Akustisen luokittelijan optimointi

Turun yliopisto
Tietotekniikan laitos
Diplomityö
Robotiikka ja autonomiset järjestelmät
Joulukuu 2024
Lassi Lehtinen

TURUN YLIOPISTO
Tietotekniikan laitos

Lassi Lehtinen: Akustisen luokittelijan optimointi

Diplomityö, 55 s., 4 liites.
Robotiikka ja autonomiset järjestelmät
Joulukuu 2024

Merenkulun liikenne tuottaa melua, joka kantautuu pitkän matkan päähän. Kohteiden tunnistamiseen on aiemmin käytetty opearaattoria. Reunalaskennalla ja konekuuntelulla luokittelu toteutetaan ympäristössä, jossa muistiresurssit ja laskentateho ovat rajalliset. Tässä tutkielmassa esitetään moderneihin syväoppimismenetelmiin perustuva yksiulotteisen signaalin luokitin. Alusten konemelun luokitukseen sovellettiin 1DCNN, SBCNN ja YAMNet -syväneuroverkkomalleja. Luokittimet saavuttivat keskimäärin 85.5% tarkkuuden käytämällä yhden sekunnin syötevektoria. Näistä malleista yksiulotteinen konvoluutio-operaatio yhdessä gammatoonisuodinten kanssa osoittautuivat vakuuttavimmaksi menetelmäksi.

Asiasanat: reunalaskenta, tekoäly, konekuuleminen, koneoppiminen, syväoppiminen, akustiikka

UNIVERSITY OF TURKU
Department of Computing

Lassi Lehtinen: Akustisen luokittelijan optimointi

Master of Science (Tech) Thesis, 55 p., 4 app. p.
Robotics and Autonomous Systems
December 2024

Maritime traffic and activity is perceived over long distances in water medium. Edge computing is a persuasive technology in replacing human operators in target detection. With specialized deep learning models targets can be detected and classified in near real-time. In this thesis is presented a evaluation of 1DCNN, SBCNN and YAMNet deep learning models for classifying ship radiated noise in underwater scenario. An average classification accuracy of 85.5% was achieved in determining the size class of a target ship. Out of all models the 1DCNN model proved to be the most sutiable for the task.

Keywords: edge computing, artificial intelligence, machine hearing, machine learning, deep learning, acoustics

Sisällys

Kuvat

Taulukot

Terminologia

Piirre feature

Ominaisuus label

Siirto-oppiminen transfer learning

Päättely inference

Reunalaskenta edge computing

Reunalaite edge device

Kvantisointi (tekoälymalli) quantization

Karsinta (tekoälymalli) pruning

Valikointitaso (tekoälymalli) pooling layer

Piirremuodostus feature engineering

Piirrevalinta feature selection

Käänteisajo backpropagation

Läpiajo forward propagation

Oppimiskerroin learning rate

STFT short-time fourier transform

Poiskytkentä Dropout

1 Johdanto

Liikenne pintavedessä ja sen tuottama melu kasvavat globaalisti. Merialuiden liikenne kattaa rahtialukset, matkustajaristeilijät, henkilökohtaiset vesiajoneuvot, sotilasalukset sekä miehittämättömät alukset. Pinta-alusten konetekniikka tuottaa laajakaistaista kohinaa, joka kantautuu vedessä pitkien etäisyyksien päähän. Vedenalaisten kohteiden tunnistus on haastava ja pitkään tutkittu aihe. Ihmisoperaattorin koulutus tunnistamistehtävään on työlästä, tunnistustyö vaatii keskittymistä ja operaattori on erehtyväinen työssään. Koneellinen signaalintunnistus on kehittynyt yhdessä koneoppimismetetelmien ja suurien datalähteiden kanssa. Veden fysikaalisista ominaisuuksista johtuen ääniaallot kantautuvat vesipatsaassa kymmenien kilometrien päähän. Matalassa vedessä vedenpinnan ja pohjan rajapintavuorovaikutukset ovat voimakkaita, mikä tekee signaalien luokittelusta haastavaa perinteisin menetelmin. Reunalaskenta on tekoälyteknologiassa yleistyvä trendi, jossa tekoälymallien päättely tapahtuu lähellä antureita. Tekoälymallit koulutetaan isommassa pilviympäristössä, josta koulutetut mallit siirretään reunalaitteille, jossa luokitus tapahtuu suljetusti. Suuret tekoälymallit ovat suuruusluokaltaan 45GB. Mallit vaativat paljon muistia ja laskentatehoa toimiakseen reaktiivisesti nykyhetkessä. Erilaisilla tekniikoilla, kuten kvantisoinnilla ja karsinnalla olemassa olevien mallien kokoaa supistetaan vähentäen mallin tarkkuutta vain nimellisesti. Supistetut mallit mahtuvat pienempään järjestelmäpiirin muistilaitteelle, jossa muistikapasiteetti on satojen kilotavujen suuruusluokassa.

Reunalaitteilla tarvitaan pienempiä ja erikoistuneita malleja, jotka soveltuvat parhaiten äänten luokitteluun ympäristössä, jossa resurssit ovat vähäiset, ja jossa järjestelmä ei voi ulkoistaa luokittelutehtävää isommalle palvelimelle. Autonomiset vedenalaiset miehittämättömät alukset sekä autonomiset laivat kykenevät haastavampiin tunnistutehtäviin syvämallien tuella. Potentiaalinen käyttökohde olisi älykäs meri, jossa vuorovaikutteiset laitteet muodostavat kokonaistilannekuvaa merialueista reaaliajassa.

1.1 Ongelma ja tavoitteet

Tässä tutkielmassa valikoidaan ja koulutetaan luokitinmalli, joka erottaa aluskohtaisia ominaisuuksia vedenalaisesta konemelusta. Seuraavat tutkimuskysymykset valittiin tarkasteluun:

- Kuinka hyvin olemassa olevat tekoälymallit soveltuvat vedenalaisten kohteiden tunnistukseen?
- Mikä on erikoistuneen mallin ja suuren mallin [1] suorituskyvyn ero?

Jotta kysymyksiin voidaan vastata, tarkasteluun otetaan aiemmin tutkittuja ääniluokittelumalleja ja verrataan niiden suorituskykyä isolla datalla koulutettuun malliin. Tarkastellaan akustisen mittausdatan johdannaispiirteitä ja niiden soveltamista luokitustehtävään. Tavoitteena on tutkia, mitkä tekijät audiodatassa johtavat kohdeluokkien separoitumiseen. Tapaustutkimukseen valitaan kaksi erikosituttua luokitinmallia ja googlen kehittämä audioluokitin YAMNet. Tarkastelussa on erityisesti, miten yksiulotteisten konvoluutio-operaatioiden käyttö vaikuttaa luokitustuloksiin. Menetelmä on saanut inspiraatiota kuuloaistin ominaisuuksista ja mallinnuksesta. Mallien suorituskykyä arviodaan luokitustarkkuuden perusteella. Tutkimuskysymyksiin pyritään tuomaan vastauksia datakeskeisillä menetelmillä. Luo-

[1] (erikoistunut malli: <1'000'000 parametria, suuri malli: >1'000'000 parametria)

kittelun kannalta keskeisiä ilmiötä tarkastellaan data-aineston tuella. Aineisto on kokoelma vedenalaisia audiomittauksia. Ongelmaa lähtestytään tutustumalla akustisen signaalin ominaisuuksiin, datajoukkojen rakenteisiin ja piirteiden irroitukseen. Tapaustutkimuksessa yhdistellään aineistoja ja menetelmiä, joita ei ole aiemmin sovellettu yhdessä tutkimuksessa.

1.2 Työn rakenne

Kappaleessa 2 hahmotellaan luokitusympäristöä. Tarkastelussa on kohteen ja mittalaitteiden asetelma sekä siihen liittyvät haasteet, audiosignaalin esitysmuodot sekä alusten eri äänenlähteistä.

Kappaleessa 3 esitetään luokittimen teoreettisia käsitteitä. Aiheina ovat vedenalainen akustiikka, psykoakustikan ilmiöt, koneoppiminen, syväoppiminen, mallien rakenteet ja arvointimetriikat.

Kappaleessa 4 kuvataan mallin koulutusprosessi. Esitellään käytetyt data-aineistot ja data-ainesiton jakaminen mallin koulutusta varten.

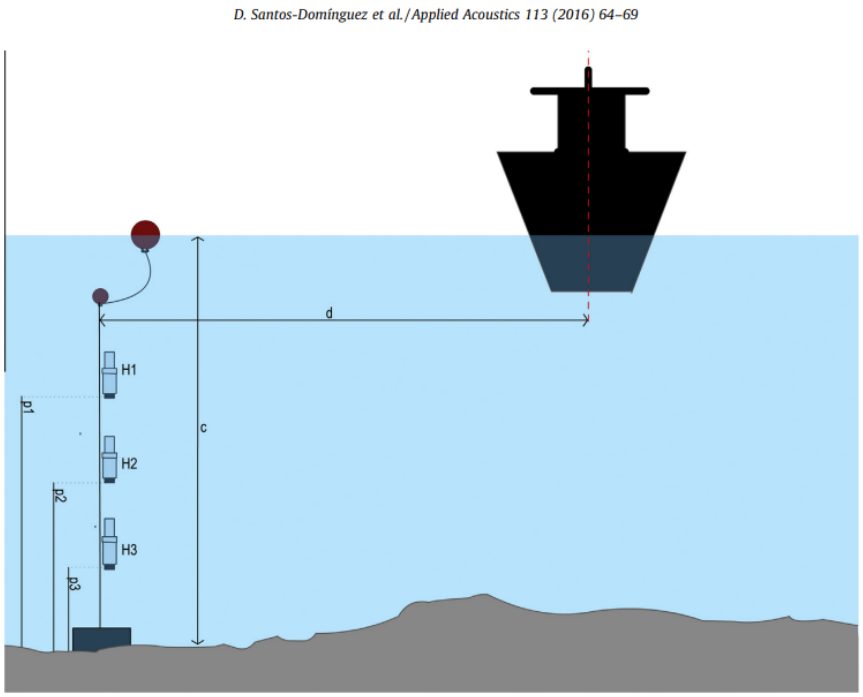
Kappaleessa 5 arvioidaan luokittimen tuloksia ja suorituskykyä, tulosten luotettavuutta ja mallien sovellettavuutta.

Kappaleessa 6 kootaan tulokset ja yleiset havainnot.

2 Ongelman kuvaus

2.1 Asetelma

Pinta-lausta havainnoidaan yhdellä mittalaitteella. Kuvassa 2.1 on esitetty tilanne, jossa kohde pyritään tunnistamaan.



Kuva 2.1: Mittaustilanteen havainnollistus [1]

Mittalaite on sijoitettu väliveteen, ja äänenpaineen aiheuttamat erot tallennetaan hydrofonilla. Havainnointi tehdään yhdellä hydrofonilla. Pinta-alus kulkee mittalaitteen ohitse oletetusti suoraa pitkin. Tarkkaa etäisyyttä kohteeseen ei tunne-

ta. Signaalin muoto vastaanottimella on riippuvainen siitä, missä pisteessä ja millä korkeudella havainto tehdään. Havaintopiste on kiinnitetty mittauksen ajan. Ennen ja jälkeen kohteen läsnäoloa havaintopisteessä koetaan ympäristön sekoittunut taustamelu. Signaali intensiteetti kasvaa kohteen lähestyessä havaintopistettä. Tämä asettaa tunnistamisen ensimmäisen haasteen. Vedessä korkeat taajuudet vaimenevat matalia taajuuksia enemmän. Tästä syystä konemelun taajuusvaste muuttuu myös etäisyyden funktiona. Toisin sanoen mitä kauempana kohde on, sitä enemmän signaaliin vaikuttaa ympäristön melu. Tästä syystä mallin opetuksessa koko mittausta ei voida pitää samanarvoisena. Luokitin-algoritmi vastaanottaa digitalsoidun signaalin ja antaa luokituksen tuloksen.

Oletetaan, että mittausympäristössä sekä mittalaittessa esiintyy eriliaset siirtofunktiot. Luokittimen tulisi kyetä oppimaan syötteestä yleisiä piirteitä, eikä taustaprosessien yksityiskohtia ja kohinaa. Koska konemelun tuottama signaali ei sisällä samalla tavalla tonaalisia rakenteita, kuten ihmisen puhe, sovelletaan tunnistukseen syväoppimismallia, joka kykenee tunnistamaan datasta komplekseja epälineaarisia suhteita. Huomioitavaa on, että jotkin koneperäiset äänet tuottavat harmoonisia rakenteita.

Eri koneoppimisen menetelmissä kohteiden tunnistamisessa on eroja. Konenäkemisessä kuvassa olevat kohteet peittävät toisensa, kun konekuulemisessa äänilähteet laskostuvat toistensa päälle. Kun kaksi eri äänenlähdettä ovat läsnä signaalissa samanaikasesti, niiden tunnistaminen on siis haasteellisempaa verrattuna tilanteeseen, jossa äänenlähteet ovat läsnä eriaikaisesti. Mittausympäristö ja -järjestelmä itsessään tuovat satunnaisuutta mittauksiin. *Audiokuva* paljastaa aika-taajuustasossa tapahtuvia hetkellisiä sekä hitaita muutoksia. Audiokuvalla tarkoitetaan jollakin menetelmällä luotua kaksiuloitteista signaaliesitystä yksiulotteisesta audiosignaalis-

ta (vrt. FFT, CQT, MFCC). Audiokuva valitaan näin sovelluskohteen mukaisesti. Optimoinnin kannalta tavoitellaan sellaista audiokuvaa, joka on pienin mahdollinen tavoiteltavan tarkkuuden aikaansaamiseksi. Kuvantunnistuksessa kohteeen läsnäolo tunnistetaan, sijaitsi kohde kuvassa missä tahansa. Audiokuvassa tunnistettava kohde on riippuvainen sen sijainnista taajuusakselilla. Jottei luokitin sitoisi kohteita tiukasti johonkin taajuuskaistaan, opetusdataa taajuusmoduloidaan, eli dataa augmentoidaan. Audiokuvan augmentoinnilla tuetaan mallin yleistymistä.

Datamalli

Perustellaan datakeskeisen mallin valitsemista. Vedenalaisten kohteiden tunnistamiseen liittyy useita ilmiöitä. Laajakaistainen konemelu syntyy kohteen mekaanisen rakenteen vuorovaikutuksista. Signaalilla on monimutkaisia spektrotemporaalisia ominaisuuksia, kuten polttomoottorin sytytysrytmin ja voimansiirron resonointi. Signaalitiessä osa värähteylystä resonoi kohteen rungossa ja johtuu muita taajuuksia voimakkaammin väliaineeseen. Signaalin vastaanottajalla havaitaan monitieheijastuksia sekä signaalin kanavoitumista eri vesikerroksien vaikutusesta. Vastaanottimen johtimet ovat alttiina häiriöille, kuten ylikuulumiselle ja virtalähteen häiriölle. Sen sijaan, että ilmiön prosessit pyritään mallintamaan tarkasti, tarkastellaan prosessin ulostuloa, ja yritetään löytää sopivia relaatiota havaintojen ja signaalin välille. Puhutaan datakeskeisestä tarkastelusta.

Datakeskeistä tarkastelua haastaa kattavien avoimen data-aineistojen puute. Useassa samankaltaista tunnistustehtävää tutkivassa lähteessä käytetty data-aineisto on ollut suljettu. Lähteissä sovelluskohteina ovat olleet koneoppimismenetelmien kehitys ja arviointi, sekä vedenalainen valvonta. Avoimet data-aineistot auttavat tulosten vertailussa keskenään. Alalla on julkaistu kaksi avointa pinta-alusten akustisista mit-

tauksista: ShipsEar [1] (2016) ja DeepShip [2] (2021). Näitä aineistoja on sovellettu useisiin tekoälyn menetelmien tutkimuksiin [3]–[5].

2.2 Audio ja luokittelu

Tässä luvussa tarkastellaan audiosignaalin taltioinnin ja eri esitysmuotojen suhtetta.

2.2.1 Digitaalinen signaali

Analogiset signaalit muunnetaan AD-muuntimella ja taltioidaan muistilaitteelle digitaalisessa vektorimuodossa $x_n = [x_1, x_2, \ldots x_n]$, jossa n on vektorin pituus. Digitalisoitu signaali voidaan esittää kokonaislukuina tai liukulukuina. Kokonaislukuja käsittelevä arkkitehtuuri on yksinkertaisempi ja halvempi. Liukulukujen aritmetiikka vaatii oman laskentayksikön, mutta tarjoaa enemmän tarkkuutta ja suuremman vaihteluvälin ennen kuin signaali saturoituu. Kokonaislukujen artimetiikassa on lisäksi otettava huomioon mahdolliset yli- ja alivuodot sekä pyöristysvirheet [6]. Signaalin muuntamisessa keskeisiä parametreja ovat bittisyvyys ja näytteenottotaajuus. Suurella bittisyvyydellä voidaan taltioida pieniä signaalin amplitudin muutoksia. Ikkunoinnilla signaali $x[n]$ pilkotaan n pituisiksi vektroreiksi. Ikkunoidut signaalit Fourier-muunnetaan, jolloin signaalista saadaan spektrogrammiesitys. Näitä kahta esitysmuotoa sovelletaan luokitukseen.

Äänenpaineen vaihteluerot ovat suuret. Ihmiskorvan kuuloaistin alin kynnys on 20 μPa (0dB SPL) ja kipukynnys yli 20 Pa (120dB SPL). Keskustelun (60dB SPL) ja katuporan (110dB SPL) välinen äänenpaineen ero on $1 : 1 \times 10^5$ [6]. Suuri dynaaminen vaihtelu puristetaan logaritmimuunnoksella:

$$dB = 10 * \log_{10} \frac{P}{P_0}$$

Amplitudieroista johtuvat erot halutaan jättää luokituksessa huomiotta. Tästä syystä sigaali *normeerataan*. Signaalin vaihteluala voidaan toteuttaa jollekin rajatulle välille. Toinen tapa on normalisoida käyttäen keskiarvoa ja keskihajontaa:

$$x[n] = \frac{x[n] - \mu}{\sigma} \tag{2.1}$$

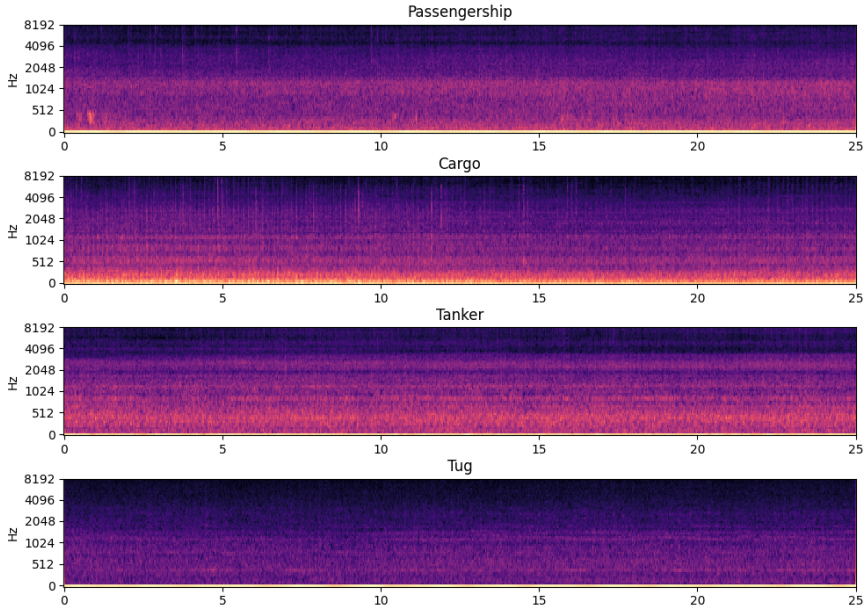
Aika-taajuustaso

Audioanalyysissä tarkastellaan signaalia aika- ja taajuustasossa. Diskreetti signaali muunnetaan aikatasosta taajuustasoon saadaan kaavalla:

$$X[k] = \sum_{n=0}^{N-1} x[n] \cdot e^{-j2\pi \frac{kn}{N}} \tag{2.2}$$

Perättäisiä FFT-muunnoksia pinotaan, jolloin saadaan matriisi $A_{n \times m}$, jossa pystyvektorit ovat taajuuskomponentteja ja vaakavektorit ajanhetkiä. Muunnosta kutsutaan lyhytkestoiseksi Fourier-muunnokseksi (*engl. STFT*). Matriisi A typistetään pienemmäksi, jolloin saadaan Mel-log-spektrogrammi (*myöh. mel-log*). Mel-log on yleisesti käytetty äänipiirre. Muunnos tehdään joukolla kolmionmuotoisia kaistanpäästösuotimia (*kts. suodipankit*). Muunnokselle valitaan lohkon pituus, ikkunointifuktio, sekä lohkojen lomitus. Lomitus ilmaistaa prosentteina, jossa 50% vastaa puolittaista lomitusta. Lohkot muunnetaan Fourier-muunnoksella, ja spektri suodatetaan mel-suodinpankilla. Tällä tavalla saatuja pinta-alusten spektrogrammeja on esitetty kuvassa 2.2.

Konemelu koostuu usein laajakaistaisista kohina-alueista. Pinta-aluksen äänekkäimmät osat ovat: voimansiirto ja moottoritekniikka, potkuri ja kavitaatio. Nämä ääänilähteet havaitaan eri taajuuskaistoilla. Moottori resonoi käydessää rungon välityksellä veteen, mikä havaitaan matalissa taajuuksissa (<100Hz). Nopeasti pyöriessä potkuri tuottaa fundamentaalitaajuuden ja useita harmoonisia äänek-



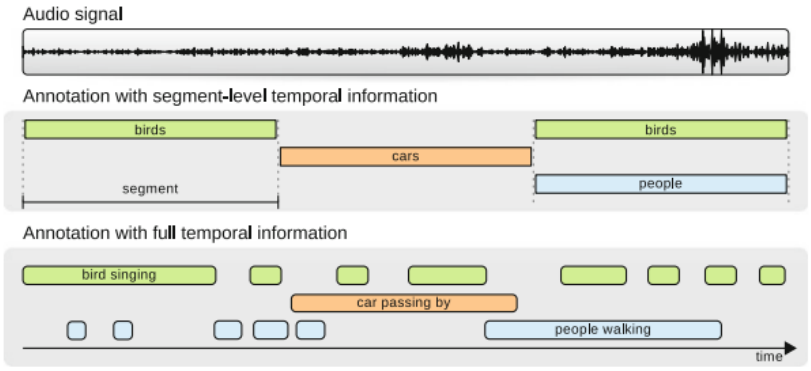
Kuva 2.2: Usean alustyypin spektrogrammi

siä (100-1000Hz). Kavitaatio havaitaan korkeissa taajuuksissa (>4kHz) [5]. Fourier-muunnoksen tarkkuus on lineaarinen kaikille taajuusalueille. Pinta-alusten yksilölliset piirteet, eli rungon koko, potkurin koko ja kulmanopeus sekä voimansiirto havaitaan keskimatalilla taajuuksilla. Lineaarinen muunnos tuottaa tasaisen resoluution, mikä ei ole täysin perusteltua alusten luokitteluun, sillä korkeilla taajuuskaistolla havaitaan kohinaa ja alustyyppeihin liittymättömiä epäolennaisia piirteitä. Epälineaarinen resoluutio saavutetaan eri muunnoksilla, kuten suodinpankeilla, tai aallokemuunnoksella.

2.2.2 Audiodatan annotoinnista

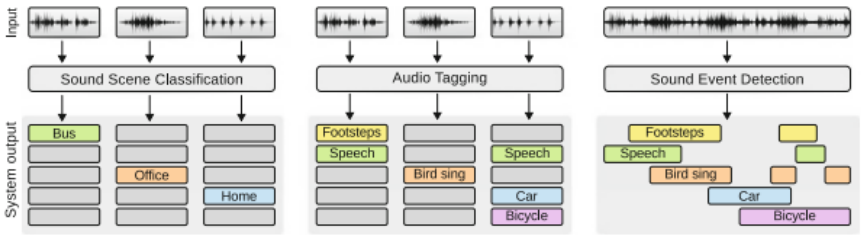
Ennen kuin malli voidaan kouluttaan, data-aineisto annotoidaan sellaiseen muotoon, että malli voi eritellä kohteet toisistaan syötteeseen perustuen. Syväoppimismallien ohjatussa opetuksessa tarvitaan suuret määrät annotoitua dataa. Kohteet audiodatassa jaetaan usein segmentteihin, joilla tarkoitetaan isompia alueita, joissa äänitoi-

mintaa esiintyy. Äänitapahtumien hetkittäiset päällekkäisyydet jäävät tällöin huomiotta. Audion annotoinnilla onkin määritelty eri tarkkuustasoja, jotka on kuvattu kaaviossa 2.3.

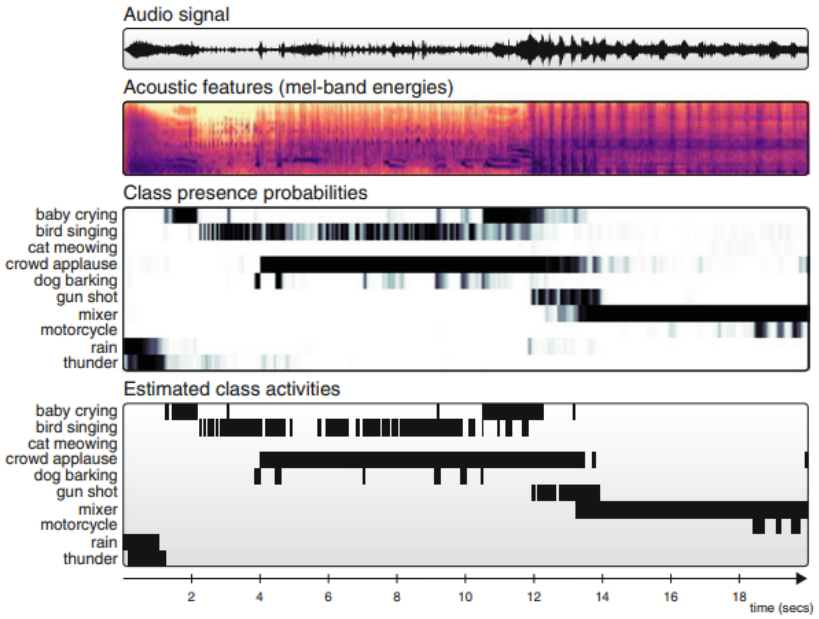


Kuva 2.3: Annotoinnin tarkkuudet [7]

Audioluokitukseen koneoppimismenetelmillä on käytetty kolmea erilaista järjestelmää: *äänimaiseman tunnistus*, *äänitunnistus* ja *äänen lokalisointi*. Järjestelmät on esitetty kuvassa 2.4. Äänimaiseman tunnistuksessa isompi ääninäyte kategorioidaan isompaan luokkaan, kuten "koti", "bussi", tai "kaupunki". Äänimaisema on sekoitettu joukko lyhyempiä ja pidempiä ääniä, jotka tapahtuvat eriaikaisesti. Äänitunnistuksessa yksittäisiä kohteita havaitaan syötteestä, kuten "koira"tai "auto". Käsitteistöstä voidaan muodostaa *taksonomia*, joka kuvaa käsitteiden välisiä suhteita (vrt. koira, eläin). Lokalisoinnissa äänilähde tunnistetaan sekä havaitaan ja paikannetaan ajassa ja taajuustasossa. Kuuluvuus luokkaan esitetään todennäköisyytenä. Binäärinen kuuluvuuus määritellään kynnysarvolla.



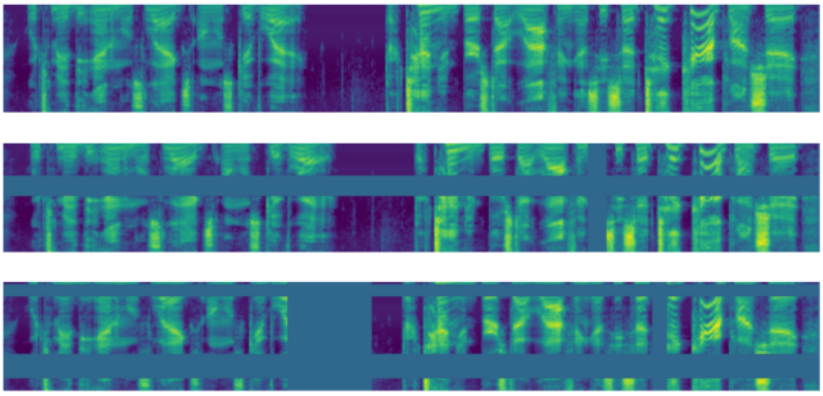
Kuva 2.4: Äänimaisema, äänitunnitus ja äänen lokalisointi



Kuva 2.5: Luokitustuloksia eri aliluokille

2.2.3 Datan augmentointi

Mallin koulutukseen käytettävän datajoukon määrää voidaan kasvattaa datan augmentoinnilla. Augmentoinnin tarkoitus on tuoda datajoukkoon sellaisia muutoksia, joilla parannetaan luokittimen kykyä oppia yleisiä piirteitä. Vektorisyötteille ja matriisisyötteille sovelletaan eri augmentointitekniikoita. Audiodatan augmentoinnissa *spektrogrammin augmentointi* [8] ja *sekoitus*[9], [10] -tekniikat ovat parantaneet yleisesti erilaisten luokittimien suorituskykyä. Spektrogrammin augmentointitekniikoita ovat: *siirtäminen*, *supistaminen* ja *venyttäminen* sekä *kohinan lisääminen*. Siirtämisessä spektrogrammia liikutetaan aikatasossa. Supistaminen ja venyttäminen vääristää tapahtuu aikatasossa. Yksiulotteisessa signaalissa käytetään *sekoitus*-tekniikkaa, *aikavenytystä* sekä kohinan lisäystä.



Kuva 2.6: Spektrogrammin augmentointitekniikoita [8]

Kohinaksi voidaan valita mielivaltaista häiriötä: väritettyä kohinaa tai yksittäisiä piikkejä. Spektrokrammeihin voidaan soveltaa *katveiden lisäämistä*. Erikokoisia katveita voidaan lisätä joko taajuustasossa tai aikatasossa. Sopivan augmentointitekniikan valinta riippuu vahvasti sovelluskohteesta. Äänimaiseman luokittelussa *sävelkorkeuden muuttaminen* osoittautui yksittäiseksi parhaaksi tekniikaksi [11]. Saman tekniikan soveltaminen alusten konemelun luokittelemiseen on osoittaunut taas vähentävän luokitustarkkuutta [12]. Samassa lähteessä on esitetty alusten konemelun

agumentointiin sovellettu tekniikka *IPS*, jossa aikatason signaalin näytteenottotaajuutta varioidaan. Tekniikalla luokitustuloksen F1-arvo parani kaikkien alustyyppien tunnistuksessa.

3 Taustateoria

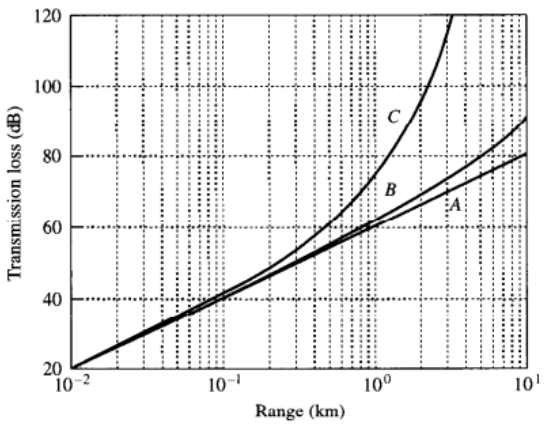
Tässä kappaleessa käsitellään teoreettinen pohjustus ja käsitteistöä, joita sovelletaan myöhemmin tapaustutkimuksessa.

3.1 Vedenalainen akustiikka

Vedenalaisella akustiikalla viitataan matalien vesistöjen akustiikkaan, mikä eroaa olennaisesti valtameriympäristön akustisista piirteistä. Valtameressä rajapintavuorovaikutukset ovat mitättömiä, ja ääniaaltojen etenemisen mallinnukseen sovelletaan eri menetelmillä kuin matalissa vesissä [13].

Ääniaallot etenevät isotermisessä vesikerroksessa vakionopeudella. Tällaiset olosuhteet vallitsevat lähinnä kokeellisissa asetelmissa. Todellisuudessa lämpötilassa esiintyy eroja, jolloin lämpötilan muutosta kuvataan lämpötilagradientilla. Äänennopeus vedessä vaihtelee hydrostaattisen paineen, lämpötilan, ja suolapitoisuuden mukaan. Valtamerissä, joissa vesipatsaan korkeus nousee ylitse kilometrin, paineen vaikutus on suurempi kuin murtovesissä, joissa vesipatsaan korkeus on alle sata metriä. Itämeren alueella äänen nopeuteen vaikuttavat pintakerrosten lämpömuutokset eri vuodenaikoina sekä suolaisen veden virtaukset Pohjamereltä [14]. Yhtälailla jokivalunnat, sadanta, haihdunta sekä jäiden sulaminen vaikuttavat meriveden suolapitoisuuteen. Lämpötilan T ja suolapitoisuuden S erot synnyttävät rajapintoja, joita

sanotaan harppauskerroksiksi. Läpötilan harppauskerros on *termokliini* ja suolapitoisuuden *halokliini*. Harppauskerrokset estävät eri syvyyksillä olevien vesikerrosten sekoittumisen. Ääniaallot taipuvat näissä rajapinnoissa kohti negatiivista gradienttia. Valtamerissä esiintyy niin kutsuttu syvä akustinen kanava [15]. Tässä kanavassa voimakkaat äänipulssi on havaittu yli 3000km etäisyydeltä. Samankaltainen kanavoitumisilmiö voi hetkellisesti syntyä matalassa vedessä, kun läpötilaerot ja suolapitoisuus aiheuttavat lokaalin minimin äänen nopeuden gradientissa. Ilmiöstä seuraa myös, että kun ääniaallot taittuvat kohti akustista kanavaa, sen ulkopuolelle syntyy katvealue, jossa ääntä ei havaita. Aluetta kutsutaan *akustiseksi varjoksi*. Näillä alueilla vastaanotin ei havaitse varjon ulkopuolisia ääniaaltoja. Vedenpinnan suuntaista gradienttia sanotaan horisontaaliseksi gradientiksi ja tätä kohtisuoraan olevaa gradienttia vertikaaliseksi. Horisontaalisen gradientin vaikutus äänen taittumiseen on käytännössä mitätön.



Kuva 3.1: Äänen vaimeneminen vedessä, A: 1kHz, B: 10kHz, C: 50kHz [15]

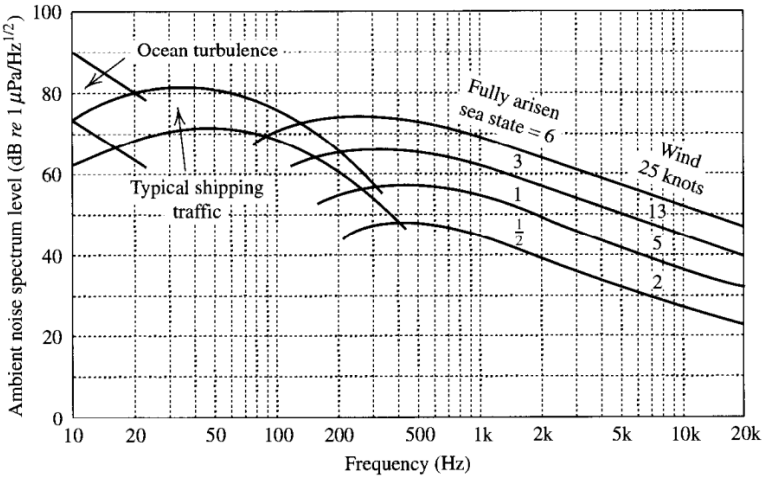
Rajapintavuorovaikutuksista johtuen pinta-alusten ja vedenalaisten kohteiden erottaminen toisistaan on osoittautunut haastavaksi [16]. Matalissa vesissä (<25m) rajapintaheijastumiset ovat yleisiä. Paineaaltorintamien propagaatiota on mallinnettu jakamalla väliaine lohkoihin (*engl. ray tracing*) ja aallon normaalimoodeilla [17]. Korkeat taajuudet vaimenevat merivedessä voimakkaammin kuin matalat taa-

juudet (kuva 3.1). Tästä johtuen matalissa vesistöissä matalat taajuudet (<50Hz) ilmenevät pitkäkestoisina tasaisina aaltoina, jotka etenevät vedenpinnan suuntaisesti. Vesikerroksissa 5°C muutos veden lämpötilakerroksissa aiheuttaa $16ms^{-1}$ eron äänennopeudessa. Ääniaaltojen etenevät aina kohti lämpötilan laskua, eli negatiivista lämpötilagradienttia kohti. Horisontaalinen gradientti ei vaikuta käytännössä juurikaan äänen etenemiseen. Matalissa vesistöissä vesikerrokset voivat sekoittua virtausten vaikutuksesta. Tällöin ainoastaan paine vaikuttaa äänennopeuteen, jolloin äänennopeus on hitaimmillaan lähellä vedenpintaa. Toisin sanoen ääniaallot taittuvat kohti pintaa, ja heijastuvat pinnasta takaisin pohjan suuntaan. Ääniaallot etenevät tällöin pinnan läheisyydessä, ja syvemmälle muodostuu akustinen varjo. Tilanne kumoutuu auringonsäteilyn vaikutuksesta, ja pinnalle muodostuu negatiivinen gradientti. Ilmiötä kutstuaan *iltapäivävaikutteeksi*. [15]

Useat rajapintaheijasteet etenevät eriaikaisesti useiden propagaatioteiden kautta lähettimen ja vastaanottimen välillä. Tämä synnyttää kaikuja sekä voimistavaa ja vaimentavaa interferenssiä. Edetessään ääniaallot heijastuvat vedenpinnasta ja merenpohjasta. Vastaanottimen kokemat pohjaheijastumat ovat heikompia lähellä veden pintaa. Mikäli äänilähde liikkuu suhteessa vastaanottimeen, syntyy rajapintavuorovaikutuksista ajassa muuttuva interferenessikuvio. Ääni kulkee vesi-ilma -rajapinnan lävitse toispuoleisesti tarkoittaen, että vedestä ei juurikaan kulje ääntä rajapinnan yli, mutta ilmasta tuleva ääni kantautuu rajapinnan ylitse.

Koska ääniaallot kulkeutuvat vedessä kauas ja rajapintavuorokaikutukset ovat voimakkaita, esiintyy veden alla useista kohinalaatuja. Akustista kohinaa merivedessä aiheuttavat merenkäynti, merenkulun liikenne, sekä sadanne ja tuuli. Nämä kohinanlähteet esiintyvät eri taajuuskaistolla. Eri kohinojen kaista-alueet on kuvattu kaavassa 3.2. Merenkulun kohina ilmenee matalilla taajuuksilla. Suuntaavalla vastaanottimella huomataan, että merenkulun kantautuu horisontaalisesti vesipatsaassa ja rajapintaheijastumiset vertikaalisesti [15]. Äänennopeuden vaihtelun on

havaittu vaikuttavan havaitun kohinan taajuusspektriin [13]. Matalissa vesissä kohina voidaan havaita erilaisena riippuen, miten harppauskerrokset ovat milloinkin sijoittuneet.



Kuva 3.2: Yleiset kohinanlähteet [15]

Kohteiden akustiseen havainnointiin käytetään kaikuluotausta. Kaikuluotaukseen käytetään aktiivista ja passiivista havainnointia. Aktiivisessa kaikukuotauksessa kohteesta saadaan tietoa lähettämällä luotauspulssi. Hydrofoni voidaan rinnastaa passiiviseksi kaikuluotaimeksi. Passiiviseen kaikuluotaukseen vaikuttavat parametrit on kuvattu kaavassa 3.1

$$SL - TL \geqslant NL - DI + DT_N \tag{3.1}$$

, jossa SL on äänenvoimakkuus kohteen luona, TL on siirtohäviö, NL ympäristön kohinan taso, DI suuntauskerroin ja DT_N havainnon kynnysarvo.

3.2 Psykoakustiikka

Ihmisen kuuloaistin mallintaminen on suuri motivaattori konekuulemisen tekniikoissa. Ihmiskorvan kyetessä erottelemaan toisistaan eri äänenlähteitä on selvää, miksi

kuuloelimen tutkiminen on keskeistä konekuulemisessa. Tästä esimerkkinä on *cocktailkutsuilmiö*, jossa kuuntelija erottaa yksittäisen puhujan muiden puhujien ja taustakohinan seasta. Koulutettu ihmisoperaattori kykenee tunnistamaan eri äänenlähteitä toisistaa, ja kouluttamaton kykenee havaitsemaan eroja konemelussa. Mitä paremmin olemassa olevan biologisen järjestelmän toiminta ymmärretään ja mallintamiseen, sitä paremmin niitä voidaan soveltaa tekoälymalleissa. Kuuloelimen komponentit ovat korvakäytävä, tärykalvo, alasin, vasara sekä sisäkorvan simpukka. Simpukan sisällä olevat simpukkatiehyt suodattavat taajuuskaistoja ääniaalloista. Simpukan on todettu regoivan äänenpaineenvaihtelua epälineaarisesti [18]. Äänenvoimakkuuden kaksinkertaistuminen vastaa äänenpaineen kymmenkertaistumista. Lisäksi taajuuskorkeuksien välinen etäisyys havaitaan epälineaarisesti. matalien taajuuden etäisyydet toisistaan havaitaan suurempana kuin korkeiden taajuuksien. Perinteinen spekstrogrammi on puolestaan resoluutioltaan tasainen, minkä vuoksi se ei ole toiminnaltaan kuuloelimen kaltainen [19].

Äänilähteiden luokitukseen on sovellettuu useita tekniikoita. Ongelmaa ollaan lähestytty joko järjestelmäkeskeisesti tai datakeskeisesti. Kuuloaisti on esimerkki olemassa olevasta järjestelmästä, joka tiedetysti suoriutuu äänenlähteiden erittelystä onnistuneesti. Korvasimpukan toimintaa on mallinnettu gammatoonisuotimilla, kepstrimuunnoksella esitetään harmonisia rakenteita, erityisesti puheentunnistuksessa. Luokituksen yleistymistä on paranneltu koneoppimismenetelmillä ja sopivilla piirreavaruuksilla, joilla kompleksit audiosignaalit esitetään harvennetussa muodossa. Datakeskeisessä ratkaisussa keskitytään sopivan kuvauksen löytämiseen. Toisesta näkökulmasta menetelmät perustetaan kuuloaistiin. Puhutaan *auditoorisesta järjestelmästä*. Gammatoonisuodin mallintaa auditoorisen järjestelmän osan sisäkorvan toimintaa. Kuten [20] osoitti, erikoituneet gammatoonisuotimet suoriutuvat tehtävässä paremmin kuin satunnaisilla parameterilla alustettu malli.

Kuuloaisin ensimmäiset olemassa olevat mallit keskittyvät kuuloelimen rakenteiden selittämiseen, eikä niinkään luokitustehtävään. Kun piirteet voidaan oppia osana mallin opetusta, olemassa olevia malleja voidaan kehittää erikoistuneeseen tunnistukseen. Sainath et al. käyttivät Mel-suodinpankkia erikoistuneiden suodinten pohajana [21]. Samalla datajoukolla koulutettu sanantunnistusmalli saavutti 5% paremman luokitustuloksen. Gammatoonipankin erikostamisesta Sheng et al. tutkivat kolmea eri GKS-kerroksen muotoa. Ensimmäisessa käytettiin kiinteää gammatoonisuodinpankkia, toisessa satunnaisesti alustettua opetettavaa yksiulotteista konvoluutiokerrosta ja kolmantena opetettavaa gammatooninsuotimilla alustettua kerrosta. Näistä viimeinen saavutti parhaan luokitustuloksen. Esitellyn mallin suotimet olivat erikoistuneet tunnistustehtävään. [20] Uusi neuroverkkoarkkitehtuuri tulee kouluttaa alusta, eikä olemassa olevaa mallia voida hyödyntää siirto-oppimisen avulla.

3.2.1 Suodinpankit

Gammatoonisuotimeksi kutsutaan lineaarista suodinta, joka saadaan gamma-jakauman ja sinikomponentin summasta. Suodinta on käytetty mallintamaan tapaa, jolla sisäkorvan simpukka reagoi äänen värähtelyihin. Yksittäinen gammatoonin impulssivaste saadaan kaavasta:

$$g(t) = at^{n-1}e^{-2\pi bt}\cos(2\pi ft + \psi) \tag{3.2}$$

, jossa $g(t)$ on impulssivaste ajanhetkellä t, a on suotimen amplitudi, b on suotimen aste, f suotimen keskitaajuus ja ψ suotimen vaihe.

ERB-asteikko mallintaa ihmiskorvan tarkkuutta erottaa yksittäinen taajuuskomponentti peiteäänen seasta. Tätä tunnistuskykä on tutkittu empiiriseillä mittauksil-

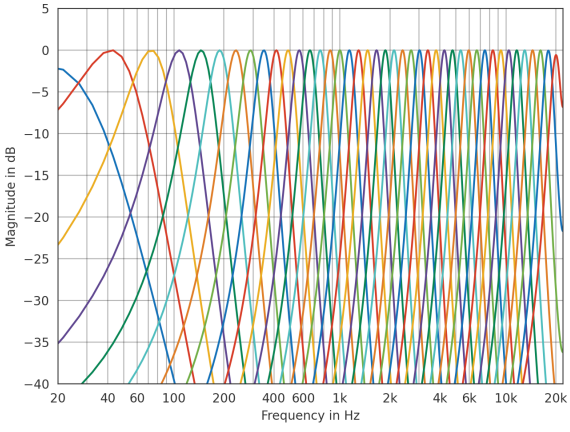
la, jossa soiva äänes erotetaan kohinasta. [18]. Tuloksena on saatu kriittiset kaistat, jotka ilmaistaan ERB-asteikolla kaavassa 3.3. Gammasuodinpankin keskitaajuudet ovat aseteltu ERB-asteikon mukaisesti. Kirjallisuudessa suodpinpankkia on käytetty syväoppimismallin piirrekerroksissa. Gammatoonipankkiin perustuvan konvoluutioneuroverkon tason, gammatoonikonvoluutiosuodin (myöh. GKS). Mallin koulutuksessa suotimia säädetään takaisin-ajolla. Tuloksena on saatu erikoistunut suodinpankki.

$$ERB(f) = 6.23 \cdot f^2 + 93.39 \cdot f + 28.52 \tag{3.3}$$

, josta on myös lineaarinen approksimaatio

$$ERB(f) = 24.7 \times (4.27 \times f + 1) \tag{3.4}$$

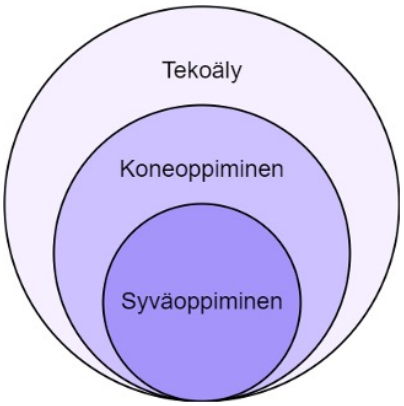
Kaavoilla 3.4 ja 3.2 saadaan suodinpankki $B_{m \times n}$, jossa m on suodinten määrä ja n yhden suotimen pituus. Kuvassa 3.3 nähdään usean gammatoonisuotimen taajuusvasteet jaettu ERB-asteikolle. Suodinten kaistanleveydet kasvavat korkeilla taajuuksilla. Tämä perustuu kuuloaistin kykyyn erotella puhtaita ääneksiä kohinan seasta. Suodatus aikatasossa esitetään kovoluutiolla. Tapaustutkimuksessa sovelletaan aikatason esitystä syväoppimismallin rakenteeseen.



Kuva 3.3: Gammasuodinpankki jaettuna ERB-asteikolle

3.3 Tekoäly ja luokittelu

Tekoälystä puhuttaessa tarkoitetaan tässä ihmisen päättelykykyä jäljentäviä toimintoja. Luokittelu kuuluu tekoälyn ja koneoppimisen piiriin (Kuva 3.4). Luokittelussa tekoälymalli kuvaa syötteen yhteen ennalta määrätyistä joukoista. *Luokitin* on algoritmi, joka opetetaan tunnistamaan syötteestä haluttuja yleisiä ominaisuuksia. Yksinkertaisin malli on binäärinen luokittelija, joka luokittelee syötteet yhteen kahtesta määritellystä alaluokasta. Edistyneemmät syväoppimismallit oppivat epälineaarisia piirteitä datajoukosta, ja luokittelevat kohteet niiden perusteella.

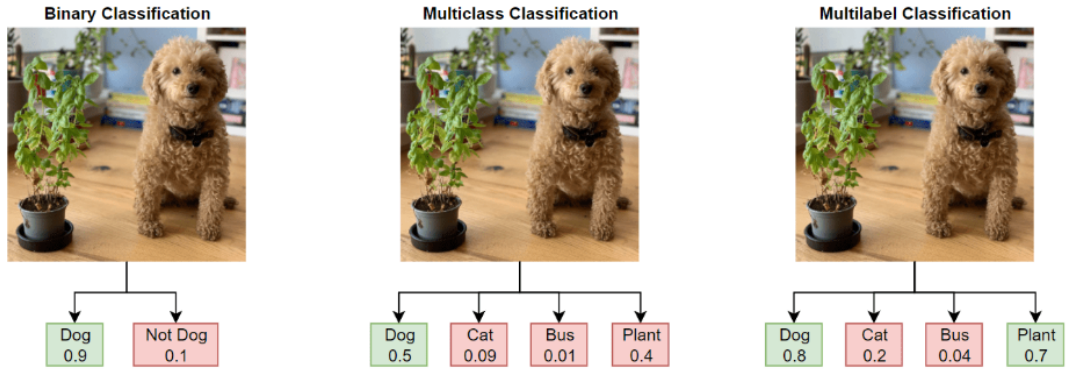


Kuva 3.4: Tekoälyn alaluokat

3.3.1 Luokittimien teoriaa

Luokittelijamallin toiminta koostuu kolmesta osasta: datan prosessointi, piirteiden muodostus ja luokitus piirteiden mukaan. Luokitin esittää kuulumisen luokkaan c_i todennäköisyytenä $P(c_i|x_1, ..., x_n)$, jossa x_n on havaintojen joukko. Yksittäistä havaintoa kutsutaan *piirteeksi*. Luokitustavat ilmaistaan eri luokitusjärjestelmillä. Yksikertaisimmillaan kohteiden joukos K yksi kohde x_i luokitellaan kuuluvaksi yhteen luokkaan $C_i \in \{c_1, c_2\}$, kuten "on koira"/"ei ole koira". Puhutaan *binäärisestä luokittimesta*. Luokituksen sanotaan olevan tällöin suljettu. Luokituspäätös $H, Z \rightarrow c_i$

tehdään jonkin havaitun kuvion Z perusteella. Kuuluminen luokkaan ilmaistaan todennäköisyydellä p_i, ja luokituspäätös tehdään asettamalla kynnysarvo k, jolloin $H(x_i \longrightarrow c_1)$, mikäli $p_i > k$ tai $H(x_i \longrightarrow c_2)$, mikäli $p_i < k$. Luokituksen sanotaan olevan *poissulkevaa*, sillä kahteen luokkaan kuuluminen samanaikaisesti ei ole mahdollista. Kun luokkia on useita, puhutaan *luokkaluokittimesta*. Luokkaluokitin on luonteeltaan myös poissulkeva. Kun useaan luokkaan kuuluminen on mahdollista, puhutaan *ominaisuusluokittimesta*. Edellä mainitut luokitusjärjestelmät on esitetty kuvassa 3.5.



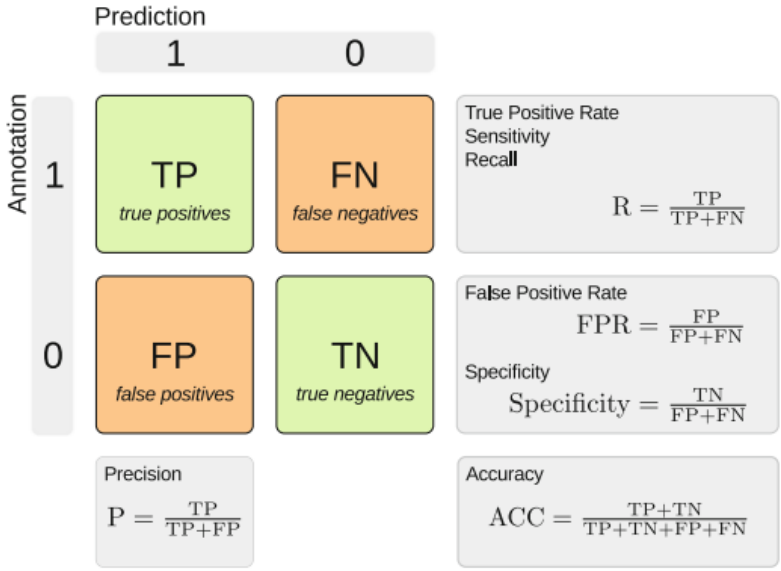
Kuva 3.5: Luokittimet. Vasemmalta: binäärinen luokitin, luokkaluokitin, ominaisuusluokitin [22]

Kynnysarvo k määrittelee ominaisuusluokittimessa kohteen kuulumisen yhteen luokkaan. Huomataan, että $\Sigma p_i = 1$. Useampi luokitin voidaan yhdistää luokitinjoukoksi $LK \in \{L_1, L_2, ..., L_n - 1, L_n\}$. Luokitinjoukon luokituspäätöksen määrittelee yhteinen sääntö. Näitä ovat esimerkiksi mediaanisääntö ja enemmistöäänestyssääntö [23].

Luokittimen arvointimetriikoita

Mallin evaluoinnissa käytetyt metriikat valitaan tapauskohtaisesti. Kuvauksia lähtöjoukosta maalijoukkoon on ääretön määrä, jolloin haasteena on valita sellainen funktio, joka on näistä kuvauksista paras. Paras kuvaus etsitään *häviöfunktiolla*.

Luokittimien suorituskykyä arviodaan käyttäen arviointimetriikoita [24]. Käytetystä mallista riippumatta luokituksen tavoitteena on separoida kohdeluokat siten, että luokitus onnistuu tarkasti. Tarkkuus tarkoittaa todellisten positiivisten ja todellisten negatiivisten määrää suhteessa kaikkiin tehtyihin luokituksiin. Jotta luokituksen laatua voidaan kuvailla tarkemmin, käytetään *täsmällisyyttä* ja *varmuus*. Useamman luokan luokittelussa sovelletaan kehittyneempia menetelmiä. [24]. *Sekaannusmatriisilla* ilmaistaan kaikki tehdyt oikeat ja väärät luokitukset. Binäärisessä luokittelussa *ROC*- ja *AUC*-metriikalla ilmenee, miten kynnysarvon säätäminen vaikutta mallin suorituskykyyn. Yleiset metriikat on esitetty kuvassa 3.6.



Kuva 3.6: Yleisiä luokituksen metriikoita [7]

3.3.2 Koneoppiminen

Koneoppimisella tarkoitetaan algoritmien ja tilastollisten mallien yhdistämistä ongelmanratkaisussa. Oletetaan, että on olemassa kuvaus lähtöjoukosta maalijoukkoon $x \rightarrow y$, jota voidaan approksimoida tarpeeksi suurella tarkkuudella. Oletukseen perustuen tavoitteena on löytää yleistyvä kuvaus datajoukon ja kuvattavan luokan

välille. Toisin kuin algoritmin kehityksessä tällöin ohjelmoija ei eksplitiisesti määrittele luokituksen käskysarjaa.

Koneoppimismenetemät jakautuvat ohjattuihin ja ohjaamattomiin menetelmiin. Ohjatuissa menetelmissä malli opetetaan annotoidulla datalla, joka on usein ihmisen annotoima. Annotoidussa datassa kuvioon sidotaan *ominaisuus*. Kohteesta tehdään havaintoja x_i, joista tunnistetaan kuvio Z, ja piirteet syötetään koneoppimismalleille. Kuvaus voi olla kategorinen tai numeerinen. Ohjaamattomassa opetuksessa ei käytetä ominaisuuksia. Ohjaamatonta opetusta sovelletaan enemmän klusterianalyysissä.

Toisin kuin syväoppimismallit useat luokitukseen käytetyt koneoppimismallit ovat luonteeltaan avoimia. Toisin sanoen niiden oppimasta rakenteesta voidaan päätellä luokituksen logiikka (vrt. päätöspuut). Joidenkin mallien rakenne vaatii, että opetusdata säilytetään muistissa. Joissain luokittelijoissa on välttämätöntä säilyttää muistiavaruudessa ne datajoukot, joilla malli opetettiin. Tällainen luokittelija on mm. KNN-luokitin, jossa lähimmät datapisteet määrittelevät uuden datapisteen luokan.

Audionluokittlusta käytetään ajoittain termiä konekuuleminen. Konekuulemisessa on neljä osaa: kuunteleva laite, audiokuvan muodostus, piirteiden luonti sekä luokittelija. Ensimmäiset kaksi vaihetta jäljittelevät kuuloaistin tehtävää: kuvastaa, miltä eri äänet kuulostavat. Nämä vaiheet muuntava kuulemista koskevan tehtävän enemmän näkemistä koskevaksi tehtäväksi. Tällöin voidaan hyödyntää konenäkemisen menetelmiä. Audiokuvan muodostaminen tällä tavalla kiinnittää piirteet, jolloin uusien piirteiden oppiminen ei ole mahdollista. Yksiulotteisen datan luokitukseen on tästä syystä kehitetty spektrimuunnoksia, jotka perustuvat yksiulotteisen datan käsittelyyn suotimilla.

3.3.3 Syväoppiminen

Syväoppimisesta puhuttaessa tarkoitetaan sellaisia neuroverkkoja, joissa sisääntulojen ja ulostulojen välissä on useita piilotettuja kerroksia. Perinteisissä koneoppimismetelmissä piirteiden irroitus ja luokitin suunnitelmaan erillään toisistaan. Toisin sanoen piirteiden valitseminen on suunnittelijan vastuulla. Syväoppimismenetelmissä neuroverkomalli oppii tunnistamaan luokituksen kannalta olennaisia piirteitä itsenäisesti. Toisaalta syväoppiminen tutustuttaa uusia haasteita mallin koulutukseen, kuten gradientin häiveneminen ja ylisovittaminen, kun dataa on rajatusti saatavilla. Koneoppimismenetelmät soveltuvat yleisesti paremmin pienempiin, ja syväoppimismenetelmät suurempiin opetusdatajoukkouihin.

Mallin koulutus ja siirto-oppiminen

Syväoppimismallia koulutetaan takaisinajo-algoritmilla [25]. Opetusprosessissa virhe toivotun ulostulon ja mallin ulostulon välillä minimoidaan iteratiisesti mahdollisimman pieneksi. Kyseessä on siis pohjimmillaan optimointiongelma. Optimointia varten tarvitaan häviöfunktio, joka määrittelee mallin ulostulossa esiintyvän virheen. Takaisinajo-algoritmilla ulostulon virheestä lasketaan mallin parametrien muutoksien vaikutukset virheeseen. Virheen derivaatat ratkaistaan ketjusäännöllä, jolloin saadaan vihreiden graidentti, ja parametreja päivitetään virhettä pienentämään suuntaan. Parametrien siirron suuruus määritetään oppimiskertoimella. Gradientin rakaisuun on kehitetty useita eri algoritmeja. Yleisiä mallin optomointialgoritmeja ovat *SGD* (*engl. stochastic gradient descent*) sekä *Adam engl. adaptive moment estimation.* Optimoijat ratkaisevat laskevan gradientin

$$w^{(t+1)} = w^{(t)} - \eta \nabla J(w^{(t)}) \tag{3.5}$$

, jossa η on oppimikerroin, $\omega^{(t)}$ mallin koulutettavat parametrit iteraatiolla t ja

$\nabla J(w^{(t)})$ häviöfunktion gradientti.

Malli koostuu isosta parametrijoukosta. Kun koko mallia koulutetaan yhtäaikaisesti, kaikkia mallin parametreja säädetään samalla takaisinajosyklillä. Osa parametrijoukosta voidaan jäädyttää, jolloin niitä ei säädetä opetussyklissä. Koko mallin parametrijoukko voidaan jakaa osajoukkouhin, joihin sovelletaan eri oppimiskerrointa. Aiemmin koulutettu malli voidaan erikoistaa uuteen tehtävään käyttämällä *siirto-oppimimista*. Uuden syväneuroverkkomallin opetus vaatii runsaasti laskentetehoa ja suuret määrät opetusdataa. Opetus tapahtuu yleisimmin pilviympäristössä palvelimilla, joissa laskentaresurssit ovat suuret. Mallin ylimmät kerrokset korvataan uusilla kerroksilla ja koulutetaan uuteen tehtävään. Alemmat kerrokset voidaan joko jäädyttää tai niitä voidaan hienosäätää pienellä oppimiskertoimella. Siirto-oppimisen rinnastetaan ihmisen kykyyn soveltaa aiempaa tietoa uuden taidon opettelemisessa.

Voi olla, että kouluttaessa mallia validointijoukon luokittelutarkkuus paranee, mutta testijoukon luokitus on huomattavasti alhaisempi. Tällöin tapahtuu mallin *ylisovittamista*. Malli oppii tällöin datajoukon yksilöllisiä piirteitä, eikä yleisiä piirteitä. Mallin ylisovittamista pyritään välttämään eri tekniikoilla, kuten regulariosinnilla, harvenuksella tai opetuserien normeerauksella. Joillain opetusparametreilla mallin validointijoukon luokitustarkkuus ei parane koulutuksen aikana, jolloin puhutaan *alisovittamisesta*. Sopiva opetusparametrijoukko, jolla vältetään edellä mainitut haasteet, löytyy usein vain kokeilujen kautta. *Hyperparametrit* ovat ennen opetusprosessia valittuja kiinnitettyjä parametreja. Hypermarametreja ei voi muuttaa mallin koulutuksen aikana. Ne voivat kuitenkin vaikuttaa mallin suorituskykyyn. Sopivimmat hyperparametrit valitaan eri hakumenetelmillä. Mallien opetus tapahtuu häviöfunktion numeerisella optimoinnilla. Yleisesti käytetty menetelmä on gradienttilasku pienissä opetuserissä, ja parametrit päivitetään käytteän *käänteisajo*-algoritmia.

Mallin koulutus kuvataan seuraavaksi. Opetusdata jaetaan kolmeen joukkoon: opetusjoukko, testijoukko ja validointijoukko. Opetusjoukolla päivitetään mallin pa-

rametereja. Validointijoukkoa käytetään todentamaan, että mallin tarkkuus paranee. Testijoukolla todetaan mallin tarkkuus, kun opetus pysäytetään. Opetusdata syötetään mallille pienerissä. Joka pienerän jälkeen parametreja päivitetään käänteisajolla. Kokonaishäviö lasketaan viimeisellä kerroksella, ja virheen gradientti saadaan osittaisderivaattoilla joka neuronille. *Opetusnopeus* määrittelee, kuinka suuresti neuroneissa muutetaan parametreja suhteeessa virheeseen. *Epokiksi* kutsutaan sitä, kun koko opetusdata on kerran käytetty opetukseen. Koska opetusdataa on usein paljon, eikä kaikkea saa tallennettua opetusympäristön muistiin, data jaetaan eriin. Yhden erän kokonaisvirhe lasketaan kumulatiivisesti, ja käänteisajo suoritetaan, kun kaikki erän sisältö on läpiajettu. Sopiva epokkien määrä ja *eräkoko* valitaan usein kokeilemalla.

Moniluokkaluokituksessa häviöfunktiona käytetään *kategorista ristientropiaa*:

$$\mathrm{CCE} = -\sum_{i=1}^{N}\sum_{j=1}^{C} y_{ij}\log(\hat{y}_{ij}) \tag{3.6}$$

, jossa N on syötteiden määrä, C luokkien määrä, ja y todellinen luokka ja $\hat{y}$ ennustettu luokka. Sopivilla opetusparametreilla pyritään välttäämään mallin yli- ja alisovittamista, jolloin malli epäonnistuu yleisten piirteiden oppimisessa. Ylisovittamisessa mallin voidaan ajatella oppivan opetusdatan liian yksityiskohtaisesti. Uutta dataa vertaillaan tällöin liian tarkasti opittuihin ominaisuuksiin. Konvoluutiotasojen kanavien koko redusoidaan sekoitustasoilla. Viimeiset kerrokset ovat tiheästi yhdistettyjä kerroksia.

Syvämallit tarvitsevat koneoppimismalleihin verrattuna isomman määrän dataa ja opetusmenetelmä vaatii paljon laskentaresursseja. Mallien opetukseen käytetään optimoitua laitteistoa, joka kykenee rinnakkaisprosessointiin. Audion luokittelussa on käytetty YAMNet- ja VGGish-malleja [26], jotka käyttävät syötteenä kaksiulotteista signaalia. Tehtävään on kehitetty myös erikoistuneita neuroverkkoarkkihtehtuureja.

3.3.4 Neuroverkot

Neuroni

Neuroverkon pienin yksikkö, neuroni, on saanut inspiraation hermoverkoista, joissa hermosolu aktivoituu sisääntulosignaalien vaikutuksesta. Merkitään x_n sisääntulovektorina ja y_j neuronin ulostulona. Yksittäisen neuronin ulostulo saadaan kaavasta:

$$y_j = \sum_{k=1}^{n} x_n \cdot w_n + b_n \tag{3.7}$$

, jossa y_j on kerroksen neuronin j ulostulo, n on sisääntulojen määrä, w_i on kerroin yhdelle syötteelle ja b_j on neuronin siirtovakio. *Tiheästi kytketyssä* kerroksessa neuronit ovat kaikki yhteydessä aiemman tason kaikkiin neuroneihin. Merkitään kerroksen indeksi j:llä ja neuronin indeksi i:llä. Tällöin useakerroksinen neuroniverkko ilmaistaan:

$$y_{j,i} = \sum_{k=1}^{n} y_{j-1,i} \cdot w_n + b_n \tag{3.8}$$

Tällaisenaan yhtälö tulkitaan peräkkäisinä lineaarisina yhtälöjoukkoina ja neuronin ulostulo voi tällöin saada mielivaisen suuren tai pienen arvon. Jotta joukolla neuroneja voidaan approksimoida mielivaltaisia funktioita [27], kaavaan 3.8 lisätään epälineaarinen aktivaatiofuktio.

$$f(y_{j,i}) = \sigma(\sum_{k=1}^{n} y_{j-1,i} \cdot w_n + b_n) \tag{3.9}$$

Aktivaatiofunktio kuvaa ulostulon rajatulle välille (0..1). Yleisiä aktivaatiofunktioita ovat *sigmafunktio*, *tanh* ja *ReLU*.

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{3.10}$$

$$tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{3.11}$$

$$ReLU(x) = \max(0, x) \tag{3.12}$$

Näistä kolmesta $ReLU$-funktiolla eliminoidaan katoavan ja paisuvan gridientin ongelma [28].

Monikerrosperseptronimallissa syötetason sisääntulo x_i kulkee näin kaikkien tasojen läpi. Sisääntulo- ja ulostulotason välisiä kerroksia kutsutaan *piilotetuiksi kerroksiksi*. Kun ajatellaan piilotettujen kerrosten toimintaa, voidaan esittää, että kerrosten neuronit aktivoituvat aiempien kerrosten neuronien summana. Toisin sanoen neuronit oppivat komplekseja kuvaksia aiempien tasojen piirteistä. Malli, jossa on yhteensä kolme tasoa voidaan siis ilmaista $f^{(3)}(f^{(2)}(f^{(1)}(x_n)))$, jossa f^n on kerros ja n kerroksen indeksi. Useat tiehästi kytketyt kerrokset kasvattavat muisti- ja laskentavaativuutta suuresti. Jokainen neuroni kytekytyy seuraavan kerroksen neuroniin, siis kytkentöjä on n^2, jossa n on neuronien lukumäärä. Yleisesti ensimmäisessä tasossa $f^{(1)}$ käytetään vähemmän neuroneja syötteen kokon nähden.

Regularisointi

Mallin oppimiin parametreihin voidaan vaikuttaa eri regularisointitekniikolla. L1 ja L2 regularisointi kehitettiin alunperin lineaariselle mallille, ja sitä on myöhemmin sovellettu koneoppimis- ja syväoppimimalleihin. L1 ohjaa parametrien asettumista lisäämällä häviöfunktioon $L(x)$ termin:

$$L(x) + \lambda \sum_{j=1}^{p} |\beta_j| \tag{3.13}$$

, joka pakottaa osan parametreista nollaksi, mikä luo yksinkertaisemman ja helpommin tulkittavan mallin. Regulaisoinnin voimakkuuden määärä opetusparamteri

λ. L2 toimii L1 tavoin, mutta kasvattaa häviöfunktiota neliöllisesti parametriin β nähden:

$$L(x) + \lambda \sum_{j=1}^{p} \beta_j^2 \tag{3.14}$$

L1 ja L2 regularisoinnit vaikuttavat lopullisiin mallin parametreihin. Syväoppimismalleissa käytetään lisäksi *poiskytkentää* tasojen välissä. Kun mallia koulutetaan, poiskytkennällä osa neuroneista saa ulostulon arvoksi 0. Tällöin neuronin parametreja ei säädetä opetuskierroksella. Poiskyntentätaso ilmaistaan välillä $[0, 1]$, joka kuvastaa todennäköisyyttä, jolla yksittäinen neuroni kytketään pois.

3.3.5 Konvoluutiokerrokset

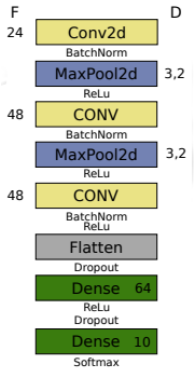
Konvoluutioneuroverkot ovat yleistyneet kuvantunnistustehtävissä, sillä tiheästi kytketyt tasot eivät käytännössä sovellu kuvantunnistukseen. Kuvantunnistukseen sovelletaan kaksiulotteista konvoluutio-operaatiota:

$$y(m, n) = \sum_{i=0}^{k_h-1} \sum_{j=0}^{k_w-1} x(m+i, n+j) \cdot w(i, j) + b \tag{3.15}$$

Yksi suodin $x(i, j)$ muodostaa neuroverkon kanavan, joka kuvastaa yhtä piirrettä. Kanavia, eli suotimia, valitaan mallin suunnitelussa haluttu määrä. Konventionaalisesti konvoluutiotasoa seuraa piirteiden normeeraustaso ja tämän jälkeen aktivaatiofunktio [29]. Mallien rakenteeseen sovelletaan usein myös *valikointitasoja* [29], jotka kutistavat kavavien dimensioita. Tämä tapahtuu juoksuttamalla ikkunaa syötteen yli, ja ottamalla ikkunan maksimiarvon tai keskiarvon. Konvoluutiokerros käyttää huomattavasti tiheitä kerroksia vähemmän parametreja. Viimeisten konvoluutiokerrosten kavavien ulostulot tasataan, ja ne syötetään tiheisiin kerroksiin. Koko mallin ulostulon aktivaatiofuktiona käytetään usein *softmax*-funktiota, joka on yleistys *sigmoid*-funktiosta usealle sisääntulolle:

$$\sigma(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{n} e^{z_j}}, \quad i = 1, 2, \ldots, n \tag{3.16}$$

Kuvassa 3.7 on kuvattu äänimaisemanluokitukseen erikoistunut malli.



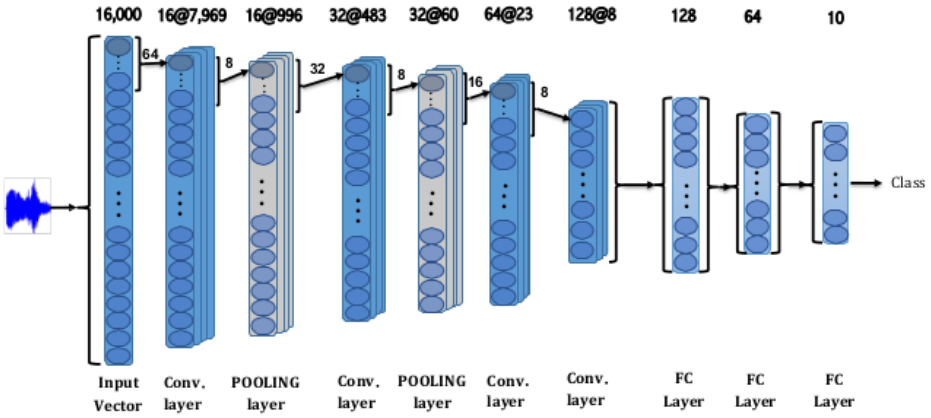
Kuva 3.7: 2D-konvoluutiomalli, SB-CNN [11]

Useita peräkkäisiä konvoluutio- ja normeeraustasoja käytetään ResNet- [30] ja MobileNet-rakenteissa [26]. Samaa operaatiota on sovellettu yksiulotteisille singaaleille. Syötevektorille konvoluutio-operaatio ilmaistaan:

$$y_i = \sum_{j=0}^{k-1} x_{i+j} \cdot w_j + b \tag{3.17}$$

Kuvassa 3.8 on kuvattu yksiulotteisen signaalin luokitinmalli (1DCNN). Malli rakennetaan samalla tavalla normeeraustasoita ja aktivaatioista. Kuvasta nähdään, miten ensimmäisten konvoluutiotasojen suotimet ovat kooltaa suurempia kuin piilotettujen tasojen suotimet. Voidaan ajatella, että ensimmäiset kerrokset havaitsevat temporaalisia piirteitä. Piilotetut kerrokset oppivat ensimmäsen tason piirteiden kombinaatioita koko syötteestä. Konvoluutioiden välissä nähdään *valikointitasoja*, joiden juoksutusarvo on 8, ja viimeisessä juoksutusarvo on 16.

Mallin alimmissa kerroksissa on tiheästi kytketyt tasot, joissa on 128, 64, ja 10 neuronia. Kuvattua mallia on sovellettu äänimaisemien tunnistuksessa. Satunnaisilla arvoilla alustetut yksiulotteiset konvoluutiokerrokset olivat tarkempia luokittelussa,



Kuva 3.8: Malli, jossa on käytetty yksiulotteista konvoluutiota [31]

kuin perinteiset piirteet [31]. Luokitustulosta saatiin parannettua 2% alustamalla syötekerroksen suotimet gammatoonisuotimilla, jolloin *1DCNN*-malli saavutti oli 89% tarkkuuden. Mallilla parametrien määrä oli 431′988.

3.3.6 Tekoälysovellukset ja reunalaitteet

GPT:n kaltaiset suuret kielimallit koostuvat miljardeista parametreista [32], ja sisältävät monimutkaisempia rakenteita verrattuna aiemmin käsiteltyihin tiheästi kytkettyihin kerroksiin ja konvoluutiokerroksiin. Mallien suuresta kokoluokasta johtuen ne ovat käytössä pilviympäristössä. Sovellukset, jotka haluavat käyttää tekoälymallia, lähettävät syötteen pilviympäristöön, ja palvelin lähettää tuloksen asiakaslaitteelle. Suunnittelemalla uusia arkkitehtuureja on voitu pienentää tekoälysovellusten parametrimäärää siten, että malleja on mahdollista käyttää paikallisesti mobiililaitteella [26]. *Reunalaskennalla* tarkoitetaan keinoälysovelluksia, jotka on toteutettu pieniin itsenäisiin järjestelmiin kuten älylaitteelle tai sulautettuun järjestelmään. Järjestelmäpiirillä on rajatusti flash-muistia (500-2000kB) ja RAM-muistia (50-250kB). Yksi kirjasto, jolla mikrokontrollerille soveltuvia malleja koulutetaan, on `tensorflow lite`. Neuroverkkomallien parametrit taltioidaan sulatuten järjestelmän flash-muistiin.

On havaittu, että mikrokontrollerilla spektrogrammin laskeminen vei suhteessa enemmän aikaa (60ms) päättelyyn verrattuna (31-81ms) [33]. Spektrogrammin laskeminen on osa datan esikäsittelyä. Sopivan mallin valintaa reunalaitteellee rajaa vähäiset käytettävissä olevat resurssit. Mallin kokonaispäättelyaika koostuu datan esikäsittelystä ja päättelystä. Matriisikertolaskuihin ja konvoluutio-operaatioon erikoistuneet laitekiihdyttimet nopeuttavat mallin toimintaa.

3.4 Piirteiden valinnasta

Yleisiä äänentunnistuspiirteitä on kuvattu taulukossa 3.1. Näitä menetelmiä käytetään kattavasti koneoppimismenetelmissä. Äänisignaalin esikäsittelylle ja piirteiden irrotukselle on olemassa monta tutkittua menetelmää. Äänisignaali jaetaan ikkunoihin, ja ikkunoille tehdään fourier-muunnos kaavalla 3.18. Signaali esitetään matriisina $S_{m \times n}$, jossa n on ikkunoiden määrä ja m taajuuskomponenttien määrä.

Menetelmän nimi	**Toiminta**
Massakeskipiste	S
Spectral rolloff	S
Spektrin kaista	S
Spektrin kontrasti	S
Spektrin tasaus	S
Polynomisovite	M
MFCC	K
GTCC	K
CQT	M
Nollanylitystaajuus	T
Mel-spektri	S

Taulukko 3.1: Olemassa olevia piirteitä ja menetelmiä. (S = taajuustaso, T = aikataso, K = kepstri, M = muu)

3.4.1 Yleiset piirteet

Prosessoimattomasta datasta on sellaisenaan vaikea tunnistaa kohteita. Tästä syystä datasta irroitetaan piirteitä, joiden oletetaan olevan parempia mittareita kohteiden tunnistamisessa. Piirteiden irrotuksen voidaan ajatella olevan eräänlainen datan häviöllinen pakkaus. Irrotetuista piirteistä voidaan vielä laskea useamman kertaluokan piirteitä. Ei ole itsestään selvää, mitkä piirteet ovat hyviä ja mitkä huonoja tunnistuksen kannalta. Siksi piirteiden eri yhdistelmiä pitää kokeilla luokituksessa. Laskentavaatimuksen vähentämiseksi on syytä jättää pois sellaiset piirteet, joiden perusteella aliluokitusta ei voida tehdä tehokkaasti, tai jotka eivät ole riippumattomia suhteessa muihin piirteisiin.

Aikadatasta irrotettavat piirteet jakautuvat aika- ja taajuusavaruuden ominaisuuksiin. Aika- ja taajuusavaruuden piirteet on esitelty taulukossa 3.1. Kuvaillaan seuraavaksi kirjallisuudessa yleisiä piirteitä:

Lyhytkestoinen Fourier-muunnos (STFT)

STFT on usean menetelmän ensimmäinen vaihe.

$$X(m,\omega) = \sum_{n=-\infty}^{\infty} x(n) \cdot w(n-m) \cdot e^{-j\omega n} \tag{3.18}$$

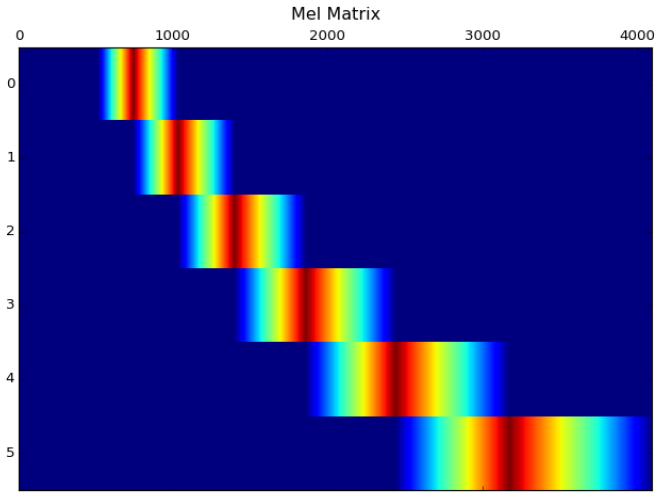
Mel-spektogrammi

Mel-asteikko mallintaa kuuloaistin sävelkoekeuksien epälineaarista etäisyyttä. Kaavan 3.19 mukaan fysikaalinen taajuus kuvataan sellaiseksi, että kuuloaisti kokee äänesten välisen etäisyyden lineaariseksi.

$$m = 2595 \log_{10}(1 + \frac{f}{700}) \tag{3.19}$$

Mel-suodinpankki koostuu kolmioinmuotoisista suotimista, jotka on luotu mel-

asteikon mukaan jollekin taajuuskaistalle. Mel-spketogrammi saadaan, kun suodinpankkia juoksutetaan audiosignaalin yli. Menetelmlä alinäytteistää taajuusavaruuden dataa poistaen raakadasta redundanttia korkeataajuuksista dataa, joka on usein kohinaa. Yksi suodinpankki on esitetty kuvassa 3.9. Ladottua mel-spketrokrammia käytetään sellaisenaan syötteenä.



Kuva 3.9: Mel-suodinpankki, jossa kuusi suodinta

Kepstri

Kepstri [34] perustuu mel-spektrin logaritmin taajuusmuunnokseen. Kepstri ilmastaan;

$$C_c = FT^{-1}[logP(\omega)] \tag{3.20}$$

$$C_r = Re\{FT^{-1}[logP(\omega)]\} \tag{3.21}$$

Kepstripiirteitä on sovellettu puheentunnistuksen tekoaälysovelluksiin. Puheentunnistuksessa on huomattu kepstrin ottaminen mel-asteikosta parantavan luokituksen tarkkuuta. Kaavalla melkepstri ilmaistaaan:

$$1. \quad X[k] = \sum_{n=0}^{N-1} x[n] e^{-j\frac{2\pi}{N}kn}$$

$$2. \quad S[m] = \sum_{k=0}^{N-1} |X[k]|^2 H_m[k]$$

$$3. \quad \log S[m] = \log\left(\sum_{k=0}^{N-1} |X[k]|^2 H_m[k]\right)$$

$$4. \quad c_n = \sum_{m=0}^{M-1} \log S[m] \cos\left[\frac{\pi n(2m+1)}{2M}\right]$$

, jossa $X[k]$ on spektri, H_m on mel-suodatettu signaali ja c_n MFCC-kertoimet. Kepstristä voidaan siis olettaa olevan hyötyä, mikäli äänenlähde tuottaa harmoonisia sarjoja.

Gammasuodinpankin kepstrikertoimet (GTCC)

GTCC:n on väitetty olevan MFCC:tä yleistyvämpi muissa kuin puheentunnistustehtävissä. Pääkomponenttien ollessa yhden kilohertsin yläpuolella tunnistustekniikat suoriutuvat yhtäläisesti [35].

$$H_m(f) = \begin{cases} 0 & \text{if } f < f_m \\ \frac{(f/f_m)^{(\alpha-1)} \cdot \exp(-\beta(f/f_m))}{\Gamma(\alpha)} & \text{if } f_m \leq f < f_{m+1} \\ 0 & \text{if } f \geq f_{m+1} \end{cases} \tag{3.22}$$

3.5 Aiemmat tutkimukset

Tässä kappaleessa tarkastellaan kirjallisuudessa käytettyjä malleja ja arviodaan tapaustutkimukseen valikoituvat mallit. Valikointikriteeteinä ovat mallin soveltuvuus

alusten luokitukseen sekä mallin koko.

Julkisten vedenalaisten mittausaineistojen määrät ovat suhteessa pieniä. Verrataan tilannetta äänimaisemaluokituksen tutkimuksen historiaan. Fonseca et al. [36] luokittelevat kattavasti vuoden 2014 jälkeen julkaistut data-aineistot. *UrbanSound8K* oli ensimmäinen julkinen tietokanta, josta on tullut standardi verailuaineisto algoritmikehityksessä. *AudioSet* on ollut opetusaineistona suurille luokitinmalleille kahdeksalla mijoonalla äänitteellään, jotka on lajiteltu 521 eri luokkaan. AudioSet on kokoelma valmiiksi prosessoituja piirteitä, eikä alkuperäistä mittausdataa ole julkisesti saatavilla [36]. Nämä tietokannat ovat mikrofonilla nauhotettuja. AudioSet-tietokanta sisältää 527 luokkaa. US8K-tietokannalla koulutettu *SB-CNN*-mallia on käytetty mikrokontrollerilla luokittelemaan äänimaisemia [33].

Luokitukseen on käytetty STFT:n pohjautuvaa logaritmista spektrogrammia. Taajuusasteikkoon on sovellettu mel-asteikkoa, ERB-asteikkoa tai gammatoonisuotimia, jotka kaikki jäljentävät auditoorista järjestelmää. Spektrogrammien viereiset kaistat sisältävät kuitenkin paljon keskenään korreloiviva kanavia, mikä lisää redundatin datan määrää. Tähän ratkaisuksi on kehitetty lisäksi kepstrimenetelmiä, mikä pienentää luokittimen syötteiden määrä merkittävästi. Kepstri saadaan spektristä diskreetillä kosiinimuunnodksella. Sovelluspiirre on johonkin nimenomaiseen käyttökohteeseen luotu piirre. Moni audioluokittelun sovelluspiirre kuvastaa augiosignaalin korkeamman tason piirteitä. Audioluokitukseen on käytetty verhokäyrää, kokonaisenergiaa, nollanylitystaajuutta ja spektripiirteitä, kuten MFCC, spektrijohdannaisia ja näiden delta sekä delta-delta -muunnoksia. GTCC-piirteet ovat suorituneet geneeristen äänten luokitukseen MFCC-piirteitä paremmin, ja GTCC:tä on sovellettu vedenalaiseen tunnistukseen[1]. Näiden käyttäminen kuitenkin suodattaa opetusdatasta piileviä piirteitä, joita on käsin tehdyssä piirteenvalinnassa vaikea nähdä, ja siten valita.

Kuvantunnistuksessa suositut konvoluutiokerrokset ovat osoittautuneet toimiviksi spektrogrammien luokituksessa[37][38]. Konvoluutiokerrokset oppivat joukon kaksiulotteisia suotimia, jotka ovat erikoistuneita eri kohteiden tunnistamiseen. Avoimesti saatavilla olevat isot konvoluutioneuroverkkomallit on koulutettu suurella määrällä dataa. Usein malli on koulutettu erottamaan kymmeniä tai satoja eri kohteita kuvasta. Kuvaluokituksesta eriytetyt audioluokitusmallit ovat kooltaan pienempiä ja suoriutuvat tehtävästä paremmin [37]. Muita erikoistuneita kerroksia on luotu eri datatyypeille. Yksiulotteista konvoluutiota on sovellettu aikasarjadatan luokitukseen. 1D-konvoluutiokerros sisältää joukon suotimia, jotka voivat olla keskenään eri pituisia. Satunnaisilla parametreilla alustettu 1D-konvoluutiokerron on suoriutunut paremmin äänenluokittelusta verrattuna kaksiulotteisiin konvoluutioihin [31]. Auditoorisen järjestelmän suotimia käyttämällä parani edelleen. CRNN-malli tehtiin lisäämällä LSTM-kerroksia konvoluutiotason ja MLP-tason väliin. Liu et al.[39] käyttivät syötteenä viiden sekunnin augmentointuja mittauksia ShipsEar-tietokannasta. Mittaukset pakattiin $(128, 216, 3)$ kokoisekeksi syötteeksi, jossa ikkunoilla oli 75% päällekkäisyys. CRNN-malli suoriutui CNN-mallia huomattavasti paremmin.

Alusten konemelun luokituksen on kehitetty erikoistunut malli SNANet [40] (2023), joka käyttää tunnistuksesa yhdistettyjä piirteitä. Malli keskittyy kolmen eri äänilähteen kuuntelemiseen: mekaanisten äänien, potkuriääniin ja kavitaation ääniin. Havaittiin, että sovellukseen erikoistunut malli saavutti 78.25% tarkkuuden CQT-piirteillä ja 74.13% tarkkuuden mel-piirteillä. SNANet on tarkastelluista malleista erikoitunein käyttötarkoitukseen. Malli jätettiin kuitenkin pois tarkastelusta kompleksin rakenteen ja suuren parametrijoukon vuoksi (>1'000'000).

Kaksiulotteisia konvoluutiokerroksia on sovellettu kuvantunnistukseen, ja ne ovat myöhemmin saaneet suosiota audiontunnistuksessa. Kuvantunnistuksessa käytettyjä

arkkitehtuureja on opetettu suurilla audiotietokannoilla. Tunnetuimpia malleja ovat MobileNet-pohjainen YAMNet ja VGG-pohjainen VGGish. Äänenluokitukseen on kehitetty räätälöityjä arkkitehtuureja: PiczakCNN[41] ja SB-CNN [11], jotka luokittelevat kohteita mellog-spekstrokrammista. eGRU[42] käyttää samanlaista arkkihtetuuria lisäen LTSM-rakenteen. EnvNet[43] ja EnvNet2[9] laskevat piirteitä suoraan mittausdatasta yksiulotteisella konvoluutiolla. Takaisinkytektyt rakenteet (eGRU) on jätetty pois niiden vaatiman parametrijoukon suuruuden vuoksi. Yksiulotteisista rakenteista 1DCNN-mallia on aiemmin sovellettu alusten luokitteluun [31]. EnvNet sekä EvnNet2 suoriutuivat 1DCNN-malliin nähden heikommin äänimaiseman tunnistuksessa.

Tapaustutkimukseen valitaan seuraavat mallit: SBCNN, 1DCNN ja YAMNet.

4 Tapaustutkimus

Tässä kappaleessa tarkastellaan mallien suoritumista alustyyppien luokituksessa. Koulutettavia malleja on yhteensä kolme: SB-CNN, 1D-CNN ja YAMNet, ja jokainen näistä koulutetaan molemmilla aineistoilla. Mallit luodaan `tensorflow` -kirjastolla, ja kaikki mallit ovat moniluokkaluokittimia. Luokkia valitaan koulutukseen tasainen edustus, joka määräytyy vähiten edustetun luokan mukaan. Taulukossa 4.1 nähdään mittausten määrä ja kesto luokittain, ja taulukossa 4.2 koulutusdatan jaot. YKsi ikkuna koulutusdatassa kestää ajassa yhden sekunnin. Ikkunoiden päällekäisyys valitaan siten, että ikkunoida tulee vähintään 1000 yhtä luokkaa kohti.

Taulukko 4.1: Aineistojen luokat ja kokonaiskestot

Aineisto	**Luokka**	**Lukumäärä**	**Kokonaiskesto (s)**
DeepShip			
	Konttialus	13	2324
	Risteilijäalus	21	1146
	Tankkeri	29	1660
	Hinaaja	4	593
ShipsEar			
	A (suurin)	17	1875
	B	19	1560
	C	30	4270
	D (pienin)	12	2455

Mallin yleistyvyyttä arvioidaan rinnakkaisaineistolla. Tähän valitaan rinnakkaisaineistosta saman alustyypin kohde. Tarkastellaan, kuinka yksimielinen mallin tuot-

Taulukko 4.2: Data-aineiston jako

Aineisto	Malli	Opetusdata	Validointi	Testijoukko	Overlap
DeepShip					
	SBCNN	4800	1600	1600	80.0%
	1DCNN	4590	2250	2250	80.0%
	YAMNet	2880	960	960	40.0%
ShipsEar					
	SBCNN	4800	1600	1600	20.0%
	1DCNN	4590	2250	2250	20.0%
	YAMNet	2880	960	960	40.0%

tama luokitustulos on yhdelle mittaukselle. Yleistymistä arviodaan edelleen luokittelemalla toisen aineiston mittauksia suppeasti.

Valikoituneet mallit

YAMNet-mallin rakenne perustuu MobileNetV1-malliin [26], joka koostuu useista erilaisista konvoluutioketjuista. Mallista on toteutettu erikokoisia arkkitehtuureja, joista yleisin on 4.2 miljoonaa parametria. Syöte muunnetaan 16kHz näytteistystajuudelle, josta lasketaan 25ms ikkunoita, joiden päällekkäisyys on 40%. Audio muunnetaan mel-spektrogrammeiksi kaistavälille 125-7500Hz. Mel-kanavia on 64. Yhden analyysi-ikkunan pituus on 0.96 sekuntia. Malli on koulutettu AudioSet-tietokannalla tunnistamaan 521 eri luokkaa. Kun malli opetetaan tunnistamaan uusia kohteita, piirrekerrokset jäädytetään ja tiheästi kytketyt kerrokset koulutetaan uudella datalla. Mallin on valikoitu mukaan, jotta erikoistuneiden mallien tuloksia voidaan suhteuttaa.

SB-CNN [11] käyttää syötteenä mel-spektrogrammia, jonka taajuuskaista on ihmisen kuuloalue ($0 - 22'050$ Hz), ja jonka syötekoko on (128×128). Yhden syötten pituus ajassa mitattuna on noin kolme sekuntia. Alkuperäiset tekijät käyttivät esikäsittelyssä ESSENTIA-kirjastoa. Tämän sijasta tässä käytetään `librosa`-kirjastoa.

Mallin toteutus `tensorflow`-kirjastolle on toteutettu [33] mukaisesti. Malli on alustettu satunnaisilla parametreilla.

1D-CNN [31] luokittelee kohteen vektorisyötteestä. Syötteen koko on $(1 \times 16'000)$. Mallin rakenne on esitetty aiemmin kuvassa 3.8. Rakenteessa konvoluutiotasojen välissä on dimensioita vähentäviä valikointitasoja. Alimissa kerroksissa on kolme tiheästi kytkettyä kerrosta. Mallin ensimmäiseen kerrokseen on upotettu kaavojen 3.2 ja 3.4 mukaan luotu suodinpankki taajuuskaistalle 20-8000Hz. Kerroksen koko W on (512×64).

Taulukko 4.3: Valikoituneet mallit

Malli	**Menetelmä**	**Parametrien määrä**	**Syötteen koko**
SB-CNN[11] [33]	mel-spektri	431'988	(128, 128)
1D-CNN[31]	raw	714'596	(16'000, 1)
YAMNet[26]	mel-spektri	4,2M	(96, 64)

4.1 Data-aineistosta

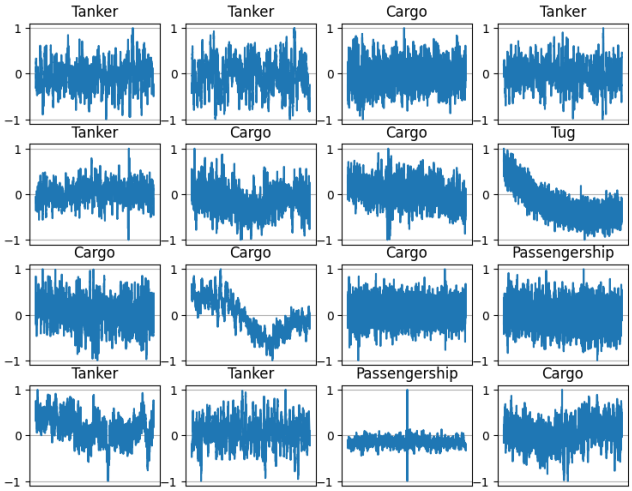
Aineistossa jokaista havaittua kohdetta on yksi mittaustiedosto. Ikkunajako perustuu ikkunoiden päällekkäisyyteen, joka ilmoitetaan prosentteina. Luokkien kokonaiskesto aineistossa on epätasainen. Tästä syystä aineistoille on valittu omat päällekkäisyydet siten, että kaikkien luokkien edustus on vähintään tuhat ikkunaa. Valitut alueet annotoidaan kuuluvaksi vain yhteen luokkaan. Näin rajatuista alajoukoista valitaan omat koulutus-, validointi- ja testidatajoukot. Kun mallit on koulutettu, ne arviodaan toisen aineiston verrattavissa olevilla luokilla.

ShipsEar -tietokanta [1] on nauhoitettu Espanjan luoteisrannikolla lähellä Vigon satamaa vuosina 2012-2013. Kanava on enimmillään 10km leveä ja 45 metriä sy-

vä. Aineistossa on dataa kalastusaluksista, lautoista, rahtialuksista, roro-aluksista, jahdeista ja pienemmistä purjeveneistä. Kokonaiskestoa aineistolla on neljä tuntia. Parhaat luokitustulokset olivat moottoriveneillä ja matkustajaaluksilla. Piirteinä käytettiin mel-spektrogrammia ja luokittimena GMM-mallia. Alukset luokiteltiin neljään luokkaan alusten koon perusteella pienistä isoihin. Meren taustakohina lisättiin viidenneksi luokaksi. Koulutukseen sovelletaan samoja luokituksia [1] mukaisesti, jossa mittaukset on kategorioitu luokkiin A-D aluksen kokoluokkien mukaan pienimmistä aluksista suurimpiin.

DeepShip -tietokanta [2] sisältää vedenalaista akustista dataa 265 eri aluksesta. Se on kerätty vuosina 2016-2018 Georgian salmella, jossa nopeusrajoitus on 1-3 solmua. Kohteiden etäisyys on noin mittauspisteestä 20-2000 metriä. Nauhoituksissa esiintyy taustakohinaa ympäröivästä teollisuustoiminnasta ja biologisista prosesseista. Tietokannan kokonaiskesto on 47 tuntia ja 4 minuuttia. Datajoukolla on saavutettu perinteisillä koneoppimismenetelmillä keskimäärin 72% luokitutarkkuus neljän eri alustypin tunnistamiseen. Alukset luokiteltiin neljään luokkaan: hinaajat, matkustaja-alukset, rahtilaivat ja säiliöalukset. Tässä tutkimuksessa sovelletaan aineiston suppeaa versiota, joka on kestoltaan kaksi tuntia. Koulutukseen sovelletaan aineiston alustyyppejä: konttialukset, säiliöalukset, risteilijät ja hinaajat.

Kuvassa 4.1 on kestoltaan sekunnin mittaisia ikkunoita eri alustyypeistä.



Kuva 4.1: Ikkunoituja audiokuvia

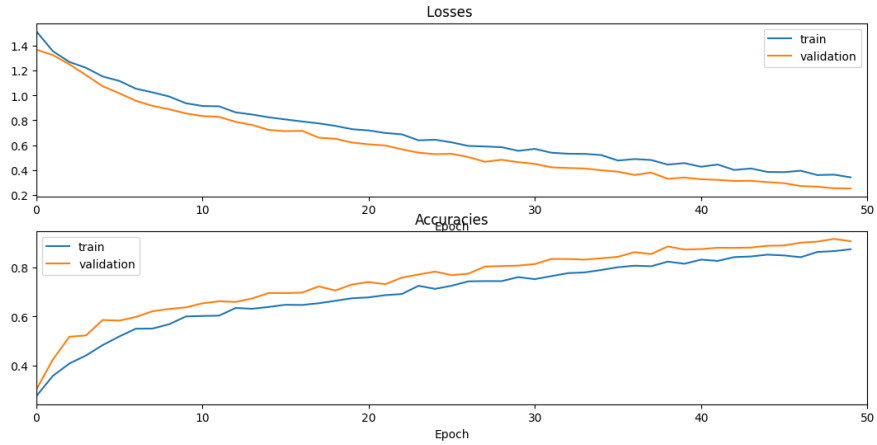
4.2 Mallin koulutus

Kaikille malleille käytetyt opetusparametrit on kuvattu kaaviossa 4.4. Ikkunat poimitaan datajoukosta ja kaikkia luokkia valitaan koulutukseen tasaisesti. Konvoluutiokerroksiin ja tiheisiin kerroksiin sovelletaan l2-regularisointia ja pois kytkentää ylisovittamisen välttämiseksi. Ikkunat normeerataan käyttäen kaavaa 2.1. Normeeraus tehdään yksi- ja kaksiuloteisille signaaleille.

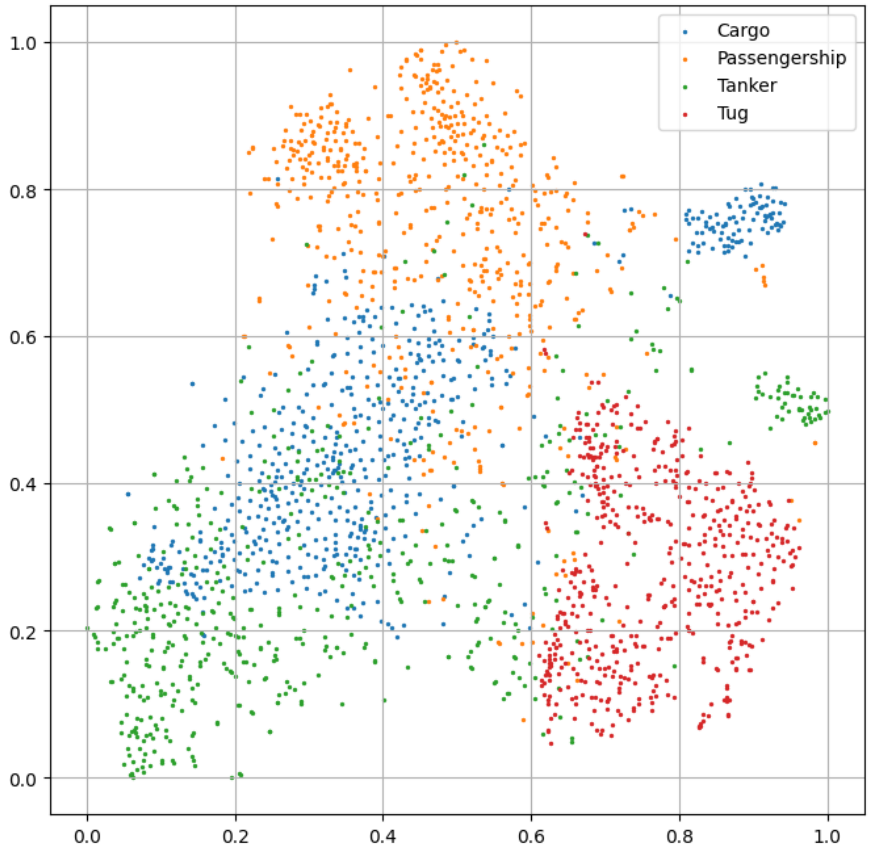
Taulukko 4.4: Koulutusparametrit

Parametri	1DCNN	SBCNN	YAMNet
Optimoija	Adam	Adam	Adam
Opetuskerroin	$6e-6$	$3e-4$	$3e-4$
Eräkoko	90	90	90
Epokit	50	50	60
Poiskytkentä	0.3	0.3	0.3
Aktivaatiofunktio	ReLU	ReLU	ReLU
l2-arvo	0.01	0.01	0.01

Tarkkuuden ja häviöfunktion muutos on esitetty kuvissa 4.2 ja 1DCNN-mallin piirretason separoituvuus kuvaajassa 4.3.



Kuva 4.2: Mallin opetusprosessi, 1DCNN, SE



Kuva 4.3: Piirrekerrosten luokkaseparaatio t-SNE -menetelmällä, 1DCNN (DS)

5 Tulokset ja pohdintaa

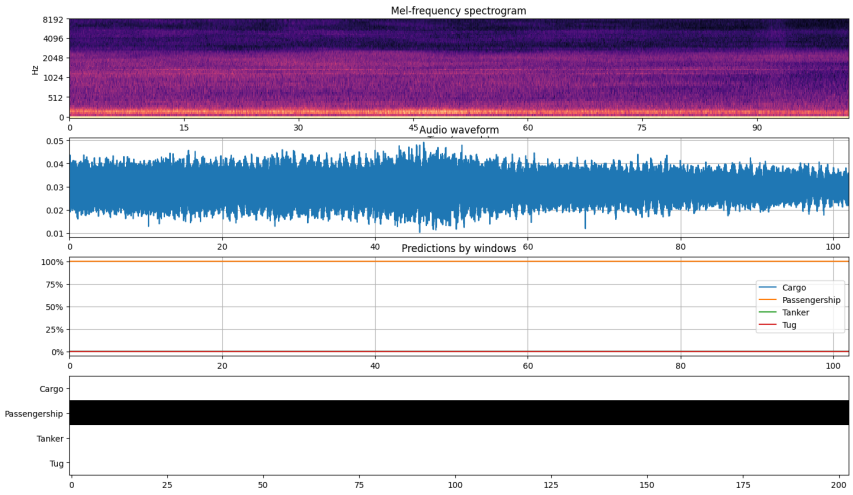
5.1 Mallin suorituskyvystä

Taulukko 5.1: Mallien tarkkuudet

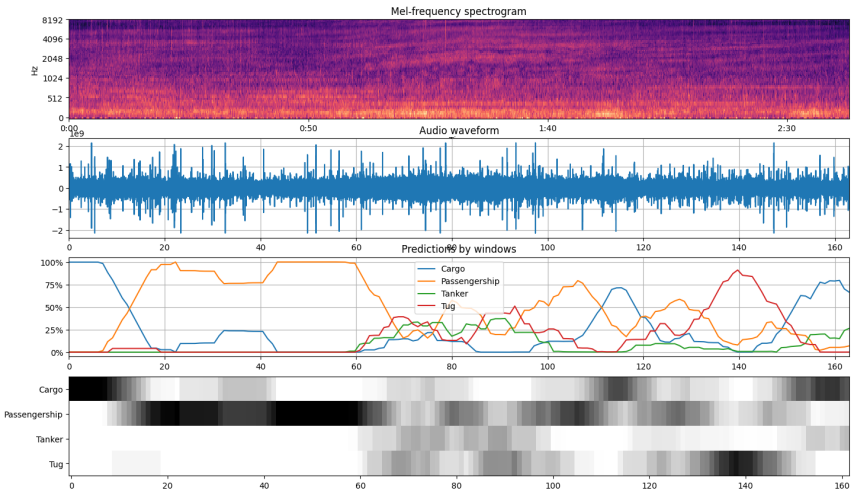
Malli	Tarkkuus (DS) (%)	Tarkkuus (SE) (%)
1DCNN	91.0%	77.9%
SBCNN	99.7%	97.3%
YAMNet	84.0%	83.1%

Mallien tarkkuudet taulukossa 5.1. Tuloksista huomataan, että erikoistuneet mallirakenteet suorituivat keskimäärin paremmin konemelun luokituksesta verrattuna MobileNetV1 perustuvaan malliin. Kaikki mallit luokittelivat DS-aineiston tarkemmin kuin SE-aineistoa. SBCNN-malli luokitteli testijoukon suurimmalla tarkkuudella. Lähtöainesitolla tehty luokitus on esitetty kuvassa 5.1. Luokitin ei kuitenkaan saavuta yksimielistä luokitusta toisen aineiston mittauksesta (kuva 5.2), vaan tuottaa muuttuvia luokitustuloksia samasta kohteesta. Voidaan siis päätellä, että malli on ylisovitettu alkuperäiseen datajoukkoo, eikä yleisty toiseen datajuokkoon. Mallien korkeamman luokitustarkkuuden DS-aineistossa (80%) voi selittää ikkunoiden suurempi päällekkäisyys verrattuna SE-aineistoon (20%). Mallin voi olettaa näkevän samaa dataa useammin, mitä suurempi on päällekkäisyys. Toisaalta sekä käytetyissä mittausjärjestelmissä että olosuhteissa on ollut aineistojen välillä eroja. Mallien

arvioinnissa ei ole sovellettu ristivalidointia tai datajoukkojen lohkoamista. Tällä saavutettavan virhearvion ei oleteta tuovan merkittävää vaikutusta, sillä käytetyt aineistot ovat itsessään suppeita. YAMNet-mallin päällekäisyyden arvo oli molemmissa datajoukoissa 40%, mikä voi selittää mallin keskimääräisen heikomman lokitustarkkuuden. Toisaalta mallin alkuperäisessä 521 luokan joukossa ei ole edustusta vedenalaisista mittauksista tai kohteista. Kaikkien mallien piirteiden separaatiot on esitetty liitteessä A.



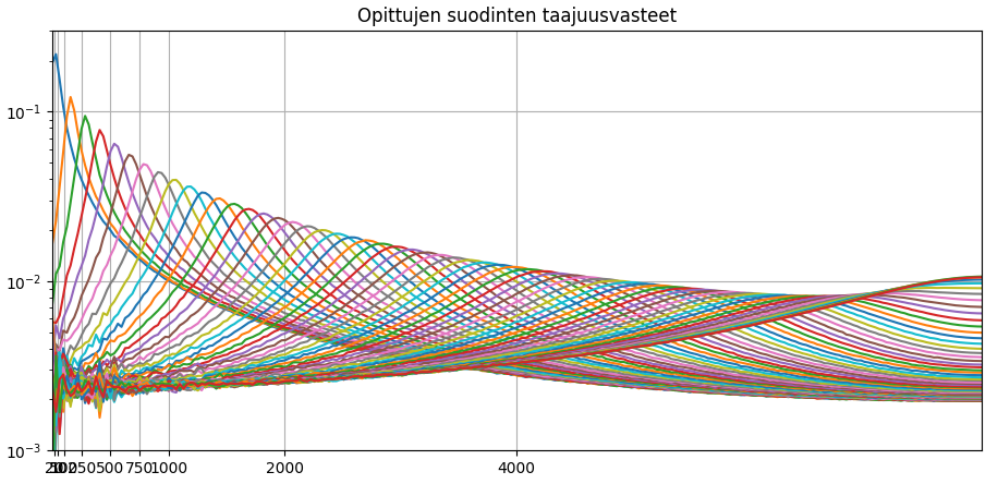
Kuva 5.1: SBCNN, koulutettu DS, mittauksessa risteilijä (SE)



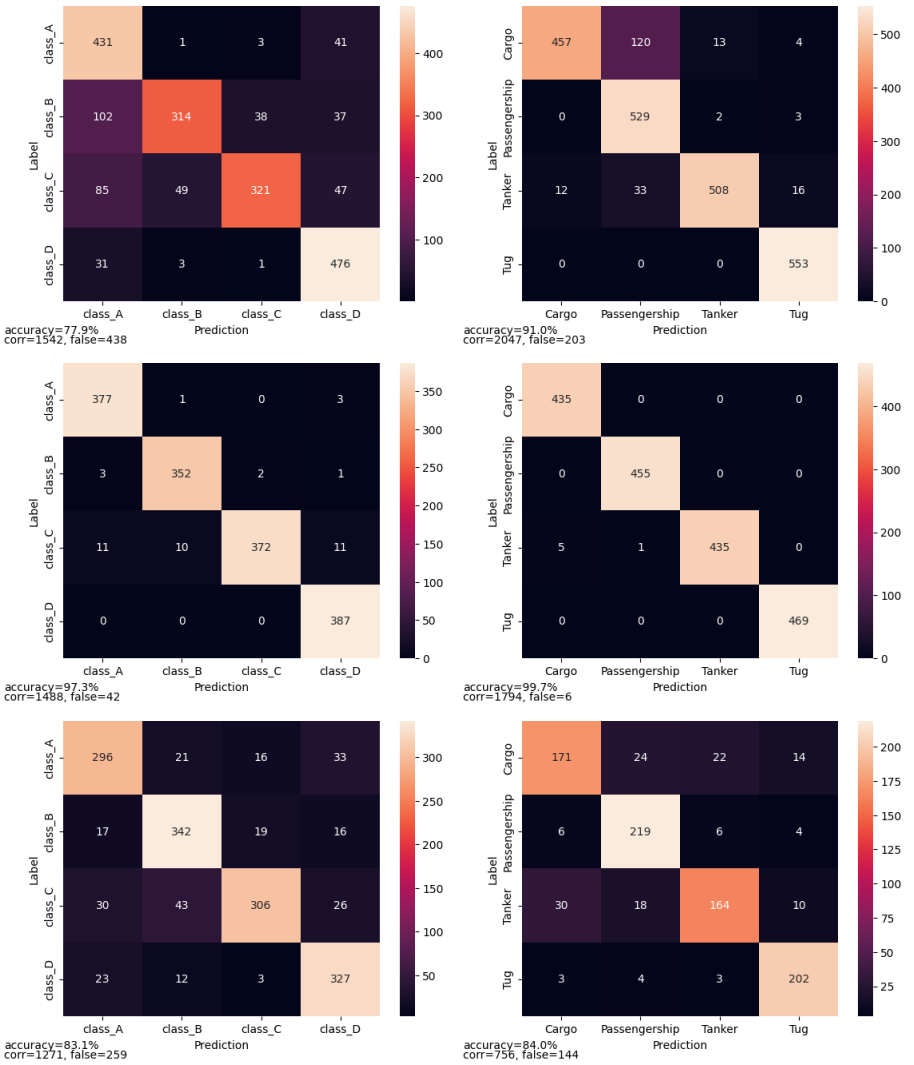
Kuva 5.2: SBCNN-malli, koulutettu: DS, mittauksessa riesteilijä (SE)

1DCNN-malli oppi kirjallisuudessa selvästi erkoistuneita suotimia [31]. DS-aineistosta

opitut suotimet näkyvät kuvassa 5.3. Suodinten taajuuskaistojen muodosta näkee, että malli ei erikoistanut merkittävästi gammatoonisuotimia. Oppimiskerroin oli molemmissa kokeissa samaa suuruusluokkaa, joten erilaisten tulsoten syy ei ole ilmeinen. Vähäinen suodinten muutos voi johtua gradientin häviämisestä tai painotetuista oppimiskertoimista. Pienestä opestusdatan määrästä johtuen havainnosta ei voi tehdä ratkaisevia päätelmiä.



Kuva 5.3: 1DCNN syötekerroksen oppimat suotimet



Kuva 5.4: Mallien sekaannusmatriisit (vas: SE, oik: DS, ylhäältä alas: 1DCNN, SBCNN, YAMNet)

5.2 Jatkokehitys

Esitetty malli ei ota huomioon aiemmin tapahtuneita luokitustuloksia tai piirteiden arvoja. Konemelun luokituksessa tämä voi osoittautua toivotuksi ominaisuudeksi. Yksi ratkaisu on käyttää luokituksessa takaisinkytkettyjä neuroverkkoja. Neuroverkkomallit käyttävät konvention mukaisesti liukulukuaritmetiikkaa matriiseille. Kertolaskuoperaatioiden lisäksi on tutkittu, miten muut operaatiot soveltuvat neuroverkkojen pohjaksi. XORNet-malli [44] toimii binäärisellä XOR-operaatorilla. Suuria neuroverkkomalleja voidaan supistaa kvantisoinnilla tai karsinnalla[45].

Data-ainestoja on kerätty yksittäisitä sijainneista läheltä valtameriä. Mallilta ei voida odottaa yhtäläista suorituskykyä useissa eri ympristöissä kuten matalilla murtovesialueilla. Kattavanmpien data-aineistojen kerääminen on välttämätöntä luotettavampien tuloksien saamiseksi.

6 Yhteenveto

Tässä tutkielmassa hahmoteltiin kohteiden tunnistukseen liittyviä vedenalaisen akustiikan ilmiötä, niiden tuomia haasteita ja datakeskeisiä ratkaisuehdotuksia ongelmaan. Siirto-oppimisella suurempi syväoppimismalli voidaan opettaa kohteiden tunnistukseen, mutta erikoistuneet mallirakenteet osoittautuvat vakuuttavammaksi ratkaisuksi. Pienempi malli on sovellettavissa pilviympäristön ulkopuolelle. Puhutaan sumu- ja reunalaskennan alueesta. Lopulta vedenalainen akustiikka luo haasteellisen ympäristön kohteiden luokittelulle. Pinta-alusten tunnistutehtävään koulutettiin kolme eri neuroverkkomallia käyttäen valtamerirannikon ympäristössä kerättyä data-aineistoa. Malleina olivat kuuloaistin inspiroima 1DCNN-malli (84.5%), äänimaisemien luokituksessa käytetty SBCNN-malli (98.5%) sekä YAMNet (83.5%). Mallit käyttävät syötteenä hydrofonilla mitattua dataa (1DCNN), mellog spektriä (SBCNN) datan johdannaispiirteitä (YAMNet). Mallit saavuttivat 77.9 - 91.0% tarkkuuden, mutta suppeasta koulutusdatan määrästä johtuen todellista luokitustarkkuutta on haastavaa arvioda. Mallien vertailu kahden aineiston välillä paljastaa, että mallit eivät tunnista yksimielisesti samaa alustyyppiä kahdella eri järjestelmällä mitattuna.

Lähdeluettelo

[1] D. Santos-Domínguez, S. Torres-Guijarro, A. Cardenal-López ja A. Pena-Gimenez, "ShipsEar: An underwater vessel noise database", eng, *Applied acoustics*, vol. 113, s. 64–69, 2016, ISSN: 0003-682X.

[2] M. Irfan, Z. Jiangbin, S. Ali, M. Iqbal, Z. Masood ja U. Hamid, "DeepShip: An underwater acoustic benchmark dataset and a separable convolution based autoencoder for classification", eng, *Expert systems with applications*, vol. 183, s. 115 270–, 2021, ISSN: 0957-4174.

[3] J. Ren, Y. Xie, X. Zhang ja J. Xu, "UALF: A learnable front-end for intelligent underwater acoustic classification system", *Ocean Engineering*, vol. 264, s. 112 394, 2022.

[4] X. Yuanchao, C. Zhiming ja K. Xiaopeng, "Improved pitch shifting data augmentation for ship-radiated noise classification", *Applied Acoustics*, vol. 211, s. 109 468, 2023.

[5] P. Zhu, Y. Zhang, Y. Huang, C. Zhao, K. Zhao ja F. Zhou, "Underwater acoustic target recognition based on spectrum component analysis of ship radiated noise", *Applied Acoustics*, vol. 211, s. 109 552, 2023.

[6] S. Smith et al., "The scientist and engineer's guide to digital signal processing", 1997.

[7] T. Virtanen, M. Plumbley ja D. Ellis, *Computational Analysis of Sound Scenes and Events*. syyskuu 2017, s. 1–422. DOI: 10.1007/978-3-319-63450-0.

[8] D. S. Park, W. Chan, Y. Zhang et al., "SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition", *Interspeech 2019*, syyskuu 2019. DOI: 10.21437/interspeech.2019-2680. url: http://dx.doi.org/10.21437/Interspeech.2019-2680.

[9] Y. Tokozume, Y. Ushiku ja T. Harada, "Learning from Between-class Examples for Deep Sound Recognition", *CoRR*, vol. abs/1711.10282, 2017. arXiv: 1711.10282. url: http://arxiv.org/abs/1711.10282.

[10] K. Xu, D. Feng, H. Mi et al., "Mixup-Based Acoustic Scene Classification Using Multi-Channel Convolutional Neural Network", *CoRR*, vol. abs/1805.07319, 2018. arXiv: 1805.07319. url: http://arxiv.org/abs/1805.07319.

[11] J. Salamon ja J. P. Bello, "Deep Convolutional Neural Networks and Data Augmentation for Environmental Sound Classification", *CoRR*, vol. abs/1608.04363, 2016. arXiv: 1608.04363. url: http://arxiv.org/abs/1608.04363.

[12] X. Yuanchao, C. Zhiming ja K. Xiaopeng, "Improved pitch shifting data augmentation for ship-radiated noise classification", eng, *Applied acoustics*, vol. 211, s. 109 468–, 2023, ISSN: 0003-682X.

[13] B. Katsnelson, V. Petnikov ja J. Lynch, *Fundamentals of shallow water acoustics*. Springer Science & Business Media, 2012.

[14] G. Grelowska ja E. Kozaczka, "Changes in conditions of acoustic wave propagation in the Gdansk deep as an effect of climate changes in the Baltic Sea region.", *Marine pollution bulletin*, vol. 160, s. 111 660, 2020. DOI: 10.1016/J.MARPOLBUL.2020.111660.

[15] F. A. R, S. J. V, C. A. B ja K. L. E, *Fundamentals of Acoustics 4th Edition*, eng. John Wiley et Sons, 1999, ISBN: 0471847895.

[16] J. Choi, Y. Choo ja K. Lee, "Acoustic Classification of Surface and Underwater Vessels in the Ocean Using Supervised Machine Learning", *Sensors*, vol. 19, nro 16, 2019, ISSN: 1424-8220. DOI: 10.3390/s19163492. url: https://www.mdpi.com/1424-8220/19/16/3492.

[17] F. Jensen, W. Kuperman, M. Porter ja H. Schmidt, *Computational Ocean Acoustics*. tammikuu 2000, vol. 47. DOI: 10.1063/1.2808704.

[18] B. C. Moore ja B. R. Glasberg, "Suggested formulae for calculating auditory-filter bandwidths and excitation patterns.", *The journal of the acoustical society of America*, vol. 74, nro 3, s. 750–753, 1983.

[19] "Machine Hearing for Industrial Fault Diagnosis", eng, vol. 2020-, s. 849–854, 2020, ISSN: 2161-8070.

[20] S. Shen, H. Yang, J. Li, G. Xu ja M. Sheng, "Auditory Inspired Convolutional Neural Networks for Ship Type Classification with Raw Hydrophone Data", *Entropy*, vol. 20, nro 12, 2018, ISSN: 1099-4300. url: https://www.mdpi.com/1099-4300/20/12/990.

[21] T. N. Sainath, B. Kingsbury, A.-r. Mohamed ja B. Ramabhadran, "Learning filter banks within a deep neural network framework", s. 297–302, 2013. DOI: 10.1109/ASRU.2013.6707746.

[22] *MATLAB: luokitusjärjestelmät*, https://se.mathworks.com/help/deeplearning/ug/multilabel-image-classification-using-deep-learning.html, Viitattu: 2024-04-05.

[23] J. Kittler, M. Hatef, R. Duin ja J. Matas, "On combining classifiers", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, nro 3, s. 226–239, 1998. DOI: 10.1109/34.667881.

[24] M. Grandini, E. Bagli ja G. Visani, "Metrics for multi-class classification: an overview", *arXiv preprint arXiv:2008.05756*, 2020.

[25] D. E. Rumelhart, G. E. Hinton ja R. J. Williams, ”Learning representations by back-propagating errors”, *nature*, vol. 323, nro 6088, s. 533–536, 1986.

[26] A. G. Howard, M. Zhu, B. Chen et al., ”Mobilenets: Efficient convolutional neural networks for mobile vision applications”, *arXiv preprint arXiv:1704.04861*, 2017.

[27] A. Barron, ”Universal approximation bounds for superpositions of a sigmoidal function”, *IEEE Transactions on Information Theory*, vol. 39, nro 3, s. 930–945, 1993. DOI: 10.1109/18.256500.

[28] R. Pascanu, T. Mikolov ja Y. Bengio, ”Understanding the exploding gradient problem”, *CoRR*, vol. abs/1211.5063, 2012. arXiv: 1211.5063. url: http://arxiv.org/abs/1211.5063.

[29] U. Michelucci, *Applied Deep Learning with TensorFlow 2: Learn to Implement Advanced Deep Learning Techniques with Python*, 2. painos. 2022, ISBN: 9781484280195; 1484280199; 9781484280201; 1484280202.

[30] K. He, X. Zhang, S. Ren ja J. Sun, ”Deep Residual Learning for Image Recognition”, *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, s. 770–778, 2015. DOI: 10.1109/cvpr.2016.90.

[31] S. Abdoli, P. Cardinal ja A. L. Koerich, ”End-to-End Environmental Sound Classification using a 1D Convolutional Neural Network”, *CoRR*, vol. abs/1904.08990, 2019. arXiv: 1904.08990. url: http://arxiv.org/abs/1904.08990.

[32] J. Koco’n, I. Cichecki, O. Kaszyca et al., ”ChatGPT: Jack of all trades, master of none”, *Inf. Fusion*, vol. 99, s. 101 861, 2023. DOI: 10.1016/j.inffus.2023.101861.

[33] J. Nordby, ”Environmental Sound Classification on Microcontrollers using Convolutional Neural Networks”, tutkielma, Norwegian University of Life Sciences, toukokuu 2019. url: http://hdl.handle.net/11250/2611624.

[34] O. Rodríguez, *Fundamentals of Underwater Acoustics*. Springer Nature Switzerland, 2023, ISBN: 9783031313189. url: https://books.google.fi/books?id=5925zwEACAAJ.

[35] X. Valero ja F. Alias, "Gammatone Cepstral Coefficients: Biologically Inspired Features for Non-Speech Audio Classification", eng, *IEEE transactions on multimedia*, vol. 14, nro 6, s. 1684–1689, 2012, ISSN: 1520-9210.

[36] E. Fonseca, X. Favory, J. Pons, F. Font ja X. Serra, "FSD50K: an Open Dataset of Human-Labeled Sound Events", *CoRR*, vol. abs/2010.00475, 2020. arXiv: 2010.00475. url: https://arxiv.org/abs/2010.00475.

[37] E. Tsalera, A. Papadakis ja M. Samarakou, "Comparison of pre-trained CNNs for audio classification using transfer learning", *Journal of Sensor and Actuator Networks*, vol. 10, nro 4, s. 72, 2021.

[38] S. Hershey, S. Chaudhuri, D. P. W. Ellis et al., "CNN Architectures for Large-Scale Audio Classification", *CoRR*, vol. abs/1609.09430, 2016. arXiv: 1609.09430. url: http://arxiv.org/abs/1609.09430.

[39] F. Liu, T. Shen, Z. Luo, D. Zhao ja S. Guo, "Underwater target recognition using convolutional recurrent neural networks with 3-D Mel-spectrogram and data augmentation", *Applied Acoustics*, vol. 178, s. 107 989, 2021, ISSN: 0003-682X. DOI: https://doi.org/10.1016/j.apacoust.2021.107989. url: https://www.sciencedirect.com/science/article/pii/S0003682X21000827.

[40] P. Zhu, Y. Zhang, Y. Huang, C. Zhao, K. Zhao ja F. Zhou, "Underwater acoustic target recognition based on spectrum component analysis of ship radiated noise", eng, *Applied acoustics*, vol. 211, s. 109 552–, 2023, ISSN: 0003-682X.

[41] K. J. Piczak, ”Environmental sound classification with convolutional neural networks”, teoksessa *2015 IEEE 25th international workshop on machine learning for signal processing (MLSP)*, IEEE, 2015, s. 1–6.

[42] J. Amoh ja K. Odame, ”An Optimized Recurrent Unit for Ultra-Low-Power Keyword Spotting”, *CoRR*, vol. abs/1902.05026, 2019. arXiv: 1902.05026. url: http://arxiv.org/abs/1902.05026.

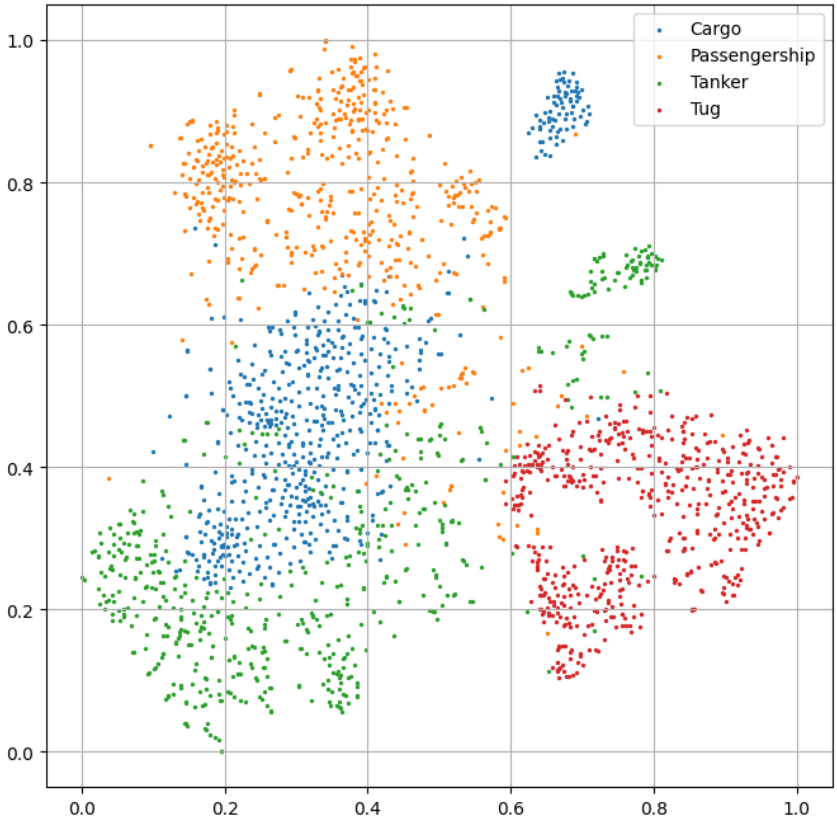
[43] K. J. Piczak, ”Environmental sound classification with convolutional neural networks”, teoksessa *2015 IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP)*, 2015, s. 1–6. DOI: 10.1109/MLSP.2015.7324337.

[44] M. Rastegari, V. Ordonez, J. Redmon ja A. Farhadi, ”XNOR-Net: ImageNet Classification Using Binary Convolutional Neural Networks”, *CoRR*, vol. abs/1603.05279, 2016. arXiv: 1603.05279. url: http://arxiv.org/abs/1603.05279.

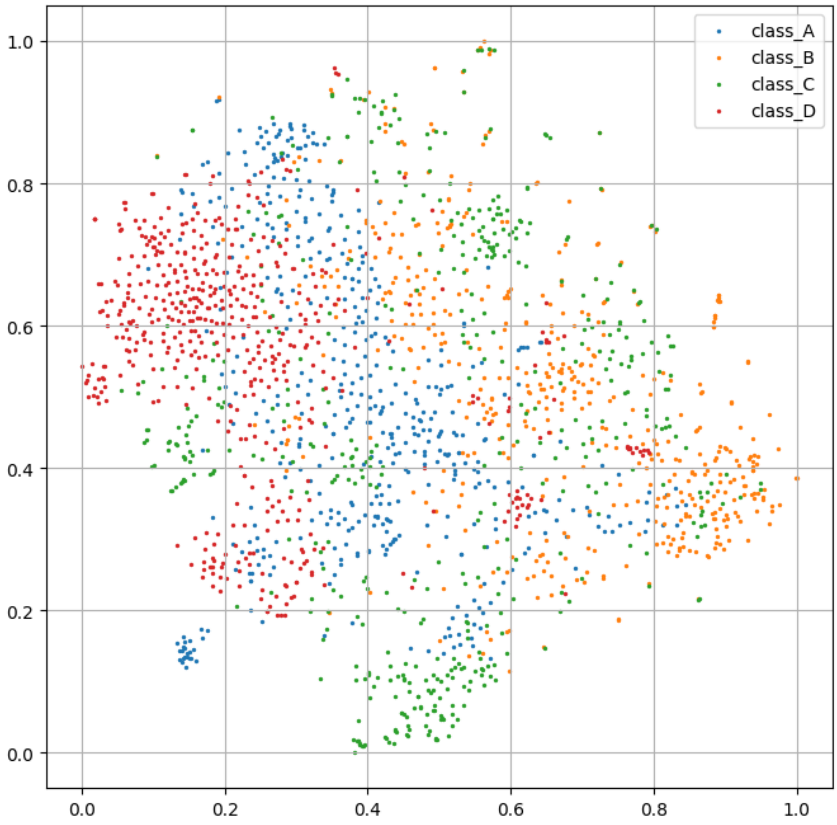
[45] M. Mohaimenuzzaman, C. Bergmeir ja B. Meyer, ”Pruning vs XNOR-Net: A Comprehensive Study of Deep Learning for Audio Classification on Edge-Devices”, *IEEE Access*, vol. 10, s. 6696–6707, 2022. DOI: 10.1109/ACCESS.2022.3140807.

Liite A Mallien koulutustulokset

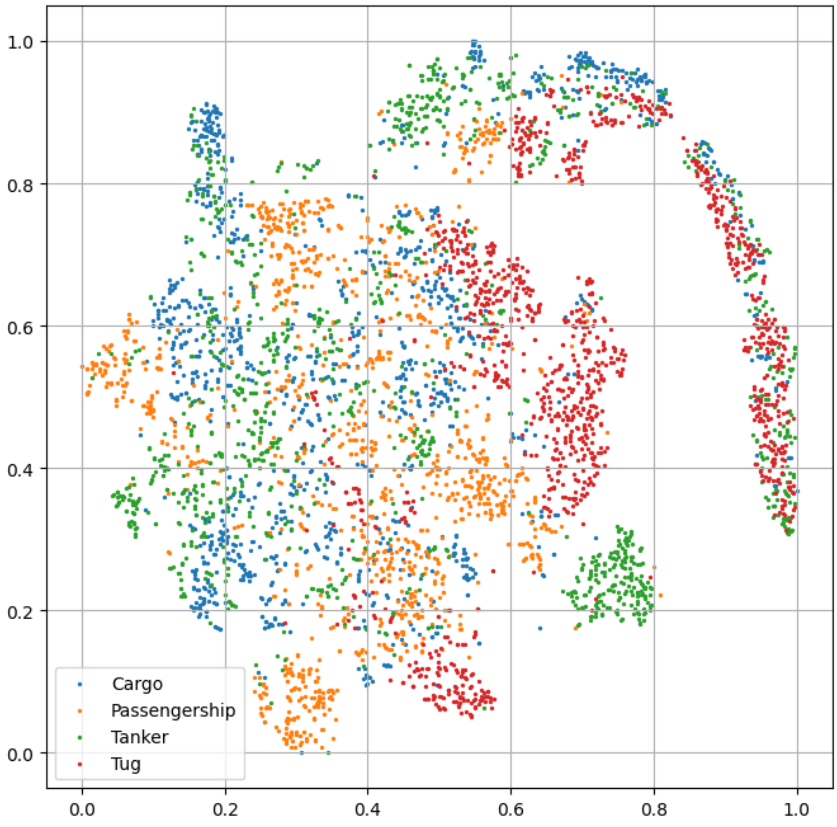
Tässä liittessä on esitetty koulutettujen mallien t-sne -visualisaatiot.



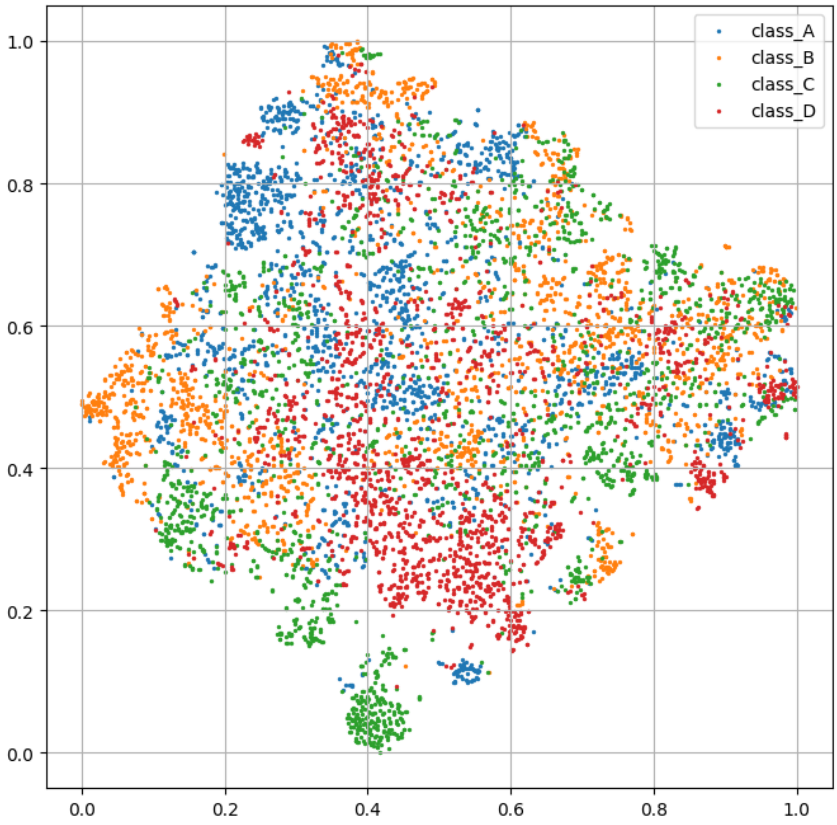
Kuva A.1: 1DCNN, aineisto: DS, t-sne



Kuva A.2: 1DCNN, aineisto: SE, t-sne



Kuva A.3: YAMNet, aineisto: DS, t-sne



Kuva A.4: YAMNet, aineisto: SE, t-sne