

SISÄLLYSLUETTELO

TIIVISTELMÄ	
ABSTRACT	
SISÄLLYSLUETTELO	
ALKULAUSE	
LYHENTEIDEN JA MERKKIEN SELITYKSET	
1. JOHDANTO	8
2. KONENÄKÖ JA KASVOJENTUNNISTUS	9
2.1. Konenäön ja kasvojentunnistuksen historia	9
2.2. Konenäön ja kasvojentunnistuksen sovellukset nykyaikana	9
2.3. Johdatus konenäkötekniikoihin	10
2.4. Kasvojentunnistus ja sen vaiheet	11
2.4.1. Kasvojen löytäminen	12
2.4.2. Kasvojen suuntaus	13
2.4.3. Kasvojen tunnistaminen	13
2.4.4. Verifiointi ja identifiointi	14
2.5. Kasvojentunnistumalleja	14
2.6. Klusterointi	14
2.7. Robotin muistijärjestelmä	16
2.7.1. Erilaiset muistirakenteet ihmisellä	16
2.7.2. Yksinkertainen robotin muisti ja representaatiot	17
2.7.3. Keinotekoiset kognitiiviset arkkitehtuurit ja muistijärjestelmät ...	17
2.7.4. Tekniset vaatimukset	18
2.7.5. Humanoidirobotin pitkäaikainen muisti	18
3. IHMISEN JA KONEEN VÄLINEN VUOROVAIKUTUS	20
3.1. Lyhyt katsaus robotiikkaan	20
3.2. Luonnollisempi vuorovaikutus	21
3.2.1. Verbaalinen ja visuaalinen vuorovaikutus	21
3.2.2. Fyysinen vuorovaikutus	22
3.2.3. Tunteet ja niiden tulkitseminen	22
3.3. Robotin käyttäytyminen ja antropomorfismi	23
3.3.1. Vuorovaikutus tuntemattoman henkilön kanssa	24
3.3.2. Vuorovaikutus ennestään tutun henkilön kanssa	25
3.3.3. Antropomorfismi	26
3.4. Tekoälyn vaikutus HRI:n kehitykseen	27
3.4.1. Tekoälyagentti ja robotin autonomisuus	27
3.4.2. Tekoälyn mahdollisuudet HRI:ssä	28
3.4.3. Robotin opettaminen	28
4. TOTEUTUS	30
4.1. Sovellusympäristö ja sen haasteet	30
4.2. Olemassa oleva toteutus	31
4.2.1. Face_tracker moduulin käyttöönotto	31
4.2.2. Refaktorointi	32
4.3. Kuinka kasvojentunnistus toteutettiin?	32

4.3.1.	Kasvojen löytämis- ja kasvojentunnistusmallien valinta	32
4.3.2.	Suorituskyvyn parannuksia	35
4.4.	Kuinka kasvojen tallentaminen ja muisti toteutettiin?	36
4.4.1.	Kasvojentunnistuksen tulosten suodattaminen	36
4.4.2.	Klusterointialgoritmin valinta.....	37
4.4.3.	Links-klusterointialgoritmin kasvojentunnistusjärjestelmään	sovitus 38
4.4.4.	ROS-viestit	38
4.5.	Kuinka toteutusta testattiin?.....	39
4.6.	Pohdinta ja jatkokehitys	40
5.	YHTEENVETO.....	43
6.	PROJEKTIN KUVAUS	44
7.	VIITTEET	45

ALKULAUSE

Haluamme kiittää Teemu Tokolaa kandidaatintyön ohjaamisesta ja Aapo Pihlajaniemeä avusta toteutuksen kanssa.

Oulussa 8. toukokuuta 2024

Matti Luukkonen
Rasmus Kurtti
Juho Hummasti

LYHENTEIDEN JA MERKKIEN SELITYKSET

ROS	The Robot Operating System - robottien ohjausjärjestelmä
CNN	Convolutional neural network - konvoluutioneuroverkko
STM	Short term memory - lyhytkestoinen muisti
LTM	Long term memory - pitkäkestoinen muisti
HRI	Human-robot interaction - ihmisen ja robotin vuorovaikutus
AI	Artificial intelligence - tekoäly
NPL	Natural language processing - luonnollisen kielen käsittely
DEA	Differential elastic actuator - differentiaalinen elastinen toimilaite
EEG	Electroencephalogram - aivosähkökäyrä
EMG	Electromyography - elektromyografia
LFW	Labelled Faces in the Wild - yleisesti käytössä oleva kuvatietokanta kasvojentunnistusmallien arviointiin
fps	Frames per second - kuvataajuus

1. JOHDANTO

Ihminen pystyy tunnistamaan kasvoja intuitiivisesti muistin ja visuaalisten piirteiden avulla. Tutkimuksista huolimatta ei ole selvää, tunnistaako ihminen kasvoja holistisesti vai analysoimalla yksittäisiä tai paikallisia piirteitä. Tässä tutkimuksessa ei käsitellä tarkemmin ihmisen kykyä tunnistaa kasvoja, mutta on hyvä tiedostaa yleisellä tasolla, kuinka kasvojentunnistus eroaa ihmisen ja koneen välillä.

Konenäkö (eng. computer vision) tarkoittaa koneiden kykyä havaita ja tulkita visuaalista tietoa kuten kuvia tai videoita. Kasvojentunnistus (eng. face recognition) on konenäön osa-alue, jossa koneet pyrkivät tunnistamaan yksilöitä tai luokittelemaan kasvoja perustuen visuaaliseen dataan. Vaikka konenäköalgoritmit ovat nykyisin todella nopeita ja tarkkoja, liittyy kasvojentunnistukseen yhä paljon haasteita. Ongelmia aiheuttavat esimerkiksi kasvojen asennon ja ilmeiden vaihtelu. Kasvojen ulkonäkö voi myös muuttua merkittävästi eri valaistusolosuhteissa.

Erityisesti messuympäristössä kohdataan useita haasteita. Ensinnäkin tarkan kuvan ottaminen videosta voi olla hankalaa, sillä vuorovaikutuksessa olevat ihmiset saattavat liikkua jatkuvasti. Lisäksi taustalla voi olla muuta liikettä, mikä vaikeuttaa kasvojen erottamista. Messuilla on myös tyypillisesti paljon ihmisiä, mutta vain yksi heistä vuorovaikuttaa kerrallaan järjestelmän kanssa. Tämä asettaa omat rajoitteensa tunnistuksen onnistumiselle. Ihmisten tunnistaminen ja luokittelu ei tarvitse kuitenkaan täysin perustua kasvojentunnistukseen, vaan robotiikassa voidaan hyödyntää esimerkiksi puheentunnistusta ja erilaisia tunnistamiseen liittyviä kysymyksiä kasvojentunnistusalgoritmien tukena.

Humanoidiroboteilla on useita tulevaisuuden sovelluskohteita. Ne voisivat toimia ihmisten rinnalla, avustaen erilaisissa tehtävissä, kuten asiakaspalvelussa. Sosiaaliset robotit voisivat olla tukena erityisesti henkilöille, jotka tarvitsevat apua arjen askareissa. Kaikissa edellä mainituissa tilanteissa tarvitaan kasvojentunnistusta yksilöiden tunnistamiseen. Kasvojentunnistusteknologian avulla robotit kykenevät toimimaan personoidummin ja tehokkaammin. Tämä voi parantaa asiakaskokemusta, lisätä turvallisuutta ja tehdä vuorovaikutuksesta robotin kanssa luontevampaa ja miellyttävämpää.

Tässä kandidaatintyössä pohditaan ja tutkitaan keinoja, kuinka voitaisiin laajentaa valmiiksi toteutettua kasvojentunnistusalgoritmia tunnistamaan ja muistamaan henkilöitä, jotka ovat jo aiemmin olleet vuorovaikutuksessa robotin kanssa. Tämä on tärkein tutkimusongelma. Tavoitteena on kehittää ratkaisuja, jotka toimisivat erityisesti messuympäristössä, jossa ihmisen ja robotin välisiä vuorovaikutustilanteita on jatkuvasti ja paljon. Tutkimuksessa syvennytään myös ihmisen ja robotin väliseen vuorovaikutukseen ja robotin käyttäytymiseen uuden ihmisen kanssa sekä sen muutokseen, kun vuorovaikutuksessa on ennestään tuttu henkilö. Kehitettävä robotti (InMoov) käyttää Robot Operating System 2 käyttöjärjestelmää (ROS2). Nykyinen toteutus on kirjoitettu Python ohjelmointikielellä ja käyttää OpenCV kirjastoa.

2. KONENÄKÖ JA KASVOJENTUNNISTUS

Tekoäly (eng. artificial intelligence, AI) on tiede, jonka tavoitteena on älykkäästi ja rationaalisesti toimivien koneiden luominen. Konenäkö on yksi tekoälyn muoto, minkä avulla tietokoneet ja laitteet voivat kerätä tietoa ympäristöstään visuaalisten lähteiden perusteella [1, 2]. Konenäkö antaa robotille kyvyn nähdä ympärilleen, mutta näkeminen perustuu konenäköalgoritmeihin ja eroaa siten ihmisen näköaistista. Kasvojentunnistus on tekniikka, jossa hyödynnetään konenäköalgoritmeja tunnistamaan ja luokittelemaan henkilöitä heidän kasvojensa perusteella.

2.1. Konenäön ja kasvojentunnistuksen historia

Useiden vuosikymmenien ajan on yritetty kehittää menetelmiä, jotka antaisivat koneelle mahdollisuuden nähdä ja ymmärtää visuaalista dataa. 1960-luvun paikkeilla kehitettiin ensimmäiset kuvaskannerit, jonka myötä tietokoneilla pystyttiin käsittelemään digitaalisia kuvia. Vuonna 1963 tietokoneet pystyivät muuntamaan kaksiulotteisia (2D) kuvia kolmiulotteisiksi malleiksi. Vuonna 1982 neurotieteilijä David Marr havaitsi, että näkö toimii hierarkkisesti, ja esitteli konealgoritmin reunojen, kulmien, kaarteiden ja vastaavien primitiivisten muotojen havaitsemiseksi. [1]

Kasvojentunnistuksella on pitkä historia ja se herättää kiinnostusta niin neurologeissa, psykologeissa kuin tietojenkäsittelytieteilijöissäkin [3]. Kasvojentunnistus alkoi manuaalisesta tunnistamisesta, jossa ihmisen tehtävänä oli vertailla kuvia ja tunnistaa niistä esimerkiksi rikollisen kasvonpiirteet. Tämä perinteinen metodi oli työläs ja altis virheille.

Ensimmäinen koneellinen kasvojentunnistusmetodi oli amerikkalaisten tutkijoiden Bledsoe ym. kehittämä puoliautomaattinen malli, jossa käyttäjältä kysyttiin kaksikymmentä eri mittaa. Kysytyjä mittoja olivat esimerkiksi suun tai silmien koko. Vuonna 1991 Turk ja Pentland esittelivät ensimmäisen kasvojentunnistusalgoritmin, nimeltään Eigenfaces [4, 3]. Vuonna 1998 FERET-ohjelman pohjalta luotiin yleiskäyttöinen suurehko tietokanta, joka sisälsi 2400 kuvaa 850 eri ihmisestä kasvojentunnistusmallien kehityksen tueksi [5]. Vuonna 2005 The Face Recognition Grand Challenge (FRGC) -kilpailu käynnistettiin kannustamaan ja edistämään kasvojentunnistustekniikan kehitystä [6]. 2010-luvulla kasvojentunnistusmalleissa tapahtui huomattavaa kehitystä, kun syväoppiminen ja siihen perustuvat kasvojentunnistusmallit kehittyivät [7]. Vuonna 2014 Facebookin Deepface-algoritmi pystyi tunnistamaan kasvoja lähes ihmissilmän tarkkuudella, saavuttaen 97% tarkkuuden [8].

2.2. Konenäön ja kasvojentunnistuksen sovellukset nykyaikana

Konenäkö on noussut tärkeäksi toimijaksi monella eri alalla. Teollisessa automaatiassa ja laadunvalvonnassa sitä käytetään esimerkiksi virheellisten komponenttien tai pilaantuneiden tuotteiden tunnistamiseen [9]. Konenäköä sovelletaan paljon myös lääketieteellisessä kuvantamisessa ja diagnoosissa, autonomisissa ajoneuvoissa ja robotiikassa sekä rakennusalalla. [10].

Kasvojentunnistuksen käyttö on laajentunut huomattavasti viime vuosina. Se ei rajoitu enää vain hallituksiin tai korkean turvallisuuden kohteisiin, vaan se on levinnyt laajasti arkipäiväiseen käyttöön. Esimerkiksi älypuhelimet hyödyntävät kasvojentunnistusta näytön lukituksessa. Vuonna 2017 iPhone X:n yhteydessä julkaistiin FaceID-kasvojentunnistusjärjestelmä, jonka virhemarginaali oli yksi miljoonasta [3].

Tietokoneiden tietoturvasa sovelletaan myös kasvojentunnistusteknologiaa. Kun käyttäjä poistuu väliaikaisesti tietokoneen äärestä, järjestelmä jatkaa jatkuvaa käyttäjän autentikointia varmistaakseen, että henkilö, joka on tietokoneen näytön edessä, on sama henkilö, joka alun perin kirjautui sisään tietokoneelle. [11, 12].

Kasvojentunnistusta käytetään myös laajalti turvallisuuden ylläpitämiseen eri ympäristöissä, kuten lentokentillä, rajanylityspaikoilla, toimistoissa, pankeissa, kodeissa ja kaupungilla. Kaupungeissa kaduilla olevilla CCTV-kameroilla voidaan tunnistaa rikollisia, joilla on etsintäkuulutus perässä. Esimerkiksi Virginian kaupunki Virginia Beach oli toinen kaupunki Yhdysvalloissa, jossa alettiin skannaamaan jalankulkijoiden kasvoja ja vertaamaan niitä kuviin rikollisista ja kadonneista ihmisistä, jotta heidät löydettäisiin [12].

2.3. Johdatus konenäkötekniikoihin

Konenäön toiminnan ja merkitysten ymmärtämiseksi on tutkittava digitaalisen kuvan rakennetta. Digitaalinen kuva on visuaalinen kokonaisuus, joka rakentuu pienistä kuvaelementeistä. Näitä kuvaelementtejä kutsutaan pikseleiksi. Jokainen pikseli on pieni neliö, joka sisältää tietoa kuvan värin ja kirkkauden tasosta tietyssä kohdassa. Digitaalinen kuva voidaan esittää matemaattisesti matriisina, johon pikselien arvot sijoitetaan. Kun oletetaan kuvan olevan suorakulmaisen muotoinen, matriisin sarakkeet kertovat kuvan leveyden ja rivit kuvan korkeuden. Mitä suurempi pikselimäärä kuvassa on, sitä tarkempi lopullinen kuva on. [13]

Digitaalisia kuvia voidaan luoda tietokoneella keinotekoisesti algoritmien tai tekoälyn avulla tai perinteisesti muuntamalla kameralla otettu valokuva digitaaliseen muotoon. Valokuvan muuntaminen digitaaliseksi perustuu signaalinkäsittelyyn. Nykyisissä kameroissa on erittäin herkkä kuvakenno, joka koostuu pienistä fotosensoreista. Jokainen fotosensori tallentaa siihen osuvan valon ja muuntaa sen signaaliksi. Näitä sähköisiä signaaleja käytetään luomaan digitaalinen kuvatiedosto. [13]

Konenäköjärjestelmien käyttäminen ja kehittämien alkavat aina visuaalisen tiedon hankinnasta. Data on yleensä 2D-kuvan tai videon muodossa. Myös 3D-kuva ja 3D-pistepilvet ovat yleisiä tyyppisiä [14]. Visuaalinen data sisältää monenlaisia piirteitä. Esimerkiksi reunat, kulmat, muodot, värit ja erilaiset tekstuurit ovat piirteitä. Perinteiset konenäön tekniikat kohteiden ja objektien havaitsemiseen sisältävät algoritmeja piirteiden tunnistamiseen. Huono puoli perinteisessä lähestymistavassa on se, että käyttäjän täytyy valita, mitkä piirteet ovat tärkeitä datassa. Erilaisten luokkien määrän kasvaessa piirteiden erottelu muuttuu yhä hankalammaksi. Tekoäly ja erityisesti syväoppiminen (eng. deep learning) tuovat tähän ongelmaan ratkaisuja. [15]

Syväoppinen on koneoppimisen alalaji, joka pyrkii mallintamaan ja simuloimaan ihmisen aivojen toimintaa neuroverkkojen avulla. Kuten aivoissa, tämä neuroverkko koostuu monista laskentasuoluista eli neuroneista, jotka suorittavat yksinkertaisen toiminnon. Yksinkertaistettuna, neuronit toimivat yhdessä toistensa kanssa, eli ne vastaanottavat ja lähettävät dataa keskenään. [15, 16] Oppiminen tapahtuu useissa peräkkäisissä tasoissa, joista jokainen edustaa yhä abstraktimpia piirteitä tai käsitteitä. Alimmat tasot tunnistavat yksinkertaisia piirteitä, kuten reunaviivoja, kun taas ylemmät tasot yhdistävät nämä piirteet monimutkaisemmiksi ominaisuuksiksi. Kehittyneet kasvojentunnistusmenetelmät käyttävät hyvin abstrakteja piirteitä.

Ihminen pystyy havainnoimaan tilanteita hyvin lahjakkaasti ja osaa keskittyä näkemiinsä yksityiskohtiin nopeasti ja käsitellä niitä aivan erillisinä kokonaisuuksina. Syväoppivissa neuroverkoissa samankaltainen prosessi perustuu siihen, että eri tasot edustavat eri objekteja. [17]

Konvoluutioneuroverkot (eng. Convolutional Neural Networks, CNN) ovat erityinen neuroverkkotyyppi. Ne perustuvat konvoluutioihin, jotka ovat matemaattisia operaatioita, joilla suodatetaan tiettyjä piirteitä syötteestä. Yleensä konvoluutioneuroverkot rakentuvat kolmesta päällekkäisestä kerroksesta: konvoluutiokerros, koontikerros (eng. pooling layer) ja luokittelukerros (eng. classification layer) [9, 16]. Konvoluutioneuroverkot ovat tehokkaita käsittelemään esimerkiksi kuvia ja niitä käytetäänkin objektien paikantamisessa, kasvojentunnistuksessa ja robotiikassa. [16]

Viime aikoina laitteiden laskentateho, muistikapasiteetti, virrankulutus ja optiikka ovat parantuneet paljon, mikä osaltaan on helpottanut syväoppimisen kehitystä. Verrattuna perinteisiin konenäkötekniikoihin, syväoppiminen mahdollistaa suuremman tarkkuuden tehtävissä, kuten kuvien luokittelu, semanttinen segmentointi ja kohteen havaitseminen. Syväoppimiseen liittyy silti vielä haasteita. Vaikka laskentateho on kasvanut, uusimmat syväoppimismenetelmät vaativat niin paljon laskutoimituksia, että tästä muodostuu merkittävä este monille organisaatioille ja tutkijoille. Lisäksi syväoppiminen tarvitsee suuria tietoaaineistoja koulutusta varten. [15]

2.4. Kasvojentunnistus ja sen vaiheet

Ihmisen kasvot ovat todennäköisesti yleisin ja tutuin biometrinen kokonaisuus. Kasvot ovat äärimmäisen monimutkainen rakenne ja kasvopiirteet voivat muuttua ajan myötä paljon. Tieteilijät pyrkivät ymmärtämään ihmiskasvojen rakennetta kehittäessään kasvojentunnistusjärjestelmiä. [2] Vaikka uudet menetelmät perustuvat usein vanhoihin menetelmiin, on niissä tiettyjä erityisiä piirteitä. Tyypillinen nykyaikainen kasvojentunnistusjärjestelmä sisältää seuraavat peruselementit [18, 19]:

1. Kasvojen löytäminen (eng. detection)
2. Kasvojen suuntaus (eng. alignment)
3. Kasvojen tunnistaminen (eng. recognition) ja representointi (eng. representation)
4. Verifiointi (eng. verification) ja identifiointi (eng. identification)

Ensimmäisessä vaiheessa kasvot paikannetaan kuvasta. Sen jälkeen kasvojen suuntaus kalibroi kasvot yhtenäiseen asentoon tai kulmaan. Sitten kasvot tunnistetaan

ja niistä luodaan representaatiot. Lopuksi representaatioita vertaillaan tietokannasta löytyviin henkilöihin. Nykyään kaikki nämä vaiheet toteutetaan syväoppivan konvoluutioneuroverkon avulla. [18]

2.4.1. Kasvojen löytäminen

Tässä kandityössä käytetään kasvojentunnistusjärjestelmän ensimmäisestä vaiheesta nimitystä "kasvojen löytäminen", koska sen avulla erottellaan tämä neljännessä vaiheesta "kasvojen tunnistaminen". Usein tästä vaiheesta käytetään suomeksi nimitystä "kasvojen tunnistus". Aluksi pyritään syötekuvan perusteella löytämään kaikki kuvassa olevat kasvot. Tämä voidaan suorittaa monella eri menetelmällä, joita on esitetty taulukossa 1.[18].

Taulukko 1. Kasvojentunnistusmenetelmien syväluokittelu

Nimi	Kuvaus
Monivaiheinen	Menetelmä havaitsee kasvot useiden vaiheiden tai askelten kautta. Aluksi luodaan monta ehdokasta, joita myöhemmin kehitetään ja parannetaan.
Yksivaiheinen	Menetelmä havaitsee kasvot suoraan ilman monivaiheista käsittelyä.
Ankkuripohjainen	Piirrekarttoihin sijoitetaan useita ankkureita. Näiden ankkureiden luokittelun ja regression perusteella löydetään kasvot. Tämä on hyvin yleinen menetelmä nykypäivänä.
Ankkuriton	Menetelmä löytää kasvot suoraan ilman valmiiksi määritettyjä ankkureita.
Multi-task-oppiminen	Menetelmät, jotka oppivat yhdessä kasvojen havaitsemisen ja luokittelun.
CPU-reaaliaikainen	Havainnointilaitteet voivat toimia reaaliajassa yhdellä CPU-ytimellä VGA-resoluution kuvissa. Käytetään esimerkiksi videoissa.
Ongelmapohjainen	Menetelmä pyrkii ratkaisemaan kasvojen löytämiseen liittyviä ongelmia. Näitä voivat olla muun muassa osittain peitetyt kasvot ja epätarkat kuvat.

2.4.2. Kasvojen suuntaus

Toisessa vaiheessa tehdään kasvojen suuntaus. Tämän toimenpiteen avulla pyritään kalibroimaan havaitut kasvot kanoniseen kulmaan. Suuntaaminen on tärkeää, koska se helpottaa seuraavien vaiheiden algoritmeja piirteiden laskennassa ja erottelemisessa. Kasvojen suuntaaminen perustuu usein kasvojen säännöllisiin rakenteisiin, kuten silmiin, nenään ja suuhun, mutta myös muita menetelmiä on kehitelty. Joissakin tutkimuksissa puhutaan myös kasvojen normalisoinnista. Sitä ei löydy kaikista malleista, mutta se tarkoittaa kasvojen leikkaamista kuvasta erilleen (ainakin toteutuksessa käytetyn deepface-algoritmin tapauksessa). [18]

2.4.3. Kasvojen tunnistaminen

Kasvojen tunnistaminen ja representointi tarkoittaa sitä, että järjestelmä muuntaa havaitut kasvopiirteet numeeriseksi vektoriksi, jota voidaan käsitellä ja vertailla verifointivaiheessa. Monet varhaisimmat kasvojentunnistusalgoritmit perustuivat etäisyysmittauksiin. Koska myös jotkin nykyisistä menetelmistä käyttävät etäisyyksiä tunnistamisessa, esitellään tässä yleisimmät matemaattiset etäisyyskaavat.

Euklidista etäisyyttä käytetään luokitteluun laajasti, koska se on yksinkertainen ja nopeampi kuin muut luokittelijat. Euklidinen etäisyys tarkoittaa kahden pisteen välistä etäisyyttä. Moniulotteisessa tapauksessa etäisyys lasketaan summaamalla jokaisen vektorin tai pisteen vastaavien alkioiden erotusten neliöt ja ottamalla tämän summan neliöjuuri. Yleensä ottaen Euklidinen etäisyys lasketaan Pythagoraan lausetta käyttämällä: [2]

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (1)$$

Manhattan-etäisyys tai L1-etäisyys (eng. city block distance) on toinen tapa mitata etäisyyttä kahden pisteen tai vektorin välillä. Se mitataan ruudukossa, kulkemalla vaakaa ja pystyyn. [2]

$$d = \sum_{i=1}^n |x_i - y_i| \quad (2)$$

Kosinin samankaltaisuus (eng. cosine similarity) on mitta kahden nolasta poikkeavan vektorin välillä, jotka on määritelty sisäisessä tuloavaruudessa. Kosinin samankaltaisuus on vektorien välisen kulman kosini, eli se on vektorien pistetulo jaettuna niiden pituuksien tulolla: [19]

$$\cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \|\mathbf{B}\|} \quad (3)$$

Nykyisin kasvojentunnistuksessa käytetään syviä konvoluutioneuroverkkoja. Kasvojentunnistusmallien koulutus perustuu tappiofunktioon (eng. loss function). Tämän funktion tarkoituksena on mitata, kuinka hyvin malli suoriutuu annetusta tehtävästä ja ohjata mallin parametrien päivityksiä koulutuksen aikana. Tappiofunktio

ovat nykyään hyvin hienostuneita ja kehittyneitä. Kun uutta syväoppimismallia kasvojentunnistukseen kehitetään, pyritään luomaan uusi, tehokkaampi tappiofunktio. [18, 19]

Tarkimmat kasvojentunnistusmenetelmät hyödyntävät kolmiulotteista kasvokuvaa. Tämä kuitenkin vaatii erityislaatuista kamerateknologiaa, joten se ei sovellu kaikkialle. On toki mahdollista luoda 2D kuvan perusteella 3D-kuva, sillä 2D-kasvopiirteiden ja 3D-kasvomuodon välillä on luonnollisesti selkeitä yhteyksiä. Vertailtaessa tavallisiin 2D-menetelmiin, 3D-malleihin liittyvät menetelmät parantavat kasvontunnistuksen suorituskykyä ja luotettavuutta erityisesti tilanteissa, joissa kasvot ovat erilaisissa asennoissa ja kulmissa. [18, 19]

2.4.4. Verifiointi ja identifointi

Kasvojen verifiointi viittaa enemmän tilanteeseen, jossa yhden henkilön kasvoja tai representaatioita verrataan toisen henkilön kasvoihin ja representaatioihin. Kasvojen identifointi taas tarkoittaa, että kasvoja verrataan kaikkiin tietokannasta löytyviin kasvoihin. [18]

Kasvojen identifioinnissa järjestelmä antaa numeerisen identifointiarvon, joka kertoo, kuinka hyvin tunnistettu henkilö vastaa muita tietokannassa olevia henkilöitä. Jos tietty kynnsarvo ylittyy, kasvot yhdistetään tietokannasta löytyvään henkilöön. Lopuksi tässä vaiheessa suoritetaan myös kasvojen klusterointi (ks. luku 2.6).

2.5. Kasvojentunnistuskalleja

Erilaisia kasvojentunnistuskalleja on lukuisia. Mallien välillä on eroja tunnistustarkkuudessa ja nopeuksissa (ks. luku 4.3.1). Taulukossa 2 on lueteltu joitakin tunnettuja kalleja. Kasvojentunnistusjärjestelmät voivat käyttää myös useampaa kalleja kasvojen tunnistamiseen.

2.6. Klusterointi

Koneoppimisessa tai tilastollisessa analyysissä klusteri tarkoittaa joukkoa samankaltaisia tietoja tai havaintoja, jotka on ryhmitelty yhteen. Ryhmittelyanalyysi tai klusterointi kasvojentunnistuksessa tarkoittaa tuntemattomien kasvokuvien ja niiden representaatioiden järjestämistä tai ryhmittelyä niin, että voidaan tunnistaa, mitkä kuvat liittyvät samaan henkilöön.

Kasvojen klusterointialgoritmit voidaan jakaa kahteen eri menetelmään: offline- ja onlineklusterointi. Offlineklusteroinnissa koko data on saatavilla. Usein myös haluttu klusterien määrä täytyy määrittää etukäteen. Offline-menetelmät ovat nopeampia suurta tietomäärää klusteroitaessa.

Onlineklusterointia käytetään tilanteissa, missä ei ole mahdollisuutta saada koko dataa samanaikaisesti, eikä klusterien määrää tiedetä valmiiksi. Onlineklusteroinnissa klusterointi suoritetaan pienissä osissa, ja dataa voidaan lisätä klusterointiin jatkuvasti. Onlineklusteroinnissa klusteroinnin rakenne muuttuu jokaisen datan lisäyksen jälkeen,

Taulukko 2. Kasvojentunnistmalleja

Nimi	Kuvaus
Eigenfaces	Varhainen malli, joka perustuu kasvojen piirteiden perusteella luotuihin ominaisvektoreihin [4].
Haar Cascade	Perustuu ns. Haar-ominaisuuksiin: algoritmi tarkastelee vierekkäisiä suorakulmaisia alueita kuvassa. Se summaa pikselien voimakkuudet kussakin alueessa ja laskee näiden avulla alueiden eroavuudet [20].
Fisherface	Menetelmä perustuu ulottuvuuksien vähentämiseen käyttäen pääkomponenttianalyysiä, minkä jälkeen sovelletaan Fisherin lineaarisen diskriminantin menetelmää [21].
DeepFace	Syväoppimiseen perustuva menetelmä, jonka kehitti Facebookin tutkimusryhmä. Käyttää konvoluutioneuroverkkoja tunnistamaan ja erottelemaan kasvoja kuvista [8].
Sface	Syväoppimismalli ja perustuu niin ikään konvoluutioneuroverkkoihin. Huomattavasti kevyempi malli muihin verrattuna [22].
ArcFace	Menetelmä, joka käyttää marginaaleja perinteisen softmax-menettelyn lisäksi luokkien erotettavuuden maksimoimiseksi [23].
FaceNet	Syväoppimismalli, joka kartoittaa kasvokuvat euklidiseen avaruuteen, jossa etäisyydet vastaavat kasvojen samankaltaisuutta [24].
OpenFace	Avoimen lähdekoodin syväoppimismalli [25].

usein kuitenkin vain osassa klusterointia. Online-klusterointi on erittäin hyödyllistä tilanteissa, joissa koko data ei ole saatavilla kerralla tai sitä on käsiteltävä hyvin pitkän aikaa. Esimerkiksi reaaliaikaisissa sovelluksissa onlineklusteroinnin etuna on se, että jokaisen datan lisäyksen kohdalla koko klusterointia ei tarvitse tehdä uudelleen, vaan klusterointi päivittyy niiltä osin, mihin lisätty data vaikuttaa. Tällaisia järjestelmiä voivat olla esimerkiksi valvontajärjestelmät. [26, 27]

2.7. Robotin muistijärjestelmä

Ihminen tarvitsee muistia voidakseen suoriutua normaaleista päivittäisistä tehtävistään. Muistin avulla ihminen käsittelee ja hyödyntää aiemmin hankkimaansa tietoa. Myös uusien tietojen ja taitojen oppiminen perustuu muistiin. Muistaminen on tärkeää ihmisen sosiaalisuuden kannalta. Ihmisen muisti on yksi hämmästyttävimmistä järjestelmistä, sillä ihmisen aivot vastaanottavat valtavasti aistitietoa ja pystyvät silti käsittelemään sen hajautetusti. Näiden ajatusten perusteella voidaan kuvitella, että samankaltainen ihmismäinen muisti olisi hyödyllinen ominaisuus myös koneella. [28, 29]

Kognitiivinen eli tiedostava järjestelmä tarvitsee havainnointikomponentteja ympäristön havaitsemiseksi, prosessointikomponentteja oppiakseen ja sopeutuakseen muuttuviin olosuhteisiin sekä toiminnan toteutuskomponentteja tavoitteiden saavuttamiseksi. Havainnointi muodostaa kognitiivisten järjestelmien perustan, mahdollistaen tietojen keräämisen ympäristöstä eri aistimodalityettien avulla. Havainnointi mahdollistaa kognitiivisille järjestelmille esimerkiksi kyvyn tunnistaa objekteja ja ihmisiä. Prosessointikomponentit voivat käsitellä tietoja ja tallentaa tulokset lopulta takaisin muistiin. Toiminnan toteutuskomponentit antavat kognitiivisille järjestelmille mahdollisuuden kääntää prosessit tarkoituksellisiksi toimiksi. Toisin sanoen näiden komponenttien avulla robotti voi saavuttaa ennalta määrätyn tavoitteen. [28]

2.7.1. Erilaiset muistirakenteet ihmisellä

Vielä nykyisinkin on epäselvää, miten ihmisen muisti tarkalleen ottaen toimii. Nykyisen tietämyksen mukaan ihmisen muisti perustuu kolmeen erilaiseen prosessiin. Aluksi ihmisen aistijärjestelmään saapuva tieto havainnoidaan ja muunnetaan. Toiseksi se joko tallennetaan lyhyt- tai pitkäaikaiseen muistiin. Lopuksi tieto voidaan palauttaa tai muistaa, mikäli se ei ole unohtunut. Tämä kuvassa 1 esitelty teoreettinen ihmismuistin malli esittelee kolme muistityyppiä: aistimuisti, joka käsittelee havaintotietoa, pitkäaikaismuisti, joka säilyttää tietoa pitkän aikaa, ja lyhytaikaismuisti, joka säilyttää tietoa toistuvan harjoittelun kautta. [29, 28]

Aistimuisti säilyttää havaittua tietoa lyhyen ajan. Ihmisillä aistireseptorit ottavat vastaan esimerkiksi visuaalista, auditiivista tai kosketukseen perustuvaa tietoa ja välittävät sen suoraan hermostolle. Jos ihminen kiinnittää johonkin aistiin erityisen huomion, tieto siirtyy työmuistiin. Työmuistia käytetään joskus synonyyminä lyhytaikaisen muistin (eng. short-term memory, STM) kanssa, mutta jotkut pitävät näitä kahta muistin muotoa erillisinä ja väittävät, että työmuisti mahdollistaa

tallennetun tiedon manipuloinnin, kun taas lyhytaikainen muisti viittaa vain tiedon lyhytaikaiseen säilyttämiseen. Pitkäaikainen muisti (eng. long-term memory, LTM) säilyttää suuren määrän tietoa pitkään. Työmuisti voi noutaa tiedot pitkäaikaismuistista, kun niitä tarvitaan prosessointiin. [28, 29]

2.7.2. Yksinkertainen robotin muisti ja representaatiot

Yksinkertaisimmillaan robotin muistijärjestelmä voi perustua vain muutamiin tietorakenteisiin. Tässä ei syvennytä enempää erilaisiin tietorakenteisiin, mutta robotti voi esimerkiksi tallentaa sijaintinsa ja liikkeensä muuttuinaan, joka voi olla lista koordinaateista. Kehittyneemmät robotit tarvitsevat monimutkaisempia tietorakenteita ja algorimeja. Ominaisuudet, kuten kasvojentunnistus, eivät onnistu ilman suurempaa muistikapasiteettia ja laskentatehoa.

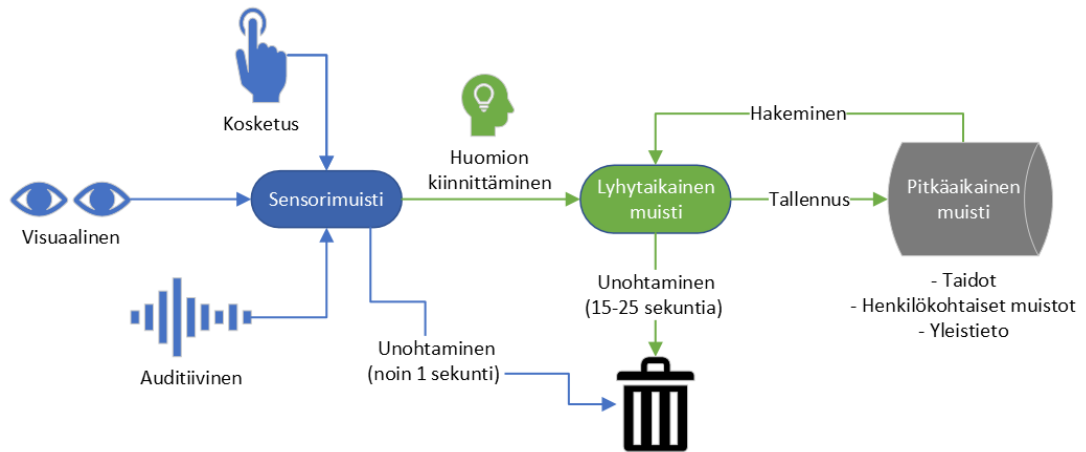
Kasvojentunnistusalgoritmit voivat tallentaa havaittujen kasvojen tietoja eri tavoin muistiin. Yleensä muistiin ei välttämättä tallenneta kuvatiedostoa. Sen sijaan käytetään jotain muuta representaatiota eli tieto esitetään jossakin muussa muodossa. Esimerkiksi numeeriset ominaisuudet, kuten ikä tai pituus, voivat olla suoria numeerisia arvoja. Kuvat taas voidaan esittää pikseleiden arvoina, väreinä tai moniulotteisena numeerisina vektoreina, jotka ilmentävät esimerkiksi kuvan muotoja ja reunaviivoja. Lisäksi voidaan käyttää tunnisteita, jotka liittyvät tunnistettuihin kasvoihin ja mahdollistavat niiden yksilöllisen seurannan ja tunnistamisen tietokannassa. Representaatioiden käyttäminen lisää algoritmien tehokkuutta huomattavasti, sillä niiden ei joka kerta tarvitse analysoida raakoja kuvatiedostoja.

2.7.3. Keinotekoiset kognitiiviset arkkitehtuurit ja muistijärjestelmät

Ihmisten kognitiivista järjestelmää on yritetty mukailla keinotekoisesti. Vaikka monenlaisia ratkaisuja on kehitetty, vielä ei kuitenkaan olla saatu aikaan toteutusta, joka pystyisi toimimaan samalla tavalla kuin ihmisen muisti. Keinotekoiset kognitiiviset arkkitehtuurit voidaan jakaa kolmeen osaan: [28]

- Kognitivistiset arkkitehtuurit, joissa tieto yleensä edustetaan symboleina.
- Emergentit arkkitehtuurit, jotka keskittyvät epäsymboliseen prosessointiin ja itseohjautuvaan oppimiseen siitä, miten tiedot liittyvät toisiinsa.
- Hybridiarkkitehtuurit, jotka yhdistävät molemmat yllä olevat arkkitehtuurityypit.

Nämä arkkitehtuurit sisältävät tyypillisiä ihmisen muistimoduuleja, kuten lyhyt- ja pitkäaikaisen muistin (ks. luku 2.7.1). Tulevaisuudessa humanoidirobotit kehittyvät ihmismäisempään suuntaan, ja tutkijat etsivät jatkuvasti ratkaisuja, kuinka kehittää roboteista älykkäämpiä ja tehokkaampia. [29] Muistia voidaan mallintaa laskennallisesti kolmen peräkkäisen varaston sarjana, kuten kuvassa 1. Ensimmäinen varasto vastaanottaa aistitiedon, toinen varasto on lyhytaikaisen muistin alue ja viimeinen varasto on pitkäaikaisen muistin alue. [29, 28]



Kuva 1. Ihmisen muistiluokat ja informaation käsittelyn prosessit. Myös keinotekoiset kognitiiviset järjestelmät mukailevat tätä mallia.

2.7.4. Tekniset vaatimukset

Perinteisen tietokoneen muisti voidaan luokitella kahteen osaan: työmuistiin (eng. working memory) ja tallennusmuistiin (eng. storage memory). Työmuisti toimii väliaikaisena tallennustilana suorittimelle ja käynnissä oleville ohjelmille. Se mahdollistaa nopean pääsyn tietoihin ja ohjelmiin, jotka ovat juuri aktiivisina käytössä. Tallennusmuisti eroaa työmuistista siten, että tallennusmuisti ei ole väliaikainen, vaan se säilyttää tiedot jopa silloin, kun tietokone sammutetaan. Aiemmat kovalevyt on korvattu nopeasti SSD-asetilla ja NAND-flash-muistilla. NAND-flash-muisti on haihtumaton, eli se tekee mahdolliseksi käyttäjätietojen ja ohjelmien tallentamisen, kun virta on kytketty pois päältä. [30]

Monimutkaiset ihmismäiset robotiikkajärjestelmät tuovat mukanaan haasteita muistijärjestelmien tekniselle toteutukselle. Robottien tulee pystyä käsittelemään todella suuria määriä dataa nopeasti. Sen takia kognitiivisen muistiarkkitehtuurin on oltava suunniteltu hajautetusti. Koska humanoidirobotin muistijärjestelmän kapasiteetit eivät voi olla niin suuria, että se pystyisi säilyttämään kaikkea dataa, täytyy muistia priorisoida, eli tallentaa ensisijaisesti tärkeitä informaatiota ja poistaa ylimääräistä ja epäoleellista dataa. [28]

2.7.5. Humanoidirobotin pitkäaikainen muisti

Kognitiivisia arkkitehtuureja tarvitaan ihmistoiminnan ennustamiseen ihmisen ja robotin välisessä vuorovaikutuksessa (ks. luku 3). Ihmisen ja robotin välinen vuorovaikutus (eng. human robot interaction, HRI) tutkii muun muassa ihmisen ja robottien suhteiden kehitystä pitkällä aikavälillä. Robotin muistot aiemmista vuorovaikutuksista voivat auttaa rakentamaan positiivisia suhteita sen kanssa aiemmin vuorovaikutuksessa olleiden henkilöiden kanssa. Tässä tapauksessa pitkäaikainen muisti on tärkeää. Jos robotin tarkoitukset keskittyvät käyttöön alueilla, joissa ihminen ja robotti eivät välttämättä tapaa useammin kuin kerran, on tärkeämpää kehittää ja suunnitella erityisesti lyhytaikaista muistijärjestelmää [31]. Muistin avulla

robotit oppivat ja sopeutuvat ihmisten tapoihin ja mieltymyksiin. Kun robotti muistaa vanhat keskustelut ihmisten kanssa, sen toiminnan ei enää tarvitse perustua sisäänrakennettuun kiinteään repertuaariin sosiaalisista käyttäytymismalleista. [32, 33]

Pitkäaikainen muisti jaetaan usein vielä deklarativiseen ja proseduraaliseen muistiin. Episodinen muisti on osa deklarativista muistia ja se sisältää henkilökohtaisia muistoja. Episodiseen muistiin tallentuu tietoa tapahtumien kontekstista, kuten paikasta, ajasta ja tunteista. Viime aikoina episodisen muistirakenteen mahdollisuuksia on alettu hyödyntämään robotiikassa, sillä sen avulla pystytään luomaan tehokkaammin toimivia sosiaalisia robotteja. [28, 34]

3. IHMISEN JA KONEEN VÄLINEN VUOROVAIKUTUS

Ihmisen ja robotin välinen vuorovaikutus (eng. human-robot interaction, HRI) on tutkimusala, joka käsittelee keinoja, joilla ihmiset ja robotit kommunikoivat ja toimivat keskenään erilaisissa ympäristöissä ja tilanteissa. HRI voidaan karkeasti jakaa neljään sovellusalueeseen:

- ihmisen ohjaama robotiikka rutiinitehtävissä, kuten osien käsittely tehdaslinjastoilla;
- etäohjaus erilaisiin ajoneuvoihin, kuten avaruusaluksiin ja sukellusveneisiin, erityisesti vaarallisissa tai vaikeasti saavutettavissa olevissa ympäristöissä;
- automaattiset ajoneuvot, joissa ihminen toimii matkustajana;
- sekä ihmisen ja robotin välinen sosiaalinen vuorovaikutus.

HRI:n ja robotiikan ero muodostuu pohjimmiltaan hieman eri tavoitteiden välille. Robotiikka keskittyy robottien luomiseen ja tutkimukseen siitä, miten ne voivat fyysisesti toimia, kun taas HRI tutkii, miten robotit toimivat sosiaalisissa ympäristöissä. [35, 36]

3.1. Lyhyt katsaus robotiikkaan

Tässä luvussa ei syvennyttä tarkemmin robotin fyysiseen rakenteeseen tai tietokoneen toimintalogiikkaan eli siihen, kuinka tietokone käsittelee ja ymmärtää sille syötetyn informaation. Mainittakoon kuitenkin: tietokoneet voivat vastaanottaa informaatiota käyttäjän syöttämänä, kuten näppäimistön tai kosketusnäytön kautta. Jo varhaisimmat tietokoneet perustuivat komentojen antamiseen jonkinlaisen käyttöliittymän kautta.

Nykyisinkin periaatteet ovat hyvin pitkälle samat, vaikka tekoäly tulee muuttamaan prosesseja paljon. Perinteinen tekstisyötteen käsittely voi perustua esimerkiksi säännöllisiin lausekkeisiin, ehdolliseen logiikkaan ja yksinkertaisiin hakualgoritmeihin. Tällaiset menetelmät voivat tunnistaa tiettyjä sanoja tai merkkijonoja ja suorittaa tiettyjä toimintoja niiden perusteella, kuten etsiä tietokannasta vastaavia tietoja tai suorittaa toimintoja ehtolausekkeen täytyessä.

Robotiikassa kiinnostavaa on erityisesti koneen kyky vastaanottaa informaatiota sensoreiden, kuten kameroiden, mikrofonien, ja erilaisten liikettä havaitsevien antureiden avulla. Nämä laitteistoelementit pystyvät keräämään robotin luonnollisen toiminnan kannalta tärkeää tietoa käyttäjistä ja ympäristöstä.

Lähtökohtaisesti ilman minkäänlaista ohjelmistoa (eng. software) toimivalla robotilla ei ole tavoitteita, muistia, tarpeita, kykyä oppia tai edes aistia ympäröivää maailmaa. Sillä on vain laitteisto (eng. hardware), aivan kuin pelkkä runko ilman tiedostavia aivoja. Robotista tulee täysin toimiva, kun laitteisto ja ohjelmisto integroidaan toisiinsa. Tämä mahdollistaa myös robotin ja ihmisen välisen vuorovaikutuksen. [35]

3.2. Luonnollisempi vuorovaikutus

Tässä alaluvussa tutkitaan ihmisen ja robotin vuorovaikutusta ihmisen näkökulmasta. Ihmiset luonnollisesti kommunikoivat eleiden, ilmeiden, kosketuksen ja liikkeiden avulla sekä tutkivat maailmaa katsomalla ympärilleen ja käsittelemällä fyysisiä esineitä. Keskeinen oletus tässä on, että heidän tulisi pystyä vuorovaikuttamaan teknologian kanssa samalla tavalla kuin he ovat tottuneet vuorovaikuttamaan todellisen maailman kanssa arkielämässä, kuten evoluutio ja koulutus ovat opettaneet [37, 38]. Tämä oletus asettaa haasteen teknologien kehittäjille, jotka pyrkivät luomaan luonnollisen vuorovaikutuksen ihmisen ja robotin välille.

Holistisissa tutkimuksissa, joissa robottien on annettu toimia autonomisesti ihmisten kanssa avoimissa ympäristöissä, on löydetty ratkaisuja HRI:n ongelmiin. Toisin sanoin on pystytty erittelemään ja tunnistamaan niitä vaatimuksia, jotka vaikuttavat ihmisen ja robotin vuorovaikutukseen. Tutkimusten perusteella kehiteltiin esimerkiksi differentiaalinen elastinen aktuaattori (DEA) (Lauria, Legault, Lavoie ja Michaud, 2008). Se mahdollisti turvallisen ja mukautuvan voimansiirron roboteissa. ManyEars-järjestelmä (Valin, Michaud ja Rouat, 2007) puolestaan tarjosi äänilähteen paikannuksen, seurannan ja erottelun kahdeksan mikrofoniin avulla, mikä paransi robotin puheentunnistusta. Tutkimusten aikana huomattiin, että interaktiivisen robotin kehittäminen on erittäin haastava tehtävä ja luonnollisen vuorovaikutuksen saavuttaminen vaatii jatkuvaa iteratiivista ja inkrementaalista prosessointia. [39, 38]

Nykyisin HRI-tutkimuksessa ajatellaan, että on olemassa useita erilaisia modaliteetteja, kuten ääni-, visuaalinen, haptinen, kinesteettinen ja proprioseptiivinen modaliteetti. Näitä modaliteetteja voidaan hyödyntää yksin tai yhdistelmänä mahdollistamaan erilaisia ihmisen ja robotin välisiä vuorovaikutustapoja, kuten äänikomentoja, visuaalisia eleitä ja fyysistä kosketusta. [40]

3.2.1. Verbaalinen ja visuaalinen vuorovaikutus

Monimuotoisessa ihmisen ja robotin välisessä vuorovaikutuksessa hyvin tärkeää on huomioida äänen ja visuaalisten eleiden merkitys. Äänimodaliteetti mahdollistaa verbaalisen kommunikaation ja visuaalinen modaliteetti antaa kyvyn tulkita visuaalisia asioita, kuten ilmeitä, eleitä, kehon kieltä ja sitä, mihin suuntaan ihminen katsoo. Roboteilla, jotka käyttävät visuaalisia modaliteetteja, on kyky havaita ympäristöään kameroiden avulla ja prosessoida visuaalista tietoa tietokonetekniikan algoritmeilla [40]. Tähän modaliteettiin sisältyy myös kasvojentunnistus.

Jotta robotti pystyy vuorovaikutukseen ihmisen kanssa, sen täytyy ymmärtää ihmisen eksplisiittiset eli suorat viestit. Tämä ei kuitenkaan yksistään riitä, kun tavoitellaan mahdollisimman luonnollista vuorovaikutusta. Tässä tapauksessa robotin täytyy kyetä tulkitsemaan myös implisiittisiä eli epäsuoria tai välillisiä ilmaisuja. [41]

Ilmeiden tunnistamisen lisäksi on tutkittu mahdollisuutta kommunikoida myös muilla kehon osilla, kuten käsien ja jalkojen liikkeillä. Tähän tarkoitukseen on ihmiskehosta tehty kolmiulotteisia malleja, jotka on jaettu eri osiin. Ihmisen liikkeet, mukaan lukien eleet, voidaan matemaattisesti esittää ominaisvektorien avulla. [42]

3.2.2. *Fyysinen vuorovaikutus*

Haptinen modaliteetti tarjoaa ihmiselle keinon kommunikoida kosketuksen kautta robotin kanssa. Tämä mahdollistaa fyysisen vuorovaikutuksen. Monissa robottileluissa on yksinkertaisia kosketusantureita, kuten nappeja. Lisäksi monimutkaisempia, hajautettuja ja kalliimpia antureita voidaan muuan muassa käyttää pistemäisten vuorovaikutusten havaitsemiseen. Tätä käytetään keinotekoisien ihon kehittämisessä. [40, 41] Kinesteettinen modaliteetti tarkoittaa robotin kykyä havaita ihmisen liikkeitä ja reagoida niihin. Lisäksi proprioseptiivinen modaliteetti luo robotille kyvyn tuntea oman kehonsa osien sijainnit, järjestyksen ja liikkeet suhteessa ympäristöön ja muihin ihmisiin. Näiden ominaisuuksien avulla robotti kykenee turvalliseen ja toimivaan fyysiseen vuorovaikutukseen ihmisten kanssa. [40]

Fyysisen vuorovaikutuksen kautta robotit voivat auttaa ihmisiä erilaisissa tehtävissä, kuten nostamalla raskaita esineitä tai avustamalla liikkumisessa. Fyysinen vuorovaikutus robotin ja ihmisen välillä ei kuitenkaan ole täysin ongelmatonta. Robotin ja käyttäjän välillä voi tapahtua törmäyksiä, jotka voivat aiheuttaa vakavia vammoja, jos robotti käyttää liikaa energiaa liikkeissään. Vammoja voidaan välttää tekemällä robotista mekaanisesti, sähköisesti ja ohjelmallisesti turvallisempi. Terävien kulmien poistaminen, liian suurten tehojen rajoittaminen ja joustavien komponenttien käyttö lisäävät turvallisuutta. Törmäyksiä voidaan myös ennakoita ja välttää erilaisten havaitsemismenetelmien ja sensorien avulla. [43]

3.2.3. *Tunteet ja niiden tulkitseminen*

Tunteet ovat ihmisen subjektiivisia kokemuksia, jotka voivat liittyä esimerkiksi mielihyvään, suruun, pelkoon, iloon, vihaan tai hämmennykseen. Kun ihmiset tapaavat toisiaan, keskustelevat ja muodostavat uusia ihmissuhteita, vuorovaikutuksessa olevat henkilöt kokevat lähes aina jonkinlaisia tunteita. Sama asia toistuu ihmisen ja robotin välillä. Vaikka robotti ei kykene edes tekoälyn avulla samalla tavalla kokemaan emotionaalisia tiloja kuin ihminen, voi se kuitenkin tunnistaa niitä ihmisessä algoritmien avulla. Tämä on tärkeää HRI:n kehityksessä, jotta tulevaisuuden sosiaaliset robotit voisivat toimia turvallisesti ja tarjota luonnollisen ja parhaan mahdollisen kokemuksen ihmiselle.

Tunteet voivat merkittävästi muuttaa viestiä: joskus ei ole niinkään tärkeää, mitä sanottiin, vaan miten sanottiin. Kasvot ovat yleensä näkyvin tapa ilmaista tunteita, mutta ne ovat myös helpoimmin hallittavissa erilaisissa sosiaalisissa tilanteissa verrattuna ääneen ja muihin ilmaisumuotoihin. Tunteiden tunnistaminen perustuu hyvin pitkälle juuri epäsuoriin eleisiin ja ilmeisiin ja niiden tulkintaan. Näiden usein hienovaraisen ja pienten liikkeiden ja ilmeiden tunnistaminen voi olla vaikeaa myös ihmiselle. Kuinka syvällisesti robotin tulisi ymmärtää nämä piirteet ollakseen älykäs? Mitä lähemmäs ihmisten tunteiden tunnistuskykyä kone pääsee, sitä luonnollisempi vuorovaikutustilanteesta tulee [44].

Tunteisiin liittyvä fysiologinen aktiivisuus voidaan erottaa nykyisin varsin tehokkaasti erilaisten fysiologisten signaalien avulla. Näitä signaaleja ovat sydämenlyöntitiheys, hengitystiheys, elektroenkefalografia (EEG) ja elektromyografia (EMG). Ahdistavat tunteet, kuten jännitys ja pelko voivat kasvattaa sykettä ja

hengitystiheyttä [45]. EEG-mittauksella voidaan mitata aivotoimintaa ja havaita aivomallien muutoksia, jotka liittyvät erilaisiin tunteisiin ja mielentiloihin. EMG-mittauksella voidaan mitata lihasten sähköistä aktiivisuutta, joka voi muuttua tunteiden mukana. Esimerkiksi kun ihminen on iloinen ja hymyilee, tietyt kasvojen lihaksista aktivoituvat. Viha taas voi aiheuttaa lihasjännitystä erityisesti hartioissa ja otsan lihaksistossa.

Edellä mainittujen fysiologisten signaalinen tulkinta vaatii nykytekniikalla kuitenkin hyvin herkkiä sensoreita, jotka tulisi käytännössä kiinnittää ihmisen kehoon. Tämä asettaa tiettyjä haasteita tunteiden tarkkaan tulkintaan ja analysointiin ihmisen ja robotin välisessä interaktiossa.

Koneet eivät ehkä koskaan tarvitse kaikkia niitä emotionaalisia taitoja, joita ihmiset tarvitsevat; kuitenkin on näyttöä siitä, että koneet tarvitsevat ainakin joitain näistä taidoista vaikuttaakseen älykkäiltä ja kyetäkseen tasokkaaseen vuorovaikutukseen ihmisten kanssa. Esimerkiksi tilanne, jossa ihminen puhuu toiselle jatkuvasti, mutta ei kuuntele ollenkaan voi tuntua keskustelussa huomiotta jäävälle osapuolelle epämiellyttävältä ja ärsyttävältä. Samalla tavalla robotin tulisi kuunnella ja vastailta ihmisen esittämiin kysymyksiin luontevalla tavalla, jotta sen kanssa olisi mukava ja helppo toimia. [44]

Usein robotteja suunnitellaan toimimaan avustavina apuvälineinä erilaisissa tilanteissa, kuten terveydenhuollossa, asiakaspalvelussa tai koulutuksessa. Näissä tilanteissa robotin kyky ymmärtää ja reagoida ihmisten tunteisiin voi olla erittäin hyödyllinen. Esimerkiksi terveydenhuollossa olisi hyvä, jos robotti kykenisi tunnistamaan potilaan tunnetiloja ja reagoimaan niihin asianmukaisesti.

3.3. Robotin käyttäytyminen ja antropomorfismi

Robotit voivat viestiä ihmisille niin ikään äänen, visuaalisen kanavan ja kosketuksen kautta. Ihmisen näköaisti mahdollistaa sen, että robotti voi tuottaa ymmärrettäviä viestejä valon, kuvien tai liikkeen avulla ihmiselle. Näköaistin lisäksi ihminen kykenee kommunikoidaan toisten ihmisten kanssa kuuloaistin avulla. Nämä kaksi kanavaa ovat tärkeitä myös ihmisen ja robotin välisessä vuorovaikutuksessa.

Yksi keino välittää visuaalisia viestejä robotilta ihmiselle on digitaalinen näyttö. Vaikka tällainen ratkaisu humanoidirobotin tapauksessa voi tuntua epäluonnolliselta, voidaan esimerkiksi kosketusnäytön avulla välttää ongelmia, joita kommunikaatio pelkästään äänen tai eleiden avulla saattaa aiheuttaa. [41]

Toinen tapa hyödyntää ihmisen visuaalista kanavaa robotille on liikuttaa sen kehoa tai sen osia. Tämä on erittäin tärkeää myös muilla kanavilla esitettyjen signaalien tukemiseksi. Kehonkielen käyttäminen on olennaista myös roboteille. Esimerkiksi silmien ja käsien ihmismäinen liikuttelu keskustelun aikana lisää vuorovaikutuksen luonnollisuutta. [41, 46]

Ääniviestit ovat ehdottomasti yksi kätevimmistä keinoista, joilla robotti pystyy kommunikoidaan ihmisten kanssa. Äänen avulla robotit voivat välittää monenlaista informaatiota ja tarjota käyttäjille mahdollisuuden kysyä ja saada vastauksia puheen avulla. Lisäksi ääniviestit voivat olla monikielisiä.

Haptiset signaalit ovat vähemmän merkittävä kommunikaatiotapa robotin ja ihmisen välillä, mutta niitä tarvitaan erityisesti silloin kun perinteiset visuaaliset tai

äänipohjaiset viestit eivät ole riittäviä tai soveltuvia. Haptiset signaalit välittyvät tuntoaistin kautta. Ne voivat sisältää erilaisia tuntemuksia, kuten värähtelyjä tai kosketuksia.

3.3.1. Vuorovaikutus tuntemattoman henkilön kanssa

Sosiaaliset humanoidirobotit alkavat yleistyä todellisissa käyttöympäristöissä, joissa tapahtuu jatkuvasti uusia spontaaneja kohtaamisia robotin ja ihmisen välillä. Tällaiselle robotille tärkeä kyky on osata lähestyä ihmistä oikealla tavalla, koska ihmiset saattavat epäröidä lähestyä konetta. Tämä johtuu yleensä siitä, että ihminen ei voi tietää valmiiksi tietyn robotin kyvyistä ja mahdollisuuksista auttaa ongelmatilanteissa. Robotin tulisi siis kyetä rohkaisemaan ihmistä vuorovaikutukseen niin, että tilanne tuntuisi luonnolliselta ja turvalliselta [47].

Eräässä tutkimuksessa tarkastellaan Robovie II -robotin ja ihmisen välistä vuorovaikutusta [48]. Tässä tutkimuksessa robotti pyrkii lapsenomaiseen aloitteellisuuteen. Tutkimuksessa vuorovaikutuksen etäisyys säädetään Hallin vuorovaikutuskategorioiden mukaisesti: intiimi (0–0,45 m); personaalinen (0,45–1,2 m); ja sosiaalinen (1,2–3,6 m). Nämä etäisyydet ovat lähtökohtaisesti ihmisten välistä vuorovaikutusta varten, mutta ne toimivat myös ihmisen ja robotin tapauksessa. Tutkimuksessa selvisi myös, jos robotti kohtaa ihmisen katseen syklistä, se mukailee ihmisten välisen katsekontaktin tyypillistä käyttäytymistä. Tämä osaltaan edistää vuorovaikutuksen luonnollisuutta, koska katsekontakti on tärkeä osa ihmisten välisiä vuorovaikutustilanteita. Lisäksi odotusaika eri toimien välissä ja liikkeen nopeus vaikuttavat siihen, koettaanko vuorovaikutustilanne luonnollisena vai epämukavana [48].

Ihmiset eivät yleensä puhu sosiaalisen etäisyyden ulkopuolella oleville ihmisille, mutta he voivat tunnistaa toistensa läsnäolon. Vuorovaikutus tällä etäisyydellä tapahtuu pääasiassa muuttamalla kehon asentoa. Tämä on tietenkin hyvin kulttuurisidonnainen asia, mutta lähtökohtaisesti keskustelu koettaisiin epäluonnolliseksi pelkästään siksi, että toista olisi vaikea kuulla ilman äänen korottamista. Koska ihmiset ovat usein epäröiviä, haluavatko he keskustella, kun he näkevät robotin ensimmäistä kertaa, tulisi robotin osata liikkua ensin sosiaaliselle etäisyydelle. Tämä vaatii robotilta kykyä ennakoita ihmisen liikkeitä niin, että se saavuttaisi luonnollisesti sosiaalisen etäisyyden ja pystyisi sen jälkeen aloittamaan keskustelun kohteen kanssa. Robotin lähestyminen kohti ihmistä on haastava ongelma, koska ihmisen täytyy pystyä tunnistamaan lähestymisyritys nonverbaalisesti etukäteen; muuten lähestytty henkilö ei ehkä tunnista robotin lähestyvän häntä tai voi ylläytyä robotin lähestymisestä. Ihmiset viestivät lähestymisen yrityksen hyvin katsekontaktilla, mutta ympäristössä, jossa on paljon ihmisiä, katsekontaktin luominen ei ole robotille yhtä helppoa [47].

Tervehdys on sosiaalinen normi ihmisten välisessä vuorovaikutuksessa. Kun kyseessä on uusi ihminen, tervehdys on usein asiallinen. Tuttujen ja ystävien kesken tervehdys voi olla rennompi. HRI:ssä, kun kommunikaatio tapahtuu ensimmäistä kertaa, voi robotti aloittaa keskustelun niin ikään tervehtimällä. Tämän jälkeen robotin tulee muuttaa käyttäytymistä sen perusteella, kenen kanssa hän on vuorovaikutuksessa. Toisin sanoen robotin tulisi kyetä identifioimaan henkilö, eli tunnistamaan henkilön ikä, sekä muita henkilökohtaisia piirteitä [47]. Näiden tietojen perusteella robotti

kykenisi paremmin vastaamaan yksilön tarpeisiin ja mieltymyksiin. Esimerkiksi vanhempien ihmisten kanssa robotit voivat kommunikoida hitaammin ja selkeämmin, kun taas lasten kanssa ne voivat olla leikkisämpiä.

Kun robotti on vuorovaikutuksessa uuden ihmisen kanssa, on kuitenkin järkevää ohjelmoida se niin, että se ei loisi itselleen liian rajattua kuvaa henkilöstä. Jos vuorovaikutus on vain lyhytaikaista, kuten auttaminen jossakin yksinkertaisessa asiassa, ei robotilla ole tarvetta muodostaa syvällistä kuvaa ihmisestä. Vasta kun on kyse sosiaalisista roboteista, jotka toimivat esimerkiksi kotiympäristössä, tai ovat osa henkilön jokapäiväistä elämää, on henkilön yksilöllisillä tarpeilla enemmän merkitystä.

3.3.2. Vuorovaikutus ennestään tutun henkilön kanssa

Sosiaalinen älykkyys pitää sisällään kyvyn ymmärtää toisten ihmisten tunteita, kehittyä sosiaalisissa vuorovaikutustilanteissa ja sopeutua yksilöllisiin tarpeisiin. Tulevaisuuden roboteilta odotetaan todennäköisesti edellä mainittuja ihmismäisiä piirteitä. Toisin sanoen erityisesti sosiaalisten robottien tulisi kyetä tutustua ihmiseen samalla tavalla kuin ihmiset tutustuvat toisiinsa. Tämän tavoitteen saavuttaminen vaatii vielä paljon tutkimista ja työtä, sillä aiheeseen liittyy käytännön ongelmien lisäksi myös useita eettisiä ja lainsäädännöllisiä haasteita. Näitä haasteita ovat muun muassa yksityisyydensuojaan ja luottamukseen liittyvät kysymykset.

Vaikka sosiaaliset robotit eivät ole vielä kovin yleisiä, on tietynlaisia sovelluksia jo laajasti käytössä. Esimerkiksi itseohjautuvaa autoa ei aina mielletä perinteiseksi robotiksi, mutta niissä on useita sosiaaliseen robotiikkaan liittyviä ominaisuuksia. Yksi sosiaalisen robotiikan tavoitteista on luoda robotteja, jotka voivat lisätä turvallisuutta, mukavuutta, tehokkuutta ja saavuttavuutta. Itseohjautuvat autot voivat vähentää onnettomuuksia, jotka aiheutuvat ihmisen inhimillisistä virheistä. Esimerkiksi väsymys tai keskittyminen johonkin muuhun kuin ajamiseen ja ympäristön tarkkailuun aiheuttavat paljon vakavia liikenneonnettomuuksia. Itseohjautuvat autot tarkkailevat ympäristöä jatkuvasti ja joka puolelta. Tämän avulla ihminen pystyy myös paremmin keskittymään muihin tehtäviin, kuten työntekoon. Itseohjautuvat autot voivat myös tarjota liikkumismahdollisuuksia ikääntyneille tai liikuntarajoitteisille.

Itseohjautuvasta autosta tulee sosiaalinen robotti, jos siihen lisätään ominaisuuksia henkilön tarpeiden ja tapojen perusteella. Jos auto esimerkiksi muistaa ja ehdottaa tiettyä matkaa tietynä ajankohtana tai se voi tulkita ihmisen olotilaa ja tehdä sen perusteella valintoja, auto toimii ikään kuin tietynlainen sosiaalinen robotti.

Tutkimukset ovat nostaneet esille myönteisiä merkkejä sosiaalisten robottien hyväksymisestä kohteiden omassa kotiympäristössä. Vaikka robotin hyväksyminen noudattaa jonkin verran samankaltaista hyväksymisprosessia kuin muiden teknologioiden, robotin täydellinen integrointi henkilöiden päivittäiseen elämään näyttää riippuvan heidän käsityksestään robotista. Jotta sosiaalinen robotti hyväksyttäisiin osaksi arkea, on tärkeää, että henkilö kokee sen hyödylliseksi. Tutkimusten perusteella jotkut ihmiset, jotka eivät kokeneet robottia tarpeelliseksi tai hyödylliseksi, lopettivat sen käytön ensimmäisten kokeilujen jälkeen. Erityisesti yksityisyyden, luottamuksen ja käyttäytymisen hallinnan negatiivinen arviointi johti robotin käytön lopettamiseen [49].

Tulevaisuuden palveluroboteille kehitetyt sovellukset vaativat usein intensiivistä kommunikaatiota ja vuorovaikutusta ihmisten kanssa. Kommunikaatioprosessi ei kuitenkaan ole pelkästään tiedonvaihtoa, vaan se on aina merkityksellinen robotin ja ihmisen välisen suhteen kehityksen kannalta. Monimutkaiset sosiaaliset vuorovaikutukset kehittyvät ajan myötä. Kaikki yksilön tähän mennessä luomat sosiaaliset suhteet ovat edellisten sosiaalisten vuorovaikutusten tulosta. Kokemukset, joita ihminen kerää elämän aikana, vaikuttavat yksilön luonteen kehittymiseen. Jos joitakin kokemuksia valikoivasti poistettaisiin, voisi syntyä epäjohdonmukaista käytöstä [33].

Pitkäaikaisessa vuorovaikutuksessa robotin kanssa vuorovaikutuksessa oleva henkilö väistämättä jossain vaiheessa tottuu robotin läsnäoloon ja oppii ymmärtämään myös robotin kyvyt ja taidot. Tässä vaiheessa ihmisen käsitys robotista ei ole enää vääristynyt, vaan robotin käytös on arvattavampaa ja sen myötä myös miellyttävämpää [32].

Robottien kehittyminen kohti ihmismäisempää olemusta ja käytöstä tuo mukanaan useita eettisiä ongelmia. Eräs keskeinen kysymys on, millainen suhde ihmisen ja robotin välillä on sosiaalisesti hyväksyttävää tulevaisuudessa? Aihe vaatii pitkäaikaista tutkimusta ja pohdintaa, sillä robotiikka tulee yleistymään kodeissa ja työelämässä. Sosiaaliset älykkäät robotit saattaisivat aiheuttaa ihmisten sosiaalisten taitojen heikkenemistä ja johtaa kyvyttömyyteen selviytyä sosiaalisista tilanteista ilman teknologista apua. Lisäksi, jos robotit eivät kykene käsittelemään ihmisten tunteita ja tarpeita oikein, se voi johtaa ihmisten epäluottamukseen teknologiaa kohtaan. Tulevaisuudessa tarvitaan myös ratkaisu siihen, miten robottien keräämiä henkilökohtaisia tietoja voidaan käsitellä turvallisesti ja niin, että niitä ei käytetä väärin esimerkiksi ihmisten yksityisyyden loukkaamiseen.

Ihmiset voivat muodostaa erilaisia suhteita robottien kanssa riippuen robotin kyvyistä, suunnittelusta ja käyttäjän yksilöllisistä mieltymyksistä ja tarpeista. Jotta tulevaisuuden robottien integraatio yhteiskuntaan ja monimutkaiseen sosiaaliseen järjestelmään onnistuisi, robotit on varustettava tarkoituksenmukaisilla sosiaalisilla taidoilla. Myös sosiaalisten robottien tuomat eettiset ongelmat vaativat paljon tutkimusta.

3.3.3. *Antropomorfismi*

Tulisiko robotin olla ihmisen kaltainen ja muotoinen? Olisiko järkevämpää, että robotin ulkonäkö ja rakenne kehittyisivät pääasiassa sen toiminnallisuuden perusteella? Antropomorfismi on käsite, joka viittaa ihmismäisten ominaisuuksien tai piirteiden antamiseen ei-inhimillisille asioille, kuten eläimille, koneille tai abstrakteille käsitteille. Antropomorfinen suunnittelu on merkittävä osa robotiikkaa ja HRI:ta, koska robotin ulkonäkö ja toiminta vaikuttavat siihen, miten ihmiset toimivat sen kanssa. Koska ihmisten mieltymykset vaihtelevat tarpeiden ja kontekstien mukaan, tulisi robotin ulkonäön vastata näitä odotuksia. Yleisesti ottaen ihmisten reaktiot määräytyvät tietyn mielikuvan perusteella. Jos robotti näyttää elävältä olennotta, sen käyttäytymisen odotetaan lähestyvän mielikuvaa ihmisen tai lemmikkieläimen käyttäytymisestä. Kun nämä mielikuvat eivät vastaa todellisuutta voi tilanteesta

aiheutua odottamattomia ja epämielisiä seurauksia robotin kanssa interaktiossa olevalle ihmiselle. [50]

Liiallinen antropomorfismi, kuten lähes täydellisesti ihmistä muistuttavat humanoidit, voi herättää ihmisissä pelkoa tai hylkimistä. Outo laakso (eng. Uncanny Valley) on termi, joka viittaa ilmiöön, jossa ihmisten mielipide ja reaktiot humanoidiroboteihin tai muihin epätodellisen ihmismäisiin olentoihin muuttuvat epämiellyttäväksi tai ahdistaviksi, kun näiden olentojen ulkonäkö ja käyttäytyminen muistuttavat liikaa oikeita ihmisiä. Hallin spatiaalisen vyöhyketeorian mukaan niin kauan kuin robotit voidaan edelleen erottaa biologisista organismeista, mikä voi olla todellisuutta vielä pitkään, on epätodennäköistä, että ihmiset reagoivat robotteihin täysin samalla tavalla kuin he saattavat reagoida ihmisiin tai muihin eläviin olentoihin samankaltaisissa tilanteissa. Tämä ilmiö on kiehtonut ihmisiä paljon ja aiheesta on tehty useita tutkimuksia. [51, 52]

Tehokas kommunikaatio ihmisen ja robotin välillä riippuu siitä, onko heillä yhteinen ymmärryksen alue. Rajapinnan suunnittelu myös robotin ulkomuodon näkökulmasta on ratkaisevan tärkeää, jotta ihminen olisi tietoinen robotin mahdollisuuksista ja pystyisi niin luonnollisesti vuorovaikuttamaan robotin kanssa. [43]

3.4. Tekoälyn vaikutus HRI:n kehitykseen

Robotiikassa ja HRI:ssä mielenkiintoisia ovat erityisesti seuraavat tekoälyn sovellusalueet: kyky autonomiaan, yksittäisten taitojen edistyminen riittävälle tasolle sekä oppiminen ja soveltamiskyky tekoälyn avulla. [53]

3.4.1. Tekoälyagentti ja robotin autonomisuus

Tekoälystä puhuttaessa agentti on entiteetti, joka pystyy tekemään havaintoja ympäristöstään ja toimimaan niiden perusteella. Robotin voidaan ajatella olevan erikoistapaus agentista, koska se sijaitsee fyysisesti "todellisessa maailmassa". [53]

Milloin robottia sitten voidaan kutsua älykkääksi? Vaikka robotti onkin fyysisesti olemassa oleva agentti, tämä ei riitä älykkyyden määritelmäksi. Lisäksi robotin tulee tulella maksimoida mahdollisuudet menestyä sille annettussa tehtävässä. Tämä vaatii monimutkaista havainnointi- ja optimisaatiokykyä. [53] Tämän lisäksi robotin tulee toimia autonomisesti. Robotiikassa autonomialla tarkoitetaan kykyä tehdä päätöksiä ja toimia ilman jatkuvaa ihmisen ohjausta. Autonomiaan liittyviä operatiivisia arkkitehtuureita:

- *Prosessiautonomia (eng. process autonomy)*. Robotti voi valita algoritmin, jota se käyttää saavuttaakseen tavoitteensa.
- *Järjestelmätila-autonomia (eng. systems-state autonomy)*. Robotti voi generoida ja valita mahdollisista vaihtoehdoista, joiden avulla se saavuttaa tavoitteensa. Esimerkiksi omaan tilan hallintaan liittyvät ominaisuudet, kuten akun lataus.
- *Aikomuksellinen autonomia (eng. intentional autonomy)*. Robotti voi vaihtaa sen tavoitetta saavuttaakseen jonkin muun tavoitteen.

- *Rajoiteautonomia* (eng. *constraint autonomy*). Robotti voi itse luoda omia tavoitteita ja rooleja. Tämä antaa robotille käytännössä hyvin laajat mahdollisuudet toimia oman tahtonsa mukaan.

3.4.2. *Tekoälyn mahdollisuudet HRI:ssä*

Tulevaisuudessa ihmisen ja robotin välisessä multimodaalisessa vuorovaikutuksessa hyödynnetään useita tekoälyn liittyviä tekniikoita. Kameroilla varustetut robotit voivat käyttää konenäköä tunnistamaan ihmisten kasvoja, liikkeitä ja eleitä. Konenäön avulla robotit voivat tehdä myös muita havaintoja ympäristöstä, kuten tunnistaa erilaisia esineitä tai esimerkiksi sääolosuhteita. Konenäköalgoritmit voivat tunnistaa myös katsekontaktit ihmisen ja robotin välillä, mikä on merkityksellinen kehitysaskel kommunikaation kannalta. Useiden kameroiden yhdenaikainen käyttö ja lineaarinen kalibrointi mahdollistavat 3D konenäön soveltamisen. Tällä voidaan välttää robottien törmäyksiä. [40, 54] Konenäöstä on kattava selvitys tämän kandidaatintyön luvussa 2.

Luonnollisen kielen käsittely (eng. *natural language processing, NLP*), on tietojenkäsittelytieteen haara, joka tutkii ensisijaisesti tietokoneiden kykyä ymmärtää ja tuottaa ihmisten kieltä. Luonnollisen kielen lause esitetään syntaksipuuna, joka näyttää lauseen eri osien väliset suhteet. Jokaiselle lauseelle oletetaan olevan vastaava yksi ja ainoa syntaksipuun. Jos lause voidaan luoda käyttäen useampaa kuin yhtä syntaksipuuta, niin lauseen kieltä ja kielioppia kutsutaan moniselitteisiksi – lauseella on useampi kuin yksi merkitys. [55]

Laajojen kielimallien (eng. *large language model, LLM*) kehitys on ollut huimaa viime vuosina. LLM:t antavat käyttäjille mahdollisuuden luoda, testata ja ajaa robottia käyttämällä luonnollisen kielen syötteitä, ilman syvempää tietämystä robotin toiminnan periaatteista ja algoritmeista. Laajoja kielimalleja voidaan siis käyttää ihmisen puheen tunnistamiseen. Sen myötä robotit ymmärtävät erilaisia komentoja tai kehoitteita paremmin. [56, 40]

Robotin käyttäytymisen kannalta merkittäviä tekoälysovelluksia ovat puhesynteesi, visuaalinen palaute ja "tunteiden" ilmaisu. Puhesynteesin avulla robotti voi tuottaa ihmisille ymmärrettäviä lauseita eri kielillä. Visuaalisella palautteella viitataan robotin mahdollisiin eleisiin ja näkyviin viesteihin. Tekoälyn avulla robotit pääsevät lähemmäs ihmisen kaltaista tunneälyä ja ne voivat sen kautta mukailla esimerkiksi empaattisia ilmauksia, jos ihminen on surullinen ja kaipaa lohdutusta. [40]

3.4.3. *Robotin opettaminen*

Robotin käyttäytymisen opettamiseen voidaan käyttää erilaisia opetustapoja ja ongelmanratkaisutekniikoita. Imitaatio-oppiminen robotilla edellyttää, että jokin muu toimija suorittaa malliksi tehtävän. Toimija voi olla ihmisen lisäksi vaikkapa toinen robotti. Toinen suosittu keino on vahvistusoppiminen (eng. *reinforcement learning*). Robotille annetaan toiminnan mukaan palautetta, joka voi olla positiivista tai negatiivista. Algoritmi kannustaa robottia toimimaan oikein ympäristön antaman positiivisen palautteen avulla ja välttelemään niitä toimia, jotka aiheuttavat negatiivista

palautetta. Ratkaisevan tärkeää on myös, että robotin opettaminen tapahtuisi todellisissa tilanteissa, joissa ne toimivat. [41]

4. TOTEUTUS

Tämän kandidaatintyön toteutuksen tavoitteena on kehittää aiempien opiskelijoiden luomaa messuesittelyrobotin kasvojentunnistusjärjestelmää lisäämällä siihen kyky muistaa aiemmin tunnistettuja kasvoja. Tämä parantaa robotin vuorovaikutuskykyä tunnistamalla aiemmin kohtaamiaan ihmisiä ja mahdollistaa robotin käyttäytymisen mukauttamisen tilanteissa, jossa robotin kanssa vuorovaikutuksessa on ennestään tuttu henkilö.

Tässä kandidaatintyössä keskitytään messuesittelyrobotin kasvojentunnistusjärjestelmän kehittämiseen. Työssä kehitetään messuesittelyrobotille alemman tason moduuli, joka kasvojentunnistuksen avulla tallentaa ja tuottaa tietoa robotin näköpiirissä olevista henkilöistä, ja lähettää tiedon eteenpäin ylemmän tason moduuleille. Työssä hyödynnetään deepface-python-kirjastoa [57], joka tarjoaa tehokkaan työkalun kasvojentunnistukseen. Kasvojen klusterointiin työssä käytetään Links online -klusterointia [58].

Messuesittelyrobotin ohjaamiseen käytetään ROS2-käyttöjärjestelmää. ROS2 eli Robot Operating System 2 on avoimen lähdekoodin ohjelma robottien korkeatasoiseen ohjelmointiin. Se toimii Ubuntu linux -jakelun päällä ja tarjoaa paljon erilaisia työkaluja robottien ohjelmointiin. ROS2:n perustuvat järjestelmät jaetaan *solmuihin* (eng. node), jotka voivat kommunikoida *julkaisemalla* (eng. publish) ja vastaanottamalla viestejä niiden tilaamissa *aiheissa* (eng. topic). [59]

4.1. Sovellusympäristö ja sen haasteet

Messuesittelyrobotin pääasiallinen käyttöympäristö on hakijanpäivillä, joissa vieraillee paljon ihmisiä ja esiintyy monenlaisia tilanteita. Robotin kasvojentunnistusjärjestelmän on kyettävä toimimaan näissä muuttuvissa tilanteissa ja tuottamaan merkityksellistä ja oikeellista tietoa robotin ylemmille toiminnasta vastaaville moduuleille. Robotin luo voi saapua yksi tai useampia ihmisiä samanaikaisesti. Lisäksi sen ohitse kulkee jatkuvasti ihmisiä, jotka eivät välttämättä pysähdy vuorovaikutukseen.

Jos robotin kanssa vuorovaikutuksessa oleva ihminen kääntää päätä, ei robotti välttämättä hetkellisesti näe kasvoja. Robotin täytyy siis tunnistaa, onko ihminen todella lähtenyt pois vai kääntynyt vain hetkellisesti.

Kasvojentunnistuksen ja vertailun on oltava riittävän tarkkoja, jotta robotin toiminta voi olla uskottavaa. Jos kasvojentunnistusjärjestelmä tunnistaa ihmisen usein väärin, ei robotti ole lainkaan ihmismäinen. Täysin varmaan tunnistukseen tuskin kuitenkaan päästään, joten täytyy olla jokin keino tunnistaa ja suodattaa virheellisiä tuloksia. Esimerkiksi: robotti tunnistaa ihmisen oikein aluksi 10 sekunnin ajan, mutta yhden sekunnin luuleekin tätä joksikin toiseksi ja sitten taas tunnistaa oikein.

Haasteena kasvojen tunnistuksessa on siis reaaliaikaisuus, tarkkuus ja virheiden tunnistaminen sekä niiden suodattaminen pois. Järjestelmän täytyy olla riittävän nopea, jotta se on mahdollisimman ihmismäinen. Jos robotilta menee minuutti tunnistaa sen kanssa keskustelevalle ihmiselle, menettää se uskottavuuden. Toteutuksessa nämä haasteet on pyritty huomioimaan ja niihin on esitetty erilaisia ratkaisuja.

4.2. Olemassa oleva toteutus

Messuesittelyrobotin ohjelmistossa on kaksi aiempaa toteutusta kasvojen tunnistamisesta nimeltään `Face_tracker` ja `emotion_detection`. `Face_tracker` paikantaa kasvot kuvasta ja lähettää kasvojen sijainnin robotin liikkumisesta vastaavalle moduulille. Lisäksi moduuli tunnistaa, että puhuuko kuvassa näkyvä henkilö analysoimalla huulien etäisyyksi. `Face_tracker` käyttää Dlib-kirjastoa kasvojen löytämiseen (face detection) ja Dlibin kasvopiiirteiden tunnistusta (eng. facial landmark detection) puhumisen tunnistamiseen. Dlibin korrelaation seurantaominaisuutta (correlation tracking) kasvojen seuraamiseen, jotta kasvojen löytämistä ei tarvitse tehdä jokaiselle videon kuvalle.

`Emotion_detection` -moduuli on rakennettu erityisesti kasvojen tunteiden tunnistamiseen. `Emotion_detection` tunnistaa kasvot kameran kuvasta, valitsee suurimmat kasvot ja tunnistaa kasvojen tunteet. Kasvojen löytämiseen käytetään Yunet-mallia.

Messuesittelyrobotin ohjelmisto ja ROS2-käyttöjärjestelmä pyörii Ubuntulla Linux jakelulla. Kehitystyön helpottamiseksi ohjelmisto voidaan asentaa virtuaalikoneelle. Ohjelmiston asentamista virtuaalikoneelle on pyritty helpottamaan käyttämällä Vagrant-ohjelmaa virtuaalikoneen luontiin. Vagrant-ohjelma ajaa virtuaalikoneen luonnin yhteydessä skriptin, joka asentaa tarvittavat ohjelmistopaketit.

Kasvojentunnistamista lähdettiin rakentamaan olemassa olevan `face_tracker` -moduulin päälle, koska tätä moduulia käytetään olemassa olevassa kasvojenliikkeen toteutuksessa. Tätä ominaisuutta pidettiin tärkeämpänä, kuin kasvojen ilmeiden ja tunteiden tunnistusta. Lisäksi `deepface`-kirjastossa on toteutus kasvojen analysointiin, kuten tunnetilojen, iän ja sukupuolen tunnistamiseen, ja tämän ominaisuuden integrointi tulevaisuudessa ei pitäisi tuottaa suuria haasteita.

4.2.1. *Face_tracker* moduulin käyttöönotto

Olemassa olevan toteutuksen käyttöönotossa tuli vastaan ongelmia. Ohjelmiston moduulien käyttämien Python-pakettien välillä oli riippuvuusongelmia. Nämä ongelmat saatiin ratkaistua poistamalla `emotion_detection` -moduuli. Tämän muutoksen myötä ohjelmisto alkoi toimimaan MacBook Pro -tietokoneella, kun käytössä oli tietokoneen sisäänrakennettu kamera. Hieman vanhemmalla Windows-pöytätietokoneella (2013) ohjelmisto ei kuitenkaan alkanut toimimaan, vaan virtuaalikone ja kaikki tarvittavat ohjelmistot tuli asentaa manuaalisesti, jotta ohjelmisto alkoi toimimaan. Haasteita tuli vastaan myös ulkoisen kameran (Logitech c310) kanssa. Videokuva toimi vain alhaisella kuvataajuudella ja videokuvan kirkkaus vaihteli ylivalottuneesta alivalottuneeseen jatkuvasti. Myöhemmin huomattiin ongelman syyksi puuttuva `mjpg`-pakkaus.

Alun perin videokuvan lukemiseen kameralta käytettiin avoimen lähdekoodin `opencv_cam` ROS-ajuria. [60] Ajurin kanssa tuli vastaan ongelmia, koska siihen ei ollut toteutettu `mjpg`-pakkausta. Tästä syystä luotiin oma kamera-ajuri käyttäen Python `opencv` -kirjastoa. Ajuri julkaisee videokuvan ROS2 aiheeseen, josta `face_tracker` moduuli tilaa videokuvan (eng. subscribe). Tästä ROS2 aiheesta uusi kuva luetaan

aina, kun sellainen on saatavilla, tai niin usein, kuin face_tracker pystyy käsittelemään videokuvaa.

4.2.2. Refaktorointi

Alkuperäisessä face_tracker moduulin toteutuksessa lähes kaikki toiminnallisuus oli yhdessä funktiossa. Se teki moduulista vaikeasti luettavan ja sen kehittämisen vaikeaksi, koska sisäkkäisiä if-lauseita ja silmukoita oli paljon. Funktio refaktoroiitiin siirtämällä osia siitä omiin funktioihin. Refaktoroinnin ansiosta koodista tuli huomattavasti luettavampaa ja helpommin ylläpidettävää, mikä helpotti työn jatkamista.

4.3. Kuinka kasvojentunnistus toteutettiin?

Toteutusta lähdettiin tekemään ratkaisten osaongelma kerrallaan. Toteutuksessa käytetään yleisesti käytössä olevaa deepface-python kirjastoa. Aluksi kokeiltiin käyttää deepfacen sisäänrakennettua find-funktiota, joka etsii kuvasta kasvot, laskee kasvojen vektoriesityksen ja vertailee vektoriesitystä kuvatietokannan kuvien vektoriesitykseen. Kokeiltiin myös deepfacen stream-funktiota, joka tekee vastaavat toimet videostreamille.

Nämä deepfaceen sisäänrakennetut funktiot eivät kuitenkaan olleet sopivia messuesittelyrobotin kasvojentunnistamisjärjestelmään, sillä ne pyrkivät löytämään kuvassa olevat kasvot olemassa olevasta tietokannasta. Messuesittelyrobotin tulee sen sijaan pystyä erottelemaan ennestään tuntemattomat kasvot.

4.3.1. Kasvojen löytämis- ja kasvojentunnistusmallien valinta

Kasvojen tunnistamista varten täytyy valita, mitä kasvojen löytämis- ja tunnistamismalleja käytetään. Jotta käytettävien mallien vertailu olisi helppoa, toteutuksessa käytetään deepface-kirjastoa [8]. Kirjaston avulla voidaan käyttää useita yleisesti käytössä olevia edistyksellisiä kasvojentunnistus- ja kasvojen löytämismalleja. Kirjastoa käyttämällä voidaan valita, mitä malleja käytetään kasvojen löytämiseen ja niiden tunnistamiseen. Kasvojen löytämiseen on käytettävissä mallit OpenCV, SSD, Dlib, MTCNN, Faster MTCNN, RetinaFace, MediaPipe ja YuNet. Kasvojen tunnistamiseen on käytettävissä mallit VGG-Face, Facenet, Facenet512, Openface, DeepFace, DeepID, ArcFace, Dlib, SFace ja GhostFaceNet [57]. Mallien välillä on suuria eroja kasvojentunnistamisen ja löytämisen tarkkuudessa sekä mallien nopeuksissa. Deepfacen mallien kombinaatioiden tarkkuuksia on vertailtu tarkasti [61], mutta mallien välisistä nopeuksista löytyy heikommin tietoa. Mallien nopeus on tärkeää, koska messuesittelyrobotin kasvojentunnistusjärjestelmän tulee pystyä toimimaan reaaliaikaisesti.

Deepfacen kasvojentunnistumalleista FaceNet-512d (98,4%) ja FaceNet-128d (97%) ylittää ihmisen kasvojentunnistamiskyvyn (97%), Dlib (96,8%), VGG-Face (96,7%) ja ArcFace (96,6%) saavuttavat ihmisen kanssa samantasoisien

kasvojentunnistuskvyn. SFace (93%) ja GhostFaceNet (90.5%) eivät aivan saavuta ihmisen kasvojentunnistuskvya. OpenFace (78.7%), DeepFace (68.7%) ja DeepID (65.6%) jäävät huomattavasti ihmisen kasvojentunnistuskvyn alapuolelle. Tuloksissa on käytetty mallille parasta kasvojen löytämismallia [61]. Heikoimmin suoriutuneet mallit OpenFace, DeepFace ja DeepID eivät ole käyttökelpoisia messuesittelyrobotin kasvojentunnistusjärjestelmään.

Kasvojenlöytämismallien nopeuksien vertailua varten luotiin testiympäristö, jossa videoita voidaan ajaa deepface:n kaikilla malleilla. Testi käy videon läpi kuva kovalta, ja kasvojentunnistusmalli tunnistaa videosta kasvot. Kasvot rajataan kuvasta ja rajattu kuva syötetään kasvojentunnistamismallille. Kasvojentunnistamismalli laskee kasvoista vektoriesityksen, jota verrataan tietokannassa olevista kasvoista laskettuihin vektoriesityksiin. Testattaessa kasvojenlöytämismalleja kasvojentunnistusmallit eivät olleet käytössä. Testissä mitattiin keskimääräistä kuvataajuutta, jolla testiä pystyttiin ajamaan. Lisäksi testi tallensi tiedon siitä, kuinka monesta kuvasta kasvojen löytämismalli löysi kasvot.

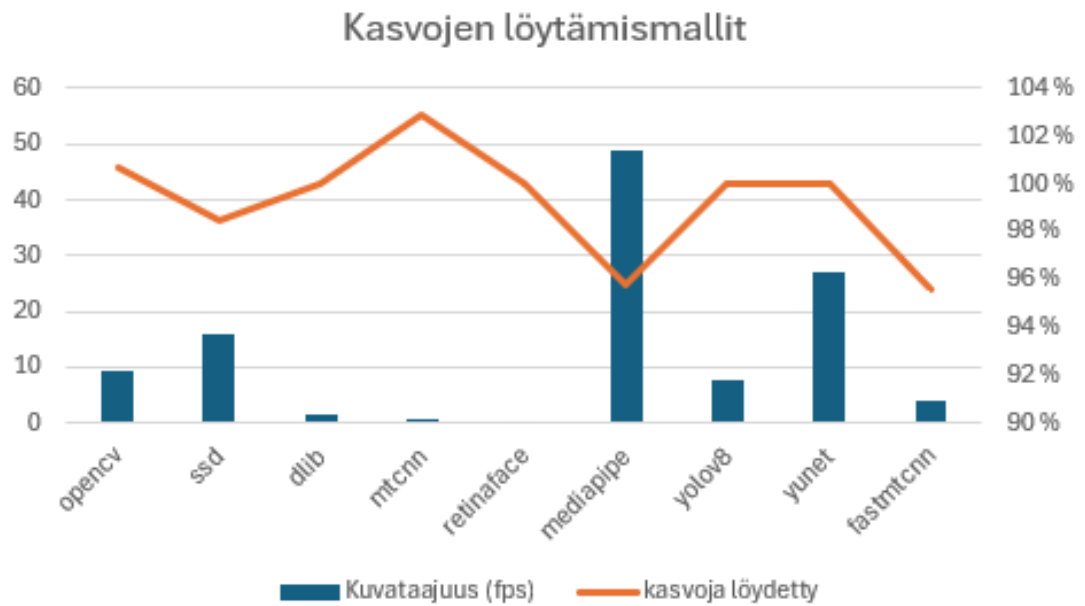
Suorituskykytestit ajettiin yhdellä 15 sekunnin videolla, jonka resoluutio on 1280x720 ja kuvataajuus 29,89 kuvaa sekunnissa. Videossa yksi henkilö liikkui ja käänteli päätä puolelta toiselle siten, että videon jokaisessa kuvassa kasvot ovat tunnistettavissa. Kasvojen löytämismallien tulisi siten tunnistaa kasvot jokaisesta kuvasta, ja kasvojentunnistamismallin tulisi löytää tietokannasta vastaavat kasvot jokaiselle kuvalle.

Suorituskykytestauksessa käytössä oli tietokone, jossa on Intel Core i7-4770 - prosessori ja 20 Gt keskusmuistia, ja testaus suoritettiin prosessorilla. Pullonkaulaksi testauksessa muodostui prosessorin suorituskyky, ja kaikilla malleilla prosessorin käyttöaste oli 100% testauksen aikana.

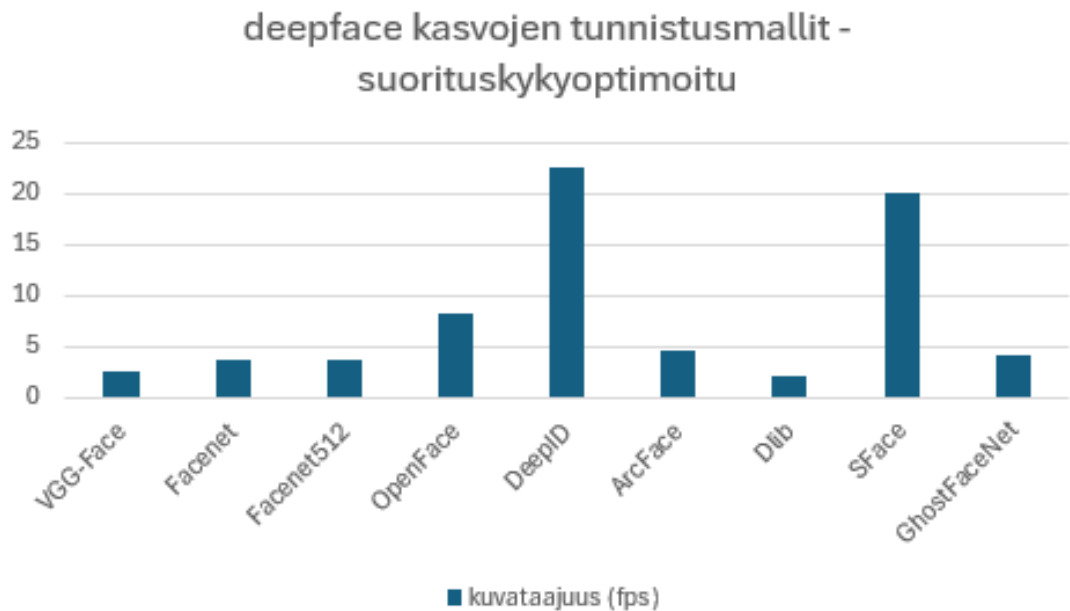
Kasvojen löytämismallien testin tuloksen näkyvät kuvassa 2. Tuloksista nähdään, että malleista mediapipe on selkeästi nopein, saavuttaen keskimääräisen kuvataajuuden 49.0 kuvaa sekunnissa. Toiseksi nopein on yunet malli (27,1 fps). Ssd- (15,9 fps) ja opencv (9,4 fps) -mallit tulevat yunet-mallin perässä. Nopein kasvojenlöytämismalli mediapipe suoriutuu YuNet mallia heikommin melkein kaikkien kasvojentunnistusmallien kanssa. [61]

Kasvojentunnistamismallien suorituskykytestauksessa käytettiin yunet-mallia kasvojen löytämiseen. Tulokset löytyvät kuvasta 3. Selkeästi nopeimmiksi malleiksi osoittautuivat DeepID (22,5 fps) ja SFace (20,1 fps). Näiden mallien perässä tulee Openface (8,2 fps). ArcFace (4,47 fps), GhostFaceNet (4,07 fps), Facenet512 (3,616 fps) ja Facenet (3,611 fps) saavuttavat samankaltaiset nopeudet. Hitaimmat mallit ovat VGG-Face (2,59 fps) ja Dlib (2,17 fps).

Nopeimmaksi osoittautunut kasvojentunnistusmalli DeepID tarkkuus ei ole messuesittelyrobotille riittävä. YuNet kasvojenlöytämismalli on nopea, ja on parhaimpien kasvojenlöytämismallien kaikkien deepface kasvojentunnistusmallien kanssa. Kasvojentunnistumalleista SFace mallin tarkkuus ja nopeus muodostavat parhaimman yhtälön. Käyttäen SFace mallia ja Yunet mallia yhdessä, kasvojentunnistuksen tarkkuudeksi voidaan saada 91,0%. Jos kasvojen löytämismalliksi valitaan nopeampi mediapipe, SFace:n tarkkuus on vain 75,4% [61]. Sface ja YuNet malleilla kasvojentunnistus saadaan reaaliaikaiseksi myös tietokoneen prosessorilla ajaessa. Toteutuksessa käytetään malleja SFace ja Yunet, koska malleilla voidaan saavuttaa hyvä tarkkuus pienillä resursseilla.



Kuva 2. Kasvojen löytämismallit



Kuva 3. Kasvojentunnistusmallien nopeudet - suorituskykyoptimoitu

Paremmen tarkkuuden saavuttaakseen kasvojentunnistusmalliksi kannattaa valita Facenet512. Facenet512 on deepface malleista kaikkein tarkin, eikä se häviä nopeudessa muille ihmisen kasvojentunnistuksen tason saavuttaville malleille nopeustestin perusteella. Käyttämällä malleja Facenet512 ja YuNet voidaan saavuttaa tarkkuus 97,9%. [61]

4.3.2. Suorituskyvyn parannuksia

Toteutuksen alkuvaiheessa käytettiin deepfacen sisäistä find-funktiota. Tämä funktio suorittaa reaaliaikaisen kasvojentunnistuksen kannalta ylimääräisiä toimenpiteitä, johon menee merkittävä osa suoritusajasta. Tarpeettomat operaatioiden löytämiseksi find-funktion sisältä käytettiin Python cProfiler työkalua. cProfilerin avulla mitattiin, mitkä taustatoimenpiteet deepfacen find-funktiossa olivat aikaavieviä operaatioita.

CProfiler tuloksista, jotka on esitetty kuvassa 4, nähdään, että 44% deepfacen find-funktion käyttämästä ajasta kuluu kuvatietokantaan liittyviin toimintoihin, kuten kuvien läpikäyntiin ja kuvatietokannasta tietokannasta laskettujen vektoriesitysten lataamiseen, käsittelyyn ja tiedostojen hajautusesityksen laskemiseen. 19% ajasta menee SFace-kasvojentunnistusmallin luomiseen, mitä ei jokaisen kuvan kohdalla tarvitse tehdä. Tuloksessa huomattavan paljon aikaa vie myös deepfacen toteutus kuvan vektoriesityksen vertailuun tietokannan kuvien vektoriesityksiin. Deepface käyttää vertailuun kosinietäisyyttä. 72 prosenttia find-funktion ajasta menee hukkaan, koska siinä tehdään ylimääräisiä toimenpiteitä.

Deepface find yunet Sface - cProfiler tulokset



Kuva 4. cprofiler tulokset deepfacen find-funktiolle - yunet - SFace

Nämä tulokset mielessä rakennettiin suorituskyvyltään nopeampaa toteutusta. DeepFace.extract_faces -funktiota käytetään kasvojen löytämiseen, ja DeepFace.represent -funktiota käytetään kasvojen vektoriesityksen laskemiseen. Yksinkertaisella toteutuksella, jossa mukana oli kasvojen löytäminen ja

kasvojentunnistaminen, mutta ei kasvojen tallentamista muistiin, keskimääräiseksi kasvojentunnistuksen kuvataajuudeksi saatiin 17,94 fps. Kasvojentunnistus ajettiin Intelin i7 4770 -prosessorilla.

4.4. Kuinka kasvojen tallentaminen ja muisti toteutettiin?

Toteutettavan kasvojentunnistusjärjestelmän ideana on rakentaa järjestelmä, joka muistaa onko robotin kanssa interaktiossa oleva henkilö robotille tuttu. Kasvojentunnistusjärjestelmän muisti on siis messujen alussa tyhjä, ja messujen kuluessa kasvojentunnistusjärjestelmä tallentaa muistiin kaikki henkilöt, jotka ovat olleet robotin kanssa interaktiossa. Aluksi muistaminen perustui representaatiolistan tallentamiseen väliaikaiseen muistiin. Kun kasvot olivat riittävän kauan tunnistettavana, järjestelmä tallensi kasvoista yhden muuttumattoman representaation. Ongelmaksi muodostui järjestelmän alttius virheille, koska representaatiota ei päivitetty ja tallennettu representaatio oli satunnainen. Toisin sanoen, jos ihminen esimerkiksi käänsi kasvonsa hetkeksi sivulle, järjestelmä loi uuden representaation eli luuli kasvoja uusiksi. Tämän ongelman ratkaiseminen vaati klusterointialgoritmia, jonka avulla samankaltaiset kasvokuvat liitetään samaan henkilöön. Klusterointialgoritmia käyttämällä kasvojen muistaminen saadaan tarkemmaksi.

4.4.1. Kasvojentunnistuksen tulosten suodattaminen

Yksittäisen kuvan tapauksessa kasvojentunnistuksen tulos ei aina ollut haluttu. Kuvassa oleva henkilö voitiin esimerkiksi tunnistaa virheellisesti muutaman kuvan ajan, tai henkilö voi hetkellisesti ajautua kamerakuvan ulkopuolelle tai kääntyä siten, että kuvasta ei löydy kasvoja. Kasvojentunnistusmoduulin oli siis suodatettava tällaisia virheellisiä tuloksia pois.

Kasvojentunnistuksen suodattamista lähdettiin rakentamaan sillä idealla, että todellinen kasvojentunnistuksen tulos saadaan vasta useammasta yksittäisestä tuloksesta katsomalla, mikä kasvojentunnistuksen tulos saadaan useimmiten aikaiseksi.

Rakennettiin järjestelmä, joka pyrkii tunnistamaan, että ovatko kuvassa näkyvät kasvot samat, kuin edellisessä kuvassa näkyvät kasvot. Kasvot määritellään samaksi kasvoksi, jos kasvojen välisten vektorien etäisyys on tarpeeksi pieni, tai jos kasvot eivät ole liikkuneet merkittävästi edellisen kuvan kasvoihin verrattaessa. Kasvojen liikkumisen tarkistus on tarpeen, koska kasvojentunnistusmallit eivät aina kykene tunnistamaan samaa naamaa, jos pää kääntyy sivulle. Kasvoille laskettiin identiteetti-arvo niin, että sadasta tuloksesta otettiin yleisin arvo.

Tämän saavuttaakseen rakennettiin kasvojentunnistukselle lyhytaikainen muisti, joka muistaa edellisessä kuvassa olleet kasvot ja niiden tiedot. Kasvojentunnistusmoduuli tallensi löydettyjen kasvojen tiedot listaan. Näitä tietoja olivat kasvojen sijainti kuvassa, viimeisin kasvojen vektoriesitys ja kasvojen identiteetti.

Tällainen muuttumattomaan representaation perustuva malli ei ollut paras mahdollinen vaihtoehto. Tutkimalla asiaa lisää, tultiin johtopäätökseen, että messuesittelyrobotti tarvitsee klusterointialgoritmin kasvojen luokitteluun eri identiteetteihin.

4.4.2. Klusterointialgoritmin valinta

Käytettävän klusterointialgoritmin tulee pystyä luokittelemaan tuntemattomat kasvot siten, että yhdessä klusterissa olisi vain yhden henkilön kasvoja. Messuesittelyrobotti kerää jatkuvasti lisää dataa uusista kasvoista messutilanteessa, eikä klustereiden määrä ole tiedossa etukäteen. Tämän takia käytössä pitää olla online-klusterointialgoritmi.

Klusterointialgoritmin luonti on vaikea ja työläs prosessi, ja uuden algoritmin rakentaminen ja testaaminen tyhjästä ei ole mielekästä kandidaatintyön puitteissa. Siksi toteutuksessa käytetään valmiista klusterointialgoritmeista. Kriteereiksi klusterointialgoritmeille asetettiin online-klusterointi ja klusterointialgoritmin aiempi käyttö kasvojen klusteroinnissa. Valinnaiseksi kriteeriksi asetettiin olemassa olevan toteutuksen löytyminen. Messuesittelyrobotille soveltuvia klusterointialgoritmeja löytyi viisi kappaletta, joista kahdesta löytyi olemassa oleva toteutus.

klusterointialgoritmi	koodi saatavilla
Video Face Clustering [62]	x[63]
Face Clustering Algorithm in Real-Time video[27]	-
online Mixture of Gaussians clustering[64]	-
Links [58]	x[65]
face clustering for movie analysis[66]	-

Koska kahdessa näistä klusterointialgoritmeista löytyi olemassa oleva toteutus, valinta keskittyi näihin algoritmeihin. Messuesittelyrobotin kasvojentunnistusjärjestelmän klusterointialgoritmiksi valikoitui Links-klusterointialgoritmi, koska sen toteutus oli helpommin ymmärrettävä, kuin Video Face Clustering -algoritmin. Koodin ymmärrettävyys oli tärkeä kriteeri algoritmin valinnassa, koska algoritmin toteutusta pitää muokata messuesittelyrobottiin soveltuvaksi.

Links on online-klusterointialgoritmi, joka pyrkii arvioimaan jokaisen klusterin todennäköisyysjakamaa siihen ennestään lisättyjen vektorien perusteella. Links-klusterointi koostuu klustereista ja aliklustereista. Klusteri sisältää n kappaletta aliklustereita, joihin vektorit lisätään. Aliklusterille lasketaan vektoreista keskipiste, ja lisäksi aliklusterit voivat olla linkitettyinä toisiinsa. Klusteroinnissa on kolme parametria, jotka tulee määrittää testauksen avulla: `cluster_similarity_threshold`, `subcluster_similarity_threshold` ja `pair_similarity_maximum`. Huomattavaa on myös se, että nämä parametrit tulee määrittää erikseen jokaiselle käytetylle kasvojentunnistusmallille.

Kun uusi vektori lisätään klusterointiin, käytetään matemaattisia funktioita ja näitä parametreja sen määrittämiseen, miten toimitaan. Vektori voidaan lisätä joko olemassa olevaan aliklusteriin tai uuteen aliklusteriin. Jos vektori lisätään uuteen aliklusteriin, tämä aliklusteri voidaan lisätä olemassa olevaan klusteriin tai uuteen

klusteriin. Klusterin päivittyessä tarkistetaan, kuuluuko sen aliklusterit vielä klusteriin, ja aliklusterin päivittyessä tarkistetaan sen linkit toisiin aliklustereihin. Klusterointi päivittyy siten jokaisella kerralla, kun uusi vektori lisätään [58]. Messuesittelyrobotin kasvojentunnistusjärjestelmässä klusteri tarkoittaa yhtä identiteettiä.

4.4.3. Links-klusterointialgoritmin sovitus kasvojentunnistusjärjestelmään

Klusterointitoteutus tulee muokata käyttämään SQL-tietokantaa. Lisäksi kasvoista täytyy tallentaa aikaleimat, milloin kasvot on nähty, jotta tiedetään, milloin messuesittelyrobotti on nähnyt kyseisen henkilön. Tässä kandidaatintyössä klusteroinnin SQL-toteutusta ei saatu valmiiksi, sillä työhön käytettävissä oleva aika kävi vähiin.

Links-klusteroinnin toteutus on tehty pythonilla käyttäen pythonin luokkia [65]. Toteutus toimi sellaisenaan, mutta siinä on kaksi puutetta. Klusterointi tallennetaan vain keskusmuistiin, eikä vektorien lisäämisestä tallenneta minkäänlaisia aikaleimoja klusterointiin. Aikaleimat ovat tarpeellisia, jotta messuesittelyrobotti tietää, onko kuvassa näkyvä henkilö tavattu aiemmin, vai onko kasvot olleet kauan kuvassa. Aikaleimat tulee tallentaa aliklusterin arvoiksi, koska Links-klusteroinnissa klusterit ovat dynaamisia ja aliklusterit voivat siirtyä pois klusteroinnin sisältä uuteen klusteriin. Klusteriin kuuluvien aliklustereiden aikatieto on kuitenkin pystyttävä yhdistämään.

Yksinkertaisimmillaan aikatiedon tallentamisen voisi toteuttaa lisäämällä aikaleima jokaiselle vektorille, joka aliklusteriin lisätään. Tämä toteutus lisäisi kuitenkin lyhyellä aikavälillä valtavan määrän aikaleimoja listaan, eikä olisi siksi järkevä.

Datan määrän vähentämiseksi päätettiin arvioida alku- ja loppuajat, milloin kasvot ovat olleet näkyvissä kuvassa. Tätä varten tarvitaan raja-arvo sille, milloin kasvot määritellään olevan yhtäjaksoisesti kameran kuvassa. Kutsutaan tätä keskustelun raja-arvoksi. Myöskään todella lyhyitä keskusteluja ei välttämättä kannata tallentaa. Siksi keskustelulle määriteltiin minimipituus. Keskustelujen alku- ja loppuajaleimat lisätään aliklusterin arvoiksi.

Kun klusterointi on suoritettu vektorille, yhdistetään klusteroinnin tuloksena saadun klusterin aliklustereiden aikatiedot. Tämä tieto voidaan lähettää korkeamman tason moduuleille, jotka ohjaavat robotin toimintaa.

4.4.4. ROS-viestit

Kasvojentunnistusjärjestelmän ROS-viestejä muokattiin toteutuksessa. Olemassa oleva kasvojentunnistusjärjestelmä lähetti kolmea ROS2-viestiä muille moduuleille:

1. Reaaliaikaista videokuvaa järjestelmän toiminnasta.
2. Kamerassa näkyvän suurimman puhuvan naaman sijainnin. Tätä tietoa käytettiin robotin pään liikuttamiseen
3. Lista kaikkien kamerassa näkyvistä kasvoista. Olemassa olevassa toteutuksessa lista sisältää käytännössä tiedon vain kasvojen sijainneista.

Kasvojentunnistusjärjestelmän videokuvan lähettäminen päivitettiin siten, että kuvassa näkyy tunnistettujen kasvojen tunniste ja tietoja klusteroinnin klusterien

määristä. Suurimman kasvon sijainnin lähettämisestä luovuttiin, ja suurimman kasvon laskenta siirrettiin robotin pään liikkeistä vastaavalle moduulille. Tämän muutoksen taustalla on ajatus siitä, että kasvojentunnistusjärjestelmän ei tule valita, mihin robotti katsoo, vaan tämä on pään liikkeestä vastaavan moduulin tehtävä.

Kasvojen tietoja lisättiin viestiin, jossa lähetettiin kaikki kuvassa näkyvät kasvot. Tietoihin lisättiin kasvoja vastaavan klusterin tunniste ja lista tiedoista, milloin kasvot ovat olleet robotin näkyvissä. Lista sisältää seuraavat tiedot: alkuaika (start_time), loppuaika (end_time) ja kesto (duration).

4.5. Kuinka toteutusta testattiin?

Toteutusta testattiin Youtube faces -tietokannan avulla [67]. Testauksessa tietokannasta otettiin 33 kansiota, joista jokainen vastaa yhtä henkilöä. Jokaisesta henkilöstä on 1–4 videota pilkottuna kuviksi. Yhteensä kuvia oli 12818. Suurimmassa osassa videoita on vain yksi henkilö, kuitenkin joissain videoissa oli päähenkilön kasvojen lisäksi myös taustalla kasvoja. Tämän takia kaikki kansiot käytiin läpi, ja laskettiin, kuinka monta klusteria kansiosta tulisi muodostua. Lisäksi tietokantaan lisättiin yksi video, jossa ei ollut yhtään henkilöä. Testauksessa käytetyssä tietokannassa oli lopulta kasvoja yhteensä 43 henkilöltä.

Toteutus muistaa henkilöt tallentamalla niistä vektorirepresentaatioita ja yhdistämällä saman henkilön representaatiot klusteriksi. Eli jokaista henkilöä kohden pitäisi tulla yksi klusteri, johon tallentuu vektoreita kasvoista mahdollisista eri suunnista. Klusterointialgoritmille voi antaa kolme parametria: cluster_similarity_threshold, subcluster_similarity_threshold ja pair_similarity_maximum, jotka vaikuttavat klusterien ja alaklusterien luomisherkkyyteen. Aluksi kokeiltiin arvoilla: cluster_similarity_threshold: 0.3, subcluster_similarity_threshold: 0.2 ja pair_similarity_maximum: 1.0. Näillä arvoilla klustereita muodostui 52 eli 9 enemmän kuin naamoja on videoissa. Toteutus muodosti osalle henkilöistä useamman klusterin, koska heidän ulkonäöllensä oli eri videoissa riittävän paljon eroa (ikäntyneet videoiden välissä, valaistus, kuvakulma, etäisyys). Klusteroinnin herkkyyttä täytyi siis säätää.

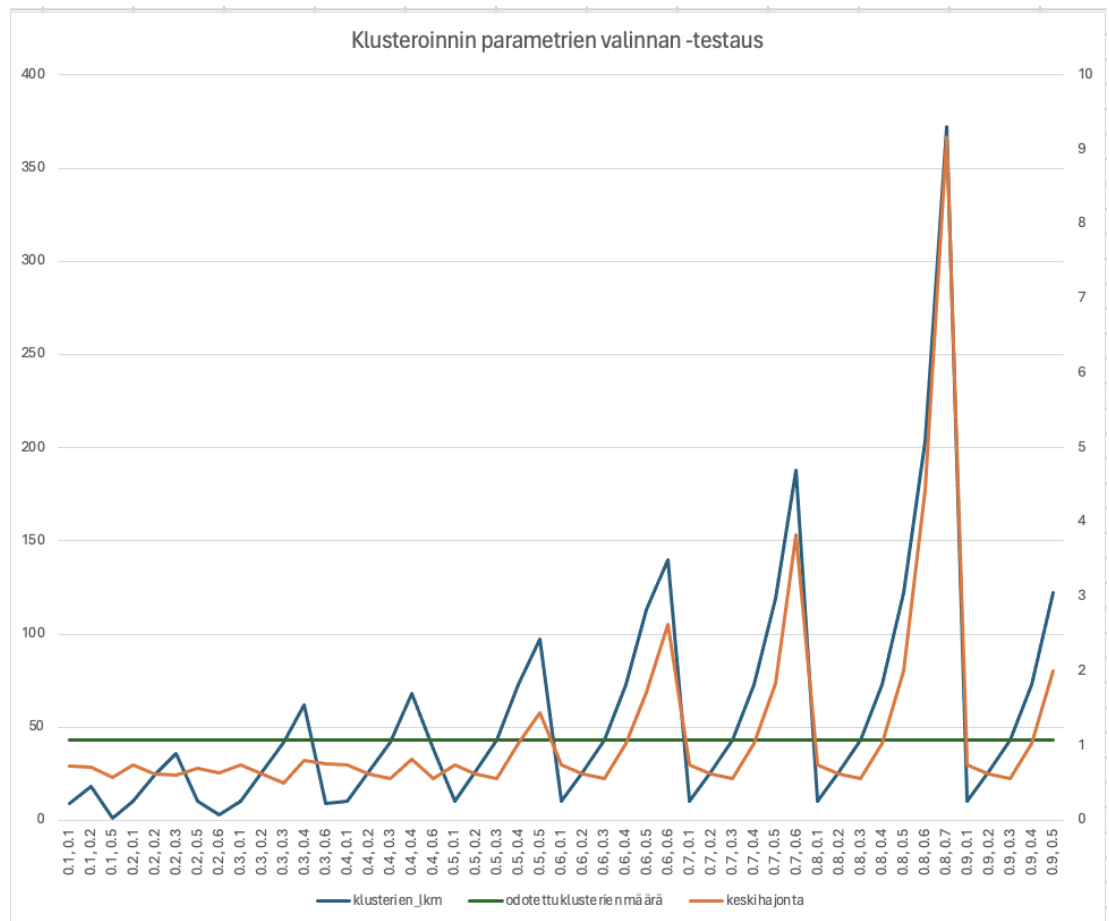
Seuraavaksi ajettiin samaa testiä silmukassa 100 kertaa, niin että kokeiltiin parametrejä cluster_similarity_threshold ja subcluster_similarity_threshold kaikilla arvoilla 0,1–0,9, jotta löydettäisiin optimaalisimmat parametrit klusterointialgoritmille. Kolmas parametri: pair_similarity_maximum, pidettiin koko ajan arvossa 1. Testauksessa käytiin läpi jokainen tietokannan kansio, ja jokaisen kansion kohdalla laskettiin klusterointiin lisättyjen klusterien määrä. Tätä verrattiin odotettuun klusterien määrän lisäykseen, ja jokaisesta kierroksesta laskettiin tämän eroavaisuuden keskihajonta. Laskettiin myös kansioiden määrä, joissa virheellistä tulosta esiintyi. Lisäksi klusteroinnista laskettiin klusterien kokonaismäärä.

Huomattavaa testauksessa on, että osalla arvopareista klusterointitoteutus antaa virheen, eikä klusterointitoteutus toimi. Yleensä virhe esiintyy, kun subcluster_similarity_threshold on suurempi, kuin cluster_similarity_threshold. Testauksen tuloksissa nämä virhetilanteet on jätetty merkkeamatta.

Kuvassa 5 näkyy laskettu keskihajonta, klusterien kokonaismäärä sekä odotettu klusterien määrä. Selkeästi parhaita arvoja tuloksista ei voi päätellä, koska kaikilla

arvoilla on enemmän tai vähemmän virhettä. Klusterointi ei siis toimi aivan täydellisesti. Kuitenkin parhaiten klusterointi toimi esimerkiksi arvoilla (0.3,0.3) tai (0.4,0.3), joilla on melko vähän virhettä.

Testauksen tuloksien perusteella näyttää, että `subcluster_similarity_threshold` arvon valinnalla on huomattavasti suurempi merkitys, kuin `cluster_similarity_threshold` arvon valinnalla. Klusterien kokonaismäärä näyttää kasvavan aina kyseisen parametrin kasvaessa. `Cluster_similarity_threshold` arvon vaikutus klusteroinnin tulokseen on pieni tai joissain tapauksissa sillä ei ole vaikutusta. Tämän syynä voi olla pieni testidatan määrä, jossa yhdestä henkilöstä otetut videot voivat olla todella samankaltaisia.



Kuva 5. Klusteroinnin tuloksia eri parametreilla

4.6. Pohdinta ja jatkokehitys

Toteutuksessa luotiin messuesittelyrobotille kasvojentunnistusjärjestelmä, joka tunnistaa, luokittelee ja tallentaa kasvoja muistiin. Kasvojentunnistusjärjestelmä pystyy yleensä tunnistamaan aiemmin robotin kanssa keskustelua käyneen henkilön.

Kasvojentunnistusjärjestelmä ei aina kykene kuitenkin yhdistämään samoja kasvoja, kun kasvot ovat todella erilaisissa asennoissa. Toteutus on muodostettu siten, että kasvojentunnistamiseen ja löytämiseen käytettyjä malleja on helppo vaihtaa, mikä

lisää toteutuksen käyttökelpoisuutta messuesittelyrobotin jatkoa ajatellen. Toteutuksen tarkkuutta voisi lisätä käyttämällä Facenet512 mallia SFace mallin sijaan kasvojen tunnistamiseen. Tämä mallin vaihto olisi mielekäs, jos käytettävän tietokoneen suorituskyky riittää mallin ajamiseen.

Vaikka kasvojentunnistusjärjestelmä ei kaikissa tilanteissa välttämättä pysty toimimaan täydellisesti, työssä luotu toteutus tuottaa ylemmän tason moduuleille tarvittavan tiedon siitä, että kommunikoiko robotti tutun vai tuntemattoman henkilön kanssa, ja milloin kyseinen henkilö on tavattu. Seuraava vaihe messuesittelyrobotin kehitystyössä voisi olla päivitetyn kasvojentunnistusjärjestelmän käyttöönotto robotin toimintaa ohjaavissa moduuleissa. Robotin käyttäytymistä ja puhetta voitaisiin muokata sen mukaan, onko robotin kanssa keskusteleva henkilö tuttu vai tuntematon. Esimerkiksi henkilökohtaisten tervehdysten tai käyttäytymisreaktioiden käyttäminen tutun henkilön kanssa kommunikoitaessa voisi olla mielekästä. Näin robotista saadaan entistä vuorovaikutteisempi ja käyttäjälle personoidumpi, millä voidaan saavuttaa miellyttävä vuorovaikutuskokemus messuympäristössä.

Työn toteutuksen myötä messuesittelyrobotti muistaa milloin kukakin henkilö on ollut robotin näkyvillä. Mielekästä olisi kuitenkin muistaa myös muita asioita tavatusta henkilöstä. Robotin muistia voitaisiin laajentaa esimerkiksi tallentamalla henkilöiden kanssa käytyjen keskustelujen aiheita ja henkilöiden nimiä. Robotti voisi tervehtiä tuttua henkilöä nimellä uudelleen tavattaessa ja viitata aikaisempiin keskusteluihin. Tämä vaatisi jonkinlaisen yhdistetyn muistin luomisen puheentunnistukselle ja kasvojentunnistukselle. Esimerkiksi episodinen muistirakenne voisi olla yksi vaihtoehto (ks. luku 2.7.5).

Vuorovaikutukseen tulisi enemmän syvyyttä, jos robotilla olisi käytettävissä myös tietoja henkilön ilmeistä ja tunteista. Tämän voisi tehdä integroimalla Emotion_detection-moduulin nykyiseen toteutukseen tai käyttämällä deepfacen kasvojen analysointimalleja. Deepface sisältää mallit iän, sukupuolen, ilmeet ja etnisyyden arviointiin. [68]

Kasvojentunnistusjärjestelmä toimii sitä hitaammin mitä enemmän kasvoja on yhtä aikaa kuvassa, eikä toteutuksessa ole tällä hetkellä minkäänlaista suodatusta sille, kuinka monet kasvot käsitellään kuvasta. Messutilanteessa tällainen suodattaminen voisi kuitenkin olla tarpeen, sillä robotin taustalla saattaa olla näkyvissä useita kasvoja.

Kasvojentunnistusjärjestelmä on toteutettu yhdessä prosessissa. Tällä toteutuksella kasvojen seuraaminen on vain niin nopeaa, kuin kasvojen tunnistaminen. Ongelman voisi ratkaista jakamalla kasvojentunnistusjärjestelmä kahteen prosessiin siten, että kasvojentunnistaminen tehtäisiin eri prosessissa, kuin kasvojen seuraaminen ja tunnistaminen. Tällä tavoin kasvojentunnistaminen voitaisiin suorittaa esimerkiksi yhdessä viidestä kuvasta, tai niin nopeasti, kuin järjestelmä siihen kykenee, kasvojen seuraamisen siitä kärsimättä.

Kasvojentunnistusjärjestelmän testaaminen jäi suppeaksi. Haasteita tuotti erityisesti testaukseen soveltuvan videokuvan löytäminen ja tarve manuaaliselle materiaalin läpikäynnille. Kasvojentunnistusjärjestelmän testaukseen olisi hyvä saada videokuvaa messuympäristöstä esimerkiksi seuraavilta hakijanpäivien messuilta.

Toteutuksessa klusteroinnin SQL-toteutus jäi kesken ajan puutteen takia. Klusterointia käyttöönotettaessa tarkoituksena oli muuntaa klusterointitoteutus käyttämään SQL-tietokantaa. Muunnoksen taustalla on tarve tallentaa klusterointi myös pysyvään muistiin. Tämän muunnoksen tekemisessä tuli vastaan kuitenkin

lukuisia ongelmia, joiden ratkaisemiseen tässä työssä ei ollut aikaa. Moniulotteinen klusterointi voitaisiin tallentaa SQL tietokantaan käyttäen JSON1-lisäosaa.

Ongelmaksi SQL-toteutuksessa muodostui se, että kasvojentunnistusjärjestelmään integroitu links-klusteroinnin toteutus on tehty käyttäen Python-luokkia ja aliklustereiden linkit on toteutettu käyttäen suoraa linkkiä python objektiin. Toteutus tulisi muokata siten, että jokaisella aliklusterilla olisi uniikki tunnus, jonka avulla aliklusterit tunnistettaisiin toisistaan. Tämä vaatisi melkein täydellisen muutoksen klusteroinnin toteutukselle, eikä aika riittänyt tämän muutoksen toteuttamiselle.

5. YHTEENVETO

Tulevaisuuden humanoidirobotit tarvitsevat kasvojentunnistusta, jotta ne voisivat toimia yhdessä ihmisten kanssa. Kasvojentunnistuksen avulla voidaan parantaa turvallisuutta, tehostaa prosesseja ja tarjota parempia palveluita ihmisille arjessa ja työelämässä.

Konenäöllä ja kasvojentunnistuksella on teknologian näkökulmasta suhteellisen lyhyt historia, sillä vasta aivan viime aikoina syväoppivat konvoluutioneuroverkot mahdollistavat tehokkaan kuvien ja videoiden analysoinnin. Kasvojentunnistuksen vaiheet ovat seuraavat: kasvojen löytäminen, kasvojen suuntaus, kasvojen tunnistaminen ja verifiointi. Jokainen vaihe on tärkeä kokonaisprosessin onnistumisen kannalta. Kasvojentunnistus ei yksistään riitä, vaan tallennettuja representaatioita tulee jatkokäsitellä ja dataa klusteroida, jotta robotin muistijärjestelmä toimisi oikein.

Luonnollisen vuorovaikutuksen edistäminen ja humanoidirobottien käyttäytyminen erilaisissa tilanteissa, ovat tärkeitä tutkimuskohteita, kun sosiaaliset robotit yleistyvät. Ihmisen pitkäaikainen suhde sosiaaliseen robottiin, keinotekoinen kognitio ja tunnealy, sekä antropomorfismi vaativat vielä paljon tutkimusta, ennen kuin sosiaaliset robotit voisivat elää luonnollisesti osana yhteiskuntaa.

Tekninen toteutus oli keskeinen osa tutkimustyötä, sillä se mahdollisti teorian soveltamisen käytäntöön ja loi konkreettista ymmärrystä tutkittavasta aiheesta. Käytännön toteutuksen avulla voitiin testata ja arvioida konenäön ja kasvojentunnistuksen toimivuutta todellisessa ympäristössä.

Teknisen toteutuksen osana tutkittiin algoritmien suorituskykyä ja tunnistusnopeutta. Kasvojen löytämis- ja tunnistusmallien välillä oli suuria eroja. Nopeimmaksi kasvojenlöytämismalliksi osoittautui DeepId, mutta sen epävarman toiminnan takia lopullisessa toteutuksessa käytetään SFace-mallia.

Lisäksi toteutuksen kautta voitiin kokeilla optimointistrategioita, jotka parantavat kasvojentunnistusjärjestelmän tehokkuutta ja luotettavuutta. Toteutuksessa käytetty Links-klusterointialgoritmi oli välttämätön kasvojentunnistuksen tehokkaan toiminnan kannalta.

Kokonaisuudessaan tämä tutkielma antaa laajan katsauksen konenäön, kasvojentunnistuksen ja ihmisen ja koneen välisen vuorovaikutuksen keskeisiin käsitteisiin ja teknologioihin.

6. PROJEKTIN KUVAUS

Kandidaatintyöprojekti alkoi 8.1. ja päättyi 8.5. Kandidaatintyötä tehtiin siis 4 kuukauden ajan. Ensimmäisten kahden kuukauden ajan kandidaatintyön eteneminen oli hidasta. Syynä hitaalle etenemiselle oli se, että työn aloittaminen koettiin vaikeaksi. Työn aihe itsessään on monimutkainen ja haastava, mutta tämän lisäksi alussa vaikeuksia tuotti muun muassa messuesittelyrobotin ohjelmistojen käyttöön ottaminen ja tutkimustyö. Lisäksi kaikilla työhön osallistuneilla oli muita kursseja tänä aikana.

Kandidaatintyön alkuvaiheessa keskityttiin kasvojentunnistukseen tutustumiseen ja messuesittelyrobotin olemassaolevan toteutuksen käyttöönottoon. Sitten tehtiin suurin osa kirjoitus -osuudesta ja toteutusta. Loppupuolella kandidiprojektia viimeisteltiin kirjoitus osuus, kirjoitettiin toteutuksesta ja testattiin toteutusta. Viimeisenä kirjoitettiin testauksen tuloksista.

Kandidaatintyöhön käytetty aika työhön osallistunutta henkilöä kohden näkyy alla olevassa taulukossa. Työhön käytetty aika ei jakautunut tasaisesti kandidintyön tekijöiden välillä.

	Matti Luukkonen	Rasmus Kurtti	Juho Hummasti
Luento	5.15	6.45	7.45
Seminaari	5.15	5.3	6.45
Tutkimus	17.45	16.45	9
Toteutus	121.15	7	26.3
Testaus	33.15	12.15	57
Kirjoitus	41.45	90	31
Tunnit yhteensä	224.3	138.15	138

7. VIITTEET

- [1] Matsuzaka Y. & Yashiro R. (2023) Ai-based computer vision techniques and expert systems. *AI* 4, ss. 289–302. URL: <https://www.mdpi.com/2673-2688/4/1/13>.
- [2] Hazim N., Al-Dabbagh S.S.M. & Esam Matti W. (2016) Face recognition: A literature review. *International Journal of Applied Information Systems* 11, ss. 21–31.
- [3] Adjabi I., Ouahabi A., Benzaoui A. & Taleb-Ahmed A. (2020) Past, present, and future of face recognition: A review. *Electronics* 9. URL: <https://www.mdpi.com/2079-9292/9/8/1188>.
- [4] Turk M. & Pentland A. (1991) Eigenfaces for Recognition. *Journal of Cognitive Neuroscience* 3, ss. 71–86. URL: <https://doi.org/10.1162/jocn.1991.3.1.71>.
- [5] Phillips P., Wechsler H., Huang J. & Rauss P.J. (1998) The feret database and evaluation procedure for face-recognition algorithms. *Image and Vision Computing* 16, ss. 295–306. URL: <https://www.sciencedirect.com/science/article/pii/S026288569700070X>.
- [6] Phillips P., Flynn P., Scruggs T., Bowyer K., Chang J., Hoffman K., Marques J., Min J. & Worek W. (2005) Overview of the face recognition grand challenge. *Teoksessa: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), nide 1, nide 1, ss. 947–954 vol. 1*. URL: <https://ieeexplore.ieee.org/abstract/document/1467368>.
- [7] Guo G. & Zhang N. (2019) A survey on deep learning based face recognition. *Computer Vision and Image Understanding* 189, s. 102805. URL: <https://www.sciencedirect.com/science/article/pii/S1077314219301183>.
- [8] Taigman Y., Yang M., Ranzato M. & Wolf L. (2014) Deepface: Closing the gap to human-level performance in face verification. *Teoksessa: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. URL: https://openaccess.thecvf.com/content_cvpr_2014/html/Taigman_DeepFace_Closing_the_2014_CVPR_paper.html.
- [9] Fan S., Li J., Zhang Y., Tian X., Wang Q., He X., Zhang C. & Huang W. (2020) On line detection of defective apples using computer vision system combined with deep learning methods. *Journal of Food Engineering* 286, s. 110102. URL: <https://www.sciencedirect.com/science/article/pii/S0260877420302004>.
- [10] Paneru S. & Jeelani I. (2021) Computer vision applications in construction: Current state, opportunities & challenges. *Automation in Construction* 132, s. 103940.

- [11] Parmar D.N. & Mehta B.B. (2014), Face recognition methods applications.
- [12] Huang T., Xiong Z. & Zhang Z. (2005) Face Recognition Applications, University of Illinois at Urbana-Champaign, Urbana, IL, 61801, USA, kappale Chapter 16. {huang, zxiong, z Zhang6}@ifp.uiuc.edu.
- [13] Burger W. & Burge M.J. (2022) Digital image processing: An algorithmic introduction. Springer Nature.
- [14] Xu S., Wang J., Wang X. & Shou W. (2019) Computer vision techniques in construction, operation and maintenance phases of civil assets: A critical review.
- [15] O'Mahony N., Campbell S., Carvalho A., Harapanahalli S., Hernandez G.V., Krpalkova L., Riordan D. & Walsh J. (2020) Deep learning vs. traditional computer vision. Teoksessa: Advances in Computer Vision: Proceedings of the 2019 Computer Vision Conference (CVC), Volume 1 1, Springer, ss. 128–144.
- [16] Voulodimos A., Doulamis N., Doulamis A. & Protopapadakis E. (2018) Deep learning for computer vision: A brief review. Computational intelligence and neuroscience 2018.
- [17] Guo M.H., Xu T.X., Liu J.J., Liu Z.N., Jiang P.T., Mu T.J., Zhang S.H., Martin R.R., Cheng M.M. & Hu S.M. (2022) Attention mechanisms in computer vision: A survey. Computational visual media 8, ss. 331–368.
- [18] Du H., Shi H., Zeng D., Zhang X.P. & Mei T. (2022) The elements of end-to-end deep face recognition: A survey of recent advances. ACM Comput. Surv. 54. URL: <https://doi.org/10.1145/3507902>.
- [19] Wang M. & Deng W. (2018) Deep face recognition: A survey. CoRR abs/1804.06655. URL: <http://arxiv.org/abs/1804.06655>.
- [20] Soo S. (2014) Object detection using haar-cascade classifier. Institute of Computer Science, University of Tartu 2, ss. 1–12.
- [21] Anggo M. & Arapu L. (2018) Face recognition using fisherface method. Teoksessa: Journal of Physics: Conference Series, nide 1028, IOP Publishing, nide 1028, s. 012119.
- [22] Boutros F., Huber M., Siebke P., Rieber T. & Damer N. (2022), Sface: Privacy-friendly and accurate face recognition using synthetic data. URL: <https://arxiv.org/abs/2206.10520>.
- [23] Deng J., Guo J., Yang J., Xue N., Kotsia I. & Zafeiriou S. (2022) Arcface: Additive angular margin loss for deep face recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence 44, s. 5962–5979. URL: <http://dx.doi.org/10.1109/TPAMI.2021.3087709>.
- [24] Schroff F., Kalenichenko D. & Philbin J. (2015) Facenet: A unified embedding for face recognition and clustering. Teoksessa: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE. URL: <http://dx.doi.org/10.1109/CVPR.2015.7298682>.

- [25] Baltrušaitis T., Robinson P. & Morency L.P. (2016) Openface: An open source facial behavior analysis toolkit. Teoksessa: 2016 IEEE Winter Conference on Applications of Computer Vision (WACV), ss. 1–10.
- [26] Kulshreshtha P. & Guha T. (2018) An online algorithm for constrained face clustering in videos. Teoksessa: 2018 25th IEEE International Conference on Image Processing (ICIP), ss. 2670–2674.
- [27] Cai Y. & Gan H. (2019) An online face clustering algorithm for face monitoring and retrieval in real-time videos. Teoksessa: 2019 IEEE Intl Conf on Parallel Distributed Processing with Applications, Big Data Cloud Computing, Sustainable Computing Communications, Social Computing Networking (ISPA/BDCloud/SocialCom/SustainCom).
- [28] Peller-Konrad F., Kartmann R., Dreher C.R., Meixner A., Reister F., Grotz M. & Asfour T. (2023) A memory system of a robot cognitive architecture and its implementation in armarx. *Robotics and Autonomous Systems* 164, s. 104415.
- [29] Ho W.C., Dautenhahn K., Lim M.Y., Vargas P.A., Aylett R. & Enz S. (2009) An initial memory model for virtual and robot companions supporting migration and long-term interaction. Teoksessa: RO-MAN 2009 - The 18th IEEE International Symposium on Robot and Human Interactive Communication, ss. 277–284.
- [30] Hwang C.S. (2015) Prospective of semiconductor memory devices: from memory system to materials. *Advanced Electronic Materials* 1, s. 1400056. URL: <https://onlinelibrary.wiley.com/doi/abs/10.1002/aelm.201400056>.
- [31] Doering M., Kanda T. & Ishiguro H. (2019) Neural-network-based memory for a social robot: Learning a memory model of human behavior from data. *J. Hum.-Robot Interact.* 8. URL: <https://doi.org/10.1145/3338810>.
- [32] Leite I., Martinho C. & Paiva A. (2013) Social robots for long-term interaction: a survey. *International Journal of Social Robotics* 5, ss. 291–308.
- [33] Dautenhahn K. (1995) Getting to know each other—artificial social intelligence for autonomous robots. *Robotics and Autonomous Systems* 16, ss. 333–356. URL: <https://www.sciencedirect.com/science/article/pii/0921889095000542>, moving the Frontiers between Robotics and Biology.
- [34] Bhin H., Lim Y. & Choi J. (2023) A robot memory management system for long-term interactions with human. Teoksessa: 2023 20th International Conference on Ubiquitous Robots (UR), ss. 532–535.
- [35] Bartneck C., Belpaeme T., Eyssel F., Kanda T., Keijsers M. & Šabanović S. (2020) *What Is Human–Robot Interaction?*, Cambridge University Press. s. 6–17.
- [36] Sheridan T.B. (2016) Human–robot interaction: Status and challenges. *Human Factors* 58, ss. 525–532. URL: <https://doi.org/10.1177/0018720816644364>, PMID: 27098262.

- [37] Valli A. (2008) The design of natural interaction. *Multimedia Tools Appl.* 38, ss. 295–305.
- [38] Ferland F., Létourneau D., Aumont A., Frémy J., Legault M.A., Lauria M. & Michaud F. (2013) Natural interaction design of a humanoid robot. *J. Hum.-Robot Interact.* 1, s. 118–134. URL: <https://doi.org/10.5898/JHRI.1.2.Ferland>.
- [39] Bethel C. & Murphy R. (2010) Review of human studies methods in hri and recommendations. *I. J. Social Robotics* 2, ss. 347–359.
- [40] Su H., Qi W., Chen J., Yang C., Sandoval J. & Laribi M.A. (2023) Recent advancements in multimodal human–robot interaction. *Frontiers in Neurorobotics* 17. URL: <https://www.frontiersin.org/articles/10.3389/fnbot.2023.1084000>.
- [41] Bonarini A. (2020) Communication in human-robot interaction. *Current Robotics Reports* 1.
- [42] Lee S.W. (2006) Automatic gesture recognition for intelligent human-robot interaction. *Teoksessa: 7th International Conference on Automatic Face and Gesture Recognition (FGR06)*, ss. 645–650.
- [43] De Santis A., Siciliano B., De Luca A. & Bicchi A. (2008) An atlas of physical human–robot interaction. *Mechanism and Machine Theory* 43, ss. 253–270. URL: <https://www.sciencedirect.com/science/article/pii/S0094114X07000547>.
- [44] Picard R., Vyzas E. & Healey J. (2001) Toward machine emotional intelligence: analysis of affective physiological state. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 23, ss. 1175–1191.
- [45] Rani P., Liu C., Sarkar N. & Vanman E. (2006) An empirical study of machine learning techniques for affect recognition in human–robot interaction. *Pattern Analysis and Applications* 9, ss. 58–69.
- [46] Sheridan T.B. (2020) A review of recent research in social robotics. *Current Opinion in Psychology* 36, ss. 7–12. URL: <https://www.sciencedirect.com/science/article/pii/S2352250X2030004X>, cyberpsychology.
- [47] Satake S., Kanda T., Glas D.F., Imai M., Ishiguro H. & Hagita N. (2009) How to approach humans? strategies for social robots to initiate interaction. *Teoksessa: Proceedings of the 4th ACM/IEEE International Conference on Human Robot Interaction, HRI '09, Association for Computing Machinery, New York, NY, USA*, s. 109–116. URL: <https://doi.org/10.1145/1514095.1514117>.
- [48] Mitsunaga N., Smith C., Kanda T., Ishiguro H. & Hagita N. (2008) Adapting robot behavior for human–robot interaction. *IEEE Transactions on Robotics* 24, ss. 911–916.

- [49] de Graaf M.M., Allouch S.B. & Klamer T. (2015) Sharing a life with harvey: Exploring the acceptance of and relationship-building with a social robot. *Computers in Human Behavior* 43, ss. 1–14. URL: <https://www.sciencedirect.com/science/article/pii/S0747563214005536>.
- [50] Fink J. (2012) Anthropomorphism and human likeness in the design of robots and human-robot interaction. Teoksessa: S.S. Ge, O. Khatib, J.J. Cabibihan, R. Simmons & M.A. Williams (toim.) *Social Robotics*, Springer Berlin Heidelberg, Berlin, Heidelberg, ss. 199–208.
- [51] Mori M., MacDorman K.F. & Kageki N. (2012) The uncanny valley [from the field]. *IEEE Robotics Automation Magazine* 19, ss. 98–100.
- [52] Walters M.L., Syrdal D.S., Dautenhahn K., Te Boekhorst R. & Koay K.L. (2008) Avoiding the uncanny valley: robot appearance, personality and consistency of behavior in an attention-seeking home scenario for a robot companion. *Autonomous Robots* 24, ss. 159–178.
- [53] Murphy R.R. (2019) *Introduction to AI robotics*. MIT press.
- [54] He Q., Feng L., Jiang G. & Xie P. (2022) Multimodal multitask neural network for motor imagery classification with eeg and fnirs signals. *IEEE Sensors Journal* 22, ss. 20695–20706.
- [55] Chowdhary K.R. (2020) *Natural Language Processing*, Springer India, New Delhi. ss. 603–649. URL: https://doi.org/10.1007/978-81-322-3972-7_19.
- [56] Karli U.B., Chen J.T., Antony V.N. & Huang C.M. (2024) Alchemist: Llm-aided end-user development of robot applications. Teoksessa: *Proceedings of the 2024 ACM/IEEE International Conference on Human-Robot Interaction, HRI '24*, Association for Computing Machinery, New York, NY, USA, s. 361–370. URL: <https://doi.org/10.1145/3610977.3634969>.
- [57] Serengil S.I. & Ozpinar A. (2020) Lightface: A hybrid deep face recognition framework. Teoksessa: *2020 Innovations in Intelligent Systems and Applications Conference (ASYU)*, IEEE, ss. 23–27. URL: <https://doi.org/10.1109/ASYU50717.2020.9259802>.
- [58] Mansfield P.A., Wang Q., Downey C., Wan L. & Moreno I.L. (2018), Links: A high-dimensional online clustering method.
- [59] Macenski S., Foote T., Gerkey B., Lalancette C. & Woodall W. (2022) Robot operating system 2: Design, architecture, and uses in the wild. *Science Robotics* 7, s. eabm6074. URL: <https://www.science.org/doi/abs/10.1126/scirobotics.abm6074>.
- [60] URL: https://github.com/clydemcqueen/opencv_cam.

- [61] Serengil S.I. & Ozpinar A. (2024) A benchmark of facial recognition pipelines and co-usability performances of modules. *Bilisim Teknolojileri Dergisi* 17, ss. 95–107. URL: <https://dergipark.org.tr/en/pub/gazibtd/issue/84331/1399077>.
- [62] Tapaswi M., Law M.T. & Fidler S. (2019), Video face clustering with unknown number of clusters.
- [63] Video face clustering - toteutus. URL: https://github.com/makarandtapaswi/BallClustering_ICCV2019.
- [64] Montero D., Aginako N., Sierra B. & Nieto M. (2022) Efficient large-scale face clustering using an online mixture of gaussians. *Engineering Applications of Artificial Intelligence* 114, s. 105079. URL: <https://www.sciencedirect.com/science/article/pii/S0952197622002330>.
- [65] Links clustering - toteutus. URL: https://github.com/QEDan/links_clustering.
- [66] Kulshreshtha P. & Guha T. (2020), Dynamic character graph via online face clustering for movie analysis.
- [67] Wolf L., Hassner T. & Maoz I. (2011) Face recognition in unconstrained videos with matched background similarity. *Teoksessa: CVPR 2011*, ss. 529–534.
- [68] Serengil S.I. & Ozpinar A. (2021) Hyperextended lightface: A facial attribute analysis framework. *Teoksessa: 2021 International Conference on Engineering and Emerging Technologies (ICEET), IEEE*, ss. 1–4. URL: <https://doi.org/10.1109/ICEET53442.2021.9659697>.