Elias Hussen

Neuroverkkojen optimointi evoluutioalgoritmin avulla eri kuvantunnistustehtävissä

Neuroverkkojen optimointi evoluutioalgoritmin avulla eri kuvantunnistustehtävissä

Tekniikan ja
innovaatiojohtamisen
akateeminen yksikkö
Diplomi-insinööri
Automaatio ja tietotekniikka

Vaasa 2025

VAASAN YLIOPISTO
Tekniikan ja innovaatiojohtamisen akateeminen yksikkö
Tekijä: Elias Hussen
Tutkielman nimi: Neuroverkkojen optimointi evoluutioalgoritmin avulla eri kuvantunnistustehtävissä: Neuroverkkojen optimointi evoluutioalgoritmin avulla eri kuvantunnistustehtävissä
Tutkinto: Diplomi-insinööri
Oppiaine: Automaatio ja tietotekniikka
Työn ohjaaja: Janne Koljonen ja Timo Mantere
Valmistumisvuosi: 2025 **Sivumäärä:** 62

TIIVISTELMÄ:
Tässä tutkielmassa tarkastellaan evoluutioalgoritmien käyttöä konvoluutioisten neuroverkkojen (CNN) optimoinnissa kuvien luokittelutehtäviä varten. Vaikka CNN:t ovatkin nousseet keskeiseksi osaksi tietokonenäköön liittyviä sovelluksia, kuten terveydenhuoltoa, itseajavia ajoneuvoja ja maataloutta, optimaalisten arkkitehtuurien suunnittelu on edelleen monimutkainen ja resursseja vaativa haaste. Tämän ongelman ratkaisemiseksi tutkimuksessa selvitetään, voiko luonnonvalinnan periaatteisiin perustuvat evoluutioalgoritmit ylittää manuaalisesti suunnitellut CNN-arkkitehtuurit kehittämällä verkkojen rakenteita automaattisesti mutaatioiden ja risteytysten avulla.

Tutkimuksessa sovelletaan evoluutio-optimointikehystä kahteen julkisesti saatavilla olevaan aineistoon: nisäkkäiden kuvien luokitteluaineistoon ja kasvojen ilmeiden tunnistamistaineistoon. Empiiriset tulokset osoittavat luokittelutarkkuuden parantuneen merkittävästi ensimmäisen evoluutio-optimointisukupolven jälkeen, noin 30 ja 20 prosenttiyksikköä.

Tulokset vahvistavat evoluutioalgoritmien potentiaalin neuroarkkitehtuurin etsinnässä, erityisesti tilanteissa, joissa manuaalinen suunnittelu on epäkäytännöllistä. Opinnäytetyö päättyy korostamalla tarvetta vankempiin mutaatiostrategioihin, adaptiivisiin valintamekanismeihin ja tulevaisuuden validointiin suuremmilla, monipuolisemmilla tietokannoilla skaalautuvuuden ja luotettavuuden parantamiseksi.

AVAINSANAT: konvoluutioneuroverkot, evoluutioalgoritmit, kuvien luokittelu, neuroverkkorakenteiden haku, syväoppimisen optimointi

Sisällys

1 Johdanto 8

2 Kirjallisuuskatsaus 10

3 Kuvantunnistus ja koneoppiminen 12

3.1 Kuvantunnistuksen perusteet 12

3.2 Kuvan esikäsittely ja ominaisuuksien poiminta 15

3.3 Koneoppimisen lähestymistavat kuvantunnistuksessa 18

3.4 Syväoppiminen ja konvolutiiviset neuroverkot (CNN) 22

4 Evoluutioalgoritmit 28

4.1 Evoluutioalgoritmien perusteet 28

4.2 Evolutiiviset neuroverkot 32

4.2.1 Evolutiivinen painon optimointi 33

4.2.2 Kehittyvät verkkotopologiat 34

4.2.3 Kehittyneet sovellukset ja tulevaisuuden näkymät 35

5 Menetelmät ja materiaalit 37

5.1 Tutkimuksen suunnittelu ja tavoitteet 37

5.2 Aineisto 40

5.2.1 Nisäkkäiden luokitteluaineisto 40

5.2.2 Kasvojen ilmeiden luokitteluaineisto 42

5.3 Tutkimuksessa käytetyt syväoppimisarkkitehtuurit 43

5.4 Verkkojen optimointiin tarkoitettu evoluutioalgoritmi 46

6 Tulokset 50

7 Johtopäätökset 58

Lähteet 60

Kuvat

Kuva 1. Esimerkkejä käsin kirjoitetuista numeroista, jotka on otettu yhdysvaltalaisista postinumeroista (Bishop, 2006, s. 2). 18

Kuva 2. Valvottu oppiminen (Nasteski, 2017, s. 5). 20

Kuva 3. Yleiskuvaus koneoppimisalgoritmeista. Koneoppiminen on tekoälyn osa-alue. Tämäkuvahavainnollistaa eri koneoppimisalgoritmien hierarkiaa, mukaan lukien valvotut ja valvomattomat sekä vahvistusoppimistekniikat. Valvotun oppimisen kaksi pääluokkaa ovat luokittelu ja regressio, jotka johtavat vastaavasti diskreetteihin/laadullisiin ja jatkuviin/laadullisiin tavoitteisiin (Rashidi, 2019, s.9). 22

Kuva 4. Kaavio, joka kuvaa osaa konvoluutiohermoverkosta ja jossa näkyy konvoluutioyksiköiden kerros, jota seuraa alinäytteenottoyksiköiden kerros. Tällaisia kerroksia voidaan käyttää useita peräkkäisiä pareja (Bishop, 2006, s. 268). 23

Kuva 5. Harva liitettävyys. Korostamme yhden tuloyksikön, x3, ja korostamme myös s:n lähtöyksiköt, joihin tämä yksikkö vaikuttaa. (Ylhäällä) Kun s muodostetaan konvoluutiolla, jonka ydin on leveydeltään 3, x vaikuttaa vain kolmeen ulostuloon. (Alhaalla) Kun s muodostetaan matriisikertoimella, kytkeytyvyys ei ole enää harva, joten x3 vaikuttaa kaikkiin ulostuloihin (Goodfellow, Bengio, & Courville, 2016, s. 336). 25

Kuva 6. Pooling alinäytteistyksen vulla. Tässä käytetään max-poolingia, jossa poolin leveys on kolme ja poolien välinen askel on kaksi. Tämä pienentää esityskokoa kaksinkertaiseksi, mikä vähentää seuraavan kerroksen laskennallista ja tilastollista taakkaa. Huomaa, että oikeanpuoleisimman pooling-alueen koko on pienempi, mutta se on otettava mukaan, jos emme halua jättää huomiotta joitakin havaintoyksiköitä (Goodfellow, Bengio, & Courville, 2016, s. 344). 26

Kuva 7. Keinotekoisen kromosomin 16-bittinen binäärijono (Negnevitsky, 2005, s. 221). 29

Kuva 8. Esimerkki rulettipyörävalinnasta (Goldberg, 1989; Negnevitsky, 2005, s. 225). 29

Kuva 9. GA-sykli (Negnevitsky, 2005, s. 227) 30

Kuva 10. Suorituskykykaaviot 100 sukupolven kuudelle kromosomille: a) huippufunktion paikallinen maksimiratkaisu ja b) globaali maksimiratkaisu (Negnevitsky, 2005, s. 230). 31

Kuva 11. Illustraatio BP-ANN-arkkitehtuurista (Back Propagation Artificial Neural Network) (Ruan, 2017, s. 4). 33

Kuva 12. Painojen koodaus kromosomissa. 34

Kuva 13. Verkon topologian suora koodaus (Negnevitsky, 2005, s. 288). 35

Kuva 14. Evoluutioalgoritmin optimointivuokaavio. 39

Kuva 15. Esimerkkikuvia aineiston eri nisäkkäistä. 41

Kuva 16. Esimerkki kuvia datan eri tunnetiloista. 42

Kuva 17. Ensimmäinen arkkitehtuuri. 44

Kuva 18. Toinen arkkitehtuuri. 45

Kuva 19. Esimerkki risteyttämisestä (Yang Xu & Yongjie Ma, 2023, s.7). 47

Kuva 20. Kasvojen ilmeiden datalla esivalmennetun aloituspopulaation verkon 1 tarkkuudet. 50

Kuva 21. Kasvojen ilmeiden datalla esivalmennetun aloituspopulaation verkon 2 tarkkuudet. 51

Kuva 22. Kasvojen ilmeiden datalla esivalmennetun aloituspopulaation verkon 3 tarkkuudet. 51

Kuva 23. Kasvojen ilmeiden datan evoluutioharjoittelusukupolven validointi tarkkuudet. 52

Kuva 24. Kasvojen ilmeiden datan evoluutioharjoittelusukupolven validointi tarkkuudet toistamiseen. 52

Kuva 25. Nisäkkäiden datalla esivalmennetun aloituspopulaation verkon 1 tarkkuudet. 53

Kuva 26. Nisäkkäiden datalla esivalmennetun aloituspopulaation verkon 2 tarkkuudet. 53

Kuva 27. Nisäkkäiden datalla esivalmennetun aloituspopulaation verkon 3 tarkkuudet. 54

Kuva 28. Nisäkkäät datan evoluutioharjoittelusukupolven validointi tarkkuudet. 54

Kuva 29. Nisäkkäät datan evoluutioharjoittelusukupolven validointi tarkkuudet toistamiseen. 55

Kuva 30. Esimerkki parhaasta arkkitehtuurista nisäkkäiden luokittelussa sukupolvessa 3 LeNet tyylillä (Koko kuva). 55

Kuva 31. Esimerkki parhaasta arkkitehtuurista nisäkkäiden luokittelussa sukupolvessa 3 LeNet tyylillä (Osa 1). 56

Kuva 32. Esimerkki parhaasta arkkitehtuurista nisäkkäiden luokittelussa sukupolvessa 3 LeNet tyylillä (Osa 2). 56

Taulukot

Taulukko 1. Kuvaus nisäkkäiden luokitteluaineistosta. 41

Taulukko 2. Kuvaus kasvojen ilmeiden luokitteluaineistosta. 42

Taulukko 3. Evoluutioalgoritmin parametrit. 48

Taulukko 4. Mutaatio-operaattori. 48

Taulukko 5. Risteytys-operaattori. 48

Lyhenteet

AI	Artifical Intelligence
ANN	Artificial Neural Network
CNN	Convolution neural network
EA	Evolution algorithm
ES	Evolutionary Strategy
GA	Genetic Algorithm
GP	Genetic Programming
OCR	Optical Character Recognition
PSO	Particle Swarm Optimization
SVM	Support Vector Machine
VGG	Visual Geometry Group

1 Johdanto

Viime vuosina tietokonenäkö on kehittynyt huomattavasti, mikä johtuu pääasiassa syväoppimisen edistymisestä. Konvoluutiohermoverkot (CNN) ovat tulleet kuvien luokittelun perustavanlaatuiseksi tekniikaksi, ja ne ovat saavuttaneet poikkeuksellisen suorituskyvyn esimerkiksi lääketieteellisessä diagnostiikassa, autonomisessa ajamisessa ja tarkkuusmaataloudessa (Rawat & Wang, 2017). Menestyksestä huolimatta optimaalisten CNN-arkkitehtuurien suunnittelu on edelleen monimutkainen tehtävä, joka riippuu usein asiantuntijoiden intuitiosta, kokeilusta ja erehdyksistä sekä merkittävistä laskentaresursseista (Elsken et al., 2019).

Näiden haasteiden voittamiseksi automatisoidut lähestymistavat neuroverkkoarkkitehtuurin suunnitteluun ovat saaneet suosiota. Niistä evoluutioalgoritmit ovat nousseet esiin lupaavana ratkaisuna, joka jäljittelee luonnonvalinnan periaatteita verkostorakenteiden kehittämisessä. Nämä algoritmit soveltavat toistuvasti valinta-, mutaatio- ja risteytysoperaatioita luodakseen CNN-malleja, jotka voivat ylittää manuaalisesti suunnitellut vastineensa (Xie & Yuille, 2017). Tämä evoluutiolähestymistapa tarjoaa skaalautuvan ja mukautuvan menetelmän syväoppimisarkkitehtuurien optimointiin, vähentäen inhimillistä puolueellisuutta ja suunnittelutyötä.

Tämä tutkimus keskittyy selvittämään, voivatko evoluutioalgoritmit parantaa CNN:n suorituskykyä kuvien luokittelutehtävissä verrattuna manuaalisesti suunniteltuihin arkkitehtuureihin.

Tämä opinnäytetyö on tarkoitettu koneoppimisen ja tietokonenäön alan tutkijoille ja ammattilaisille, erityisesti niille, jotka ovat kiinnostuneita syväoppimisarkkitehtuurien evoluutio-optimoinnista. Se voi myös tarjota hyödyllisiä näkemyksiä tekniikan opiskelijoille ja alan ammattilaisille, jotka haluavat ymmärtää, miten evoluutioalgoritmeja voidaan soveltaa CNN-optimointiin.

Tutkielma rakentuu seitsemästä pääluvusta. Luvussa 2 esitellään aiheeseen liittyvä kirjallisuuskatsaus, jossa tarkastellaan aiempia tutkimuksia ja keskeisiä teoreettisia lähtökohtia konvoluutioneuroverkkojen ja evoluutioalgoritmien osalta. Luvussa 3 käsitellään kuvantunnistusta ja koneoppimista sekä esitellään syväoppimisen ja konvoluutioneuroverkkojen toimintaperiaatteet. Luvussa 4 syvennytään evoluutioalgoritmien teoriaan ja niiden soveltamiseen neuroverkkojen optimoinnissa. Luvussa 5 kuvataan tutkimuksessa käytetyt menetelmät ja aineistot, mukaan lukien luokitteluaineistot, valitut verkkoarkkitehtuurit ja optimoitu evoluutioalgoritmi. Luvussa 6 esitellään tutkimuksen tulokset, ja lopuksi luvussa 7 esitetään johtopäätökset, joissa tarkastellaan tutkimuksen keskeisiä havaintoja, menetelmän soveltuvuutta sekä mahdollisia jatkotutkimusaiheita.

2 Kirjallisuuskatsaus

Kuvien tunnistaminen on kehittynyt keskeiseksi tehtäväksi tietokonenäköalalla, erityisesti integroimalla syväoppimismalleja, jotka parantavat digitaalisten kuvien luokittelua, havaitsemista ja segmentointia eri aloilla. Nuthakki, Buddiga ja Koganti (2024) esittävät tutkimuksessaan kattavan vertailevan arvioinnin useista syväoppimisarkkitehtuureista – erityisesti CNN:stä, MobileNetV2:sta ja DenseNetistä121 – binääristen kuvien luokittelutehtävissä, joihin kuuluu mansikan laadunarviointi.

Kirjoittajat korostavat kuvantunnistuksen kasvavaa merkitystä sellaisilla aloilla kuin elintarvikkeiden jalostus, maatalous, terveydenhuolto ja autonomiset järjestelmät. He osoittavat, miten kuvapohjaista luokittelua voidaan soveltaa monimutkaisissa ympäristöissä laadunvalvonnan varmistamiseksi, manuaalisen työn vähentämiseksi ja lajittelu- ja luokittelutehtävien tehokkuuden parantamiseksi. Esimerkiksi ehdotettu CNN-arkkitehtuuri saavutti 99,8 prosentin tarkkuuden poimittavien ja poimimattomien mansikoiden erottamisessa, mikä on huomattavasti parempi kuin MobileNetV2 (96,7 %) ja DenseNet121 (93,6 %) samassa tietokokonaisuudessa (Nuthakki et al., 2024, s. 5-7).

Heidän työnsä kirjallisuuskatsauksessa tutkitaan myös hybridimalleja ja ominaisuuksien fuusiointistrategioita, joissa yhdistetään perinteisiä koneoppimisen luokittelijoita syviin neuroverkkoihin perustuviin ominaisuuksien poimijoihin. Esimerkkinä on CNN-SVM-hybridi, joka tuotti 97,5 prosentin tarkkuuden binäärisessä tomaattiluokittelussa ja 96,7 prosentin tarkkuuden moniluokkaisissa tehtävissä käyttäen InceptionV3:a ominaisuuksien poimijana (s. 2). Tällaiset lähestymistavat havainnollistavat, miten siirto-oppiminen ja valmiiksi koulutetut mallit voivat parantaa luokittelusuoritusta jopa rajallisilla harjoitusaineistoilla.

Lisäksi artikkelissa tarkastellaan ResNet50:n ja VGG16:n kaltaisten konvoluutioarkkitehtuurien soveltamista, jotka yhdistettynä VGG16-ResNet-hybridimalliin saavuttivat 99,78 prosentin tarkkuuden kuivien elintarvikkeiden

luokittelussa (s. 1). Nämä tulokset korostavat arkkitehtuurien fuusion tehokkuutta tunnistuksen kestävyyden parantamisessa.

Tarkkuusmittareiden lisäksi kirjoittajat tarkastelevat myös optimointitekniikoita. Esimerkiksi sumeassa mallissa ananaksen kypsyyden luokittelussa käytettiin parvioptimointia (Particle Swarm Optimization, PSO), joka paransi tarkkuutta yli 93 prosenttiin (Nuthakki et al., 2024, s. 3). Nämä havainnot viittaavat siihen, että syväoppimisen ja metaheurististen algoritmien yhdistäminen voi tuottaa lupaavia tuloksia visuaalisissa luokittelutehtävissä, joihin liittyy biologista vaihtelua ja monimutkaisia tekstuureja.

Yhteenvetona tarkasteltu kirjallisuus osoittaa, että vaikka CNN:t ovat edelleen erittäin tehokkaita binäärisissä luokittelutehtävissä, suorituskykyä voidaan parantaa edelleen siirto-oppimisen, datan lisäämisen, hybridimallien fuusioinnin ja evolutiivisen optimoinnin tekniikoiden avulla (Sun, 2020). Nämä oivallukset muodostavat teoreettisen perustan evoluutioalgoritmien soveltamiselle tässä tutkielmassa, jonka tavoitteena on optimoida syviä neuroverkkoja kuvapohjaiseen päätöksentekoon rajoitetuissa reaalimaailman ympäristöissä.

3 Kuvantunnistus ja koneoppiminen

Tässä luvussa esitetään kuvantunnistusjärjestelmien teoreettinen ja tekninen tausta koneoppimisen näkökulmasta. Luvussa 3.1 aloitetaan yleiskatsauksella ohjattuun oppimiseen ja sen rooliin kuvien luokittelussa ja regressiotesteissä. Luvussa 3.2 esitellään konvoluutiohermoverkot (CNN), jotka ovat kuvapohjaisten mallien perustana oleva arkkitehtuuri. Kohdassa 3.3 tarkastellaan tarkemmin luokittelua ja regressiota valvotussa oppimisessa, kun taas kohdassa 3.4 tutkitaan CNN-verkkojen poolausta, konvoluutio-operaatioita ja rakenteellisia etuja, jotka tekevät niistä sopivia kuvatehtäviin, joissa on spatiaalisia riippuvuuksia.

3.1 Kuvantunnistuksen perusteet

Tietokonenäön ja digitaalisen kuvankäsittelyn alana kuvantunnistus alkaa siitä, mitä kuva on. Digitaalinen kuva määritellään kaksiulotteisena funktiona *f(x,y),* jossa *x* ja *y* edustavat paikkakoordinaatteja, ja funktion arvo missä tahansa koordinaattiparissa vastaa kuvan intensiteettiä tai harmaustasoa kyseisessä pisteessä (Gonzalez & Woods, 2018: 18-19). Kun *x*, *y* ja intensiteettiarvot ovat diskreettejä ja äärellisiä, kuvan sanotaan olevan digitaalinen. Digitaalisen kuvan yksittäisiä elementtejä kutsutaan pikseleiksi – lyhenne sanoista ”kuvaelementit” – jotka muodostavat pienimmät osoitteelliset yksiköt ruudukkomaisessa kuvamatriisissa. Kukin pikseli sisältää voimakkuutta (tai väriä, jos kyseessä on värillinen kuva) koskevia tietoja, jotka yhdessä muodostavat kuvassa olevien kohteiden visuaalisen esityksen.

Digitaalisella kuvankäsittelyllä tarkoitetaan digitaalisten tietokoneiden käyttöä digitaalisten kuvien käsittelyyn. Tähän prosessiin kuuluu monenlaisia tekniikoita, joiden tarkoituksena on parantaa, analysoida ja tulkita kuvia. Digitaalisen kuvankäsittelyjärjestelmän syötteenä on digitaalinen kuva, ja operaation tyypistä riippuen tuloksena voi olla parannettu kuva, joukko poimittuja piirteitä tai abstrahoitu tulkinta sisällöstä (Gonzalez & Woods, 2018: 18-19). Tämä kyky on erityisen arvokas,

koska toisin kuin ihmisen näköjärjestelmä, joka rajoittuu sähkömagneettisen säteilyn näkyvään spektriin, digitaaliset kuvantamisjärjestelmät voivat toimia laajalla taajuusalueella, mukaan lukien gammasäteet, röntgensäteet, ultravioletti, infrapuna ja radioaallot. Tämä mahdollistaa sovellukset niinkin erilaisilla aloilla kuin tähtitieteessä, lääketieteellisessä diagnostiikassa, teollisuustarkastuksissa ja kaukokartoituksessa.

Kuvantunnistuksen ymmärtäminen edellyttää sen sijoittamista digitaalisen kuvankäsittelyn laajempaan kokonaisuuteen. Gonzalezin ja Woodsin (2018) mukaan jatkumo matalan tason kuvankäsittelystä korkean tason tietokonenäkötehtäviin voidaan jakaa kolmeen päävaiheeseen: matalan tason, keskitason ja korkean tason käsittelyyn. Matalan tason käsittelyyn kuuluvat perustavanlaatuiset operaatiot, kuten kuvan kohinanpoisto, kontrastin säätö ja terävöittäminen. Näissä operaatioissa on kyse kuvan laadun parantamisesta, ja niille on ominaista, että niissä on sekä kuvan tulo että lähtö. Esimerkiksi terävöitysalgoritmi voi ottaa epäselvän kuvan ja tuottaa siitä parannetun version, jossa on selkeämmät reunat ja hienommat yksityiskohdat.

Keskitason käsittelyyn kuuluu monimutkaisempia tehtäviä, kuten kuvan segmentointi, reunojen havaitseminen, ominaisuuksien erottaminen ja kohteiden luokittelu. Nämä prosessit ottavat yleensä kuvan syötteenä ja palauttavat kuvasta poimittuja merkityksellisiä attribuutteja tai rakenteita. Tässä vaiheessa kuvaa ei enää käsitellä pelkkänä pikselikokoelmana vaan pikemminkin sellaisten kohteiden tai alueiden koostumuksena, joilla on tunnistettavia ominaisuuksia. Esimerkiksi lääketieteellisen kuvan segmentointi eri kudoksia tai elimiä edustaviin alueisiin on tyypillinen keskitason tehtävä. Tällaiset tekniikat muodostavat kuvantunnistusjärjestelmien selkärangan, sillä niiden tehtävänä on eristää kuvasta ne olennaiset osat, jotka on tunnistettava (Gonzalez & Woods, 2018: 18-19).

Korkean tason käsittely on korkean tason käsittelyä, jonka tavoitteena on jäljitellä ihmisen kaltaista visuaalisen syötteen tulkintaa. Korkean tason kuvankäsittely on usein päällekkäistä tekoälyn (AI) kanssa, jossa tavoitteena on poimia visuaalisesta datasta

semanttinen merkitys ja tehdä päätöksiä tämän tiedon perusteella. Tässä yhteydessä sellaiset tehtävät kuin kohtausten ymmärtäminen, toiminnan tunnistaminen ja autonominen navigointi kuuluvat korkean tason kuvatulkinnan piiriin (Gonzalez & Woods, 2018: 18-19).

Kuvankäsittelyn, kuva-analyysin ja tietokonenäön väliset erot eivät aina ole selkeästi määriteltyjä. Käytännössä nämä alueet ovat usein päällekkäisiä. Kuva-analyysi, josta joskus käytetään nimitystä kuvan ymmärtäminen, sijoittuu kuvankäsittelyn ja tietokonenäön väliin, ja se käsittelee ensisijaisesti määrällisen ja kuvailevan tiedon poimimista kuvista. Kuvien tunnistaminen, joka määritellään kuvassa olevien kohteiden tai kuvioiden tunnistamiseksi ja luokittelemiseksi, kuuluu tiukasti keskitason ja korkean tason tietojenkäsittelyn alaan. Esimerkiksi kasvojentunnistusjärjestelmissä prosessiin kuuluu kuvan alkuperäinen parantaminen (matala taso), kasvojen havaitseminen ja segmentointi (keskitaso) ja lopulta yksilön tunnistaminen (korkea taso) (Gonzalez & Woods, 2018: 18-19).

Konkreettinen esimerkki tästä jatkumosta voidaan nähdä optisissa merkintunnistusjärjestelmissä (OCR). Ensin järjestelmä ottaa kuvan asiakirjasta (kuvan hankinta), sitten se vähentää kohinaa ja säätää kontrastia (matalan tason käsittely), segmentoi asiakirjan yksittäisiin merkkeihin (keskitason käsittely) ja lopuksi luokittelee jokaisen merkin vastaaviksi kirjaimiksi tai symboleiksi (tunnistaminen). Kehittyneemmissä järjestelmissä järjestelmä voi jopa tulkita tekstin merkityksen tai analysoida tunteita (korkean tason käsittely). Kuten Gonzalez ja Woods (2018) korostavat, tällaiset reaalimaailman sovellukset havainnollistavat kuvankäsittelyn eri vaiheiden välisten rajojen hämärtymistä ja korostavat, että vankkojen kuvantunnistusjärjestelmien kannalta on välttämätöntä integroida useita käsittelytasoja.

Yhteenvetona voidaan todeta, että kuvantunnistus perustuu syvälle digitaalisen kuvankäsittelyn periaatteisiin. Prosessissa raakakuvatiedot muutetaan jäsennellyksi, merkitykselliseksi informaatioksi useiden käsittelyvaiheiden avulla (Chakraborty & Saha,

2023). Nämä vaiheet heijastavat etenemistä kuvan laadun parantamisesta yksittäisten komponenttien tunnistamiseen ja lopulta monimutkaisten kohtausten tulkintaan. Tarkkojen ja tehokkaiden kuvantunnistusjärjestelmien kehittämiseksi on välttämätöntä ymmärtää kuvan esittämisen perusteet, pikselipohjaiset operaatiot ja kuvankäsittelytehtävien hierarkkinen luonne.

3.2 Kuvan esikäsittely ja ominaisuuksien poiminta

Kuvien tunnistamiseen kuuluu paitsi visuaalisen tiedon tallentaminen myös sen valmistelu jatkoanalyysia varten sopivaan muotoon. Tässä kuvien esikäsittely ja ominaisuuksien louhinta ovat keskeisiä vaiheita ennen syväoppimismallien käyttöä (Sujatha & Selvathi, 2021; Basha et al., 2024). Esikäsittelyn tarkoituksena on parantaa raakakuvatietoa ja vähentää ei-toivottuja vaihteluita, kun taas ominaisuuksien louhinta keskittyy kuvan keskeisten piirteiden tunnistamiseen ja esittämiseen tiiviissä ja informatiivisessa muodossa.

Teknisestä näkökulmasta digitaalinen kuva koostuu kuvaelementtien eli pikseleiden joukosta, jotka tallentavat kirkkaus- tai väriarvoja alueellisessa ruudukossa. Kamerat ovat ensisijaisia välineitä tällaisten kuvien ottamiseen. Ne voivat vaihdella tavallisista optisista kameroista erikoistuneisiin kuvantamislaitteisiin, kuten magneettikuvauslaitteisiin, ultraääniantureihin ja kaukokartoituslaitteisiin, kuten synteettisen aukon tutkiin (Nixon & Aguado, 2020). Tuloksena saadut kuvat vaihtelevat suuresti resoluution, formaatin ja kohinatason suhteen, mikä edellyttää räätälöityjä esikäsittelytekniikoita ennen kuin luotettava analyysi voidaan aloittaa.

Esikäsittely sisältää tyypillisesti operaatioita, kuten kontrastin säätöä, kohinanpoistoa ja normalisointia, joiden kaikkien tarkoituksena on valmistella tiedot ominaisuuksien louhintaa varten. Esimerkiksi ultraäänikuvat voivat vaikuttaa kohinaisilta kudosten signaalihäiriöiden vuoksi, kun taas pehmytkudosten magneettikuvat voivat peittää alleen hienommat rakenteelliset yksityiskohdat. Soveltamalla esikäsittelytekniikoita,

kuten tasoitusta tai suodatusta, näiden kuvien laatua voidaan parantaa merkittävästi, jotta tärkeät rakenteet, kuten verisuonet tai kasvaimet, erottuisivat paremmin (Nixon & Aguado, 2020, s. 1-2).

Esikäsittelyvaiheeseen voi kuulua myös synteettisten kuvien tuottaminen, jolla on arvokas rooli visuaalisten algoritmien kestävyyden ja tehokkuuden arvioinnissa. Synteettisten kuvien, kuten matemaattisesti määriteltyjen muotojen tai tunnettujen tekstuurien koosteiden avulla tutkijat voivat arvioida, kuinka hyvin heidän järjestelmänsä havaitsevat reunat, erottavat tekstuurit tai segmentoivat alueita, joilla on tunnettuja ominaisuuksia. Nämä kuvat ovat ihanteellisia algoritmien kalibrointiin, koska niiden rakenne ja ominaisuudet ovat täysin kontrolloituja (Nixon & Aguado, 2020, s. 2).

Esikäsittelyn jälkeen käytetään ominaisuuksien louhintaa sellaisten kuvioiden tunnistamiseksi, jotka ovat sekä merkityksellisiä että laskennallisesti hyödyllisiä. Esimerkiksi ihmisen kasvojen tunnistaminen kamerakuvasta edellyttää tiettyjen muotojen, kuten nenän, silmien ja kulmakarvojen, eristämistä. Nämä osat toimivat mitattavina piirteinä, joiden avulla algoritmit voivat kuvata ja lopulta tunnistaa kasvot (Nixon & Aguado, 2020, s. 2). Samankaltaisia periaatteita sovelletaan lääketieteellisessä kuvantamisessa, jossa ominaisuuksien louhinnalla voidaan eristää anatomisia rakenteita, tai kaukokartoituksessa, jossa maanpeitteen tekstuurierot (esim. lehdet vs. tiet) ohjaavat tulkintaa.

Monissa reaalimaailman sovelluksissa ominaisuuksien louhinta ulottuu tekstuurianalyysiin, reunojen havaitsemiseen ja muodon mallintamiseen. Tietokonenäköjärjestelmät tukeutuvat tällaisiin piirteisiin pelkistääkseen monimutkaista visuaalista tietoa helpommin hallittavaan muotoon, jossa olennainen tieto säilyy. Hyvä esimerkki on Brodatzin tekstuurikuvien analysointi, joita on skannattu ja koottu tietokokonaisuuksiksi, jotta voidaan testata algoritmien kykyä tunnistaa ja luokitella alueita tekstuurikuvioiden perusteella (Nixon & Aguado, 2020, s. 2).

Kuvatiedot eivät myöskään aina ole peräisin reaalimaailman kohtauksista. Syntetisoidut kuvat ovat ratkaisevassa asemassa algoritmien testauksessa ja arvioinnissa, sillä ne tarjoavat kontrolloidut olosuhteet suorituskyvyn mittaamiseen. Esimerkiksi täydellisistä geometrisista muodoista tai kuvioiduista tekstuureista koostuva kuva mahdollistaa tarkan reunan havaitsemisen, muodon tunnistamisen ja segmentoinnin testaamisen. Näiden synteettisten kuvien rajat ovat tarkkoja, joten ne ovat ihanteellisia ominaisuuksien louhintamenetelmien herkkyyden ja tarkkuuden arvioimiseksi (Nixon & Aguado, 2020, s. 2-3).

Kuvalähteiden moninaisuus vaikuttaa myös esikäsittely- ja ominaisuuksien uuttamisstrategioihin. Vaikka kamerakuvat kaappaavat näkyvää valoa, monet lääketieteelliset tai teolliset järjestelmät työskentelevät spektrin ei-näkyvillä osilla, kuten infrapuna- tai ultraäänilähteillä. Tämä tarkoittaa, että esikäsittelymenetelmät on räätälöitävä paitsi kuvan sisällön myös modaliteettikohtaisten ominaisuuksien, kuten kohinaprofiilien ja resoluutiorajoitusten, mukaan (Nixon & Aguado, 2020, s. 2).

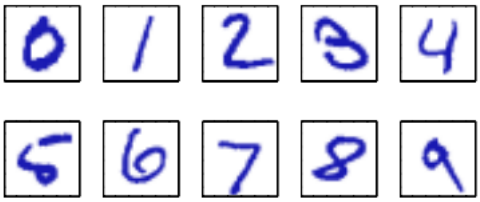
Ihmisen ja tietokonenäön välinen yhteys on myös informatiivinen ominaisuuksien louhinnan ymmärtämisessä. Vaikka molemmat järjestelmät pyrkivät tulkitsemaan paikkatietoa, ne toimivat varsin eri tavoin. Ihmisen näkö on erityisen hyvä tulkitsemaan suhteellisia etäisyyksiä ja tunnistamaan kuvioita myös huonossa valaistuksessa. Sen kyky tehdä tarkkoja mittauksia on kuitenkin rajallinen. Toisaalta tietokonenäköjärjestelmät voivat saavuttaa suuren tarkkuuden tiettyjen yksityiskohtien mittaamisessa ja tunnistamisessa, kun niitä tuetaan asianmukaisella esikäsittelyllä ja hyvin poimituilla piirteillä, vaikka niillä voi olla vaikeuksia kohtausten tulkinnassa, elleivät älykkäät algoritmit ohjaa niitä (Nixon & Aguado, 2020, s. 3).

Yhteenvetona voidaan todeta, että kuvien esikäsittely ja ominaisuuksien poiminta eivät ole vain teknisiä välttämättömyyksiä, vaan ne ovat nykyaikaisten kuvantunnistusjärjestelmien kriittisiä mahdollistajia. Olipa tehtävänä sitten kasvojen tunnistaminen, lääketieteellisten sairauksien diagnosointi tai ympäristön muutosten

analysointi, nämä vaiheet muuttavat raa'an, meluisan tiedon jäsennellyksi muodoksi, jota voidaan tulkita algoritmisesti. Näin ollen ne muodostavat minkä tahansa käytännön tietokonenäköputken selkärangan, sillä ne määrittävät sekä tunnistustulosten tarkkuuden että tehokkuuden.

3.3 Koneoppimisen lähestymistavat kuvantunnistuksessa

Koneellisesta oppimisesta on tullut keskeinen menetelmä nykyaikaisissa kuvantunnistusjärjestelmissä, ja sen avulla algoritmit voivat automaattisesti löytää kuvioita suurista tietokokonaisuuksista ja tehdä älykkäitä päätöksiä visuaalisen syötteen perusteella. Alalla, joka perustuu vuosikymmeniä jatkuneeseen tilastolliseen tutkimukseen ja laskennalliseen kehitykseen, keskitytään automaattiseen rakenteen tunnistamiseen datasta käyttäen algoritmeja, jotka yleistyvät hyvin aiemmin näkemättömiin syötteisiin (Bishop, 2006, s. 1-4). Tämä kyky yleistää — tulkita tarkasti uutta, ennennäkemätöntä tietoa — on tehokkaiden koneoppimisjärjestelmien tunnusomainen piirre.



Kuva 1. Esimerkkejä käsin kirjoitetuista numeroista, jotka on otettu yhdysvaltalaisista postinumeroista (Bishop, 2006, s. 2).

Kuvassa 1 klassinen esimerkki, jota käytetään havainnollistamaan koneoppimista kuvantunnistuksessa, on käsinkirjoitettujen numeroiden tunnistamisen ongelma, jossa jokainen numero (0-9) esitetään 28 × 28 pikselin kokoisena harmaasävykuvana. Nämä kuvat voidaan tasoittaa vektoreiksi, joissa on 784 numeerista arvoa, joista kukin koodaa pikselin voimakkuuden. Tavoitteena on luokitella kukin syöttövektori oikeaan numeroluokkaan. Vaikka tämä saattaa vaikuttaa suoraviivaiselta, käsialan tyylien, viivan paksuuden, kohdistuksen ja kohinan vaihtelu tekee siitä ei-triviaalin tehtävän. Yritys

ratkaista se manuaalisesti koodattujen sääntöjen tai heuristiikkojen avulla johtaa usein hauraisiin järjestelmiin, joita vaivaavat poikkeukset ja epäjohdonmukaisuudet (Bishop, 2006, s. 2).

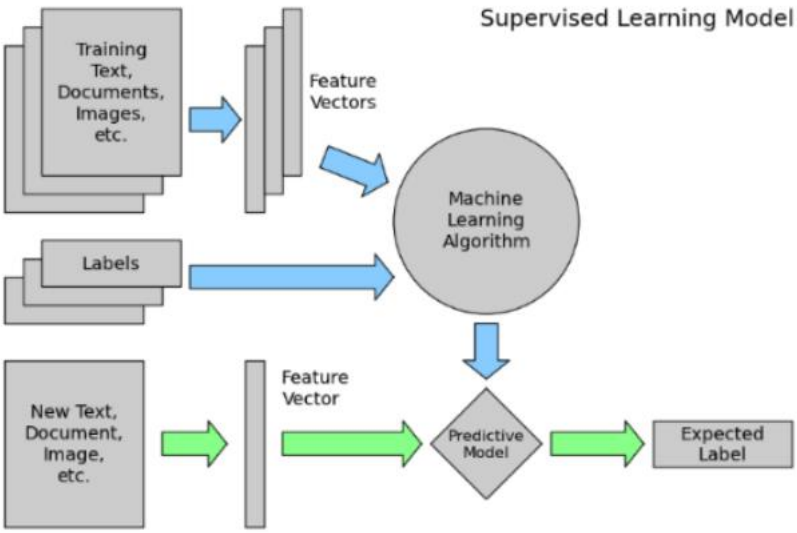
Sen sijaan valvottu oppiminen tarjoaa vankan ja tietoon perustuvan lähestymistavan ongelmaan. Valvotussa oppimisessa algoritmi koulutetaan merkityllä tietokokonaisuudella, joka koostuu tulo- ja lähtöpareista. Numeroiden tunnistustehtävässä tämä harjoitusjoukko koostuu suuresta määrästä kuvavektoreita x, jotka on yhdistetty tunnettuihin kohdelappuihin t, jotka osoittavat oikean numeron. Oppimisalgoritmi säätää ennakoivan mallin *y(x)* parametreja siten, että se pystyy harjoittelun jälkeen luokittelemaan uudet numerokuvat oikein määrittelemällä ne johonkin ennalta määritellyistä luokista (Bishop, 2006, s. 2). Tämä koulutusprosessi on keskeinen lähes kaikissa nykyaikaisissa kuvantunnistusjärjestelmissä, erityisesti neuroverkkoihin perustuvissa järjestelmissä.

Monien koneoppimisjärjestelmien keskeinen osa on esikäsittelyvaihe, jossa raakakuvadata muutetaan luokitteluun sopivammaksi esitykseksi. Esimerkiksi numeroiden tunnistuksessa numerot skaalataan ja keskitetään usein kiinteän kokoiseen rajaavaan laatikkoon, jotta merkityksettömät vaihtelut, kuten sijainti- tai kokoerot, voidaan minimoida. Tämä esikäsittelyvaihe — johon usein viitataan myös ominaisuuksien poistona — vähentää luokan sisäistä vaihtelua ja yksinkertaistaa luokittelutehtävää (Bishop, 2006, s. 2). Asianmukainen esikäsittely on olennaisen tärkeää sen varmistamiseksi, että sekä koulutus- että testidataa käsitellään johdonmukaisesti ja että tarpeeton monimutkaisuus poistetaan syötteestä.

Joissakin sovelluksissa, erityisesti niissä, jotka vaativat reaaliaikaista suorituskykyä (esim. kasvojen havaitseminen live-videossa), esikäsittelyä käytetään myös laskennallisen kuormituksen vähentämiseksi. Tunnettu esimerkki on Viola-Jones-algoritmi, joka käyttää yksinkertaisia mutta erittäin tehokkaita ominaisuuksia, kuten suorakulmaisten alueiden keskimääräistä intensiteettiä. Nämä piirteet on helppo laskea, mutta ne myös säilyttävät

keskeisen erottelevan tiedon, mikä mahdollistaa nopean ja tarkan luokittelun (Bishop, 2006, s. 3). Tällainen dimensioiden pienentäminen on käytännössä ratkaisevan tärkeää, sillä se yksinkertaistaa mallin tehtävää ja nopeuttaa käsittelyä ilman, että suorituskyky kärsii merkittävästi — edellyttäen, että olennaista tietoa ei menetetä muunnoksen aikana.

Monien kuvantunnistustehtävien perustana on valvottu oppiminen, jossa kullekin syötteelle annetaan tunnettu tuloste. Nämä tehtävät jakautuvat yleensä kahteen luokkaan: luokittelu ja regressio. Luokittelussa tavoitteena on luokitella syötteet kiinteään joukkoon erillisiä luokkia — esimerkiksi tunnistaa, sisältääkö kuva kissan, auton vai puun. Regressiossa taas voitaisiin ennustaa jatkuvia arvoja, kuten pään asennon kulmaa tai kohteen etäisyyttä kamerasta. Nasteskin (2017) mukaan luokittelu kuvaa syötetietoja ennalta määritettyihin luokkiin, kun taas regressio kuvaa syötteitä reaaliarvoisiksi tuloksiksi.



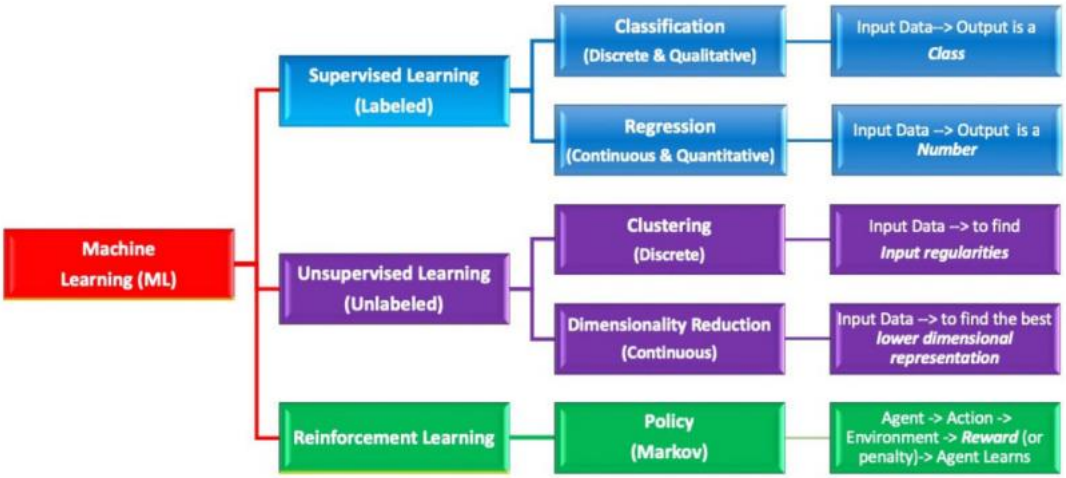
Kuva 2. Valvottu oppiminen (Nasteski, 2017, s. 5).

Kuten kuvassa 2 on esitetty, valvotun oppimisen työvaiheessa malli koulutetaan leimatuilla esimerkeillä (X, y) ja se oppii yleistämään tämän kartoituksen näkymättömiin tietoihin, mikä mahdollistaa vankan suorituskyvyn ennustustehtävissä testauksen aikana (Nasteski, 2017, s. 5). Näin ollen luokittelu on edelleen keskeistä perinteisessä

kuvantunnistuksessa, kun taas regressiota sovelletaan erikoistuneilla osa-alueilla, kuten esimerkiksi poseerauksen arvioinnissa tai syvyyden ennustamisessa.

Toisin kuin valvotussa oppimisessa, valvomattomat oppimismenetelmät toimivat ilman merkittyjä tuotoksia. Näillä tekniikoilla pyritään löytämään datan piilotetut rakenteet - kuten klusterit, tiheysjakaumat tai matalaulotteiset representaatiot - pelkästään syötteen ominaisuuksien perusteella. Kuvantunnistuksessa valvomattomia menetelmiä käytetään usein tehtäviin, kuten samankaltaisten kuvien klusterointiin, visuaalisten käsitteiden löytämiseen tai ulottuvuuden vähentämiseen visualisointia varten (Bishop, 2006, s. 3). Esimerkiksi pääkomponenttianalyysia (PCA) käytetään yleisesti korkea-ulotteisen kuvatiedon projisoimiseen pienempiin aliavaruuksiin, mikä auttaa paljastamaan kuvioita ja yksinkertaistamaan syöttöavaruutta.

Kolmas paradigma, vahvistusoppiminen, eroaa sekä valvotusta että valvomattomasta oppimisesta. Vahvistusoppimisessa agentti oppii olemaan vuorovaikutuksessa ympäristönsä kanssa tekemällä toimia ja saamalla palautetta palkkioiden tai rangaistusten muodossa. Toisin kuin valvotussa oppimisessa, agentille ei kerrota oikeaa toimintatapaa, vaan sen on tutkittava ja pääteltävä paras strategia kokemuksen kautta. Vahvistusoppimista sovelletaan yleisemmin peräkkäisiin päätöksentekotehtäviin, kuten pelaamiseen tai robotiikkaan, mutta sen periaatteet vaikuttavat yhä enemmän myös visuaalisiin tunnistusjärjestelmiin, erityisesti skenaarioissa, joissa päätökset kehittyvät ajan kuluessa (Bishop, 2006, s. 4).



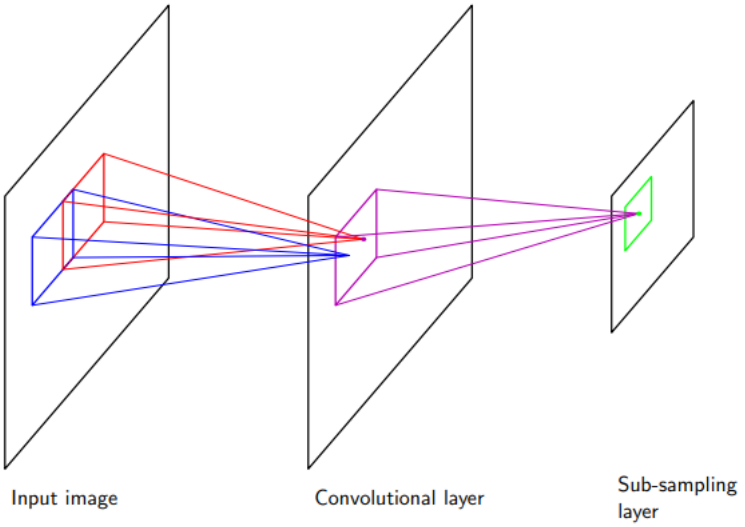
Kuva 3. Yleiskuvaus koneoppimisalgoritmeista. Koneoppiminen on tekoälyn osa-alue. Tämäkuvahavainnollistaa eri koneoppimisalgoritmien hierarkiaa, mukaan lukien valvotut ja valvomattomat sekä vahvistusoppimistekniikat. Valvotun oppimisen kaksi pääluokkaa ovat luokittelu ja regressio, jotka johtavat vastaavasti diskreetteihin/laadullisiin ja jatkuviin/laadullisiin tavoitteisiin (Rashidi, 2019, s.9).

Kuvassa 3 nähdään vielä yleiskuvaus koneoppimisalgoritmeista. Kaikissa näissä lähestymistavoissa perimmäinen tavoite on sama: suunnitella järjestelmiä, jotka pystyvät tulkitsemaan visuaalista tietoa tarkasti ja tehokkaasti hyödyntämällä esimerkkien avulla opittuja malleja. Koneoppiminen tarjoaa tähän tarvittavat tilastolliset ja laskennalliset puitteet, ja sen integroiminen kuvantunnistukseen on mullistanut alan. Kun nykyaikaisten tietokokonaisuuksien koko ja monimutkaisuus kasvavat ja kun sovellukset vaativat yhä enemmän reaaliaikaista suorituskykyä ja suurta tarkkuutta, koneoppimisen lähestymistavoista — erityisesti syviä neuroverkkoja käyttävistä lähestymistavoista — on tulossa välttämättömiä välineitä älykkäiden kuvapohjaisten järjestelmien suunnittelussa.

3.4 Syväoppiminen ja konvolutiiviset neuroverkot (CNN)

Syväoppiminen, koneoppimisen osa-alue, on mullistanut kuvantunnistuksen ottamalla käyttöön erittäin ilmaisuvoimaisia malleja, kuten konvoluutiohermoverkot (CNN). CNN:t

on erityisesti suunniteltu käsittelemään dataa, jolla on ruudukkomainen topologia, kuten 2D-kuvia tai 1D-aikasarjoja, hyödyntämällä tällaisissa tietorakenteissa esiintyviä tilahierarkioita ja malleja (Goodfellow, Bengio, & Courville, 2016, s. 330). Niiden arkkitehtuuri ammentaa inspiraatiota ihmisen näköaivokuoresta ja jäljittelee tiettyjä biologisia periaatteita, mikä mahdollistaa tehokkaat ja skaalautuvat ratkaisut tietokonenäkötehtäviin.



Kuva 4. Kaavio, joka kuvaa osaa konvoluutiohermoverkosta ja jossa näkyy konvoluutioyksiköiden kerros, jota seuraa alinäytteenottoyksiköiden kerros. Tällaisia kerroksia voidaan käyttää useita peräkkäisiä pareja (Bishop, 2006, s. 268).

Konvoluutioverkon rakenne on esitetty kuvassa 4. Konvoluutiokerroksessa yksiköt on järjestetty tasoihin, joista kutakin kutsutaan piirrekartaksi (feature map). Konvoluutiohermoverkoissa on kolme mekanismia: paikalliset reseptiiviset kentät, painojen jakaminen ja alinäytteenotto (Bishop, 2006, s. 268).

Kaavasta 1 nähdään, että jos sovellamme tällaista painotettua keskiarvo-operaatiota joka hetki, saamme uuden funktion s, joka antaa tasoitetun arvion avaruusaluksen sijainnista (Goodfellow, Bengio, & Courville, 2016, s. 331):

$$s(t) = \int x(a)w(t - a)da \tag{1}$$

Kaavan 2 operaatiota kutsutaan konvoluutioksi. Konvoluutio-operaatio merkitään yleensä tähdellä (Goodfellow, Bengio, & Courville, 2016, s. 331):

$$s(t) = (x * w)(t) \tag{2}$$

Jos yhtälössä 3 oletetaan nyt, että x ja w määritellään vain kokonaisluvulla t (aika), voimme määritellä diskreetin konvoluution (Goodfellow, Bengio, & Courville, 2016, s. 332):

$$s(t) = (x * w)(t) = \sum_{a=-\infty}^{\infty} x(a)w(t - a) \tag{3}$$

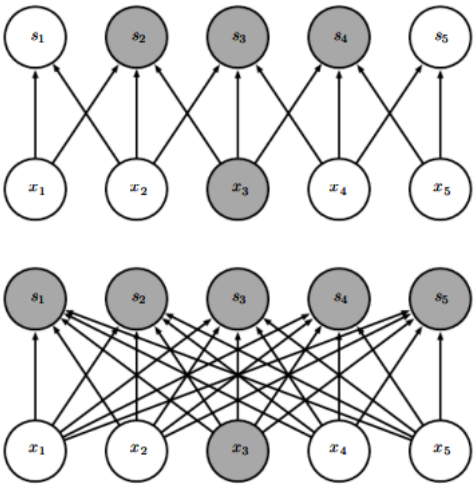
Yhtälössä 4 näemme, miltä näyttää, kun käytämme konvoluutiota useammalla kuin yhdellä akselilla kerrallaan. Jos esimerkiksi käytämme kaksiulotteista kuvaa I syötteenä, haluamme todennäköisesti käyttää myös kaksiulotteista ydintä K (Goodfellow, Bengio, & Courville, 2016, s. 332):

$$S(i,j) = (I * K)(i,j) = \sum_m \sum_n I(m,n)K(i - m, j - n) \tag{4}$$

Konvoluutio on kommutatiivinen, mikä tarkoittaa, että voimme kirjoittaa sen yhtälön 5 mukaisesti (Goodfellow, Bengio, & Courville, 2016, s. 332):

$$S(i,j) = (K * I)(i,j) = \sum_m \sum_n I(i - m, j - n)K(m,n) \tag{5}$$

CNN:ien ytimessä on konvoluutio-operaatio, matemaattinen työkalu, jota käytetään suodattimien eli ytimien soveltamiseen tulomatriisiin. Tämä operaatio laskee systemaattisesti paikallisia painotettuja summia naapuripikseleistä, ja sitä käytetään täysin kytkettyjen kerrosten sijasta, mikä vähentää merkittävästi parametrien määrää (Goodfellow et al., 2016, s. 331-332). Tyypillisessä ympäristössä 2D-konvoluutiota sovelletaan kuvatietoihin, liu'utetaan ydintä spatiaalisten ulottuvuuksien yli ja lasketaan ominaisuuskarttoja, jotka vangitsevat paikallisia kuvioita, kuten reunoja, tekstuureja tai kaltevuuksia.



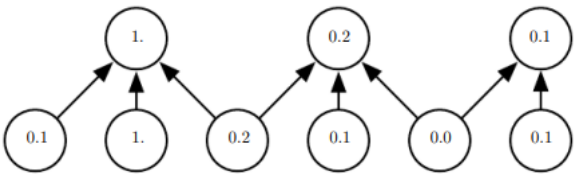
Kuva 5. Harva liitettävyys. Korostamme yhden tuloyksikön, x3, ja korostamme myös s:n lähtöyksiköt, joihin tämä yksikkö vaikuttaa. (Ylhäällä) Kun s muodostetaan konvoluutiolla, jonka ydin on leveydeltään 3, x vaikuttaa vain kolmeen ulostuloon. (Alhaalla) Kun s muodostetaan matriisikertoimella, kytkeytyvyys ei ole enää harva, joten x3 vaikuttaa kaikkiin ulostuloihin (Goodfellow, Bengio, & Courville, 2016, s. 336).

Yksi konvoluution tärkeimmistä eduista on sen tuoma harva liitettävyys**,** josta nähdään esimerkki kuvassa 5. Sen sijaan, että jokainen lähtöyksikkö olisi yhteydessä jokaiseen tuloyksikköön (kuten perinteisissä monikerroksisissa perceptroneissa), CNN:t rajoittavat yhteydet pieniin, paikallisiin alueisiin. Tämä vähentää huomattavasti laskennallista monimutkaisuutta ja muistivaatimuksia, mikä mahdollistaa mallien skaalaamisen korkearesoluutioisille kuvasyötteille (Goodfellow et al., 2016, s. 334). Kuten erilaisissa arkkitehtonisissa visualisoinneissa osoitetaan, konvoluutiolähestymistapa on useita kertaluokkia tehokkaampi kuin tiheät matriisikertomukset, erityisesti käsiteltäessä suuria syötteitä, kuten korkearesoluutioisia kuvia (Goodfellow et al., 2016, s. 340).

Konvoluutiokerrosten toinen merkittävä etu on parametrien jakaminen, jolloin samaa suodatinta sovelletaan kaikkiin tulon paikkoihin. Tämä vähentää huomattavasti opittavien parametrien määrää ja mahdollistaa sen, että verkko havaitsee piirteen riippumatta sen sijainnista kuvassa (Goodfellow et al., 2016, s. 335-336). Esimerkiksi yksi

reunan havaintosuodatin, jota sovelletaan tasaisesti koko kuvaan, voi korostaa reunoja riippumatta siitä, missä ne esiintyvät — mikä osoittaa ekvivalenssia translaation suhteen, ominaisuutta, jossa ulostulo siirtyy samalla tavalla, kun tulo siirtyy (Goodfellow et al., 2016, s. 338). CNN:t eivät kuitenkaan ole luonnostaan samanarvoisia muunnoksille, kuten rotaatiolle tai skaalautumiselle, ja tällaisten muunnosten käsittely vaatii lisäarkkitehtuuria tai datan lisäystekniikoita.

Toinen CNN:ien olennainen osa on pooling-operaatio, joka tyypillisesti seuraa konvoluutiota ja epälineaarisia aktivointivaiheita. Pooling vähentää ominaisuuskarttojen spatiaalisia ulottuvuuksia tiivistämällä paikallisia alueita (esim. ottamalla maksimi tai keskiarvo), mikä lisää laskennallista tehokkuutta ja edistää invarianssia pienille translaatioille (Goodfellow et al., 2016, s. 339-341). Esimerkiksi maksimipoolaus valitsee suurimman aktivoinnin määritellyn ikkunan sisällä, jolloin ominaisuuden voimakkain läsnäolo säilyy, mutta tarkka sijaintitieto hylätään. Tämä on erityisen hyödyllistä luokittelutehtävissä, joissa piirteen läsnäolo on tärkeämpää kuin sen tarkka sijainti.



Kuva 6. Pooling alinäytteistyksen vulla. Tässä käytetään max-poolingia, jossa poolin leveys on kolme ja poolien välinen askel on kaksi. Tämä pienentää esityskokoa kaksinkertaiseksi, mikä vähentää seuraavan kerroksen laskennallista ja tilastollista taakkaa. Huomaa, että oikeanpuoleisimman pooling-alueen koko on pienempi, mutta se on otettava mukaan, jos emme halua jättää huomiotta joitakin havaintoyksiköitä (Goodfellow, Bengio, & Courville, 2016, s. 344).

Kuvasta 6 nähdään, kuinka Pooling mahdollistaa myös alinäytteistyksen, joka pienentää edelleen seuraavien kerrosten syöttökokoa, mikä alentaa syvempien verkkokerrosten muisti- ja käsittelykustannuksia. Käytännön arkkitehtuurissa pooling-kerrokset käyttävät usein yhtä suurempia askelia paikkojen ohittamiseen, mikä parantaa sekä laskentanopeutta että yleistämiskykyä. Lisäksi kehittyneet muunnelmat, kuten

opittavissa oleva pooling, dynaaminen pooling ja aluekohtainen pooling (esim. kvadranttipooling), mahdollistavat CNN:n sopeutumisen datan ominaisuuksiin tai optimaalisten aggregointimallien oppimisen (Goodfellow et al., 2016, s. 344).

CNN:ien menestys johtuu niiden kyvystä yhdistää paikallisten ominaisuuksien louhinta ja hierarkkisen esityksen oppiminen. Alemmat kerrokset kaappaavat yksinkertaisia piirteitä (esim. reunat ja tekstuurit), kun taas syvemmät kerrokset integroivat nämä piirteet abstraktimmiksi representaatioiksi (esim. objektin osat tai kokonaiset objektit). CNN-verkot oppivat hierarkkisia piirre-edustuksia alimmista reunapiirteistä aina korkeampiin semanttisiin rakenteisiin (Shen et al., 2017; Litjens et al., 2017). Tämän koostumushierarkian ansiosta CNN:t voivat oppia monimutkaisia kuvioita erittäin tehokkaasti, mikä johtaa huippuluokan suorituskykyyn monissa erilaisissa näkötehtävissä, kuten kuvien luokittelussa, objektien havaitsemisessa ja kasvojentunnistuksessa.

Yhteenvetona voidaan todeta, että CNN:stä on tullut perustavanlaatuinen teknologia nykyaikaisessa kuvantunnistuksessa, mikä johtuu suurelta osin siitä, että ne käyttävät tehokkaasti harvaa vuorovaikutusta**,** parametrien jakamista ja käännös-ekvivalentteja esityksiä sekä integroivat epälineaarisia aktivointi- ja yhdistämisoperaatioita. Niiden arkkitehtuuri mahdollistaa skaalautuvan oppimisen suurilla kuvatietoaineistoilla, ja ne muodostavat nykyään lähes kaikkien tietokonenäön kilpailukykyisten syväoppimismallien selkärangan.

4 Evoluutioalgoritmit

Koneoppimisessa evoluutioalgoritmit tarjoavat houkuttelevan vaihtoehdon tai täydennyksen perinteisille koulutusalgoritmeille, erityisesti navigoitaessa ei-konveksissa hakutiloissa tai viritettäessä malleja, joilla on erilliset rakenteet (Xue, 2021). Tämä tekee niistä erityisen hyödyllisiä tehtävissä, kuten neuroverkkojen painojen optimoinnissa, hyperparametrien virityksessä ja mallien arkkitehtuurien automaattisessa suunnittelussa.

Kohdassa 4.1 esitellään evoluutioalgoritmien perusperiaatteet ja kuvataan yksityiskohtaisesti geneettisten algoritmien mekanismit, hyvyysfunktiot ja niiden rooli optimaalisten ratkaisujen etsimisessä. Kohdassa 4.2 syvennetään tätä tutkimalla evoluutioneuroneuroverkkoja, jotka ovat hybridi lähestymistapa, jossa EA:ta käytetään sekä neuroverkkojen painojen että rakenteiden optimointiin, mikä auttaa voittamaan joitakin takaisinkytkennän rajoituksia.

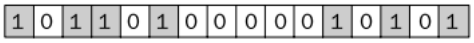
Yhdessä nämä kohdat havainnollistavat, miten evoluutiotekniikat edistävät mukautuvien, älykkäiden järjestelmien kehittämistä eri aloilla, kuten energiamallinnuksessa, kuvankäsittelyssä ja päätöksenteon tukemisessa.

4.1 Evoluutioalgoritmien perusteet

Evoluutioalgoritmit (EA) ovat biologisen evoluution periaatteiden innoittamia optimointitekniikoita. Nämä laskentamenetelmät tarkentavat iteratiivisesti ratkaisukandidaattien populaatiota käyttämällä valintaa, mutaatiota ja rekombinaatiota vastaavia operaatioita. Laajempi ala, joka tunnetaan nimellä evoluutiolaskenta, käsittää useita menetelmiä, kuten geneettiset algoritmit (GA), evoluutiostrategiat (ES) ja geneettinen ohjelmointi (GP) (Negnevitsky, 2005, s. 219-220).

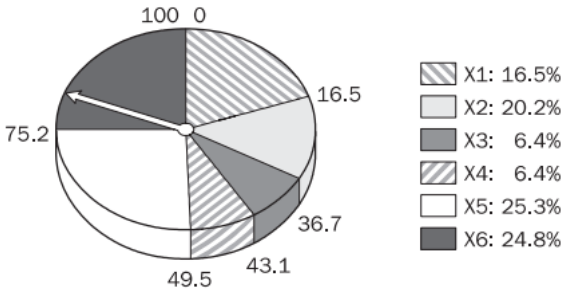
EA:n peruslähtökohtana on, että älykäs käyttäytyminen - joka ymmärretään kyvyksi sopeutua muuttuvaan ympäristöön — voi syntyä evoluutioprosessien simuloinnista. Kuten Turing (1950) ehdotti, älykkyyttä ei rajoita muoto tai biologia, ja se voi ilmetä järjestelmissä, jotka kykenevät mukautuvaan oppimiseen. Evoluutiolaskenta heijastaa tätä mallintamalla tapaa, jolla populaatiot kehittyvät ajan myötä kohti yhä optimaalisempia ratkaisuja (Negnevitsky, 2005, s. 219).

Keskeistä tässä prosessissa on evolutiivisen sopivuuden käsite, joka luonnollisissa järjestelmissä tarkoittaa organismin kykyä selviytyä ja lisääntyä. EA:ssa tämä tarkoittaa Fitness-funktiota, joka arvioi kvantitatiivisesti, kuinka hyvin ratkaisukandidaatti toimii tietyllä ongelma-alueella (Hartl & Clark, 1989; Atmar, 1994; Negnevitsky, 2005, s. 220). Korkean sopivuuden omaavat yksilöt valikoituvat todennäköisemmin lisääntymään, mikä ajaa populaatiota kohti parempia ratkaisuja ajan myötä.



Kuva 7. Keinotekoisen kromosomin 16-bittinen binäärijono (Negnevitsky, 2005, s. 221).

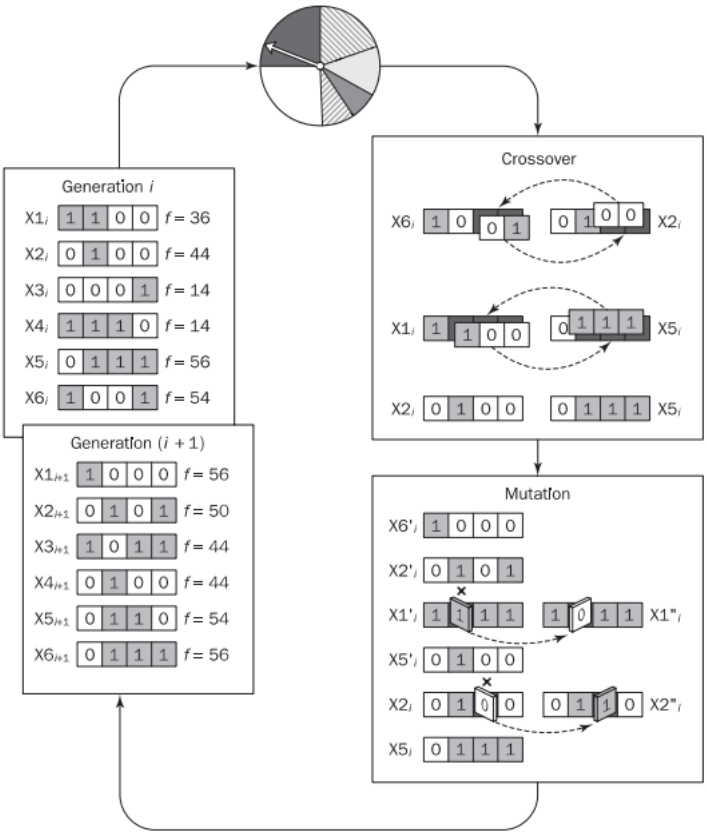
Ratkaisukandidaatit koodataan - usein kromosomia muistuttavina binäärijonoina - ja niille tehdään evoluutio-operaattoreita, kuten valinta, ristiintaulukoinnin ja mutaation avulla. Nämä operaatiot ohjaavat hakua ratkaisuavaruudessa. John Hollandin (1975) esittämä geneettinen algoritmi on tunnetuin EA-tyyppi. Se alkaa satunnaisesti luodusta populaatiosta ja etenee syklien (sukupolvien) kautta, joissa kromosomeja valitaan, yhdistetään uudelleen ja mutatoidaan (Negnevitsky, 2005, s. 221-223).



Kuva 8. Esimerkki rulettipyörävalinnasta (Goldberg, 1989; Negnevitsky, 2005, s. 225).

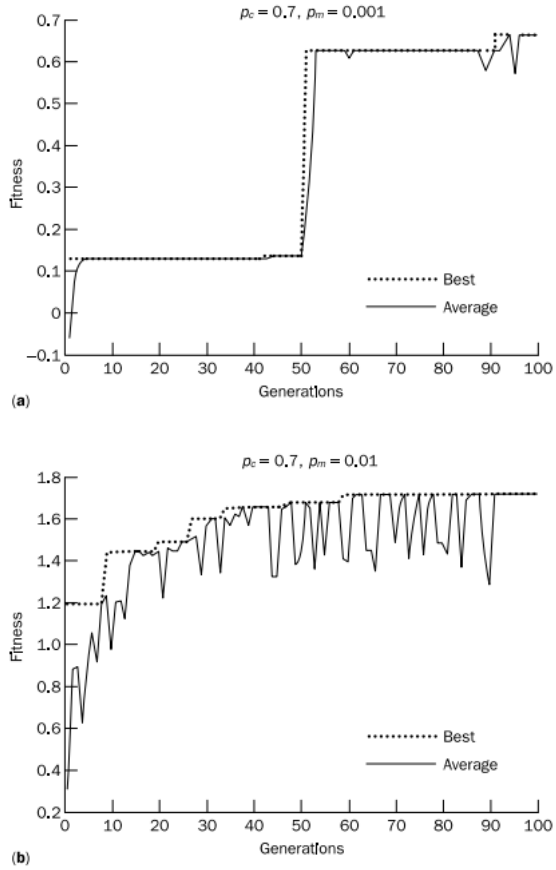
Valinta perustuu tyypillisesti hyvyysarvoon; yksilöillä, joilla on korkeampi hyvyysarvo, on suurempi mahdollisuus tulla valituksi risteytykseen. Yksi suosittu menetelmä on kuvassa 8 näkyvä rulettipyörävalinta, jossa kukin kromosomi vie osan valinta avaruudesta suhteessa sen hyvyysarvoon (Goldberg, 1989; Negnevitsky, 2005, s. 224-225).

Risteytysoperaattori vaihtaa segmenttejä vanhempien kromosomien välillä jälkeläisten luomiseksi. Jos risteytystä ei tapahdu (mikä tapahtuu tietyllä todennäköisyydellä), jälkeläiset ovat täsmällisiä klooneja. Mutaatio puolestaan tuo mukanaan satunnaisia muunnelmia kääntämällä kromosomissa olevia bittejä. Vaikka mutaatio on harvinaista, se on ratkaisevan tärkeää ennenaikaisen konvergenssin estämiseksi ja populaation monimuotoisuuden ylläpitämiseksi (Negnevitsky, 2005, s. 225-227).



Kuva 9. GA-sykli (Negnevitsky, 2005, s. 227)

Kuvassa 9 havainnollistetaan GA-sykliä. Siinä GA jatkaa iterointia, kunnes lopetuskriteeri täyttyy - usein ennalta määritetty sukupolvien määrä tai suorituskyvyn taso. Tuloksena syntyvä populaatio konvergoi tyypillisesti lähes optimaalisiin ratkaisuihin. Algoritmi voi kuitenkin joskus pysähtyä paikallisiin optimiarvoihin, minkä vuoksi mutaatio ja populaation monimuotoisuus ovat ratkaisevan tärkeitä (Negnevitsky, 2005, s. 227-230).



Kuva 10. Suorituskykykaaviot 100 sukupolven kuudelle kromosomille: a) huippufunktion paikallinen maksimiratkaisu ja b) globaali maksimiratkaisu (Negnevitsky, 2005, s. 230).

Suorituskykyä sukupolvien aikana havainnollistetaan usein suorituskykykaavioiden avulla, joissa esitetään iteraatioiden keskimääräiset ja parhaat hyvyysarvot. Näistä käy ilmi, miten populaatio kehittyy ja konvergoi (Negnevitsky, 2005, s. 230-231). Kuvasta 10 nähdään, kuinka mutaatioarvon nostaminen 0,001:stä 0,01:een parantaa hyvyysarvoa (Fitness) huomattavasti.

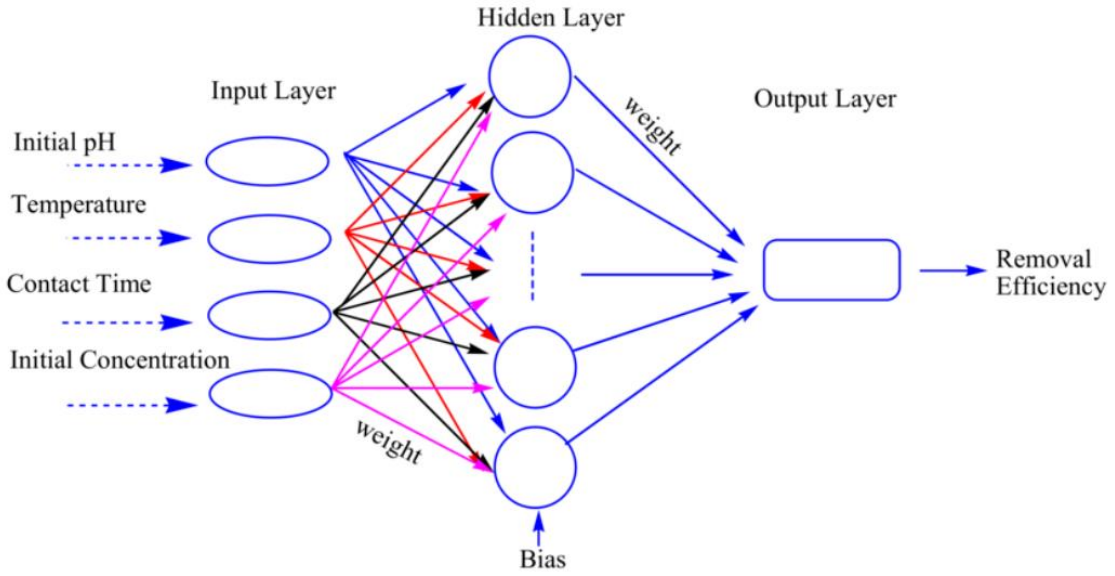
GA:iden teoreettinen perusta on skeemateoriassa, joka selittää, miten korkean kelpoisuuden omaavat osamallit (skeemat) leviävät sukupolvien kautta. Hollandin (1975) mukaan keskimääräistä parempaa hyvyysarvoa antavat skeemat selviytyvät ja lisääntyvät todennäköisemmin, varsinkin jos ne ovat matalan järjestyksen ja lyhyen pituuden omaavia, jolloin ne ovat vähemmän alttiita ristiin kytkennän tai mutaation aiheuttamille häiriöille (Negnevitsky, 2005, s. 231-233).

Joustavuudestaan huolimatta EA:t eivät ole yleisesti ottaen optimaalisia. Huono koodaus tai parametrien viritys voi haitata suorituskykyä tai vääristää hakua. Huolellisesti sovellettuna evoluutioalgoritmit ovat kuitenkin tehokkaita epälineaaristen, korkea-ulotteisten ja huonosti ymmärrettyjen optimointiongelmien ratkaisemisessa (Negnevitsky, 2005, s. 233-234).

Yhteenvetona voidaan todeta, että evoluutioalgoritmit tarjoavat vankan, biologisesti inspiroituneen kehyksen monimutkaisten hakuavaruuksien tutkimiseen. Niiden stokastinen mutta mukautuva luonne tekee niistä erityisen sopivia alueille, joilla perinteiset menetelmät epäonnistuvat tai ovat epäkäytännöllisiä.

4.2 Evolutiiviset neuroverkot

Vaikka keinotekoisia neuroverkkoja (artificial neural networks, ANN) käytetään laajalti monimutkaisiin hahmontunnistus- ja funktioiden approksimointitehtäviin, niillä on omat rajoituksensa. Yksi perinteisten neuroverkkojen suunnittelun suurimmista haasteista on koulutus ja topologian valinta. Yleisesti käytetty backpropagation-algoritmi on joustava ja matemaattisesti helposti lähestyttävä, koska se perustuu differentioituviin aktivointifunktioihin, mutta se konvergoi usein Sub optimaalisiin paikallisiin minimiin (Negnevitsky, 2005, s. 285). Tämä voi vakavasti haitata verkon kykyä yleistää reaalimaailman sovelluksissa.



Kuva 11. Illustraatio BP-ANN-arkkitehtuurista (Back Propagation Artificial Neural Network) (Ruan, 2017, s. 4).

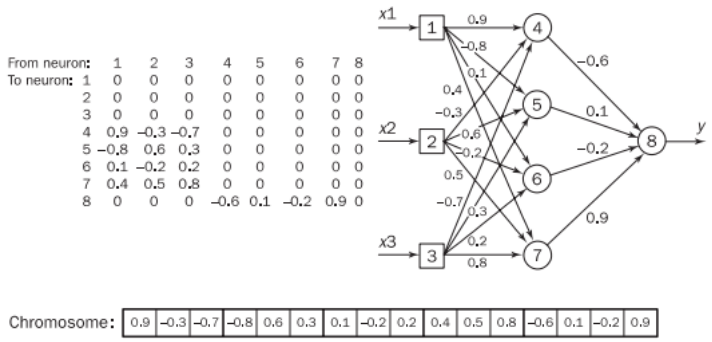
Kuva 11 havainnollistaa takaisinkulkeutuvan keinotekoisen neuroverkon (BP-ANN) arkkitehtuuria. Se koostuu kolmesta kerroksesta: sisääntulo-, piilo- ja ulostulokerroksesta**.** Painot (värilliset viivat) yhdistävät jokaisen neuronin kerrosten välillä, ja oppimiseen kuuluu näiden painojen säätäminen ennustusvirheen minimoimiseksi backpropagation-algoritmin avulla.

Näiden ongelmien ratkaisemiseksi evoluutiolaskenta tarjoaa vaihtoehtoisen lähestymistavan. Käyttämällä geneettisiä algoritmeja (GA) globaalina hakustrategiana sekä painojen optimointia että arkkitehtuurin suunnittelua voidaan käsitellä stokastisina optimointiongelmina. Tämä on synnyttänyt käsitteen evolutiiviset neuroverkot - hybridijärjestelmät, joissa yhdistyvät luonnonvalinnan periaatteet ja keinotekoinen neuro-oppiminen (Negnevitsky, 2005, s. 286).

4.2.1 Evolutiivinen painon optimointi

Evolutiivinen painojen optimointi korvaa gradienttipohjaisen oppimisen populaatiopohjaisella haulla. Kukin neuroverkko on koodattu kromosomiksi, jossa painot edustavat geenejä. Esimerkiksi feedforward-verkko, jossa on 16 painotettua yhteyttä,

voidaan kuvata 16 geenin kromosomiksi, jossa jokainen geeni on reaaliluku, joka edustaa painoarvoa (Negnevitsky, 2005, s. 286). Tämä kromosomi voidaan edelleen ryhmitellä, jotta säilytetään toiminnallisesti toisiinsa liittyvien painojen rakennuspalikat, mikä parantaa periytymistä ristiin kytkennän aikana.



Kuva 12. Painojen koodaus kromosomissa.

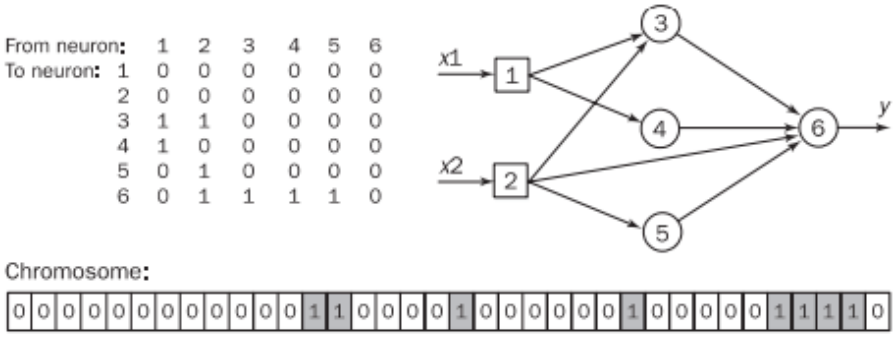
Kuvassa 12 esimerkki painojen koodauksesta kromosomissa. hyvyysarvon arviointi saadaan aikaan purkamalla kromosomi neuroverkoksi, ajamalla verkko harjoitusjoukon läpi ja laskemalla hyvyysarvotulos, joka on kääntäen verrannollinen neliövirheiden summaan (SSE). Tehtävässä paremmin suoriutuvat verkot saavat korkeamman Fitness-arvon ja lisääntyvät todennäköisemmin (Negnevitsky, 2005, s. 286).

Risteytysoperaatiot vaihtavat geneettistä materiaalia kahden vanhemman kromosomin välillä ja luovat jälkeläisiä, jotka perivät painokokoonpanot molemmilta. Mutaatio tuo vaihtelua häiritsemällä yhtä tai useampaa kromosomin painoa pienellä satunnaisarvolla (Negnevitsky, 2005, s. 287). Toistuvalla valinnalla, lisääntymisellä ja vaihtelulla sukupolvien aikana algoritmi kehittää verkkoja, joiden painot ovat yhä optimoidumpia.

4.2.2 Kehittyvät verkkotopologiat

Painojen virittämisen lisäksi evoluutiolaskentaa voidaan soveltaa myös optimaalisten neuroverkkoarkkitehtuurien etsimiseen. Verkkotopologia – kuten piilokerrosten määrä,

neuronien määrä kerrosta kohti ja yhteydet — määrittää usein mallin menestyksen, mutta se valitaan perinteisesti heuristiikan tai kokeilemalla ja erehtymällä (Negnevitsky, 2005, s. 287).



Kuva 13. Verkon topologian suora koodaus (Negnevitsky, 2005, s. 288).

Kuvassa 13 nähdään kuinka, suoraa koodausta käyttäen syöttöverkon liitettävyys voidaan esittää binäärimatriisina, jossa kukin matriisin merkintä ilmaisee neuronien välisen yhteyden olemassaolon tai puuttumisen. Tämä matriisi litistetään sitten geneettisiin operaatioihin soveltuvaksi binääriseksi kromosomiksi (Negnevitsky, 2005, s. 288).

Hyvyysarvoa arvioidaan jälleen verkon SSE:n avulla ennalta määritellyn harjoitteluepookin määrän jälkeen. Geneettisiä operaattoreita, kuten rivipohjaista risteytystä ja bittien kääntämistä mutaatiota, käytetään topologioiden kehittämiseksi sukupolvien aikana. Evoluutiosykli jatkuu, kunnes tietty lopetuskriteeri, kuten asetettu sukupolvien määrä, täyttyy (Negnevitsky, 2005, s. 288-289).

4.2.3 Kehittyneet sovellukset ja tulevaisuuden näkymät

Evolutiivisten neuroverkkojen käyttöala ulottuu painojen ja topologian optimointia laajemmalle. Evoluutiostrategioita voidaan käyttää myös aktivointifunktioiden kehittämiseen ja syöttöominaisuuksien valintaan, erityisesti skenaarioissa, joissa on kyse korkeaulotteisesta tai jäsentymättömästä datasta (Negnevitsky, 2005, s. 289). Tällainen sopeutumiskyky on tehnyt evolutiivisista neuroverkoista kasvavan

tutkimuskiinnostuksen kohteena, erityisesti ongelmissa, joissa hakuavaruus on suuri ja huonosti ymmärretty.

Tulevaisuudessa voidaan tutkia adaptiivisia koodausjärjestelmiä, dynaamista populaation mitoitusta ja integrointia muihin metaheuristisiin strategioihin. Kuten Negnevitsky (2005) toteaa, evolutiiviset neuroverkot ovat lupaava paradigma, jonka avulla voidaan rakentaa älykkäitä järjestelmiä, jotka kykenevät itseorganisoitumaan, sopeutumaan ja yleistymään.

5 Menetelmät ja materiaalit

Tässä luvussa kuvataan yksityiskohtaisesti tutkimuksesta, aineistoista, syväoppimisarkkitehtuureista ja evoluutioalgoritmista. Kaikki kokeet toteutettiin HP 14 em0xxx kannettavalla tietokoneella ja MATLAB-ohjelmistossa käyttämällä mutaatioihin ja risteytyksiin räätälöityjä skriptejä yhdessä tavallisten syväoppimisen työkalupakettien kanssa. Menetelmä suunniteltiin toistettavaksi ja muunneltavaksi muihin kuvien luokitteluongelmiin.

5.1 Tutkimuksen suunnittelu ja tavoitteet

Tämän tutkimuksen tavoitteena on tutkia konvoluutiohermoverkkojen (CNN) arkkitehtuurien optimointia kuvantunnistustehtäviin käyttämällä evoluutioalgoritmeja. Ensisijaisena tavoitteena on selvittää, voiko hybridi-lähestymistapa, jossa yhdistetään syväoppiminen ja bioinspiroitu optimointi, parantaa luokittelun suorituskykyä rajoitetuissa laskentaympäristöissä.

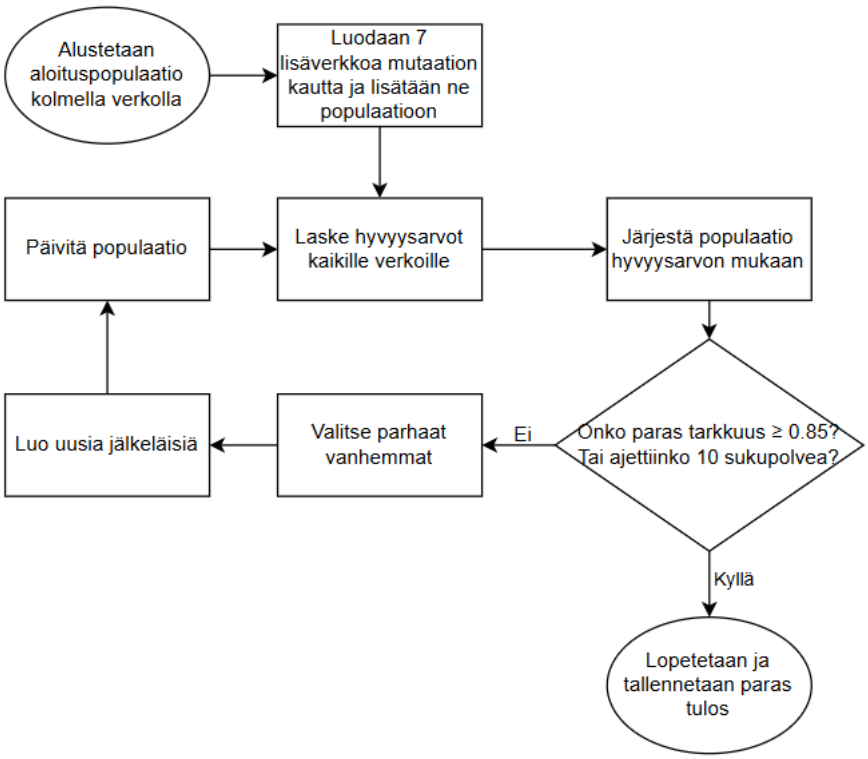
Tutkimuksen suunnittelu noudattaa kokeellista ja vertailevaa metodologiaa. Alkuvaiheessa koulutetaan joukko CNN-perusmalleja kahdella kuvadatajoukolla: toisessa keskitytään nisäkkäiden lajien luokitteluun ja toisessa kasvojen ilmeiden tunnistamiseen. Nämä verkot koulutetaan käyttämällä perinteisiä syväoppimisen käytäntöjä, kuten kiinteitä kerroskonfiguraatioita ja koulutusta Adam-optimoijalla rajoitetun määrän ajanjaksoja.

Toisessa vaiheessa painopiste siirtyy automaattiseen arkkitehtuurin etsintään evoluutioalgoritmien avulla. Tässä vaiheessa parhaiten suoriutuneet CNN-perusmallit toimivat evoluutioprosessin alkupopulaationa. Lisäverkot luodaan mutaatio- ja risteytysoperaatioilla, jotka muuttavat arkkitehtonisia elementtejä, kuten konvoluutiosuodattimien lukumäärää, askelarvoja tai verkon syvyyttä. Kunkin verkon sopivuus arvioidaan validoinnin ja testitarkkuuden perusteella, jotka toimivat yleistettävyyden mittareina.

Evoluutiokierros jatkuu ennalta määritellyn sukupolvien määrän ajan, ja verkot valitaan, risteytetään ja mutatoidaan niiden sopivuuden perusteella. Kehittyneiden verkkojen suorituskykyä verrataan sitten perusmalleihin, jotta voidaan määrittää, voiko evoluutioprosessi optimoida CNN-arkkitehtuureja tehokkaasti tarkkuuden ja kestävyyden kannalta.

Tämä kaksivaiheinen suunnittelu – manuaalinen perusmallinnus ja sen jälkeen automatisoitu evoluutio – mahdollistaa syvällisen tutkimuksen evoluutiomenetelmien eduista ja rajoituksista syväoppimiseen perustuvassa kuvantunnistuksessa. Mallien suorituskyvyn parantamisen lisäksi tutkimuksessa pyritään tuomaan esiin arkkitehtuurin optimoinnin käytännön haasteita, kuten laitteistorajoitukset, koulutuksen vakaus ja laskennan tehokkuus.

Seuraavat tutkimuskysymykset ohjaavat metodologiaa:
(1) Voivatko evoluutioalgoritmit parantaa CNN:n suorituskykyä kuvien luokittelussa verrattuna manuaalisesti suunniteltuihin verkkoihin?
(2) Millaiset arkkitehtuurimuutokset ovat tehokkaimpia mallin yleistettävyyden parantamisessa? Näihin kysymyksiin vastaamalla tutkimus pyrkii edistämään älykkäiden, itseoptimoituvien neuroverkkoarkkitehtuurien jatkuvaa kehittämistä, jotka soveltuvat käytettäväksi todellisissa kuvantunnistustilanteissa.



Kuva 14. Evoluutioalgoritmin optimointivuokaavio.

Evoluutiokoulutusprosessi on kuvattu kuvassa 14. Se alkaa populaation alustamisella ja lisäverkostojen luomisella mutaation avulla. Kunkin verkoston suorituskyky arvioidaan ja lajitellaan sopivuuden perusteella. Parhaiten suoriutuneet valitaan vanhemmiksi uusien jälkeläisten tuottamiseksi, ja prosessi toistuu, kunnes tarkkuuskynnys 0,85 saavutetaan tai kymmenen sukupolvea on suoritettu.

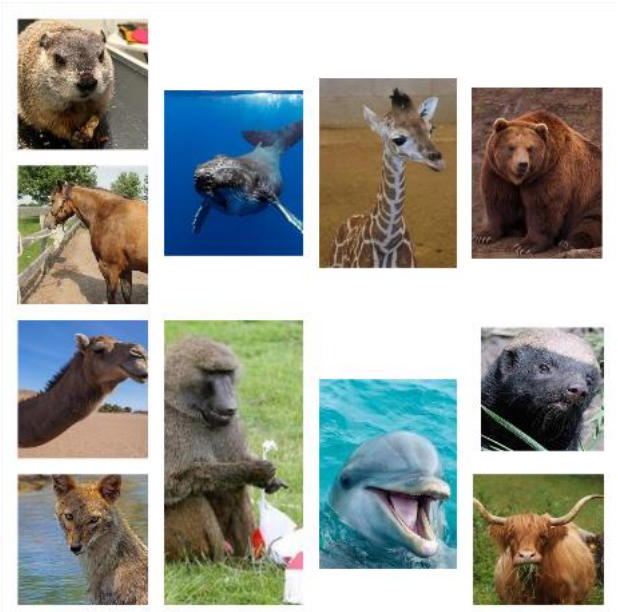
Luvussa 5.3 tutkitaan tutkimuksen aloituspopulaatioina käytettyjä syväoppimisarkkitehtuureja. Luvussa 5.4 syvennytään sitten evoluutioalgoritmiin.

5.2 Aineisto

Tässä tutkimuksessa käytetään kahta erillistä aineistoa evoluutioalgoritmeilla optimoitujen konvoluutioisten neuroverkkojen (CNN) suorituskyvyn ja yleistettävyyden arvioimiseksi: nisäkkäiden lajiluokitteluaineisto ja kasvojen ilmeiden tunnistamisaineisto. Aineistot valittiin niiden monimuotoisuuden, saatavuuden ja merkityksen perusteella.

5.2.1 Nisäkkäiden luokitteluaineisto

Nisäkkäiden luokittelua koskeva tietokanta koostuu eri nisäkkäiden lajeista otetuista korkean resoluution RGB-kuvista. Kuvat on kerätty ja järjestetty luokkakansioihin, joista kukin edustaa erillistä lajia. Tietokanta sisältää yhteensä 11 luokkaa, joihin kuuluu sekä kotieläimiä että villieläimiä. Kaikkien kuvien koko on 256×256 pikseliä, näin ollen syöttömitat ovat yhdenmukaiset kaikissa CNN-arkkitehtuureissa. Kuvassa 15 nähdään esimerkkikuvia kyseisistä nisäkkäistä. Taulukosta 1 nähdään myös kuvaus nisäkkäiden luokitteluaineistosta.



Kuva 15. Esimerkkikuvia aineiston eri nisäkkäistä.

Taulukko 1. Kuvaus nisäkkäiden luokitteluaineistosta.

Nisäkkäät	**Määrä**
Paviaani	330
Mäyrä	310
Sininen Valas	240
Ruskea Karhu	300
Kameli	254
Delfiini	311
Kirahvi	305
Metsämurmeli	309
Ylämaankarja	311
Hevonen	303
Sakaali	278
Yhteensä	3251

Kyseinen aineisto jaettiin kolmeen osajoukkoon käyttämällä 60-20-20-jakoa:

Koulutusjoukko (60 %): Käytetään mallin koulutukseen.

Validointijoukko (20 %): Käytetään mallin arviointiin koulutuksen aikana ja evoluutiovalinnan ohjaamiseen.

Testisarja (20 %): Käytetään lopullisen mallin suorituskyvyn arviointiin evoluution jälkeen.

Tämän tietokannan avulla voidaan arvioida CNN-verkkojen kykyä tunnistaa luonnollisia piirteitä, kuten turkin tekstuuri, ruumiinmuoto ja taustatekijät, jotka asettavat luokittelulle ainutlaatuisia haasteita.

5.2.2 Kasvojen ilmeiden luokitteluaineisto

Toinen tässä tutkimuksessa käytetty tietojoukko kohdistuu ihmisten perusilmeiden tunnistamiseen. Kuvasta 16 nähdään, kuinka se koostuu harmaasävyisistä kuvista, joiden resoluutio on 48×48 pikseliä, ja kattaa seitsemän tunnetilaa: viha, inho, pelko, ilo, suru, yllätys ja neutraalius. Tietojoukkoa käytetään yleisesti affektiivisessa tietojenkäsittelyssä ja tunteita tunnistavassa tekoälytutkimuksessa. Taulukosta 2 nähdään myös kuvaus ilmeiden luokitteluaineistosta.

Kuva 16. Esimerkki kuvia datan eri tunnetiloista.

Taulukko 2. Kuvaus kasvojen ilmeiden luokitteluaineistosta.

Viha	3999
Inho	436
Pelko	4097
Ilo	7215
Suru	4965
Yllätys	4830
Neutraali	3171
Yhteensä	28709

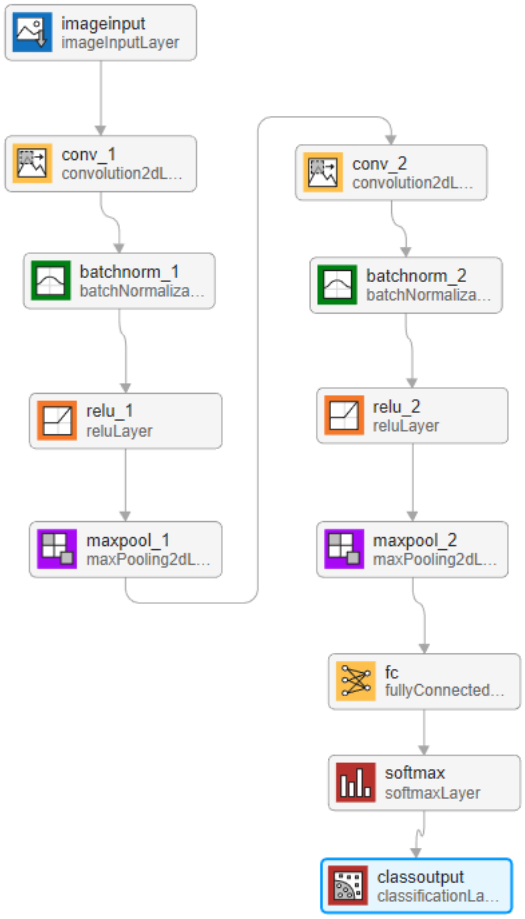
Samanlainen 60-20-20-jako käytettiin jakamaan aineisto koulutus-, validointi- ja testisarjoihin. Tämä aineisto on erityisen hyödyllinen arvioitaessa CNN:ien kykyä havaita hienovaraisia visuaalisia piirteitä ja ilmeitä vaihtelevissa valaistusolosuhteissa ja kasvojen suunnissa.

5.3 Tutkimuksessa käytetyt syväoppimisarkkitehtuurit

Tämän tutkimuksen ydin on konvoluutioisten neuroverkkojen (CNN) käyttö ja optimointi kuvantunnistustehtävissä. CNN:t ovat osoittautuneet erittäin tehokkaiksi erottamaan kuvista piirteiden spatiaalisia hierarkioita, minkä ansiosta ne soveltuvat monimutkaisiin luokittelutehtäviin, kuten nisäkkäiden lajien tunnistamiseen ja kasvojen ilmeiden havaitsemiseen. Tässä tutkimuksessa tarkastellut arkkitehtuurit vaihtelevat syvyydeltään ja monimutkaisuudeltaan, mikä mahdollistaa niiden suorituskyvyn perusteellisen tutkimisen, kun niitä kehitetään evoluutioalgoritmilla.

Kuva 17. Ensimmäinen arkkitehtuuri.

Alun perin suunniteltiin kolme erilaista CNN-arkkitehtuuria, jotka koulutettiin perinteisillä takaisinkytkentämenetelmillä. Nämä mallit toimivat evoluutioalgoritmin alkupopulaationa. Ensimmäinen arkkitehtuuri kuvassa 17 on kevyt CNN, joka koostuu kolmesta konvoluutiosta, joiden suodatussyvyys kasvaa, ja jota seuraa erän normalisointi, ReLU-aktivointifunktiot ja max-pooling-operaatiot. Se päättyy täysin kytkettyyn kerrokseen, joka vastaa datajoukon luokkien lukumäärää. Tämä verkko on tarkoitettu laskennallisesti tehokkaaksi vertailukohtana.



Kuva 18. Toinen arkkitehtuuri.

Toinen verkkoarkkitehtuuri kuvassa 18 on yksinkertainen ja tehokas konvoluutiohermoverkko, joka on suunniteltu kuvien luokitteluun. Jokainen blokki sisältää konvoluutio kerroksen, erän normalisoinnin, ReLU-aktivoinnin ja maksimi poolauksen. Nämä kerrokset poimivat ja tarkentavat kuvan piirteitä ja pienentävät samalla kuvan ulottuvuuksia. Viimeiset kerrokset koostuvat täysin kytketystä kerroksesta, softmax-kerroksesta todennäköisyyslähtöä varten ja luokittelukerroksesta

Kolmas ja monimutkaisin arkkitehtuuri on syvä CNN, jossa on useita konvoluutioblokkeja, joista kukin koostuu konvoluutioiden sarjasta, erän normalisoinnista, ReLU-aktivoinneista ja alinäytteenottokerroksista. Siinä on pullonkaulakerrokset ja dropout-regularisointi ylisovittumisen vähentämiseksi. Tämän arkkitehtuurin syvyys mahdollistaa erittäin epälineaaristen suhteiden mallintamisen ja monimutkaisten kuvioiden

havaitsemisen kuvadatoissa. Sen parametrien lukumäärän kasvu kuitenkin lisää myös ylisovittumisen ja laskennallisen tehottomuuden riskiä.

Kaikki CNN:t toteutettiin ja koulutettiin MATLAB-ohjelmistossa Deep Learning Toolbox -työkalupaketin avulla. Koulutus suoritettiin Adam-optimoijalla, jonka alkuperäinen oppimisnopeus oli 0,001 ja enimmäisepokkimäärä 10. Jokainen verkko arvioitiin sen validointitarkkuuden perusteella, joka toimi evoluutioprosessin ensisijaisena sopivuusmittarina.

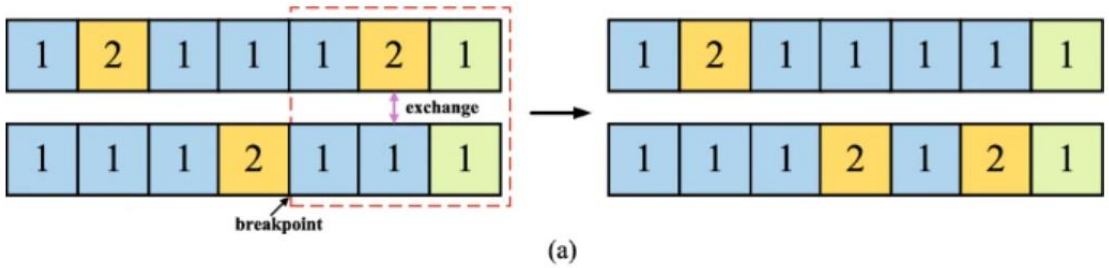
Alustavan koulutuksen jälkeen arkkitehtuurit käytiin läpi evoluutio-optimointiprosessissa, jossa sovellettiin mutaatio- ja risteytysoperaatioita uusien ehdokasmallien luomiseksi. Mutaatiot kohdistuivat rakenteellisiin komponentteihin, kuten konvoluutiosuodattimen kokoon, suodattimien lukumäärään ja askelpituuteen. Risteytys yhdisteli vanhempien verkkojen kerroksia hybridien jälkeläisten tuottamiseksi. Valitsemalla ja kehittäen malleja toistuvasti validointitulosten perusteella algoritmi pyrki löytämään arkkitehtuureja, joilla oli erinomainen tarkkuus ja yleistettävyys.

Tämä evoluutiopohjainen lähestymistapa CNN-optimointiin mahdollisti manuaalisen kokeilun ja erehdyksen ylittävän laajemman suunnittelutilan tutkimisen. Sen avulla mallit pystyivät mukauttamaan monimutkaisuuttaan dynaamisesti suorituskykypalautteen perusteella, mikä johti lopulta arkkitehtuureihin, jotka ovat paitsi tarkempia myös sopivimpia tässä tutkimuksessa käytettyjen datajoukkojen ominaisuuksiin.

5.4 Verkkojen optimointiin tarkoitettu evoluutioalgoritmi

Tässä tutkimuksessa käytetty evoluutioalgoritmi toimii metaheuristisena optimointistrategiana syväoppimisarkkitehtuurien parantamiseksi. Sen tavoitteena on kehittää neuroverkkomallien populaatiota peräkkäisten sukupolvien aikana valitsemalla, yhdistelemällä ja muuntamalla ehdokasarkkitehtuureja niiden suorituskyvyn perusteella.

Alkuperäinen populaatio koostuu muutamasta manuaalisesti suunnitellusta konvoluutioverkosta sekä satunnaisesti luoduista tai muunnelluista versioista. Jokainen populaation yksittäinen verkko arvioidaan validointidatajoukolla, ja sen sopivuus mitataan validoinnin ja testin tarkkuuden painotetulla yhdistelmällä.



Kuva 19. Esimerkki risteyttämisestä (Yang Xu & Yongjie Ma, 2023, s.7).

Parhaiten suoriutuneet verkot valitaan seuraavan sukupolven vanhemmiksi. Nämä verkot käyvät läpi risteytyksen, jossa kahden vanhemman arkkitehtuurin osia yhdistetään kuvan 19 tavoin uuden lapsiverkon luomiseksi. Mutaatioita sovelletaan myös konvoluutioverkon parametreihin, kuten askelmittaan ja suodattimien lukumäärään, pienien satunnaisten muutosten aikaansaamiseksi, mikä ylläpitää populaation monimuotoisuutta ja estää ennenaikaisen konvergenssin. Prosessin keskeyttämisen estämiseksi on otettu käyttöön uudelleenkokeilumekanismi, joka estää virheellisten mutaatioiden aiheuttamien virheellisten verkkojen prosessin keskeyttämisen. Elitismiä sovelletaan säilyttämällä kunkin sukupolven parhaat verkot. Tämä varmistaa, että ratkaisujen laatu ei heikkene ajan myötä.

Prosessi jatkuu ennalta määritellyn sukupolvien määrän ajan tai kunnes tyydyttävä sopivuuskynnys saavutetaan. Tämä lähestymistapa mahdollistaa verkon rakenteen ja suorituskyvyn asteittaisen parantamisen ilman manuaalista säätämistä, mikä tekee siitä hyödyllisen strategian hermoverkkoarkkitehtuurin etsimiseen laskennallisten rajoitusten alaisissa tilanteissa.

Taulukko 3. Evoluutioalgoritmin parametrit.

Parametrit	**Arvot**
Aloituspopulaatio	3
Populaatio	10
Sukupolvet	10
Mutaatioarvo	0.6
Kynnysraja	85%
Max. ajanjakso eli epoch koulutuksen aikana	8

Taulukko 4. Mutaatio-operaattori.

Parametrit	**Arvot**
Filter count eli suodattimien lukumäärä	±8 suodatinta, rajoitettu alueelle [8, 128]
Stride eli askelpituus	Muunnetaan ±1, rajoitettu arvoon [1, 4]
Padding eli reunatäydennys	Säilytetty alkuperäisestä kerroksesta

Taulukko 5. Risteytys-operaattori.

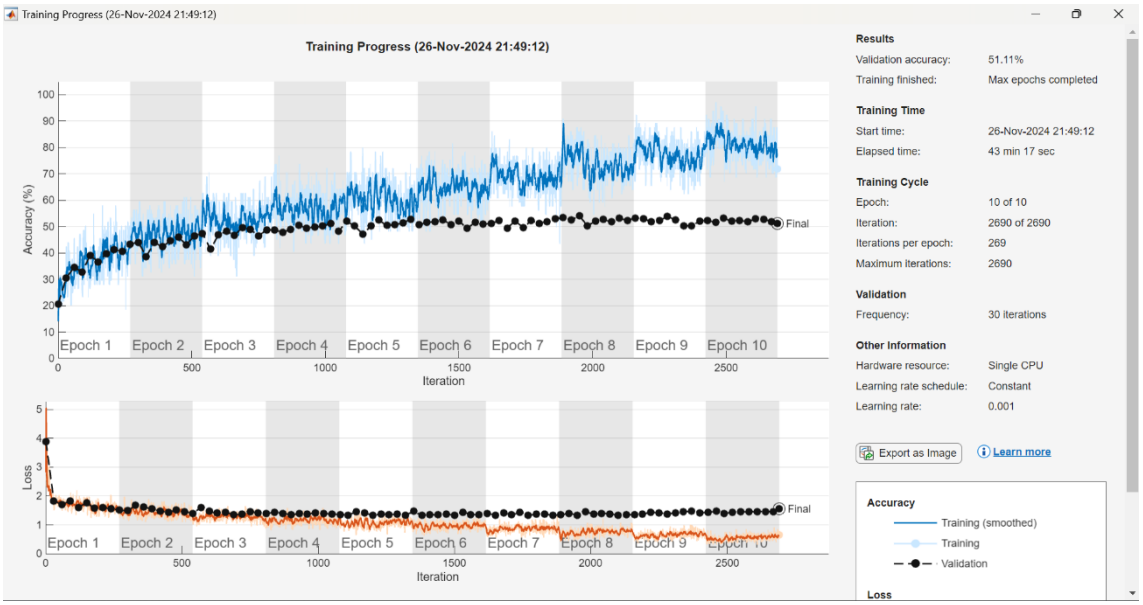
Parametrit	**Arvot**
Yhdistettävät	Kaksi periytyvää arkkitehtuuria
Leikkauspiste	Kummankin verkon puoliväli (keskikerros)

Evoluutiolaskennassa käytetyt keskeiset parametrit ja operaattorit on koottu taulukoihin 3–5. Taulukossa 3 esitetään evoluutioprosessin pääasiallinen kokoonpano, mukaan lukien alkupopulaatio, populaation koko, sukupolvien lukumäärä, mutaatioarvo ja varhaisen pysäyttämisen tarkkuuskynnys. Taulukossa 4 kuvataan mutaatio-operaattori, joka tuo verkkoon satunnaisia rakenteellisia muutoksia muokkaamalla

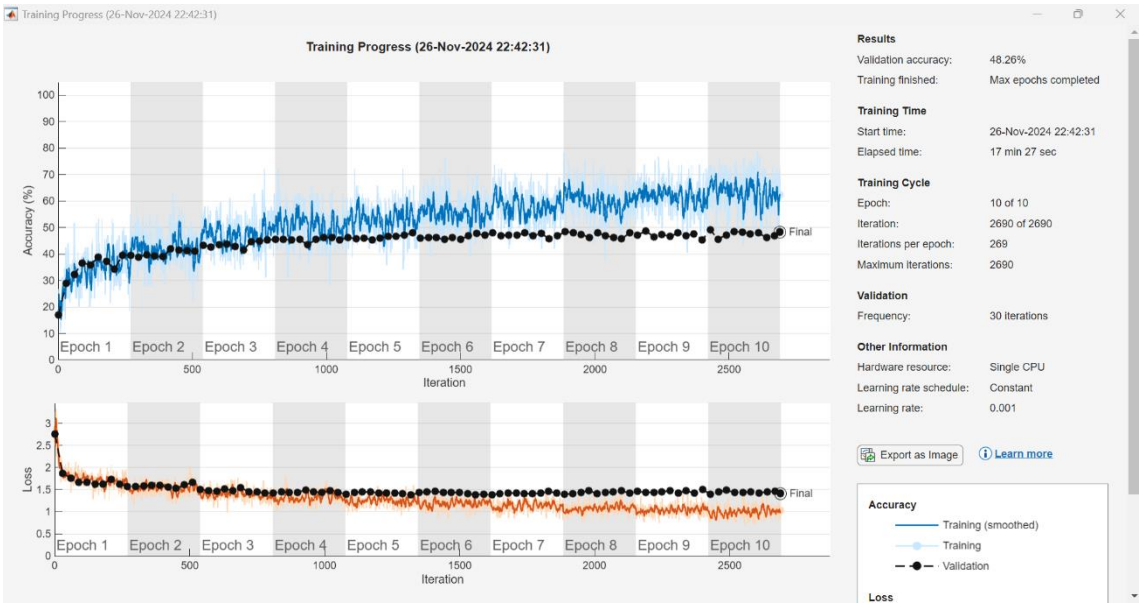
konvoluutiosuodattimien lukumäärää (±8 suodatinta, välillä [8, 128]) ja askelarvoja (±1, välillä [1, 4]) säilyttäen samalla alkuperäisen täytekonfiguraation tilallisen johdonmukaisuuden varmistamiseksi. Taulukossa 5 määritellään risteytysoperaattori, joka yhdistää kaksi vanhemman arkkitehtuuria jakamalla kunkin verkon keskipistekerroksessa ja yhdistämällä ne uuden verkon tuottamiseksi. Yhdessä nämä parametrit ja operaattorit ohjaavat evoluutiotutkimuksen tutkimuksen ja hyödyntämisen välistä tasapainoa, mikä mahdollistaa uusien konvoluutioverkkokokonaisuuksien löytämisen ja estää liiallisen laskennallisen monimutkaisuuden.

6 Tulokset

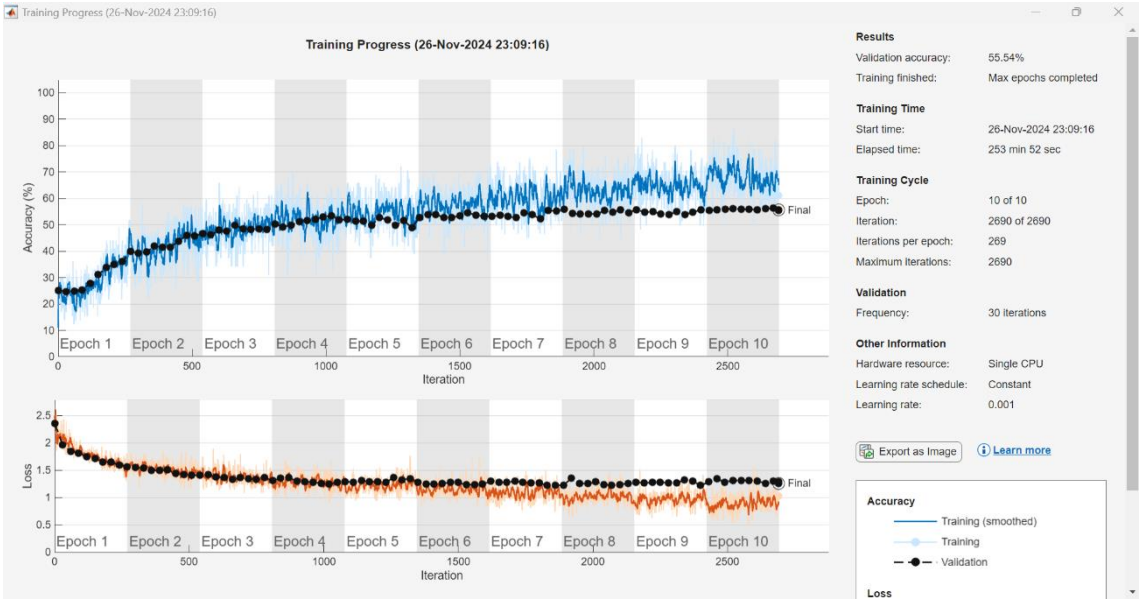
Tässä luvussa esitellään tulokset, jotka on saatu syvän oppimisen mallien harjoittelusta ja validoinnista kahteen erilliseen tietokokonaisuuteen: toisessa on kyse kasvojen ilmeiden tunnistamisesta ja toisessa nisäkäslajien luokittelusta. Kutakin mallia arvioitiin sen tarkkuuden ja häviösuuntausten perusteella harjoittelu- ja validointijaksojen aikana. Tulokset visualisoidaan osoittamaan oppimisen eteneminen ja evolutiivisen optimoinnin tehokkuus viiden sukupolven aikana.



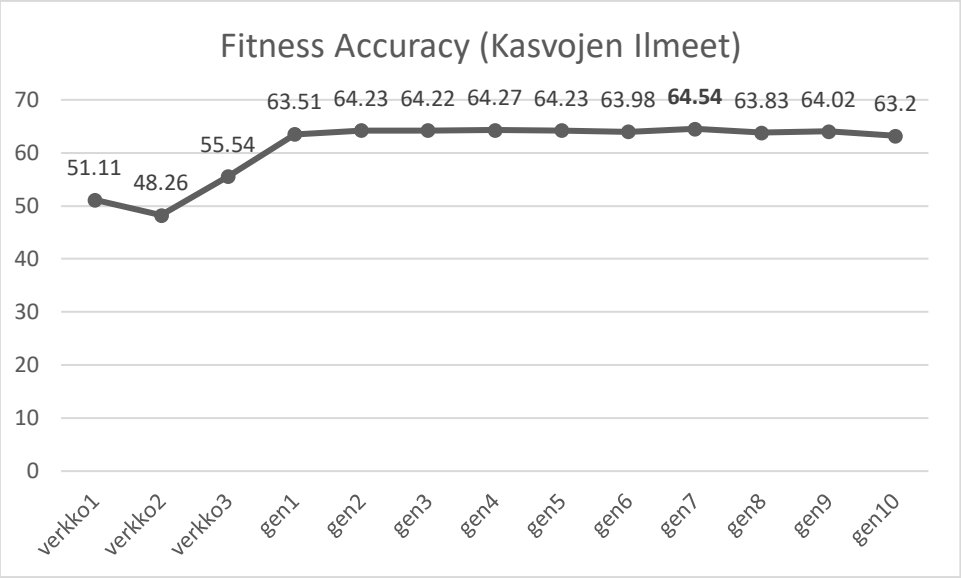
Kuva 20. Kasvojen ilmeiden datalla esivalmennetun aloituspopulaation verkon 1 tarkkuudet.



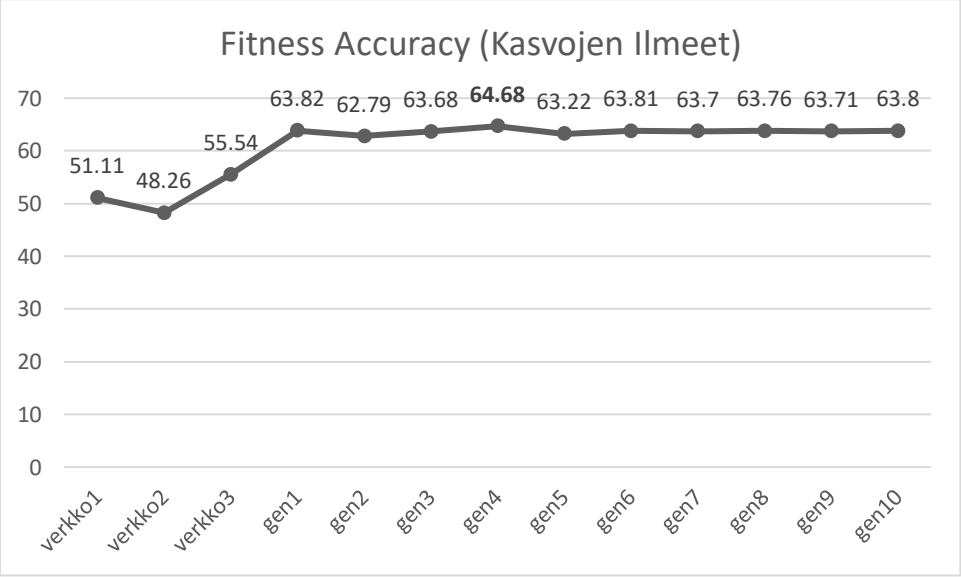
Kuva 21. Kasvojen ilmeiden datalla esivalmennetun aloituspopulaation verkon 2 tarkkuudet.



Kuva 22. Kasvojen ilmeiden datalla esivalmennetun aloituspopulaation verkon 3 tarkkuudet.



Kuva 23. Kasvojen ilmeiden datan evoluutioharjoittelusukupolven validointi tarkkuudet.

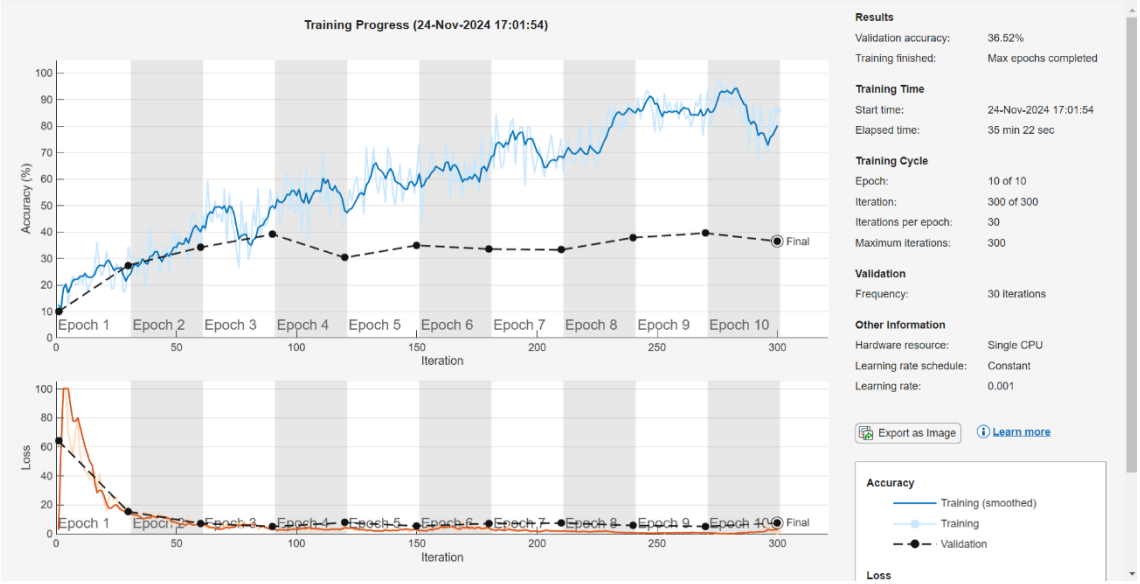


Kuva 24. Kasvojen ilmeiden datan evoluutioharjoittelusukupolven validointi tarkkuudet toistamiseen.

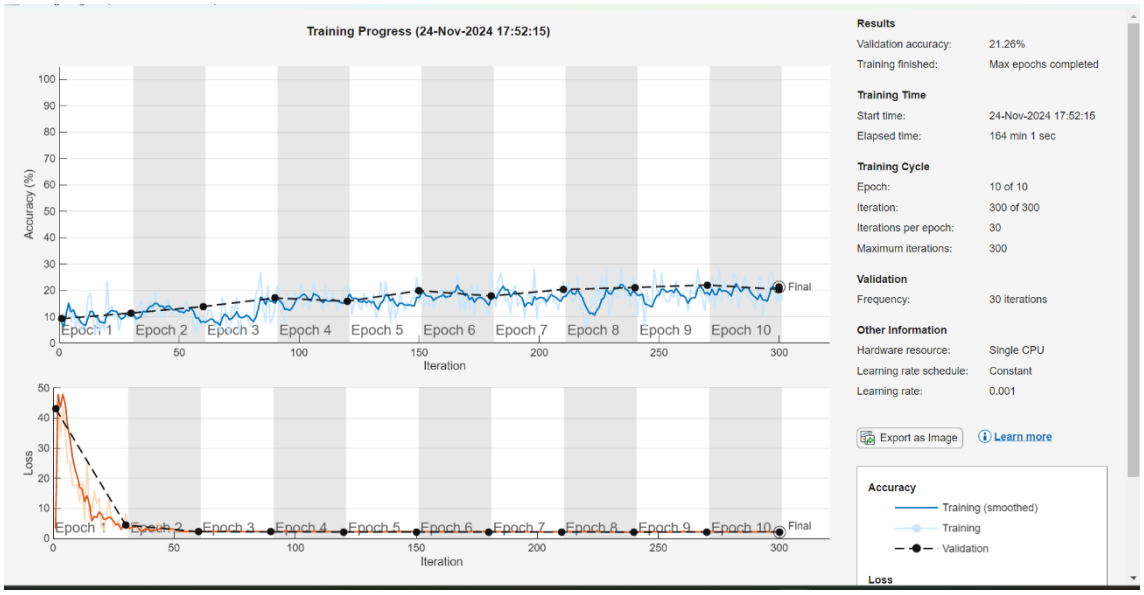
Kasvojen ilmeet -tietokokonaisuuden osalta alkuperäiset esivalmennetut verkot (verkot 1–3) saavuttivat validointitarkkuudet 51 %, 48 % ja 56 % (ks. kuvat 20–22). Ensimmäisen evoluutioharjoittelusukupolven jälkeen validointitarkkuus parani merkittävästi 63–64 prosenttiin, joka pysyi tasaisena seuraavissa sukupolvissa (gen1-gen10), kuten kuvassa

23 näkyy. Parhaaseen tulokseen päästiin seitsemännessä sukupolvessa (gen7). Kuva 23 viittaa siihen, että evoluutioalgoritmi optimoi arkkitehtuurin onnistuneesti prosessin alkuvaiheessa ja että lisämutaatiot eivät tuottaneet huomattavia parannuksia, mikä johtui mahdollisesti konvergenssista tai mallin kapasiteetin rajoituksista. Huomataan myös, että kuvan 24 tulokset suhteellisen samalta kuvan 23 kanssa. Parhaaseen tulokseen päästiin kuitenkin neljännessä sukupolvessa (gen4). Optimointi kesti molemmissa noin 42 tuntia.

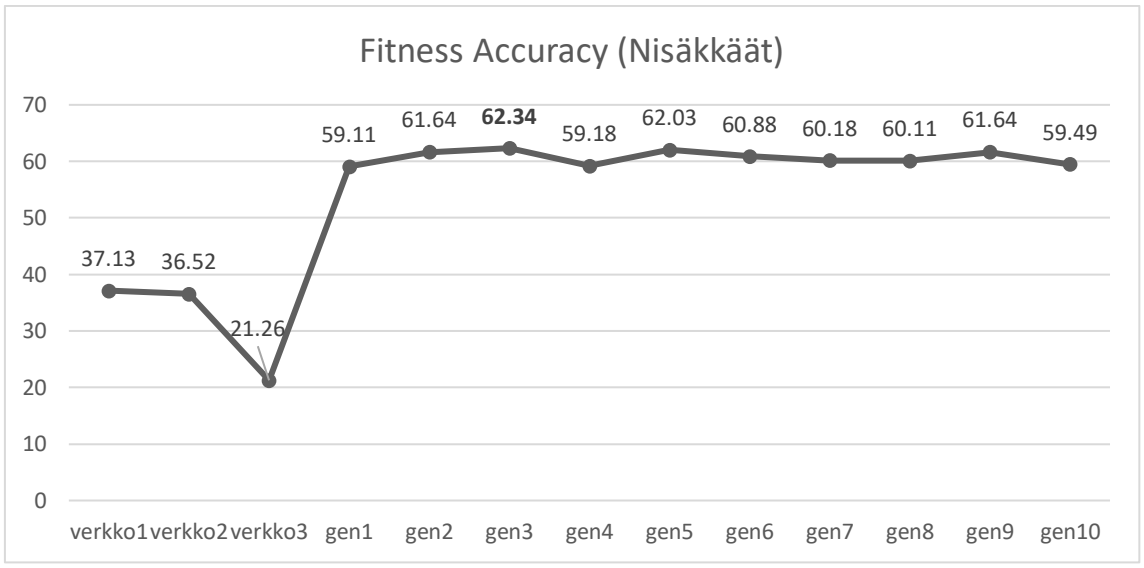
Kuva 25. Nisäkkäiden datalla esivalmennetun aloituspopulaation verkon 1 tarkkuudet.



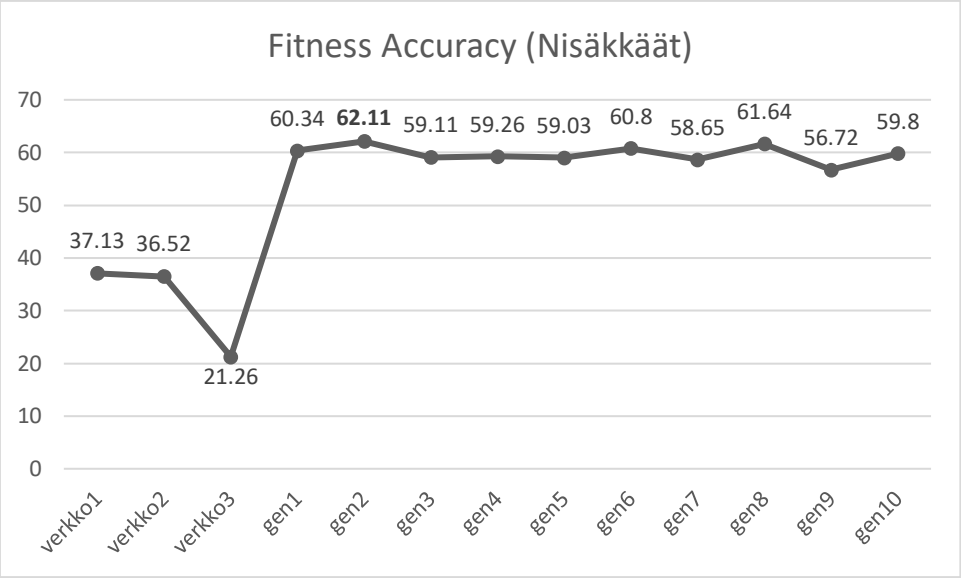
Kuva 26. Nisäkkäiden datalla esivalmennetun aloituspopulaation verkon 2 tarkkuudet.



Kuva 27. Nisäkkäiden datalla esivalmennetun aloituspopulaation verkon 3 tarkkuudet.

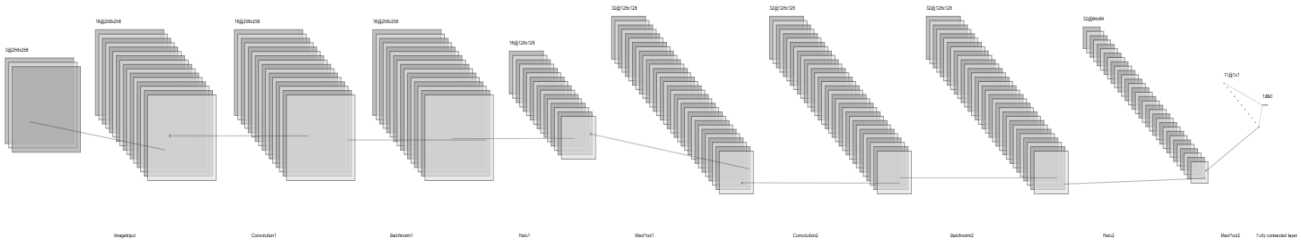


Kuva 28. Nisäkkäät datan evoluutioharjoittelusukupolven validointi tarkkuudet.

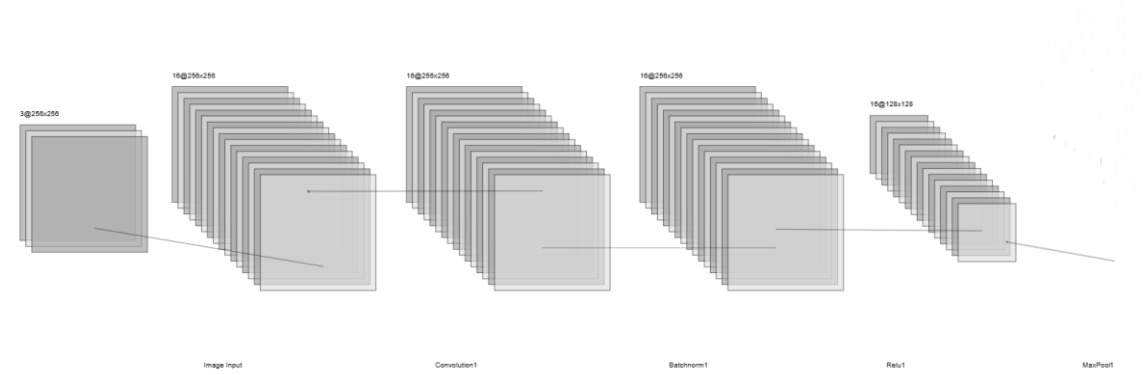


Kuva 29. Nisäkkäät datan evoluutioharjoittelusukupolven validointi tarkkuudet toistamiseen.

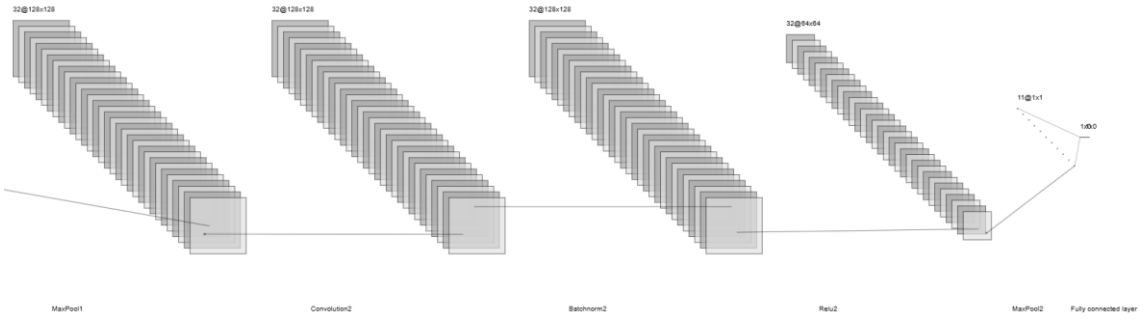
Nisäkkäiden luokittelutietokannan tapauksessa alkuperäiset esivalmennetut verkot saavuttivat 36, 21 ja 37 prosentin validointitarkkuudet (kuvat 25–27). Evoluutioalgoritmin soveltamisen jälkeen sukupolven 3 parhaiten suoriutuva malli saavutti 62,34 prosentin tarkkuuden. Tämä suorituskyky säilyi johdonmukaisesti kaikissa kymmenessä sukupolvessa, mikä viittaa varhaiseen konvergenssiin, kuten kuvasta 28 nähdään. Kuvan 29 tulokset näyttävät suhteellisen samalta kuvan 28 kanssa. Optimointi molemmissa kesti noin 18 tuntia.



Kuva 30. Esimerkki parhaasta arkkitehtuurista nisäkkäiden luokittelussa sukupolvessa 3 LeNet tyylillä (Koko kuva).



Kuva 31. Esimerkki parhaasta arkkitehtuurista nisäkkäiden luokittelussa sukupolvessa 3 LeNet tyylillä (Osa 1).



Kuva 32. Esimerkki parhaasta arkkitehtuurista nisäkkäiden luokittelussa sukupolvessa 3 LeNet tyylillä (Osa 2).

Kuvassa 30 nähdään esimerkki parhaasta arkkitehtuurista nisäkkäiden luokittelussa sukupolvessa 3 LeNet tyylillä. Se muistuttaa verkon 2 arkkitehtuuria kuvassa 18. Kuvassa 31 parhaiten suoriutuva konvoluutioinen neuroverkkoarkkitehtuuri (CNN) nisäkkäiden tietojoukolle alkaa syöttökerroksella, joka käsittelee 256×256 RGB-kuvia käyttäen nollakeskeistä normalisointia. Kuvassa 31 ensimmäinen konvoluutiokerros soveltaa 16 suodatinta, joiden koko on 3×3, stride eli askelpituus [1 1] ja sama padding eli reunatäydennys, minkä jälkeen 16 kanavalle sovelletaan eränormalisointia. Tätä seuraa ReLU-aktivointifunktio ja maksimi-pooling-operaatio 2×2-suodattimella ja askelmitalla [2 2], mikä pienentää spatiaalisia ulottuvuuksia puoleen.

Kuvassa 32 toinen konvoluutiokerros soveltaa 32 suodatinta, joiden koko on 3×3, askelpituus [1 1] ja sama reunatäydennys, minkä jälkeen seuraa jälleen eränormalisointi 32 kanavalla, ReLU-aktivointi ja toinen max pooling -operaatio 2×2-suodattimella ja askelmitalla [2 2]. Tulos välitetään sitten täysin kytkettyyn kerrokseen, jossa on 11 neuronia, mikä vastaa 11 nisäkäsluokkaa tietojoukossa.

Verkko sisältää yhteensä noin 1,45 miljoonaa opittavaa parametria, joista suurin osa sijaitsee täysin kytketyllä kerroksella. Koulutuksen jälkeen tämä arkkitehtuuri saavutti korkeimman hyvyysarvo tarkkuuden kaikista testatuista malleista ja valittiin siksi parhaiten suoriutuvaksi CNN:ksi nisäkkäiden tietojoukkoon.

Aloituspopulaation harjoittelun edistymistä kuvaavat kuvaajat havainnollistavat edelleen harjoittelun ja validoinnin tarkkuuden dynamiikkaa 10 epookin eli ajanjakson aikana kuvissa 21, 22, 23, 25, 26 ja 27. Useimmissa tapauksissa malleissa koulutustarkkuus kasvoi tasaisesti ja tappio pieneni vastaavasti. Jonkin verran ylisovittamista on kuitenkin havaittavissa, erityisesti malleissa, joissa validointitarkkuus ei seurannut koulutustarkkuuden kehitystä. Tästä huolimatta evoluutioalgoritmi säilytti tulosten vakauden ja vältti katastrofaalisen suorituskyvyn heikkenemisen.

Kaiken kaikkiaan tulokset osoittavat, että vaikka evolutiivinen optimointi paransi tehokkaasti mallin suorituskykyä alkusukupolvessa, parannukset vakiintuivat sen jälkeen. Tämä viittaa siihen, että hyperparametrien hienosäätö tai arkkitehtuurin monimuotoisuuden lisääminen saattaa olla tarpeen suorituskyvyn parantamiseksi edelleen tulevissa töissä.

7 Johtopäätökset

Tässä opinnäytetyössä tutkittiin syväoppimisarkkitehtuurien ja evoluutioalgoritmien käyttöä konvoluutiohermoverkkojen (CNN) optimoinnissa kuvien luokittelutehtävissä. Tutkimuksen motiivina oli automaattisten kuvantunnistusjärjestelmien kasvava merkitys eri aloilla, kuten maataloudessa, terveydenhuollossa ja kasvojen ilmeiden analysoinnissa (Kamilaris, 2018). Tutkimuksessa pyrittiin evolutiivisen optimointikehyksen avulla kehittämään CNN-pohjaisia malleja luokittelutarkkuuden parantamiseksi peräkkäisten sukupolvien kautta käyttämällä risteytys- ja mutaatio-operaattoreita arkkitehtuurivariaatioiden tuottamiseksi.

Empiiriset tulokset osoittivat, että esivalmennettujen verkkojen alkuperäinen suorituskyky parani merkittävästi ensimmäisten evoluutiosukupolvien jälkeen. Sekä nisäkkäitä että kasvojen ilmeitä koskevissa tietokokonaisuuksissa hyvyysarvon tarkkuus kasvoi noin 30 prosenttiyksikköä ja 10 prosenttiyksikköä evoluutio-optimoinnin soveltamisen jälkeen. Seuraavat sukupolvet eivät kuitenkaan johtaneet lisäparannuksiin, mikä viittaa konvergenssiin tai rajoituksiin kehittyvän populaation geneettisessä monimuotoisuudessa. Lisäksi mutaatiopohjaiset muutokset tuottivat usein virheellisiä verkkokonfiguraatioita, mikä edellytti varamekanismeja koulutusprosessin ylläpitämiseksi.

Vaikka evoluutioalgoritmi onnistui tunnistamaan parempia verkkokonfiguraatioita alkuvaiheessa, se paljasti myös haasteita, kuten laskennalliset kustannukset, arkkitehtuurin yhteensopivuus ja koulutuksen epävakaus. Nämä havainnot korostavat, että tarvitaan parempaa kontrollia kerroksittaisissa geneettisissä operaatioissa ja älykkäämpiä mutaatiostrategioita, joissa mahdollisesti käytetään rakennetietoisia heuristiikkoja tai vahvistusoppimista ohjattua optimointia varten.

Johtopäätöksenä voidaan todeta, että tämä tutkimus vahvistaa evoluutioalgoritmien potentiaalin neuroverkkoarkkitehtuurin etsinnässä, erityisesti skenaarioissa, joissa aluespesifiset optimoinnit tai manuaalinen viritys eivät ole käytännöllisiä. Tulevassa

tutkimuksessa olisi keskityttävä parantamaan mutaation kestävyyttä, integroimaan validointipohjaisia valintamekanismeja ja soveltamaan tätä kehystä laajempiin ja monimutkaisempiin tietokokonaisuuksiin sen yleistettävyyden ja skaalautuvuuden arvioimiseksi.

Lähteet

Asaniczka. (2022). Mammals image classification dataset (45 animals) [Data set]. Kaggle. https://www.kaggle.com/datasets/asaniczka/mammals-image-classification-dataset-45-animals

Basha, S. M., Rajasekaran, A., & Rao, M. (2024). *A survey of image feature extraction techniques.* International Journal of Scientific Research in Engineering and Technology, 10(3), 143–152. https://ijsret.com/wp-content/uploads/2024/05/IJSRET_V10_issue3_145.pdf

Bishop, C. M. (2006). Pattern recognition and machine learning. Springer.

Chakraborty, S., & Saha, S. (2023). *Deep learning models for digital image processing: A review.* Artificial Intelligence Review, 56(11), 14213–14258. https://doi.org/10.1007/s10462-023-10631-z

Elsken, T., Metzen, J. H., & Hutter, F. (2019). Neural architecture search: A survey. *Journal of Machine Learning Research, 20*(55), 1–21.

Gonzalez, R. C., & Woods, R. E. (2018). Digital image processing (4th ed.). Pearson.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. MIT Press.

Kamilaris, A., & Prenafeta-Boldú, F. X. (2018). *Deep Learning in Agriculture: A Survey.* Computers and Electronics in Agriculture, 147, 70–90.

Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., van der Laak, J. A. W. M., van Ginneken, B., & Sánchez, C. I. (2017). *A survey on deep learning in medical image analysis.* Medical Image Analysis, 42, 60–88. https://doi.org/10.1016/j.media.2017.07.005

Nasteski, V. (2017). An overview of the supervised machine learning methods. Horizons B, 4(1), 51–62. https://doi.org/10.20544/HORIZONS.B.04.1.17.P05

Negnevitsky, M. (2005). Artificial intelligence: A guide to intelligent systems (2nd ed.). Addison-Wesley.

Nixon, M. S., & Aguado, A. S. (2019). Feature extraction and image processing for computer vision (4th ed.). Academic Press.

Nuthakki, S., Buddiga, S. K. P., & [Third author not listed]. (2024, June). Exploring deep learning models for image recognition: A comparative review. ResearchGate. https://www.researchgate.net/publication/374829279

Rashidi, H. H., Tran, N. K., Vali Betts, E., Howell, L. P., & Green, R. (2019). Artificial intelligence and machine learning in pathology: The present landscape of supervised methods. Academic Pathology, 6, 1–12. https://doi.org/10.1177/2374289519873088

Rawat, W., & Wang, Z. (2017). Deep convolutional neural networks for image classification: A comprehensive review. *Neural Computation, 29*(9), 2352–2449. https://doi.org/10.1162/neco_a_00990

Ruan, W., Shi, X., Hu, J., Hou, Y., Fan, M., Wei, X., & Cao, R. (2017). Modeling of malachite green removal from aqueous solutions by nanoscale zerovalent zinc using artificial neural network. Applied Sciences, 7(12), 1325. https://doi.org/10.3390/app7121325

Sambare, M. (2022). FER2013 facial expression recognition dataset [Data set]. Kaggle. https://www.kaggle.com/datasets/msambare/fer2013

Shen, D., Wu, G., & Suk, H.-I. (2017). *Deep learning in medical image analysis.* Annual Review of Biomedical Engineering, 19, 221–248. https://doi.org/10.1146/annurev-bioeng-071516-044442

Sujatha, C., & Selvathi, D. (2021). *A survey of image pre-processing techniques for medical images.* International Journal of Science and Research (IJSR), 10(5), 1845–1850. https://www.researchgate.net/publication/352184973_A_Survey_of_Image_Pre-Processing_Techniques_for_Medical_Images

Sun, Y., Xue, B., Zhang, M., & Yen, G. G. (2020). *Evolving Deep Convolutional Neural Networks for Image Classification.* IEEE Transactions on Evolutionary Computation, 24(2), 394–407. https://doi.org/10.1109/TEVC.2019.2916183

Xie, L., & Yuille, A. (2017). Genetic CNN. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (pp. 1379–1388).

Xu, Y., & Ma, Y. (2023). Evolutionary neural architecture search combining multi-branch ConvNet and improved transformer. Neural Computing and Applications. https://doi.org/10.1007/s00521-023-08544-2

Xue, B., Zhang, M., Browne, W. N., & Yao, X. (2021). *A Survey on Evolutionary Deep Learning: Algorithms and Applications.* IEEE Transactions on Evolutionary Computation, 25(5), 894–912. https://doi.org/10.1109/TEVC.2021.3058210