

Lauri Sintonen

**KEINOTEKOISTEN NEUROVERKKOJEN
HYÖDYNTÄMINEN AUTOMAATTISESSA LINTUJEN
TUNNISTAMISESSA ÄÄNEN PERUSTEELLA**



JYVÄSKYLÄN YLIOPISTO
INFORMAATIOTEKNOLOGIAN TIEDEKUNTA
2018

TIIVISTELMÄ

Sintonen, Lauri

Keinotekkoisten neuroverkkojen hyödyntäminen automaattisessa lintujen tunnistamisessa äänen perusteella

Jyväskylä: Jyväskylän yliopisto, 2018, 54 s.

Tietojärjestelmätiede, kandidaatintutkielma

Ohjaaja(t): Halttunen, Veikko

Tässä tutkielmassa esitellään keinotekkoisten neuroverkkojen hyödyntämistä automaattisessa lintujen tunnistamisessa äänen perusteella. Keskeisenä motiivina ovat sekä neuroverkkoihin tutustuminen että lintujen automaattisen tunnistamisen mahdollisuudet luonnon monimuotoisuuden suojelun, lintuharrastajien ja opetuksen näkökulmasta. Linnut reagoivat ympäristön muutoksiin suhteellisen nopeasti, minkä takia niitä tarkkailemalla voidaan saada tärkeää tietoa luonnonsuojelun näkökulmasta. Lintujen tarkkailu on myös suosittu harrastus, joten myös harrastajat voisivat hyötyä neuroverkkoavusteisesta lintujen tunnistamisesta. Lisäksi Opetushallituksen tavoitteisiin kuuluu, että peruskoulussa opittaisiin muun muassa lajintuntemusta sekä tuntemaan kasvien ja eläinten elämänvaiheita. Neuroverkkoja voisi olla mahdollista hyödyntää myös opetuksessa. Tutkielmassa luodaan ensin yleinen kuva neuroverkoista, minkä jälkeen käsitellään neuroverkkojen hyödyntämistä lintujen tunnistamisessa äänen perusteella. Tutkielmassa käsitellään myös konvoluutioneuroverkkoja, sillä ne ovat olleet viime aikoina keskeisessä asemassa niin äänen- kuin kuvantunnistuksessa ja niitä sovelletaan myös lintujen tunnistamiseen. Tutkielma toteutetaan kirjallisuuskatsauksena, ja lähteinä käytetään pääasiassa neuroverkkojen teoriataustaan liittyviä artikkeleita sekä lähteitä, joissa neuroverkkoja on hyödynnetty automaattisen lintujen äänten tunnistamisen kilpailuissa. Lintujen tunnistaminen neuroverkkojen avulla toimii paremmin kuin muut olemassa olevat teknologiat, mutta niiden käytännön hyödyt ovat rajalliset esimerkiksi suhteellisen raskaan laskennan vuoksi.

Asiasanat: tekoäly, koneoppiminen, ohjattu oppiminen, keinotekoiset neuroverkot, konvoluutioneuroverkot, äänentunnistus, lintujen tunnistus

ABSTRACT

Sintonen, Lauri

Artificial neural networks in automatic bird recognition by their sound

Jyväskylä: University of Jyväskylä, 2018, 54 pp.

Information systems, Bachelor's thesis.

Supervisor(s): Halttunen, Veikko

This thesis covers artificial neural networks and how they are utilized in automatic recognition of bird species based on the sounds they produce. The main motives are both to learn about neural networks and the potential of automatic bird recognition in nature conservation. Birds react rapidly to changes in the environment and observing birds using neural networks could produce useful information for nature conservation. Birdwatching is popular as a hobby and also birdwatchers could benefit from automatic recognition of bird species. In addition, the Finnish National Agency for Education aims to teach children to identify species and to learn about plants and animals in general. Neural networks could be utilized also in teaching. First, neural networks are presented in general, after which the utilization of neural networks in bird recognition is examined. The thesis also covers convolutional neural networks since they have played a key role in image recognition as well as sound recognition and they are also applied in bird recognition. The thesis is conducted as a literature review and the source material is mainly formed of articles about the theoretical background of neural networks, and reports in which neural networks are utilized in competitions of automatic bird recognition by sound. Recognizing birds using neural networks has been more successful than using other technologies but the practical benefits are limited for instance because of relatively intensive needs for computing power.

Keywords: artificial intelligence, machine learning, supervised learning, artificial neural networks, convolutional neural networks, sound recognition, bird recognition

KUVIOT

| | |
|--|----|
| KUVIO 1 Perseptroni, jossa on kolme syötettä ja yksi ulostulo..... | 12 |
| KUVIO 2 Kolmikerroksinen neuroverkko, jossa on vasemmalla syötekerros, keskellä piilokerros ja oikealla ulostulokerros..... | 13 |
| KUVIO 3 Porrassfunktio, jossa on ulostulona on joko 0 tai 1..... | 14 |
| KUVIO 4 Paloittain määritelty funktio, jossa on ulostulona on mitä tahansa lukuja väliltä [-0.5, 0.5]..... | 15 |
| KUVIO 5 Sigmoid-funktio, jossa on ulostulona mitä tahansa lukuja väliltä [0, 1]. Vaihtelu on vähäisintä lähellä rajoja 0 ja 1..... | 15 |
| KUVIO 6 Funktion globaali minimi..... | 18 |
| KUVIO 7 A: optimaalinen globaali minimi. B: lokaali minimi, joka on lähellä optimia. C: lokaali minimi, joka on kaukana optimista..... | 20 |
| KUVIO 8 Satulapiste, jonka naapuripisteet ovat yhtä suuret..... | 20 |
| KUVIO 9 e:n derivaatta a:n suhteen voidaan laskea välimuuttuja c:n avulla..... | 22 |
| KUVIO 10 Ulostulokerroksen virhe (engl. cost) C aktivaatioiden a funktiona...23 | |
| KUVIO 11 Vasemmalla alisovittaminen, jossa dataa yleistetään liikaa. Keskellä hyvin yleistävä funktio. Oikealla ylisovittaminen, jossa tulkitaan turhaan syvä, laaksomainen kuvaaja..... | 25 |
| KUVIO 12 Piirrekarttojen muodostuminen konvoluutiossa. X-kerros vastaa syötekerrosta ja A-kerros piirrekarttoja. F vastaa ulostulokerrosta..... | 28 |
| KUVIO 13 Konvoluutioverkko, jossa X-kerros vastaa syötekerrosta, A-kerros piirrekerrosta, B-kerros yhdistämiskerrosta ja F-kerros ulostulokerrosta..... | 30 |
| KUVIO 14 Vasemmalta oikealle: 28x28 pikselin kokoinen syöte, konvoluutiokerros ja yhdistämiskerros..... | 31 |

TAULUKOT

| | |
|---|----|
| TAULUKKO 1 Automaattisen lintujen äänten tunnistuksen keskitarkkuuksien keskiarvot..... | 37 |
|---|----|

SISÄLLYS

| | |
|---|----|
| TIIVISTELMÄ..... | 2 |
| ABSTRACT..... | 3 |
| KUVIOT..... | 4 |
| TAULUKOT..... | 4 |
| SISÄLLYS..... | 5 |
| 1 JOHDANTO..... | 6 |
| 2 KEINOTEKOISET NEUROVERKOT..... | 11 |
| 2.1 Neuronin ja perseptronin..... | 12 |
| 2.2 Aktivointifunktiot..... | 14 |
| 2.3 Neuroverkon opettaminen..... | 15 |
| 2.3.1 Virhefunktio..... | 16 |
| 2.3.2 Gradienttimenetelmä..... | 17 |
| 2.3.3 Vastavirta-algoritmi..... | 21 |
| 2.3.4 Opettamisen ongelmia koneoppimisessa..... | 24 |
| 2.4 Konvoluutioneuroverkot..... | 26 |
| 2.4.1 Konvoluutio..... | 27 |
| 2.4.2 Yhdistäminen..... | 29 |
| 3 NEUROVERKOT LINTUJEN ÄÄNTEN TUNNISTUKSESSA..... | 32 |
| 3.1 Äänentunnistuksen haasteita..... | 34 |
| 3.2 Vuosien 2016 ja 2017 BirdCLEF-kilpailujen raporttien vertailua..... | 35 |
| 3.2.1 Audiosignaalin käsittely..... | 36 |
| 3.2.2 Datatunnetun kasvattamismenetelmät..... | 36 |
| 3.2.3 Neuroverkkokoarkkitehtuurit..... | 38 |
| 3.2.4 Metadatan hyödyntäminen..... | 39 |
| 3.2.5 Lintujen tunnistuskilpailujen raporttien tuloksia..... | 40 |
| 3.3 Johtopäätökset..... | 42 |
| 4 YHTEENVETO..... | 45 |
| LÄHTEET..... | 48 |

1 JOHDANTO

Tekoäly on ollut viime aikoina paljon keskusteltu aihe. Online-sanakirja Merriam-Webster (2018) määrittelee tekoälyn tietotekniikan tieteenhaaraksi, jossa simuloidaan tietokoneiden älykästä toimintaa. Toisen Merriam-Websterin määritelmän mukaan tekoälyllä tarkoitetaan koneen kykyä imitoida ihmistä. Tieteen termipankin (2018) mukaan tekoäly on "ihmisen teknisillä menetelmillä valmistama älykkääseen toimintaan kykenevä olio". Ottamatta kantaa älykkyyden määritelmään kyse on siis jonkinlaisesta koneen älykkäästä toiminnasta. Tunnettuja koneiden älykkään toiminnan esimerkkejä ovat esimerkiksi itseajavat autot, kuvan- ja puheentunnistus sekä suurten datamäärien analysointi eri tarkoituksia varten (LeCun, Bengio & Hinton, 2015). Tässä tutkielmassa käsitellään tekoälyn osalta keinotekoisia neuroverkkoja, joita tullaan käsittelemään pelkällä neuroverkko-sanalla.

Neuroverkkojen tutkimuksella on takanaan pitkä historia. Esimerkiksi Schmidhuber (2015) näkee, että neuroverkkojen matemaattinen historia ulottuu vähintään Leibnizin (1684) aikakaudelle 1600-luvun loppuun. Goodfellow, Bengio ja Courville (2016) kirjoittavat, että syväoppiminen, jossa neuroverkkoja usein hyödynnetään, on tutkimuksessa esiintynyt ajan mittaan myös muilla nimillä ja sen historia ulottuu ainakin 1940-luvulle. He puhuvat kolmesta syväoppimisen kehityksen aallosta: kybernetiikka 1940–1960-luvulla, konnektionismi 1980–1990-luvulla sekä nykyinen syväoppiminen alkaen vuodesta 2006 (Goodfellow ym., 2016). Nykyään neuroverkot on saatu toimimaan paremmin käytännössä ja niitä sovelletaan yhdessä muiden teknologioiden kanssa monenlaisiin tehtäviin esimerkiksi puheentunnistuksessa (Ting, Yong & Mirhassani, 2013), liikennemerkkien tunnistamisessa (Zhu, Zhang & Lu, 2006), go-lautapelin mullistaneessa AlphaZero-tekoälyssä (Silver ym., 2017) ja automattisessa lintujen äänten tunnistamisessa (Sevilla & Glotin, 2017).

Haykinin (2005) mukaan keinotekoinen neuroverkko on biologisten hermoverkkojen inspiroima laskennallinen malli, joka koostuu keinotekoisista, toisiinsa verkottuneista neuroneista. Neuronit ovat siis perusyksiköitä, jotka esimerkiksi laskevat sisääntulevien arvojen perusteella yhden koko verkkoa edustavan ulostuloarvon. Laskettua ulostuloarvoa verrataan ennalta määriteltyyn raja-arvoon, ja raja-arvon ylittyessä tapahtuu jokin vaste. Vastaavasti jos ulostuloarvo on raja-arvoa pienempi, vaste jää tapahtumatta (Haykin, 2005).

Neuroverkoille on mahdollista opettaa datan tulkintaa syöte–tavoitepareilla niin, että ne oppivat tunnistamaan esimerkiksi niille syötettyä puhetta (Ting ym., 2013). Toisin sanoen neuroverkko oppii vertaamalla löytämiään tuloksia ennalta määriteltyyn tavoitteeseen ja tämän vertailun perusteella neuroverkko voi oppia parantamaan suoritustaan.

Neuroverkkoja voidaan soveltaa puheen lisäksi myös esimerkiksi lintujen äänten tunnistamiseen, sillä lintujen äänet ja ihmisen puhe ovat tietokoneelle vain samanarvoista ääntä. Esimerkiksi Sprengel, Jaggi, Kilcher ja Hofmann (2016), Martinsson (2017) sekä Sevilla ja Glotin (2017) ovat hyödyntäneet neuroverkkoja lintujen tunnistamisessa puheentunnistuksen keinoin. Seuraavaksi perustellaan miksi lintujen tunnistaminen ylipäätään voi olla tärkeää, ja että miksi lintujen tehokkaalle tunnistamiselle on kysyntää.

Maailmalla on meneillään globaali biodiversiteettikriisi (Thomas ym., 2004). Tämä tarkoittaa sitä, että maailmasta katoaa eliölajeja entistä nopeammin ja ihmistoiminnan vuoksi sukupuuttovauhti on kolminkertaistunut viimeisen sadan vuoden aikana (Ricketts ym., 2005). Myös ihmiset ovat riippuvaisia ympäröivästä luonnosta ja sen tarjoamista ekosysteemipalveluista, millä tarkoitetaan Sodhin ja Ehrlichin (2011) mukaan esimerkiksi luonnossa ilmaiseksi saatavilla olevia "palveluita" kuten makeaa vettä, puuta sekä polttoainetta. Myös he toteavat, että meneillään olevan kuudennen massasukupuuton syyt, kuten ilmastonmuutos, elinympäristöjen häviäminen ja pirstoutuminen tapahtuvat suurelta osin ihmistoiminnan vaikutuksesta (Ehrlich, 2011).

Biodiversiteetin eri tasoja suojeltaessa yksi lähestymistapa on tarkkailla indikaattorilajeja, ja monet lintulajit ovat sellaisia (IUCN, 2018). Indikaattorilajilla tarkoitetaan eliölajia "joka on erityisen herkkä jollekin ympäristötekijälle, jolloin lajin esiintyminen osoittaa kyseisen tekijän" (Tirri ym., 2006). Neuroverkkoavusteinen lintujen tunnistaminen äänen perusteella voisi toimia apuna indikaattorilajien tunnistamisessa ja seuraamisessa ja näin sillä olisi osuutensa myös biodiversiteetin suojelussa.

Luonnonsuojelullisten tavoitteiden lisäksi lintuja voi olla paikallaan tuntea paremmin myös muista syistä. Opetushallituksen (2014) tavoitteisiin on kirjattu se, että peruskoulussa opittaisiin muun muassa lajintuntemusta sekä tuntemaan

kasvien ja eläinten elämänvaiheita. Tämän ohella tavoitteena on myös opettaa lapsille ja nuorille elottoman ja elollisen luonnon peruspiirteet. Kuitenkin Relanderin (2013) mukaan sekä suomalaisten alakoululaisten että luokanopettajaopiskelijoiden lajintuntemustaidot ovat tyydyttävällä tai heikolla tasolla. Neuroverkkoavusteista lintujen ääniä tunnistavaa sovellusta voisi olla mahdollista hyödyntää niin lintulajien opiskelussa kuin opettamisessakin.

Edellä mainittujen seikkojen lisäksi maailman linnusto mahdollistaa varsin suosittuun harrastusmahdollisuuden lukemattomille ihmisille, jotka haluavat oppia tunnistamaan lintuja. Esimerkiksi 30 lintuyhdistyksen muodostamassa BirdLife Suomessa on yli 13 000 jäsentä (BirdLife Suomi ry, 2018a). Tästä luvusta jäävät tilastojen ulkopuolelle kaikki ne harrastajat, jotka eivät ole rekisteröityneet mihinkään jäsenyhdistykseen. Rekisteröitymättömiä lintuharrastajia lienee Suomessa moninkertaisesti BirdLife Suomen jäsenmäärään verrattuna. Lintuharrastus on suosittua myös muualla maailmassa. Vuonna 2016 Yhdysvalloissa oli Statistan (2016) mukaan noin 11,5 miljoonaa lintuharrastajaa eli noin 3,5 prosenttia koko Yhdysvaltain väestöstä. Näin ollen myös tämän takia neuroverkkoavusteiselle lintujen tunnistamiselle lienee paljonkin kysyntää, sillä neuroverkot voisivat mahdollisesti tukea niin tutkimusta, luonnonsuojelua kuin lintuharrastustakin. Lisäksi lintujen äänet toimivat esimerkkitapauksena äänentunnistuksen tutkimuksessa ylipäätään.

Tutkielman tutkimusongelma on: *miten neuroverkkoja voidaan hyödyntää lintujen äänten tunnistamisessa?* Tutkielman aiheen käsittely jakautuu kahteen osaan. Ensimmäinen osa tutkielmasta rajautuu keinotekoisiiin neuroverkkoihin, jotka sijoittuvat hierarkiassa tekoälyn alle koneoppimisen osa-alueelle (Nielsen, 2015). Siinä pohjustetaan tutkielman kannalta relevantti neuroverkkoja koskeva teoreettinen viitekehys, joka luonnollisesti on neuroverkkoavusteisen lintujen äänten tunnistamisen taustalla. Toisessa osassa tutustutaan esimerkkien avulla siihen, miten neuroverkkoja voidaan hyödyntää lintujen äänten tunnistamisessa.

Tutkielma toteutetaan kirjallisuuskatsauksena. Lähteitä on haettu pääasiassa Jyväskylän yliopiston JYKDOK–finna-hakupalvelusta, Google Scholarista sekä Web of Sciencesta. Aineistoa kerätessä on pyritty valitsemaan sellaisia lähteitä, joilla on luokitus Tieteellisten seurain valtuuskunnan Julkaisufoorumissa (Julkaisufoorumi, 2017). Neuroverkkojen osalta käsitellään pääasiassa koneoppimiseen, syväoppimiseen ja neuroverkkoihin liittyvää aineistoa. Koska osa neuroverkkoihin liittyvistä käsitteistä on luonteeltaan matemaattisia, käytetään myös lähteitä, joiden avulla kyseiset käsitteet voidaan määritellä. Neuroverkkoihin liittyviä lähteitä on haettu pääasiassa hakusanoilla "deep learning" (engl. syväoppiminen), "machine learning" (engl. koneoppiminen) ja "artificial neural networks" (engl. keinotekoiset neuroverkot).

Konvoluutioneuroverkkojen osalta aineistona käytetään osittain samoja lähteitä kuin neuroverkkojen yleisessä esittelyssä, mutta myös spesifejä, konvoluutioneuroverkkoihin liittyviä lähteitä. Konvoluutioneuroverkkojen lisäaineistoksi etsittiin lähteitä hakusanalla "convolutional neural network" (engl. konvoluutioneuroverkko).

Toisessa osassa käytetyt lähteet ovat peräisin automaattisen lintujen äänten tunnistamisen kilpailuista nimeltä BirdCLEF (Goëau, Glotin, Vellinga, Planqué, Rauber & Joly, 2015; Goëau, Glotin, Vellinga, Planqué, Rauber & Joly, 2014; Goëau, Glotin, Vellinga, Planqué & Joly, 2016; Goëau, Glotin, Vellinga, Planqué & Joly, 2017). Niissä kilpaillaan siitä, kenen automaattinen ratkaisu onnistuu tunnistamaan parhaiten eri lintulajeja niiden äänten perusteella. Kilpailuista käsitellään viime vuosien parhaiten pärjänneitä ratkaisuja, ja niissä kaikissa on hyödynnetty neuroverkkoja lintujen äänten tunnistamisessa. Aineistoon kuuluu vertailun vuoksi myös muita kuin neuroverkkopohjaisia automaattiseen lintujen äänten tunnistukseen liittyviä lähteitä. Lintujentunnistusta koskevia lähteitä on etsitty niin ikään pääasiassa JYKDOK-finna-hakupalvelusta, Google Scholarista sekä Web of Sciencesta. Hyväksi lähdemateriaaliksi osoittautui myös Seppo Fagerlundin (2014) automaattista lintujen tunnistusta koskeva väitöskirja, jonka kautta löytyi kattavasti aiheeseen liittyvää kirjallisuutta.

Tutkielma koostuu johdannosta, kahdesta sisältöluvusta sekä yhteenvedosta. Ensimmäisessä sisältöluvussa tarkastellaan neuroverkkoja. Sen alaluvuissa käsitellään neuroverkkoja sen rakenteeseen ja toimintaan liittyvien käsitteiden avulla. Ensimmäisessä alaluvussa esitellään neuronin ja perseptronin sekä paino, vakiotermi ja neuronin tulos. Lisäksi esitellään neuroverkon kerrokset. Toisessa alaluvussa käsitellään neuroverkon neuronien aktivoitumiskäytännöitä. Kolmannessa alaluvussa tarkastellaan neuroverkkojen opettamista ja sen alaluvuissa esitellään siihen liittyvät käsitteet. Näitä käsitteitä ovat virhefunktio, gradienttimenetelmä ja vastavirta-algoritmi. Kolmannen alaluvun lopussa käsitellään lyhyesti joitakin neuroverkon opettamiseen liittyviä ongelmia, joita ovat ali- ja ylisovittaminen sekä neuroverkkojen laskennalliseen raskauteen liittyvät ongelmat. Neljännessä alaluvussa esitellään lintujen äänten tunnistamisen kannalta olennainen neuroverkkotyyppi, konvoluutioneuroverkot. Siihen liittyvistä käsitteistä käsitellään omilla alaluvuillaan konvoluutio (engl. convolution) sekä yhdistäminen (engl. pooling).

Toisessa sisältöluvussa käsitellään sitä, miten neuroverkkoja on hyödynnetty automaattisessa lintujen äänten tunnistamisessa käytännössä. Sen ensimmäisessä alaluvussa esitellään johdatuksena automaattisen lintujen äänten tunnistamisen kannalta relevantteja äänentunnistuksen haasteita. Toisessa alaluvussa tarkastellaan esimerkkien avulla sitä, miten neuroverkkoja

on hyödynnetty lintujen äänen tunnistamisessa. Esimerkkien aineistona käytetään parhaiten menestyneiden osallistujien raportteja automaattisen lintujen äänen tunnistamisen BirdCLEF-nimisistä kilpailuista. Toisen alaluvun alaluvuissa esitellään tarkasteltavien raporttien osalta audiosignaalin käsittelyä, datan kasvattamismenetelmiä, neuroverkko, metadatan hyödyntämistä sekä lintujentunnistuskilpailujen tuloksia. Kolmannessa alaluvussa käsitellään luvussa syntyneitä johtopäätöksiä.

Viimeisessä luvussa, yhteenvedossa, koostetaan yhteen tutkielmassa käsiteltyjä asioita ja kerrataan aiempien lukujen johtopäätöksiä. Yhteenvedon lopussa esitetään joitakin jatkotutkimustarpeita.

2 KEINOTEKOISET NEUROVERKOT

Tässä luvussa käsitellään neuroverkkoja ja esitellään olennaisimmat niihin liittyvät käsitteet kuten neuronin ja perseptronin sekä painon, vakiotermin ja neuronin tulos. Tämän jälkeen esitellään neuroverkon kerrokset. Lisäksi käsitellään aktivointifunktioita ja neuroverkon opettamista, joista jälkimmäisen yhteydessä avataan läheisiä käsitteitä kuten virhefunktioita, gradienttimenetelmää sekä vastavirta-algoritmia. Sitten esitellään keskeisiä koneoppimiseen liittyviä ongelmia. Esiteltäviä ongelmia ovat neuroverkkojen opettamiseen liittyvät ongelmat: ali- ja ylisovittaminen, sekä neuroverkkojen laskennallisen raskauden tuomat ongelmat, jotka rajoittavat käytännön hyödyntämistä esimerkiksi älypuhelimissa. Lopuksi esitellään automaattisen lintujen äänten tunnistamisen kannalta olennainen neuroverkkotyyppi, konvoluutioneuroverkot.

Tämän luvun aineistona käytetään pääasiassa neuroverkkoihin liittyvää aineistoa. Koska osa neuroverkkoihin liittyvistä käsitteistä on luonteeltaan matemaattisia, käytetään tarpeen mukaan myös lähteitä, joiden avulla kyseiset käsitteet voidaan käsitellä. Konvoluutioneuroverkkojen osalta aineistona käytetään osittain samoja lähteitä kuin neuroverkkojen yleisessä esittelyssä, mutta myös spesifejä, konvoluutioneuroverkkoihin keskittyviä lähteitä.

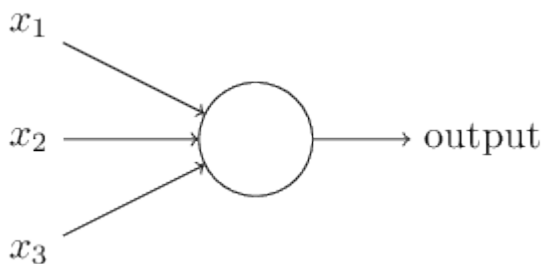
Tässä luvussa rajataan ja esitellään tässä tutkielmassa käytetty neuroverkkoja koskeva teoreettinen viitekehys. Esitellyt asiat luovat pohjan kolmannelle luvulle, jossa esitellään neuroverkkojen hyödyntämistä automaattisessa lintujen äänten tunnistamisessa sekä tarkastellaan aihetta esimerkkien avulla. Koska tässä luvussa käsitellyt asiat eivät liity ainoastaan lintujen tunnistamiseen vaan yleisesti neuroverkkoihin, niitä ei tässä luvussa liitetä erikseen lintujen äänten tunnistamiseen, vaan huomio keskittyy neuroverkkoihin. Kaikki tässä luvussa esitellyt asiat taustoittavat kolmatta

lukua, jossa käsitellään neuroverkkojen hyödyntämistä automaattisessa lintujen äänen tunnistamisessa.

2.1 Neuron ja perseptroni

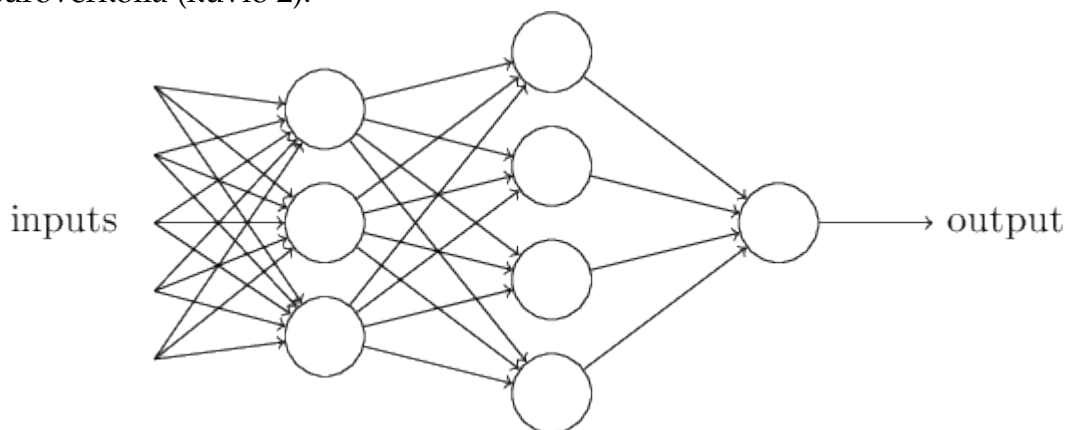
Yleinen käsitys neuroverkoista voidaan luoda niin, että kuvaillaan ensiksi niitä yksinkertaisia osia, joista neuroverkko koostuu: neuroneita ja perseptroneita. Keinotekoiset neuroverkot kehittyivät ennen varsinaista toteutusta aivojen toiminnan inspiroimaksi malliksi. Aluksi muun muassa McCulloch ja Pitts (1943) mallinsivat propositio- eli lauselogiikan funktioita käyttäen neuronin käsitettä, jota sekä Goodfellow ym. (2016) että Haykin (2005) kutsuvat McCulloch–Pitts-neuroniksi. Heidän näkemyksensä oli, että jos hermoimpulssien katsotaan toimivan binäärisellä joko tai -logiikalla, aivoja voisi mallintaa totta–valetta-ajatuksella toimivan propositiologiikan näkökulmasta.

Kehitellessään mallia siitä, miten aivot varastoivat ja järjestelevät informaatiota aivoissa, Rosenblatt (1958) sovelsi aiempaa ajatusta neuronista ja esitteli perseptronin eli sellaisen neuronin, joka esittää yleistävästi hypoteettista hermostoa tai järjestelmää yrittämättä mallintaa minkään tietyn eliölajin hermostoa. Ajatuksen mukaan perseptroniin tulevien ärsykkeiden vaikutuksesta aktivoituu jokin vaste samaan tapaan kuin eliölajien hermosoluissa. Perseptronin ja McCulloch–Pitts-neuronin ero on se, että Rosenblatt käytti perseptronissa painon käsitettä. Hänen mukaansa perseptronin impulsseja eli sisääntuloja voidaan painottaa niin, että toisilla sisääntuloilla on toisia enemmän vaikutusta vasteeseen eli ulostuloon. Sisääntulojen yhteenlaskettuja painoja eli painotettua summaa verrataan perseptronille annettuun vakiotermiin eli raja-arvoon, joka määrittää sen, miten kevyesti vaste aktivoituu perseptronissa (Rosenblatt, 1958). Nielsen (2015) kuvaa perseptronia yksinkertaisella mallilla, jossa kolmesta sisääntulosta aktivoituu yksi ulostulo (kuvio 1).



KUVIO 1 Perseptroni, jossa on kolme syötettä ja yksi ulostulo (Nielsen, 2015)

Nielsen (2015) käyttää esimerkkinä yksinkertaista neuroverkkoa, joka koostuu kolmesta kerroksesta: 1) syötekerros, 2) piilokerros, jossa voi olla monta kerrosta ja 3) ulostulokerros. Kukin kerros koostuu perseptroneista, joita voi olla monta rinnakkain tai vain yksi. Kukin perseptroni ottaa vastaan dataa ja voi antaa kyseistä dataa vastaavaksi ulostuloksi esimerkiksi 0 tai 1. Kaikki edellisen kerroksen perseptronien binääriset ulostulot ovat seuraavan kerroksen jokaisen perseptronin sisääntuloja, joten aiemmin verkossa sijaitsevat perseptronit vaikuttavat kaikkiin myöhemmin verkossa sijaitseviin perseptroneihin. Lopulta ulostulokerrokseen – tässä esimerkissä yhteen perseptroniin – sisääntulevien binääristen signaalien perusteella muodostuu koko neuroverkkoa edustava binäärinen ulostulo eli tulos 1 tai 0. Siihen, miten koko neuroverkon lopullinen ulostulo määräytyy, vaikuttavat sisääntulojen painotus ja perseptronien vakiotermit eli raja-arvo. Kullekin sisääntulolle määritetään paino, eli luku, joka määrittää sisääntulon painoarvon eli sen, kuinka suuri vaikutus sillä on perseptronin ulostuloon. Suurempi painoarvo vastaa suurempaa vaikutusta ulostuloon. Perseptroneille määritetään myös raja-arvo, eli luku, joka antaa ulostuloksi 1, jos sisääntulojen yhteenlasketut painot eli painotettu summa on suurempi kuin kyseinen raja-arvo. Mitä pienempi luku raja-arvo on, sitä todennäköisemmin neuroverkon ulostuloksi muodostuu 1. Esimerkiksi jos kolmen sisääntulon painoarvot ovat 1, 2 ja 3, painotettu summa on $1+2+3$ eli 6. Raja-arvo voi olla esimerkiksi 5, jolloin painotettu summa 6 on sitä suurempi, ja neuroverkon ulostulo on täten 1. Jos raja-arvoksi on asetettu 7, painotettu summa on raja-arvoa pienempi, minkä seurauksena neuroverkko antaa ulostuloksi 0 (Nielsen, 2015). Nielsen havainnollistaa käsitystä neuroverkosta yksinkertaisella kolmen kerroksen neuroverkolla (kuvio 2):

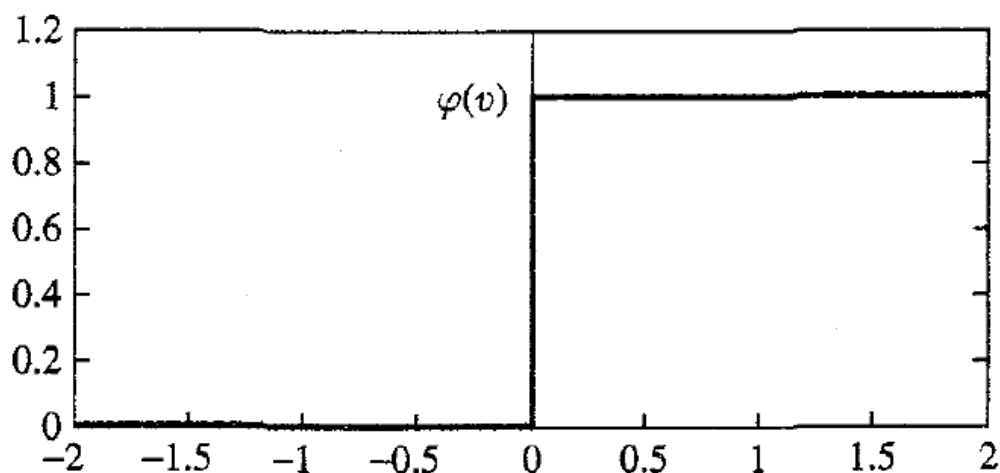


KUVIO 2 Kolmikerroksinen neuroverkko, jossa on vasemmalla syötekerros, keskellä piilokerros ja oikealla ulostulokerros (Nielsen, 2015)

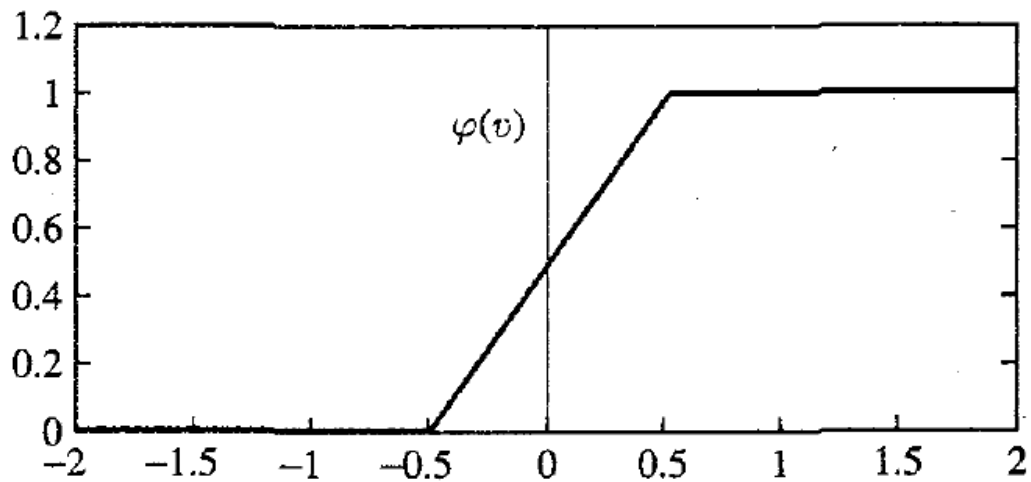
2.2 Aktivointifunktiot

Edellisissä alaluvuissa on todettu, että neuroverkon neuronit laskevat sisääntuloarvojen perusteella ulostuloarvon. Tähän mennessä tutkielmassa on esitetty, että neuroverkon laskema ulostuloarvo voi olla binäärinen 0 tai 1. On myös mahdollista, että laskettava arvo on jotakin muuta. Se, minkälainen neuronin laskema ulostuloarvo on, määräytyy neuronin aktivointifunktion perusteella.

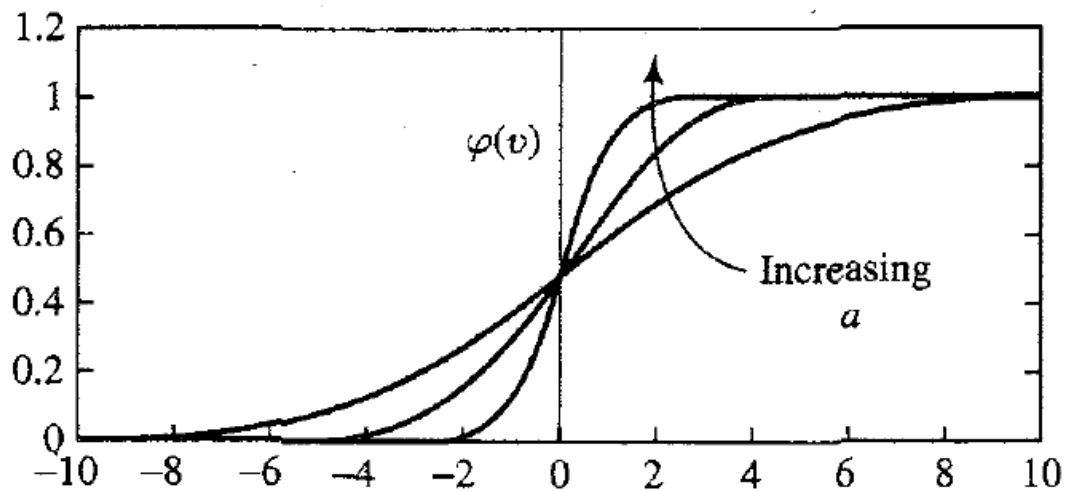
Haykinin (2005) mukaan aktivointifunktion avulla saadaan rajattua ulostulon arvojoukko tietylle välille ja se lasketaan ennen neuronin ulostuloarvon lähettämistä seuraavalle neuronille. Haykin esittelee esimerkkinä kolme aktivointifunktiota, joita voidaan hyödyntää tapauskohtaisesti tavoitteesta riippuen. Esimerkiksi jos neuronin ulostulo on binäärinen, aktivointifunktiona on porraskfunktio, jossa mahdolliset ulostuloarvot ovat joko 0 tai 1 (kuvio 3). Aktivointifunktiona voidaan käyttää myös jonkinlaista paloittain määriteltyä funktiota, jolla voidaan laskea ulostuloarvoiksi mitä tahansa reaalilukuja väliltä $[-0.5, 0.5]$ (kuvio 4). Haykin esittelee vielä neuroverkoissa ehkä yleisimmin käytetyn sigmoid-funktion (kuvio 5), jolla lasketaan ulostuloarvoksi mikä tahansa reaaliluku väliltä $[0, 1]$ (Haykin, 2005). Nielsen (2015) lisää, että neuronina, jonka aktivointifunktiona käytetään sigmoid-funktiota, kutsutaan yleisesti sigmoid-neuroniksi. Lisäksi hän huomauttaa, että sigmoid-funktiossa arvojoukon puolivälin tienoilla vaihtelua on eniten ja lähellä rajoja 0 ja 1 vaihtelu on vähäisempää. Tästä ominaisuudesta on hyötyä verkkojen opettamisessa, jota käsitellään seuraavassa alaluvussa.



KUVIO 3 Porraskfunktio, jossa on ulostulona on joko 0 tai 1 (Haykin, 2005)



KUVIO 4 Paloittain määritelty funktio, jossa on ulostulona on mitä tahansa lukuja väliltä $[-0.5, 0.5]$ (Haykin, 2005)



KUVIO 5 Sigmoid-funktio, jossa on ulostulona mitä tahansa lukuja väliltä $[0, 1]$. Vaihtelu on vähäisintä lähellä rajoja 0 ja 1 (Haykin, 2005)

2.3 Neuroverkon opettaminen

Jotta neuroverkko saadaan toteuttamaan sille annettua tehtävää mahdollisimman hyvin, sen tulee pystyä parantamaan tulostaan (Haykin, 2005). Tämä voidaan saavuttaa opettamalla neuroverkkoa. Opettaminen on käytännössä neuroverkon neuronien painoarvojen ja raja-arvojen säätämistä

niin, että koko neuroverkon tulos paranee (Goodfellow ym., 2016). Kun neuroverkkoa opetetaan, se voi oppia esimerkiksi tunnistamaan paremmin sille syötettyä puhetta (Ting, Yong & Mirhassani, 2013) tai lintujen ääniä (Sevilla & Glotin, 2017; Sprengel ym., 2016).

Goodfellow ym. (2016) jakavat koneoppimisen opetusalgoritmit karkeasti kahteen osaan, ohjaamattomaan oppimiseen ja ohjattuun oppimiseen. Heidän mukaansa ohjaamattomassa oppimisessa koneelle syötetään dataa, josta se pyrkii ilman erillisiä ohjeita ryhmittelemään syötteitä samankaltaisuuksien perusteella. Esimerkiksi kone voi pystyä luokittelemaan dataa sosiaalisista verkostoista niin, että verkoston hierarkiat tai suhteet saadaan selville huomattavasti tehokkaammin kuin mihin ihminen pystyisi (Newman & Girvan, 2004). Tämän tutkielman kannalta tärkeämpi ja myös lintujen ääniä tunnistavissa neuroverkoissa käytetty koneoppimisen opetusalgoritmien kategoria on ohjattu oppiminen. Ohjatussa oppimisessa koneelle annetaan syötetyn opetusdatan lisäksi tavoite, johon neuroverkon tulosta verrataan. Nielsen (2015) käyttää esimerkkinä sitä, miten neuroverkko oppii tunnistamaan käsinkirjoitettuja numeroita. Kun neuroverkolle syötetään esimerkiksi kuva numerosta 0, mukana annetaan myös tieto siitä, että kyseessä on numero 0. Kun koko opetusdatan mukana on syötetty riittävän monta kuva-tavoite-paria, neuroverkko oppii lopulta tunnistamaan käsinkirjoitettuja numeroita itse ilman, että sille tarvitsee enää syöttää kuvan mukana tietoa siitä, mikä numero milloinkin on kyseessä. Koneen oppimisprosessi ohjatussa oppimisessa perustuu Haykinin (2005) mukaan esimerkiksi virheen minimoimiseen, ja hän kutsuu kyseistä prosessia virheenkorjausoppimiseksi.

2.3.1 Virhefunktio

Haykin (2005) kirjoittaa, että neuroverkon opettamisessa tarvitaan tietoa siitä, kuinka suuri neuroverkon virhe on. Hän jatkaa, että virheluku voidaan laskea vertaamalla neuroverkon tulosta ennalta annettuun tavoitteeseen. Kun virheluku on laskettu, pyritään selvittämään, miten neuroverkon painoja ja raja-arvoja tulee vähän kerrassaan säätää niin, että virheluku pienenee. Tähän pyritään, sillä mitä pienempi virhe on, sitä paremmin neuroverkko suoriutuu tehtävästään. Haykin huomauttaa, että virheenkorjausoppiminen ei ole ainoa tapa opettaa konetta ohjatusti, mutta virheen korjaamiseen perustuva oppiminen on tämän tutkielman kannalta oleellinen opetustapa (Haykin, 2005).

Neuroverkon virheluku voidaan laskea jollakin virhefunktiolla, jossa otetaan huomioon kaikki neuroverkon painot ja raja-arvot. Sekä Goodfellow ym. (2016), Schmidhuber (2015), Haykin (2005) että Nielsen (2015) käyttävät

esimerkkivirhefunktiona keskineliösummaa, jonka kaava on Nielsenin kirjoittamana:

$$C(w, b) \equiv \frac{1}{2n} \sum_x \|y(x) - a\|^2$$

Nielsen avaa, että kaavassa kirjain w vastaa kaikkia neuroverkon painoja, b kaikkia neuroverkon raja-arvoja, n opetusdatan sisäänmenojen yhteenlaskettua määrää ja a kaikkien ulostulojen eli aktivointifunktion tuottamien aktivaatioiden vektoria kun sisäänmeno on x . Sigma-kirjain (Σ), jonka alla on x , vastaa kaikkien sisäänmenojen summaa. Funktio $y(x)$ vastaa neuroverkolle annettua tavoitetta, josta tulosvektori a vähennetään. Kullakin neuroverkon neuronilla on oma virhelukunsa ja koko neuroverkon virhe koostuu kaikkien neuroneiden yhteenlasketuista virheistä. Jotta neuroverkon tulos paranisi, tavoitteena on saada muutettua painoja ja raja-arvoja niin, että virhefunktio saataisiin mahdollisimman pieneksi, eli $C(w, b) \approx 0$. Yksi virhefunktion minimoimisen keino on gradienttimenetelmä, jota käsitellään seuraavassa alaluvussa.

2.3.2 Gradienttimenetelmä

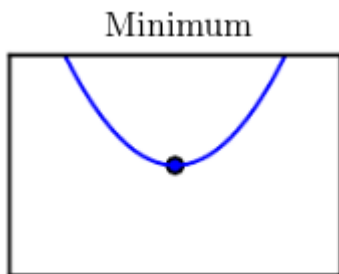
Goodfellow ym. (2016) toteavat, että useimpiin syväoppimisalgoritmeihin sisältyy haluttujen tavoitteiden saavuttamiseksi jonkinlaista optimointia. Hänen mukaansa optimoinnissa joko minimoidaan tai maksimoidaan jokin funktio $f(x)$ muuttamalla sen parametreja x . Hän jatkaa, että neuroverkkojen minimoinnin tapauksessa funktiota voidaan lähteestä riippuen kutsua joko tappiofunktioksi (engl. loss function), maksufunktioksi (engl. cost function) tai virhefunktioksi (engl. error function) (Goodfellow ym., 2016). Tässä tutkielmassa käytetään edellä mainituista käsitteistä viimeistä, virhefunktioita.

Gradienttimenetelmästä (engl. gradient descent) puhuttaessa on tarpeen avata itse gradientin käsitettä. Pitkäranta (2015) pitää gradienttia monen muuttujan "yleistettynä derivaattana". Hän tarkentaa, että derivaatta voidaan tulkita esimerkiksi fysiikan sovelluksissa funktion hetkelliseksi muutosnopeudeksi. Toisin sanoen esimerkiksi yhden muuttujan funktion tapauksessa derivaatta on funktion tangentin kulmakerroin, eli sen avulla saadaan selville paljonko funktion $f(x)$ arvo muuttuu, kun x :n arvoa muutetaan. Siinä missä yhden muuttujan tapauksessa käytetään derivaattaa, gradienttia käytetään kun muuttujia on monta. Gradientin yhteydessä käytetään tavallisen derivaatan sijaan muuttujien osittaisderivaattoja, eli funktion derivaattoja

kunkin muuttujan suhteen muiden muuttujien pysyessä vakioina (Pitkäranta, 2015).

Pitkäranta (2015) jatkaa, että gradientin avulla voidaan laskea, miten funktion arvot muuttuvat kun pisteestä x siirrytään jonkin yksikkövektorin osoittamaan suuntaan. Samalla gradientti osoittaa myös, miten funktion minimoiminen on mahdollista. Pitkäranta nimittäin tiivistää, että "Funktio kasvaa voimakkaimmin gradientin suuntaan, vähenee voimakkaimmin negatiivisen gradientin suuntaan --" (Pitkäranta, 2015, s. 775). Gradientin vastavektori osoittaa siis suunnan, jossa funktio vähenee voimakkaimmin. Neuroverkkoja opettaessa minimoitava funktio on virhefunktio ja sen minimoimiseksi käytetään usein gradienttimenetelmää, jossa hyödynnetään tietoa siitä, että funktio vähenee gradientin vastavektorin suuntaan (Goodfellow ym., 2016).

Edellä esiteltyä gradientin kuvailua selkeyttää Nielsenin (2015) esittelemä analogia, jossa minimoitava funktio kuvitellaan laaksoksi. Laakson reunalle kuvitellaan pallo, joka valuu vähän kerrassaan pitkin laakson seinämää ja päätyy lopulta laakson pohjalle. Laakson pohjalla on funktion globaali minimi, eli piste, jossa funktio on minimoitu. Oheisessa kuvassa on esimerkki tilanteesta, jossa pallo on laskeutunut virhefunktion pohjalle globaaliin minimiin (kuvio 6).



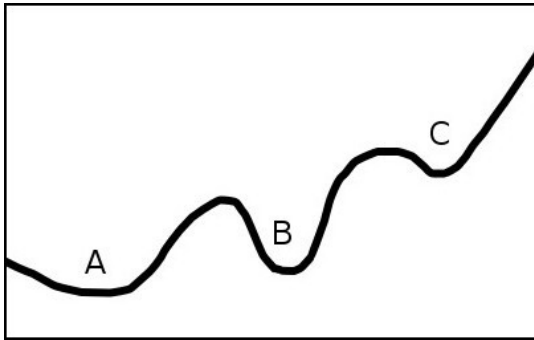
KUVIO 6 Funktion globaali minimi (Goodfellow ym., 2016)

Edellä esitettyssä analogiassa laakso vastaa neuroverkon virhefunktioita. Haykin (2005) kirjoittaa, että neuroverkkoja opettaessa pyritään selvittämään, miten kaikkia neuroverkon painoja ja raja-arvoja tulisi muuttaa neuroverkon virhefunktion minimoimiseksi. Kun painoihin ja raja-arvoihin tarvittavat muutokset ovat selvillä, on Nielsenin (2015) laaksovertauksessa selvitetty suunta, johon pallon tulisi valua. Tämän jälkeen palloa siirretään laaksossa pieni askel alamäkeen, mikä tarkoittaa neuroverkkoa minimoitaessa uuden pisteen laskemista gradientin vastavektorin suuntaan, eli siihen suuntaan, jossa virhefunktio vähenee eniten. Laaksovertauksen askel vastaa virhefunktion

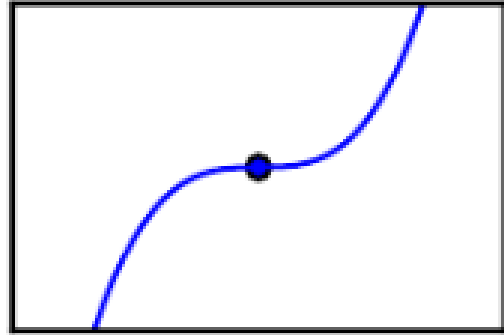
uutta pistettä laskiessa oppimisnopeutta, eli sopivan kokoista vakiota, joka määrittää uuden pisteen etäisyyden suhteessa vanhaan. Uusia pisteitä lasketaan iteratiivisesti, kunnes virhefunktio on minimoitu (Haykin, 2005).

Gradientin esittelyn yhteydessä mainittiin, että gradientissa tarvitaan muuttujien osittaisderivaattoja, eli funktion derivaattoja kunkin muuttujan suhteen muiden muuttujien pysyessä vakioina (Pitkäranta, 2015). Nielsenin (2015) mukaan neuroverkon tapauksessa osittaisderivaatat selvitetään kaikkien neuronien välisten painojen ja raja-arvojen suhteen. Kun osittaisderivaatat on selvitetty, voidaan niiden avulla käyttää gradienttimenetelmää ja laskea uusia pisteitä gradientin vastavektorin eli neuroverkon virhefunktion nopeimman vähenemisen suuntaan, kunnes virhefunktio on minimoitu (Nielsen, 2015).

Funktiota minimoitaessa on mahdollista, että optimointialgoritmi päättyy kyllä minimiin, mutta ei globaaliin minimiin. Goodfellow ym. (2016) kirjoittavat, että funktiossa voi olla globaalin minimin lisäksi muitakin sellaisia pisteitä, jotka ovat naapuripisteitään matalampia. Luonnollisesti gradienttimenetelmällä ei näissä tapauksissa löydetä pienempää arvoa, sillä Nielsenin (2015) esittelemässä laaksovertauksessa jouduttaisiin siirtymään ylämäkeen. Tällaista pistettä kutsutaan lokaaliksi minimiksi ja virhefunktion arvo voi tällöin olla kaukana globaalista minimistä. Goodfellow ym. (2016) jatkavat, että mitä enemmän muuttujia on, sitä vaikeammaksi optimointi voi monien mahdollisten lokaalien minimien takia osoittautua. Kuviossa 7 on esimerkki kuvaajasta, jossa näkyvät sekä globaali minimi että kaksi lokaalia minimiä, joista toisen arvo on melkein yhtä pieni kuin globaalin minimin arvo (kuvio 7). Toinen lokaali minimi taas on kaukana globaalista ja optimoinnissa sitä tulee välttää. Goodfellow ym. (2016) lisäävät, että globaalin ja lokaalin minimin lisäksi optimoinnissa voidaan päätyä myös pisteeseen, jonka naapuripisteet eivät ole pienempiä eivätkä suurempia kuin piste johon verrataan, ja sitä kutsutaan satulapisteeksi. Kuviossa 8 on esimerkkikuvaaja satulapisteestä (kuvio 8).



KUVIO 8 A: optimaalinen globaali minimi. B: lokaali minimi, joka on lähellä optimia. C: lokaali minimi, joka on kaukana optimista (Goodfellow ym., 2016 mukaan)



KUVIO 7 Satulapiste, jonka naapuripisteet ovat yhtä suuret (Goodfellow ym., 2016)

Goodfellow ym. (2016) kirjoittavat, että jos opetusdataa on paljon, voi olla jokseenkin hidasta käyttää jokaisessa gradienttimenetelmän iteraatiossa koko opetusdataa gradientin laskemiseen. Tästä syystä käytetään usein stokastista gradienttimenetelmää, jota he väittävät Nielsenin (2015) ohella ehkä kaikkein käytetyimmäksi opetusalgoritmiksi koneoppimisen parissa ja etenkin syväoppimisessä. Siinä neuroverkko koulutetaan koko opetusdatan sijaan opetusdatasta satunnaisesti koostettujen pienten otosten avulla. Virhefunktion gradientti lasketaan jokaisessa gradienttimenetelmän iteraatiossa näiden otosten perusteella. Satunnaisuuden vuoksi gradientin suunta ei ole välttämättä nopeimman vähenemisen suunta, mutta se on kuitenkin laskennallisesti tehokas arvio suunnasta. Lisäksi satunnaisuus auttaa lokaalien minimien ja satulapisteiden välttämiseksi (Goodfellow, 2016). Nielsen (2015) selventää stokastista gradienttimenetelmää luvuilla. Jos otoskoko on esimerkiksi $n = 60\,000$ ja siitä satunnaistettujen pienten otosten koko on $m = 10$, gradientin estimointi nopeutuu 6000-kertaisesti. Hän tähdentää myös, että stokastisessa gradienttimenetelmässä on tilastollista vaihtelua, mutta pienten otoskokojen avulla laskettu gradientti osoittaa yleisesti ottaen oikeaan suuntaan. Nielsen jatkaa, että arvio gradientista on neuroverkon opettamisen tapauksessa riittävä, eikä gradienttia tarvitse aina laskea koko aineiston datalla (Nielsen, 2015).

Edellä esitetyn perusteella gradienttimenetelmää voidaan hyödyntää neuroverkon opettamisessa. Sen avulla on mahdollista minimoida neuroverkon virhefunktion, mikä tarkoittaa käytännössä neuroverkon painojen ja raja-arvojen muuttamista niin, että virhefunktion pienenee. Neuroverkon neuronien väliset painot ja raja-arvot eivät kuitenkaan ole toisistaan riippumattomia. Koska neuroverkon neuronien ulostulot vaikuttavat myöhemmin verkossa sijaitseviin neuroneihin, ei ole itsestään selvää, miten kutakin paino- ja raja-arvoa tulisi

muuttaa. Tämän ongelman ratkaisemiseksi tarvitaan vielä vastavirta-algoritmiä, jota käsitellään seuraavassa alaluvussa.

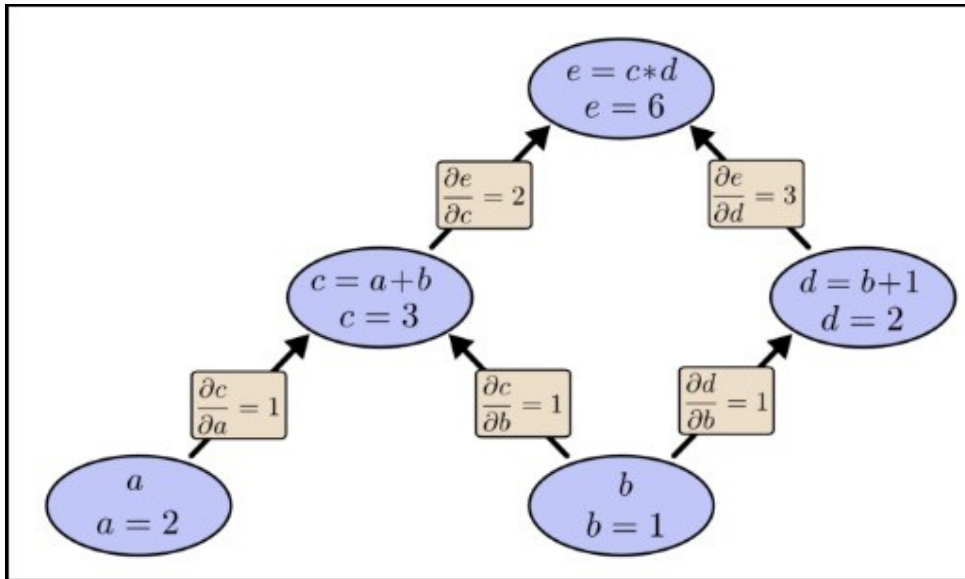
2.3.3 Vastavirta-algoritmi

Edellisessä alaluvussa esitettiin, että neuroverkkoa voidaan opettaa hyödyntämällä gradienttimenetelmää niin, että lasketaan sekä neuroverkon että kaikkien neuronien virhefunktioiden gradientit. Gradienttien avulla saadaan selville tieto siitä, miten neuroverkon neuroneiden paino- ja raja-arvoja tulee muuttaa, jotta neuroverkon virhefunktio vähenisi. Koska neuroverkon neuronien tulokset vaikuttavat jäljempänä neuroverkossa sijaitseviin neuroneihin, gradienttien laskemiseen tarvitaan gradienttimenetelmän lisäksi vielä toinen algoritmi, jossa otetaan huomioon neuronien vaikutus toisiinsa. Yksi sellainen algoritmi on vastavirta-algoritmi.

Vastavirta-algoritmi on neuroverkkojen opettamisessa keskeinen algoritmi, sillä sen avulla neuronien virhefunktioiden gradienttien laskeminen on laskennallisesti tehokasta (Goodfellow ym., 2016). Vastavirta-algoritmin nimi viittaa siihen, että jotakin tapahtuu vastavirtaan. LeCun ym. (2015) kirjoittavat, että gradienttimenetelmää varten tarvittavat gradientit lasketaan vastavirtaan niin, että algoritmissa käydään järjestyksessä läpi kaikki kerrokset ulostulokerroksesta syötekerrokseen. He jatkavat, että vastavirta-algoritmissa neuroverkkojen yhteydessä ei ole kysymys muusta kuin derivoinnin ketjusäännön käytännöllisestä soveltamisesta (LeCun ym., 2015).

Derivoinnin ketjusäännössä on taustalla ajatus siitä, että jos muuttujan derivaatta toisen muuttujan suhteen ei voida suoraan laskea, se voi olla mahdollista laskea toisten, "välissä" olevien muuttujien kautta (Olah, 2015). Olah selkeyttää ajatusta graafilla, jossa käytetään muuttujia a, b, c, d ja e (kuvio 9). Oletetaan, että pyritään selvittämään, miten graafissa olevan muuttujan a muutos vaikuttaa muuttujaan e. Koska muuttujat a ja e eivät ole suorassa yhteydessä toisiinsa, muuttujan a vaikutusta muuttujaan e ei voida laskea suoraan. Laskeminen on kuitenkin mahdollista derivoinnin ketjusäännöllä, eli kaavalla, jossa ensin lasketaan muuttujan c derivaatta muuttujan a suhteen ja kerrotaan se muuttujan e derivaatalla muuttujan c suhteen (Olah, 2015):

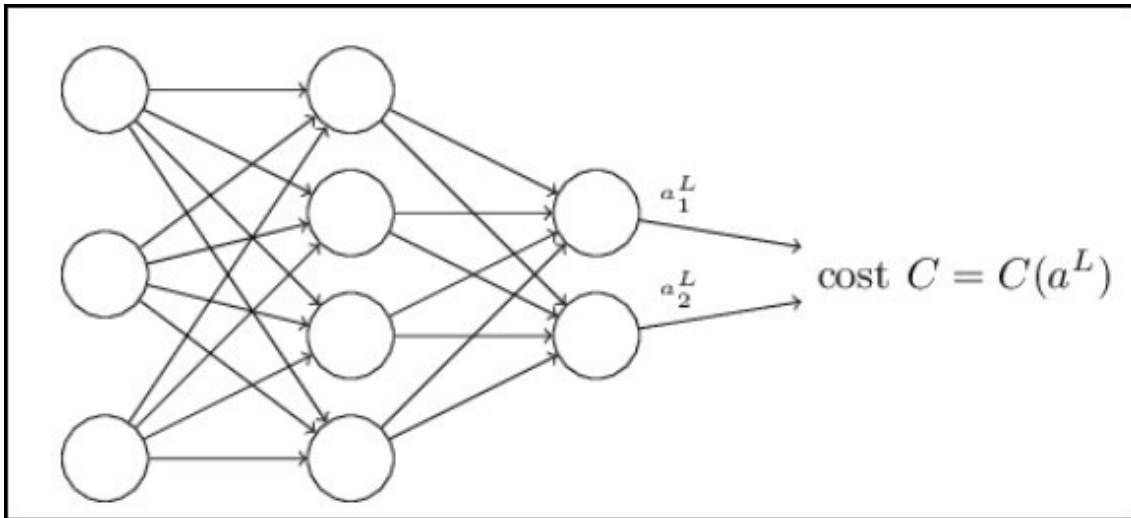
$$\frac{\partial e}{\partial a} = \frac{\partial c}{\partial a} \frac{\partial e}{\partial c}$$



KUVIO 9 e:n derivaatta a:n suhteen voidaan laskea välimuuttuja c:n avulla (Olah, 2015)

Vastavirta-algoritmissa neuronien virhefunktioiden gradientteja lasketaan siis derivoinnin ketjusäännöllä (Nielsen, 2015; Goodfellow ym., 2016; Olah, 2015). Nielsen jatkaa, että koska neuroverkon aiempina sijaitsevat neuronit aktivoituneen ja painotettuine summineen vaikuttavat koko neuroverkon ulostuloon, käytetään derivoinnin ketjusääntöä, jossa otetaan huomioon se, miten muutos kussakin muuttujassa vaikuttaa kunkin neuronin virhefunktioon. Virhefunktioihin vaikuttavia muuttujia ovat neuroverkon neuronien paino- ja raja-arvot sekä painotettu summa, joka on neuronin painon, raja-arvon sekä edellisen kerroksen aktivaation summa (Nielsen, 2015).

Nielsenin (2015) mukaan vastavirta-algoritmissa lasketaan ensiksi neuroneiden painotetut summat sekä aktivaatiot, joita tarvitaan virhelukujen laskemiseen. Hän jatkaa, että painotetut summat ja aktivaatiot lasketaan neuroverkon syötekerroksesta ulostulokerrokseen, "eteenpäin", sillä luonnollisesti sillä tavoin alkuperäisten syötekerrokseen annettujen syötteiden vaikutus kulkeutuu neuroverkossa neuronista toiseen. Tämän jälkeen selvitetään koko neuroverkon virheluku, eli ulostulokerroksen virheluku edellisen kerroksen aktivaatioiden ja painotettujen summien arvojen perusteella (Nielsen, 2015). Edellisen kerroksen aktivaatiot ja painotetut summat riittävät ulostulokerroksen virheluvun laskemiseen, koska aiemmin neuroverkossa sijaitsevien neuroneiden vaikutus kulkeutuu neuroverkossa lopulta ulostulokerrokseen. Nielsen havainnollistaa ulostulokerroksen virheluvun laskemista kuvalla (kuvio 10).



KUVIO 10 Ulostulokerroksen virhe (engl. cost) C aktivaatioiden a funktiona (Nielsen, 2015)

Kun koko neuroverkon virheluku eli ulostulokerroksen virheluku on selvitetty, vastavirta-algoritmissa selvitetään muiden kerrosten virheluvut järjestyksessä niin, että ulostulokerroksesta päädytään kerros kerrokselta aina syötekerrokseen asti (Nielsen, 2015). Nielsen jatkaa, että tässä vaiheessa selvitetään ensin ulostulokerroksesta edeltävän kerroksen virheluku niin, että laskussa otetaan huomioon aiemmin laskettu ulostulokerroksen virheluku. Samalla tavoin lasketaan myös muiden kerroksien virheluvut järjestyksessä niin, että ulostulokerroksesta päädytään kerros kerrokselta aina syötekerrokseen asti. Laskuissa otetaan siis huomioon aina yhtä edempänä olevan kerroksen virheluku, minkä ansiosta yhteys koko neuroverkon virhelukuun säilyy koko vastavirta-algoritmin ajan (Nielsen, 2015).

Tässä vaiheessa vastavirta-algoritmissa on laskettu kaikki gradienttimenetelmää varten tarvittavat virheluvut. Nielsenin (2015) mukaan näitä virhelukuja voidaan hyödyntää aiemmin esitellyssä gradienttimenetelmässä. Hän jatkaa, että virhelukujen perusteella voidaan laskea gradienttimenetelmässä tarvittavat osittaisderivaatat neuroverkon neuronien painojen ja raja-arvojen suhteen. Kun osittaisderivaatat on laskettu, voidaan painot ja raja-arvot päivittää niin että koko neuroverkon virheluku pienenee. Nielsenin mukaan neuroverkkoa opetettaessa edellä esiteltyä proseduuria toistetaan, kunnes neuroverkon virheluku on nolla tai riittävän lähellä nollaa (Nielsen, 2015).

2.3.4 Opettamisen ongelmia koneoppimisessa

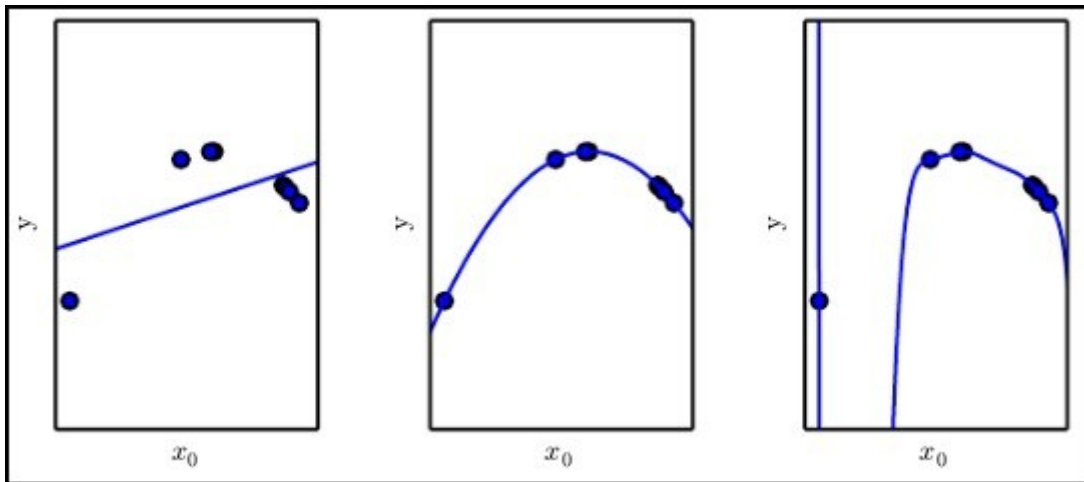
Tässä alaluvussa käsitellään joitakin opettamiseen liittyviä ongelmia koneoppimisessa, joita ovat ali- ja ylisovittaminen sekä neuroverkkojen käytännön hyödyntämistä vaikeuttava laskennallinen raskaus.

Goodfellow ym. (2016) kirjoittavat, että koneoppimisen ja niin myös neuroverkkojen keskeinen haaste on se, että opetusdatan lisäksi oppimisalgoritmin täytyy toimia hyvin algoritmille ennennäkemätöntä dataa käytettäessä. He esittävät, että koneoppimisessa huomioidaan yleisesti kaksi erilaista virhettä: opetusvirhe (engl. training error) sekä testivirhe (engl. test error). Opetusvirhe liittyy virheluvun minimoimiseen opetusdatan suhteen ja testivirhe taas kertoo, miten algoritmi suoriutuu ennennäkemättömästä datasta. Algoritmin kykyä toimia hyvin ennennäkemättömällä datalla kutsutaan yleistämiseksi (Goodfellow ym., 2016).

Edellä mainittiin, että kaksi tekijää, opetusvirhe ja testivirhe, määrittävät sen, miten hyvin oppimisalgoritmi suoriutuu. Goodfellow ym. (2016) kirjoittavat, että opettamisessa on tavoitteena minimoida opetusvirhe sekä saada opetus- ja testivirheen väli mahdollisimman pieneksi. Heidän mukaansa näin voidaan vastata koneoppimisessa kahteen keskeiseen ongelmaan, jotka ovat alisovittaminen ja ylisovittaminen. He jatkavat, että alisovittaminen ilmenee, kun algoritmi ei onnistu saamaan testivirhettä riittävän pieneksi. Ylisovittaminen taas ilmenee silloin, kun opetus- ja testivirheen väli on liian suuri (Goodfellow ym., 2016). Nielsen (2015) mainitsee ylisovittamisen neuroverkkojen suureksi haasteeksi, mutta ei käsittele alisovittamista lainkaan. Myös muissa tutkimuksissa alisovittamista ei välttämättä mainita ollenkaan tai sitä käsitellään paljon vähemmän kuin ylisovittamista, mistä voitaneen päätellä, että ylisovittaminen nähdään tutkimuksessa suurempana ongelmana.

Dougherty (2013) selventää, että ylisovittamisen voi ajatella niin, että algoritmi oppii opetusdatan niin hyvin, että se ei toimi yleistävästi testidatan kanssa. Toisin sanoen algoritmi tulkitsee opetusdataa liian tarkasti ja oppii siitä sellaisia irrelevantteja piirteitä, joita on vain opetusdatassa (Bramer, 2013). Testidatasta näitä opetusdatan yksityiskohtaisia piirteitä ei välttämättä löydy, jolloin yleistäminen ei toimi (Dougherty, 2013).

Goodfellow ym. (2016) kirjoittavat, että kun algoritmi ei saa testivirhettä riittävän pieneksi, ilmenee alisovittamista. Van der Aalst ym. (2008) selventävät, että alisovittava algoritmi yleistää dataa liikaa niin, että virhettä syntyy sen takia. Alisovittamista voi tapahtua esimerkiksi silloin, jos opetusdataa on liian vähän (Dougherty, 2013). Goodfellow ym. (2016) havainnollistavat ali- ja ylisovittamista kolmella kuvaajalla (kuvio 11).



KUVIO 11 Vasemmalla alisovittaminen, jossa dataa yleistetään liikaa. Keskellä hyvin yleistävä funktio. Oikealla ylisovittaminen, jossa tulkitaan turhaan syvä, laaksomainen kuvaaja (Goodfellow ym., 2016)

Tutkimuksissa keskitytään ali- ja ylisovittamisen kohdalla pääasiassa siihen, miten ylisovittamista voidaan välttää. Joitakin keinoja tähän ovat esimerkiksi opetusdatan lisääminen (Nielsen, 2015), neuronien väliaikainen poistaminen verkosta (Srivastava ym., 2014) ja opettamisen lopettaminen riittävän aikaisin (Wan ym., 2013).

Neuroverkkojen käyttömahdollisuuksia voisi parantaa esimerkiksi mahdollisuus hyödyntää älypuhelimella toimivaa, lintujen ääniä tunnistavaa sovellusta. Kuitenkin, vaikka neuroverkkoja voidaan optimoida laskennallisesti suhteellisen tehokkaiksi, suurten datamäärien käsittely asettaa rajoitteita neuroverkkojen hyödyntämiselle käytännön tasolla (Goodfellow ym., 2016). Neuroverkkojen opettamisessa käytetään nykyään usein tietokoneella tehokkaita grafiikkasuorittimia, joiden muisti riittää perinteisiä mikroprosessoria paremmin neuroverkkojen algoritmeissa käytettyjä laskutoimituksia varten (Goodfellow ym., 2016). Älypuhelimissa on huomattavasti rajallisemmat resurssit kuin tietokoneissa. Despois (2018) kirjoittaa, että vaikka älypuhelimista tehdään tehokkaampia, niiden laskennallinen teho, akunkesto ja tallennustila eivät riitä neuroverkkojen käyttöön yhtä hyvin kuin tietokoneiden tehokkaat grafiikkasuorittimet. Hän jatkaa, että on olemassa tekniikoita, joiden avulla neuroverkkoja voidaan jollain tasolla hyödyntää älypuhelimissa, mutta jos opetusdataa on todella paljon, älypuhelimien teho ei riitä neuroverkkojen opettamiseen (Despois, 2018).

2.4 Konvoluutioneuroverkot

Tässä alaluvussa käsitellään automaattisen lintujen äänten tunnistamisen kannalta olennaista neuroverkkotyyppiä, konvoluutioneuroverkkoja. LeCun ym. (2015) kirjoittavat, että konvoluutioneuroverkot ovat neuroverkkotyyppi, jota on 2000-luvun alusta asti hyödynnetty menestyksekkäästi erilaisten objektien ja alueiden havaitsemisessa, osittamisessa sekä tunnistamisessa kuvista. He listaavat konvoluutioneuroverkkojen eri käyttökohteiksi esimerkiksi liikennemerkkien tunnistamisen (Cireşan, Meier, Masci & Schmidhuber, 2012) ja biologisten kuvien segmentoinnin (Ning ym., 2005). LeCun ym. jatkavat listaa vielä ihmisten havaitsemiseen kuvista (Sermanet, Kavukcuoglu, Chintala, & LeCun, 2013) ja etenkin kasvojentunnistusta he pitävät suurena viimeaikaisena kehitysaskeleena (Taigman, Yang, Ranzato & Wolf, 2014) (LeCun ym., 2015).

LeCunin ym. (2015) mukaan konvoluutioneuroverkkojen avulla voidaan prosessoida kuvien lisäksi mitä tahansa monen matriisin muodossa olevaa dataa, minkä takia ne soveltuvat myös lintujen äänten tunnistamiseen (Sevilla & Glotin, 2017). Esimerkkeinä LeCun ym. (2015) mainitsevat 1-ulotteiset taulukot signaaleja ja sekvenssejä varten, 2-ulotteiset taulukot kuvien tai kuviksi muutettuja ääniä eli spektrogrammeja (Abdel-Hamid ym., 2014) varten ja 3-ulotteiset taulukot esimerkiksi videoita varten. Heidän mukaansa konvoluutioverkkojen mahdollisuudet ovat saaneet useimmat suuret teknologiayritykset kuten Googlen, Facebookin, Microsoftin, IBM:n, Yahoo!n, Twitterin, ja Adoben tutkimaan ja kehittämään konvoluutioneuroverkkoihin liittyviä projekteja ja tuotteita (LeCun ym., 2015).

Myös lintujen äänten tunnistuksessa hyödynnetään konvoluutioneuroverkkoja, sillä lintujen ääniä sisältävät äänitteet voidaan muuttaa kuviksi, spektrogrammeiksi, minkä jälkeen niitä voidaan luokitella kuvantunnistuksen keinoin konvoluutioneuroverkoilla (esim. Sevilla & Glotin, 2017; Sprengel ym., 2016 ja Martinsson, 2017). Tässä alaluvussa käsitellään konvoluutioneuroverkkoja kuvantunnistuksen näkökulmasta, mutta koska ääni voidaan muuttaa kuvaksi, käsiteltyjä asioita voidaan soveltaa myös lintujen äänten tunnistamiseen.

Seuraavaksi kuvaillaan miten konvoluutioneuroverkot toimivat, ja mitkä ovat keskeiset konvoluutioneuroverkkoihin liittyvät käsitteet. Tärkeimpiä konvoluutioneuroverkkoihin liittyviä käsitteitä ovat konvoluutio- ja yhdistämiskerros sekä suodatin ja piirrekartta.

2.4.1 Konvoluutio

Tässä aluvussa kuvaillaan konvoluutioneuroverkkojen konvoluutio-operaatiota. Konvoluutioneuroverkkoa voidaan käyttää kuvantunnistuksessa esimerkiksi niin, että kuvasta pyritään tunnistamaan käsinkirjoitettuja numeroita etsimällä piirteitä, joiden perusteella numerot voidaan tunnistaa (LeCun, Bottou, Bengio & Haffner, 1998).

Konvoluutioneuroverkon sisääntulona voidaan käyttää kuvaa, joka voidaan nähdä kaksiulotteisena pikseleistä koostuvana matriisina (LeCun ym., 2015). Matriisissa kukin pikseli vastaa yhtä neuronua, mutta toisin kuin tässä tutkielmassa aiemmin esiteltiin, konvoluutioneuroverkon tapauksessa kaikki neuronit eivät ole yhteydessä toisiinsa (Nielsen, 2015). Nielsen kutsuu sisääntulona käytettyä matriisia kerrokseksi ja jatkaa, että konvoluutio-operaatiossa kaikkien neuronien kytkemisen sijaan luodaan toinen, sisääntulokerrosta pienempi kerros, jossa kukin neuroni on muodostettu sisääntulokerroksessa toisiaan lähellä olevien neuronien perusteella. Tätä pienempää kerrosta kutsutaan piirrekartaksi (engl. feature map) (Goodfellow ym., 2016).

Piirrekartta muodostetaan tarkastelemalla alkuperäistä kuvaa osa kerrallaan. Goodfellow ym. (2016) kirjoittavat, että kuvassa voi olla tuhansia tai miljoonia pikseleitä, mutta siitä voidaan etsiä pieniä merkityksellisiä piirteitä, kuten ääriviivoja, suodattimen (engl. kernel) avulla. Abdel-Hamid ym. (2014) vertaavat suodatinta ikkunaan, jonka läpi alkuperäistä kuvaa tarkastellaan osa kerrallaan pienin harppauksin. He jatkavat, että harppauksen pituutta kutsutaan askelpituudeksi (engl. stride length). Olah (2014) selventää piirrekartan muodostumista kuvalla, jossa X-kirjaimista muodostettu kerros vastaa sisääntulokerrosta, A-kirjaimista muodostunut kerros piirrekarttoja ja F-kirjaimella merkattu kerros ulostulokerrosta (kuvio 12).



KUVIO 12 Piirrekarttojen muodostuminen konvoluutiassa. X-kerros vastaa syötekerrosta ja A-kerros piirrekarttoja. F vastaa ulostulokerrosta (Olah, 2014)

LeCun ym. (2015) kirjoittavat, että kuvissa toisiaan lähellä sijaitsevat pikselit korreloivat usein vahvasti, joten ne voivat muodostaa aiheita (engl. motif), joita on helppo havaita. He jatkavat, että jos aihe löytyy jostakin kuvan osasta, se voi löytyä myös muualta kuvasta. Suodattimen ansiosta kukin piirrekartan neuronin muodostetaan toisiaan lähellä olevien sisääntulokerroksen neuronien perusteella, ja siksi kullakin suodattimen läpi muodostetulla piirrekartan neuronilla on samat paino- ja raja-arvot (Bishop, 2006).

Suodatin määrittää sen, millaista piirrettä kuvasta milloinkin etsitään (Goodfellow ym., 2016). Tämä johtaa siihen, että kaikki saman suodattimen läpi luodut piirrekartan neuronit etsivät kuvasta samaa piirrettä. Guo ym. (2016) kirjoittavat, että tämän ansiosta etsityn piirteen sijainnilla ei ole väliä, sillä suodatin käy askel askeleelta läpi koko kuvan. Käytännössä tämä tarkoittaa sitä, että jos kuvasta pyritään etsimään esimerkiksi kasvoja, ne voivat olla kuvassa missä tahansa ja konvoluutioneuroverkko löytää ne sijainnista huolimatta (Goodfellow ym., 2016).

LeCun ym. (2015) kirjoittavat, että piirrekartoista muodostuu lopulta konvoluutiokerros, joka on määrältään arkkitehtuurista riippuva joukko piirrekarttoja, joissa kukin vastaa tietoa suodattimen avulla etsitystä piirteestä.

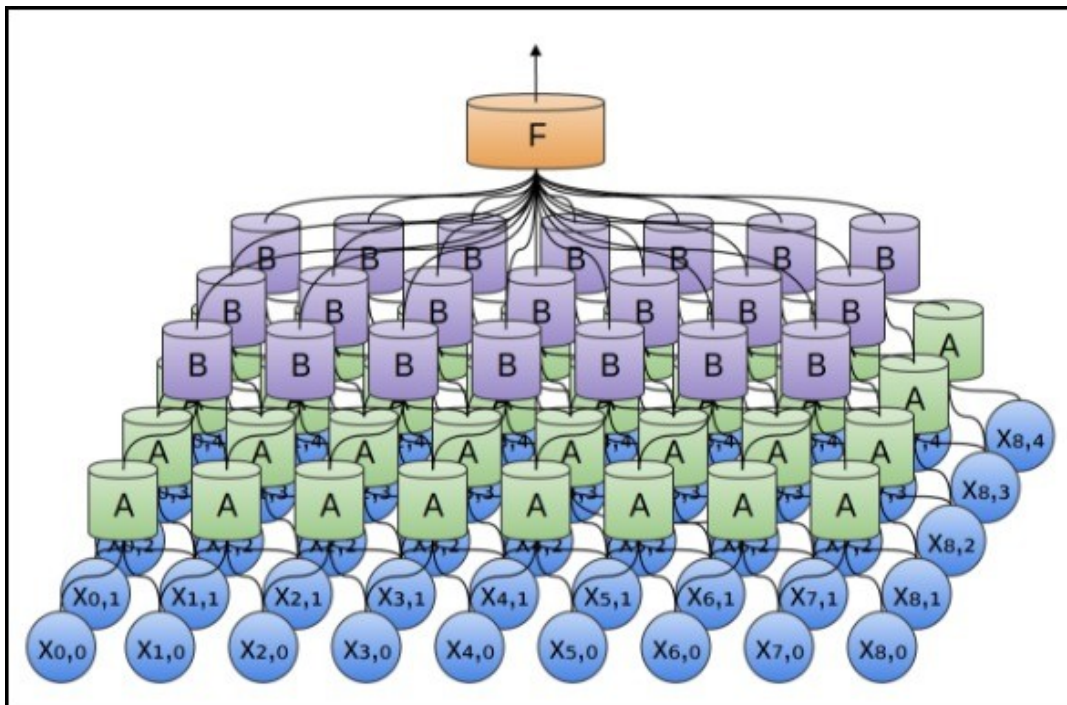
He tarkentavat, että konvoluutiokerroksen rooli on havaita yhteyksiä edellisen kerroksen lähekkäin olevien neuronien väliltä.

2.4.2 Yhdistäminen

Yhdistäminen (engl. pooling) on vaihe, joka tapahtuu yleensä heti konvoluution jälkeen (Nielsen, 2015). LeCun ym. (2015) kirjoittavat, että siinä missä konvoluutiokerroksessa havaitaan yhteyksiä edellisen kerroksen lähekkäin olevien neuronien väliltä, yhdistämiskerroksessa yhdistetään semanttisesti samankaltaiset piirteet yhteen jollakin yhdistämismenetelmällä. Yhdistämismenetelmiä on useita (Liu ym., 2017), mutta useimmissa aineiston tutkimuksissa käytetään yhdistämisoperaation esimerkkinä maksimiyhdistämistä (engl. max pooling).

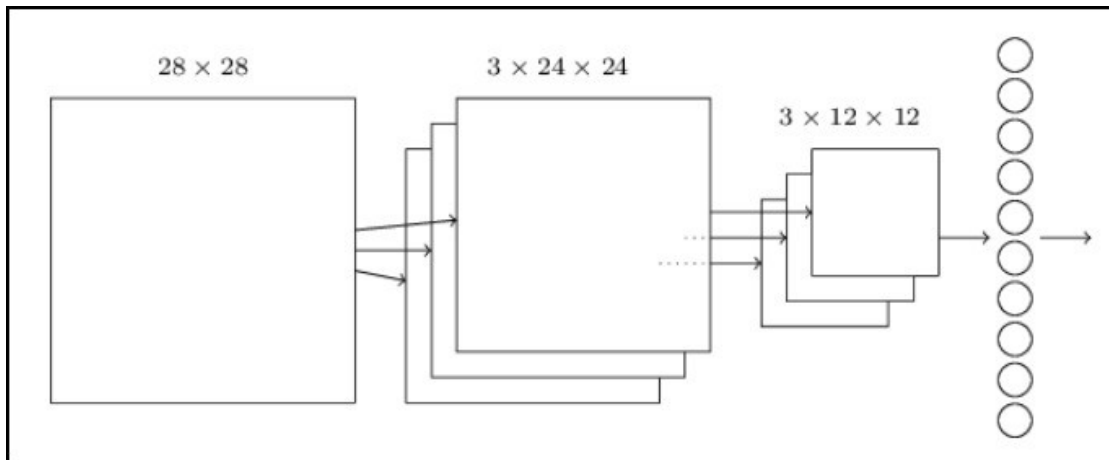
LeCunin ym. (2015) mukaan maksimiyhdistämisessä lasketaan yhteen piirrekartassa lähekkäin olevien neuroneiden maksimiarvot, minkä seurauksena yhdistämiskerrokseen tiivistyy tieto samankaltaisista piirteistä. Abdel-Hamid ym. (2014) selventävät, että tiivistämisen takia yhdistämiskerroksen piirrekartat pienenevät edellisen kerroksen piirrekarttoihin nähden, ja että yhdistämiskerrokseen muodostuu yhtä monta piirrekarttaa kuin edelliseen kerrokseen.

Nielsen (2015) kirjoittaa, että yhdistämiskerroksessa yksinkertaistetaan konvoluutiokerroksen informaatiota. Hän jatkaa, että yhdistämiskerroksen piirrekartaksi muodostuu konvoluutiokerroksesta tiivistetty piirrekartta. Koska konvoluutiokerroksessa on Nielsenin mukaan yleensä useita piirrekarttoja, maksimiyhdistämistä tai muuta yhdistämismenetelmää sovelletaan jokaiseen konvoluutiokerroksen piirrekarttaan erikseen niin, että tiivistettyjä piirrekarttoja muodostuu yhdistämiskerrokseen konvoluutiokerrosta vastaava määrä. Olah (2014) havainnollistaa yhdistämiskerrosta kuvalla (kuvio 13).



KUVIO 13 Konvoluutioverkko, jossa X-kerros vastaa syötekerrosta, A-kerros piirrekerrosta, B-kerros yhdistämiskerrosta ja F-kerros ulostulokerrosta (Olah, 2014)

LeCun ym. (2015) kirjoittavat, että samassa konvoluutioneuroverkossa voi olla useita peräkkäisiä konvoluutio- ja yhdistämiskerroksia. Konvoluutioneuroverkon lopullinen ulostulo määrittyy neuroverkon käyttötarkoituksen mukaan. Nielsen (2015) käyttää esimerkkinä yksinkertaistettua konvoluutioneuroverkkoa, jossa sisääntulokerroksesta muodostetaan kolme piirrekarttaa sisältävä konvoluutiokerros, josta taas muodostetaan kolme pienempää piirrekarttaa yhdistämiskerrokseen. Koska Nielsenin esimerkissä pyritään tunnistamaan käsinkirjoitettuja numeroita, neuroverkon viimeisenä kerroksena on kymmenen neuronin sisältävä ulostulokerros, josta voidaan nähdä neuroverkon antama arvio siitä, mikä numero syötteenä annetussa kuvassa on (kuviokuva 14).



KUVIO 14 Vasemmalta oikealle: 28×28 pikselin kokoinen syöte, konvoluutiokerros ja yhdistämiskerros (Nielsen, 2015)

3 NEUROVERKOT LINTUJEN ÄÄNTEN TUNNISTUKSESSA

Edellisessä luvussa on esitelty neuroverkkoja ja niihin liittyviä lähikäsitteitä, joista kaikki näkyvät neuroverkkoavusteisessa lintujen äänten tunnistamisessa. Esimerkiksi konvoluutioneuroverkkoja voidaan käyttää lintujen äänten tunnistamisessa (Sprengel ym., 2016) ja niiden opettamisessa käytetään usein vastavirta-algoritmia ja gradienttimenetelmää (LeCun ym., 2015). Automaattisissa lintujen ääniä tunnistavissa neuroverkoissa käytetään aktivointifunktioita esimerkiksi korjatussa lineaarisessa funktiossa eli ReLU-funktiossa (engl. lyh. rectified linear unit) (Sprengel ym., 2016) ja softmax-funktiossa (Kahl ym., 2017). Kaikissa tässä tutkielmassa käsitellyissä automaattiseen lintujen äänten tunnistamiseen liittyvissä raporteissa ylisovittamisen ongelmaan pyritään vastaamaan käyttämällä erilaisia datan kasvattamismenetelmiä (Sprengel ym. 2016; Kahl ym., 2017; Martinsson, 2017; Fazekas ym., 2017; Sevilla & Glotin, 2017).

Tässä luvussa käsitellään sitä, miten neuroverkkoja voidaan hyödyntää automaattisessa lintujen äänten tunnistuksessa. Aluksi esitellään johdatuksena automaattiseen äänentunnistukseen liittyviä haasteita, jotka liittyvät erityisesti lintujen äänten tunnistamiseen. Haasteet on kerätty automaattista lintujen äänten tunnistusta käsittelevistä lähteistä.

Haasteiden jälkeen käsitellään sitä, miten neuroverkkoja on käytännössä hyödynnetty lintujen äänten tunnistamisessa. Alaluvun alussa listataan joitakin sellaisia automaattisessa lintujen äänten tunnistuksessa käytettyjä menetelmiä, jotka eivät perustu neuroverkkoihin. Aiemmin käytettyjen menetelmien listaamisen jälkeen tarkastellaan esimerkkejä, joissa neuroverkkoja on käytännössä hyödynnetty automaattisessa lintujen äänten tunnistamisessa. Esimerkkien aineistona käytetään parhaiten menestyneiden osallistujien

raportteja automaattisen lintujen äänten tunnistamisen BirdCLEF-nimisistä kilpailuista (Goëau ym., 2016; Goëau ym., 2017). Tarkasteltaviksi on valittu raportit vuosilta 2016 ja 2017, koska niiden voittajat ovat saavuttaneet edellisiä vuosia paremmat tulokset, ja koska niissä on hyödynnetty neuroverkkoja. Neuroverkkopohjaiset ratkaisut ovat myös sijoittuneet kyseisissä kilpailuissa parhaille sijoille. Toisin sanoen neuroverkkopohjaiset ratkaisut ovat pärjänneet parhaiten BirdCLEF-kilpailuissa vuosina 2016 ja 2017 (Goëau ym., 2016; Goëau ym., 2017). Edellä mainittujen raporttien lisäksi tässä luvussa tarkasteltavaan aineistoon kuuluu yksi pro gradu -tutkielma, joka käyttää samaa aineistoa kuin yksi edellä mainituista raporteista (Martinsson, 2017).

Johtopäätöksinä pohditaan, miten tutkielmassa ilmi käyneet asiat vastaavat tutkimusongelmaan "*miten neuroverkkoja voidaan hyödyntää lintujen äänten tunnistamisessa?*" ja verrataan niitä johdannossa esiteltyihin motiiveihin. Johdannossa motiiveiksi luetellaan tavoitteita sekä luonnonsuojelun, opetuksen että lintuharrastuksen näkökulmasta.

Fagerlund (2014) näkee ongelmalliseksi sen, että niin monessa tutkimuksessa keskitytään ennalta määrättyyn spesifiin ongelmaan ja tutkimusta varten kerätty data on kerätty kyseistä tutkimusta varten. Lisäksi hän kirjoittaa, että kerätty data ei välttämättä ole yleisesti saatavilla, ja että lintujen äänistä ei ole olemassa kunnollista julkista tietokantaa. Hän arvioi, että niin kauan kun lintujen ääniä sisältävää julkista tietokantaa ei ole, erilaisten metodien vertailu tulee olemaan vaikeaa (Fagerlund, 2014).

Nykyään julkinen lintujen ääniä sisältävä tietokanta on olemassa. Goëau ym., (2015) kirjoittavat, että on olemassa eri puolilta maailmaa kerätty julkinen aineisto lintujen äänistä, jota käytetään BirdCLEF-kilpailujen aineistona. Aineisto on peräisin xeno-canto.org-sivustolta (Xeno-canto Foundation, 2018), ja siihen kuului vuonna 2015 yli 240 000 lintujen ääniä sisältävää äänitettä, joista noin 33 000:aa käytettiin BirdCLEF 2015 -kilpailussa (Goëau ym., 2015). Nykyään Xeno-cantoon on lisätty on jo yli 400 000 äänitettä (Xeno-canto Foundation, 2018). Vaikka Xeno-cantoon on lisätty näennäisesti paljon äänitteitä, Sprengel ym. (2016) kirjoittavat, että neuroverkkojen kunnollista opettamista varten tarvittaisiin vielä enemmän dataa.

BirdCLEF-kilpailut ovat osoittautuneet varsin suosituksi kilpailuksi, sillä vuosittain kilpailuun on osallistunut yli 75 osanottajaa. Vuonna 2014 BirdCLEF-kilpailuun osallistui 87 joukkuetta ja vuoden 2015 osallistujamäärä oli 137 (Goëau ym., 2014; Goëau ym., 2015). Vuosina 2016 osanottajia oli 84, ja vuonna 2017 osallistui 78 joukkuetta (Goëau ym., 2016; Goëau ym., 2017). Goëau ym. (2015) selittävät runsasta osallistujamäärää sillä, että luonnonääniin keskittyvät bioakustiikkaa tutkivat yhteisöt ovat kiinnostuneita testaamaan teknologioitaan kilpailun tarjoamalla datalla. Lintujentunnistusta voidaan siis käyttää äänentunnistuksen esimerkkinä bioakustiikan alueella.

BirdCLEF-kilpailujen lisäksi on ollut myös muita muita samankaltaisia lintujentunnistuskilpailuja., kuten lintuaiheisen haasteen antanut MLSP 2013 (engl. lyh. machine learning and signal processing) (Briggs ym., 2012) sekä lintuaiheisen haasteen antanut NIPS4B 2013 (engl. lyh. neural information processing scaled for bioacoustics) (Lasseck, 2013). Tässä tutkielmassa tarkastellaan kuitenkin vain BirdCLEF-kilpailuja, sillä niitä on järjestetty viime vuosina ja niissä kilpailevat ratkaisut ovat nykyaikaisemmalla tasolla.

3.1 Äänentunnistuksen haasteita

Tässä alaluvussa käsitellään lintujen automaattiseen äänten tunnistamiseen liittyviä haasteita. Haasteet liittyvät niin lintujen määrään äänitysympäristössä kuin itse lintujen ääntelyn moninaisuuteen. Myös se, millaista tietoa linnunlaulusta pyritään keräämään, aiheuttaa tiettyjä rajoitteita. Kun lintujen äänten automaattista tunnistusta aletaan tehdä, on olennaista määrittää tehtävään sopivat rajaukset sekä kysyä oikeita kysymyksiä (Martinsson, 2017).

Martinssonin (2017) mukaan luokitteluun voi liittyä monia kysymyksiä. Esimerkiksi voidaan olla kiinnostuneita siitä, onko äänitteessä lintu vai ei, jolloin lintujen luokittelu on binääristä. Toinen kiinnostuksen kohde voi olla olla lintuyksilöiden määrä äänitteessä. Toisaalta voi olla tarpeen erottaa äänitteestä lintulajeja, tai voidaan olla kiinnostuneita lintujen ääntelyn ajoituksesta (Martinsson (2017).

Luokittelun rajauksen lisäksi lintujen äänten automaattiseen tunnistamiseen liittyy myös muita haasteita. Sprengel ym. (2016) listaavat seuraavat haasteet:

- Taustahäly
- Monen linnun samanaikainen ääntely
- Lintujen laulu- ja kutsuäänien erot
- Lajien välinen vaihtelu
- Äänitteiden pituuksien vaihtelu
- Lintujen runsas lajimäärä

McIlraith ja Card (1995) lisäävät haasteisiin vielä lintujen yksilölliset erot ja niiden laulujen dialektit eli murteet. Dialektilla tarkoitetaan sitä, että saman lajin lintujen ääntelyssä voi olla samaan aikaan sekä riittävästi samankaltaisuutta että erilaisuutta (Lemon, 1975). Ajatus on tuttu, sillä dialekti ilmenee myös ihmisten keskuudessa. Eri puolilla Suomea asuvat ihmiset puhuvat samaa kieltä, mutta käyttävät eri murteita. Murteista huolimatta

ihmiset silti ymmärtävät toisiaan. Lintujen dialektista Lemon (1975) käyttää esimerkkinä Yhdysvalloissa eläviä punakardinaaleja (*Cardinalis cardinalis*), joista saman lajin yksilöt saattavat sijainnista riippuen laulaa hyvinkin eri tavoin.

3.2 Vuosien 2016 ja 2017 BirdCLEF-kilpailujen raporttien vertailua

Tässä aluvussa esitellään, miten neuroverkkoja on hyödynnetty automaattisessa lintujen äänten tunnistamisessa. Neuroverkkojen osalta siinä on sovellettu lähinnä konvoluutioneuroverkkoja niiden toimivuuden takia. Lintujen äänten tunnistamisessa on käytetty ennen konvoluutioneuroverkkoja myös muita teknologioita, joita ei neuroverkkoihin rajatun aiheen vuoksi käsitellä tässä tutkielmassa. Esimerkkejä muista lintujen äänten tunnistamisessa hyödynnetyistä teknologioista ovat:

- *Markovin piilomalli* (engl. hidden markov model) (Jancovic & Munevver, 2015; Kogan & Margoliash, 1998; Trawicki, Johnson & Osiejuk, 2005; Tyagi, Hedge, Murthy & Prabhakar, 2006),
- *Gaussin sekoitemalli* (engl. Gaussian mixture model) (Kwan ym., 2006; Lee, Hsu, Shih & Chou, 2013; Jancovic, Kokuer, Zakeri & Russell, 2013),
- *Itseorganisoituvat kartat* (Selin, Turunen & Tanttu, 2006; Somervuo & Härmä, 2003),
- *Koneoppimisen MIML-viitekehys* (engl.lyh. multi-instance multi-label framework) (Briggs ym., 2012)
- *Kaksiulotteiset cepstral-kertoimet* (engl. two-dimensional cepstral coefficients) (Lee, Han & Chuang, 2008).

Konvoluutioneuroverkkoja on hyödynnetty automaattisessa lintujen äänten tunnistuksessa esimerkiksi BirdCLEF 2017 -kilpailun voittaneessa ratkaisussa (Sevilla & Glotin, 2017).

Tässä aluvussa tarkasteltavia asioita automaattiseen lintujen äänten tunnistamiseen liittyen ovat automaattista lintujen äänten tunnistamista käsittelevissä raporteissa käytetyt tavat käsitellä audiosignaalia, datan kasvattamismenetelmät, neuroverkkoarkkitehtuurit sekä metadatan hyödyntäminen.

Tarkastelun alla on automaattisen lintujen äänten tunnistamisen osalta vuosien 2016 ja 2017 BirdClef-kilpailujen neljä parhaiten menestynyttä raporttia. Lisäksi aineistoon kuuluu vielä yksi lähde, joka ei perustu

lintujentunnistuskilpailuun (Martinsson, 2017). Se on pro gradu -tutkielma, jossa käytetään lähtö- ja vertailukohtana BirdCLEF 2016 -kilpailun voittaneen Sprengelin ym. (2016) työtä. Martinssonin (2017) käyttämä aineisto on sama kuin BirdCLEF 2016 -kilpailussa käytetty aineisto. Koska Martinsson käyttää samaa aineistoa sekä samoja mittalukuja tulosten esittelyssä kuin BirdCLEF 2016 -kilpailun osanottajat, hänen työnsä on vertailukelpoinen muiden artikkelien kanssa.

3.2.1 Audiosignaalin käsittely

Martinsson (2017) kirjoittaa, että puhdas audiodata ei sovi neuroverkon syötteeksi, joten audiosignaali täytyy muuntaa oikeanlaiseen muotoon, spektrogrammeiksi. Spektrogrammit ovat kuviksi muutettuja äänitiedostoja (Abdel-Hamid ym., 2014). Toisin sanoen, koska audiosignaali muutetaan spektrogrammiksi, äänentunnistuksesta tuleekin kuvantunnistusta. Konvoluutioneuroverkot soveltuvat kuvantunnistukseen hyvin, ja siksi äänentunnistuksessa käytetään usein niitä (LeCun ym., 1998). Muunnos spektrogrammiksi voi tapahtua eri tavoin, mutta signaalinkäsittelyn keinoja ei käsitellä tässä tutkielmassa. Esimerkiksi Sprengel ym. (2016) ja Martinsson (2017) käyttävät lyhyen aikavälin Fourier'n muunnosta (engl. short time Fourier transform) (Kraniasuskas, 1994).

Sprengel ym. (2016), Martinsson (2017), Fazekas ym. (2017) ja Kahl ym. (2017) erottelevat audiosignaalin kahteen eri osaan, signaaliosaan ja hälyosaan. Sprengel ym. (2016) kirjoittavat, että signaaliosat vastaavat lintujen ääniä tai lauluja ja hälyosat sisältävät hälyä tai hiljaisuutta. Signaali/häly-jaon jälkeen sekä signaali- että hälyosat muutetaan spektrogrammeiksi eli kuviksi. Signaaliosia voidaan hyödyntää neuroverkon opettamisessa ja myös hälyosalla on osuutensa esimerkiksi seuraavassa alaluvussa käsiteltävissä datan kasvattamismenetelmissä.

3.2.2 Datan kasvattamismenetelmät

Äänitiedostoiksi tallennettua dataa lintujen äänistä on parhaimmillaankin rajallinen määrä. Sprengel ym. (2016) käyttivät opetusjoukkona 24607 äänitiedostoa, mutta raportoivat, että neuroverkkojen kunnollista opettamista varten tarvitaan enemmän dataa. Tästä syystä kaikissa tässä luvussa käsitellyissä automaattista lintujen äänten tunnistamista koskevissa raporteissa käytettiin datan kasvattamismenetelmiä. Martinsson (2017) kirjoittaa, että datan kasvattaminen on tapa lisätä käytettävissä olevien otosten määrää aineistossa

tekemällä muunneltuja versioita olemassa olevista otoksista. Seuraavaksi esitellään artikkeleissa käytettyjä datan kasvattamismenetelmiä.

Hällyn lisääminen on kaikissa tässä luvussa käsiteltävissä raporteissa käytetty datan kasvattamismenetelmä. Tässä vaiheessa hyödynnetään audiosignaalin käsittelyn aikana luotuja hälyä sisältäviä spektrogrammeja. Sprengel ym. (2016) selittävät, että käytännössä häly-spektrogrammi lisätään opetusnäytteen päälle niin, että signaali-spektrogrammiin kuuluu nyt myös satunnainen häly-spektrogrammi tai monta sellaista. Näin hällyn avulla dataan saadaan muuntelua (Sprengel ym., 2016). Fazekas ym. (2017) käyttävät hälyä satunnaisesti niin, että osaan näytteistä ei tule hälyä ollenkaan. Kahl ym. (2017) käyttävät sekä synteettistä hälyä että äänitteistä irrotettua luonnollista hälyä. Heidän mukaansa useimmat mallit oppivat jättämään synteettisen hällyn huomiotta, mikä vahvistaa niitä myös realistisemman hällyn sietämisessä.

Yksi datankasvatusmenetelmä on *siirtäminen ajassa*. Sekä Sprengel ym. (2016) että Martinsson (2017) että Fazekas ym. (2017) kasvattavat dataa siirtämällä spektrogrammia ajassa satunnaisesti. Sprengel ym. (2016) kirjoittavat, että käytännössä tämä tarkoittaa sitä, että spektrogrammi katkaistaan satunnaisesta kohdasta kahteen osaan, vaihdetaan osien paikkaa ja liimataan ne taas yhdeksi spektrogrammiksi. He jatkavat, että näin neuroverkko opetetaan siihen, että lintujen ääniä voi kuulua lajista riippumatta koska tahansa. Toinen hyöty on se, että neuroverkko pakotetaan sietämään epäsäännöllisyyttä, jolloin siitä tulee robustimpi (Sprengel ym., 2016).

Dataa voidaan kasvattaa myös *muuttamalla äänenkorkeutta* spektrogrammissa. Sekä Sprengel ym. (2016), Martinsson (2017), Kahl ym. (2017) että Fazekas ym. (2017) muuttavat spektrogrammin äänenkorkeutta. Esimerkiksi Fazekas ym. (2017) muuttavat äänenkorkeutta satunnaisesti viidellä prosentilla. Kahl ym. (2017) pitävät äänenkorkeuden muuttamista tärkeimpänä käyttämänään datan kasvattamismenetelmänä.

Datan kasvattamisessa voidaan käyttää myös *saman luokan äänitteiden yhdistämistä* yhdeksi äänitteeksi. Fazekas ym. (2017) liittävät 70 prosentin todennäköisyydellä yhteen äänitteet, joissa on samaan luokkaan kuuluvien lintujen ääniä. Myös Martinsson (2017) tekee vastaavaa yhdistämistä ja selittää, että yhdistäminen hyödyttää siten, että verkolle voidaan näin syöttää enemmän relevanttia dataa kerralla. Myös Sprengel ym. (2016) ja Kahl ym. (2017) yhdistävät saman luokan äänitteitä.

Edellä esiteltyjen datan kasvattamismenetelmien lisäksi kahdessa artikkelissa käytettiin myös muita datan kasvattamismenetelmiä. Fazekas ym. (2017) yhdistivät *yhteen spektrogrammiin lähialueen lintujen ääniä sisältäviä spektrogrammeja*. Myös *äänitteen äänenvoimakkuuden muuttaminen* satunnaisesti viidellä prosentilla oli yksi heidän käyttämänsä datan kasvatusmenetelmä.

BirdCLEF 2017 -kilpailun voittaneet Sevilla ja Glotin (2017) käyttävät muihin nähden osittain samoja, mutta pääosin erilaisia tapoja kasvattaa dataa. Tämä johtuu osin siitä, että he käyttävät Googlen Inception-v4-konvoluutioneuroverkkoa, jossa on valmiina erilaisia tapoja prosessoida kuvia. Sevilla ja Glotin jatkavat, että dataan saadaan muuntelua esimerkiksi *muuttamalla satunnaisesti värisävyyä, kontrastia ja kirkkautta*. Heidän mukaansa Inception-v4 on suunniteltu niin, että syötteenä voidaan käyttää värikkäitä spektrogrammeja, mikä lisää muuntelua huomattavasti mustavalkokuvaan verrattuna (Sevilla & Glotin, 2017).

3.2.3 Neuroverkkoarkkitehtuurit

Tässä alaluvussa tarkastellaan sitä, millaisia neuroverkkoarkkitehtuureja automaattista lintujen äänten tunnistamista koskevissa artikkeleissa on kuvattu. Osassa ratkaisuista on suhteellisen yksinkertainen ja matala arkkitehtuuri, mutta osassa hyvinkin syvä. Esimerkiksi Fazekas ym. (2017) käyttävät neljää konvoluutiokerrosta, Martinsson (2017) 18 kerrosta syvää neuroverkkoa ja Sevillan ja Glotinin (2017) arkkitehtuurissa on satoja kerroksia.

Sprengel ym. (2016) käyttävät konvoluutioneuroverkkoa, jossa on viisi konvoluutiokerrosta, joista jokaista seuraa maksimiyhdistämiskerros. He kuvaavat, että kun syötteenä annettu spektrogrammi on kulkeutunut läpi konvoluutiokerroksista, seuraa vielä yksi täysin kytketty kerros (engl. dense layer tai fully connected layer) ja sen jälkeen softmax-kerros. Softmax-funktio on Nielsenin (2015) mukaan funktio, joka saa neuronien ulostulot vaikuttamaan toisiinsa niin, että kun yhden neuronin arvo nousee, muiden neuronien arvot laskevat vastaavasti. Nielsen jatkaa, että softmax-kerroksen neuronien ulostulojen yhteenlaskettu summa on aina 1, joten ne voi ajatella todennäköisyysjakaumana (Nielsen, 2015). Myös Sprengel ym. (2017) selventävät, että softmax-kerroksen tarkoitus on generoida todennäköisyydet jokaiselle luokalle eli esimerkiksi lintulajeille.

Fazekas ym. (2017) lähestyvät automaattista lintujen äänten tunnistamista 4-kerroksisella konvoluutioneuroverkolla, joista jokaista seuraa maksimiyhdistämiskerros kuten Sprengelin ym. (2016) ratkaisussakin. Tässä toteutuksessa on kuitenkin erona se, että Fazekas ym. (2017) käyttävät kahta sisääntulokerrosta, joista toiseen annetaan syötteenä spektrogrammi ja toiseen tunnistusapuna käytettävä metadata.

Myös Kahl ym. (2017) käyttävät konvoluutioneuroverkkoa. He käyttävät eri malleja, joissa kussakin on viidestä kuuteen konvoluutiokerrosta, joita kutakin seuraa maksimiyhdistämiskerros. Heidän ratkaisussaan on

neuroverkon ulostulona kolme täysin kytkettyä kerrosta ja aivan viimeisenä softmax-kerros (Kahl ym., 2017).

Martinssonin (2017) ratkaisu eroaa edellisistä niin, että se on arkkitehtuuriltaan huomattavasti syvempi. Hän käyttää kahdeksatoista kerrosta syvää konvoluutioneuroverkon sovellusta, residuaalinearoverkkoa. Residuaalinearoverkossa käytetään niin kutsuttuja oikotieyhteyksiä (engl. shortcut connections) vastauksena neuroverkon syvyydestä johtuvaan katoavan gradientin ongelmaan (engl. vanishing gradients) (He, Zhang, Ren & Sun, 2016). Martinsson (2017) kirjoittaa, että oikotieyhteyksien ansiosta neuroverkko on helpompi opettaa. Vaikka Martinssonin arkkitehtuuri on syvä, se on perusteiltaan samankaltainen kuin ne ratkaisut, joita Sprengel ym. (2017), Fazekas ym. (2017) ja Kahl ym. (2017) käyttävät. Martinsson (2017) kirjoittaa, että hänen ratkaisunsa oli tehty Sprengelin ym. (2016) työn innoittamana, eikä se ollut innoittajaansa parempi. Verkon monimutkaisuus ei siis tässä tapauksessa parantanut tuloksia.

Arkkitehtuurin monimutkaisuudesta voi kuitenkin olla hyötyä. Kaikista monimutkaisin arkkitehtuuri oli nimittäin BirdCLEF 2017 -kilpailun voittajan ratkaisussa. Sevilla ja Glotin (2017) käyttivät ratkaisussaan Googlen Inception-v4-neuroverkkoa. Siinä on satoja kerroksia, mutta verkon muodostavat perusyksiköt ovat kuitenkin pääpiirteissään sellaisia, joita on käsitelty tässäkin tutkielmassa. Sevilla ja Glotin (2017) kirjoittavat, että Inception-v4-neuroverkossa on valtava määrä konvoluutiokerroksia, joskaan signaali ei kulkeudu verkon läpi täysin lineaarisesti. Inception-v4:n toiminnan läpikäynti on kuitenkin tämän tutkielman rajauksen ulkopuolella, joten sitä ei käsitellä enempää.

3.2.4 Metadatan hyödyntäminen

Sekä Martinsson (2017), Fazekas ym. (2017) että Kahl ym. (2017) kirjoittavat, että useimmissa aineiston äänitteissä on mukana metadataa. Fazekas ym. (2017) selventävät, että metadata sisältää tietoa esimerkiksi sijainnista, ajankohdasta ja maanpinnan korkeudesta äänityspaikassa. He jatkavat, että metadatan käyttö ei kuitenkaan ole täysin ongelmaton. Heidän mukaansa epäolennaisten tai tyhjen arvojen vuoksi metadataa on esikäsiteltävä ennen sen hyödyntämistä. Esimerkiksi puuttuvien arvojen kentät he täyttävät saman lajin toisten yksilöiden tiedoilla (Fazekas ym., 2017).

Kahl ym. (2017) mainitsevat kyllä metadatan olemassaolon aineistossa, mutta päättävät olla käyttämättä sitä lintujen luokittelussa. Syytä he eivät kerro, mutta se liittyy todennäköisesti siihen, että ratkaisusta saa huomattavasti yksinkertaisemman, kun metadataa ei tarvitse ottaa huomioon.

Sekä Martinsson (2017) että Fazekas ym. (2017) hyödyntävät metadatan neuroverkoissaan. Fazekas ym. (2017) käyttävät neuroverkkoarkkitehtuuria, jossa on kaksi sisääntuloa, joista toinen on varattu syötteenä käytetyille spektrogrammille ja toinen metadatalle. Fazekas ym. (2017) käyttävät jokaisessa kokeilussaan metadatan, eivätkä esittele vertailua, josta näkisi metadatan vaikutuksen verrattuna tilanteeseen, jossa metadatan ei hyödynnetä. Heidän ratkaisunsa oli kuitenkin BirdCLEF 2017 -kilpailun toiseksi paras, joten voitaneen olettaa, että metadatan vaikutus näkyy tuloksissa (Fazekas ym., 2017). Martinsson (2017) käyttää yhtä sisääntuloa ja raportoi, että metadatan käyttö paransi tuloksia verrattuna tilanteeseen, jossa metadatan ei hyödynnetty.

Martinsson (2017) nostaa esille aineistossa olevan metadatan ongelmia. Hänen mukaansa esimerkiksi jonkin lintulajin kaikki äänitteet voivat olla peräisin samalta seudulta saman henkilön äänittämiä. Hän jatkaa, että tämän takia yksipuolisen metadatan perusteella tehdyt johtopäätökset voivat olla vinoutuneita, mikä täytyy tarvittaessa ottaa huomioon (Martinsson, 2017).

3.2.5 Lintujentunnistuskilpailujen raporttien tuloksia

Tässä alaluvussa vertaillaan tässä luvussa esiteltyjen automaattista lintujen äänten tunnistamista koskevien artikkeleiden tuloksia. Kaikissa artikkeleissa käytettiin tulosten esittelyssä keskitarkkuuksien keskiarvoa, jota Manning, Raghavan ja Schütze (2009) pitävät yleisenä tarkkuuden mittarina. He jatkavat, että keskitarkkuus on jonkin joukon k parhaimpien tarkkuuksien tarkkuusarvojen keskiarvo. Tämän tutkielman kontekstissa keskitarkkuuksien keskiarvo tarkoittaa käytännössä sitä, että menetelmässä lasketaan välille $[0, 1]$ normalisoitu arvo, joka vastaa todennäköisyyksien keskiarvoa sille, laulaako jokin tietty lintu äänitteessä vai ei (Goëau ym., 2016). Tyypillisesti keskitarkkuuksien keskiarvot yksittäisessä järjestelmässä vaihtelevat välillä $[0.1, 0.7]$ (Manning ym., 2009). Luvut voidaan siis tulkita todennäköisyyksinä ja Martinsson (2017) käyttääkin keskitarkkuuksien keskiarvoista prosenttimuotoista merkintätapaa.

Goëau ym. (2016) kirjoittavat, että vuonna 2016 oli ensimmäinen BirdCLEF-kilpailu, jossa konvoluutioneuroverkko saavutti muita ratkaisuja paremmat tulokset. He jatkavat, että kilpailun voitti konvoluutioneuroverkkoa käyttänyt ratkaisu, sillä Sprengelin ym. (2016) konvoluutioneuroverkko saavutti yksittäisiä linnunlauluja tunnistettaessa keskitarkkuuksien keskiarvon 0,69, mikä oli huomattava parannus BirdCLEF 2015 -kilpailun parhaimpiin tuloksiin nähden. BirdCLEF 2015 -kilpailun kaksi parasta tulosta olivat 0,45 ja 0,58 (Goëau ym., 2016). Kun huomioon otettiin myös taustalla laulavat linnut, keskitarkkuuksien keskiarvo oli Sprengelin ym. (2016) ratkaisussa 0,55, siinä

missä BirdCLEF 2015 -kilpailussa vastaava luku oli parhaassa ratkaisussa 0,414 (Goëau, Glotin, Vellinga, Planqué, Rauber & Joly, 2015).

Goëau ym. (2016) näkevät, että Sprengelin ym. (2016) konvoluutioneuroverkon suoritus oli merkittävä siinä mielessä, että toisin kuin useimmissa konenäköhaasteissa, sitä ei hienosäädetty lainkaan ennen suoritusta. He kirjoittavat myös, että kilpailussa oli mukana myös sellaisia konvoluutioneuroverkkoja, jotka eivät pärjänneet erityisen hyvin. Konvoluutioneuroverkkojakin täytyy siis käyttää oikein, että ne toimivat mahdollisimman hyvin (Goëau ym., 2016).

Vuonna 2017 konvoluutioneuroverkkojen tulokset BirdCLEF 2017 -kilpailussa parantuivat vuoteen 2016 verrattuna. Kilpailun voittajat, Sevilla ja Glotin (2017), saivat yksittäisiä linnunlauluja tunnistettaessa keskitarkkuuksien keskiarvoksi 0,714, mikä oli edellisen vuoden voittajan, Sprengelin ym. (2016), arvoa 0,69 suurempi. Kun taustalla laulavat linnut otettiin huomioon, Sevillan ja Glotinin (2017) tulos oli 0,616 ja edellisenä vuonna Sprengelin ym. (2016) tulos 0,55. Myös BirdCLEF 2017 -kilpailussa toiseksi tulleet Kahl ym. (2017) saivat paremmat tulokset kuin Sprengel ym. (2016). Heidän tuloksensa olivat yksittäisten lintujen osalta 0,678 ja taustalla laulavien lintujen kanssa 0,605. BirdCLEF 2017 -kilpailulla kolmanneksi jääneet Fazekas ym. (2017) eivät pärjänneet edellisen vuoden voittajalle, vaan he saivat keskitarkkuuksien keskiarvoiksi 0,579 ja 0,511. Myös Martinssonin (2017) tulokset olivat Sprengelin ym. (2016) tuloksia matalammat: 0,679 ja 0,538. Alla olevassa taulukossa voidaan nähdä edellä esitetyt BirdCLEF 2017 -kilpailun tulokset selkeämmin tarkasteltavassa muodossa (taulukko 1).

TAULUKKO 1 Automaattisen lintujen äänen tunnistuksen keskitarkkuuksien keskiarvot

| | Vain yksittäiset linnunlaulut | Mukana taustalla laulavat linnut |
|--------------------------|-------------------------------|----------------------------------|
| Sevilla ja Glotin (2017) | 0,714 | 0,616 |
| Kahl ym. (2017) | 0,678 | 0,605 |
| Sprengel ym. (2016) | 0,690 | 0,550 |
| Martinsson (2017) | 0,679 | 0,538 |
| Fazekas ym. (2017) | 0,579 | 0,511 |

Goëau ym. (2017) näkevät BirdCLEF 2017 -kilpailun merkittävimpänä lopputuloksena sen, että parhaiten toimiva järjestelmä perustui puhtaasti kuvantunnistukseen pohjautuvaan konvoluutioneuroverkkoon nimeltä Inception-v4. He näkevät ilmiössä yleisen trendin, eli konvergenssin, jossa samat menetelmät voivat toimia parhaiten sovellusalueesta riippumatta. He

jatkavat, että vaikka automaattisen lintujen tunnistamisen järjestelmät toimivat vuonna 2017 edellistä vuotta paremmin, ovat ne silti globaaliin tasoon verrattuna vaatimattomia. Nimittäin esillä olevien sovellusten mahdollisuudet lintujen tunnistuksessa käytännön tasolla ovat heidän mukaansa vielä rajoitettuja. Suurena syynä tähän lienee se, että vaikka neuroverkot ovat tehokkaita, ne vaativat silti suhteellisen paljon laskutehoa (Goodfellow ym., 2016), ja että mobiililaitteiden laskennallinen teho, akunkesto ja tallennustila eivät riitä neuroverkkojen käyttöön yhtä hyvin kuin tietokoneiden tehokkaat grafiikkasuorittimet (Despois, 2018). Kuitenkin Goëau ym. (2017) näkevät potentiaalia neuroverkoissa ja etenkin metadatan hyödyntämisessä niin, että sovellukset voisivat tulevaisuudessa osoittautua käyttökelpoisiksi myös käytännössä.

3.3 Johtopäätökset

Tässä aluvussa esitellään tutkielman aikana syntyneitä johtopäätöksiä. Aluvussa pohditaan sitä, miten tutkielmassa ilmi käyneet asiat vastaavat tutkimusongelmaan "*miten neuroverkkoja voidaan hyödyntää lintujen äänten tunnistamisessa?*" Lisäksi tutkielmassa ilmenneitä asioita verrataan johdannossa esitelyihin motiiveihin, joita ovat tavoitteet luonnonsuojelun, opetuksen sekä lintuharrastuksen näkökulmasta. Toisin sanoen aluvussa tarkastellaan, että voiko neuroverkoista olla hyötyä käytännössä, ja että mitkä asiat vaikuttavat neuroverkkojen soveltamiseen lintujen äänten tunnistamisessa.

Neuroverkkoja voidaan hyödyntää automaattisessa lintujen äänten tunnistuksessa, sillä BirdCLEF-kilpailujen tulosten perusteella konvoluutioneuroverkot toimivat automaattisessa lintujen äänten tunnistamisessa muita käytettyjä teknologioita paremmin (Goëau ym., 2016; Goëau ym., 2017). Vaikka kilpailujen ansiosta eri teknologioita on helpompi vertailla, kilpailujen tuloksista ei kuitenkaan käy ilmi selvästi se, miten hyvin neuroverkkoavusteinen lintujen tunnistaminen käytännössä toimii. Eri teknologioiden välisen vertailun lisäksi voisi olla paikallaan tehdä vertailua lintuasiantuntijoiden ja neuroverkkojen välillä. Voi nimittäin olla, että vaikka neuroverkot toimivat olemassa olevista teknologioista parhaiten, ne ovat vielä asiantuntijoita selvästi jäljessä. Neuroverkoista voi olla silti hyötyä lintulajien tunnistamisessa ja luonnonsuojelun kannalta tärkeän datan keräämisessä, mutta neuroverkon suorittamien linnuntunnistusten tasoon ei välttämättä voi luottaa. Esimerkiksi Martinssonin (2017) esille tuoma huoli aineiston yksipuolisuudesta voi vaikuttaa siihen, millä tasolla neuroverkko oppii tunnistamaan lintuja. Martinssonin huomio liittyy äänitteiden metadataan:

Xeno-canton (2018) aineistossa jonkin lintulajin kaikki äänitteet saattavat olla peräisin samalta seudulta ja saman henkilön äänittämiä. Neuroverkon suorittamien linnuntunnistusten paikkaansapitävyyteen ei siis voi välttämättä luottaa.

Neuroverkkojen käytännön hyödyntäminen automaattisessa lintujen äänten tunnistamisessa riippuu osin käytettävissä olevista resursseista. Esimerkiksi älypuhelimissa on huomattavasti rajallisemmat resurssit kuin tietokoneissa. Älypuhelimien laskennallinen teho, akunkesto ja tallennustila eivät riitä neuroverkkojen käyttöön yhtä hyvin kuin tietokoneiden tehokkaat grafiikkasuorittimet (Despois, 2018). Tämä asettaa rajoitteita esimerkiksi harrastus- ja opetuskäytön näkökulmasta. Toisaalta tehokkaat tietokoneet eivät ole nykyään harvinaisia kotikäytössä, joten lintujen äänten tunnistaminen neuroverkkojen avulla lienee kuitenkin realistinen mahdollisuus myös harrastus- ja opetuskäytössä. Esimerkiksi älypuhelimella äänitettyjä äänitteitä voisi myöhemmin tunnistaa neuroverkkoavusteisesti tietokoneella.

Luonnonsuojelullisesta näkökulmasta ajatellen käytettävyyssrajoitteet eivät välttämättä ole niin suuri ongelma. Esimerkiksi voisi olla mahdollista asettaa rannikolle järjestelmä, joka tunnistaisi lintujen kevätmuuton yhteydessä yli lentäviä lintuja äänen perusteella ja tallentaisi havainnot tietokantaan. Automaattinen järjestelmä voisi kerätä dataa ympäri vuorokauden, joten näin saisi paljon tärkeää dataa. Tällä hetkellä lintudataa kerätään harrastajien toimesta (BirdLife Suomi ry, 2018b). Tällaisessa järjestelmässä lienee haasteena sen reaaliaikaisuus ja äänityksen laatu. Reaaliaikainen tunnistaminen voi olla huomattavasti erilaista kuin esimerkiksi lintujen tunnistaminen äänitteistä BirdCLEF-kilpailujen tapaan. Voi olla esimerkiksi mahdollista, että reaaliaikainen järjestelmä tulkitsee linnuiksi ääniä, jotka eivät ole lintujen luomia. Jos tästä tai muusta syystä virheellisiä tunnistuksia syntyy systemaattisesti ympäri vuorokauden ja pitkällä aikavälillä, kerättyyn dataan ei voisi luottaa ja se olisi käyttökeltovotonta.

Tutkielmassa tarkastellun aineiston pohjalta ei selvinnyt se, että olisiko neuroverkkoavusteista lintujen äänten tunnistamista mahdollista käyttää esimerkiksi älypuhelimella niin, että neuroverkko on koulutettu etukäteen tietokoneella ja siirretty sen jälkeen älypuhelimeen. Jos neuroverkon tehokas toiminta perustuu oikeiden painojen ja raja-arvojen löytämiseen, miksei näitä arvoja voisi laskea etukäteen tietokoneella? Kun opetusta varten tarvittavaa dataa ei tarvitsisi tallentaa puhelimeen, ja kun laskentaa ei tarvitsisi toteuttaa älypuhelimella, tilankäytön ja laskennallisen raskauden rajoitteet eivät välttämättä olisikaan rajoitteita. Lisäksi jos opettamista ei voi tehdä etukäteen, opettamiseen liittyvä laskenta voisi olla mahdollista toteuttaa pilvilaskennan keinoilla.

Tulevaisuudessa voisi olla kiinnostavaa tietää, miten hyvin neuroverkot käytännössä tunnistavat lintuja. Tämän voisi saada selville kilpailuttamalla ihmisiä ja neuroverkkoja keskenään. Tutkielmassa tarkastellun aineiston perusteella automaattinen lintujen äänten tunnistus ei tällä hetkellä ole realistista älypuhelimilla, sillä niiden laskennallinen teho, akunkesto ja tallennustila ovat rajalliset neuroverkkojen opettamista ajatellen. Goëau ym. (2017) näkevät, että tulevaisuudessa laskennan raskauden ongelmaan voisi olla mahdollista vastata esimerkiksi tiedon tislamisella (engl. knowledge distillation) (Hinton, Vinyals & Dean, 2015). Toinen heidän ehdottama tapa on tehokkaampien neuroverkkoarkkitehtuurien sekä opetusproseduurien suunnittelu. Lisäksi neuroverkkojen opettaminen voisi olla mahdollista joko toteuttaa etukäteen tietokoneella tai opettaa neuroverkkoa pilvilaskennan keinoin, jolloin älypuhelimien tallennustila ja laskennalliset rajoitteet eivät olisi ainakaan niin suuri rajoite.

4 YHTEENVETO

Tässä kandidaatintutkielmassa on tutkittu keinotekoisien neuroverkkojen hyödyntämistä automaattisessa lintujen äänten tunnistamisessa. Tavoitteena on ollut esitellä neuroverkkoja yleisellä tasolla ja tarkastella esimerkkien kautta miten neuroverkkoja voidaan hyödyntää lintujen äänten tunnistamisessa, sekä vastata tutkimuskysymykseen: *miten neuroverkkoja voidaan hyödyntää lintujen äänten tunnistamisessa?*

Johdannon jälkeen ensimmäisessä sisältöluvussa on esitelty neuroverkkoja ja niihin liittyviä käsitteitä, kuten neuronien ja perseptronien sekä painoa, vakiotermejä ja neuronien tulosta. Lisäksi on käsitelty aktivointifunktioita sekä neuroverkkojen opettamista, minkä yhteydessä on avattu neuroverkkojen opettamiseen liittyviä käsitteitä, virhefunktioita, gradienttimenetelmää sekä vastavirta-algoritmeja. Lopuksi ensimmäisessä sisältöluvussa on esitelty koneoppimiseen liittyviä ongelmia, joita ovat ali- ja ylisovittaminen sekä neuroverkkojen laskennalliseen raskauteen liittyvät ongelmat.

Toisessa sisältöluvussa on käsitelty esimerkkien avulla sitä, miten neuroverkkoja on hyödynnetty lintujen äänten tunnistamisessa. Esimerkkien aineistona on käytetty raportteja lintujen tunnistamisen kilpailuista. Ensiksi on käsitelty automaattiseen lintujen äänten tunnistamiseen liittyviä äänentunnistuksen haasteita, minkä jälkeen on vertailtu lintujentunnistuskilpailujen raportteja. Vertailussa on tarkasteltu audiosignaalin käsittelyä, datan kasvattamismenetelmiä, neuroverkkoarkkitehtuuria, metadatan hyödyntämistä ja lintujentunnistuskilpailujen raporttien tuloksia. Toisen sisältöluvun lopussa on käsitelty luvun aikana syntyneitä johtopäätöksiä.

Tutkielmassa on tullut esille, että neuroverkkojen avulla voidaan hyödyntää suuria datamääriä ja niitä sovelletaan monella eri tieteenalalla (LeCun ym. 2015). Neuroverkkoja sovelletaan esimerkiksi

puheentunnistuksessa (Ting, Yong & Mirhassani, 2013), liikennemerkkien tunnistamisessa (Zhu, Zhang & Lu, 2006), go-lautapelin mullistaneessa AlphaZero-tekoälyssä (Silver ym., 2017). Viime vuosina on osoitettu, että neuroverkot voivat toimia puheentunnistuksen lisäksi myös lintujen äänten tunnistuksessa (Goëau ym., 2016; Goëau ym., 2017). Olemassa olevista teknologioista parhaiten toimivat automaattisessa lintujen äänten tunnistamisessa konvoluutioneuroverkot (Goëau ym., 2016; Goëau ym., 2017).

Neuroverkkojen soveltamisen tehokkuus ja tarkkuus automaattisessa lintujen äänten tunnistamisessa on käytännön tasolla vielä rajallista ja monella muulla alalla ollaan kehityksessä automaattista lintujen tunnistusta edellä (Goëau ym., 2017; Goodfellow ym., 2016). Tämä johtuu ainakin kahdesta syystä. Ensimmäinen syy on se, että vaikka neuroverkot ovat tehokkaita, niiden opettaminen vaatii silti suhteellisen paljon laskutehoa (Goodfellow ym., 2016). Toinen mahdollinen syy on se, että neuroverkkoja on hyödynnetty lintujen äänten tunnistuksessa vasta vähän aikaa. BirdCLEF-kilpailuista ensimmäisenä käytettiin neuroverkkoja vuonna 2015, mutta neuroverkot eivät näkyneet silloin vielä parhaimmissa tuloksissa (Goëau ym., 2015). Seuraavana vuonna BirdCLEF 2016 -kilpailun voitti Sprengelin ym. (2016) konvoluutioneuroverkkopohjainen ratkaisu. Goëau ym. (2016) raportoivat, että parannus vuoden 2015 tuloksista seuraavan vuoden tuloksiin oli heidän tietoonsa ensimmäinen yhtä suuri parannus laajamittaisen bioakustisen luokittelun alueella.

Vaikka suuret datamäärät ja laskentatehon raskaus asettavatkin rajoitteita neuroverkkojen opettamiselle esimerkiksi älypuhelimien avulla, tutkielman aineistosta ei selvinnyt, voisiko neuroverkkoja joko opettaa etukäteen tai suorittaa opettamista koskeva laskenta pilvilaskennan avulla. Jos neuroverkkojen opettamisen tarve voitaisiin ulkoistaa, tilankäytön ja laskennallisen raskauden rajoitteet eivät välttämättä olisikaan rajoitteita.

Tutkielmassa on esitetty, että neuroverkkojen ilmaantuminen automaattisen lintujen tunnistamisen pariin on ollut suuri kehitysaskel lintujen äänten tunnistamisessa ja ehkä myös muunlaisessa luonnon äänten luokittelussa. Vaikka neuroverkkojen hyödyntämisessä käytännön tasolla on parantamisen varaa, niiden mahdollisuudet lintujen äänten tunnistuksessa vaikuttavat silti lupaavilta. Mikäli neuroverkkojen hyödyntäminen lintujen tunnistuksessa osoittautuu myöhemmin käytännölliseksi ja tehokkaaksi esimerkiksi niin, että neuroverkkoja voidaan soveltaa älypuhelimissa, lienee odotettavissa parannuksia esimerkiksi luonnonsuojelun kannalta ja harrastajien näkökulmasta. Neuroverkkoavusteiset kokeilut lintujen äänten tunnistamisessa tuovat mahdollisesti myös yleisesti tärkeää tietoa äänentunnistukseen. Näin ollen neuroverkkojen hyödyntäminen automaattisessa lintujen tunnistuksessa äänen perusteella voi hyödyttää myös muita tieteenaloja tai sovelluskohteita.

Neuroverkkoavusteisen lintujen äänten tunnistamisen jatkotutkimuskohteita ovat ainakin tehokkaampien arkkitehtuurien kehittäminen, jotta neuroverkkojen opettaminen olisi tehokkaampaa (Goëau ym., 2017). Lisäksi voisi olla kiinnostavaa kilpailuttaa ihmisiä ja neuroverkkoja lintujen tunnistamisessa, jotta saataisiin selville se, miten neuroverkot suhteutuvat lintujen tunnistuksessa ihmisiin. Jotta neuroverkkoja voitaisiin käyttää esimerkiksi älypuhelimella, voisi olla hyödyllistä selvittää pilvilaskennan mahdollisuuksia neuroverkkojen opettamisessa niin, että opettamista ei tarvitsisi toteuttaa älypuhelimella.

LÄHTEET

- Abdel-Hamid, O., Mohamed, A., Jiang, H., Deng, L., Penn, G. & Yu, D. (2014). Convolutional neural networks for speech recognition. *Audio, Speech, and Language Processing, IEEE/ACM Transactions on*, 22(10), 1533–1545.
- artificial intelligence. (2018). Merriam-Webster.com. Haettu 15.5. osoitteesta <https://www.merriam-webster.com/dictionary/artificial%20intelligence>
- BirdLife Suomi ry (2018a). Jäsenyhdistykset. Haettu 20.5.2018 osoitteesta <https://www.birdlife.fi/jarjesto/jasenyhdistykset/>
- BirdLife Suomi ry (2018b). Linnuston seuranta Suomessa. Haettu 25.5.2018 osoitteesta <https://www.birdlife.fi/suojelu/seuranta/valtakunnallinen/>
- Bishop, C.M. (2006). *Pattern recognition and machine learning*. Berkeley, Kalifornia, Yhdysvallat: Springer.
- Bramer, M. (2007). *Principles of data mining* (2. uud. painos). Portsmouth, Iso-Britannia: Springer.
- Briggs, F., Lakshminarayanan, B., Neal, L., Fern, X.Z., Raich, R., Hadley, S.J.K.,...Betts, M. G. (2012). Acoustic classification of multiple simultaneous bird species: A multi-instance multi-label approach. *The Journal of the Acoustical Society of America*, 131(6), 4640.
- Cireşan, D., Meier, U., Masci, J. & Schmidhuber, J. (2012). Multi-column deep neural network for traffic sign classification. *Neural Networks*, 32, 333–338.
- Despois, J. (2018, 24. huhtikuuta). How smartphones manage to handle huge Neural Networks. Haettu 25.5.2018 osoitteesta <https://heartbeat.fritz.ai/how-smartphones-manage-to-handle-huge-neural-networks-269debc243d>
- Dougherty, G. (2013). *Pattern recognition and classification: an introduction*. Camarillo, Kalifornia, Yhdysvallat: Springer.
- Fazekas, B., Schindler, A., Lidy, T. & Rauber, A. (2017). *A multi-modal deep neural network approach to bird-song identification*. Itävalta, Wienin teknillinen korkeakoulu.

- Goëau, H., Glotin, H., Vellinga, W., Planqué, R. & Joly, A. (2016). LifeCLEF Bird Identification Task 2016: The arrival of Deep learning. Teoksessa: *Working Notes of CLEF 2016 - Conference and Labs of the Evaluation forum* (440–449). Evora, Portugali, syyskuu 2016.
- Goëau, H., Glotin, H., Vellinga, W., Planqué, R. & Joly, A. (2017). LifeCLEF Bird Identification Task 2017. Teoksessa: *Working notes of CLEF 2017 - Conference and Labs of the Evaluation Forum* (1–9). Dublin, Irlanti, syyskuu 2017.
- Goëau, H., Glotin, H., Vellinga, W., Planqué, R., Rauber, A. & Joly, A. (2014). LifeCLEF Bird Identification Task 2014. Teosessa: *CLEF: Conference and Labs of the Evaluation Forum*. Sheffield, Iso-Britannia, syyskuu 2014.
- Goëau, H., Glotin, H., Vellinga, W., Planqué, R., Rauber, A. & Joly, A. (2015). LifeCLEF Bird Identification Task 2015. Teoksessa: *Working Notes of CLEF 2015 - Conference and Labs of the Evaluation forum* (1391). Toulouse, Ranska, syyskuu, 8-11, 2015.
- Goodfellow, I., Bengio, Y. & Courville, A. (2016). Deep learning. MIT Press. Haettu osoitteesta <http://www.deeplearningbook.org/>
- Guo, Y., Liu, Y., Oerlemans, A. Lao, S., Wu, S. & Lew, M.S. (2016). Deep learning for visual understanding: a review. *Neurocomputing*, 187, 27–48.
- Haykin, S. (2005). *Neural networks, a comprehensive foundation* (9. uud. painos). Hamilton, Ontario, Kanada: Pearson Education.
- Hinton, G., Vinyals, O. & Dean, J. (2015). Distilling the knowledge in a neural network. Haettu 25.5.2018 osoitteesta <https://www.cs.toronto.edu/~hinton/absps/distillation.pdf>
- ImageCLEF. (2018). BirdCLEF 2018. Haettu 20.5.2018 osoitteesta <http://www.imageclef.org/node/230>
- IUCN (2018). The IUCN Red List of Threatened Species. Haettu 20.5.2018 osoitteesta <https://www.iucn.org/theme/species/our-work/iucn-red-list-threatened-species>
- Jancovic, P. & Munevver, K. (2015). Acoustic Recognition of Multiple Bird Species Based on Penalized Maximum Likelihood. *Signal Processing Letters, IEEE*, 22(10), 1585–1589.

- Julkaisufoorumi. (2017, 12. kesäkuuta). Julkaisufoorumi. Haettu 22.5.2018 osoitteesta <http://www.julkaisufoorumi.fi/fi/julkaisufoorumi>
- Kahl, S., W, Wilhelm-Stein, T., Hussein H., Klinck, H., Kowerko, D., Ritter, M & Eibl, M. (2017). *Large-scale bird sound classification using convolutional neural networks*. Saksa, Chemnitzin teknillinen korkeakoulu.
- Kogan, A. J. & Margoliash, D. (1998). Automated recognition of bird song elements from continuous recordings using dynamic time warping and hidden Markov models: A comparative study. *The Journal of the Acoustical Society of America*, 103(4), 2185.
- Kraniauskas, P. (1994). A plain man's guide to the FFT. *Signal Processing Magazine, IEEE*, 11(2), 24–35.
- Kwan, C., Ho, K., Mei, G., Ren, Z., Zhang, Y., Lao, D.,...,Kwan, C. (2006). An Automated Acoustic System to Monitor and Classify Birds. *Eurasip Journal on Applied Signal Processing*, 2006(3), 1–19.
- Lasseck, M. (2013). *Bird song classification in field recordings: winning solution for NIPS4B 2013 competition*. Saksa, Berliinin luonnontieteellinen museo.
- Lecun, Y., Bengio, Y. & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436–444.
- Lecun, Y., Bottou, L., Bengio, Y. & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278–2324.
- Lee, C., Han, C. & Chuang, C. (2008). Automatic classification of bird species from their sounds using two-dimensional cepstral coefficients. *Audio, Speech, and Language Processing, IEEE Transactions on*, 16(8), 1541–1550.
- Lee, C., Hsu, S., Shih, J. & Chou, C. (2013). Continuous Birdsong Recognition Using Gaussian Mixture Modeling of Image Shape Features. *Multimedia. IEEE Transactions on*, 15(2), 454–464.
- Leibniz, G. W. (1684). Nova methodus pro maximis et minimis, itemque tangentibus, quae nec fractas, nec irrationales quantitates moratur, et singulare pro illis calculi genus. *Acta Eruditorum*, 467–473.
- Lemon, R. E. (1975). How birds develop song dialects. *The Condor*, 77(4), 385–406.

- Linnainmaa, S. (1970). *The representation of the cumulative rounding error of an algorithm as a Taylor expansion of the local rounding errors* (Pro gradu – tutkielma). Helsingin yliopisto.
- Liu, W., Wang, Z., Liu, X., Zeng, N. Liu, Y. & Alsaadi, F.E. (2017). A survey of deep neural network architectures and their applications. *Neurocomputing*, 234, 11–26.
- Manning, C.D., Raghavan, P. & Schütze, H. (2009) *An introduction to information retrieval*. Cambridge, Iso-Britannia: Cambridge University Press. Haettu 20.5. osoitteesta <https://nlp.stanford.edu/IR-book/pdf/irbookonlinereading.pdf>
- Martinsson, J. (2017). Bird species identification using convolutional neural networks (Pro gradu –tutkielma). Department of Computer Science and Engineering. Chalmersin teknillinen korkeakoulu, Göteborgin yliopisto.
- McCulloch, W. & Pitts, W. (1990). A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biology*, 52(1), 99–115.
- Mcilraith, A.L. & Card, H.C. (1995). Birdsong recognition with DSP and neural networks. Teoksessa: *IEEE WESCANEX 95. Communications, Power, and Computing. Conference Proceedings., IEEE*, 2 (409–414). Manitoba, Winnipeg, Yhdysvallat.
- Newman, M. & Girvan, M. (2004). Finding and evaluating community structure in networks. *Physical Review E*, 69(2).
- Nielsen, M.A. (2015). *Neural networks and deep learning*. Determination Press.
- Olah, C. (2014, 8. heinäkuuta). Conv nets: a modular perspective. Haettu 21.5. osoitteesta <https://colah.github.io/posts/2014-07-Conv-Nets-Modular/>
- Olah, C. (2015, 31. elokuuta). Calculus on computational graphs: backpropagation. Haettu 20.5. osoitteesta <https://colah.github.io/posts/2015-08-Backprop/>
- Opetushallitus (2014). *Perusopetuksen opetussuunnitelman perusteet 2014*. (4. painos). Helsinki: Next Print Oy. Haettu 22.5.2018 osoitteesta http://oph.fi/download/163777_perusopetuksen_opetussuunnitelman_perusteet_2014.pdf

- Pitkäranta, J. (2015). *Calculus fennicus*. Painopaikka Picaset, Helsinki: Avoimet oppimateriaalit ry.
- Puri, M., Solanki, A., Padawer, T., Tipparaju, S.M., Moreno, W.A. & Pathak, Y. (2016). Chapter 1 - introduction to artificial neural network (ANN) as a predictive tool for drug design, discovery, delivery, and disposition: basic concepts and modeling. Teoksessa *Artificial Neural Network for Drug Design, Delivery and Disposition*, Academic Press, (3–13). Boston, Massachusetts, Yhdysvallat: Academic Press.
- Relander, T. (2013). *KELTATULKKU, VALASLINTU JA LAULUKOTKA - Lintujen lajitunnistus alakoulussa* (Pro gradu –tutkielma). Jyväskylän yliopisto.
- Ricketts, T., Brooks, T., Hoffmann, M., Stuart, S., Balmford, A., Purvis, A.,... Whiteman, C. (2005). *Ecosystems and Human Well-being: Current State and Trends, Volume 1*. NW, Washington: Millennium Ecosystem Assessment.
- Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6), 386–408.
- Schmidhuber, J. (2015). Deep learning in neural networks: an overview. *Neural networks : the official journal of the International Neural Network Society*, 61, 85.
- Selin, A., Turunen, J. & Tantt, J.T. (2006). Wavelets in recognition of bird sounds. *EURASIP Journal on Advances in Signal Processing*, 2006(1).
- Sermanet, P., Kavukcuoglu, K., Chintala, S. & LeCun, Y. (2013). Pedestrian detection with unsupervised multi-stage feature learning. Teoksessa *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on* (3626–3633). Portland, Oregon, Yhdysvallat.
- Sevilla, A. & Glotin, H. 2017. *Audio Bird Classification with Inception-v4 extended with Time and Time-Frequency Attention Mechanisms*. Ranska, Toulonin yliopisto
- Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A.,..., Hassabis, D. (2017). Mastering the game of go without human knowledge. *Nature*, 550(7676), 354.
- Sodhi, N.S. & Ehrlich, P.R. (2011). *Conservation Biology for All*. New York: Oxford University Press.

- Somervuo, P. & Härmä, A. (2003). Analyzing bird song syllables on the self-organizing map. Teoksessa *Workshop on Self-Organizing Maps (WSOM'03)*.
- Sprengel, E., Jaggi, M., Kilcher, Y. & Hofmann, T. (2016). Audio based bird species identification using deep learning techniques. CLEF 2016 Working Notes. Zürich, Sveitsi: Eidgenössische Technische Hochschule (ETH).
- Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. & Salakhutdinov, R. (2014). *Journal of Machine Learning Research*, 15, 1929–1958.
- Statista (2016). Number of participants in birdwatching in the United States from 2006 to 2016 (in millions)*. Haettu 20.5.2018 osoitteesta <https://www.statista.com/statistics/191207/participants-in-birdwatching-in-the-us-since-2006/>
- Taigman, Y., Yang, M., Ranzato, M. & Wolf, L. (2014). DeepFace: closing the gap to human-level performance in face verification. Teoksessa *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on* (1701–1708). Columbus, Ohio, Yhdysvallat.
- tekoäly. (2018). Tieteen termipankki. Haettu 15.5. osoitteesta <http://tieteentermipankki.fi/wiki/Filosofia:teko%C3%A4ly>
- Thomas, J. A., Telfer, M. G., Roy, D. B., Preston, C. D., Greenwood, J.J.D., Asher, J.,...,Lawton, J.H. (2004). Comparative losses of british butterflies, birds, and plants and the global extinction crisis. *Science*, 303(5665), 1879–81.
- Tirri, R., Lehtonen, J., Lemmetyinen, R., Pihakaski, S. & Portin, P. (2006). *Biologian sanakirja*. Helsinki: Otava.
- Ting, H., Yong, B. & Mirhassani, S.M. (2013). Self-adjustable neural network for speech recognition. *Engineering Applications of Artificial Intelligence*, 26(9), 2022–2027.
- Trawicki, M.B., Johnson, M.T. & Osiejuk, T.S. (2005). Automatic Song-Type Classification and Speaker Identification of Norwegian Ortolan Bunting (*Emberiza Hortulana*) Vocalizations. *2005 IEEE Workshop on Machine Learning for Signal Processing*, 277–282.
- Tyagi, H., Hegde, R.M., Murthy, H.A. & Prabhakar, A. (2006). Automatic identification of bird calls using Spectral Ensemble Average Voice Prints. Teoksessa *Signal Processing Conference, 2006 14th European Signal Processing Conference* (1–5).

- Van Veen, F. (2016, 14. syyskuuta). The neural network zoo. Haettu 20.5. osoitteesta <http://www.asimovinstitute.org/neural-network-zoo/>
- Wan, L., Zeiler, M., Zhang, S., LeCun, Y. & Fergus, R. (2013). Regularization of neural networks using DropConnect. Teoksessa *30th International Conference on Machine Learning, ICML 2013* (2095–2103). Atlanta, Georgia, Yhdysvallat, elokuu 16-21, 2013.
- Xeno-canto Foundation. 2018. Sharing bird sounds from around the world. Haettu 23.5.2018 osoitteesta <https://www.xeno-canto.org/>
- Zhu S., Zhang Y., Lu X. (2006) Detection for triangle traffic sign based on neural network. Teoksessa Wang J., Yi Z., Zurada J.M., Lu BL., Yin H. (toim.) *Advances in Neural Networks - ISNN 2006, Lecture Notes in Computer Science*, vol 3973. Berlin, Heidelberg: Springer-Verlag.