



Sentinel-1 SAR-kuvien kohinan poisto koneoppimisen näkökulmasta

Tarja Ajo

Opinnäytetyö, AMK

Toukokuu 2022

Tietojenkäsittely ja tietoliikenne

Insinööri (AMK), tieto- ja viestintäteknikka

Ajo Tarja

Sentinel-1 SAR-kuvien kohinan poisto koneoppimisen näkökulmasta

Jyväskylä: Jyväskylän ammattikorkeakoulu. Toukokuu 2022, 113 sivua.

Tietojenkäsittely ja tietoliikenne. Tieto- ja viestintäteknikan tutkinto-ohjelma. Opinnäytetyö AMK.

Julkaisun kieli: suomi

Julkaisulupa avoimessa verkossa: kyllä

Tiivistelmä

SAR-tutkakuvien käyttö maan pinnan tutkimiseen on viime vuosina yleistynyt optisten satelliittikuvien rinnalle. SAR-tutkien etu optisiin laitteisiin verrattuna on kyky kuvantaa maata sääolosuhteista ja valoisuudesta riippumatta. Tämän lisäksi SAR-tutkat antavat tietoa maan pinnan rakenteesta ja mahdollistavat kuvantamisen kasvillisuus- ja pintakerrosten läpi.

Jyväskylän ammattikorkeakoulun Tieto tuottamaan -hankkeen pilottitapauksessa tutkitaan mahdollisuutta hyödyntää SAR-kuvia maastoalueiden anomalioiden tunnistamiseen koneoppimisen avulla. SAR-kuvien käyttöä hankaloittaa kuitenkin kuvissa mustina ja valkoisina pisteinä näkyvä kohina, joka aiheutuu kuvantamiseen käytettyjen mikroaaltojen heijastumien vuorovaikutuksesta.

Kohinan poistamiseen on kehitetty lukuisia eri menetelmiä, joista ensimmäiset ovat olleet optisille kuville kehitettyjä. Myöhemmin kohinanpoistomalleja on muotoiltu erityisesti SAR-kuvissa esiintyvän kohinan poistamiseen. Viime vuosina tähän on sovellettu myös neuroverkkomalleja, joilla on saatu lupaavia tuloksia. Näiden innoittamana pilottitapauksen rinnalla haluttiin tutkia mahdollisuutta soveltaa neuroverkoilla kehitettyjä kohinanpoistomalleja pilottitapauksessa käytettävälle datalle ja saada tietoa käytettävän kohinanpoistomenetelmän optimaalisesta valinnasta. Koska anomalioiden tunnistaminen oli tavoitteena toteuttaa koneoppimisen avulla, tutkittiin kohinanpoistomenetelmän valintaa ja SAR-datasta hyödynnettäviä ominaisuuksia tästä näkökulmasta.

Kohinanpoistomenetelmien tutkimus toteutettiin valitsemalla vertailuun muutama potentiaalinen menetelmä aikaisemman tutkimustiedon pohjalta. Vertailuun valittiin Lee, Frost, Refined Lee -suodattimet, Frost ja Refined Lee -suodattimien yhdistelmä sekä Refined Lee -suodattimen soveltaminen kahdesti. Lee ja Frost -suodattimissa käytettiin kahta eri suodatusikkunankokoa. Näiden lisäksi neuroverkkopohjaisista menetelmistä tutkittiin Speckle2Void-mallia, mutta tämä ei sellaisenaan soveltunut pilottitapauksen datalle, minkä vuoksi käytännöntestaus tehtiin vain edellä luetelluille menetelmille.

Valittuja menetelmiä tutkittiin vertailemalla subjektiivisesti suodatettuja kuvia, laskemalla suodatukselle ENL-arvo, suodatukseen kulunut aika ja tutkimalla Random Forest -menetelmällä tehdyn maankäytön luokittelun tuloksia eri suodattimien kanssa.

Avainsanat (asiasanat)

Sentinel-1, SAR, kohina, kohinanpoisto, Random Forest, SNAP

Muut tiedot (salassa pidettävät liitteet)

Ajo Tarja

Speckle filtering of Sentinel-1 SAR-images from machine learning's point of view

Jyväskylä: JAMK University of Applied Sciences, May 2022, 113 pages.

Information and Communications. Degree Programme in Information and Communication Technology. Bachelor's thesis.

Permission for open access publication: Yes

Language of publication: Finnish

Abstract

The use of SAR images has increased in land cover studies during the last years. The advantage of synthetic aperture radars over optical imaging systems is their ability to image regardless of weather or time of day. Furthermore, synthetic aperture radars provide useful information about land surface and enable imaging through vegetation and land surface layers.

The use of SAR images in anomaly detection with the help of machine learning was studied in a pilot case in Jyväskylä University of Applied Science. The complication in using SAR images is the salt and pepper noise called speckle that makes interpretation and further use of images difficult. Filtering of speckle is an important part of SAR preprocessing.

Many speckle filtering methods have been developed over the years and recently deep learning methods have been applied to speckle filtering. These methods have shown promising results, but comparative studies are few. For this reason, the possibility to use new speckle filtering methods for the data used in pilot case was studied. Also, information about optimal selection of speckle filtering method and useful properties in SAR data was needed. Because the detection of anomalies was to be carried out using machine learning, the research was implemented from machine learning's point of view.

The research was carried out by choosing a few potential speckle filters based on previous studies. The chosen speckle filters were Lee, Frost, Refined Lee and combination of Frost and Refined Lee filters. In Lee and Frost filters filter window sizes of 5x5 and 7x7 were tested. In addition to former filters, new Speckle2Void speckle filter was planned to be tested. During research it became clear that this method was designed to be used to PolSAR data.

The speckle filters were compared carrying out a land cover classification with each filter using Random Forest. From the results of Random Forest land cover classification key figures were analysed. Preprocessed SAR images were also examined subjectively, calculating ENL-value and comparing processing times.

Keywords/tags (subjects)

Sentinel-1, SAR, speckle, speckle filtering, Random Forest, SNAP

Miscellaneous (Confidential information)

Sisältö

| | | |
|----------|---|-----------|
| 1 | Johdanto | 5 |
| 2 | Tutkimusasetelma | 6 |
| 2.1 | Tutkimusongelman kuvaus | 6 |
| 2.2 | Tutkimuksen tavoitteet ja tutkimuskysymykset | 6 |
| 2.3 | Tutkimusmenetelmät | 7 |
| 3 | SAR-tutkateknologia | 9 |
| 3.1 | Tutkateknologioiden perusteita | 9 |
| 3.2 | SLAR- ja SAR-tutkat | 9 |
| 3.3 | SAR-kuvat | 12 |
| 3.3.1 | Säteilyn takaisinsironna ja kuvan muodostus | 12 |
| 3.3.2 | Geometriset vääristymät | 13 |
| 3.3.3 | Aallonpituuden vaikutus tutkavasteeseen | 13 |
| 3.3.4 | Dielektrinen vakio | 14 |
| 3.3.5 | Polarisaation hyödyntäminen | 14 |
| 3.3.6 | Kohina SAR-kuvissa | 15 |
| 4 | Sentinel-1-satelliitit ja niiden SAR-data | 17 |
| 4.1 | Sentinel-1-satelliitit | 17 |
| 4.2 | Sentinel-1-satelliittien SAR-data | 17 |
| 4.3 | SAR-datan esikäsittely | 18 |
| 4.4 | Polarimetrisen SAR-datan käsittely | 21 |
| 5 | Konvoluutioneuroverkot ja neuroverkkojen kouluttaminen | 22 |
| 5.1 | Neuroverkot | 22 |
| 5.1.1 | Neuroverkkojen yleinen rakenne | 22 |
| 5.1.2 | Neuroni | 22 |
| 5.1.3 | Aktivaatiofunktiot | 23 |
| 5.2 | Konvoluutioneuroverkot | 23 |
| 5.3 | Blind-spot-tekniikka | 25 |
| 5.4 | Koulutus- ja testidata | 25 |
| 6 | Kohinanpoistomenetelmät | 27 |
| 6.1 | Kohinan poistamisen tavoitteet, menetelmien jaottelua ja ongelmanratkaisutapoja... 27 | |
| 6.2 | Kohinanpoistomenetelmien vertailuun käytettyjä metriikoita ja kaavoissa esiintyviä käsitteitä..... 28 | |
| 6.3 | Tunnettuja kohinanpoistomenetelmiä | 30 |
| 6.3.1 | Boxcar | 30 |

| | | |
|-----------|--|-----------|
| 6.3.2 | Median | 31 |
| 6.3.3 | Frost | 31 |
| 6.3.4 | Lee | 32 |
| 6.3.5 | Refined Lee | 33 |
| 6.3.6 | Lee Sigma ja Improved Sigma Lee | 34 |
| 6.3.7 | Gamma Maximum a Posteriori (Gamma MAP) | 36 |
| 6.3.8 | Kuan | 36 |
| 6.4 | Viime vuosien kohinanpoistomenetelmiä | 37 |
| 6.4.1 | Noise2Void..... | 37 |
| 6.4.2 | Laine, Karras, Lehtinen ja Aila..... | 38 |
| 6.4.3 | Speckle2Void..... | 39 |
| 6.5 | Kohinanpoistomenetelmien tutkimus | 41 |
| 6.5.1 | Vertailevat teorialtutkimukset | 42 |
| 6.5.2 | Vertailevat tapaustutkimukset | 45 |
| 6.6 | Kohinanpoistomenetelmien valinta maankäytön luokittelua koskevissa tutkimuksissa | 45 |
| 6.7 | Kohinanpoistomenetelmien tutkimukseen liittyviä havaintoja..... | 47 |
| 7 | Kohinanpoistamisen vaikutuksesta SAR-kuvien luokitteluun koneoppimisen avulla ... | 49 |
| 7.1 | SAR-kuvien luokitteluongelmat..... | 49 |
| 7.2 | SAR-kuvien segmentointi | 50 |
| 8 | Toteutuksen suunnitelma | 53 |
| 8.1 | Vertailuun valitut kohinanpoistomenetelmät | 53 |
| 8.2 | Menetelmien vertailu..... | 55 |
| 8.3 | Tutkimuksessa käytetyt teknologiat | 56 |
| 9 | Tutkimuksen toteutus..... | 58 |
| 9.1 | Tutkimusaineiston kerääminen..... | 58 |
| 9.2 | Yksittäisen SAR-kuvan esikäsittely ja alueiden vertailu | 59 |
| 9.3 | Speckle2Void-mallin käyttö kohinan poistamiseen | 61 |
| 9.4 | Random Forest -luokittelu | 63 |
| 9.5 | Luokittelussa hyödynnettävän SAR-datan tutkiminen | 65 |
| 10 | Tutkimustulokset..... | 67 |
| 10.1 | Kohinanpoistomenetelmien vertailu | 67 |
| 10.1.1 | Suodatettujen kuvien subjektiivinen vertailu ja ENL-arvot..... | 67 |
| 10.1.2 | Menetelmien ajallinen vertailu | 68 |
| 10.1.3 | Random Forest -luokittelun tulokset..... | 69 |
| 10.2 | SAR-datan hyödyntäminen maaston luokitteluongelmissa..... | 72 |

| | |
|--|-----------|
| 11 Tulosten tulkintaa | 76 |
| 11.1 SAR-teknologian tuottaman datan hyödyntäminen maastonluokitteluongelmissa | 76 |
| 11.2 Neuroverkkoihin perustuvat menetelmät kohinan poistamisessa..... | 76 |
| 11.3 Kohinanpoistomenetelmän valinta SAR-kuvien esikäsittelyssä ja Suomen maastoalueille sopivat kohinanpoistomenetelmät | 78 |
| 11.3.1 Kohinanpoistomenetelmän valinta SAR-kuvien esikäsittelyssä | 78 |
| 11.3.2 Suomen maastoalueille sopiva menetelmä..... | 80 |
| 11.3.3 Speckle2Void-menetelmä kohinan poistamisessa | 81 |
| 12 Pohdinta | 83 |
| 12.1 Tutkimuksen luotettavuus | 83 |
| 12.2 Jatkotutkimusaiheita | 84 |
| Lähteet | 85 |
| Liitteet | 94 |
| Liite 1. Suodatuksen tulokset RGB-kuvina | 94 |
| Liite 2. ENL-arvot | 97 |
| Liite 3. Random Forest -luokittelun tulokset | 98 |
| Liite 4. Kohinanpoistoon kulunut aika..... | 103 |
| Liite 5. SAR-kuvien esikäsittely snappy-moduulilla | 104 |
| Liite 6. ENL-arvojen laskeminen | 108 |
| Liite 7. Esikäsittelyn funktiot | 111 |
| Kuviot | |
| Kuvio 1. SLAR-kuvantamisen suunnat ja etäisyydet | 10 |
| Kuvio 2. SLAR-kuvantamisen kulmia ja käsitteitä | 10 |
| Kuvio 3. SAR-datan esikäsittelyn vaiheet..... | 19 |
| Kuvio 4. Reunan tunnistusikkunat. | 34 |
| Kuvio 5. GRD-kuvat sisältöinen SNAP-ohjelmassa..... | 58 |
| Kuvio 6: Yksittäisen kuvan käsittely, ENL-arvojen laskeminen ja käsittelyyn kuluneen ajan mittaaminen..... | 59 |
| Kuvio 7. Koulutus- ja testidatan rajaaminen..... | 65 |
| Kuvio 8. Random Forest -luokittelun tuloksia kuvana. | 72 |
| Kuvio 9. VH-polarisaation intensiteetti-arvot heinäkuussa ja kesäkuussa. | 73 |
| Kuvio 10. Kahden SAR-kuvan pinottu RGB-kuva ja aluetta vastaava maastokartta..... | 74 |
| Taulukot | |

| | |
|--|----|
| Taulukko 1. SAR-tutkien aallonpituuksia ja käyttökohteita..... | 13 |
| Taulukko 2. Polarisaatioiden takaisinsironnan suhteellinen määrä. | 15 |

1 Johdanto

Satelliittikuvat ovat tuoneet valtavasti uutta tietoa käden ulottuville maapallon ja ympäröivän avaruuden tutkimiseen. Satelliittikuvia on tähän mennessä hyödynnetty niin turvallisten merireittien selvittämiseen kuin tulvien seurantaan ja käyttömahdollisuuksia olisi myös suunnattomasti enemmän teknologian sen salliessa. Satelliiteissa on useita spektroskopiaan, optiikkaan ja tutkateknologioihin perustuvia kuvantamismenetelmiä, joista yksi on Synthetic Aperture Radar, lyhyemmin SAR-tutka. SAR-tutkan avulla ympäristöä on mahdollista kuvantaa pimeässä, sade- ja pilvialueiden, sekä tarvittaessa myös kasvillisuuskerrosten läpi. Tämän lisäksi tutkan vastaanottamasta signaalista on mahdollista saada tietoa maanpeitteen ja materiaalien koostumuksesta. SAR-teknologiasta on tullut arvokas lisä optisten kuvantamislaitteiden rinnalle, joita esteet näkyvyydessä ovat pitkälti rajoittaneet.

Jyväskylän ammattikorkeakoulun Tieto tuottamaan -hankkeen pilottitapauksessa tutkitaan mahdollisuutta tunnistaa anomalioita ja luokitella niiden aste satelliittikuvista koneoppimisen avulla. SAR-kuvat ovat valikoituneet tutkittavaksi kohteeksi, sillä näitä on mahdollista saada sääolosuhteista riippumatta. Yksi SAR-kuvien hyödyntämistä rajoittava tekijä on kuitenkin kuvissa näkyvä kohina, jota esiintyy kuvantamiseen käytettävästä laitteistosta riippumatta. Kohina voi tehdä kuvista epätarkkoja ja hankalia hyödyntää. Ongelmaan on pyritty vastaamaan kehittämällä erilaisia kohinanpoistomenetelmiä, joilla SAR-kuvia käsitellään jälkikäteen. Viime vuosina menetelmissä on myös alettu soveltaa konvoluutioneuroverkkoja, joiden on todettu olevan erinomaisia muun muassa kuvien luokittelussa.

Tässä opinnäytetyössä tutkittiin ja vertailtiin satelliittikuvissa esiintyvän kohinan poistoon kehitettyjä menetelmiä ja pyrittiin löytämään optimaalisin menetelmä Suomen maastoalueita kuvaavien SAR-kuvien suodattamiseen. Työssä nostettiin tarkasteluun uusia neuroverkoilla kehitettyjä menetelmiä vanhempien menetelmien rinnalle, sekä tarkasteltiin kohinanpoistomenetelmän valintaa koneoppimisella tehtävän luokittelun näkökulmasta. Kohinanpoistomenetelmien vertailu toteutettiin soveltamalla niitä Euroopan avaruusjärjestön (ESA) Sentinel-1-satelliittien SAR GRD-kuviin ja vertailemalla tuloksia sekä subjektiivisesti, että Random Forest -luokittelun kautta.

2 Tutkimusasetelma

2.1 Tutkimusongelman kuvaus

Hankkeen pilottitapauksessa tutkitaan ja pilotoidaan mahdollisuutta tunnistaa erilaisia anomaliaita Suomen maastoalueilla satelliittikuvien ja koneoppimisen avulla. Tällaisia anomaliaita voivat esimerkiksi olla maastopaloalueet, myrskyjen aiheuttamat tuhoalueet metsissä tai tulvat. Yksittäisiä ja toisistaan kaukana olevia tuhoalueita voi olla lukuisia, joita on maan päältä hankalaa havaita ja aikaa vievää tarkistaa tuhojen laajuutta. Yhtenä esimerkkinä viimeaikaisista anomaliaista voi mainita Paula-myrskyn aiheuttamat tuhoalueet, jotka olivat laajimmillaan 10–30 hehtaarin kokoisia (Paula-myrsky teki Koillismaan metsiin miljoonavahingot 2021.). Tämän suuruusluokan anomalioiden voisi jo olettaa olevan havaittavissa satelliittikuvista ja mahdollisesti tunnistettavissa koneoppimisen avulla. Vaikka optiset satelliittikuvat ovat parhaimmillaan tarkkoja ja näistä voi havaita pieniäkin muutoksia, liittyy niiden käyttöön säästä ja vuorokauden ajasta johtuvia rajoituksia. Siksi pilottitapauksessa keskitytään SAR-kuvien hyödyntämiseen, missä nämä tekijät eivät ole este.

2.2 Tutkimuksen tavoitteet ja tutkimuskysymykset

Olennainen osa hankkeen pilottitapausta on saatavilla olevaan dataan ja SAR-kuviin perehtyminen, jotta kuvia osataan tulkita ja anomaliat voidaan tunnistaa kuvista. Kattava pohjatyö on keskeistä myös ongelman mahdollisten lähestymistapojen tunnistamiseksi, mahdottomien ratkaisutapojen poissulkemiseksi ja oikeiden tietojen hyödyntämiseksi. Siten tutkimuksessa halutaan koostaa riittävä ymmärrys SAR-teknologiasta ja SAR-kuvien tulkinnasta.

Tärkeä osa koneoppimismenetelmien käyttöä on datan esikäsittely, jolla on merkittävä vaikutus tulosten tarkkuuteen ja luotettavuuteen (Gulati & Raheja 2021, 195). Datan esikäsittelyn merkitys tuloksiin näkyy niin perinteisemmissä päätöspuumenetelmissä (Gulati & Raheja 2021, 195), kuin konvoluutioneuroverkkojen avulla tehtävässä kuvienluokittelussakin (Pal & Sudeep 2016, 1780). Datasta riippuen kuvien esikäsittelyyn voi kuulua muun muassa datan lisäämistä, keskiarvon ja varianssin normalisointia, kuvien koon muokkausta tai niiden käsittelyä kohinan poistamiseksi. SAR-datan tapauksessa kohinan poistamista tehdään, sillä poikkeuksetta kaikissa SAR-kuvissa esiintyy jossain määrin valkoisia, mustia ja harmaansävyisiä pikseleitä näkyvää kohinaa, joka voi tehdä kuvista epäselviä ja hankaloittaa kuvien tulkintaa ja kohteiden tunnistamista kuvista. Erilaisista kohinanpoistomenetelmistä on tehty vertailevia tutkimuksia, mutta uusia syväoppimiseen perustuvia

menetelmiä on kehitetty aivan viime vuosina ja vertailevia tutkimuksia koneoppimisen näkökulmasta on vähän.

Hankkeen pilottitapauksessa pyritään ensisijaisesti hyödyntämään ESA:n avoimena datana tarjoamia Sentinel-1-satelliittien SAR-kuvia. Kyseisiä kuvia on saatavilla eri versioina, raakadatana ja osittain esikäsiteltyinä. Osittain esikäsitellytkin kuvat tarvitsevat vielä viimeistelyä niiden jatkokäyttöä varten, mutta tarvittavat vaiheet, näiden joukossa kohinan poistaminen, riippuvat siitä mihin kuvia halutaan käyttää. Tästä johtuen eri suodattimia ja näiden konfiguraatioita kehoitetaan vertailemaan sopivimman tuloksen saavuttamiseksi (Braun 2020, 10). Tässä opinnäytetyössä pyritään löytämään sopivin kohinanpoistomenetelmä pilottitapauksessa käytettävälle Sentinel-1-satelliittien IW-moodin GRD-datalle ja tutkimaan, voidaanko kuvien käsittelyyn hyödyntää uusia neuroverkkoihin perustuvia menetelmiä. Jos valitulla kohinanpoistomenetelmällä on vaikutusta pilottitapauksen kuvien luokitteluprosessiin, halutaan myös tietää missä määrin valittu menetelmä vaikuttaa tuloksiin.

Edellä esitetty tutkimusongelma voidaan pukea seuraavien tutkimuskysymysten muotoon, joihin pyritään vastaamaan sekä tietopohjan että tutkimuksen avulla.

- Mitä SAR-tutkateknologian tuottamaa dataa voidaan hyödyntää maastonluokitteluongelmissa?
- Millaisia neuroverkkoihin perustuvia menetelmiä SAR-tutkakuvissa esiintyvän kohinan poistamiseksi on kehitetty? Mihin menetelmät perustuvat ja miten ne poikkeavat toisistaan?
- Miten esikäsitelyssä käytettävä kohinanpoistomenetelmä tulisi valita?
- Soveltuuko jokin kohinanpoistomenetelmä erityisen hyvin Suomen maastoalueille?

2.3 Tutkimusmenetelmät

Tutkimusmenetelmänä työssä käytetään tapaustutkimusta, jonka valintaa puoltaa Laineen, Bambergin ja Jokisen (2008, 9–12) esittämä tapaustutkimuksen määritelmä ja vertailu tilastolliseen tutkimukseen nähden. Määrällisen ja tapaustutkimuksen vertailussa Laine ja muut (2020, 12) nostavat tapaustutkimuksen piirteistä esille erityisesti tutkimuskohteen ja tutkimuksen päämäärän.

Opinnäytetyön tavoitteena on ymmärtää SAR-kuvien tulkintaa ja luokittelua, sekä tutkia pientä joukkoa kohinanpoistomenetelmiä rajatun tyyppisille SAR-kuville, mikä vastaa paremmin tapaus-tutkimuksen luonnetta kuin määrällistä tutkimusta. Vaikka Laineen ja muiden (2020, 12) tekemän vertailun perusteella tutkimusmenetelmäksi tulisi valita tapaus-tutkimus, huomauttavat he, ettei tämä sulje pois myös määrällisen tutkimuksen aineistojen tai menetelmien käyttöä tapaus-tutkimuksessa. Edellä esitettyjen tavoitteiden saavuttamiseksi tulevassa opinnäytetyössä onkin tarkoitus hyödyntää sekä määrällisen tutkimuksen menetelmiä että aikaisemmista tutkimuksista saatuja tietoja, mikä Laineen ja muiden mukaan ei ole tapaus-tutkimuksille tavatonta. (Laine ym. 2020, 9–12.)

Tapaus-tutkimuksen valintaan johdattaa myös käytettävä aineisto. Laine ja muut (2020, 12) kirjoittavat tapaus-tutkimusten otoksista seuraavaa: *”Tapaus-tutkimuksessa ei saada samassa mielessä edustavaa otosta, sillä vaikka tyypillisiä tapauksia löytyykin, keskimääräistä tapausta ei ole olemassa.”*. Mahdollisesti maankäytön osuuksien perusteella voitaisiin valita datasettiin otos kaupunki-, metsä-, pelto- ja järviolueista Suomessa, mutta käytännössä SAR-kuvien sisältö vaihtelee paljon, eikä kuvista voi erottaa tyypillistä tapausta. Lisäksi työn rajaamisen takia ei voitane esikäsitellä, luokitella ja analysoida suurta määrää SAR-kuvia, mikä tyypillisesti on määrällisen tutkimuksen tunnusmerkkejä (Vilka 2007, 17). Tutkimuksessa pyritään tekemään vertailua käytettyjen suodattimien kesken numeerisesti, mutta tilastollisten tunnuslukujen laskeminen ei ole tutkimuksen tavoitteena.

3 SAR-tutkateknologia

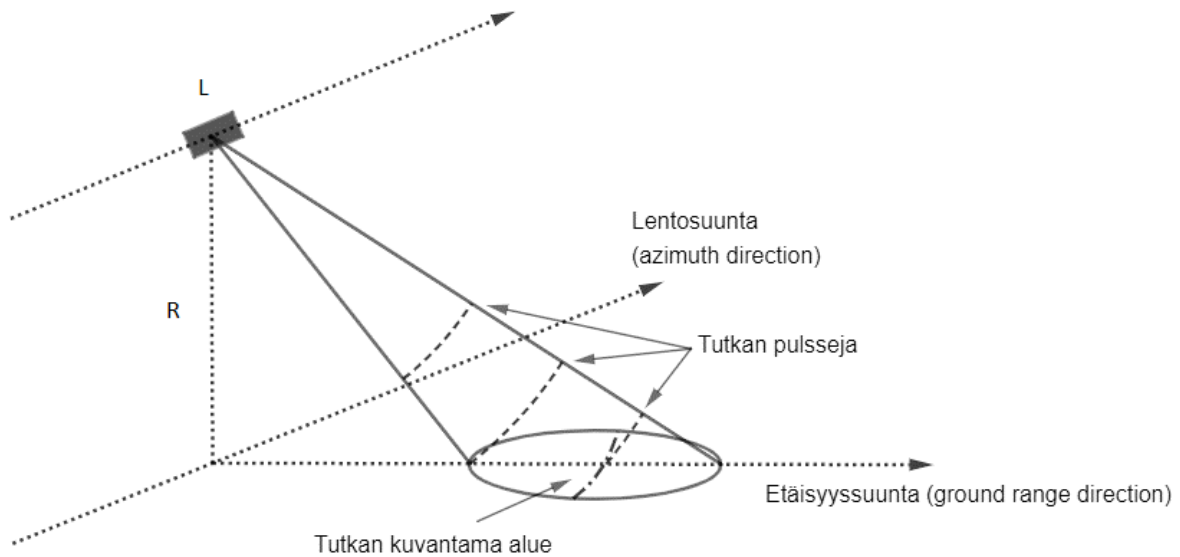
3.1 Tutkateknologioiden perusteita

Tutkateknologiat perustuvat aineiden ja kappaleiden ominaisuuteen heijastaa, läpäistä ja absorboida elektromagneettisen säteilyn eri aallonpituuksia. Aineen molekyyli rakenne ja fysikaaliset ominaisuudet määrittävät sen läpäisemän säteilyn tyypin ja määrän, mitä voidaan hyödyntää, kun kappaleiden rakenteesta halutaan saada tietoa ja kappaleita halutaan tutkia väliaineen läpi (Souyris 2014, 115–116; Smith 2008, 471–473). Kappaleista takaisin heijastuva säteily on mahdollista vastaanottaa ja kun säteily lähetetään pulssina, voidaan kappaleen etäisyys havaintopaikasta määrittää pulssin edestakaiseen matkaan käyttämän ajan perusteella. Jos säteilyä kohdistetaan havainnoitavaan ympäristöön pulsseina tietyllä frekvenssillä, voidaan ympäristöä myös havainnoida yhtäjaksoisesti. (Meyer 2019, 21.)

3.2 SLAR- ja SAR-tutkat

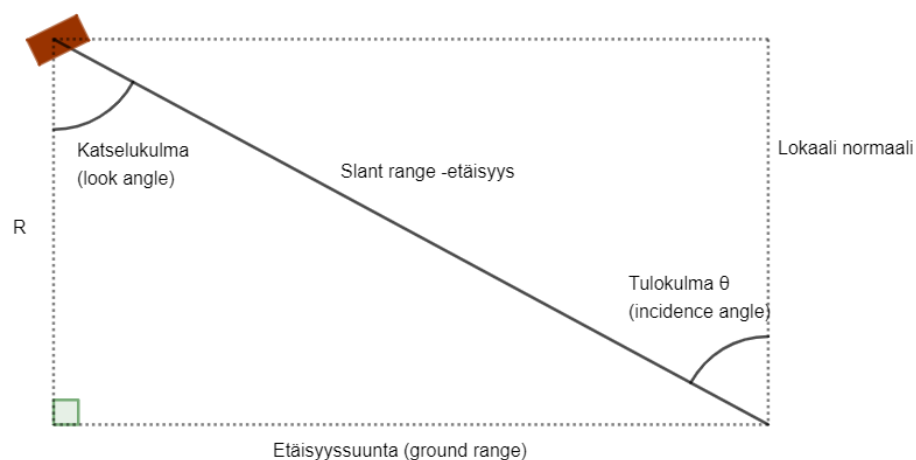
Käyttötarkoituksesta riippuen tutkan lähettämä signaali voidaan kohdistaa koko ympäristöön tai keskitetysti tiettyyn suuntaan. Kun signaali suunnataan viistosti pulsseina havainnoitavaan kohteeseen, voidaan tasomaisesta kappaleesta erottaa pinnanmuotoja takaisin heijastuneiden signaalien aikaerojen perusteella. Ilmiö mahdollistaa tutkateknologian käyttämisen maanpinnan kuvantamistarkoituksiin ja on tullut tunnetuksi nimellä Side-Looking Airborne Radar (SLAR). Karkeasti tutkateknologiat jaetaan kuvantaviin ja ei-kuvantaviin tutkiin (Podest 2018). (Woolfson 2011, 117–125; Meyer 2019, 22.)

Kuvantavien tutkien etu verrattuna optisiin kameroihin on niiden kyky kuvantaa valaistuksesta ja säästä riippumatta. Koska tutkat ovat kuvantamiseen käytetyn säteilyn lähde ja riippumattomia valosta, voidaan kuvia saada myös öisin. Tämän lisäksi käytetyt mikroaallot läpäisevät ohuempia pilvikerroksia, minkä vuoksi pilvien luoma optinen este ei ole ongelma kuvantaville tutkille. (Meyer 2019, 21.)



Kuvio 1. SLAR-kuvantamisen suunnat ja etäisyydet

SLAR-kuvien muodostuksessa keskeisiä käsitteitä ovat kulmat ja etäisyydet, joiden avulla kuvan erottelukyky, eli resoluutio, määritellään. Näihin käsitteisiin pohjautuen tehdään myös myöhemmin tarvittavat perspektiivistä johtuvat korjaukset. SLAR-tekniikalla kuvannettaessa saadun kuvan spatiaalinen resoluutio muodostuu tutkan etäisyys suunnan (ground range direction) ja lentosuunnan (azimuth direction) erottelukyvyyistä. Tutkan lentosuunta on nimensä mukaisesti tutkan lento-reitin mukainen suunta. Etäisyysuunta puolestaan on lentosuuntaa kohtisuorassa oleva suunta, johon tutkan signaali kohdistetaan (ks. kuvio 1). (Van Zyl & Kim 2011, 3–4.)



Kuvio 2. SLAR-kuvantamisen kulmia ja käsitteitä

Maan pinnan myötäinen etäisyserottelukyky ρ_G (eng. ground range resolution) saadaan johdettua tutkan signaalin myötäisestä viistosuuntaisesta resoluutiosta ρ_P (2) (eng. slant range resolution). Etäisyysuuntainen resoluutio puolestaan määräytyy kuvantamiseen käytetyn pulssin kestosta τ . Kaksi maan pinnalla olevaa kappaletta voidaan erottaa toisistaan, jos näistä takaisin heijastuneiden kaikujen aikaero on enemmän kuin pulssin kesto. Lyhyemmillä etäisyyksillä kappaleista takaisin heijastuneet signaalit limittyvät, eikä näistä voida enää erottaa onko kaiku tullut yhdestä vai useammasta kohteesta. Trigonometrian ja Kuviossa 2 esitetyn tulokulman θ avulla signaalin suuntainen viistosuuntainen resoluutio voidaan muotoilla etäisyysuuntaiseksi resoluutioksi (2). (Meyer 2019, 22; Van Zyl & Kim 2011, 7.)

$$\begin{aligned} (1) \quad & \rho_P = \frac{c \cdot \tau}{2} \\ (2) \quad & \rho_G = \frac{\rho_P}{\sin \theta_i} = \frac{c \cdot \tau}{2 \cdot \sin \theta_i} \\ (3) \quad & \rho_A \approx \frac{\lambda}{L} \cdot R \end{aligned}$$

Lentosuunnan mukainen resoluutio ρ_A määritellään vastaavalla idealla, mutta resoluutio määräytyy käytettävästä aallonpituudesta λ , antennin pituudesta L ja kuvantamiskorkeudesta R (2). Jotta SLAR-tekniikalla saatujen kuvien lentosuuntainen resoluutio saataisiin riittävälle tarkkuudelle, täytyy joko kuvantamisetäisyyttä pienentää tai antennin kokoa kasvattaa yhtälön (2) mukaisesti. Käytettäessä kuvantamiseen lentokoneita, resoluutioksi saadaan vielä tyydyttäviä arvoja, mutta satelliittien tapauksessa antennin koon kasvattaminen tarvittaviin mittasuhteisiin ei ole käytännössä mahdollista. Antennin koosta johtuva rajoite on myöhemmin saatu ratkaistua keinotekoisesti simuloimalla tarvittavan kokoinen antenni. Tutkateknologioista, joissa käytetään hyväksi synteettisesti luotuja suuria antenneja, käytetään nimitystä Synthetic Aperture Radar (SAR). (Meyer 2019, 22; Woolfson 2011, 117–125.)

Kuvioita 1 ja 2 tarkastelemalla lukija voi huomata, että näissä on havainnollistettu yksinkertaistettuja tapauksia, joissa maan kaarevuutta tai tulokulman vaihtelua tutkan kuvantamalla alueella ei ole esitetty. Todellisuudessa tulokulma kasvaa etäisyyden kasvaessa ja etäisyysuuntainen resoluutio paranee tutkan kuvantaman alueen kauimmaista reunaa kohti. Tämän lisäksi pinnanmuotojen paikalliset vaihtelut vaikuttavat tulokulmaan ja edelleen tutkan pikselikohtaiseen resoluutioon, kuten Van Zyl ja Kim (2011, 8) kirjassaan huomauttavat. Satelliittien tapauksessa maan kaarevuus ja topologiset vaihtelut täytyy ottaa huomioon. (Van Zyl & Kim 2011, 5–8; Range Doppler Terrain Correction Operator n.d.)

3.3 SAR-kuvat

3.3.1 Säteilyn takaisinsironta ja kuvan muodostus

Kun tutkan signaali kohtaa kappaleen, säteily voi heijastua takaisin, hajaantua useampaan eri suuntaan tai absorboitua kohteeseen. Osuudesta, joka heijastuu kappaleesta suoraan takaisin tutkan vastaanottimeen, käytetään nimitystä *takaisinsironta* (ASAR Product Handbook 2007, 394). Takaisinsironnan määrä tietyllä aallonpituudella riippuu useista tekijöistä, kuten kappaleen koosta, pinnanmuodosta ja kappaleen materiaalista (ASAR Product Handbook 2007, 19). SAR-kuva muodostetaan vertailemalla lähetetyn signaalin intensiteettiä vastaanotetun takaisinsironnan intensiteettiin (Meyer 2019, 25). SAR-kuvassa yksittäisen pikselin arvo edustaa takaisinsironnan määrää kyseiseltä alueelta siten, että vaaleat pikselit vastaavat suurta takaisinsirontaa ja tummat vähäistä (ASAR Product Handbook 2007, 19). Lähetettävän signaalin intensiteetti pyritään saamaan mahdollisimman suureksi, jotta muodostettavasta kuvasta voitaisiin paremmin erottaa erilaisia säteilyä heijastavia kappaleita (Van Zyl & Kim 2011, 8).

Tutkasignaalin vuorovaikutuksen tyypit kappaleiden kanssa voidaan jakaa muutamaankin eri päätyyppiin kappaleen pinnan ja niissä tapahtuvien heijastumisien mukaan. Podest (2018) jakaa takaisinsironnan mekanismit neljään eri luokkaan, jotka ovat tasaiset pinnat (eng. *smooth surface*), kaksoisheijastumiset (eng. *double bounce*), karkeat pinnat (eng. *rough surface*) ja massaheijastumiset (eng. *volume scattering*). Tasaiset pinnat toimivat lähes peilien tavoin, joista tutkasignaali heijastuu tutkan sensorista pois päin ja alue näyttää kuvassa mustalta. Kaksoisheijastumisissa tutkan signaali heijastuu kahden toisiaan kohtisuorassa olevan tasan pinnan kautta takaisin tutkaan. Tällaisia heijastumisia tapahtuu usein kaupunkialueilla, joissa tutkan signaali heijastuu ensin asfaltista rakennuksen seinän kautta takaisin tutkaan. Vastaavia heijastumisia voi tapahtua myös metsien reunoilla ja tulva-alueilla, joissa esiintyy toisiinsa nähden kohtisuorassa olevia pintoja. Kaksoisheijastumisissa takaisinsironnan määrä on suurta, minkä vuoksi tällaiset alueet näkyvät tutkakuvissa kirkkaina. (Podest 2018; Meyer 2019, 28.)

Karkeiksi pinnoiksi luokitellaan pinnat, joissa erilaiset epätasaisuudet saavat signaalin hajaantumaan. Pinnan näyttäytyminen karkeana riippuu siinä esiintyvien korkeuserojen ja käytettävän aallonpituuden suhteesta. Mitä suurempia pinnan korkeusvaihtelut ovat suhteessa aallonpituuteen, sitä enemmän tapahtuu signaalin hajontaa ja mahdollisuus takaisinsirontaan kohti tutkaa kasvaa.

Massaheijastumisissa signaali kohtaa lukuisia pieniä kappaleita, joista heijastuminen tapahtuu. Massaheijastumista voi tapahtua väliaineessa tai esimerkiksi puiden latvuksissa, missä oksat ja lehdet aiheuttavat signaalin kimpoilua toistensa välillä. (Podest 2018; Meyer 2019, 26-27.)

3.3.2 Geometriset vääristymät

Koska kuvantamiseen käytetyissä tutkateknologioissa säteily kohdistetaan viistosti maanpintaan nähden, esiintyy SAR-tutkakuvissa perspektiivistä johtuvia vääristymiä. Kohtisuoraan otettuihin optisten kameroiden kuviin verrattuna SAR-tutkakuvissa näkyvät korkeat kappaleet voivat näyttää kallistuneilta, tasaisilta tai jättää takanaan olevat kappaleet piiloon. Kuvissa näkyvä vaikutelma riippuu katselukulmasta ja pinnanmuodosta. (Meyer 2019, 24.)

3.3.3 Aallonpituuden vaikutus tutkavasteeseen

Signaalin läpäisevyyteen eri väliaineissa voidaan vaikuttaa muuttamalla käytettävää aallonpituutta. Pidemmät aallonpituudet yleensä läpäisevät väliaineita paremmin, riippuen kohteen tiheydestä, koosta ja fysikaalisista ominaisuuksista. Siten aallonpituutta vaihtelemalla voidaan saada tietoa erilaisista kappaleista sekä materiaalien koostumuksista. Esimerkiksi aallonpituudet 2,4–3,8 cm eivät läpäise metsästä puiden latvuksia, kun taas 15–30 cm aallonpituudet voivat saavuttaa maan pinnan latvusten läpi. Lisäksi lyhyemmillä aallonpituuksilla pinnat näyttävät karkeammilta, sillä takaisinsirontaa tapahtuu enemmän, kun kappaleet ovat suurempia suhteessa aallonpituuteen. Tästä syystä 2,4–3,8 cm aallonpituuksia käyttäviä tutkia sanotaan myös korkean resoluution tutkiksi. SAR-tutkassa yleisimmin käytettävät aallonpituudet sekä niiden pääasialliset sovelluskohteet on koottu taulukkoon 1. (Podest 2018; Meyer 2019, 26–29.)

Taulukko 1. SAR-tutkien aallonpituuksia ja käyttökohteita

| Kaista (band) | Aallonpituus (cm) | Käyttökohteita |
|---------------|-------------------|---|
| X | 2,4–3,8 | Kasvillisuuden, biomassan, kaupunkialueiden ja merien tutkiminen. |
| C | 3,8–7,5 | Muutosten havainnointi, maanviljelyn ja meriliikenteen seuraaminen. |
| S | 7,5–15 | Maanviljelyn seuraaminen. |
| L | 15–30 | Maankuoren, maan kosteuden, biomassan ja kasvillisuuden tutkiminen. |

3.3.4 Dielektrinen vakio

Käytettävän aallonpituuden lisäksi signaalin läpäisyyn väliaineessa vaikuttaa väliaineen dielektrinen vakio, joka kuvaa materiaalin kykyä johtaa energiaa. Pienillä dielektrisen vakion arvoilla energian absorboituminen väliaineeseen on suurempaa, mikä näkyy signaalin vähäisenä takaisinsirontana. Dielektriseen vakioon vaikuttaa väliaineen kosteuspitoisuus, joka vähentää mikroaaltoenergian absorboitumista väliaineeseen ja aiheuttaa voimakkaampaa takaisinsirontaa. Poikkeuksen tähän sääntöön tekevät vesialueet, jotka korkeasta dielektrisen vakion arvosta huolimatta näkyvät tutkakuvissa tummina alueina. Tämä johtuu veden pinnan käyttäytymisestä peilipinnan tavoin heijastaen tutkan signaalin kokonaan pois tutkan vastaanottimesta. Koska väliaineen dielektrinen vakio riippuu sen sisältämästä veden määrästä, voivat ympäristön olosuhteiden muutokset, kuten sateet ja lumen sulaminen näkyä muutoksina satelliittikuvissa. (Podest 2018.)

3.3.5 Polarisaation hyödyntäminen

Polarisaatiolla tarkoitetaan elektromagneettisen aallon värähtelysuuntaa etenemissuuntaansa nähden. Säteily voi kiertyä etenemissuuntansa ympäri tai olla lineaarista, jolloin säteilyn polarisaatio on vakio eikä muutu sen edetessä. (Meyer 2019, 27.) Usein SAR-tutkissa säteilyä voidaan lähettää ja vastaanottaa joko horisontaalisena (H) tai vertikaalisena (V). Näitä yhdistelemällä säteilyn lähettämisen- ja vastaanottamistavat voivat olla HH, HV, VH ja VV, missä ensimmäinen kirjain tarkoittaa lähettämissuuntaa ja jälkimmäinen vastaanotettua. Tutkien polarisaatioista puhuttaessa tutkan tai sen tuottaman datan sanotaan olevan "dual-pol" tai "quad-pol", joilla tarkoitetaan mitattujen polarisaatioiden määrää. *Dual-pol* termillä tarkoitetaan satelliitin mittaavan kahta polarisaatiota ja *quad-pol* termillä neljää. Polarisaatioita mittaaviin SAR-tutkiin viitataan nimityksellä "polarimetric SAR" ja usein lyhyesti nimityksellä "PolSAR". (Pinto 2018; Podest 2018)

Tutkan lähettämän ja vastaanottaman signaalin polarisaatioita tutkimalla voidaan tunnistaa erilaisia materiaaleja ja kappaleita, sillä eri polarisaatioyhdistelmien mitattu intensiteetti riippuu kohteesta. Takaisinsirontan mekanismit polarisaation osalta riippuvat heijastavasta pinnasta. Kovilta ja karkeilta pinnoilta (rough surface scattering), kuten vähäisen kasvillisuuden alueilta ja teiden pinnoilta, yhdistelmien VV ja HH intensiteetti on suurempaa kuin yhdistelmillä HV tai VH. Vastaava ilmiö tapahtuu myös signaalin heijastuessa kaksi kertaa (double bounce scattering) ennen signaalin suuntautumista takaisin tutkaan. Runsaassa kasvillisuudessa (volume scattering) signaalin

heijastuminen on monimutkaisempaa ja heijastumisia tapahtuu paljon. Esimerkiksi puiden latvuk-
sissa signaali heijastuu useampaan kertaan ja sen polarisaatio voi muuttua. Tästä johtuen HV- ja
VH-yhdistelmien mitatut korkeat intensiteetit viittaavat kasvillisuuteen, hiekkaan ja huokosiin
maatyyppeihin. Meyerin (2019, 28) esittämät eri polarisaatioiden takaisinsironnan suhteet on esi-
tetty taulukossa 2. (Meyer 2019, 27–28.)

Taulukko 2. Polarisaatioiden takaisinsironnan suhteellinen määrä.

| Heijastumisen tyyppi | Takaisinsironnan suhteellinen määrä |
|----------------------|---|
| Karkeat pinnat | $ S_{VV} > S_{HH} > S_{HV} $ tai $ S_{VH} $ |
| Kaksoisheijastumiset | $ S_{HH} > S_{VV} > S_{HV} $ tai $ S_{VH} $ |
| Massaheijastumiset | Pääasiassa $ S_{HV} $ ja $ S_{VH} $ |

3.3.6 Kohina SAR-kuvissa

Käytettävästä tutkateknologiasta riippumatta SAR-tutkakuvissa esiintyy rakeisuutena, kirkkaina ja
tummina pikseleinä ilmenevää kohinaa. Kohina aiheutuu resoluutiosolun sisällä tapahtuneista lu-
kuisista mikroaaltojen heijastumisista, heijastumisten hajonnasta ja niiden yhteisvaikutuksesta.
Lopullinen tutkan vastaanottama intensiteetti yksittäisestä resoluutiosolusta on näiden satunnais-
ten heijastumisten summa. Kun heijastumisten yhteisvaikutus on konstruktivista, havaitaan takai-
sinsironta voimakkaampana ja SAR-kuvissa tämä näkyy kirkkaana pikselinä. Hajottavalla yhteisvai-
kutuksella on puolestaan intensiteettiä vähentävä vaikutus ja kuvan pikseli nähdään tummana.
Myös täysin homogeenisten alueiden, kuten pellot ja ruohikot, kohdilla havaitaan kohinaa, joka voi
huomattavasti vaihdella vierekkäistenkin pikseleiden välillä. (Podest 2018; Meyer 2019, 24.)

Kuvissa esiintyvää kohinaa yritetään pääasiassa vähentää kahdella tavalla. Ensimmäinen näistä on
kuvien Multi-Look-prosessointi (myös multilook-prosessointi ja lyhyesti multilooking), missä tutkan
signaali jaetaan useampaan kaistaleeseen ja tutkavasteista erotetaan Doppler-ilmion perusteella
toisistaan riippumattomia intensiteettejä suhteessa samaan pikseliin. Kun kaistaleiden pikseliarvot
yhdistetään ja näistä lasketaan keskiarvo, saadaan yksittäisen resoluutiosolun keskimääräinen ta-
kaisinsironta ja kohinanmäärää pikselissä vähennettyä. Multi-Look-prosessointi saatetaan tehdä
myös tulkitsemisen helpottamiseksi, sillä prosessissa pikseleiden koko voidaan muuttaa

suorakulmaisista lähes neliöiksi. Multi-Look-prosessin kääntöpuolena on kuitenkin resoluution heikentyminen. Arvojen lukumäärään, joista keskiarvo on laskettu yksittäiselle pikselille, viitataan nimityksillä ”number of looks” tai lyhyemmin merkinnällä ”N-look”, missä N on keskiarvon laskemiseen käytettyjen arvojen lukumäärä. Tällä tavalla käsittelemättömästä kuvasta voidaan puhua termillä *single look* ja merkitä lyhyesti 1-look. Keskiarvoistetuista kuvista puhutaan usein lyhyesti termillä *multilook*. (Argentini, Lapini, Bianchi & Alparone 2013, 9; Podest 2018; Van Zyl & Kim 2011, 20–22; Multilook Operator n.d.)

Toinen tapa vähentää kohinan määrää kuvissa on käsitellä ne jälkikäteen erilaisilla kohinanpoistomenetelmillä. Osa menetelmistä on yksinkertaisia kuvapikseleiden arvoja keskiarvoistavia suodatimia ja osa pohjautuu kohinan todennäköisyyden mallintamiseen. Sekä Argentini ja muut (2013, 9) että Meyer (2019, 24) ovat kertoneet kohinan todennäköisyyttä mallinnettavan yhtälöllä (4), kun heijastuksen aiheuttavien kappaleiden lukumäärä on suuri ja kokonaisintensiteetin reaali- ja imaginaariosia voi pitää normaalijakautuneina. Argentini ja muut (2013) mukaan yhtälön (4) voi myös esittää yhtälön (5) muodossa, jota sanotaan *multiplikaatiiviseksi* malliksi. Kohinan mallintamisesta yhtälön (4) muodosta Meyer on lisännyt, että esitys on pätevä keskiresoluution SAR-kuvien homogeenisille alueille. Yhtälössä (4) I on pikselin intensiteetti ja σ^0 kuvaa takaisinsironnan osuutta lähetetystä signaalista. Tarkemmin erilaisia kohinanpoistomenetelmiä esitellään luvussa 6. (Argentini ym. 2013, 9; Meyer 2019, 24–25.)

$$(4) \quad pdf(I|\sigma^0) = \frac{1}{\sigma^0} e^{-\frac{I}{\sigma^0}}$$

$$(5) \quad I = \sigma\mu$$

Edellä esitettyjen keinojen lisäksi kohinaa on mahdollista vähentää yhdistämällä useampia eri aikoina kuvannettuja SAR-kuvia. Tähän menetelmään viitataan nimityksellä *Multi-Temporal Speckle Filtering*. (Multi-temporal Speckle Filter Operator, n.d.)

4 Sentinel-1-satelliitit ja niiden SAR-data

4.1 Sentinel-1-satelliitit

Sentinel-1-satelliitit ovat osa ESA:n ja EU:n yhteistä Copernicus-ohjelmaa, jonka tarkoituksena on tarjota ajantasaista ja laadukasta satelliittidataa eri käyttäjäryhmille, sovelluksille ja tutkimuskäyttöön. Copernicus-ohjelmaan kuuluu kuusi Sentinel-satelliittiperhettä, jotka keräävät tietoa eri osalualueista ja joista Sentinel-1-satelliitit ovat sarjan ensimmäisiä. Sentinel-1 satelliittiryhmä koostuu kahdesta samanlaisesta satelliitista, Sentinel-1A ja Sentinel-1B, joilla havainnoidaan maan pinnalla tapahtuvia muutoksia oikealle suunnatun C-aallonpituusalueen SAR-tutkan avulla (Instrument Payload n.d.). Sentinel-1-satelliittien C-SAR-tutka mittaa kahta polarisaatiota, jotka riippuvat käytetystä moodista (Instrument Payload n.d.). (Sentinel-1: Radar mission 2014, Sentinel-1: Overview n.d., Copernicus Sentinel-1 Infographic 2021)

4.2 Sentinel-1-satelliittien SAR-data

Sentinel-1-satelliiteissa käytössä oleva tutka voi toimia neljässä eri moodissa, joista Inteferometric Wide Swath (IW)-moodia käytetään oletuksena maan pinnan kuvantamiseen. Muut moodit ovat Stripmap (SM), Extra Wide Swath (EW) ja Wave Mode (WV). Näistä SM on jatkumoa edellisillä ERS- ja Envisat-avaruuslennoilla kerätylle datalle ja EW on tarkoitettu meri- ja napa-alueiden, sekä jään kuvantamiseen. WV-moodilla voidaan puolestaan tutkia aaltojen suuntaa ja korkeutta merialueilla. Näissä moodeissa kerättyä dataa on saatavilla eri muodoissa ja eri tavoin prosessoituina. (Instrument Payload n.d.)

Moodien data on jaettu kolmeen eri luokkaan, Level-0, Level-1 ja Level-2, datan käsittelyasteen ja käyttötarkoituksen mukaan. Level-0-tason data on raakadataa, joka tulee käsitellä ennen käyttöä (Level-0 n.d.). Level-1-data on osittain esikäsiteltyä ja tämän tason tuotteita voi moodista riippuen olla saatavilla yhtä tai kahta versiota (Data Products n.d.). SW, IW ja EW-moodeille 1-tason datasta on saatavilla erikseen Single Look Complex (SLC) ja Ground Range Detected (GRD) -dataa (Data Products n.d.). Level-2-tason data on 1-tason datasta johdettua tietoa, joka voi sisältää tietoa muun muassa merialueiden tuulikentistä ja muutosnopeudesta (Level-2 n.d.)

Level-1-tason SLC-datassa etäisyydet ovat viistoetäisyys suunnan mukaisia ja yksittäisissä kuvapikseleissä on tieto aallon amplitudista ja faasista kompleksilukuina. Koska suurempi alue usein kuvannetaan paloittain ohuempina kaistaleina, sisältää SLC-data useampia kuvia, jotka käsittävät tiedon yksittäisestä kaistaleesta yhdellä polarisaatioilla. Meyerin (2019) mukaan SLC-datan pääasiallinen käyttötarkoitus on interferometria, jolla tutkitaan maan pinnan muotoja ja muodonmuutoksia. SAR-kuvilla tehtävään interferometriaan viitataan usein lyhenteellä InSAR. (Braun & Veci 2021, 2; Meyer 2019, 30–33; Level-1 SLC Products n.d.)

GRD-data on kuvannetuista kaistaleista koottua, etäisyysuuntaan muutettua ja multilook-prosesoitua dataa. GRD-kuvien kuvapikselit edustavat takaisinsironnan signaalin voimakkuutta yksittäisenä arvona, eikä tietoa aallon faasista enää ole datassa mukana. Multilook-prosessoinnin vuoksi kuvissa esiintyvää kohinaa on saatu vähennettyä ja kuvapikselit ovat lähes neliöitä, mutta resoluutio on heikompi SLC-dataan verrattuna. Koska signaalin tulokulma vaihtelee tutkan kuvantamalla alueella, vaihtelevat myös resoluutio, ENL ja tulokulma kaistaleiden välillä. Korkean resoluution IW GRD-datassa spatiaalinen resoluutio on n. 20,3–20,4x22,5–22,6 metriä, ENL 4,3–4,4 ja pikselin koko 10x10 metriä. GRD-dataa käytetään usein visualisointeihin, kartoitukseen ja muutosten tutkimiseen. (Level-1 GRD Products n.d.; IW GRD Resolutions n.d.) Lähes kaikkien moodien dataa eri tasoilla on saatavilla Copernicus Open Access Hubissa.

4.3 SAR-datan esikäsittely

SAR-datan esikäsittelyllä pyritään korjaamaan luvussa 3 esitettyjä vääristymiä, parantamaan datan laatua ja tekemään datasta käsittelykelpoista jatkokäyttöön. Kaikki käsittelyketjun vaiheet eivät ole pakollisia ja niiden tarpeellisuus riippuu datasta ja sen jatkokäytöstä. Näiden vaiheiden lisäksi prosessointiketjuun voi tulla lisävaiheita riippuen siitä, käsitelläänkö yhtä kuvaa, vai esimerkiksi pintaanko useampia kuvia yhteen lisäinformaation saamiseksi. Kuviossa 3 on esitetty yksittäiselle GRD-kuvulle useimmiten tehtävät esikäsittelytoimet. (Zohary n.d; Meyer 2019, 46–48; Braun 2020, 2–13.)



Kuvio 3. SAR-datan esikäsittelyn vaiheet

Lentoratatietojen korjaus (orbit file)

Satelliitin kulkema lentorata on etukäteen laskettu, mutta suurten taivaankappaleiden vetovoima, maan epätasainen gravitaatiokenttä ja aurinkotuuli voivat muuttaa satelliitin lentorataa. Tämän vuoksi satelliitit keräävät tietoja niiden todellisesta lentoradasta, joiden perusteella satelliitin tarkka sijainti kuvantamishetkellä saadaan selville. Lentoratatietojen käyttäminen ei ole välttämätöntä, mutta niiden avulla satelliittikuvien maantieteellinen tarkkuus saadaan mahdollisimman hyväksi. Lentoratatietojen korjaus tehdään lisäämällä ja soveltamalla satelliittidataan Orbit-tiedosto, joka on saatavilla pari viikkoa kuvantamisen jälkeen. (Meyer 2019, 46; Zohary n.d.)

Kuvan rajaaminen (subsetting)

Yksittäiset satelliittikuvat ovat tiedostoina suuria, joiden käsittely ilman kuvien rajaamista olisi raskasta. Tästä syystä satelliittikuvista usein rajataan pienempi otos, johon loput esikäsittelytoimet tehdään. (McVittie, 2019.)

Anturilähtöisen kohinan poistaminen (thermal noise reduction)

Aaltoliikkeiden yhteisvaikutuksesta johtuvan kohinan lisäksi SAR-kuvissa voi esiintyä vastaanottimesta johtuvaa additiivista kohinaa, jonka saattaa havaita kuvista alueilla, joilla takaisinsironnan määrä on usein vähäistä. Tällaisia alueita ovat esimerkiksi tasaiset pinnat, kuten tyyni veden pinta tai asfaltoitu tie. Anturilähtöisen kohinan määrä muuhun kohinaan verrattuna on vähäistä, eikä sen poistamista ohjeisteta kaikissa Sentinel-1 GRD -kuvien käsittelyä koskevissa oppaissa. Thermal Denoising of Products Generated by the S-1 IPF -dokumentaation mukaan anturilähtöinen kohina

tulee merkitykselliseksi vain kun signaalien keskiarvo on pieni, minkä vuoksi vaiheen tarpeellisuus riippuu käytettävästä datasta. (Gomez 2020; Thermal Denoising of Products Generated by the S-1 IPF 2017.)

Datan kalibrointi (calibration)

SAR-datan kalibroinnissa kuvan digitaaliset pikseliarvot muutetaan kalibroiduksi takaisinsironnaksi, joka vastaa normalisoitua tutkan poikkileikkausta. Kalibrointiprosessi huomioi lisäksi sensorikohtaiset ominaisuudet ja signaalin tulokulman, minkä vuoksi kalibroidut tutkakuvat ovat vertailukelpoisia keskenään. Datan kalibrointi ei ole välttämätöntä, jos kuville tehdään laadullista analyysiä, mutta silloin kun kuvia halutaan vertailla keskenään tai niille halutaan tehdä määrällistä analyysiä, kalibrointi on tehtävä. (Braun 2020, 7; Zohary n.d.; Calibration Operator n.d.)

Multilooking (multilooking)

Multilooking on luvussa 3.3.6 esitetty, ei-välttämätön menetelmä SAR-kuvissa esiintyvän kohinan poistamiseksi ja kuvien tulkitsemisen helpottamiseksi. Datan käsittelyssä käyttäjän tulee valita osuuksien määrä, joiden mukaan multilooking tehdään.

Kohinan poistaminen (speckle filtering)

Kohinan poistamisessa kuvaa käsitellään vaihtoehtoisilla kohinanpoistosuodattimilla, joiden tarkoituksena on vähentää kuvissa näkyvää rakeisuutta. Käytettävä kohinanpoistosuodatin voidaan valita suodattimien ominaisuuksien ja kuvan jatkokäyttötarkoituksen mukaan. Eri kohinanpoistomenetelmiä käsitellään luvussa 6.

Geometrinen vääristymien poistaminen (terrain correction)

Tässä vaiheessa kuva muutetaan karttakoordinaattijärjestelmään ja tutkan signaalin sivusuuntaisesta suuntaamisesta aiheutuvat geometriset vääristymät, kuten varjostumat, päällekkäisyys ja muotojen vääristymiset korjataan mahdollisimman lähelle oikeaa. Muunnokseen tarvitaan korkeusmalli (eng. digital elevation model, DEM), jonka perusteella oikea geometria voidaan laskea.

Korkeusmalleista on useampia eri versioita, joista tulee valita kyseisille leveysasteille sopiva malli. Korkeusmalleista GETASSE30 on neljän eri korkeusmallin yhdistelmä ja soveltuu käytettäväksi myös pohjoisella pallonpuoliskolla. (Range Doppler Terrain Correction Operator n.d.; Zohary n.d..)

Intensiteetin logaritmuunnos (Linear to dB)

Viimeisenä vaiheena SAR-datan esikäsittelyssä tehdään logaritmuunnos, jolla datan intensiteetti muunnetaan logaritmiseen asteikkoon desibeleiksi. Intensiteettiarvot ilman logaritmuunnosta voivat olla hyvin lähellä toisiaan, jolloin intensiteettieroja on vaikea havaita SAR-kuvasta. Logaritmuunnos skaalaa intensiteettiarvot normaalijakauman suuntaan, minkä ansiosta kuvan kontrastia saadaan paremmaksi ja kuvaa on helpompaa tulkita. Tämä vaihe ei ole välttämätön, mutta voi tuoda kuvista esiin niissä aikaisemmin heikosti näkyviä piirteitä. (Braun 2020, 13; Meyer 2019, 48.)

4.4 Polarimetrisen SAR-datan käsittely

PolSAR-datan käsittely noudattaa osittain samoja vaiheita kuin edellä esitetty GRD-datan käsittely. Jotta PolSAR-datan tieto voitaisiin täysin hyödyntää, käytetään prosessoinnissa kompleksista SLC-dataa (Veci 2019), ja siten käsittelyvaiheissa voi esiintyä variaatioita. Ohjeita eri polarisaatioita sisältävän SLC-datan käsittelyyn löytyy huomattavasti vähemmän ja näiden sisältö vaihtelee, johon tuen mahdollisesti juuri vaihtoehtojen suuremmasta määrästä datan käsittelyvaiheissa.

ESA:n polarimetrisen SAR-datan käsittelyä koskevassa oppaassa GRD-datan käsittelyn tapaan ohjeistetaan ensin rajaamaan kuvasta aluksi pienempi osa-alue. Rajaamisen jälkeen data kalibroidaan ja datasta muodostetaan kovarianssi- tai koherenssimatriisi jatkokäsittelyä varten. Luotua matriisia tarvitaan muun muassa polarimetrisen datan kohinan suodattamisessa, missä kohinanpoistomenetelmät ottavat syötteenä matriisin. Erityisesti PolSAR-datan suodatukseen kehitetyt kohinanpoistosuodattimet huomioivat eri polarisaatiot ja säilyttävät tiedon amplitudista ja faasista. Sentinel Toolboxista löytyvät kohinanpoistosuodattimet polarimetrisen SAR-datan käsittelyyn ovat Boxcar, Improved Lee Sigma, Refined Lee ja Intensity Driven Adaptive Neighbourhood (IDAN). (Veci 2019.) ESA:n oppaassa listattujen vaiheiden lisäksi polarimetrisen SLC-datan käsittelyyn näkee ohjeistettavan myös muita GRD-datan käsittelystä tuttuja vaiheita.

5 Konvoluutioneuroverkot ja neuroverkkojen kouluttaminen

5.1 Neuroverkot

Neuroverkot ovat laskennallisia malleja, joissa yksinkertaisia laskentayksiköitä on yhdistetty toisiinsa verkkomaiseksi rakenteeksi. Yhdistämällä lukuisia laskentayksiköitä saadaan mallinnettua vaativia ja vaikeasti määriteltäviä ongelmia käyttämättä monimutkaisia algoritmeja. Yksinkertaisuuden lisäksi neuroverkkojen etu perinteisiin algoritmeihin verrattuna on niiden kyky mukautua erilaisiin ongelmiin. (Gaupe 2007, 1–3; Kelleher 2019, luku 3.)

5.1.1 Neuroverkkojen yleinen rakenne

Neuroverkot kuvataan usein kerroksiin järjestettyinä, joista ensimmäinen on *sisääntulokerros*, keskimmaisistä käytetään nimitystä *piilotettu kerros* ja viimeinen on *ulostulokerros*. Silloin kun neuroverkossa on enemmän kuin yksi piilotettu kerros, neuroverkkoa sanotaan *syväksi neuroverkoksi*. Sisääntulokerroksessa on tieto syötteiden muistipaikoista, jotka annetaan syötteinä ensimmäisen piilotetun kerroksen neuroneille. Piilotettujen kerrosten neuronit suorittavat laskentaa annetuista syötteistä ja laskennan tulos annetaan syötteenä seuraavan kerroksen neuroneille. Neuroverkon viimeisen, eli ulostulokerroksen arvot kuvaavat lopullista neuroverkon laskemaa päätöstä annetuista syötteistä. (Kelleher 2019, luku 3.)

5.1.2 Neuron

Neuron on yleisnimitys neuroverkkojen yksittäiselle laskentayksikölle. Neuronin tehtävä neuroverkossa on prosessoida syötteenä annettuja tietoja ja joko aktivoitua tai olla aktivoitumatta niiden perusteella. Käytännössä neuronille syötetään reaalitylukuarvoja, jotka summataan yhteen ja tulkinta tehdään summan perusteella. Koska osa annetuista syötteistä voi olla merkittävämpiä kuin toiset, jokaisella syötteellä on painoarvo, ja tulos on summa painokertoimen ja syötteen tuloista. Matemaattisesti tämä voidaan esittää lausekkeen (6) muodossa, missä lukuarvot x_i ovat joko sisääntulokerroksen arvoja tai edellisen kerroksen neuroneiden ulostuloja, w_i näiden painokertoimet ja z syötteiden painotettu summa. (Mueller & Massaron 2019, luku 7.) Neuroverkon koulutuksessa tavoitteena on löytää optimaaliset painokertoimet w_i , joilla malli kuvaisi parhaiten käsiteltävää ilmiötä (Kelleher 2019, luku 3).

$$(6) \quad z = \sum_{i=1}^n x_i \cdot w_i$$

5.1.3 Aktivaatiofunktiot

Koska reaalimaailman ongelmat ovat luonteeltaan harvoin lineaarisia, tarvitaan laskentaan osa, jolla ei-lineaarista luonnetta voidaan mallintaa. Tämä saadaan toteutettua aktivaatiofunktioiden avulla, jolle painotettu summa z syötetään ja neuronin ulostulona on aktivaatiofunktion arvo painotetusta summasta (7). Käytettävä aktivaatiofunktio valitaan käsiteltävän ongelman luonteen perusteella. Syvässä neuroverkossa eri kerroksilla voidaan käyttää eri aktivaatiofunktioita, mutta saman kerroksen neuroneilla käytetään kuitenkin vain yhtä aktivaatiofunktiota. (Kelleher 2019, luku 3.)

$$(7) \quad \text{ulostulo} = \text{aktivaatiofunktio}(z)$$

5.2 Konvoluutioneuroverkot

Konvoluutioneuroverkot, Convolutional Neural Networks (CNN), ovat neuroverkkoja, joissa käytetään konvoluutiokerroksia neuroverkon osana. Konvoluutioilla voidaan tunnistaa datassa esiintyviä korreloivia osia, kuten vierekkäisten pikseleiden arvot kuvassa. Useamman konvoluutiokerroksen käytöllä datasta voidaan erottaa monimutkaisempia piirteitä, ensin yksinkertaisia muotoja, seuraavaksi näistä muodostuvia rakenteita ja lopulta suurempia kokonaisuuksia. Esimerkkinä tässä tekstissä puhutaan kuvista asiayhteyden vuoksi, mutta konvoluutioita voidaan käyttää myös rakenteiden tunnistamiseen esimerkiksi signaaleissa ja videodatassa. (Venkatesan & Li 2017, 89.)

Konvoluutiokerrokset sijoitetaan neuroverkon alkuun tunnistamaan syötteestä rakenteita, minkä jälkeen neuroverkossa käytetään täysin yhdistettyä kerroksia luokittelu- tai regressio-ongelman ratkaisemiseen (Venkatesan & Li 2017, 97). Tavallisesti konvoluutioneuroverkoissa käytetään muutamia toisiaan seuraavia konvoluutiokerroksia, joiden välillä voi olla tietoa tiivistäviä ja ylikouluttamisen riskiä vähentäviä pooling-kerroksia (Géron 2017, 363). Konvoluutioneuroverkkojen rakenne voi kuitenkin vaihdella ja kerrokset voidaan jakaa eri haaroihin, joista tieto lopulta yhdistetään halutun lopputuloksen aikaansaamiseksi. Kuvien tapauksessa tämä voi esimerkiksi tarkoittaa kuvan eri osien käsittelemistä eri haaroissa sokean pisteen (eng. blind-spot) aikaan saamiseksi tai kuvan rakenteiden tarkastelua eri mittakaavoissa (Laine, Karras, Lehtinen & Aila 2019, 2). (Géron 2017,

363, 369.) Muita käyttötarkoituksia voi esimerkiksi olla datan yhdistäminen useammasta lähteestä, kuten Gaetano, Ienco, Ose ja Cresson (2018) ovat maankäytön luokittelua koskevassa tutkimuksessaan tehneet (Gaetano ym. 2018, 4).

Konvoluutiot

Konvoluution täsmällinen määritelmä kuvataan kahden funktion $f(x)$ ja $g(x)$, $f, g: \mathbb{R} \rightarrow \mathbb{C}$ välisenä operaationa, jonka tuloksena on funktioista muodostettu integraali $(f * g)$. Määritelmä (8) kattaa konvoluutiot n -ulottuvuuksille, mutta käytännönsovelluksissa konvoluutioita tehdään yksi-, kaksi- ja kolmiulotteisissa avaruuksissa. Useimmiten yksi-, kaksi- ja kolmiulotteisten avaruuksien konvoluutiot esitetään matriisien avulla ja summasarjoina silloin, kun on kyse diskreeteistä funktioista. (Tyni 2019; Varjo 2015).

$$(8) \quad (f * g)(x) := \int_{\mathbb{R}^n} f(x)g(x - y) dy$$

Yleinen käyttö konvoluutioille on ollut niiden soveltaminen syötteen muokkaamiseen, kuten kohinan poistoon signaalinkäsittelyssä tai kuvankäsittelyssä kuvan sumentamiseen tai terävöittämiseen (Varjo 2015, Siltanen 2018). Konvoluutioilla on myös mahdollista tunnistaa annetusta syöteestä rakenteita ja malleja, minkä vuoksi konvoluutioneuroverkkoja on enenevässä määrin alettu hyödyntää muiden muassa anomalioiden, puheen ja kuvien tunnistuksessa. (Yalçın 2021, luku 7.)

Muotojen tunnistaminen valokuvassa tehdään tarkastelemalla kuvasta pientä osa-aluetta, jonka yksittäiset pikseliarvot antavat viitettä siitä, onko kohdassa jokin muoto tai muu tunnistettava rakenne (Venkatesan & Li 2017, 89–90). Sama tarkastelu voidaan toistaa kuvan eri kohtiin, eri kville, ja mahdollisesti havaita samanlainen toistuva rakenne tarkasteltavissa kohdissa. Silmämääräisesti tarkastelemalla tämä voidaan havaita samaa suuruusluokkaa olevina arvoina, mutta sama tarkastelu voidaan toteuttaa myös laskennallisesti. Jos jonkin tietyn muodon esiintymistä kuvassa halutaan tutkia, voidaan tarkasteltava muoto kuvata matriisiin ja verrata sitä kuvaan. Laskennallisesti vertailu tehdään konvoluutioilla, joissa valokuvan pieni osa-alue ($a_{m \times n}$ -matriisi) käsitellään $b_{k \times k}$ -matriisilla (Venkatesan & Li 2017, 89–90). Kerrottavaa matriisia kutsutaan syötteeksi tai sisääntulomatriisiksi, ja kertojaa kerneliksi, filteriksi tai suodattimeksi (Korpihalkola 2021; Varjo

2015). Konvoluutiontulosta, ulostulomatriisia saatetaan kutsua myös nimellä *feature map* (Venkatesan & Li 2017, 92).

Valokuvien tapauksessa on kyse kaksi- tai kolmeulotteisista konvoluutioista riippuen valokuvan värikanavien määrästä. Mustavalkoisessa valokuvassa värikanavia on yksi ja tehtävät konvoluutiot yksinkertaistuvat 2-ulotteisiin konvoluutioihin. Useamman värikanavan tapauksessa, esimerkiksi RGB-kuvissa, konvoluutiot ovat kolmiulotteisia. Tällöin konvoluutio voidaan toteuttaa kolmiulotteisella kernel-ikkunalla tai jakamalla eri kanavien konvoluutiot omiin haaroihin, joissa jokaisessa käytetään omaa kerneliä. (Dumoulin & Visin 2018, 6–8.)

5.3 Blind-spot-tekniikka

Konvoluutioneuroverkoissa blind-spot-tekniikalla tarkoitetaan konvoluutiota, jossa yksi tai useampi pikseli halutaan jättää pois laskutoimituksista. Tavoiteltu lopputulos on mahdollista toteuttaa useammilla eri tavoilla ja viime vuosien kohinanpoistomenetelmissä sokeapistete on muun muassa saatu aikaiseksi korvaamalla kernel-ikkunan keskimäinen pikseli satunnaisella arvolla tai antamalla $n * m$ -muotoisen kernel-ikkunan käsitellä neljä eri rotaatiota samasta $2n + 1$ -kokoisesta kuvasta (Laine ym. 2019; Krull 2019). Tilanteen mukaan voi olla tarkoituksenmukaista jättää laskutoimituksista pois useampia pikseleitä, mikä voidaan toteuttaa siirtämällä kernel-ikkunaa (Molini, Valsesia, Fracastoro & Magli 2022). Useimmiten blind-spot-tekniikoiden vaihtelut liittyvät laskennan tehokkuuteen, poisjätettäviin pikseleihin tai käsiteltävien pikseleiden järjestykseen. Esimerkiksi myöhemmin esiteltävän Laine ja muut (2019) kohinanpoistomenetelmässä käytettävälle blind-spot-tekniikalle on esitetty korvaava vaihtoehto, jossa samoja pikseleitä ei käsitellä useampaan kertaan (Honzátko, Bigdeli, Türetken & Dunbar 2020).

5.4 Koulutus- ja testidata

Koneoppimismallin koulutuksessa data jaetaan koulutus- ja testidataan, joista testidata on mallin hyvyyden arviointia varten. Koneoppimismalli on mahdollista ylikouluttaa käytettävissä olevaan dataan, jolloin malli on oppinut ennustamaan koulutukseen käytettävät arvot erinomaisesti tai jopa täydellisesti. Todellisuudessa ollaan kuitenkin kiinnostuneita siitä, miten hyvin malli yleistyy ja millaisia ennusteita mallilla saadaan ei-koulutukseen käytetyille arvoille. Tähän arviointiin

tarvitaan testidataa. Useimmiten koulutusdatan ja validointiin jätetyn testidatan suhde on 80 % opetukseen ja 20 % testaukseen. (Géron 2017, 29.)

Mallin koulutuksen onnistumisessa koulutukseen käytetyllä datalla on iso merkitys. Liian vähäisellä datamäärällä malli ei opi tunnistamaan käsiteltävää ongelmaa ja tuloksena mallin ennusteet eivät osu oikeaan. Jotta mallilla saadaan tarkkoja ennusteita ja malli saadaan yleistymään myös koulutusdatan ulkopuolelle, dataa täytyy olla paljon, sen tulee sisältää ilmiötä ennustavia piirteitä ja olla ilmiötä kuvaava. Datan kuvaavuudella tarkoitetaan tilannetta, jossa koulutusdata ei sisällä tapauksia, joita mallilla halutaan ennustaa. Näiden edellytysten lisäksi on tärkeää, että data on luotettavaa. Jos datassa on paljon virheitä, mallin on vaikea oppia ennustamaan haluttua asiaa. (Géron 2017, 22–25.)

6 Kohinanpoistomenetelmät

6.1 Kohinan poistamisen tavoitteet, menetelmien jaottelua ja ongelmanratkaisutapoja

Kohinan poistamisessa kuvista pidetään tärkeänä rakeisuuden suodatusta säilyttäen samalla kuvien yksityiskohdat, reunat ja pintojen rakenne. SAR-kuvien tapauksessa tämän lisäksi on tärkeää, että käytettävä filtti säilyttää avaruudellisen ja radiometrisen resoluution, sekä tilastolliset ominaisuudet kuvissa. Filttierin ei tulisi myöskään vääristää kappaleiden muotoa tai hävittää pieniä yksityiskohtia. (Yommy, Liu & Wu 2015, 260; Lee, Jurkevich, Dewaele & Wambacq 1994, 320; Lee, Aisworth & Wang 2017, 5303.)

SAR-kuvien kohinan poistamiseen kehitetyt filttierit voidaan jakaa useammalla eri tavalla kategoriioihin niissä käytettyjen menetelmien mukaan. Yksi yleinen jako filttierien välillä on erottelu mukautuviin (adaptive) ja ei-mukautuviin (non-adaptive) filttierihin. Adaptiivisilla filttierillä tarkoitetaan kohinanpoistomenetelmiä, joissa kohinan suodatus tehdään kuvasta laskettujen tilastollisten tunnuslukujen perusteella. Ei-mukautuvat filttierit ovat siten suodattimia, jotka eivät ota huomioon kuva- tai aluekohtaista vaihtelua ja joissa suodatus tehdään huomioimatta tilastollista vaihtelua. (Lee 1981.) Toinen filttierien toimintaa kuvaava jaottelu on jako lokaaleihin ja ei-lokaaleihin filttierihin. Lokaalit filttierit laskevat suodatuksessa käytetyt tunnusluvut yhden filttieröinti-ikkunan sisältä ja kohinan suodattaminen tehdään näiden tunnuslukujen perusteella filttier-ikkunan sisältä mälle alueelle. Ei-lokaalit filttierit sen sijaan laskevat suodatukseen käytettävät tunnusluvut useammista samanlaisista alueista ja näitä arvoja keskiarvoistetaan tietyillä painoarvoilla. (Lee ym. 2017, 5303; Argenti, Lapini, Alparone & Bianchi 2013, 19; Molini ym. 2022.)

Yksittäisten menetelmien käyttämään ongelmanratkaisutapaan saatetaan viitata sanoilla "spatial", "transform" ja "wavelet". Avaruudellisella (eng. spatial) lähestymistavalla tarkoitetaan menetelmän poistavan kohinaa kuvan pikseleiden sijainnin perusteella. Muunnoksella (eng. transform) viitataan menetelmässä tehtyyn kohinan mallintamisen muunnokseen ja "wavelet"-termillä kohinanpoisto-ongelman lähestymistä aaltojen tutkimuksen kautta. Viimeisenä nostettakoon vielä esille erittely syväoppimismenetelmiin, jotka edustavat alan viimeisintä suuntausta. Edellä mainitut jaottelut eivät välttämättä ole toisiaan poissulkevia ja menevät osittain päällekkäin. Esimerkiksi Lee-,

Frost-, Kuan- ja Refined Lee -filtterit ovat lokaaleja adaptiivisia filttäreitä. (Argenti ym. 2013, 19; Molini ym. 2022.)

Monet kohinanpoistomenetelmien ensimmäisistä ratkaisuista ovat lokaaleja spatiaalisia menetelmiä ja myöhemmin katseiden suunnattua erityisesti SAR-kuviin, näille on pyritty löytämään korvaajia muiden lähestymistapojen kautta. Koska menetelmien ja näiden taustalla olevan teorian kirjo on hyvin laaja, keskitytään seuraavissa luvuissa esittelemään ja tekemään vertailua vain tunnetuimpien perinteisten menetelmien ja syväoppimismenetelmien avulla kehitettyjen kohinanpoistomenetelmien välillä.

6.2 Kohinanpoistomenetelmien vertailuun käytettyjä metriikoita ja kaavoissa esiintyviä käsitteitä

Kuvien laatua kuvaavat metriikat jaotellaan usein objektiivisiin ja subjektiivisiin metodeihin. Objektiiivisilla metodeilla tarkoitetaan numeerisia, kuvista laskettuja tunnuslukuja ja subjektiivisilla ihmisen havainnointikykyyn perustuvaa arviointia. (Horé & Ziou 2010, 2366). Kuten SAR-kuvien kohinanpoistomenetelmien kanssa, filttäreiden paremmuutta kuvaavien numeeristen metriikoiden ongelmana on kohinattoman kuvan puuttuminen. Useat menetelmät vaativat kohinattoman kuvan tunnusluvun laskemiseksi, eikä tätä ole SAR-kuvien tapauksessa mahdollista saada (Gomez, Buemi, Jacobo-Berlles & Mejail 2016, 1297). Tästä syystä valittaviin metriikoihin ja metriikoiden tulkintaan tulee kiinnittää etenkin SAR-kuvien tapauksessa huomiota. Kohinanpoistomenetelmien lisääntyessä myös menetelmiä kuvaavien metriikoiden kehitykseen on panostettu ja keksitty menetelmiä, jotka eivät vaadi kohinatonta kuvaa (Gomez ym. 2016, 3). Myös kuvan eri ominaisuuksia, kuten kulmien ja homogeenisten alueiden laatua kuvaavia indeksejä on kehitetty erikseen. Näiden käyttö eri tutkimuksissa on kuitenkin vaihtelevaa sekä tutkimusten iän että tutkimuskohteiden vuoksi, minkä vuoksi eri tutkimuksia voi olla hankalaa vertailla keskenään. Seuraavassa esitellään muutamia yleisesti kohinanpoistomenetelmien vertailuun käytettyjä ja vakiintuneita metriikoita.

Keskineliöpoikkeama (Mean Square Error, MSE)

Keskineliöpoikkeama kertoo havaittujen ja ennustettujen arvojen keskimääräisen etäisyyden toisistaan. Keskineliösumma lasketaan kaavan (9) mukaan, missä m on kuvan korkeus pikseleinä, n

kuvan leveys pikseleinä, x_{ij} alkuperäisen kuvan pikseliarvo kohdassa ij ja \hat{x}_{ij} suodatetun kuvan pikseliarvo kohdassa ij .

$$(9) \quad MSE = \frac{\sum_{j=1}^m \sum_{i=1}^n (x_{ij} - \hat{x}_{ij})^2}{m \cdot n}$$

Peak Signal to Noise Ratio (PSNR)

Peak Signal to Noise Ratio (PSNR) kertoo kuvan suurimman mahdollisen pikseliarvon ja kohinan suhteen (Banerjee & Chaudhuri, 2018). PSNR määritellään keskineliöpoikkeaman avulla ja esitetään usein kaavan (10) muodossa, jossa x_{PEAK}^2 on kuvan datatyyppin mukainen, korkein mahdollinen arvo. PSNR on yleisesti käytetty mittari kuvien laadun mittaamiseen ja on paljon käytetty indeksi myös SAR-kuvien vertailuun. Korkeampi PSNR-arvo viittaa parempaan suodatetun kuvan laatuun ja pieni suureen eroavaisuuteen kuvien kesken (Gomez ym. 2016, 1297; Horé & Ziou 2010, 2366).

$$(10) \quad PSNR = 10 \cdot \log_{10} \left(\frac{x_{PEAK}^2}{MSE} \right)$$

Equivalent Number of Looks (ENL)

Kohinan määrän arviointiin SAR-kuvissa käytetään usein Equivalent Number of Looks (ENL) -indeksiä, joka lasketaan kuvan homogeenisesta kohdasta rajatun alueen keskiarvon \bar{x} ja keskihajonnan s neliöiden suhteen. ENL-indeksiä pidetään usein hyvänä mittarina kohinan määrän mittaamiseen, eikä ENL vaadi kohinatonta kuvaa indeksin laskemiseksi. Ongelmana indeksin laskemisessa voi kuitenkin olla homogeenisen alueen löytäminen kuvasta, minkä vuoksi ENL-indeksi saatetaan joskus laskea useamman pienen homogeenisen alueen ENL-arvojen keskiarvona. ENL-indeksin arvo määritellään intensiteettikuvulle kaavalla (11), jonka suuri arvo viittaa hyvään kohinanpoistomenetelmään (Masoomi, Hamzehyan & Shirazi 2012, 66.)

$$(11) \quad ENL = \frac{1}{(CV)^2} = \frac{\bar{x}^2}{s^2}$$

6.3 Tunnettuja kohinanpoistomenetelmiä

Seuraavissa aliluvuissa esitellään lyhyesti kirjallisuudessa usein esiintyviä SAR-kuvien esikäsittelyyn käytettyjä kohinanpoistomenetelmiä. Näistä Boxcar, Median, Frost, Gamma MAP, Lee, Refined Lee, Lee Sigma, IDAN ja Kuan ovat laajalti tunnettuja menetelmiä, jotka Kuan-filtteriä lukuun ottamatta löytyvät myös SAR-kuvien käsittelyyn tehdystä Sentinel-1 Toolbox -ohjelman kohinanpoistomenetelmistä. Menetelmistä Boxcar- ja Median-filtterit ovat alun perin olleet digitaalisten kuvien käsittelyyn tarkoitettuja, mutta ovat yleistyneet käyttöön myös SAR-kuvien käsittelyssä.

Useammista erityisesti SAR-kuville suunnitelluista kohinanpoistomenetelmistä on vuosien kuluessa kehitetty uusia versioita, joilla on vastattu haasteeseen polarimetristen SAR-kuvien käsittelystä tutkateknologioiden kehityttyä. Useita yhtä polarisaatiota käsitteleviä menetelmiä voi soveltaa myös polarimetriselle datalle, joskaan menetelmät eivät ota huomioon eri polarisaatioiden korrelaatioita. Useimmiten kohinanpoistomenetelmän nimessä ei eritellä onko suodatin tarkoitettu SAR- vai PolSAR-datalle, vaan tämä ilmenee suodattimien määritelmästä ja tutkimusten lähdeviiteistä. Tämän tutkimuksen yhteydessä tällaisia suodattimia ovat esimerkiksi olleet Boxcar, Improved Sigma Lee ja Refined Lee.

Eri suodattimien esittelyä on rajattu tutkimuksessa käytettyjen GRD-kuvien kohinanpoistossa ohjeistettuihin suodattimiin, jotka eivät erityisesti huomioi polarimetristä dataa tai soveltuvat sekä single-look- että multilook-datalle. Kohinanpoistomenetelmien kuvauksissa ja kaavoissa merkintä I viittaa havaittuun kuvaan, R filtteriöinnin tuloksena olevaan ideaalikuvaan ja L filtteriöinti-ikkunaan.

6.3.1 Boxcar

Yleisesti Boxcar-nimityksellä viitataan suodatusmenetelmiin, jotka tasoittavat tarkasteltavaa aluetta. Silloin kun tasoitus perustuu keskiarvon laskemiseen tietyn kokoisesta ikkunasta, menetelmään saatetaan viitata myös termillä "averaging filter" tai "moving average filter" (Lee, Grunes & Grandi 1999, 2365; Richards 1993, 121). Perusesitys Boxcar-menetelmälle on kaavan (12) mukainen, jossa lasketaan keskiarvo $M \times N$ -kokoisen filtteriöinti-ikkunan sisäisistä pikseliarvoista. (Richards 1993, 118–121.)

$$(12) \quad R(x, y) = \frac{1}{MN} \sum_{m=1}^M \sum_{n=1}^N \phi(m, n)$$

Boxcar-menetelmän huonoiksi puoliksi luetaan sen taipumus sumentaa terävät reunat ja resoluution huonontuminen (Richards 1993, 119; Lee ym. 1994, 321). Kaavan 12 tapauksessa tasoituksen määrää voidaan muuttaa filtteri-ikkunan kokoa muuttamalla (Richards 2013, 119), mutta esityksen (12) lisäksi Boxcar-menetelmästä on myös erilaisia muunnoksia, joilla voidaan paremmin määrittää, milloin pikselin arvo korvataan filtteri-ikkunan pikselien keskiarvolla (Parhad, Aher & Warhade 2021, 2; Richards 1993, 119). Usein tutkimuksissa viitataan kuitenkin Boxcar -menetelmällä määrittämään (12) (Bouchemakh, Smara, Boutarfa & Hamadache 2008, 2; Lee ym. 1999, 2365; Lee, Ainsworth, Wang & Chen 2015, 1150; Yahia, Ali, Mortula, Abdelfattah, Mahdy & Arampola 2020, 859).

6.3.2 Median

Median-suodattimessa käytetään filtteri nimen mukaisesti kuvapikselien mediaania ääriarvojen poistamiseksi. Menetelmässä kuvan pikseleitä tarkastellaan $n \times n$ -ikkunan kokoisissa osissa, jolla kuva käydään järjestelmällisesti läpi ja tarkasteltavan alueen keskimäinen pikseliarvo korvataan ikkunan kaikkien arvojen mediaanilla. (Lee ym. 1994, 321.) Menetelmän kerrotaan olevan tehokas, kun kohina on yksittäistä pistemäistä kohinaa (Lee ym. 1994, 321) ja soveltuvan tilanteisiin, joissa kohinaa on vähemmän kuin puolet ikkunan sisältäminen pikselien lukumäärästä (SAR Preprocessing – Sentinel-1 2019, 37). Silloin kun kohinaa on tarkasteluikkunan sisällä paljon, on mahdollista, että menetelmä korvaakin keskimäisen pikselin ääriarvolla. Vastaavasti pienet yksityiskohdat saattavat kadota, kun arvoja korvataan toisilla (SAR Preprocessing – Sentinel 1 2019, 37). Menetelmän muiksi ongelmiksi kerrotaan tarkkojen reunojen sumentuminen, ohuiden lineaaristen piirteiden häivytyminen ja kohteiden muodon muuttuminen (Lee ym. 1994, 321). Yleisesti filttereitä käsittelevissä aineistoissa menetelmän sanotaankin sopivan huonosti kohinan poistamiseen.

6.3.3 Frost

Frost-menetelmässä $(2n + 1) \times (2n + 1)$ -kokoisen filtteri-ikkunan keskimäiselle pikselille $I(x, y)$ lasketaan uusi arvo ikkunan painotettujen pikseliarvojen summana. Kutakin pikseliarvoa painotetaan eksponentiaalisesti filtteri-ikkunasta laskettujen tilastollisten tunnuslukujen ja pikselin

etäisyyden kautta. (Banerjee, Chaudhuri, Mehra & Misra 2020, 110.) Banerjee ja muut (2020) ovat muotoilleen Frost-filtterin matemaattisen esityksen muotoon (13)

$$(13) \quad R(x, y) = \sum_{k=i-n}^{i+n} \sum_{l=j-n}^{j+n} h(k, l) I(k, l)$$

jossa

$$(14) \quad h(k, l) = K_1 e^{-K_d C_l^2 t}$$

on pikselin $I(k, l)$ painokerroin. Painokertoimessa (14) K_1 on normalisointikerroin (normalizing coefficient), K_d vaimennuskerroin (damping factor), C_l lokaali filtteri-ikkunan keskihajonnasta ja keskiarvosta laskettu variaatiokerroin, ja t pikselin $I(k, l)$ euklidinen etäisyys pikselistä $I(x, y)$. Painokertoimen (9) kaavasta voi nähdä, että suuremmalla variaatiokertoimella ja etäisyydellä pikselin arvoa painotetaan vähemmän. Santoso, Pebrianti, Bayuajji ja Zain (2015, 12) mukaan suurella variaatiokertoimen arvolla filtteri pyrkiikin säilyttämään terävät reunat ja vaihtelun ollessa pientä, filtteri keskiarvoistaa pikseleiden arvoa enemmän.

Frost-filtterin kyky säilyttää kuvassa näkyvät reunat ja poistaa kohinaa riippuu suuresti myös vaimennuskertoimen optimaalisesta valinnasta. Pienellä vaimennuskertoimen arvolla filtteri poistaa kohinaa tehokkaammin, mutta toisaalta tällöin myös reunat sumentuvat (Banerjee ym. 2020, 10). Banerjee ja muut (2020, 110) huomauttavat, että tästä syystä samaa vaimennuskertoimen arvoa ei tulisi käyttää koko kuvaan vaan kertoimen olisi hyvä mukautua filtteri-ikkunan tilastollisiin tunnuslukuihin. Joissakin Frost-filtterin esityksissä vaimennuskertoimelle näkeekin keskihajonnasta ja kernel-ikkunan koosta johdettuja arvoja (Santoso ym. 2015, 12).

6.3.4 Lee

Yksi tunnetuimmista kohinanpoistomenetelmistä on Lee-suodatin, josta on kehitetty myös lukuisia variaatioita. Perinteisen Lee-suodattimen lisäksi eri muunnoksia ovat mm. Refined Lee, Nonlocal Lee, Enhanced Lee, Modified Lee ja Lee Sigma (Parhad, Aher & Warhade 2021, 1; Banerjee ym. 2018, 68). Lee-filtterin ensimmäinen versio on määritelty kaavalla (10)

$$(15) \quad R(t) = \bar{I}(t) + W(t)(I(t) - \bar{I}(t)),$$

missä $\bar{I}(t)$ on filtteri-ikkunan pikseleiden keskiarvo, $I(t)$ ikkunan keskimmäisen pikselin arvo ja $W(t)$ painokerroin. Lee määrittelee painokertoimen $W(t)$ kohinallisen kuvan variaatiokertoimen $C_I = \frac{\sigma_L}{\bar{I}(t)}$ ja kohinattoman kuvan variaatiokertoimen $C_R = \frac{1}{\sqrt{ENL}}$ kautta kaavalla $W(t) = 1 - \frac{C_R^2}{C_I^2}$, joiden laskemisessa kohinan on oletettu noudattavan normaalijakaumaa (Banerjee ym. 2018, 68–69). Sekä Banerjee ja muut (2018, 69) että Parhad ja muut (2021, 1) toteavat Lee-filtterin ensimmäisen version ongelmaksi oletuksen kohinan määrästä SAR-kuvissa, jonka mukaan kohina on samanlaista riippumatta tarkasteltavasta alueesta. Kuvalle jossa esiintyy paljon vaihtelevia alueita, tämä oletus ei kuitenkaan pidä paikkaansa ja suodatus johtaa reunojen ja yksityiskohtien sumentumiseen. Tätä ongelmaa korjaamaan on siten kehitetty eri muunnoksia.

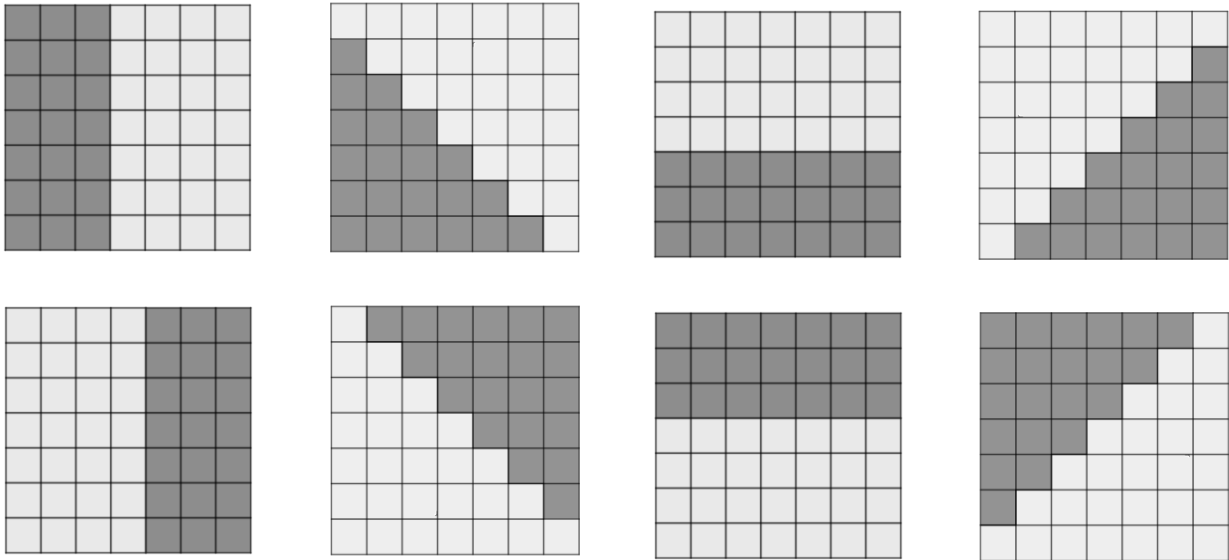
6.3.5 Refined Lee

Refined Lee Filter, lyhyemmin Refined Lee -nimellä useimmiten viitataan Jong-Sen Leen vuonna 1981 julkaistuun paranneltuun menetelmään Lee-filtteristä (Lee ym. 1994, 326; Argenti ym. 2013, 16; Medasani & Reddy 2018, 169). Toisinaan Refined Lee Filter -nimityksellä saatetaan taas viitata Yommy, Liu & Wun (2015) parantamaan Lee-filtteriin, missä suodatusikkunan sisältä valitaan K-Nearest Neighbors -algoritmillä k-kappaletta tunnuslukujen laskemiseen käytettävää pikseliarvoa (Banerjee ym. 2018, 69). Tässä esitetään näistä menetelmistä ensiksi mainittu.

Yksi alkuperäisen Lee-filtterin ongelmista oli kohinan poistaminen reunojen raja-alueilta. Lee (1981) toteaa, että korkean kontrastin alueilla, kuten objektien reunoilla, menetelmä jättää kohinan suodattamatta. Jotta kohina reuna-alueilla saataisiin poistettua, Lee määrittelee Refined Lee -filtterissä uudelleen pikseleiden ympäristön ja siten alueen, josta lokaali keskiarvo ja varianssi lasketaan. Kun lokaali varianssi saadaan pienemmäksi, menetelmä suodattaa myös reuna-alueilla esiintyvän kohinan. (Lee 1981, 380–381; Argenti ym. 2013, 16.)

Alueen määrittäminen tehdään tutkimalla reunan suuntaa 7*7-kokoisen filtteri-ikkunan sisällä 3*3-kokoisissa osissa. 3*3-ikkunoiden pikseleistä lasketaan keskiarvo, joita vertaillaan keskenään lopullisen pikseleiden osajoukon valitsemiseksi. Osajoukon määrittämiseen on kahdeksan kuviossa 4 esitettyä vaihtoehtoa, joihin 7*7-ikkunan osajoukko on merkitty vaaleanharmaalla ja joista oikea

vaihtoehto selvitetään keskiarvojen vertailulla. Kun pikseliä ympäröivä osajoukko on saatu määritettyä, sen arvoista lasketaan lokaali keskiarvo ja varianssi, ja kohina suodatetaan Lee-filtterin mukaisesti. Alkuperäisenä ajatuksena menetelmän käytössä on ollut soveltaa sitä vain reuna-alueiden pikseleihin, jotta laskenta-ajan lisääntyminen pysyisi vähäisenä. (Lee 1981, 382–388.)



Kuvio 4. Reunan tunnistusikkunat.

6.3.6 Lee Sigma ja Improved Sigma Lee

Refined Lee -filtterin tavoin Lee Sigma parantaa alkuperäistä Lee-filtteriä rajaamalla keskiarvon ja varianssin laskemiseen käytettävää joukkoa. Lee Sigma -menetelmässä oletetaan, että kohina on normaalijakautunutta ja tasoittaa vain pikseliarvot, jotka poikkeavat enintään 2σ :n verran tarkasteltavan pikselin arvosta, jolloin jakauman kertymäfunktion arvoksi saadaan 0,955. Tuloksena jakauman hännillä olevia merkittävästi poikkeavia pikseleiden arvoja ei oteta mukaan keskiarvon ja varianssin laskemiseen. Menetelmän ongelmana on kuitenkin keskiarvon ja varianssin laskemisessa tuleva virhe, sillä kohinan jakautuminen ei ole symmetristä eikä menetelmä osaa ottaa huomioon nykyään saatavilla olevan SAR-datan vaihtelevuutta. (Lee 1983, 86; Medasani & Reddy 2018, 169; Lee, Wen, Ainsworth, Chen & Chen 2009, 202–203.)

Ongelman korjaamiseksi Lee Sigma -filtteristä on kehitetty paranneltuja versioita, jotka huomioivat paremmin kohinan jakautumisen. Lee, Wen, Ainsworth, Chen ja Chen vuonna 2009 julkaistussa menetelmässä sigman valita perustuu kohinan todennäköisyysjakauman määrittämiseen, millä pyritään korjaamaan mahdollinen vino jakauma suodatettavan kuvan pikseliarvoissa. (Argenti ym.

2013, 18–19; Lee ym. 2009, 204). Pikseliarvojen jakauman lisäksi Lee ja muut (2009) ovat parannelussa Sigma Lee -menetelmässä halunneet oikaista aikaisemman menetelmän kaksi muuta epäkohtaa, jotka ovat kirkkaiden pikseleiden sumentuminen ja yksittäisten tummien pikseleiden suodattamattomuus (Lee ym. 2009, 212).

Improved Sigma Lee -filterissä pikseliarvojen jakauman vinoutuneisuus huomioidaan laskemalla niiden todennäköisyysjakauma (16) single- ja multilook SAR-datan “number of looks” -tiedon (N) sekä tarkasteluikkunan kohinattoman keskiarvon \bar{I} avulla. Koska SAR-datasta ei voida suoraan laskea kohinatonta keskiarvoa, arvioidaan tämä MMSE-menetelmällä (Minimum Mean Square Error), missä lokaali keskiarvo ja varianssi lasketaan 3*3-ikkunoista. (Lee ym. 2009, 204–205.)

$$(16) \quad p_N(I) = \frac{N^N I^{N-1}}{(N-1)! \bar{I}^N} \exp(-NI/\bar{I})$$

Todennäköisyysjakauman $p_N(I)$ kautta menetelmässä määritellään seuraavaksi pikseliarvojen jakauman raja-arvot I_1 ja I_2 siten, että ne toteuttavat yhtälöt (17) ja (18) arvolla ξ ja jakauman keskiarvolla \bar{I} . Tämän jälkeen lopullinen kohinan suodatus tehdään MMSE-menetelmällä, jossa lokaali keskiarvo ja varianssi lasketaan 7*7 tai suuremmista ikkunoista siten, että laskennassa käytetyt pikseliarvot on rajattu arvojen I_1 ja I_2 mukaan. (Lee ym. 2009, 204–205; Lee ym. 2015, 1152.)

$$(17) \quad \xi = \int_{I_1}^{I_2} p_N(I) dI$$

$$(18) \quad \bar{I} = \frac{1}{\xi} \int_{I_1}^{I_2} p_N(I) dI.$$

Menetelmän ensimmäisen vaiheen keskiarvon ja varianssin laskemisesta Lee ja muut (2009) mainitsevat artikkelissaan, että näiden laskemiseen voidaan käyttää joko 3*3- tai 5*5-kokoista ikkunaa. Useimmiten SAR-kuvien pikseliarvot eivät ole tilastollisesti riippumattomia 3*3-kokoisen ikkunan alueella, mutta 5*5-ikkuna voi toisaalta aiheuttaa reunojen ja pienten kohteiden sumentumista. Menetelmän käytön helpottamiseksi Lee ja muut (2009) ovat lisäksi valmiiksi taulukoineet lopulliseen suodatukseen tarvittavia arvoja erilaisille kuville ja kertymäfunktion arvoille. Kertymäfunktion arvoista Lee ja muut (2009) toteavat, että maaston ja satojen luokitteluun suositetaan useimmiten kertymäfunktion arvoja 0,8 ja 0,9. (Lee ym. 2009, 204–205.)

6.3.7 Gamma Maximum a Posteriori (Gamma MAP)

Gamma MAP -filtteri on Kuan, Sawchuck, Strand ja Chavelin vuonna 1987 kehittämä todennäköisyyslaskentaan (MAP, Maximum likelihood) perustuva suodatin, jossa Gamma-etuliite viittaa oletukseen kohinan todennäköisyysjakaumasta. Gamma MAP -filtterin muita oletuksia ovat olleet kohinan multiplikaatiivinen luonne, havaitun kohinan riippumattomuus toisistaan ja kuvannettavan alueen todellisen intensiteetin keskiarvon ja varianssin noudattaminen normaalijakaumaa. Myöhemmin on kuitenkin todettu, että oletus intensiteettien normaalijakautuneisuudesta ei aina parhaiten kuvaa todellisuutta ja siten myös Gamma MAP -filtteristä on kehitetty erilaisia muunnoksia, joissa on käytetty muita jakaumia. Erotuksena alkuperäisen ja varioitujen Gamma MAP -filttereiden välillä saatetaan käyttää etuliitteitä, kuten Gaussian-Gamma MAP ja Gamma-Gamma MAP, joilla viitataan oletukseen kuvannettavan alueen intensiteettien jakaumasta. (Nezry 2014; Lopez, Nezry, Touzi & Laur 1990, 2409–2412.)

Tutkimuksessa hyödynnettävän SNAP-ohjelmiston Help-osiossa viitataan lähteenä Mansoupour, Rajabi ja Blais (2006) tutkimukseen, jossa Gamma MAP -filtteri määrittellään kaavan (14) mukaan. Mansoupour ja muiden (2006) mukaan kaavassa (19) R on laskettava pikseliarvo, \bar{I} lokaali keskiarvo, σ alkuperäisen kuvan varianssi ja I syöte. Lokaali keskiarvo ja varianssi arvioidaan kuvasta $n \times n$ -kokoisen filtteri-ikkunan sisältä (Lopez ym. 1990, 2409).

$$(19) \quad R^3 - \bar{I}R^2 + \sigma(R - I) = 0$$

6.3.8 Kuan

Lee-filtterin tapaan Kuan-kohinanpoistomenetelmässä suodatetaan ikkunan keskimäinen pikseli kaavan (15) mukaisesti, mutta painokerroin $W(t)$ lasketaan eri tavalla. Kuan-filtterissä painokerroin määrittellään kaavan (20) kautta,

$$(20) \quad W = \frac{1 - C_U^2 / C_I^2}{1 + C_U^2},$$

missä $C_U = \frac{1}{\sqrt{ENL}}$ ja $C_I = \frac{\sigma_L}{I(t)}$ on lokaali, filtteri-ikkunan sisältä laskettu variaatiokerroin. (Banerjee & Chaudhuri 2018, 70; Parhad, Aher & Warhade 2021, 2.) Aki, Tabbara & Yaacoub (2012) mukaan

variaatiokertoimen C_U laskemisessa ENL -arvon tilalla on täsmällisempää käyttää $NLOOK$ -arvoa, joka teoriassa onkin sama kuin ENL . Koska ENL -arvolla on kuitenkin vaikutusta suodatuksen tasoon, on tämän valintaan esitetty erilaisia menetelmiä manuaalisen kokeilun sijasta. (Akl, Tabbara, Yaacoub 2012, 53.)

6.4 Viime vuosien kohinanpoistomenetelmiä

Tässä luvussa esitellään SAR-kuvien kohinan poistoon kehitetty Speckle2Void ja sen kehitystyön taustalla olleet digitaalisille kuville suunnitellut Noise2Void ja Laine, Karras, Lehtinen ja Ailan (2019) kehittämä kohinanpoistomenetelmä. SAR-kuvien kohinan poistamiseksi on viime vuosina kehitetty useampia neuroverkkoihin perustuvia menetelmiä. Tutkittaviin menetelmiin valittiin Speckle2Void, jonka käytöstä tiedettiin, ettei mallin kouluttamiseen tarvita kohinattomia kuvia, joiden hankkiminen SAR-kuvien tapauksessa ei ole mahdollista.

6.4.1 Noise2Void

Noise2Void-menetelmässä Krull, Buchholz ja Jug (2019) ovat lähteneet ratkomaan konvoluutioneuroverkkojen kouluttamiseen tarvittavien datasettien ongelmaa. Aikaisemmat konvoluutioneuroverkoilla kehitetyt menetelmät kohinan poistamiseen tarvitsevat kouluttamiseen sekä kohinallisia että puhtaita kuvia tai kohinallisia kuvapareja samasta kohteesta, mitkä ovat molemmat käytännössä vaikeita toteuttaa. Krull ja muut (2019) ovat pyrkineet ratkaisemaan ongelman käyttämällä itseoppivaa neuroverkkoa ja siinä blind-spot-tekniikkaa, jossa ennustettavaa pikseliä ja sen intensiteettiä ei huomioida koulutuksessa. Blind-spot-tekniikan käyttöä Krull ja muut (2019) perustelevat neuroverkkojen taipumuksella suoraan oppia ennustamaan pikselin arvo samaksi, ellei ennustettavaa pikseliä jätetä pois laskuista. (Krull ym. 2019; 2124–2127.) Menetelmän ansiosta samaa kohinallista kuvaa voidaan siten käyttää neuroverkon koulutuksessa sekä sisääntulo- että ulostuloarvoina (Laine, Karras, Lehtinen & Aila 2019, 1–2).

Yksittäinen kuvapikseli voidaan ajatella muodossa $x = s + n$, missä x on havaittu arvo, s kohinaton signaali ja n kohina. Signaalin ja kohinan arvoista Krull ja muut (2019, 2125) ovat tehneet kaksi oletusta: yksittäisten pikseleiden signaalit eivät ole toisistaan riippumattomia ja pikseleiden kohina on riippumatonta signaalin arvosta. Toisin sanoen kohinan määrä yksittäisessä pikselissä ei riipu

ympäröivien pikseleiden arvosta, minkä takia kohinaton signaali s voidaan ennustaa ympäröivistä pikseleistä (Krull ym. 2019, 2125–2126).

Blind-spot-tekniikka on Noise2Void-menetelmässä toteutettu kopioimalla tarkasteluikkunasta satunnainen pikseli ja sen intensiteetti ennustettavan pikselin tilalle. Näin peitetään pikselin havaittu arvo ja estetään neuroverkon ei-toivottu oppiminen sisääntuloarvon kopioitumisesta suoraan ulostuloarvoksi. Käytännössä gradienttien laskeminen yllä kuvatulla tavalla jokaiselle kuvan pikselille olisi hyvin raskasta, joten Krull ja muut (2019) ovat toteuttaneet kohinan poistamisen satunnaisesti valittujen 64×64 -ikkunoiden sisällä, joista puolestaan satunnaisesti valitaan n -kappaletta suodatettavia pikseleitä. (Krull ym. 2019, 2127–2128.)

6.4.2 Laine, Karras, Lehtinen ja Aila

Noise2Noise- ja Noise2Void-menetelmien innoittamina Laine, Karras, Lehtinen ja Aila (2019) ovat tutkineet kohinallisista kuvista saatavaa dataa ja kehittäneet Noise2Void-menetelmän blind-spot-tekniikkaa tehokkaammaksi. Kehittämässään menetelmässä Laine ja muut (2019) kiertävät tarkasteluikkunaa neljästä eri suunnasta ennustettavan pikselin ympärillä, mikä johtaa samaan tulokseen kuin yksittäisen pikselin poisjättäminen. Käytännössä vastaanottavan ikkunan kiertäminen tehdään syöttämällä neuroverkolle neljä rotaatiota samasta kuvasta yhdessä neuroverkon haarassa, millä saadaan pidettyä koulutettavien painoarvojen määrä vähäisempänä kuin syötettäessä sama kuva neljälle eri haaralle. (Laine ym. 2019, 2.)

Blind-spot-tekniikan tehostamisen lisäksi Laine ja muut (2019) ovat ottaneet uudelleen tarkasteluun kohinan mallin, jonka perusteella ennuste lasketaan kohinattomalle pikselille. Laine ja muut (2019) huomauttavat pelkällä blind-spot-tekniikalla lasketun ennusteen jättävän huomiotta hyödyllistä tietoa, joka voitaisiin saada kohinallisen pikselin arvosta ja näin johtavan heikkoihin tuloksiin kohinan poistamisessa. Tästä johtuen Laine ja muut (2019) yhdistävät kohinallisesta pikselistä y ja pikselin ympäristöstä Ω_y saadun tiedon, ja ovat muotoilleet kohinallisen pikselin y pistetodennäköisyyden muotoon (21)

$$(21) \quad p(y|\Omega_y) = \int p(y|x) p(x|\Omega_y) dx,$$

missä $p(y|x)$ on y :n pistetodennäköisyys ehdolla x ja $p(x|\Omega_y)$ kohinattoman pikselin x pistetodennäköisyys ehdolla Ω_y . Pistetodennäköisyyden $p(y|x)$ voidaan ajatella mallintavan kohinaa ja $p(x|\Omega_y)$ olevan alkuperäisen pikselin arvo ilman kohinaa. Esityksestä (21) Laine ja muut (2019) ovat edelleen johtaneet säännön kohinattoman pikselin x pistetodennäköisyydelle ehdolla y ja Ω_y (22), missä huomioidaan sekä pikseliä ympäröivät arvot että pikselin havaittu kohinallinen arvo. (Laine ym. 2019, 3.)

$$(22) \quad p(x|y, \Omega_y) \propto p(y|x) p(x|\Omega_y)$$

Yllä olevasta säännöstä $p(x|\Omega_y)$ voidaan selvittää neuroverkon avulla, minkä jälkeen kohinattoman pikselin x arvo voidaan laskea MMSE-menetelmällä (Molini ym. 2022). Kohinallisen pikselin y pistetodennäköisyydestä (21) voidaan approksimoida kohinattoman arvon x todennäköisyys ehdolla Ω_y , mitä tietoa voidaan hyödyntää x :n todennäköisyyden laskemisessa ehdoilla y ja Ω_y . (Laine ym. 2019, 3.)

6.4.3 Speckle2Void

Speckle2Void kehitettiin erityisesti SAR-kuvien kohinanpoiston näkökulmasta, missä aikaisemmat konvoluutioneuroverkkoihin perustuvat menetelmät ovat käyttäneet mallin koulutuksessa valvottua oppimista ja vaatineet koulutukseen myös puhtaita kuvia. Näissä menetelmissä optisia kuvia on pidetty pohjatotuutena ja koulutukseen käytettävät kuvat on saatu lisäämällä kuviin synteettistä kohinaa. Molini, Valsesia, Fracastoro ja Magli (2020) mukaan tämä voi kuitenkin johtaa virheisiin ja radiometrinen ominaisuuksien huonoon säilymiseen. Tästä johtuen Molini ja muut (2020) ovat ottaneet Speckle2Void-menetelmän pohjalle Laine ja muiden (2019) mallin (22) kohinattoman pikselin todennäköisyydestä sekä blind-spot-tekniikan tarvittavien parametrien selvittämiseen. (Molini, Valsesia, Fracastoro & Magli 2020.)

Laineen ja muiden (2019) luomaa kehystä hyödyntäen Molini ja muut (2020) ovat muotoilleet esityksen sopimaan SAR-kuvissa esiintyvään kohinaan. SAR-kuvien kohina on multiplikatiivista ja usein sen oletetaan noudattavan käänteistä Gamma-jakaumaa, jonka on todettu mallintavan hyvin erityisesti heterogeenista SAR-dataa (Molini ym. 2020). Tämän oletuksen turvin Molini ja muut (2020) ovat muotoilleet kohinattoman pikselin x_i pistetodennäköisyyden muotoon

$$(23) \quad p(x_i|\Omega_{y_i}) = \text{inv}\Gamma(\alpha_{x_i}, \beta_{x_i})$$

missä α_{x_i} ja β_{x_i} riippuvat ympäristön Ω_{y_i} arvoista. Edelleen kohinattoman pikselin x_i todennäköisyydelle ehdoilla y_i ja Ω_{y_i} on johdettu kaava (24) ja kohinallisen pikselin y_i ehdollisen todennäköisyyden muotoon (25)

$$(24) \quad p(x_i|y_i, \Omega_{y_i}) = \text{inv}\Gamma(L + \alpha_{x_i}, \beta_{x_i} + L_{y_i})$$

$$(25) \quad p(x_i|\Omega_{y_i}) = \frac{L^L y_i^{L-1}}{\beta_{x_i}^{-\alpha_{x_i}} \text{Beta}(L, \alpha_{x_i})(\beta_{x_i} + L_{y_i})^{L + \alpha_{x_i}}}$$

Näiden esitysten pohjalta blind-spot-konvoluutioneuroverkko koulutetaan minimoimaan häviöfunktio (26), siten että parametrit α_{x_i} ja β_{x_i} mahdollisimman hyvin toteuttavat SAR-kuvan havaitut arvot. (Valesia 2020; Molini ym. 2020)

$$(26) \quad l = - \sum_i \log p(y_i|\Omega_{y_i})$$

Yllä oleva esitys Speckle2Void-menetelmälle on Molini ja muiden vuonna 2020 esittämä ensimmäinen versio Speckle2Void-menetelmästä, johon tekijät ovat vielä myöhemmin tehneet parannuksia neuroverkkomalliin, häviöfunktioon ja koulutusprosessiin (Molini ym. 2022).

Mallin suoriutumista valvotun oppimisen ja multitemporal-menetelmiin Molini ja muut (2020) vertailivat sekä numeerisesti että subjektiivisesti optisten kuvien ja SAR-kuvien avulla. Kohinan poistoa optisten kuvien tapauksessa oli testattu vertailuun yleistyneistä kuvista lisäämällä niihin syntettistä kohinaa ja laskemalla tuloksista PSNR-arvo. SAR-kuvien tapauksessa numeeriseen vertailuun mittarina käytettiin ENL-arvoa. PNSR-arvolla vertailtuna menetelmät saivat lähes yhtäläisiä tuloksia. ENL-arvolla mitattuna Speckle2Void näytti tasoittaneen kuvia vähiten, mutta subjektiivisessa vertailussa menetelmän saattoi huomata säilyttäneen pieniä yksityiskohtia ja tekstuuria erinomaisesti. Kuviossa 5 on esitetty Molini ja muiden (2020) tekemä vertailu optisille kuville ja SAR kuville. Kuvasarjojen viimeisenä on tulos Speckle2Void-mallilla.



Fig. 2. Synthetic images: Noisy, PPB (21.13 dB), SAR-BM3D (22.71 dB), SAR-CNN (23.37 dB), our method (23.32 dB).

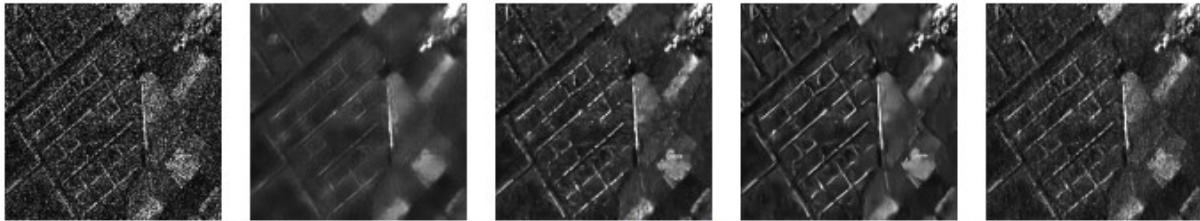


Fig. 3. Real SAR images: Noisy, PPB, SAR-BM3D, SAR-CNN, our method.

Kuvio 5. Molini ja muiden (2020) tekemä Speckle2Void-mallin vertailu (Molini ym. 2020).

6.5 Kohinanpoistomenetelmien tutkimus

Edellä esitellyistä kohinanpoistomenetelmistä Boxcar, Median, Frost, Lee, Refined Lee, Lee Sigma ja Kuan edustavat perinteisiä ja hyvin tunnettuja menetelmiä. Näin ollen näitä on ehditty tutkia ja vertailla sekä keskenään että uusien menetelmien kanssa useasta eri näkökulmasta. Tutkateknologioiden kehityttyä ja useamman polarisaatioyhdistelmän mittaamisen yleistyttyä filttareiden tutkimuksessa on enenevässä määrin selvitetty PolSAR-datalle soveltuvia kohinanpoistomenetelmiä. Perinteisten menetelmien vertailua on kuitenkin tehty vielä viime vuosinakin.

Suuri osa tutkimuksista keskittyy tekemään vertailua teorian, eri metriikoiden ja mallikuvien kautta. Näiden lisäksi löytyy myös muutamia käyttötapaustutkimuksia, joissa filttarin valintaa on tutkittu käytännönongelman näkökulmasta. Tutkimusten analysointi on jaoteltu seuraaviin lukuihin käytetyn tutkimusmenetelmän mukaan ja lopuksi koottu kohinanpoistomenetelmien tutkimuksissa esiintyneitä ongelmia, näihin liittyviä havaintoja ja pohdintaa. Vaikka ensimmäisissä tutkimuksissa on tehty vertailua myös laskennallisen tehokkuuden näkökulmasta, tämä tarkastelu on jätetty pois, sillä ongelma ei ole enää yhtä olennainen kuin muutama vuosikymmen sitten. Lisäksi aiheen tarkastelua on rajattu jättämällä käsittelemättä vain PolSAR-datan näkökulmasta tehdyt tutkimukset pilottitapauksessa käytettävän datan vuoksi.

6.5.1 Vertailevat teorialuokittelut

Ensimmäisen laajan vertailun SAR-kuvien käsittelyyn tarkoitettujen kohinanpoistomenetelmien välillä tekivät Lee, Jurkevich, Dewaele, Wanbacq ja Oosterlinck vuonna 1994. Vertailtavina menetelminä olivat Mean, Median, Lee, Refined Lee, Sigma Lee, Kuan ja Frost, sekä geometrinen ja morfologinen filtti. Näiden lisäksi Mean ja Median -filttereitä oli tutkimuksessa testattu useammalla eri filtti-ikkunan ko'illa. Mean ja Median-filttereistä Lee ja Jurkevich toteavat, etteivät nämä sovellu SAR-kuvien kohinan poistoon kohinan multiplikatiivisen luonteen takia, mutta olivat otettu mukaan tutkimukseen vertailun vuoksi. (Lee ym. 1994, 320-321.)

Filttereiden vertailun Lee ja muut (1994) toteuttivat sekä NASA/JPL AIRSAR ja SIR-B-satelliittien SAR-kuvilla, että erilaisia viivoja ja pisteitä sisältävällä kuvalla, johon oli synteettisesti lisätty kohinaa. Satelliittien SAR-kuvista oli valittu kaksi erilaista kuvaa, joissa toisessa oli kuva kaupunkialueesta meren rannalla ja toisessa peltoja ja metsiä maaseudulta. Tutkimuksessa filttereiden toimintaa analysointiin useammasta eri näkökulmasta ottaen huomioon tasoitus tasaisilla alueilla, kohinan poisto reuna-alueilla, sekä pienten kohteiden, muotojen ja tekstuurien säilyvyys. Näitä Lee ja muut (1994) tutkivat suodatettuja kuvia subjektiivisesti tarkastelemalla ja käyttämällä erilaisia mittareita. (Lee ym. 1994, 321, 331–336.)

Tutkimuksen tuloksena Lee ja muut (1994) eivät nostaneet yhtä filttiä ylitse muiden, vaan ohjeisivat valitsemaan kohinanpoistomenetelmän toivotun lopputuloksen ja käyttötarkoituksen mukaan. Tutkimuksessa käsiteltyjä kuvia ja näistä laskettuja arvoja vertailemalla saattoi todeta, että parhaiten homogeenisia alueita tasoittivat Mean- ja geometrinen filtti, mutta toisaalta nämä säilyttivät heikosti ohuita pistemäisiä kappaleita. Parhaiten reunoja ja ohuita viivoja säilytti Refined Lee -filtti, joka homogeenisten alueiden tarkastelussa suoriutui muihin filttireihin verrattuna keskiarvoisesti. Tästä johtuen tutkijat totesivat Refined Lee -filterin olevan sopiva SAR-kuvien visuaaliseen tarkasteluun. Lähes yhtä hyväksi tähän tarkoitukseen Lee ja muut (1994) mainitsevat Lee- ja Frost-suodattimet, jotka tasoittivat samankaltaisia alueita Refined Lee:tä paremmin, mutta olivat huonompia muilla mittareilla tarkasteltuina. SAR-kuvien segmentointiongelmiiin Lee ja muut (1994) arvelivat geometrisen filterin sopivan hyvin homogeenisten alueiden tasoittumisen vuoksi. Toisaalta Lee ja muut (1994) huomauttavat, että myös Refined Lee -filterillä on mahdollista saada aikaa vastaava tulos, kun filttiä käytetään useampaan kertaan. Tutkijat nostavat esille myös

mahdollisuuden käyttää kahta eri filtteriä yhdessä, jolloin kuvaa saadaan suodatettua toivotulla tavalla. (Lee ym. 1994, 338.)

Pienimuotoisempia tutkimuksia luvussa 6.3. esiteltyjen filttäreiden soveltuvuudesta SAR-kuville on Leen ja muiden (1994) tutkimuksen jälkeen tehty useampia. Santoso, Pebrianti, Bayuaji ja Zain (2015) vertailivat Frost-, Lee-, Median- ja Mean-filttereitä ALOS-PALSAR-kuville. Vertailu toteutettiin tutkimalla kuvan laatua mittaavia mittareita (MSE, PSNR, SNR ja AD), sekä subjektiivisesti suodatettuja kuvia tarkastelemalla. Santoso ja muut (2015, 14) totesivat Frost-filtterin parhaimmaksi pienimpien MSE- ja AD-arvojen, sekä suurimpien PSNR- ja SNR-arvojen perusteella. Kuvien visuaalisessa vertailussa tutkijat nostivat myös Frost-menetelmän parhaimmaksi tekstuurien ja reunojen säilyvyyden vuoksi. (Santoso ym. 2015, 12–14.) Sentinel-1-satelliittien GRD-kuville Ansari, Ghos, Behera ja Gupta (2020) puolestaan ovat nostaneet parhaimmaksi Boxcar-menetelmän, jolla MSE- ja AD-arvot olivat vertailtujen menetelmien pienimmät ja PNSR suurin. ENL-mittarilla mitattuna suurin lukema oli saatu Lee-filtterillä, ja Frost- ja Gamma MAP -filtterit olivat vertailussa seuraavina. (Ansari, Ghos, Behera & Gupta 2020, 92.)

Vuonna 2021 Parhad, Aher ja Warhade vertailivat Boxcar, Frost, Kuan, Lee ja Nonlocal Lee -filttereitä Flevoland-datasetillä, joka sisältää SAR-kuvia P-, L- ja C-aallonpituusalueilta ja sisältäen kaikki polarisaatiot. Filttäreiden suoriutumista oli tutkittu kullekin aallonpituusalueelle/kaistalle erikseen laskemalla ENL-, SSI-, ESIV- ja ESIH-arvot, sekä kuvan suhteen keskiarvon ja keskihajonnan. Lisäksi suodattimia oli tutkittu objektiivisesti suodatettuja kuvia vertailemalla. Tuloksista Parhad, Aher ja Warhade (2021, 4) toteavat, ettei tutkituista filttäreistä voi nostaa yhtä ylitse muiden, joka olisi kaikilla mittareilla muita parempi. Sen sijaan tutkijat alleviivaavat, että filtterin valinnan tulisi pohjautua käytettävän sovelluksen vaatimuksiin. Vaikka Nonlocal Lee ja Frost-filttereiden todettiin olevan visuaaliseen tarkasteluun muita parempia, huomauttivat Parhad, Aher ja Warhade (2021), että Lee-filtterin iteratiivisella käytöllä voitaisiin saada vastaavia tuloksia. Tutkijat myös toivat esiin mahdollisuuden yhdistellä kahta eri filtteriä halutun lopputuloksen saamiseksi. Esimerkkeinä annettiin Sigma- ja Median-filtterin yhdistäminen, joista Median-filtteriä voitaisiin käyttää yksittäisten kohinapikseleiden poistamiseen. (Parhad ym. 2021, 4.)

Kuten ensimmäisten SAR-kuville sovellettujen kohinanpoistomenetelmien tapauksessa, moni syväoppimiseen perustuva kohinanpoistomenetelmä on alun perin suunniteltu käytettäväksi

optisille kuville. Syväoppimismenetelmillä saadut erinomaiset tulokset optisten kuvien kohinanpoistossa on antanut toivoa hyvistä tuloksista myös SAR-kuvilla, mutta Fracastoro, Magli, Poggi, Scarpa, Valsesia ja Verdilovan mukaan (2021) näillä ei kuitenkaan ole saatu odotettuja tuloksia, vaikka parannusta edeltäviin menetelmiin on ollut. Myöhemmin syväoppimismenetelmillä on kehitetty myös erityisesti SAR-kuville suunnattuja suodattimia, mutta näiden vertailua hankaloittavat muiden muassa tutkimuksissa vaihtelevasti koulutukseen käytetyt datasetit, vertailussa käytetyt kuvat ja avoimen koodin puute, minkä vuoksi useat tutkimukset eivät ole toistettavia. (Fracastoro, Magli, Poggi, Scarpa, Valsesia & Verdilova 2021, 1–2, 14.)

Yleisesti syväoppimiseen pohjautuvista menetelmistä Fracastoro ja muut (2021) nostavat esille haasteet kouluttamiseen käytetyssä datassa. Koska SAR-kuvien tapauksessa kohinattomia kuvia ei ole mahdollista saada, ovat erityisesti valvotun oppimisen menetelmät ongelmallisia. Kouluttamista varten datasettejä on luotu lisäämällä kuviin synteettistä kohinaa tai käyttämällä lukuisista kuvista muodostettuja kohinattomia kuvia (multitemporal fusion). Edellisissä tilanne ei vastaa todellista kohinaa SAR-kuvissa ja jälkimmäisessä ongelmana on sekä datasettiin tarvittavien kuvien määrä että tilastollisten ominaisuuksien muuttuminen. (Fracastoro ym. 2021, 4–7.)

Datasetin kokoamiseen liittyvät ongelmat ovat johtaneet kehitykseen itseoppiviin malleihin perustuvissa kohinanpoistomenetelmissä, mutta näiden tuloksista Fracastoro ja muut (2021) toteavat, että yksittäisten menetelmien ja koko suuntauksen tutkimus on vielä lapsenkengissä ja vaatii lisää tutkimusta. Fracastoro ja muut (2021) kertovat, että uudet menetelmät perustuvat optisten kuvien kohinanpoistoon kehitettyihin menetelmiin ja näistä tutkimuksessa nimetään Noise2Noise, Noise2Self, Noise2Void ja Laine ja muiden (2019) -menetelmä. Vaikka Fracastoro ja muut (2021) eivät vertaile uusimpia menetelmiä keskenään, osoittavat he näihin menetelmiin liittyviä ongelmia, jotka ovat 6.4 luvussakin esiintuotuja. Muita huomioita uusimmista menetelmistä on, että vaikka itseoppivien menetelmien tapauksessa olisi mahdollista hyödyntää käsittelemättömiä SAR-kuvia, näissä on kuitenkin saatettu käyttää kuvia, joihin synteettistä kohinaa. (Fracastoro ym. 2021, 13–14.)

Vuonna 2021 tekemässään tutkimuksessa Fracastoro ja muut (2021) eivät erikseen ole nostaneet esille täysin polarimetrisen SAR-datan käsittelyä syväoppimispohjaisilla kohinanpoistomenetelmillä. Tämän voi arvella johtuvan haasteesta ylipäättään vertailla menetelmiä tutkimusten heikon

toistettavuuden takia ja Fracastoro ja muut (2021, 7) mainitsevatkin menetelmien kehityksessä käytetyistä dataseiteistä, ettei näitä useimmiten ole tarkasti kuvattu. Ottaen huomioon myös vertailevien tutkimusten vähäisyyden syväoppimismenetelmiin pohjautuvista kohinanpoistomenetelmistä, aiheen käsittely vaatinee ensin yleisempää tutkimusta.

6.5.2 Vertailevat tapaustutkimukset

Eri kohinanpoistomenetelmiä tutkivia tapaustutkimuksia, joissa filttäreiden suoriutumista olisi vertailtu konkreettisten käyttötapauksen näkökulmasta, on tehty teoriaan keskittyviä tutkimuksia huomattavasti vähemmän. Yksi tällainen on Ananto, Putri, Hadi, Hanum, Wiryawan, Prabaswara ja Arjasakusuman (2019) tutkimus eri kohinanpoistomenetelmien toimivuudesta metsien biomassan arvioinnissa. Tutkimuksessa vertailtiin Frost, Gamma-MAP, Median ja Refined Lee -filttereitä Sentinel-1-satelliittien C-band SAR-kuvien ja kenttätutkimuksen avulla. Sentinel-1-satelliittien SAR-kuvista sekä maan päältä kerätystä datasta tehtiin lineaarisen regression ennuste biomassalle, minkä R^2 - ja RMSE-arvoilla eri filttäreiden kanssa arvioitiin kohinanpoistomenetelmien sopivuutta luokitteluprosessissa.

Tutkimuksen tuloksena Ananto ja muut (2019) nimeävät Frost-filtterin yhdessä VH-polarisaation kanssa tuottavan parhaimman korrelaation biomassan ja takaisinsironnan välillä. Tutkituissa menetelmissä ei kuitenkaan ollut merkittäviä eroja, vaan filtteriä huomattavasti suurempi merkitys oli käytetyllä polarisaatiolla, joista VH osoittautui paremmaksi biomassan arviointiin kuin VV-polarisaatio. Frost-filtterin hienoisen etumatkan muihin filttäreihin Ananto ja muut (2019) päättelevät johtuvan filtterin kyvystä tasoittaa kohinaa ja säilyttää vaihtelevan maaston piirteet. Heikoiten biomassan ja takaisinsironnan yhteyttä saatiin selitettyä käytettäessä Refined Lee -filtteriä yhdessä VV-polarisaation kanssa. Tästä tutkijat kuitenkin huomauttavat, että erot ovat voineet johtua kerättyjen näytteiden vähäisyydestä ja aiheesta tarvittaisiin tutkimus huomattavasti suuremmalla otoskoolla. (Ananto ym. 2019.)

6.6 Kohinanpoistomenetelmien valinta maankäytön luokittelua koskevissa tutkimuksissa

Yksistään maankäytön luokittelua SAR-kuvista on tutkittu paljon ja vaihteleviin kohteisiin. Näissä ei ole tutkittu kuvien suodatukseen käytetyn filtterin vaikutusta luokitteluun, mutta tutkimuksista

saa viitettä useimmin sovelletuista kohinanpoistomenetelmistä maankäytön luokittelun ja anomaalioiden tutkimisessa. Osassa tutkimuksista on myös esitetty perusteluita filtterin valintaan. Tähän katsaukseen etsittiin tutkimuksia, joissa oli hyödynnetty Sentinel-1-satelliittien SAR-kuvia ja pyritty joko tunnistamaan poikkeamia, tekemään kasvillisuuden tai maa-alueiden luokittelua koneoppimisen avulla.

Lasaponara ja Tucci (2019) tutkivat Italian eteläosissa tapahtuneen maastopaloalueen tunnistamista ja palon vakavuuden luokittelua Radar Burn Difference (RBD) ja Radar Burn Ratio (RBR) -lukuja avulla. BRD- ja RBR-luvuista tutkijat olivat muodostaneet kuvat, joista paloalueita voitiin tutkia ja luokitella ISODATA-menetelmällä. SAR-kuvien kohinanpoistoon Lasaponara ja Tucci (2019) olivat käyttäneet kahta menetelmää – yksittäisten SAR-kuvien suodatusta ja keskiarvoistamista useammasta SAR-kuvasta. Yksittäisen SAR-kuvan kohinanpoistoon Lasaponara ja Tucci (2019) olivat valinneet Lee-suodattimen, jonka he kertoivat olevan tunnettu ja löytyvän useimmista kaupallisista ja ilmaisista ohjelmista. Tutkijat havaitsivat, että paloalue näkyi selvästi havaittavana homogeenisena alueena kaikissa kuvissa. Selkeimmäksi eroksi mainittiin kohinan määrä yksittäisen SAR-kuvan ja useammasta kuvasta koostetun kuvan välillä, missä viimeksi mainitussa kohina oli selkeästi vähäisempää. Polarisaatioista paloalueen todettiin erottuvan paremmin VH-polarisaatiolla. Maastopaloalueiden tunnistamisen tarkkuudeksi tutkimuksessa oli saatu 91 % ja palon tasolle 89 %. (Lasaponara & Tucci 2019, 917–912.)

Vuonna 2021 Udali, Lingua ja Persson (2021) tutkivat puolestaan Ruotsin eteläosien puulajien ja metsätyypin luokittelua Random Forest -menetelmän avulla. SAR-kuvien esikäsittelyyn he olivat valinneet Sigma Lee -filtterin 5x5-ikkunalla ja sigman arvolla 0.9. Filtterin valintaa tutkijat perustelivat kirjallisuuskatsauksella, jonka mukaan Lee Sigma -filtteri säilyttää avaruudellisen resoluution sumentamatta reunoja ja pistemäisiä kohteita, ja sigman arvon 0.9 soveltuvan yleisille sovelluksille. Luokittelun tuloksina saatiin kokonaistarkkuudeksi 0.94 ja Kappa-arvoksi 0.86. (Udali ym. 2021.) Suurempien maa-alueiden luokittelua SMV-algoritmeilla tutkineet Orliková ja Horák (2019) puolestaan ilmoittivat valinneensa filtteriksi Refined Leen 3*3-ikkunalla, perustellen valintaa reunojen säilymisellä kohinan suodatuksessa.

Edellisten lisäksi muissa vastaavissa maa-alueiden luokittelua koskevissa artikkeleissa esikäsittelyyn valittuja filttäreitä olivat Enhanced Lee ja Gamma Map. Useimmiten valittuja filttäreitä olivat

Refined Lee ja Gamma MAP, mutta näiden, kuten muidenkaan filttereiden valintaa kaikissa tutkimuksissa ei oltu perusteltu. Dimov, Löw, Ibrakhimov, Stulina ja Conrad (2017) olivat tutkineet SAR-datan hyödyntämistä peltojen luokitteluun verrattuna optisen datan käyttöön. SAR-datan suodattamiseen Dimov ja muut (2017, 812) olivat valinneet Gamma MAP -filtterin VV-polarisaarion Sentinel-1 SAR-datalle Ozdarici ja Akyurek (2010) tutkimuksen perusteella. Dimov ja muiden (2017, 813) tutkimuksen mielenkiintoisena tuloksena oli, että sadon luokittelu onnistui paremmin filttämättömällä datalla. (Mercier, Betbeder, Rumiano, Baudry, Gond, Blanc, Bourgoïn, Cornu, Ciudad, Marchamalo, Pocard-Chapuis & Hubert-Moy 2019, 6; Prudente, Sanches, Adami, Skakun, Oldoni, Xaud, Xaud & Zhang 2021, 4101; Abdikan, Sanli, Ustuner & Calò 2016, 785.)

6.7 Kohinanpoistomenetelmien tutkimukseen liittyviä havaintoja

Vaikka luvussa 6.3 esitellyistä kohinanpoistomenetelmistä on tehty erilaisia vertailevia tutkimuksia vielä viime vuosiin saakka, voinee menetelmistä todeta, että ongelmana on enää sopivan filtlerin valitseminen kulloinkin kyseessä olevalle sovellukselle. Kuten Lee ja muut (1994) ja vuosia myöhemmin myös Parhad ja muut (2021) tutkimuksissaan totesivat, ei filtereistä voi nostaa yhtä ylitse muiden ja filtlerin valinta tulisi tehdä kyseessä olevan sovelluksen pohjalta. Perinteisten menetelmien vertaileva tutkimus vielä viime aikoina johtuneekin lisääntyneestä avoimen datan määrästä ja sitä myöten mahdollisten sovelluskohteiden lisääntymisestä. ESA:n tarjoamat GRD-kuvat ovat muihin alemman tason kuviin verrattuna helppoja ja kevyitä käsitellä, ja tarjoavat siten hyvän lähtökohdan alustaville tutkimuksille eri sovelluskohteista tai kun käsiteltävää dataa on paljon. Siten näiden menetelmien vertailulla tiettyjen erikoistapauksien näkökulmasta on edelleen paikkansa.

Tutkimusten suurta määrää selittänee myös kohinanpoiston arviointiin liittyvät haasteet. Verrattuna optisten kuvien kohinanpoiston tutkimukseen, SAR-kuvista ei ole valikoitunut muutamia yleisesti käytettäviä referenssikuvia, kuten esimerkiksi Lena tai Cameraman, joilla suodatuksen määrää voitaisiin vertailla eri tutkimusten välillä. Tähän on mahdollisesti vaikuttanut avoimen datan puute, jota olisi tarpeeksi ja erilaisiin käyttötarkoituksiin sopivaa. Hyvän datalähteen löydyttyä tämän voisi kuitenkin olettaa vakiintuvan tutkimusaineistoksi. Vastaavasti suodatetun kuvan laadun vertailemiseen käytetään vaihtelevasti erilaisia tunnuslukuja. Tehdystä vertailusta voi kuitenkin huomata, että Lee-filtteri on muodostunut eräänlaiseksi mittariksi, johon uusia menetelmiä peilaetaan.

Sen lisäksi, että tutkimuksissa käytetään vaihtelevasti erilaisia tunnuslukuja, ovat muutamat näistä ongelmallisia. Kuten osa suodattimista, osa tunnusluvuista on alun perin tarkoitettu kuvanlaadun arviointiin, kun kohinan määrää kuvassa voidaan verrata alkuperäiseen, kohinattomaan kuvaan. Saman tunnusluvun käyttö SAR-kuvien tapauksessa vaatii siten huolellista tulkintaa, sillä esimerkiksi MSE- ja PNSR-luvut voivat näyttää hyvää tulosta suodattimelle, joka ei todellisuudessa suodata kohinaa. Esimerkkinä Ansari ja muut (2021) ovat tutkimuksessaan nostaneet Boxcar-menetelmän vertailtujen menetelmien parhaimmaksi MSE- ja PNSR-arvojen perusteella, mutta alkuperäisenä kuvana on käytetty kohinallista SAR-kuvaa ja vertailtavana kohteena suodatettua kuvaa. Lisäksi Horé ja Ziou (2010, 2366) nostavat esille, että MSE- ja PNSR-mittareiden käytöstä on saatu ristiriitaisia tuloksia, sillä erilaiset muutokset samaan kuvaan saattavat antaa saman MSE:n ja edelleen PSNR-arvon. Ongelmaan on kuitenkin pyritty vastaamaan kehittämällä mittareita, jotka eivät vaadi kohinatonta kuvaa, ja näiden yleistymisen voi ajatella tekevän aiheen tutkimuksesta luotettavampaa.

Kuten Fracastoro ja muut (2021) syväoppimiseen perustuvista menetelmistä tekemässään tutkimuksessa totesivat, on tutkimuksissa käytetty data monenkirjavaa ja toteutuksen kuvaus usein puutteellista. Erityisen ongelmallista tämä on SAR-kuvien tapauksessa, sillä kuten luvusta 4 voi huomata, voi samasta datasta olla saatavilla eri tasoisia kuvia ja näitä voidaan käsitellä vaihtoehtoisilla tavoilla jo ennen kohinan poistamista, mikä edelleen vaikuttaa lopputulokseen. Tämä hankaloittaa tutkimusten vertailua ja nakertaa niiden luotettavuutta. Tutkimusten luotettavuuden ja toistettavuuden ohella tarkka dokumentointi voisi edesauttaa menetelmien yleistymisessä. Vaikka uusilla menetelmillä on saatu lupaavia tuloksia, ne eivät näytä yleistyneen käyttöön luvussa 6.3 esiteltyjen kohinanpoistomenetelmien tapaan. Aiheeseen perehtymättömän näkökulmasta uutta menetelmää voi olla hankala ottaa osaksi sovelluksia, jos se on vaikeaselkoinen ja heikosti dokumentoitu.

Kohinanpoistamiseen syväoppimisen avulla on herätty vasta viime vuosina, minkä vuoksi näistä löytyvät tutkimukset ovat pääasiassa menetelmien kehittäjien itsensä kirjoittamia. Valitettavasti monen menetelmän vertailua estää ongelmat niiden toistettavuudesta, minkä takia näistä ei voine odottaa lisätutkimuksia. Syväoppimisen ohella aiheen tutkimuksen suuntausta voi tulevaisuudessa värittää aikaisemmin kehitettyjen, täysin PolSAR-datalle suunniteltujen, menetelmien käyttö kohinan korrelaation käsittelyn jälkeen. Moni näistä menetelmistä on pohjautunut siihen

olettamukseen, että kohina on tilastollisesti riippumatonta (Lapini ym. 2014, 1044). Näin ei kuitenkaan Lapini ja muiden (2014) mukaan ole, mutta data voidaan käsitellä ennen kohinan suodatusta, minkä ansiosta useilla menetelmillä on saatu aikaisempia parempia tuloksia. Mahdollisesti myös Laine ja muiden (2019) ja Molini ja muiden (2020) pistetodennäköisyyden esitykset kohinalliselle ja kohinattomille pikseliarvoille saattavat innoittaa tutkimaan asiaa uudessa valossa.

7 Kohinanpoistamisen vaikutuksesta SAR-kuvien luokitteluun koneoppimisen avulla

Tässä kuvussa tarkastellaan SAR-kuvien luokittelua ja luokitteluun yleisesti käytettyjä menetelmiä kohinanpoistamisen näkökulmasta. Luvun tarkoituksena ei ole tehdä tyhjentävää selvitystä käytetyistä luokittelumenetelmistä tai käydä luokittelumenetelmien toimintaa yksityiskohtaisesti läpi, vaan antaa yleiskuva yleisimmistä käyttötapauksista SAR-kuvien luokittelussa ja tuoda esiin SAR-kuvien kohinanpoistamisen mahdollista vaikutusta luokittelumenetelmien tuottamaan tulokseen. Tästä johtuen seuraaviin lukuihin on nostettu esiin vain muutamia yleisiä ja toisistaan selkeästi poikkeavia luokittelutapauksia ja -menetelmiä.

7.1 SAR-kuvien luokitteluongelmat

SAR-kuvien luokitteluongelmissa voi havaita kaksi päähaaraa – yksittäisten kappaleiden tunnistaminen kuvista ja kuvan pikseleiden luokittelu yhtenäisiin alueisiin. Jälkimmäisessä tapauksessa puhutaan usein myös kuvan segmentoinnista (eng. semantic segmentation, myös image segmentation). Yksittäiset kappaleet, joiden tunnistamista SAR-kuvista on tutkittu, ovat esimerkiksi laivat ja lentokoneet. Kuvien segmentointi puolestaan voi esimerkiksi käsitellä satojen, jään, metsäalueiden, palaneiden alueiden tai infrastruktuurin tunnistamiseen kuvista.

Useissa tutkimuksissa kappaleiden tunnistamiseen sovelletaan konvoluutioneuroverkkoja, jotka ovat osoittautuneet tehokkaiksi kuvien luokitteluongelmissa. Kuvien segmentointiin puolestaan näkee käytettävän useita erilaisia menetelmiä, mukaan lukien konvoluutioneuroverkkoja. Koska konvoluutioneuroverkkojen perusteita ja kuvantunnistamisen periaatetta konvoluutioneuroverkoilla on käsitelty luvussa 5, esitellään seuraavassa vain SAR-kuvien segmentointi. Luvussa 5 esitettyjen tietojen pohjalta voi kuitenkin todeta, että pikseliarvojen muuttuminen reunojen läheisyydessä ja edelleen muodon muuttuminen voi vaikuttaa objektin tunnistamiseen, erityisesti jos

kuvista tunnistettavat kappaleet muistuttavat toisiaan. Toisaalta myös reuna-alueilla esiintyvä kohina voi vaikeuttaa muodon tunnistamista, mikä puolestaan voi aiheuttaa luokittelun epäonnistumisen.

7.2 SAR-kuvien segmentointi

SAR-kuvien segmentointiin on useita eri mahdollisuuksia. Satelliittikuvien käsittelyyn tarkoitettua SNAP-ohjelmasta alueiden luokitteluun löytyy sekä valvottuja että valvomattomia menetelmiä, kuten Random Forest, KNN, K-Means, Maximum Likelihood ja K-Means Cluster Analysis. Googlen hie-man vastaavasta Google Earth Engine -alustalta puolestaan löytyvät Random Forest, CART, Naive-Bayes ja Support Vector Machines -menetelmät. Näiden lisäksi kuvien segmentointia on mahdollista tehdä neuroverkkojen avulla, kuten Mathis Lamarre (n.d) kertoo ESA:n verkkosivuilla julkaistussa artikkelissa *Spatiotemporal crop type classification with Deep Learning applied to SAR time series*.

Luokitteluun valittava menetelmä riippuneee käyttötarkoituksesta ja tilanteesta. Useissa koulutusmateriaaleissa, kuten NASA:n ja ESA:n RUS Copernicus -alustan oppaissa maa-alueiden luokittelua on ohjeistettu Random Forest -menetelmällä. (Podest, McCullum, Torres-Perez & McCartney 2020; Gomez 2018). Random Forest -menetelmän käyttöön ohjeistaa myös Braun (2020) ESA:n Sentinel-1 Toolbox – SAR-based landcover classification with Sentinel-1 GRD products -oppaassa, minkä lisäksi oppaassa on käsitelty K-Means Cluster Analysis -luokittelun käyttö maa-alueiden segmentoinnissa.

SNAP-ohjelmasta ja Google Earth Engine -alustalta valmiiksi ohjelmoituna löytyvät luokittelumenetelmät lienevät aiheeseen tutustuville helppoja tapoja tehdä kuvien segmentointia, sillä menetelmät eivät vaadi edellistä osaamista neuroverkoista tai niiden luomista. Tämän lisäksi luokittelua on mahdollista tehdä pienellä datamäärällä, kun taas neuroverkot voivat tarvita enemmän dataa luotettavien tulosten saamiseen. Tutkimuksen uutena suuntauksena ja tarkempaa analyysiä vaativissa sovelluksissa konvoluutioneuroverkot mahdollisesti nähdään myös parempina luokittelumenetelminä. Koska kuvien segmentointiin SAR-kuvien tapauksessa yleisesti ohjeistetaan Random Forest -menetelmän käyttöä, esitellään menetelmän lyhyesti seuraavassa kappaleessa.

Random Forest -menetelmässä luokittelu toteutetaan useamman päätöspuun avulla, millä vältetään yksittäisten päätöspuiden kanssa ongelmana oleva mallin herkyys koulutusdatan muutoksille. Yksittäisessä päätöspuussa tehdään luokiteltavan asian piirteiden vaiheittaista vertailua puun solmukohdissa, joissa piirteen arvoa verrataan solmun ehtoon. Muutos piirteen arvossa voi siis saada puun luokittelemaan objektin toiseen luokkaan riippuen muutoksen suunnasta ja suuruudesta. Random Forest -luokittelussa tulokseksi katsotaan luokka, jonka päätöspuiden enemmistö on ennustanut syötteelle. Tästä johtuen muutos koulutusdatassa ja siten yksittäisen päätöspuun ennusteessa ei välttämättä aiheuta muutosta mallin antamaan ennusteeseen. Vaikka erilliset päätöspuut eivät pystyisi antamaan tarkkaa ennustetta, useamman päätöspuun joukko voi silti olla hyvä ennustaja ja yleensä tarkkuus paranee, kun puiden lukumäärää kasvatetaan. (Géron 2017, 177–191.)

Satunnaisuus menetelmän nimessä viittaa yksittäisten päätöspuiden koulutuksessa käytettyihin ennustavien piirteiden osajoukkoihin, jotka valitaan kullekin päätöspuulle satunnaisesti. Valitsemalla koulutukseen vain osa piirteistä, päätöspuista saadaan erilaisia ja niiden tuottamat ennusteet perustuvat luokiteltavan asian eri piirteisiin. Tämän ansiosta Random Forest -menetelmässä saadaan myös tietoa tärkeimmistä ennustavista piirteistä sen lisäksi, että syötteen luokaksi saadaan useampia ennusteita. (Géron 2017, 185–191; Kirk 2017, 74–75.)

Luokittelua tehtäessä ennusteiden lisäksi ollaan kiinnostuneita mallin hyvyydestä ja luokittelun onnistumisesta. Kuten muitakin luokittelumalleja arvioitaessa, tärkeitä mittareita ovat sekaannusmatriisi (confusion matrix), virheettömyys (accuracy) ja tarkkuus (precision). Sekaannusmatriisissa ilmoitetaan oikein ja väärin menneiden luokittelujen lukumäärät vaihtoehdoittain: oikeat positiiviset (TruePositives), väärät positiiviset (FalseNegatives), oikeat negatiiviset (TrueNegatives) ja väärät negatiiviset (FalseNegatives). Sekaannusmatriisin arvoista lasketaan luokittelun virheettömyys tietylle luokalle oikein menneiden ennusteiden suhtena kaikkiin luokan alkioihin. Tarkkuus puolestaan kertoo oikeiden positiivisten suhteen kaikkiin positiivisiksi luokiteltuihin alkioihin. (Géron 2017, 83–84.) Muita luokittelun arvioimiseen käytettyjä mittareita voivat ovat Recall-arvo, RMSE ja virhe (bias).

Kun Random Forest -menetelmää halutaan hyödyntää maa-alueiden luokitteluun SAR-datassa, rajataan SAR-kuvasta erilaisia alueita, jotka kuuluvat määriteltyihin luokkiin ja alueiden sisältämät pikselit toimivat koulutusdatana. Jos alueilla tapahtuvia muutoksia halutaan tutkia aikasarjana,

Random Forest -luokittelu voidaan toteuttaa useammille SAR-kuville ja vertailla näitä keskenään. (Braun 2020; Gomez 2018.) Vaikka periaatteeltaan kuvien segmentointiin käytetyt menetelmät voivat poiketa toisistaan paljonkin, voinee kohinanpoistamisen sanoa vaikuttavan luokitteluun epäsuotuisasti, jos menetelmässä tehdään vertailua yksittäisten lukuarvojen kautta.

8 Toteutuksen suunnitelma

8.1 Vertailuun valitut kohinanpoistomenetelmät

Kohinanpoistomenetelmistä vertailuun valittiin suodattimet Lee, Frost ja Refined Lee, joista Lee- ja Frost-suodattimista käytettiin filtteri-ikkunoiden kokoja 5x5 ja 7x7. Näiden lisäksi testattiin Refined Lee -filtterin soveltamista SAR-dataan kahdesti sekä Refined Lee- ja Frost -filttereiden yhdistelmää Frost-filtterin ikkunako'illa 5x5 ja 7x7. Frost-filtterin vaimennuskertoimen arvo päätettiin pitää oletusarvossa 2, jolla suodattimen tulisi tasoittaa enemmän kohinaa kuin suuremmilla vaimennuskertoimen arvoilla. Perinteisten suodattimien lisäksi tutkimuksessa oli tavoitteena testata Speckle2Void-menetelmän toimivuutta pilottitapauksessa käytetyille Level-1-tason GRD-kuville. Speckle2Void-menetelmään ja SAR-dataan perehdyttäessä kuitenkin selvisi, ettei malli ole sellaiseenaan käytettävissä Sentinel-1 GRD-kuville. Tästä johtuen menetelmän tarkastelu on jätetty teoriatasolle.

Tutkimuksessa vertailtavien suodattimien valinta tehtiin anomalioiden tunnistamiseen mahdollisesti käytettävien luokittelumenetelmien näkökulmasta ja edeltävien tutkimustulosten perusteella. Koska Suomessa laajat maastopalot ovat harvinaisia ja tulvien sijainnit ennakoitavissa, ajateltiin todennäköisimpänä anomaliana myrskyjen aiheuttamia metsätuhoja. Muutosten havaitsemiselle oleellisina asioina voidaan pitää reunojen erottuvuutta ja homogeenisten alueiden tasoittumista. Molemmat helpottavat sekä visuaalista tarkastelua että alueiden tunnistamista koneoppimisen avulla. Reunojen terävöitymisestä on hyötyä koneoppimisen näkökulmasta silloin, kun tutkitaan muutoksia esimerkiksi metsä- tai jokialueiden reunoissa. Toisaalta jos kuvasta halutaan luokitella yhtenäisiä alueita, hyödytään lähekkäisten pikseliarvojen samankaltaisuudesta ja tällöin homogeenisten alueiden tasoittuminen voisi saada muutosalueet paremmin erottuviksi. Koska kuvia ei pilottitapauksessa käytetä tietyn muotoisten kappaleiden tunnistamiseen, muotojen säilymistä ei tässä tutkimuksessa katsota prioriteetiksi.

Kohinanpoistomenetelmiä vertailevissa tutkimuksissa Refined Lee on noussut usein suositelluksi vaihtoehdoksi, kun kuvista halutaan terävöittää kappaleiden reunoja ja muokata kuvia paremmiksi visuaaliseen tarkasteluun. Refined Lee oli myös toinen parhaiten toimivista suodattimista Ananto ja muiden (2019) tutkimuksessa, jossa filttäreitä oli vertailtu biomassan arvioimisen näkökulmasta. Ananto ja muut (2019) totesivat tutkimuksen tuloksina, että filttäreistä Median, Frost, Gamma

MAP ja Refined Lee parhaiten VH-polarisaation kanssa toimi Frost ja seuraavaksi Refined Lee, kun takaisinsironnan ja kenttätutkimuksessa kerättyjen arvojen vastaavuutta tutkittiin lineaarisen regression kautta. VV-polarisaation kanssa Refined Lee toimi heikoiten, mutta kasvillisuuden tutkimiseen VH-polarisaatiota voidaan pitää oleellisempänä.

Maa-alueiden luokittelua koskevilla tutkimuksilla filttäreiden valinta ja perustelut filtteriin valinnalle ovat olleet kirjavia. Osassa tutkimuksista menetelmän valintaa on perusteltu kirjallisuuskatsauksilla, osassa lyhyesti toteamalla, että käytettävä filtteri on riittävä tutkimuksen tarkoituksiin tai menetelmän valintaa ei ole perusteltu. Näistä tutkimuksista mainittavina menetelminä voi nostaa Sigma Lee- ja Gamma MAP -suodattimet, joiden käytölle ei kuitenkaan tämän opinnäytetyön tutkimuskatsauksessa ole löydetty suurempaa tukea. Sigma Lee -suodattimen käytöstä on todettu, että sen käyttö ei sovellu SAR-kuville, sillä kohina ei vastoin menetelmän oletuksia ole normaali jakautunut. Gamma MAP -suodatin puolestaan ei useammassa tutkimuksessa ole noussut parhaimpien joukkoon reunojen erottelukyvyn tai homogeenisten alueiden tasoittumisen näkökulmasta.

Myöskään perinteistä Lee-filtteriä harvemmin mainitaan soveltuvaksi menetelmäksi SAR-kuville. Tutkimuskatsauksen perusteella voi kuitenkin todeta, että Lee-filtteristä on muodostunut eräänlainen mittari uusien filttäreiden suoriutumiseksi, minkä vuoksi tämä otettiin mukaan vertailukohdaksi. Edellisten tutkimusten perusteella voi olettaa, että Lee tuottaa vertailtavista menetelmistä huonoimmat tulokset etenkin reunojen erottuvuuden osalta. Toisaalta filtteriin kyky tasoittaa homogeenisiä alueita voi tuoda yllättäviä näkökulmia yhtenäisten alueiden erottuvuuteen. Koska edellisten tutkimusten pohjalta on vahvoja viitteitä Lee-filtterin tuloksista suhteessa menetelmiin Frost ja Refined Lee, voidaan tutkimuksen luotettavuutta myös arvioida peilaamalla filttäreiden keskinäistä paremmuutta aikaisempiin tuloksiin ja katsoa noudattavatko nyt tehtävän tutkimuksen tulokset samaa linjaa aikaisempien kanssa.

Yksittäisten menetelmien lisäksi tutkimuksissa on tuotu esiin mahdollisuus soveltaa filttäreitä useampaan kertaan ja yhdistellä eri filttäreitä toivotun lopputuloksen aikaan saamiseksi. Koska maa-alueiden luokittelussa ja muutosalueiden tunnistamisessa usein käytetty menetelmä on pikselien luokittelu Random Forest -menetelmällä, päätettiin tuloksia arvioida erityisesti tämän menetelmän tavoitteiden näkökulmasta. Näin ollen kuviin halutaan kohtalaisen selkeärajaisia ja yhtenäisille alueille toisiaan lähellä olevia pikseliarvoja. Tähän tavoitteeseen arvioitiin mahdollisesti

päästävän Refined Lee ja Frost-filttereiden yhdistelmällä, jossa molempien filttäreiden oletettiin säilyttävän kohteiden reunat, mutta Frost-filtterin tasoittavan enemmän kohinaa homogeenisilla alueilla. Koska filttäreissä käytettävillä parametreillä on myös vaikutusta filtterin toimintaan, vertailussa huomioitiin myös filtterin toiminta kahdella eri ikkunan koolla.

Lee, Frost, Refined Lee sekä Refined Leen ja Frost -filtterin yhdistelmän lisäksi vertailuun oli tarkoitus ottaa neuroverkoilla kehitetty Speckle2Void, joka nähtiin lupaavaksi menetelmäksi kohinan poistamiseen. SAR-dataan ja Speckle2Void-menetelmään perehdyttäessä kuitenkin paljastui, ettei menetelmää voi suoraan käyttää pilottitapauksen datalla ja sen käyttö olisi vaatinut muutoksia, joiden ei enää katsottu olevan tämän opinnäytetyön rajoissa. Tästä johtuen Speckle2Void-menetelmän tutkimus esitetään vain teoriatasolla.

8.2 Menetelmien vertailu

Kohinanpoistomenetelmiä ei voitu vertailla yksittäisten kappaleiden luokittelun kautta, sillä datasetin kokoaminen tähän tarkoitukseen olisi ollut opinnäytetyön laajuuteen nähden liian työläs. Avoimena datana löytyy SAR-kuvia sisältäviä datasettejä, mutta näissä datan edeltävästä käsitte-lystä ei ole dokumentaatiota ja näiden käyttö ei olisi lisännyt tietoa maastoalueille soveltuvista filttäreistä. Tästä johtuen menetelmien vertailuun päätettiin valita Random Forest, jota usein myös näkee ohjeistettavan maa-alueiden luokitteluun SAR GRD -kuvista. Random Forest -luokittelulla eri menetelmien vaikutuksista saataisiin myös vertailukelpoisia lukuja.

Random Forest -luokittelun toteutuksesta tässä tutkimuksessa tulee kuitenkin huomauttaa, että maa-alueiden luotettava luokittelu vaatisi useita parannuksia. Ensinnäkin maa-alueiden nimikointi edellyttäisi alueen tarkastamisen, jotta koulutusdataan ei päätyisi väärin luokkiin osoitettuja alueita. Toisekseen luokitteluun ei voitaisi käyttää vain kahta kuvaa, sillä vuodenaikojen vaihtelu ja sää vaikuttavat takaisinsirontaan, ja siten kuvissa näkyvä tutkavaste vaihtelee näiden mukaan. Alueilta eri vuodenaikoina takaisin tulevaa intensiteettiä pitäisi myös tutkia ennen luokittelua, millaisten alueiden erottaminen toisistaan voi ylipäätään olla mahdollista. Todennäköisesti useammalla-kaan kuvalla eri vuoden aikoina ei vielä saataisi hyviä tuloksia, sillä parempaan erottelukykyy-vaadittaisiin kaikki neljä polarisaatiota. Luokittelua varten tulisi myös tutkia millaisia ennustavia piirteitä datasta voidaan koostaa nyt käytettyjen lisäksi. Kohinanpoistomenetelmien vertailuun käytettävissä olevan datan ja nimikointien katsottiin kuitenkin olevan riittäviä.

Random Forest -luokittelun lisäksi kohinanpoistofilttereiden toimintaa haluttiin tutkia subjektiivisesti tarkastelemalla homogeenisten alueiden tasoittumista, muotojen säilymistä ja rajojen terävyyttä. Lisäksi homogeenisilta alueilta päätettiin mitata ENL-arvo, jonka avulla voitaisiin mahdollisesti verrata tuloksia aikaisempiin tuloksiin. Laskenta-ajan vertailu otettiin mukaan datasetin koon ja käsittelyajan kertaantumisen näkökulmasta sen prosessoinnissa, vaikka yksittäisen kuvan tapauksessa käsittelyajoissa ei voi olettaa olevan merkittäviä nykylaitteilla eroja.

8.3 Tutkimuksessa käytetyt teknologiat

SAR-kuvien käsittelyyn ja kohinanpoistomenetelmien vertailuun tarvittiin useampia teknologioita, joista eniten käytetyt esitellään lyhyesti seuraavissa kappaleissa. Kustakin menetelmästä esitetään teknologian pääpiirteet, perustelut valinnalle ja mihin tätä tutkimuksessa käytettiin.

SNAP ja Snappy

SNAP (Sentinel Application Platform) on Sentinel-satelliittien käsittelyyn kehitetty ohjelma, jolla voi sekä tehdä datan esikäsittelyyn tarvittavat toimenpiteet että analysoida dataa. Ohjelman asennuksessa valitaan käsiteltävän datan mukainen lisäpaketti, joista tässä opinnäytetyössä käytettiin Sentinel1 Toolboxia (S1TBX). SNAP on Java-pohjainen ohjelma, mutta sen käyttöön on kehitetty myös Python-moduuli snappy, jolla lähes kaikki SNAP-ohjelman toiminnot voi toteuttaa Python-ohjelmointikielellä.

Jupyter Notebook

Jupyter Notebook on selainpohjainen ohjelmointiympäristö, jonka etuja perinteisempiin ohjelmiin verrattuna ovat selkeä ulkoasu ja mahdollisuus tehdä dokumentointia siististi samaan tiedostoon. Jupyter Notebook valittiinkin käytettäväksi teknologiaksi tulosten visualisointia ajatellen. Osaltaan teknologian valintaan vaikutti myös se, että useissa SAR-kuvien käsittelyä koskevissa oppaissa ohjelmointi huomattiin tehdyn Jupyter Notebookin avulla ja Notebookin käyttö mahdollistaisi myöhemmin muiden aiheeseen liittyvien alustojen, kuten Google Earth Enginen, testaamisen SAR-kuvien käsittelyssä.

QGIS

Tutkimuksessa tehtävään Random Forest -luokittelun koulutukseen tarvitaan luokkia kuvaavia alueita, jotka on mahdollista määrittää SNAP-ohjelmassa luokittelua tehtäessä. Alueiden määrittely tehtäisiin kuitenkin SAR-kuvasta, josta etenkin maastoalueilla voi olla hankala tietää, millainen alue kuvassa oikeastaan näkyy. Jotta koulutukseen käytettävien alueiden määrittäminen olisi luotettavampaa, käytettiin tähän karttojen tekemiseen kehitettyä QGIS-ohjelmaa, jossa voi tarkastella tutkittavaa kohdetta muun muassa tie-, satelliitti- ja maastokarttatasoina.

9 Tutkimuksen toteutus

9.1 Tutkimusaineiston kerääminen

Kohinanpoistomenetelmien vertailussa hyödynnettiin pilottitapauksessa käytettyjä Sentinel-1A- ja Sentinel-1B-satelliittien IW-moodin Level-1-tason GRD-kuvia, jotka oli pilottitapauksen alussa laaduttu projektin käytössä olevalle palvelimelle. Kyseiset kuvat ovat korkean resoluution kuvia, jotka sisältävät polarisaatiot VH ja VV. Kuvat käsittävät alueita Keski-Suomesta vuodelta 2020 ja Koillismaahan, Oulunkaaren ja Oulun seutukunnan alueet, joilla myrskytuhoja esiintyi kesällä 2021.

Random Forest -luokittelun coregistration-vaihetta varten tarvittiin useampia täsmälleen samasta kohdasta ja samasta suunnasta otettuja satelliittikuvia. Tähän tarkoitukseen sopivia kuvia tietokannan kuvista löytyi kaksi mahdollista tiedostoa. Ensimmäinen kuva saatiin kesäkuun alusta ja myrskyn jälkeinen kuva heinäkuun alusta. Kuviossa 6 tiedostojen sisältämät polarisaatiot ovat sekä amplitudi- että intensiteettiarvoina Bands-kansiossa. Muita kuviin liittyviä tietoja voi tarkastella metadatatista.

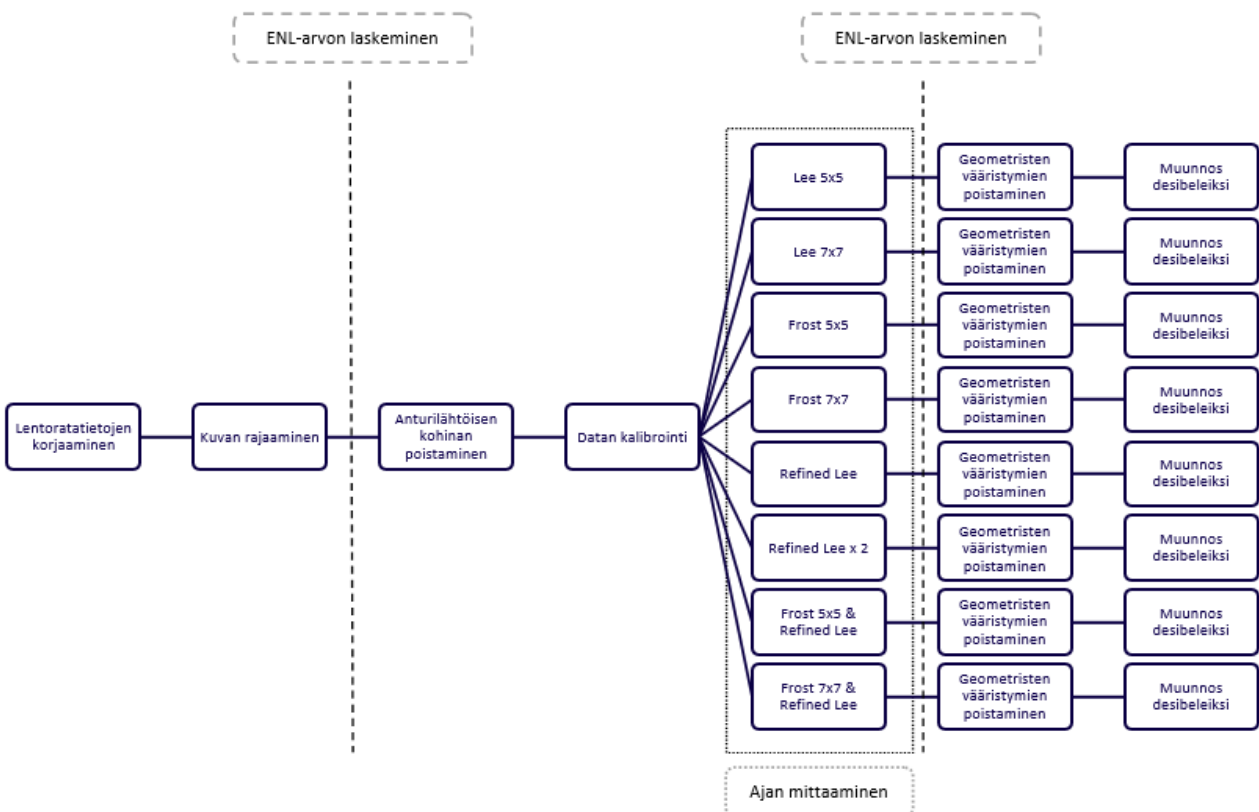
The screenshot shows the SNAP software interface. On the left, the Product Explorer displays a tree view of metadata for two Sentinel-1A GRD products. The 'Abstracted_Metadata' folder is expanded, showing various metadata fields. On the right, the 'Abstracted_Metadata' table is displayed, listing metadata fields, their values, types, units, and descriptions.

| Name | Value | Type | Unit | Description |
|------------------------------|-----------------------------------|---------|------|---------------------------------|
| PRODUCT_TYPE | GRD | ascii | | Product type |
| SPH_DESCRIPTOR | Sentinel-1 IW Level-1 GRD Product | ascii | | Description |
| MISSION | SENTINEL-1A | ascii | | Satellite mission |
| ACQUISITION_MODE | IW | ascii | | Acquisition mode |
| antenna_pointing | right | ascii | | Right or left facing |
| BEAMS | - | ascii | | Beams used |
| SWATH | - | ascii | | Swath name |
| PROC_TIME | 02-JUN-2021 16:56:04.183524 | uint32 | utc | Processed time |
| Processing_system_identifier | ESA Sentinel-1 PF 003.31 | ascii | | Processing system identifier |
| orbit_cycle | 232 | int32 | | Cycle |
| REL_ORBIT | 116 | int32 | | Track |
| ABS_ORBIT | 38163 | int32 | | Orbit |
| STATE_VECTOR_TIME | 02-JUN-2021 15:32:39.000000 | uint32 | utc | Time of orbit state vector |
| VECTOR_SOURCE | - | ascii | | State vector source |
| incidence_near | 30.23844814300537 | float64 | deg | |
| incidence_far | 45.680681228637695 | float64 | deg | |
| slice_num | 5 | int32 | | Slice number |
| data_take_id | 295183 | int32 | | Data take identifier |
| first_line_time | 02-JUN-2021 15:33:43.339436 | uint32 | utc | First zero doppler azimuth time |
| last_line_time | 02-JUN-2021 15:34:08.338326 | uint32 | utc | Last zero doppler azimuth time |
| first_near_lat | 64.42060089111328 | float64 | deg | |
| first_near_long | 27.359703459122024 | float64 | deg | |
| first_far_lat | 64.87431335449219 | float64 | deg | |
| first_far_long | 32.617645383258086 | float64 | deg | |
| last_near_lat | 65.89249420166016 | float64 | deg | |
| last_near_long | 26.47640189118108 | float64 | deg | |
| last_far_lat | 66.36045837402344 | float64 | deg | |
| last_far_long | 32.03244446768759 | float64 | deg | |
| PASS | ASCENDING | ascii | | ASCENDING or DESCENDING |
| SAMPLE_TYPE | DETECTED | ascii | | DETECTED or COMPLEX |

Kuvio 6. GRD-kuvat sisältöineen SNAP-ohjelmassa.

9.2 Yksittäisen SAR-kuvan esikäsittely ja alueiden vertailu

Yksittäisen SAR-kuvan esikäsittely tehtiin luvussa 4.3 esitetyn käsittelyketjun mukaan siten, että vaiheista toteutettiin muut paitsi kuvan multilooking-prosessointi. Multilooking jätettiin käsittelyketjusta pois, sillä Level-1-datan kuvauksen mukaan tämä on jo kerran tehty. Esikäsittelyyn tutustuttiin ensin SNAP-ohjelmalla, kun kuvista etsittiin sopivia vertailukohtia ja tutkittiin kohinanpoiston vaikutusta erilaisille alueille. Myöhemmin samat vaiheet toteutettiin ohjelmallisesti snappy-moduulilla kirjoittamalla esikäsittelyn vaiheet omiksi funktioikseen, jotta prosessointi olisi mahdollista tehdä nopeasti uudelleen ja jotta eri menetelmien vertailua voitaisiin tehdä myös ajallisesti. Yksittäisen kuvan esikäsittelyyn valittiin SAR-kuva heinäkuun alusta. Kuviossa 7 on tiivistettynä esitetty kuvalle tehty esikäsittely ja kappaleissa kuvattu esikäsittelyn vaiheista tarkemmat tiedot. Kaikki kuviossa 7 esitetyt vaiheet tehtiin molemmille polarisaatioille VH ja VV. Kuvien esikäsittelyssä käytetyt koodit löytyvät liitteestä 5, ENL-arvojen mittaaminen liitteestä 6 ja esikäsittelyyn kirjoitetut funktiot liitteestä 7.



Kuvio 7: Yksittäisen kuvan käsittely, ENL-arvojen laskeminen ja käsittelyyn kuluneen ajan mittaaminen.

Lentoratatietojen korjaus

SNAP-ohjelmassa lentorataa koskevien tietojen korjaus onnistuu Apply Orbit File -toiminnolla, jonka tulisi ladata lentoratatiedostot automaattisesti ja korjata tiedot näiden mukaan. Vuoden 2021 kuvilla tämä ei kuitenkaan onnistunut, eikä vian tarkkaa syytä ole tiedossa. Asiasta on keskustelua ESA:n Step-foorumilla, joskaan ongelmaan ei opinnäytetyön toteutus- tai kirjoitusvaiheessa ollut löytynyt ratkaisua. Lentoratatietojen korjauksen saa kuitenkin tehtyä lataamalla oikean ajanjakson lentoratatiedoston manuaalisesti Copernicus Open Access Hubista ja sijoittamalla sen SNAP:n RESORB-kansioon, josta ohjelma osaa hakea tiedostot ja tehdä lentorataa koskevat korjaukset. Sama ongelma koskee sekä graafista käyttöliittymää, että koodilla tehtävää käsittelyä.

Kuvan rajaaminen

Jotta esikäsittely ei olisi tarpeettoman raskasta, kuvasta on hyvä rajata tarkasteluun pienempi osa-alue. Vertailua oli aluksi tarkoitus tehdä muutamien pienempien osa-alueiden avulla Jupyter Notebookissa, mutta vertailua tehdessä kuitenkin todettiin, että kuvien tarkasteluun soveltuu huomattavasti paremmin SNAP-ohjelma. Tämän vuoksi esikäsittelyssä kuvasta rajattiin yksi iso alue ja prosessoinnin lopuksi käsitellyt kuvat tallennettiin BEAM-DIMAP-formaatissa, joka oli mahdollista avata SNAP-ohjelmassa tarkasteltavaksi. Heinäkuun alun kuvasta rajattiin osa-alue koordinaatteihin 27.432 65.338, 27.747 65.338, 27.747 65.268, 27.432 65.268, 27.432 65.338, missä tuhoalueita oletettiin esiintyvän.

Kohinan määrän arviointiin käytetyn ENL-arvon mittaaminen tehdään rajaamalla kuvasta homogeeninen alue, josta lasketaan ENL luvussa 6.2 esitetyn kaavan (11) mukaisesti. Koska kuvasta oli vaikea löytää suurta homogeenista aluetta, rajattiin kuvasta kaksi pienempää aluetta – toinen järven ja toinen metsän kohdalta. Ensimmäiset ENL-arvot laskettiin osa-alueiden rajaamisen jälkeen ja toiset kohinanpoistomenetelmien soveltamisen jälkeen.

Anturilähtöisen kohinan poistaminen ja datan kalibrointi

Vaikka anturilähtöisen kohinan poistaminen ei ole välttämätöntä, tämä päätettiin tehdä, jotta kohteesta johtuvan kohinan määrää kuvissa voitaisiin paremmin arvioida. Datan kalibroinnissa ja

esikäsittelyn myöhemmissä vaiheissa käytettiin Braunin (2020) ohjeistuksen mukaan kaistojen intensiteettiarvoja ja näistä laskettuja kalibroituja arvoja. Arvojen kalibrointi tehtiin molemmille intensiteettiarvoille VV ja VH. Kalibroinnin tuloksena ovat arvot Sigma0_VV ja Sigma0_VH , missä Sigma0 viittaa tutkan normalisoituun poikkileikkaukseen.

Kohinan poistaminen

Kohinan poistaminen snappy-moduulin avulla tehtiin kirjoittama Lee, Frost ja Refined Lee -suodatimille omat funktionsa. Samaa tai eri filttäreitä sovellettaessa useampaan kertaan samalle kuvalle, filtteriointi tehtiin ensin yhdellä menetelmällä, jonka ulostulolle suoritettiin filtteriointi uudelleen.

Geometrinen vääristymien poistaminen

Geometrinen vääristymien poistamisessa DEM-malliksi valittiin GETASSE30, sillä oletuksena SNAPissa oleva DEM-malli ei kata arktisia alueita. ESA:n Level-1 Pre-Processing Algorithms -kuvauksen mukaan GETASSE30 on yhdistelmä neljästä datasetistä, jotka yhdessä kattavat myös pohjoisimmat ja eteläisimmät alueet. Vaikka geometrinen vääristymien poistaminen tekee muutosten tarkastelusta ja kohteiden tunnistamisesta hankalaa verrattuna perinteisiin karttoihin, korjaus kuitenkin muuttaa kuvat oikein päin, mikä helpottaa kuvan tarkastelua.

Muunnos desibeleiksi

Prosessoinnin viimeisenä vaiheena tehtiin arvojen muuttaminen desibeleiksi, jotta intensiteettierojen havaitseminen subjektiivisesti olisi helpompaa.

9.3 Speckle2Void-mallin käyttö kohinan poistamiseen

Ennen Speckle2Void-mallin testaamista tieto sen soveltumisesta SAR-kuville oli perustunut aiheesta kirjoitettuun artikkeliin. Artikkelissa tutkimusaineistoksi oli kerrottu TerraSAR-X-satelliitin keräämä data, joka nimensä mukaisesti on tehnyt kuvantamista X-aallonpituudella. Tutkimuksen edetessä ja tietämyksen kasvaessa SAR-datasta, Speckle2Void-mallin repositoriosta ja sen sisältämistä koodeista huomattiin, että Speckle2Void-malli on kirjoitettu SLC-datalle, joka sisältää SAR-datan tiedot kompleksilukuina. Koska GRD- ja SLC-tuotteet poikkeavat tässä suhteessa oleellisesti

toisistaan, ei perinteisten kohinanpoistomenetelmien ja Speckle2Void-menetelmän vertailua voitu toteuttaa suunnitellulla tavalla. Speckle2Void-mallia tutkittiin kuitenkin koodia lukemalla ja tekemällä suppea testaus eri syötteillä.

Tutkimusartikkelissaan Molini ja muut (2022) ovat kertoneet kohinan korrelaatioiden käsittelyn olevan suodatuksen oleellinen vaihe. Vastoin pitkään vallalla ollutta käsitystä, SAR-kuvissa esiintyvä kohina ei ole pikseleittäin riippumatonta eikä täytä monissa kohinanpoistomalleissa tehtyä oletusta kohinan luonteesta. Tästä johtuen useimmat kohinanpoistomallit eivät Lapini, Banchi, Argenti ja Alparonen (2014, 1044) mukaan toimi tarkoitetulla tavalla, mutta ongelma on ratkaistavissa käsittelemällä data ennen kohinan poistamista. Lapini ja muiden (2014, 1044) mukaan myös aikaisemmin kehitetyillä malleilla on saatu parempia tuloksia, kun data on ensin esikäsitelty. Oletuksen täyttämiseksi Molini ja muut (2020) olivat muuttaneet TerraSAR-X SLC -kuvat kompleksilukuja sisältäviksi .mat-tiedostoiksi ja edelleen käsitelleet kohinan riippuvuuden Lapini ja muiden (2014) tutkimuksen perusteella. Esikäsitelyn tuloksena on saatu kompleksilukuja sisältäviä kuvia, joista on edelleen generoitu koulutus- ja testidataa rajaamalla kuvista pienempiä alueita.

Koska Sentinel-1A- ja Sentinel-1B-satelliittien GRD-kuvissa arvot ovat intensiteetti- ja amplitudiarvoja reaalityyppinä, ei GitHub-repositorioon jaettua koodia riippuvuuksien poistamiseksi voitu käyttää. Speckle2Void-mallin käyttämistä ilman kuvien esikäsitelyä pohdittiin ja suppeasti myös testattiin .mat- ja .tiff -tiedostotyypeillä, joiden voitiin repositorion DataGenerator.py-koodin mukaan olettaa sopivan mallille syötteeksi. Koska repositoriossa ei ollut tarkempaa tietoa oikeanlaisten Matlab-tiedostojen sisällöstä tai kuvien muuttamisesta .mat-tiedostoiksi, käytettiin alkuperäisinä kuvina GeoTiff-tiedostoja, joiden muuntamisen Matlab-tiedostoiksi tiedettiin olevan mahdollista. Kuvat muutettiin tiedostomuutoksen yhteydessä 10000x10000-kokoisiksi ja lisättiin koulutus- ja testikuvia sisältäviin kansioihin, joista kuvat annettiin syötteenä mallille. Näillä tiedostoilla ei kuitenkaan saatu mallia ajettua vaan mallin soveltaminen virheilmoitusten mukaan epäonnistui koulutusdatan generointivaiheessa. Syynä tähän pohdittiin olevan testauksessa käytettyjen kuvien dimensioiden pienempi lukumäärä tai muu eroavaisuus käytetyn datan rakenteessa.

Yksi vaihtoehto mallin käyttämisen mahdollistamiseksi olisi ollut tutkia oikeanlaisen syötteen rakennetta ja sitä, onko GRD-kuvia mahdollista muuttaa tähän rakenteeseen. Toinen mahdollisuus olisi ollut muuttaa Speckle2Void-mallin koulutukseen kirjoitettuja koodeja, jotta syötteenä olisi

voitu antaa GRD-kuvia halutussa muodossa ilman muutoksia niiden rakenteeseen. Vaikka tutkimuksessa toivottiin saatavan hyviä tuloksia Speckle2Void-mallin avulla, näiden ratkaisujen todettiin olevan epävarmoja ja laajentavan opinnäytetyötä sekä sisällöltään että työmäärältään liikaa. Lisäksi kirjoitetun koodin muuttaminen GRD-datalle sopivaksi olisi ollut kyseenalaista, sillä mallia ei alun perin ollut tähän tarkoitettu, eikä vertailtavana mallina voisi sanoa olevan Speckle2Void. Näin ollen mallin soveltamisesta jouduttiin toteamaan, ettei kirjoitettu koodi sellaisenaan sovellu GRD-datalle.

9.4 Random Forest -luokittelu

Työssä kohinanpoistomenetelmien vertailuun mittarina käytetty Random Forest -luokittelu tehtiin SNAP-ohjelmalla. Luokittelua ei tehty snappy-moduulin avulla, sillä luokittelun koulutus- ja testidatan keräämisessä vaiheiden tuloksia ja rajattuja alueita tuli tarkastella myös kuvista. Tulosten dokumentointia varten SNAP katsottiin myös tarkoitukseen paremmin soveltuvaksi.

Luokittelua varten satelliittikuvista haettiin kaksi Sentinel-1A-satelliitin SAR GRD -kuvaa, joille tehtiin SAR-datan esikäsittelyvaiheista lentoratatietojen korjaus, kuvan rajaus, anturilähtöisen kohinan poistaminen ja kalibrointi ennen kuvien pinoamista (coregistration). Luokittelua varten kuva rajattiin koordinaatteihin 65.338, 27.432, 65.268 ja 27.747, missä ei tiedettävästi ollut myrskytuhoja. Myrskytuhoalueita ei haluttu luokitteluun käytettävään alueeseen, sillä muutoin mukaan olisi tullut alueita, joissa on mahdollisesti kahta erilaista takaisinsirontaa eri aikoina. Alue sopi myös muutoin hyvin luokittelun tekemiseen, sillä alueella oli metsiä, järviä, suota ja peltoja, joista koulutus- ja testidataa oli selkeä kerätä ja luokitteluun saataisiin useampi luokka.

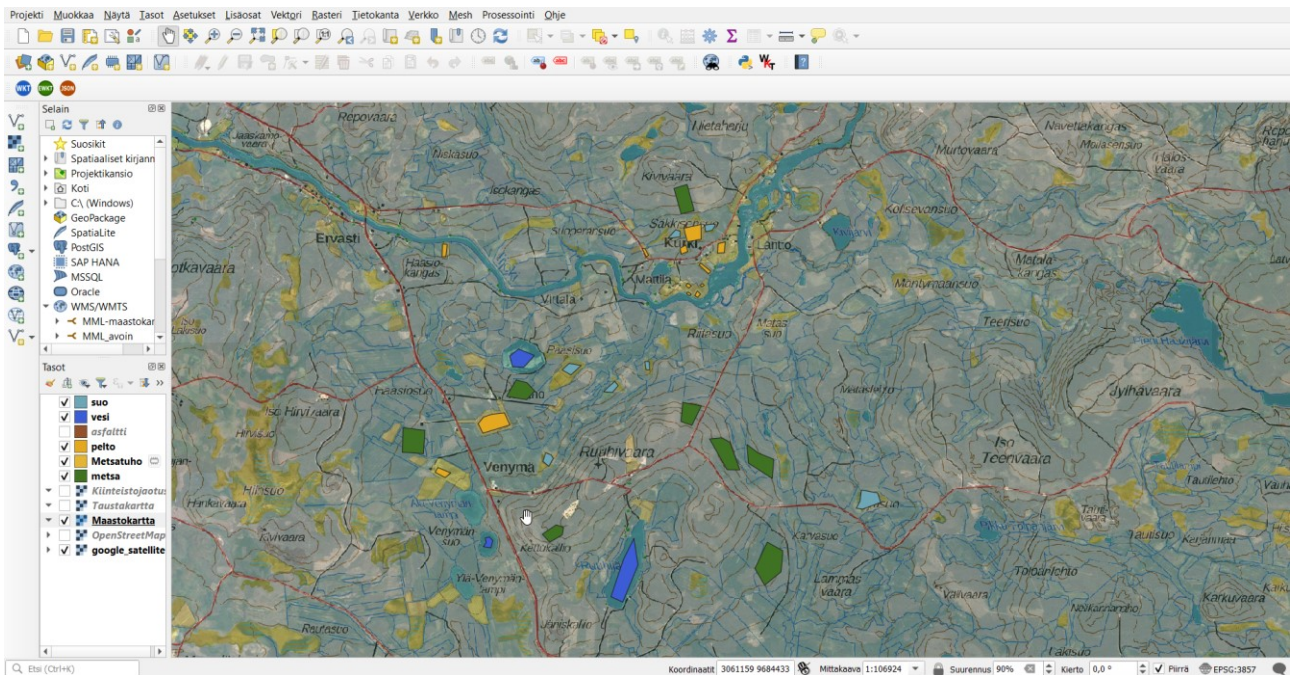
Pinoamisen tuloksena SNAP-ohjelmassa saadaan yksi tuote, jossa kahden kuvan polarisaatiot ovat erillisinä kaistoina. Useammat polarisaatiot voidaan visualisoida esimerkiksi yhtenä RGB-kuvana, jossa eri polarisaatioiden takaisinsironnan intensiteetti kuvataan omilla väreillään. Ennen luokittelun jatkamista pinoamisen tulos tarkistettiin Braunin (2020) ohjeistuksen mukaan RGB-kuvasta, jossa ei tulisi näkyä rajojen sumentumisia. Tämän jälkeen pinotun kuvan kaikille polarisaatioille suoritettiin kohinan ja geometrinen vääristymien poistaminen, sekä muunnos desibeleiksi.

Jotta Random Forest -luokitteluun saataisiin useampia ennustavia piirteitä ja näistä voitaisiin satunnaisesti valita pienempiä osajoukkoja, tehtiin pinotusta kuvasta pääkomponenttianalyysi (PCA,

principal component analysis) ja lisäominaisuuksien generointi. Braunin (2020) mukaan pääkomponenttianalyysillä saadaan pinotuista kaistoista tunnistettua eroavaisuuksia ja tiivistettyä kaistojen informaatiota uusiksi korreloimattomiksi kaistoiksi. Lisäominaisuudet puolestaan ovat kuvasta laskettuja numeerisia johdannaisia, jotka kuvaavat kertovat kuvan ominaisuuksista.

Ennustavien piirteiden lisäksi mallin koulutukseen tarvittiin kuvasta rajattuja alueita koulutus- ja testidataksi. Koska datan keräämisessä haluttiin varmistaa, että luokkiin päätyy vain oikeanlaisia alueita, tehtiin näiden rajaaminen QGIS-ohjelmassa. SNAP-ohjelmassa rajaaminen olisi joututtu tekemään SAR-kuvista, mikä todettiin erittäin epätarkaksi. Yksinkertaisuuden ja koulutusalueiden valinnan luotettavuuden vuoksi koulutukseen päädyttiin valitsemaan neljä luokkaa: "Pelto", "Vesi", "Metsä" ja "Suo". Eri metsätyyppejä olisi haluttu rajata omiksi luokikseen, mutta eri puula-jeja ei olisi voitu luotettavasti tunnistaa koulutusdataa kerättäessä.

QGIS-ohjelmassa on mahdollista luoda erilaisia karttatasoja, joiden avulla maa-alueita voi halutesaan tarkastella esimerkiksi satelliittikuvien tai maastokarttojen avulla. Kuviossa 8 näkyy ohjel- massa tehty koulutus- ja testidatan rajaaminen. Rajaamisessa apuna käytettiin maasto-, tausta-, kiinteistöjenjaotus- ja satelliittikarttatasoja. Päällekäisten tasojen läpinäkyvyyttä voi muokata, jol- loin datan keräämisen voi tehdä useamman karttatason avulla samanaikaisesti. QGIS-ohjelman hy- vänä puolena on myös mahdollisuus viedä alueet ESRI Shapefile -tiedostoiksi ja edelleen SNAP-oh- jelmaan, jolloin alueiden rajaaminen täytyi tehdä vain kerran. SNAP-ohjelmassa alueiden rajaamisesta jokaiselle filtteröinnin tulokselle erikseen olisi aiheutunut paljon virhettä, sillä rajaa- mista ei olisi voitu tehdä yhtä tarkasti jokaisella kerralla.



Kuvio 8. Koulutus- ja testidatan rajaaminen.

Koulutus- ja testidatan keräämisen jälkeen data tuotiin SNAP-ohjelmaan ESRI Shapefile -tiedostoina tuotteiden Vector Data -tietoihin, joista alueet voitiin valita SNAP:n Random Forest Classifier-toiminnossa koulutusvektoreiksi. Koulutuksessa koulutettavien näytteiden määräksi valittiin 5000 ja puiden lukumääräksi ensin 250 puuta, minkä katsottiin riittävän tarkkojen tulosten saavuttamiseen, mutta koulutusajan pysyvän kohtuullisena. Koulutuksen jälkeen tuloksia tarkastellessa huomattiin, että tuloksissa esiintyi selittämätöntä vaihtelua suodattimien kesken, minkä vuoksi luokittelu toistettiin vielä 500 puulla kolme kertaa jokaisen filterin kohdalla. Suuremmalla puiden määrällä luokittelun tulosten vaihtelun samaa filteriä käytettäessä arvioitiin vähenevän ja suodattimien keskinäisen vertailun olevan luotettavampaa. Luokittelun tulokset 500 puulla on koottu liitteen 1 taulukoihin.

9.5 Luokittelussa hyödynnettävän SAR-datan tutkiminen

Yhtenä tutkimusongelman osana esitettiin tiedon hankkiminen SAR-tutkateknologian tuottaman datan hyödyntämisestä maastonluokitteluongelmissa. Random Forest -menetelmän yhtenä tuloksena saatiin listaus tärkeimmistä selittävästä tekijöistä, joiden perusteella voidaan tehdä johtopäätöksiä aiheesta. Luokittelun lisäksi vastauksia on pyritty löytämään luomalla kattava tietopohja aiheesta ja tutkimaan miten teoriaosassa esitetyt tekijät näkyvät valitulla maastoalueella. Aiheen

tutkiminen tehtiin esikäsittelyvaiheiden jälkeen yksittäisiä kuvia tarkastelemalla, vertailemalla kesäkuun ja heinäkuun kuvia keskenään, sekä pinottujen RGB-kuvien kautta.

10 Tutkimustulokset

Seuraavissa luvuissa kerrotaan työn tutkimusvaiheesta saadut tulokset. Tulosten raportoinnissa samaan kohinanpoistomenetelmään eri ikkunako'illa viitataan 5x5- tai 7x7-merkinnöillä suodattimen nimen perässä.

10.1 Kohinanpoistomenetelmien vertailu

10.1.1 Suodatettujen kuvien subjektiivinen vertailu ja ENL-arvot

Suodatettujen kuvien subjektiivisessa vertailussa kiinnitettiin huomiota aikaisemmissa tutkimuksissa esitettyihin kriteereihin, kuten alueiden tasoittumiseen, reunojen erottumiseen ja muotojen säilymiseen. Näiden lisäksi kuvista pyrittiin tutkimaan mahdollisesti luokitteluun vaikuttavia tekijöitä. Suodatusten tulokset RGB-kuvina löytyvät liitteestä 1. Tulkinnan helpottamiseksi liitteen 1 kuville ei ole tehty geometrinen vääristymien poistamista, jolloin kuvat vastaavat enemmän totuttuja maasto- ja satelliittikuvia.

Alueiden tasoittumisen osalta havaittiin, että silmämääräisessä arvioinnissa Lee 7x7 -suodatin tasoitti järven pintaa parhaiten, seuraavaksi tässä parhaimmiksi arvioitiin Frost 7x7 ja Refined Leen ja Frost 7x7 -suodattimien yhdistelmä. Tasoituksen määrän voi myös todeta lasketuista ENL-arvoista, joissa on lukumäärällisesti samansuuntaisia tuloksia. ENL-arvojen perusteella eniten tasaisia alueita ovat suodattaneet Lee 7x7- ja Frost 7x7 -suodattimet, sekä suodattimien Frost ja Refined Lee yhdistelmät, joskin Lee 5x5 ja Refined Lee x 2 ovat ENL-arvojen perusteella suodattaneet järvalueen VH-polarisaatioarvoja edellisten veroisesti. Ennen suodatusta lasketuista ENL-arvoista voi todeta, että arvot karkeasti vastaavat Sentinel-1-satelliittien SAR-kuville ilmoitettua 4,3 ENL-arvoa.

Reunojen erottuvuudessa selkeästi heikoimmaksi todettiin Lee 7x7 -suodatin. Kauempaa katsottuna sumentumista on vaikea havaita, mutta lähemmässä tarkastelussa reunojen sumentumisesta voi todeta selkeitä eroja. Esimerkiksi Kuvion 8 kartassa näkyvät pienet lammet Kivivaaran luoteispuolella erottuvat Lee 7x7 -suodattimella suodatetussa kuvassa muihin verrattuna heikommin. Saman voi todeta voimalinjoille tehdyn hakkuukaistaleen ja Ruuhivaarasta Kivijärvelle vievän tien osalta, joka on Lee 7x7 -suodattimella hävinnyt näkyvistä lähes kokonaan. Muiden suodattimien kohdalla teiden tai hakkuukaistaleen näkyvyydessä ei huomattu merkittäviä eroja. Frost 7x7 -

suodattimella tie Ruuhivaarasta Kivijärvelle erottuu muihin verrattuna hieman heikommin, mutta on edelleen näkyvissä.

Muotojen säilymistä pyrittiin arvioimaan QGIS-ohjelman avulla maasto- ja satelliittikuvista. Järvien ja muiden maastoalueiden osalta muotojen säilymisen arviointi oli hankalaa, sillä suodattamattomassa kuvassa näkyvät kohteet eivät välttämättä muodoiltaan vastanneet maasto- ja satelliittikuvien näkymää. Parhaiten maasto- ja satelliittikuviin verraten oikeanlaiset muodot ja pienet kohteet erotti kuitenkin Refined Lee, kun taas Lee 7x7 -suodattimella muodot ovat muuttuneet sumeiksi ja pienet kohteet, kuten li-joen varrella olevat talot, suhteettoman suuriksi.

Tutkimukseen valituista kesäkuun ja heinäkuun pinotuista kuvista tarkasteltiin mahdollisten myrskytuhoalueiden erottumista eri suodattimilla. Kesäkuun ja heinäkuun takaisinsirontojen intensiteeteissä saattoi havaita pientä eroa VH-polarisaatiolla. Liitteen 1 suodattamattomaan kuvaan tällainen alue on merkitty punaisella neliöllä. Alueen erottumisen paremmuutta eri suodattimilla on kuitenkin lähes mahdotonta subjektiivisesti arvioida ja koska muutoksen syystä ei ollut varmuutta, ei tätä tutkittu enempää.

Edellisten havaintojen lisäksi suodatettuja kuvia pyrittiin tarkastelemaan intensiteettiarvojen muutoksen suunnan näkökulmasta, minkä oletettiin vaikuttavan tutkimuksessa tehdyn Random Forest -luokittelun tuloksiin. Vaikka suodatin onnistuisi tasoittamaan alueilla esiintyvää kohinaa, tämä ei välttämättä tarkoita onnistunutta luokittelua. Suodatuksen suuntaa RGB-kuvissa arvioitiin väri- ja sävyerojen perusteella, joita havaittiin erityisesti rinteillä ja oletetuilla hakkuuaukeilla, kuten Kokkovaaran ja Näätävaaran rinteillä sekä näiden läheisillä aukeilla alueilla, jotka satelliittikuvien perusteella oletettiin hakkuuaukeiksi. Liitteen 1 kuvista voi todeta, että hakkuuaukeiden kesken eri suodattimilla näkyy sävyeroja, joskaan näitä on lähes mahdotonta asettaa järjestykseen. Rinteillä näkyvän korkean intensiteetin puolestaan voi todeta olevan sitä voimakkaampaa, mitä voimakkaampaa suodatusta kuvalle on tehty.

10.1.2 Menetelmien ajallinen vertailu

Tulokset suodatukseen kuluneesta ajasta eri kohinanpoistomenetelmillä on koottu liitteen 4 taulukkoon 11. Tutkimusta tehdessä huomattiin, että käsittelyyn kulunut aika vaihteli hieman eri ajokerroilla, minkä takia samalle suodattimelle on taulukoitu useampia aikoja. Saman suodattimen

pienestä vaihtelusta huolimatta taulukoiduissa arvoissa voi havaita johdonmukaisuutta käytetyn ikkunakoon ja suodatukseen käytettyjen menetelmien lukumäärän mukaan. Isompaa ikkunaa käytettäessä suodatukseen on kulunut vähemmän aikaa, mikä sopii yhteen suodatettavien osa-alueiden vähentyneen lukumäärän ja tehtävien laskutoimitusten kanssa, kun koko kuva käydään läpi suuremmissa alueissa.

Vastaavasti käytettäessä suodatukseen useampaa menetelmää, suodatukseen kuluneen ajan voi todeta olevan karkeasti summa käytetyistä menetelmistä. Ainoan poikkeuksen tähän teki suodattimien Frost ja Refined Lee yhdistelmä ikkunakoolla 7x7, jonka viimeisen ajokerran tulokseksi saatiin kaksi sadasosaa hitaampi aika, kuin sama yhdistelmä ikkunakoolla 5x5. Vaihtelu selittyy kuitenkin Refined Lee -suodattimen käytössä tapahtuvista aikaeroista, jotka itsessään vaihtelivat useammalla tuhannesosalla. Lee, Frost ja Refined Lee välillä ei voi sanoa olevan merkittäviä eroja, kun Lee- ja Frost-suodattimissa käytetään ikkunakokoa 7x7. Tässä tulee kuitenkin muistaa, että myös Refined Lee -suodattimessa suodatukseen käytetyn ikkunan koko on 7x7.

Ajan vertailuun käytettiin alkuperäisestä SAR-kuvasta rajattua aluetta, jota muutoinkin käytettiin tutkimuksessa luokittelun tekemiseen ja alueiden tutkimiseen. Rajatun alueen koko oli 1606x1083 pikseliä, mikä oli n. 0,41 % koko kuvan koosta. Testattaessa suodatukseen kuluva aika myös koko SAR-kuvaan, käsittelyaikojen suhteissa toisiinsa oli nähtävissä sama ilmiö kuin pienemmillä kuvako'oilla. Isommilla kuvako'oilla ajoissa oli kuitenkin havaittavissa enemmän vaihtelua eikä käsittelyaikojen kasvamisen voitu todeta kasvavan samassa suhteessa kuvan koon kanssa.

10.1.3 Random Forest -luokittelun tulokset

Random Forest -luokittelussa tuloksista tutkittiin tarkkuutta, virheettömyyttä ja ennusteiden lukumäärällistä onnistumista luokittain, sekä luokittelun onnistumisprosenttia (% correct predictions) kokonaisuutena. Koska tulokset vaihtelivat eri luokittelukerroilla, tehtiin luokittelu jokaiselle suodattimen tulokselle kolme kertaa. Näiden lisäksi luokittelu tehtiin kolme kertaa suodattamattomalle SAR-kuvalle, joista taulukoitiin paras tulos. Luokittelun tulokset on koottu liitteen 3 taulukoihin.

Taulukon 10 arvoista voi lukea, että toistuvasti pienin onnistumisprosentti saatiin Refined Lee-suodattimella. Huonoimmillaan Refined Lee -suodattimella oikein ennustettuja arvoja saatiin 90,58 %

ja parhaimmillaan 94,00 %. Kun arvoa verrataan ilman suodatusta tehdyn luokittelun onnistumisprosenttiin 84,62 %, voi kuitenkin todeta, että myös Refined Lee -suodattimella on saatu selkeästi parempi lopputulos. Yksittäinen paras tulos 97,38 % saatiin Frost-suodattimella ikkunakoolla 7x7, mutta suodattimen kaksi muuta tulosta jäivät alle 96 prosenttiin. Toistuvasti yli 96:n onnistumisprosentteja saatiin Lee, sekä suodattimien Frost ja Refined Lee yhdistelmällä ikkunakoolla 7x7. Lee 7x7 -suodattimella saatiin tulokset 96,18 %, 96,04 % ja 96,18 %. Suodattimien Frost ja Refined Lee yhdistelmällä puolestaan 96,38 %, 96,04 % ja 96,80 %. Edellisten lisäksi myös suodattimien Frost ja Refined Lee yhdistelmä ikkunakoolla 5x5 tuotti kaksi yli 96 %:n tulosta, 96,70 % ja 96,20 %, jotka molemmat ovat parempia kuin Lee 7x7 -suodattimella saadut tulokset.

Yksittäisten luokkien kohdalla parhaimpia tuloksia saatiin Metsä-luokalle, jossa huonoimmillaankin tarkkuus oli Refined Lee -filterillä 97,36 %. Tässä luokassa erityisesti vääriä negatiivisia tuloksia oli vähän ja lähes kaikilla luokittelukerroilla väärien negatiivisten lukumäärä jäi alle kymmeneen, kun kuvaa oli suodatettu jollakin filterillä. Hankalammin luokiteltavia arvoja oli Pelto-, Suo- ja Vesi -luokissa. Pelto-luokassa parhain tulos 99,12 % saatiin Frost 7x7-suodattimella ja seuraavaksi parhaimmat tulokset 98,08 % ja 98,18 % suodattimien Frost 5x5 ja Refined Lee yhdistelmällä. Muilla suodattimilla tulokset jäivät alle 98 prosenttiin. Heikoimman tuloksen 93,88 % antoi jälleen Refined Lee. Vastaavia tuloksia Pelto-luokan kanssa saatiin Suo-luokassa, jossa suodattimien Frost ja Refined Lee yhdistelmällä ja Frost 7x7-suodattimella on saatu yli 98 %:n tarkkuuksia. Toistuvasti yli 97,5 prosentin tarkkuuksia saatiin lisäksi Lee 7x7 -suodattimella.

Muista luokista tulosten perusteella hieman poikkeava on Vesi-luokka, jossa Lee 7x7-suodattimen kanssa kaikilla kolmella luokittelukerralla saatiin yli 98 % tarkkuus. Lee 7x7 -suodattimen lisäksi Frost 7x7 -suodattimella saatiin tarkkuudelle tulos 98,02 % ja muut vahvasti tasoittavat kohinanpoistomenetelmät ovat tuottaneet lähes 98 %:n tarkkuuksia. Vesi-luokassa huomionarvoista on myös väärien negatiivisten lukumäärät, joita on selkeästi enemmän kuin vääriä positiivisia. Muiden luokkien kohdalla väärien positiivisten ja väärien negatiivisten välillä ei ole ollut havaittavissa yhtä selkeää jakautumista, mutta Vesi-luokassa väärien tulosten voi havaita painottuvan vääriin negatiivisiin tuloksiin.

Tehtäessä vertailua saman suodattimen kesken eri ikkunako'oilla, taulukoiden 6–10 tuloksista voi huomata, että luokitteluiden tulokset ovat useimmissa tapauksissa parantuneet, kun ikkunan

kokoa on kasvatettu. Vesi-luokassa ikkunan koon kasvattaminen näyttää vähentäneen erityisesti väärien negatiivisten osuutta ja hieman myös väärien positiivisten osuutta. Metsä-luokassa puolestaan väärien negatiivisten arvojen voi havaita vähentyneen ikkunan kokoa kasvatettaessa ja väärien negatiivisten, joita jo ennestään oli vähän, pysyneen suunnilleen samana. Vastaavasti Suo-luokassa parannusta näyttää tulleen etenkin väärien positiivisten määrän vähenemisessä, mutta väärien negatiivisten vähenemisessä ei ole yhtä selkeää trendiä. Myöskään Frost ja Refined Lee -suodattimien yhdistelmällä ikkunan koon kasvattaminen ei enää ole tuonut selvää muutosta väärien positiivisten tai väärien negatiivisten määrään. Lisäksi Pelto-luokassa ikkunakoon kasvattaminen näyttää jopa hieman kasvattaneen väärien positiivisten negatiivisten tulosten lukimääriä.

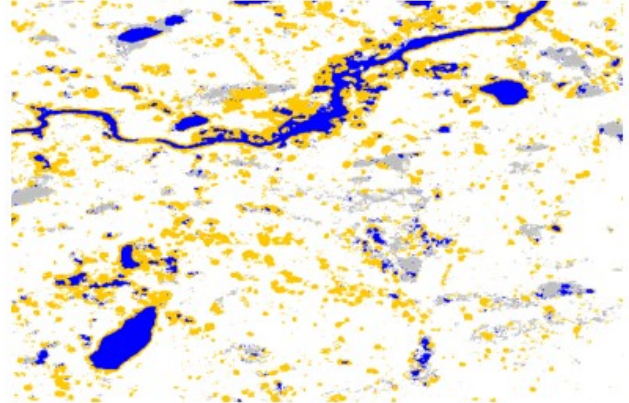
SNAP-ohjelmassa Random Forest -luokittelun tuloksista visuaaliseen vertailuun otettiin Refined Lee, Lee 7x7 sekä suodattimien Frost 7x7 ja Refined Lee yhdistelmä, joiden sekä luokittelun numeerisen vertailun että kohinan poiston subjektiivisen tarkastelun perusteella todettiin selvästi poikkeavan toisistaan. Kuvion 9 ensimmäiseen kuvaan on rajattu luokiteltua aluetta vastaava alue satelliittikuvasta. Muissa Kuvion 9 kuvissa on luokittelujen tulokset eri suodattimilla käsiteltyinä, missä sinisellä näkyvät alueet on luokiteltu luokkaan "Vesi", keltainen luokkaan "Pelto", harmaa luokkaan "Suo" ja valkoinen "Metsä"-luokkaan. Tehdystä geometrinen vääriskymien käsittelystä johtuen satelliittikuvassa ja luokitellussa kuvassa näkyvien kohteiden sijainnit ja muodot eivät täysin vastaa toisiaan.

Luokittelun tuloksia ja satelliittikuvaa vertailemalla voidaan todeta, että kaikilla suodattimilla väärin luokiteltuja alueita ovat olleet osa pelloista ja soista. Tällaisia aluita voi erottaa esimerkiksi Ruuhijärven pohjoispuolella, missä iso peltoalue on luokiteltu vedeksi ja Kivijärven luoteispuolella, missä suuri suoalue on luokiteltu pelloksi ja vedeksi. Kun kuvista vertailee väärin luokiteltuja alueita, voi huomata, että Ruuhijärven pohjoispuolella vedeksi luokitellut alueet näyttävät suuremmilta Lee 7x7 -filtteriä käytettäessä, kuin Refined Lee tai suodattimien Frost 7x7 ja Refined Lee yhdistelmällä tehdyillä suodatuksissa.

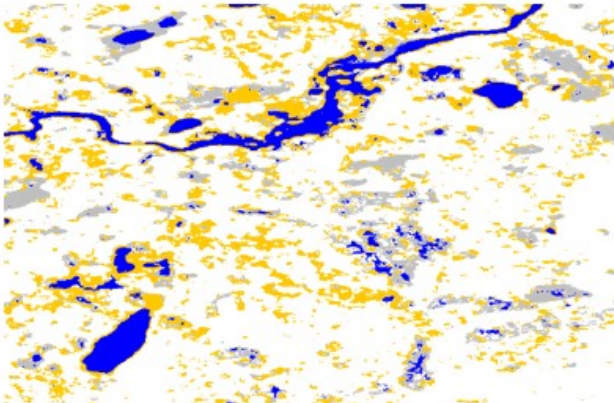
a) Maastokartta



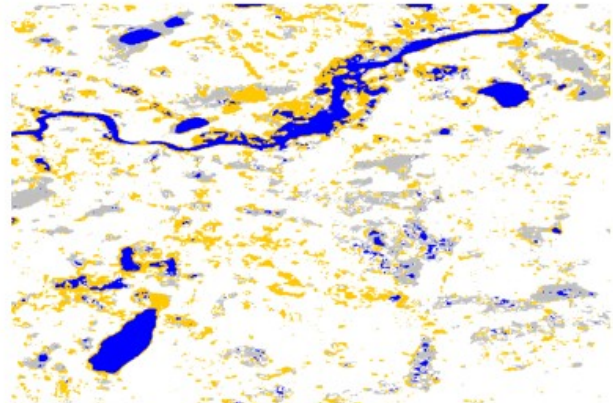
b) Refined Lee



c) Lee 7x7



d) Frost 7x7 & Refined Lee



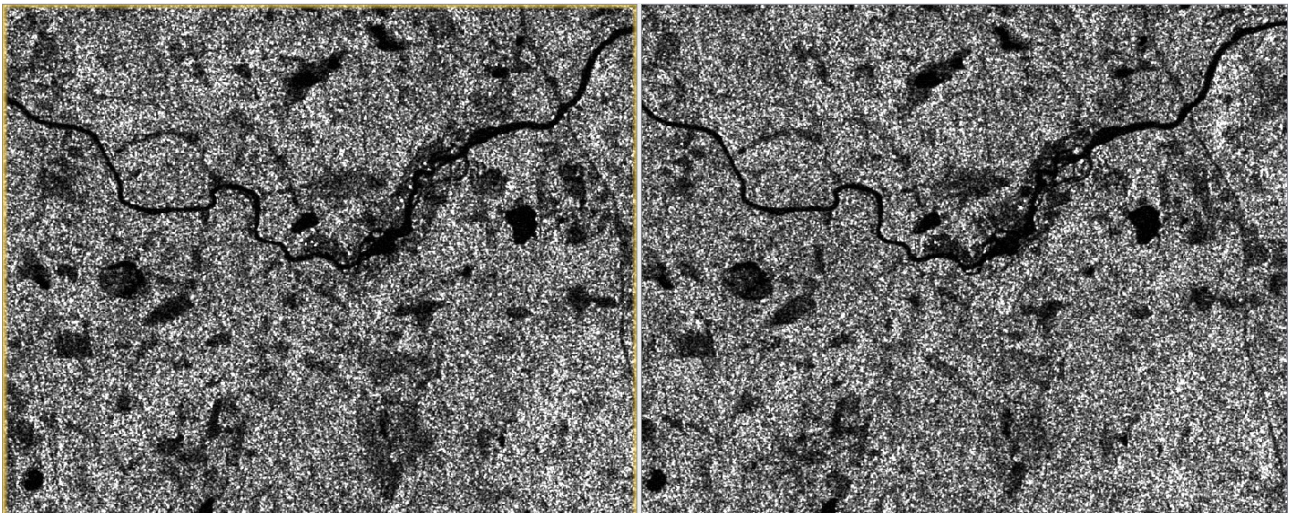
Kuvio 9. Random Forest -luokittelun tuloksia kuvana.

10.2 SAR-datan hyödyntäminen maaston luokitteluongelmissa

SAR-datan hyödyntämistä maaston luokitteluongelmissa tutkittiin kirjallisuuteen perehtymällä, tutkimalla SAR-kuvia ja tekemällä maaston luokittelua SAR-kuvista Random Forest -menetelmällä. Kirjallisuudessa vahvimmin esiintuotuja asioita olivat käytetty aallonpituus, polarisaatiot ja signaalin tulokulma, jotka osoittautuivat myös tutkimuksen käytännöntoteutuksessa tärkeiksi tiedoiksi SAR-kuvia ja luokittelun tuloksia tulkittaessa. Tutkimusta tehdessä näkyvimpiä tietoja SNAP-ohjelmassa ja snappy-moduulissa olivat signaalin intensiteetti- ja amplitudi-arvot eri polarisaatioilla, mutta datan esikäsittelyn eri vaiheiden toteuttamiseen tarvitaan myös yksityiskohtaisempaa tietoa muun muassa satelliitin lentoradasta, signaalin tulokulmasta ja sensorikohtaisista ominaisuuksista.

Kuvien subjektiivisessa tulkinnassa intensiteettiarvojen sekä pohjatiedon avulla pystyttiin karkeasti erottamaan suo-/pelto-, vesi- ja metsäalueita toisistaan. Kun aallon käyttäytyminen erilaisissa

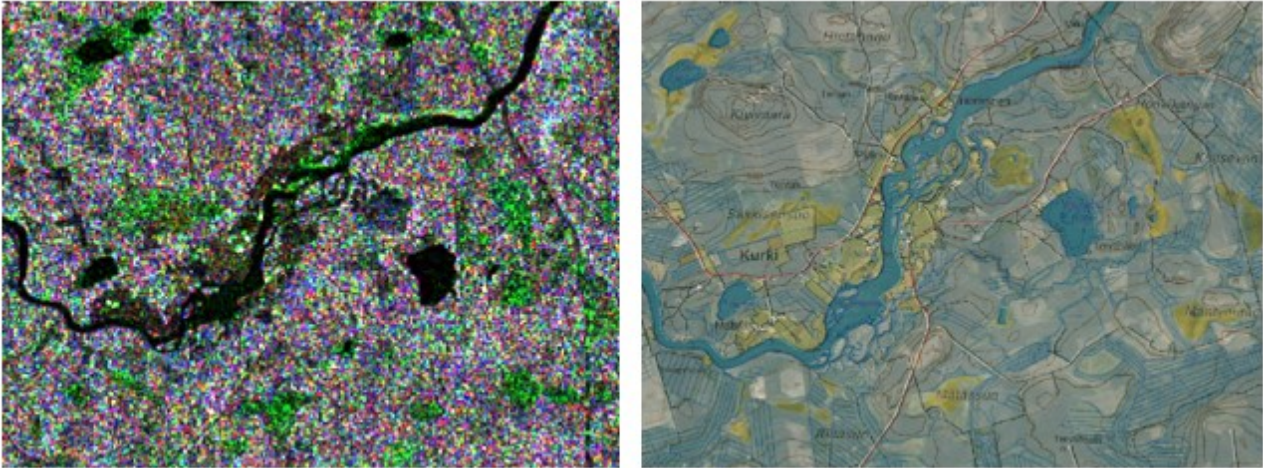
väliaineissa tunnetaan, voidaan päätellä millainen kohde kuvissa näkyy ja analysoida kohteen ominaisuuksia. Tähän tutkimukseen valituissa kuvissa aallon absorboituminen ja heijastuminen erilaisista väliaineista näkyi eroina suo-/pelto-, vesi- ja metsäalueiden välillä. Nämä alueet pystyttiin erottamaan toisistaan vertailemalla eri aikoina otettuja SAR-kuvia ja tutkimalla intensiteettien vaihtelua. Suuret erot takaisinsironnassa samalla alueella viittaavat väliaineeseen, jossa veden määrä vaikuttaa takaisinsirontaan. Tämän perusteella tummat alueet, joissa näkyy voimakasta vaihtelua kuvien välillä, voitiin tulkita soiksi tai pelloiksi, pysyvästi tummat alueet vedeksi ja vaaleammat alueet metsiksi. Kuviossa 10 ilmiön voi havaita esimerkiksi kuvien oikeassa ylälaudassa voimalinjan viereisillä suoalueilla.



Kuvio 10. VH-polarisaation intensiteettiarvot heinäkuussa ja kesäkuussa.

Pelkkiä yksittäisestä polarisaatiosta saatuja intensiteettiarvoja tutkimalla alueiden erottaminen oli hyvin hankalaa, minkä takia lähes yhtä tärkeäksi ominaisuudeksi intensiteettiarvojen kanssa osoitautui aaltojen polarisaatio, jota tutkimalla saatiin lisäinformaatiota tutkittavien kohteiden rakenteesta. Tässä tutkimuksessa ilmiön voi havaita kuviosta 11, jossa ensimmäiseen kuvaan on pinottu kahden eri SAR-kuvan polarisaatiot ja toisessa kuvassa esitetään aluetta vastaava maastokartta. Pinotussa RGB-kuvassa kesäkuussa mitattu VH-polarisaatio näkyy sinisellä, heinäkuussa mitatuista polarisaatioista VH punaisella ja VV vihreällä. RGB-kuvaa ja maastokarttaa vertailemalla voi huomata, että metsäisiltä alueilta on pääasiassa mitattu VH-polarisaatiota, kun taas suoalueilta on enimmäkseen mitattu VV-polarisaatiota. Poikkeuksen tähän tekee kuitenkin Kivijärven yläpuolella näkyvä Autiosuo, jolta on ollut vähäistä, enimmäkseen sinisenä näkyvää takaisinsirontaa. Vaikka

optisesta satelliittikuvasta Autiosuo ei erottaisi muista suoalueista poikkeavaksi, on tutkakuvassa näkyvä ero selkeä. Mahdollinen selitys eroihin voi olla alueen vetisyys ja erilainen kasvillisuus verrattuna muihin suoalueisiin.



Kuvio 11. Kahden SAR-kuvan pinottu RGB-kuva ja aluetta vastaava maastokartta.

Vähemmän selkeä, mutta maaston ja kuvien tulkinnessa oleellinen asia liittyy tutkan kuvantamissuuntaan ja signaalin tulokulmaan. Kuvion 11 RGB-kuvaa ja maastokarttaa tutkimalla voi huomata, että sähkölinjalle raivattu kaistale metsien keskellä ja tiet erottuvat pätikkäin tummina viivoina pinotussa SAR-kuvassa. Vastaavasti vaarojen länsirinteillä voi havaita suurempaa takaisinsirontaa kuin kuin itäisillä rinteillä. Syy tähän löytyy tutkan kuvantamissuunnasta maanpintaan nähden ja signaalin tulokulmasta suhteessa maaston muotoihin. Raivatun metsäkaistaleen ja teiden kohdalla muodostuu varjokohtia, josta takaisinsirontaa ei tule. Vaarojen länsipuolella rinne puolestaan muodostaa heijastumiselle otollisen kulman ja takaisinsironnan määrä on suurempaa. Random Forest -luokittelussa tätä tietoa ei hyödynnetty, mutta kuvien subjektiivisessa tarkastelussa voitiin tehdä oletuksia maaston muodoista.

Osa edellä esitetyistä, kuvien subjektiivisessa tulkinnessa tehdyistä havainnoista, näkyy myös Random Forest -luokittelun tuloksissa. Kaikissa luokittelussa VH-polarisaatioiden intensiteettiarvoista laskettu keskiarvo on noussut suurimmaksi ennustavaksi tekijäksi. Tämän jälkeen parhaiten onnistuneissa luokittelussa, joissa oikein ennustettujen prosenttiosuus on yli 96, selittävänä tekijänä esiintyy kahta poikkeusta lukuun ottamatta VH-polarisaation entropia. Näiden jälkeen tuloksissa

esiintyy enemmän vaihtelua, ja ennustaviksi tekijöiksi on noussut kesäkuun VH ja VV-polarisaation intensiteettiarvoja, pääkomponentteja ja intensiteettiarvojen homogeenisuus VH-polarisaatiolla.

11 Tulosten tulkintaa

11.1 SAR-teknologian tuottaman datan hyödyntäminen maastonluokitteluongelmissa

Ensimmäisenä tutkimuskysymyksenä kysyttiin, mitä SAR-teknologian tuottamaa dataa voidaan hyödyntää maastonluokitteluongelmissa. Yleisesti SAR-kuvia tulkittaessa ja Random Forest -luokittelussa tärkeiksi ominaisuuksiksi SAR-datassa osoittautuivat intensiteetti- ja amplitudiarvot eri polarisaatioilla. SAR-datan käsittelyyn kuuluu kuitenkin olennaisena osana datan esikäsittely, mihin puolestaan tarvitaan datan metatiedoissa olevia tietoja. Vähintäänkin yhtä tärkeäksi käytössä olevan datan ohella nähtiin tietopohja datan tulkitsemiseen.

Vaikka datassa varsinaiseen kuvien tulkintaan ja maanluokittelun tekemiseen hyödynnettäviä tietoja ovat vain polarisaatiot ja näiden intensiteettiarvot, voitiin näistä koostaa lisäinformaatiota, kun käytettävissä oli useampia kuvia eri ajanjaksoilta. Esimerkiksi Random Forest -luokittelussa yhdeksi ennustavaksi tekijäksi nousi VH-polarisaation entropia, jota ei datassa alun perin ole. Tämän tiedon voikin katsoa hyvin luonnehtivan tutkittavaa aluetta, sillä osa tutkittavista alueista on herkempiä vallitseville sääolosuhteille kuin toiset, ja siten takaisinsironnan entropia voi olla toisilla alueilla merkittävästi suurempaa.

Tutkimusta tehtäessä todettiin, että kahdella polarisaatiolla samankaltaisia alueita, kuten suo- ja peltoalueet, oli hankala erottaa toisistaan. Mahdollisesti useammilla SAR-kuvilla olisi voitu tunnistaa piirteitä, jotka olisivat helpottaneet alueiden tunnistamista. Yksi mahdollinen apu olisi voinut olla myös neljän eri polarisaation intensiteettiarvojen vertailu, joiden takaisinsirontojen suhteista on kerrottu luvussa 3.3.5. Eri aallonpituuksilla kuvannettu data voisi myös auttaa samankaltaisten alueiden erottamisessa, sillä X-, C- S- ja L-kaistoilla sama pinta voi näyttäytyä tasaisena tai karkeana. Loppujen lopuksi hyvin samankaltaisten alueiden tunnistaminen toisistaan voi olla pelkällä SAR-datalla lähes mahdotonta, jos alueiden ominaisuudet eivät aaltojen vuorovaikutuksen osalta poikkea toisistaan.

11.2 Neuroverkkoihin perustuvat menetelmät kohinan poistamisessa

Osa tutkimusongelmaa oli selvitystyö neuroverkoilla kehitetyistä kohinanpoistomenetelmistä ja niiden mahdollisesta soveltamisesta pilottitapauksessa käytettävään dataan. Aiheesta haluttiin

tietää, millaisia neuroverkkoihin perustuvia menetelmiä SAR-kuvissa esiintyvän kohinan poistamiseksi on kehitetty, mihin menetelmät perustuvat ja miten ne poikkeavat toisistaan. Menetelmien soveltamisesta ja tehokkuudesta verrattuna perinteisiin menetelmiin pyrittiin saamaan tietoa valitsemalla menetelmistä yksi lupaava ja vertailemalla eri menetelmiä segmentointiongelman näkökulmasta.

Tässä tutkimuksessa tarkastellut neuroverkkoihin perustuvat kohinanpoistomenetelmät ovat olleet konvoluutioneuroverkoja, joista osa on alun perin kehitetty optisille kuville ja osa SAR-kuville. Erilaisia syväoppimismenetelmiä vaikuttaa olevan lukumäärällisesti paljon, mutta vertailevia tutkimuksia on tehty vähän. Ilmeisesti konvoluutioneuroverkoilla saadut hyvät tulokset sekä kuvien tunnistuksessa että optisten kuvien kohinanpoistossa on luonut buumin kohinanpoistosuodattimien kehitykseen syväoppimismenetelmillä, mutta olemassa olevien suodattimien vertailusta ei olla oltu yhtä lailla kiinnostuneita tai tätä ei ole voitu tehdä laajassa mittakaavassa. Koska menetelmien tutkimus on vielä lapsenkengissä, nämä eivät vielä näytä yleistyneen maankäytön luokittelua tai anomaliaita koskeviin tutkimuksiin.

Yleisesti erona kohinanpoistomenetelmien välillä voi yksinkertaistaen kertoa olevan periaatteen kohinattoman pikseliarvon laskemisesta, eli mistä arvoista ja miten kohinattoman pikselin arvo tulisi laskea. Menetelmien vaihtelun taustalla on yhtenä tekijänä vaikuttanut erilaiset näkemykset mallintaa kohinaa, jonka luonteesta on vuosien kuluessa saatu lisää tietoa. Taustalla on ollut oletus kohinan korreloimattomuudesta, minkä on myöhemmin osoitettu paikkaansa pitämättömäksi. Lisäksi kohinan noudattamasta todennäköisyysjakaumasta on ollut vaihtelevia näkemyksiä ja kohinattoman pikselin pistetodennäköisyydelle on annettu erilaisia esityksiä. Kohinan korreloinnin käsittelyyn on kuitenkin kehitetty ratkaisuja, joiden on todettu parantavan olemassa olevien kohinanpoistomenetelmien tuloksia.

Erilaisten syväoppimismenetelmien suurena periaatteellisena erona näyttää olevan neuroverkon koulutukseen käytetty data, jota on SAR-kuvien tapauksessa haasteellista saada. Kuten jo aikaisemmin on todettu, SAR-datasta ei ole mahdollista saada koulutus- ja testidataan kohinattomia vastineita, minkä takia ongelma on täytynyt ratkaista tavalla tai toisella. Osa tutkijoista on pyrkinyt ratkaisemaan ongelman lisäämällä kuviin synteettistä kohinaa, osa puolestaan on hylännyt tämän

vaihtoehdon ja pyrkinyt kehittämään tavan, joissa kohinattomia kuvia ei tarvita. Tässä tutkimuksessa esitellyistä menetelmistä ongelma oli ratkaistu käyttämällä blind-spot-neuroverkkoja.

Neuroverkoilla kehitetyistä menetelmistä suuntauksen, jossa käytetään oikeita SAR-kuvia koulutusmateriaalina, voidaan katsoa olevan lupaava. Vaikka kohinan luonteen tutkimuksessa on päästy pitkälle, on kyseenalaista, voidaanko kuviin luoda synteettisesti kohinaa todellista tilannetta vastaavalla tavalla. Tästä syystä oikeita SAR-kuvia koulutuksessa käyttävät menetelmät voidaan katsoa paremmiksi ja tutkimuksen tulisi keskittyä näihin. Synteettisen kohinan lisäämisen voi myös nähdä yhtenä ylimääräisenä työvaiheena, joka menetelmää käyttöön otettaessa tulisi tehdä.

11.3 Kohinanpoistomenetelmän valinta SAR-kuvien esikäsittelyssä ja Suomen maastoalueille sopivat kohinanpoistomenetelmät

11.3.1 Kohinanpoistomenetelmän valinta SAR-kuvien esikäsittelyssä

Viimeisinä tutkimuskysymyksenä kysyttiin, miten esikäsittelyssä käytettävä kohinanpoistomenetelmä tulisi valita ja soveltuuko jokin kohinanpoistomenetelmä erityisen hyvin Suomen maastoalueille. Random Forest -menetelmällä tehdystä luokittelusta saatiin viitettä siitä, että kuvien segmentointiin soveltuu maastoalueita voimakkaasti suodattava kohinanpoistomenetelmä. Yksittäistä selvästi muita parempaa menetelmää ei voida tuloksista nostaa esille, sillä useammalla eri suodattimella saadut tulokset olivat lähellä toisiaan ja luokittelujen tulokset voivat vaihdella myös samaa suodatinta käytettäessä. Kuitenkin Lee 7x7, Frost 7x7 sekä suodattimien Frost ja Refined Lee yhdistelmillä, jotka tasoittivat kohinaa tutkituista menetelmistä eniten, voidaan sanoa saadun parhaimpia tuloksia.

Valittua kohinanpoistomenetelmää enemmän tuloksiin näyttää vaikuttaneen ikkunan koko, jolla suodatusta on tehty. Samaa suodatinta käytettäessä havaittiin, että ikkunakoon kasvattaminen useimmiten paransi luokittelun tulosta. Kahden eri suodattimen yhdistelmää käytettäessä tulokset eivät kuitenkaan enää merkittävästi parantuneet, vaan pysyivät lähes samoina tai saattoivat hieman huonontua. Tulosten heikentymistä oli havaittavissa Pelto-luokassa, missä väärin positiivisten lukumäärä hieman kasvoi. Syy tähän voi löytyä luokkia ennustavista tekijöistä, missä kohinan entropia nousi yhdeksi selittäväksi tekijäksi parhaiten suoriutuneilla suodattimilla. Kun kohinaa tasoitetaan, intensiteettiarvojen entropia vähenee. Tietyillä alueilla takaisinsironnan vaihtelu voi

kuitenkin olla ominaista ja auttaa tunnistamaan samankaltaisia maa-alueen toisista. Esimerkkinä pellot, joilla vedenmäärä ja viljelykasvien kasvu kasvukauden aikana voivat vaikuttaa takaisinsironnan määrään.

Jos segmentoinnin lisäksi kuville tarvitsee tehdä subjektiivista tarkastelua, tulee tällöin ottaa huomioon myös rajojen terävöityminen. Tässä tapauksessa tutkituista suodattimista parhaimmiksi voidaan nostaa suodattimien Frost ja Refined Lee yhdistelmät, jotka olivat subjektiivisessa tarkastelussa Lee 7x7 -menetelmää parempia. Suodattimien Frost ja Refined Lee suodattimien yhdistelmällä ikkunan koko 5x5 nähtiin tarkemmaksi kuin 7x7, mutta kyseessä on subjektiivinen vertailu, mihin myös tarkasteluetäisyydellä on vaikutusta.

Suodatusten ajallisessa vertailussa ei nähty merkittäviä eroja, mutta tässä on syytä ottaa huomioon käytetyn datan vähäisyys ja erojen kertautuminen, kun suodatettavia kuvia on paljon. Ajallisten erojen merkitykseen menetelmän valinnassa vaikuttanee myös työskentelytapa ja käyttötarkoitus, jolla esikäsittelyä tehdään. Jos kuvien esikäsittely tehdään snappy-moduulin avulla, voidaan useampien kuvien esikäsittely siirtää laskentapalvelimelle tehtäväksi ja mahdollisesti hyödyntää käsittelyyn kuluva aika.

Yleisesti kohinanpoistomenetelmien valinnasta voi sanoa, että kohinan poistaminen millä tahansa tutkituista menetelmistä paransi luokitteluprosessissa oikein ennustettujen arvojen osuutta ja helpotti kuvien tulkintaa verrattuna siihen, että suodatusta ei olisi tehty ollenkaan. Kun luokittelussa halutaan mahdollisimman hyviä tuloksia, kannattaa menetelmiä vertailla ja pohtia menetelmän valintaa sovelluskohteen tavoitteiden näkökulmasta. Vaikka tässä tutkimuksessa parhaimmat tulokset saatiin suodattimien Frost ja Refined Lee yhdistelmällä, ei tämä sulje pois sitä mahdollisuutta, että muilla vastaavilla suodattimilla tai niiden yhdistelmillä saataisiin yhtä hyviä tuloksia. On myös mahdollista, että jonkin toisen datan, luokiteltavien alueiden tai luokittelumenetelmän näkökulmasta suodattimien Frost ja Refined Lee yhdistelmä ei tuota parhaimpia tuloksia vaan kohinaa tulisi suodattaa enemmän tai vähemmän.

Edellä esitetyn päätelmän oikeellisuutta on hankala arvioida edeltävien tutkimusten pohjalta, sillä käytetty data tai suodattimet eivät välttämättä vastaa toisiaan. Dimov ja muiden (2017) tutkimuksessa, jossa tutkijaryhmä yritti luokitella satoalueita toisistaan, tuloksena oli, että Gamma-

suodattimen käytöllä saatiin huonompia tuloksia kuin suodattamattomalla SAR-datalla. Tässä täytyy kuitenkin huomioida, että tutkimuksen käytössä oli vain VV-polarisaation arvot, joista myös tutkijat totesivat, että VH-polarisaatio on herkempi kasvillisuudelle. On kuitenkin mahdollista, että tutkimuksen tulos kertoo suodatuksen määrästä, mikä tietyn tyyppisille alueille on sopiva. Tutkitut alueet ovat mahdollisesti olleet takaisinsironnaltaan niin samankaltaisia, että vähäisenkin suodatus on heikentänyt luokittelumenetelmien kykyä erottaa alueet toisistaan.

11.3.2 Suomen maastoalueille sopiva menetelmä

Koska vastaavia segmentoinnin kautta kohinanpoistomenetelmiä arvioivia tutkimuksia ei löydetty, on vaikea arvioida ovatko tulokset ominaisia käytetylle datalle vai yleisesti maastonluokitteluongelmille. Braun (2020) on maa-alueiden luokittelua koskevassa ohjeuksessa suositellut käytettäväksi vahvoja filttäreitä, mutta tiettyjä suodattimia ei ole nimetty. Tietyn kohinanpoistomenetelmän sopivuutta juuri Suomen maastoalueille oli alun perin pohdittu erilaisten metsätyyppien näkökulmasta, mutta tutkimusta tehdessä maaston heterogeenisuus ja pienet yksityiskohdat osoittautuivat myös oleelliseksi tarkastelukohdaksi. Suomessa maastoalueilla maasto on sanalla sanoen hyvin vaihtelevaa, pieniä pirstaleisia alueita ja ohuita yksityiskohtia on paljon. Tästä syystä käytettävän suodattimen ei tulisi kadottaa pieniä yksityiskohtia kuvista, eikä menetelmä saisi suodattaa kuvia liikaa.

Eri metsätyyppien tunnistamisen näkökulmasta suodattimen valintaan ei löydetty suoraa vastausta luokitteluprosessin kautta tai olemassa olevista tutkimuksista. Ananto ja muiden (2019) tutkimuksessa Frost-filtterillä biomassan ja takaisinsironnan määrän oli todettu korreloivan parhaiten, mutta tutkimus ei käsitellyt eri puulajeja vaan kasvillisuuden määrää. Kuitenkin Frost-suodattimen käytöllä saatiin luokittelussa parempia tuloksia kuin Refined Lee -suodattimella Ananto ja muiden (2019) tutkimuksen tapaan. Vaikka tutkimukset olivat luonteeltaan erilaisia, voi näiden perusteella pohtia suodattimen valintaa. Mahdollisesti tietynlaisessa maastossa esiintyvät lajit eivät itsessään aiheuta vaihtelua luokitteluiden tuloksissa eri suodattimilla, vaan valittavan suodattimen sanelee luokkien poikkeavuuden määrä toisistaan takaisinsironnan osalta.

11.3.3 Speckle2Void-menetelmä kohinan poistamisessa

Työn käytännöntoteutuksessa yhdeksi vertailtavaksi menetelmäksi valittiin Speckle2Void siitä saattujen lupaavien tulosten ja koulutuksessa käytetyn datan vuoksi. Tutkimusta tehdessä kuitenkin paljastui, että menetelmä on kehitetty neljä polarisaatiota sisältävälle datalle, eikä sitä voitu sellaisenaan soveltaa tutkimusaineistoon. Vaihtoehtoina olisi ollut joko käytettävän datan tai kirjoitetun koodin muokkaaminen. Tämän ei kuitenkaan enää katsottu kuuluvan opinnäytetyön laajuuteen, vaan nähtiin jatkotutkimuksen aiheeksi. Molini ym. (2020) tekemästä vertailusta muihin kohinanpoistomenetelmiin ja erityisesti menetelmää esittelevässä tutkimuspaperissa näytetyistä suodattamista kuvista voi kuitenkin arvioida sen sopivuutta Random Forest -luokittelun kanssa käytettäväksi suodattimeksi.

Tämän tutkimuksen tuloksista voidaan päätellä, että Random Forest -menetelmällä hyvän luokittelutuloksen saavuttamiseksi käytetyn suodattimen tulisi vähentää kohinaa tasaisilla alueilla kuitenkin kadottamatta sillä esiintyvää luonnollista vaihtelua. Subjektivisessa vertailussa (ks. Kuvio 5) Speckle2Void-menetelmä näytti säilyttäneen pintojen rakenteen hyvin verrattuina muihin menetelmiin, minkä vuoksi menetelmän voisi olettaa tuottavan hyviä tuloksia, kun luokittelun selittäviksi tekijöiksi koostetaan intensiteettiarvojen entropia ja homogeenisuus. Myöskään eri polarisaatioiden välisten intensiteettiarvojen vertailun takia menetelmän ei haluta muuttavan alueellisia arvoja liikaa, sillä tämä voisi vaikeuttaa eri polarisaatioilla saatavien intensiteettiarvojen vertailua keskenään.

Koska käytännön tutkimuksessa ongelmanratkaisutapa on harvoin alusta asti selvillä, olisi suotavaa, että käytettävä kohinanpoistomenetelmä olisi hyvä useammasta näkökulmasta. Käytettävää menetelmää ei haluttaisi kesken tutkimuksen vaihtaa, jos ongelman ratkaisutapa muuttuu. Konvolutioneuroverkoilla tehtävän luokittelun näkökulmasta Speckle2Void-mallin voisi olettaa toimivan hyvin myös objektien tunnistuksessa, sillä Kuvion 5 perusteella malli terävöittää hyvin reunoja ja säilyttää pieniä yksityiskohtia.

Datan esikäsittelyn työmäärän kannalta Speckle2Void tulisi olemaan työlämpi, sillä malli tulisi kouluttaa ennen sen soveltamista. Tehdyn luokittelun näkökulmasta voisi myös pohtia, halutaanko tätä lisätyötä tehdä, sillä luokittelussa saadut tulokset olivat lukumäärällisesti jo yksinkertaisimmillakin suodattimilla hyviä. Tässä on kuitenkin muistettava, että luokittelussa valitut alueet olivat

Pelto- ja Suo-luokkia lukuun ottamatta hyvin erilaisia ja tällaisen karkeaan luokitteluun, mikä on mahdollista tehdä myös optisista kuvista, ei välttämättä SAR-dataa haluttaisi muutoinkaan käyttää. Oikeassa käyttötapauksessa tunnistettavat alueet voivat olla ominaisuuksiltaan hyvin lähellä toisiaan ja tällöin kohinanpoistamisessa voitaisiin tarvita kehittyneempää menetelmää. Tällaisessa tapauksessa useampien polarisaatioiden käyttäminen ja datan rikastaminen muilla tiedoilla olisivat todennäköisesti myös ratkaisuvaihtoehtoja kohinanpoistamisen rinnalla.

12 Pohdinta

12.1 Tutkimuksen luotettavuus

Tutkimuksessa pyrittiin tekemään kattava selvitys kohinanpoistomenetelmien vertailevista tutkimuksista ja niiden tuloksista. Tietopohjaan löydettiin aiheesta paljon tutkimuksia, mutta näiden pohjalta oli hankalaa saada vertailukelpoista tietoa tutkittavista suodattimista. Osittain vertailevat tutkimukset olivat suppeita, joissa vertailtiin vain muutamia suodattimia keskenään. Tutkimuksissa käytetyt nimitykset menetelmistä, metriikat ja data myös vaihtelivat, minkä takia menetelmien vertailu oli hankalaa ja koottu tieto on pirstaleista. Aiheeseen läheisesti liittyviä tutkimuksia olisi myös ollut paljon enemmänkin, mutta työn laajuuden rajaamiseksi näihin ei voitu perehtyä. Siten koottu tietopohja jää vain pintaraapaisuksi aiheesta.

Vaikka tutkimustulosten kokoaminen ja käytännön toteutuksessa suodatettujen kuvien vertailu oli hankalaa, voi subjektiivisessa vertailussa saatujen tulosten sanoa olevan saman suuntaisia kuin aikaisemmissa tutkimuksissa. Myös kuvista ennen suodatusta lasketut ENL-arvot vastasivat ESA:n sivuilla Sentinel-1-satelliittien SAR-kuville ilmoitettuja ENL-arvoja, mikä viittaisi siihen, ettei kuvien käsittelyssä ole tehty virheitä.

Random Forest -luokittelun osalta on useampia seikkoja, joiden voi tunnistaa vaikuttavan tutkimuksen luotettavuuteen. Yksi näistä tekijöistä on koulutusalueiden valinta. Vaikka koulutusalueiden rajaamisessa käytettiin apuna useampia karttoja ja optisia satelliittikuvia, olisi tarkka rajaus edellyttänyt alueiden tarkistamista paikan päällä. Näin voisi todeta myös koulutuksen tuloksista, joita tarkastellessa osa suoalueista näytti luokitellun vedeksi. Luokittelu vedeksi voi tosin olla olosuhteista riippuen aivan oikea, jos suoalueelle on kertynyt vettä lammiksi asti.

Koulutusalueiden valinnassa olisi myös voitu tutkia tilannetta, jossa alue olisi rajattu tarkasti kahden alueen väliltä. Tällöin pienet erot reunojen tarkentumisessa olisivat saattaneet nostaa esiin eroja suodattimien välillä ja muuttaa luokittelujen tuloksia. Koulutusalueiden valinnassa useamman luokan käyttäminen ja erityisesti samankaltaisten luokkien erottaminen saattaisi myös tuottaa erilaisia tuloksia. Yleisesti käytettävissä olevan datan määrää oli pieni ja luotettavaan luokitteluun olisi tarvittu useampia kuvia samalta alueelta. Tästä johtuen saatujen tulosten voi sanoa olevan vain suuntaa antavia.

12.2 Jatkotutkimusaiheita

Työn yhtenä tavoitteena oli testata Speckle2Void-mallia pilottitapauksessa käytettävään dataan. Tämä ei suoraan onnistunut muokkaamalla dataa tai menetelmää, mutta voisi olla mahdollista tutkimalla työssä alun perin käytettyä dataa ja sen muuttamista .mat-tiedostoiksi. Tavoiteltavana vaihtoehtona voisi pitää kahta polarisaatiota sisältävän datan muokkaamista neljä polarisaatiota sisältävän datan muotoon, sillä olisi kyseenalaista muokata Speckle2Void-mallia ja tämän jälkeen tutkia menetelmää. Datan muokkaamisessa tulee kuitenkin ottaa huomioon myös sen esittäminen kompleksilukuina ja mahdolliset vaikutukset mallin toimintaan, minkä takia aihe ei välttämättä ole täysin suoraviivainen.

Tässä tutkimuksessa kohinanpoistomenetelmien vertailuun päätettiin käyttää Random Forest -menetelmää pitkälti käytössä olevan datan vuoksi. Kohinanpoistomenetelmien vertailua olisi voitu tehdä myös luokittamalla yksittäisiä kuvia tiettyyn luokkaan, jos tällainen datasetti olisi ollut käytävissä. Mahdollisesti tällaisessa luokittelussa reunojen erottuminen olisi noussut tärkeämmäksi tekijäksi homogeenisten alueiden tasoittumisen sijaan. Aiheen tutkiminen voisi tuoda näkökulmia sekä optisten kuvien että SAR-kuvien esikäsittelyyn ennen tehtävää luokittelua.

Koska tutkimustuloksista havaittiin, että heikoimmat tulokset saatiin samankaltaisissa luokissa, voisi jatkossa tehtävissä tutkimuksissa keskittyä vertailemaan kohinanpoistomenetelmän valintaa ja luokittelun onnistumista näiden näkökulmasta. Lisäksi edellisessä luvussa esitetty pohdinta koulutusalueiden rajaamisesta alueiden reunoilta olisi hyvä ottaa tutkimukseen mukaan, sillä tällä arvellaan olevan vaikutusta tarkemmin rajaavien suodattimien eduksi. Mahdollisesti aiheesta ja aiheeseen läheisesti liittyvistä asioista saadaan tulevana vuosina enenevässä määrin tutkimustietoa, mikä voi luoda uusia näkökulmia kohinanpoistomenetelmien vertailuun. Yksi tällainen aihe voi olla takaisinsironnan tutkiminen erilaisilta alueilta, sekä sato- ja metsätyyppien tunnistaminen toisistaan.

Lähteet

- Abdikan, S., Balik Sanli, F., Üstüner, M. & Calò, F. 2016. Land Cover Mapping Using Sentinel-1 SAR Data. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLI-B7, 757–761. Viitattu 25.2.2022. https://www.researchgate.net/publication/304338234_LAND_COVER_MAPPING_USING_SENTINEL-1_SAR_DATA.
- Akl, A., Tabbara, K. & Yaacoub, C. 2012. Edge-based suboptimal Kuan filtering for speckle noise reduction. *International Conference on Engineering and Technology (ICET)*, 1–5. Viitattu 2.4.2022. <https://ieeexplore.ieee.org/document/6395073>, IEEE Xplore.
- Ananto, W. H. G, Putri, A. F. S., Hadi, H. A., Hanum, D. N, Wiryawan, B. K. P, Prabaswara, R. R. & Arjasakusuma, S. 2019. Performance of various speckle filter methods in modelling forest above-ground biomass using Sentine-1 data: case study of Barru Regency, South Sulawesi. *6th Geoinformation Science Symposium. Proceedings of SPIE*, vol 11311. Viitattu 31.1.2022. https://scholar.google.com/scholar_url?url=https://www.spiedigitallibrary.org/conference-proceedings-of-spie/11311/113110P/Performance-of-various-speckle-filter-methods-in-modelling-forest-aboveground/10.1117/12.2549036.pdf&hl=en&sa=T&oi=ucasa&ct=ufr&ei=bU-JIYsGiH4TQmAGJv574Cg&scisig=AAGBfm17VbQ5o1AsSA0-70Uco_Pof1SKtA.
- Ansari, J., Ghosh, S. M., Behera M. D., Gupta, S. K. 2020. A Study on speckle Removal Techniques for Sentinel-1A SAR Data Over Sundarbans, Mangrove Forest, India. *IEEE India Geoscience and Remote Sensing Symposium (InGARSS)*, 90–93. Viitattu 4.2.2022. <https://ieeexplore.ieee.org/document/9358929>, IEEE Xplore.
- Argenti, F., Lapini, A., Bianchi., T. & Alparone, L. 2013. A Tutorial on Speckle Reduction in Synthetic Aperture Radar Images. *IEEE Geoscience and Remote Sensing Magazine*, 1, 3, 6–35. Viitattu 5.2.2022. <https://ieeexplore.ieee.org/document/6616053>, IEEE Xplore.
- ASAR Product Handbook. 2007. European Space Agency. Viitattu 17.1.2022. <https://earth.esa.int/eogateway/documents/20142/37627/ASAR-Product-Handbook.pdf>.
- Banerjee, S. & Chaudhuri, S. S. 2018. A Review on various Speckle Filters used for despeckling SAR Images. *Second International conference on computing Methodologies and Communication (IC-CMC)*, 68–73. Viitattu 30.1.2022. <https://ieeexplore.ieee.org/document/8487958>, IEEE Xplore.
- Banerjee, S., Chaudhuri, S. S, Mehra, R. & Misra, A. 2020. A Comprehensive Survey on Frost Filter and its Proposed Variants. *5th International Conference on communication and Electronics Systems (ICCES)*, 109–114. Viitattu 30.1.2022. <https://ieeexplore.ieee.org/document/9137869>, IEEE Xplore.
- Braun, A. 2020. SAR-based landcover classification with Sentinel-1 GRD products. *Opas Sentinel-1 Toolboxilla tehtävään maanpinnan luokitteluun*. European Space Agency. Viitattu 11.1.2022. <http://step.esa.int/docs/tutorials/S1TBX%20Landcover%20classification%20with%20Sentinel-1%20GRD.pdf>.

- Braun, A. & Veci, L. 2021. TOPS Interferometry Tutorial. Opas Sentinel-1 Toolboxilla tehtävään interferometriaan. European Space Agency. Viitattu 25.3.2022. https://step.esa.int/docs/tutorials/S1TBX%20TOPSAR%20Interferometry%20with%20Sentinel-1%20Tutorial_v2.pdf.
- Bouchemakh, L., Smara, Y., Hamadache, Z. 2008. A Comparative Study of Speckle Filtering In Polarimetric Radar SAR Images. 3rd International Conference on Information and communication Technologies: From Theory to Applications, 1–6. Viitattu 4.2.2022. <https://ieeexplore.ieee.org/document/4530040>, IEEE Xplore.
- Calibration Operator. N.d. Ohje Sentinel Application Platform (SNAP)-ohjelman Help -käyttöoppaassa. European Space Agency.
- Copernicus Sentinel-1 Infographic. 2021. Sentinel-1-ohjelman infografiikka Euroopan avaruusjärjestön verkkosivuilla. Viitattu 20.2.2022. <https://sentinels.copernicus.eu/web/sentinel/-/copernicus-sentinel-1-infographic>.
- Data Products. N.d. Sentinel-1-satelliittien tuottaman datan kuvaus Euroopan avaruusjärjestön verkkosivuilla. Viitattu 20.2.2022. <https://sentinels.copernicus.eu/web/sentinel/missions/sentinel-1/data-products>.
- Dumoulin, V. & Visin, F. 2018. A guide to convolution arithmetic for deep learning. arXiv. Viitattu 16.9.2021. <https://arxiv.org/pdf/1603.07285.pdf>.
- Fracastoro, G., Maggii, E., Poggi, G., Scarpa, G., Valsesia, D. & Verdilova, L. 2021. Deep Learning Methods For Synthetic Aperture Radar Image Despeckling: An Overview Of Trends And Perspectives. IEEE Geoscience and Remote Sensing Magazine, 9, 2, 29–51. Viitattu 21.2.2022. <https://arxiv.org/pdf/2012.05508.pdf>.
- Gaetano, R., Ienco, D., Ose, K. & Cresson, R. 2018. A Two-Branch CNN Architecture for Land Cover Classification of PAN and MS Imagery. Remote Sensing, 10, 11, 1746. Viitattu 12.2.2022. <https://www.mdpi.com/2072-4292/10/11/1746>.
- Gaupe, D. 2007. Principles of artificial neural networks. 2. painos. Singapore: World Scientific Publishing Company. <https://janet.finna.fi>, ProQuest Ebook Central.
- Géron, A. 2017. Hands-On Machine Learning with Scikit-Learn & TensorFlow. Sepastopol, CA: O'Reilly Media, Inc.
- Gomez, M. C. 2018. RUS Webinar: Deforestation Monitoring with Sentinel-1 – LAND07. Webinaari. RUS Copernicus. Viitattu 25.2.2022. <https://www.youtube.com/watch?v=W2wRO8lLgNU>.
- Gomez, M. C. 2020 Processing Sentinel-1 Data using snappy. Webinaari. RUS Copernicus. Viitattu 27.2.2022. <https://www.youtube.com/watch?v=PiU68g3WRIY&t=3280s>.
- Gomez, L., Buemi, M. E., Jacobo-Berlles, J. C. & Mejail, M. E. 2016. A New image quality Index for Objectively evaluating Despeckling Filtering in SAR Images. IEEE Journal of Selected topics in Applied Earth Observations and Remote Sensing, 9, 3, 1297–1307. Viitattu 30.1.2022. <https://ieeexplore.ieee.org/document/7208801>, IEEE Xplore.

Gulati, V. & Raheja, N. 2021. Efficiency Enhancement of Machine Learning Approaches through the Impact of Preprocessing Techniques. 6th International Conference on Signal Processing, Computing and Control (ISPC), 191–196. Viitattu 16.1.2022. <https://ieeexplore.ieee.org/document/9609474>, IEEE Xplore.

Honzátko, D., Bigdeli, S. A., Türetken, E. & Dunbar L. A. 2020. Efficient Blind-Spot Neural Network Architecture for Image Denoising. 7th Swiss Conference on Data Science (SDC), 59-60. Viitattu 25.2.2022. <https://ieeexplore.ieee.org/document/9145025>, IEEE Xplore.

Horé, A. & Ziou, D. 2010. Image Quality Metrics: PSNR vs. SSIM. 20th International conference on Pattern Recognition, 2366-2369. Viitattu 30.1.2022. <https://ieeexplore.ieee.org/document/5596999>, IEEE Xplore.

Instrument Payload. N.d. Sentinel-1-satelliittien esittelyteksti Euroopan avaruusjärjestön verkkosivuilla. Viitattu 20.2.2022. <https://sentinels.copernicus.eu/web/sentinel/missions/sentinel-1/instrument-payload#:~:text=SENTINEL%2D1%20carries%20a%20single,s%20X%2Dband%20down-link%20capacity>.

IW GRD Resolutions. N.d. IW GRD datatuotteiden kuvaus Euroopan avaruusjärjestön verkkosivuilla. Viitattu 21.2.2022. <https://sentinels.copernicus.eu/web/sentinel/technical-guides/sentinel-1-sar/products-algorithms/level-1-algorithms/ground-range-detected/iw>.

Kelleher, J. D. 2019. Deep Learning. Cambridge, MA: The MIT Press. Viitattu 25.3.2022. <https://janet.finna.fi>, Skillsoft Books ITPro.

Kirk, M. 2017. Thoughtful Machine Learning with Python. Seapastopol, CA: O'Reilly Media, Inc.

Korpihalkola, J. Syväoppiminen. Kurssimateriaali Syväoppiminen-kurssille. Jyväskylä: Jyväskylän ammattikorkeakoulu. Viitattu 16.9.2021. <https://syvaoppiminen-ttow1400-3001.pages.labranet.iamk.fi/syvaoppiminen-materiaalit/>.

Krull, A., Buchholz, T.-O & Jug, F. 2019. Noise2Void – Learning Denoising from Single Noisy Images. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2124–2132. Viitattu 12.2.2022. <https://ieeexplore.ieee.org/document/8954066>, IEEE Xplore.

Laine, M., Bamberg, J. & Jokinen, P. 2008. Tapaustutkimuksen käytäntö ja teoria. Teoksessa Tapaustutkimuksen taito. 2. painos. Helsinki: Gaudeamus Helsinki University Press, 9–38. Viitattu 17.1.2022. <https://janet.finna.fi>, Elliblibrary.

Laine, S., Karras, T., Lehtinen, J. & Aila, T. 2019. High-Quality Self-Supervised Deep Image Denoising. 33rd Conference on Neural Information Processing Systems (NeurIPS 2010). Canada, Vancouver. Viitattu 12.2.2022. <https://arxiv.org/pdf/1901.10277.pdf>.

Lamarre, M. N.d. Spatiotemporal crop type classification with Deep Learning applied to SAR time series. Artikkelit Euroopan avaruusjärjestön sivuilla. European Space Agency. Viitattu 25.4.2022.

<https://philab.phi.esa.int/usecases/spatiotemporal-crop-type-classification-with-deep-learning-applied-to-sar-time-series/>.

Lapini, A., Bianchi, T., Argenti, F. & Alparone, L. 2014. Blind Speckle Decorrelation for SAR Image Despeckling. Viitattu 13.2.2022. <https://ieeexplore.ieee.org/document/6487399>, IEEE Xplore.

Lasaponara, R. & Tucci, B. 2019. Identification of Burned Areas and Severity Using SAR Sentinel-1. IEEE Geoscience and Remote Sensing Letters, 16, 6, 917–921. Viitattu 9.5.2022. <https://ieeexplore.ieee.org/document/8697110>, IEEE Xplore.

Lee, J. -S. 1981. Refined filtering of image noise using local statistics. Computer Vision, Graphics, and Image Processing, 15, 380-389.

Lee, J. -S. 1983. A simple speckle smoothing algorithm for synthetic aperture radar images. IEEE Transactions on Systems, Man, and Cybernetics, SMC-13, 1, 85–89. Viitattu 19.2.2022. <https://ieeexplore.ieee.org/document/6313036>, IEEE Xplore.

Lee, J. -S., Ainsworth, T. L., Wang, Y. & Chen, K. -S. 2015. Polarimetric SAR Speckle Filtering and the Extended Sigma Filter. IEEE Transactions on Geoscience and Remote Sensing, 53, 3, 1150–1160. Viitattu 4.2.2022. <https://ieeexplore.ieee.org/document/6866154>, IEEE Xplore.

Lee, J.-S., Ainsworth, T. L. & Wang, Y. 2017. A review of polarimetric SAR speckle filtering. IEEE International Geoscience and Remote Sensing Symposium (IGARSS), 5303-5306. Viitattu 21.2.2022. <https://ieeexplore.ieee.org/document/8128201>, IEEE Xplore.

Lee, J. -S., Grunes, M. R & de Grandi, G. 1999. Polarimetric SAR speckle filtering and its implication for classification. IEEE Transactions on Geoscience and Remote Sensing, 37, 5, 2363–2373. Viitattu 31.1.2022. <https://ieeexplore.ieee.org/document/789635>, IEEE Xplore.

Lee, J. -S., Jurkevich, I., Dewaele, P., Wambacq, P. & Oosterlinck, A. 1994. Speckle Filtering of Synthetic Aperture Radar Images: A Review. Remote Sensing Reviews, 8, 4, 313–340.

Lee, J.-S., Wen, J.-H., Ainsworth, T. L., Chen, K.-S. & Chen, A. J. 2009. Improved Sigma Filter for Speckle Filtering of SAR Imagery. IEEE Transactions on Geoscience and Remote Sensing, 47, 1, 202–213. Viitattu 20.2.2022. <https://ieeexplore.ieee.org/document/4689358>, IEEE Xplore.

Level-0. N.d. Level-0-tason datatuotteiden kuvaus Euroopan avaruusjärjestön verkkosivuilla. Viitattu 20.2.2022. <https://sentinel.esa.int/web/sentinel/user-guides/sentinel-1-sar/product-types-processing-levels/level-0>.

Level-1 GRD Products. N.d. Level-1 GRD -datatuotteiden kuvaus Euroopan avaruusjärjestön verkkosivuilla. Viitattu 20.2.2022. <https://sentinels.copernicus.eu/web/sentinel/technical-guides/sentinel-1-sar/products-algorithms/level-1-algorithms/ground-range-detected>.

Level-1 SLC Products. N.d. Level-1 SLC -datatuotteiden kuvaus Euroopan avaruusjärjestön verkkosivuilla. Viitattu 20.2.2022. <https://sentinels.copernicus.eu/web/sentinel/technical-guides/sentinel-1-sar/products-algorithms/level-1-algorithms/single-look-complex>.

Level-2. N.d. Level-2-tason datatuotteiden kuvaus Euroopan avaruusjärjestön verkkosivuilla. Viitattu 20.2.2022. <https://sentinel.esa.int/web/sentinel/user-guides/sentinel-1-sar/product-types-processing-levels/level-2>.

Lopez, A., Nezry, E., Touzi, R. & Laur, H. 1990 Maximum A Posteriori Speckle Filtering And First Order Texture Models In SAR Images. 10th Annual International symposium on Geoscience and Remote Sensing, 2409–2412. Viitattu 26.3.2022. <https://ieeexplore.ieee.org/document/689026>, IEEE Xplore.

Mansoupour, M. & Blais, R. 2006. Effects and performance of speckle noise reduction filters on active radar and SAR images. Proc. ISPRS. Viitattu 2.4.2022. https://www.researchgate.net/publication/239959635_Effects_and_performance_of_speckle_noise_reduction_filters_on_active_radar_and_SAR_images.

Masoomi, A., Hamzehyan, R. & Shirazi, N. C. 2021. International Journal of Machine Learning and Computing, 2,1, 62–70. Viitattu 31.1.2022. <http://www.ijmlc.org/papers/90-L0007.pdf>.

McVittie, A. 2019. Flood mapping using Snappy. Opas Snappy-moduulilla tehtävään tulvakartoitukseen. European Space Agency. Viitattu 26.2.2022. https://dges.carleton.ca/courses/Intro-SAR/LAB%20A%20-%20Carleton%20SAR%20Training%20-%20SAR_processing_FINAL.pdf.

Medasani, S. & Reddy, G. U. 2018. Analysis and Evaluation of Speckle Filtering Using Polarimetric Synthetic Aperture Radar Through Local Statistics. Second International Conference on Electronics, Communication and Aerospace Technology (ICECA), 169–174. Viitattu 31.1.2022. <https://ieeexplore.ieee.org/document/8474567>, IEEE Xplore.

Mercier, A., Betbeder, J., Rumiano, F., Baudry, J., Gond, V., Blanc, L., Bourgoïn, C., Cornu, G., Ciudad, C., Marchamalo, M., Pocard-Chaipus, R. & Hubert-Moy, L. 2019. Evaluation of Sentinel-1 and 2 Time Series for Land Cover Classification of Forest-Agriculture Mosaics in Temperate and Tropical Landscapes. Remote Sensing, 11, 8. Viitattu 25.2.2022. <https://www.mdpi.com/2072-4292/11/8/979>.

Meyer, F. 2019. Spaceborne Synthetic Aperture Radar: Principles, Data Access, and Basic Processing Techniques. Julkaisussa The Synthetic Aperture Radar (SAR) Handbook: Comprehensive Methodologies for Forest Monitoring and Biomass Estimation. Viitattu 16.1.2022. https://gis1.serviglobal.net/TrainingMaterials/SAR/SARHB_FullRes.pdf.

Michellucci, U. 2018. Applied Deep Learning: A Case-Based Approach to Understanding Deep Neural Networks. New York: Apress. Viitattu 25.3.2022. <https://janet.finna.fi>, Skillssoft.

- Molini, A. B., Valsesia, D., Fracastoro, G., Magli, E. 2020. Towards Deep Unsupervised Sar Despckling with Blind-Spot Convolutional Neural Networks. ArXiv. Viitattu 13.2.2022. <https://arxiv.org/pdf/2001.05264.pdf>.
- Molini, A. B., Valsesia, D. Fracastoro, G. & Magli, E. 2022. Speckle2Void: Deep Self-Supervised SAR Despeckling With Blind-Spot Convolutional Neural Networks. IEEE Transactions on Geoscience and Remote Sensing, 60, 1–17. Viitattu 13.2.2022. <https://ieeexplore.ieee.org/document/9383788>, IEEE Xplore.
- Mueller, J. P. & Massaron, L. 2019. Deep Learning for Dummies. Hoboken, New Jersey: John Wiley & Sons, Inc. Viitattu 25.3.2022. <https://janet.finna.fi>, Skillssoft.
- Multilook Operator. N.d. Ohje Sentinel Application Platform (SNAP)-ohjelman Help -käyttöoppaassa. European Space Agency.
- Multi-Temporal Speckle Filter Operator. N.d. Ohje Sentinel Application Platform (SNAP)-ohjelman Help -käyttöoppaassa. European Space Agency.
- Nezry, E. 2014. Adaptive Speckle Filtering in Radar Imagery. London, United Kingdom: IntechOpen. Viitattu 2.4.2022. <https://www.intechopen.com/chapters/46699>.
- Orliková, L. & Horák, J. 2019. Land Cover Classification Using sentinel-1 SAR Data. International Conference on Military Technologies (ICMT), 1–5. Viitattu 25.2.2022. <https://ieeexplore.ieee.org/document/8870125>, IEEE Xplore.
- Ozdarici, A. & Akyurek, Z. 2020. A Comparison of SAR filtering techniques on agricultural area identification. San Diego, California. ASPRS 2010 Annual Conference. Viitattu 23.4.2022. <http://www.asprs.org/a/publications/proceedings/sandiego2010/sandiego10/Ozdarici.pdf>.
- Pal K. K., Sudeep K. S. 2016. Preprocessing for image classification by convolutional neural networks. IEEE International, Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), 1778–1781. Viitattu 16.1.2022. <https://ieeexplore.ieee.org/document/7808140>, IEEE Xplore.
- Parhad, S. V., Aher, S. A. & Warhade, K. K. 2021. A Comparative Analysis of Speckle Noise Removal in SAR Images. 2nd Global Conference for Advancement in technology (GCAT), 1–4. Viitattu 4.2.2022. <https://ieeexplore.ieee.org/document/9587458>, IEEE Xplore.
- Paula-myrsky teki Koillismaan metsiin miljoonavahingot. 2021. Uutinen Metsäkeskuksen verkkosivuilla 24.6.2021. Viitattu 25.3.2022. <https://www.metsakeskus.fi/en/node/1322>.
- Pinto, N. 2018. Introduction to Polarimetric SAR. NASA ARSET -webinaarisarjan 3. osa. Viitattu 1.16.2022. <https://www.youtube.com/watch?v=-xU4oE66pgY>.
- Podest, E. 2018. Basics of Synthetic Aperture Radar (SAR). NASA ARSET -webinaarisarjan 1. osa. Viitattu 1.16.2022. <https://www.youtube.com/watch?v=Xemo2ZpduHA&t=693s>.

Podest, E., McCullum, A., Torres-Perez, J. L. & McCartney, S. 2020. Forest Mapping and Monitoring with SAR Data: Land Cover Classification with Radar and Optical Data. Viitattu 20.2.2022. <https://www.youtube.com/watch?v=raXA3gnb94Q>.

Prudente, V. H. R., Sanches, I. D., Adami, M., Skakun, S., Oldoni, L. V., Xaud, H. A. M., Xaud, M. R. & Zang, Y. 2020. Sar Data for Land Use Land Cover Classification in a Tropical Region with Frequent Cloud Cover. International Geoscience and Remote Sensing Symposium (IGARSS), 4100–4103. Viitattu 25.2.2022. <https://ieeexplore.ieee.org/document/9323404>, IEEE Xplore.

Range Doppler Terrain Correction Operator. N.d. Ohje Sentinel Application Platform (SNAP)-ohjelman Help -käyttöoppaassa. European Space Agency.

Richards, J. A. 1993. Remote Sensing digital image Analysis An Introduction. 2nd edition. Berlin, Heidelberg: Springer Berlin Heidelberg.

Santoso, W. A., Pebrianti, D., Bayuaji, L. & Zain, J. M. 2015. Performance of various speckle reduction filters on synthetic aperture Radar image. 4th International conference on software Engineering and Computer Systems (ICSECS), 11–14. Viitattu 29.1.2022. <https://ieeexplore.ieee.org/document/7333103>, IEEE Xplore.

SAR Preprocessing – Sentinel 1. 2019. Carleton University. Luento 7A. Powerpoint-diaesitys. Viitattu 2.4.2022. https://dges.carleton.ca/courses/IntroSAR/Winter2019/SECTION%207A%20-%20Carleton%20SAR%20Training%20-%20SAR_processing_Sentinal1%20-%20FINAL.pdf.

Sentinel-1: Overview. N.d. Esittelyteksti Sentinel-1-ohjelmasta Euroopan avaruusjärjestön verkkosivuilla. Viitattu 25.10.2021. <https://sentinels.copernicus.eu/web/sentinel/missions/sentinel-1/overview#:~:text=The%20Sentinel%2D1%20mission%20is,dealing%20with%20environment%20and%20security>.

Sentinel-1: Radar mission. 2014. Euroopan avaruusjärjestön esittelyvideo Sentinel-1-ohjelmasta. Viitattu 25.10.2021. <https://www.youtube.com/watch?v=FJWzLxdSMYA>.

Siltanen, S. 2018. Lecture 1 – Convolution (1D). Lentotalenne. Helsinki: Helsingin yliopisto. Viitattu 16.9.2021. <https://www2.helsinki.fi/fi/unitube/video/efb69e38-5160-4a7b-a296-f5797c294f5c>.

Smith, J. G. 2008. Organic Chemistry. 2. painos. New York: McGraw-Hill.

Souyris, J.-C. 2014. Physics of Radar Measurement. Teoksessa Remote Sensing Imagery. London, England: ISTE Ltd; Hoboken, New Jersey: John Wiley & Sons, Inc., 83–122. Viitattu 16.1.2022. <https://janet.finna.fi>, Ebook Central.

Thermal Denoising of Products Generated by the S-1 IPF. 2017. S-1 Mission Performance Centre. Viitattu 16.3.2022. <https://sentinels.copernicus.eu/documents/247904/2142675/Thermal-Denoising-of-Products-Generated-by-Sentinel-1-IPF.pdf/11d3bd86-5d6a-4e07-b8bb-912c1093bf91?t=1511973926000>.

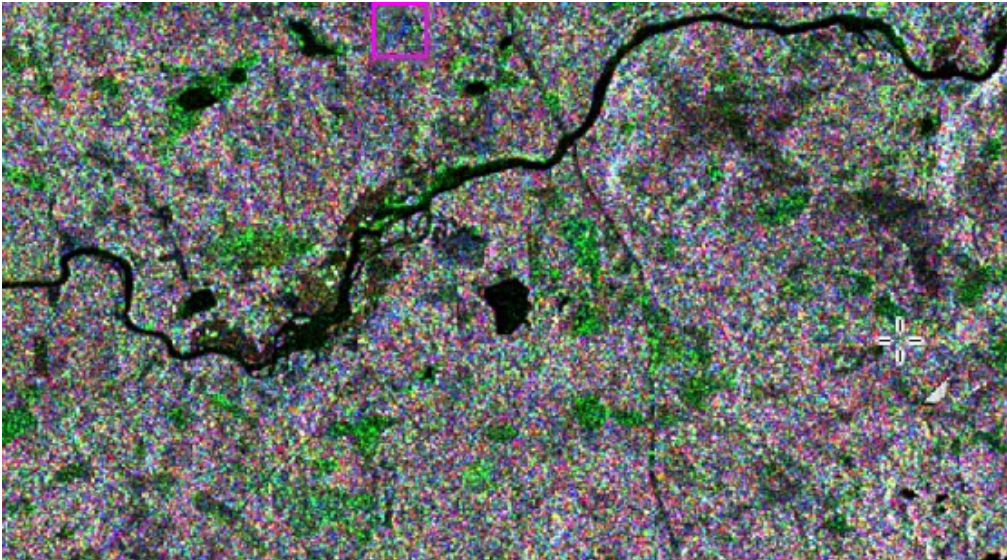
- Tyni, T. 2019. L^p spaces and convolution. Luento 7, osa 1. Helsinki: Helsingin yliopisto. Viitattu 15.9.2021. <https://www2.helsinki.fi/fi/unitube/video/811cbef2-26b1-49f3-919e-26754e831845>.
- Udali, A., Lingua, E. & Persson H. 2021. Assessing Forest Type and Tree species Classification using Sentinel-1 C-Band SAR Data in Southern Sweden. Remote Sensing. 13, 16, artikkelin numero=3237. Viitattu 25.2.2022. <https://www.mdpi.com/2072-4292/13/16/3237/htm>.
- Valsesia, D. 2020. You do not need clean images for SAR despeckling with deep learning. Artikkelit Towards Data Science -verkkojulkaisussa. Viitattu 13.2.2022. <https://towardsdatascience.com/you-do-not-need-clean-images-for-sar-despeckling-with-deep-learning-fe9c44350b69>.
- Van Zyl, J. & Kim, Y. 2011. Synthetic aperture radar polarimetry. Hoboken, New Jersey: John Wiley & Sons, Inc. Viitattu 25.1.2022. <https://janet.finna.fi>, Ebook Central.
- Varjo, S. 2015. Mitä on konvoluutio? Tutustu kuvankäsittelyyn. Työohje tieteentyöpäiville 2015. Oulu: Oulun yliopisto. Viitattu 15.9.2021. <https://docplayer.fi/23923669-Mita-on-konvoluutio-tutustu-kuvankasittelyyn.html>.
- Veci, L. 2019. Polarimetric Tutorial. Opas Sentinel-1 Toolboxilla tehtävään polarimetrisen SAR-datan käsittelyyn. Viitattu 25.3.2022. <https://step.esa.int/docs/tutorials/S1TBX%20Polarimetry%20Tutorial.pdf>.
- Venkatesan, R. & Li, B. 2017. Convolutional Neural Networks in Visual Computing: A Concise Guide. Boca Raton, Florida: CRC Press. Viitattu 25.3.2022. <https://janet.finna.fi>, EBSCOhost Ebooks.
- Vilka, H. 2017. Tutki ja mittaa. Määrällisen tutkimuksen perusteita. Viitattu 17.1.2022. https://trepo.tuni.fi/bitstream/handle/10024/98723/Tutki-ja-mittaa_2007.pdf?sequence=1&isAllowed=y.
- Woolfson, M. M. 2011. The Fundamentals of Imaging: From Particles to Galaxies. London: Imperial College Press. Viitattu 1.16.2022. <https://janet.finna.fi>, Ebook Central.
- Yahia, M., Ali, T., Mortula, M. M., Abdelfattah, R., Mahdy, S. E. & Arampola, N. S. 2020. Enhancement of SAR Speckle Denoising Using the Improved Iterative Filter. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 13, 859–871. Viitattu 4.2.2022. <https://ieeexplore.ieee.org/document/9007383>, IEEE Xplore.
- Yalçın, O. 2021. Applied Neural Networks with TensorFlow 2: API Oriented Deep Learning with Python. New York: Apress.
- Yommy, S. A., Liu R. & Wu, A. S. 2015. SAR Image Despeckling Using Refined Lee Filter. 7th International Conference on Intelligent Human-Machine Systems and Cybernetics, 260–265. Viitattu 19.2.2022. <https://ieeexplore.ieee.org/document/7334965>, IEEEXplore.

Zohary, M. N.d. Lab 3 A: SAR Preprocessing – Sentinel-1. Diaesitys. Carleton University. Viitattu 21.2.2022. https://dges.carleton.ca/courses/IntroSAR/LAB%203A%20-%20Carleton%20SAR%20Training%20-%20SAR_processing_FINAL.pdf.

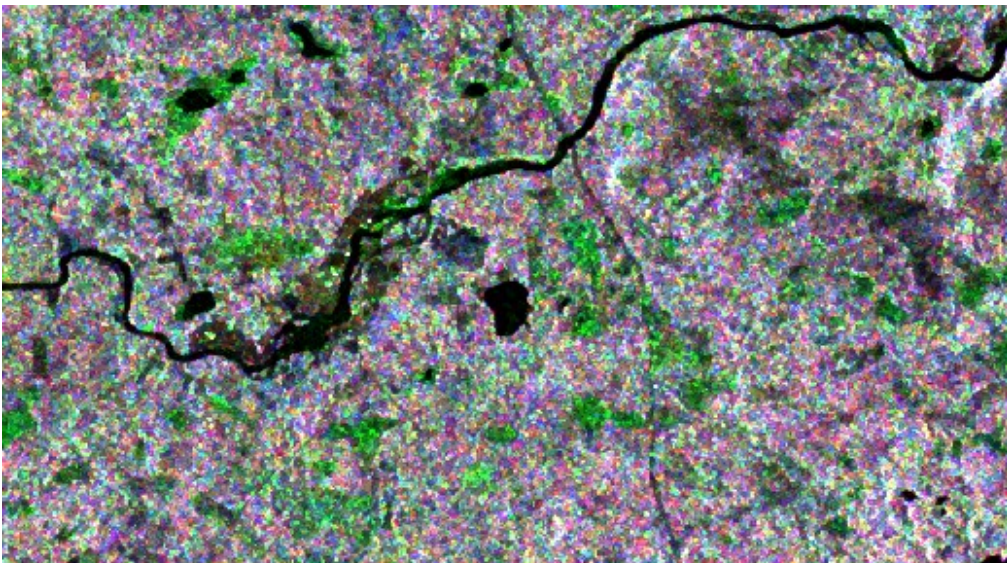
Liitteet

Liite 1. Suodatuksen tulokset RGB-kuvina

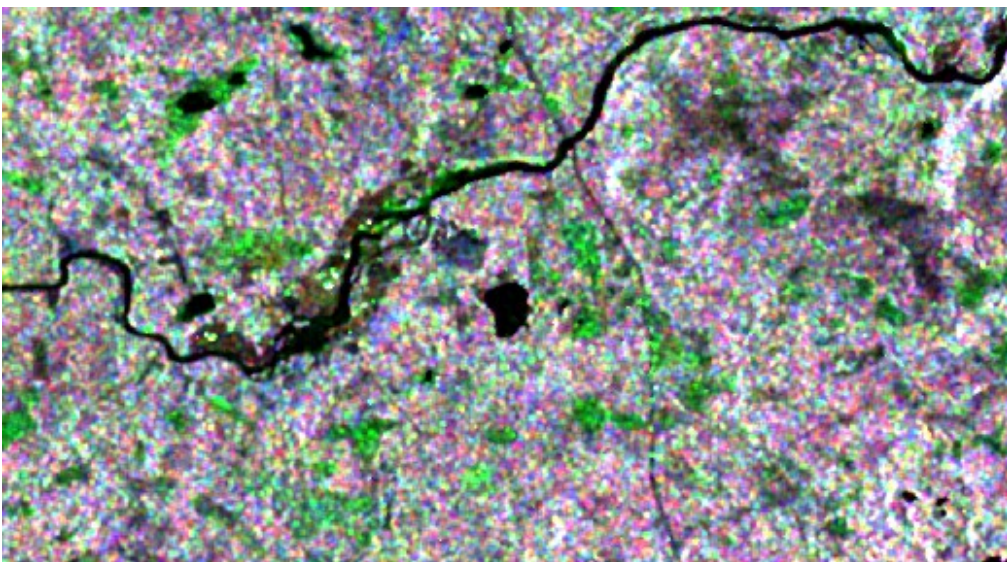
a) Suodattamaton



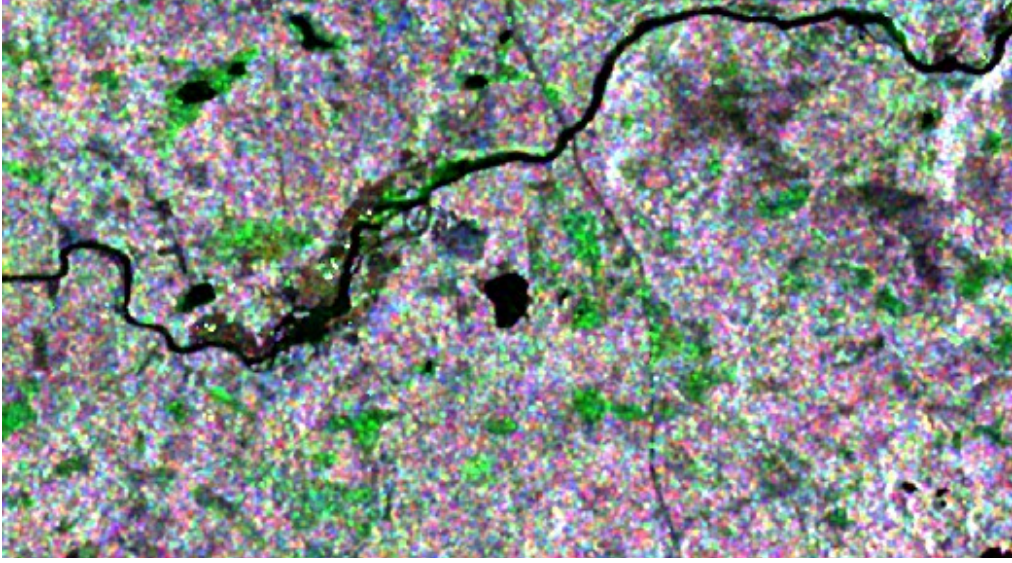
b) Refined Lee



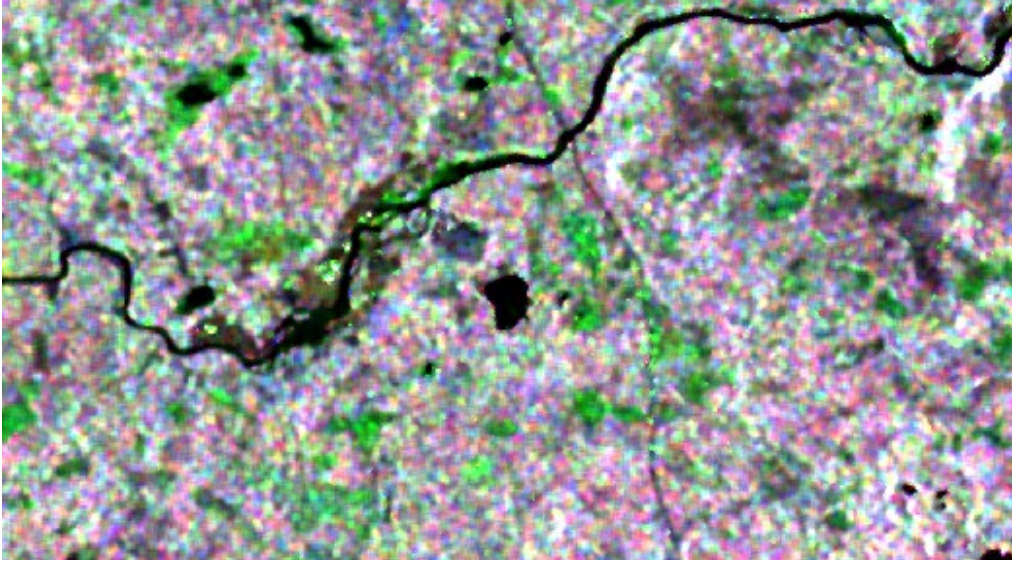
c) Lee 5x5



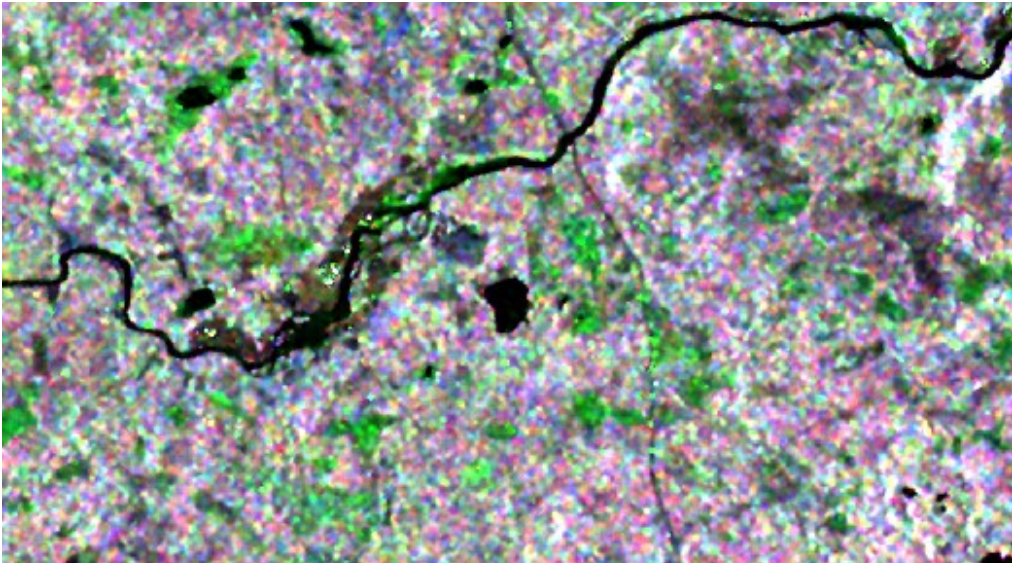
d) Frost 5x5



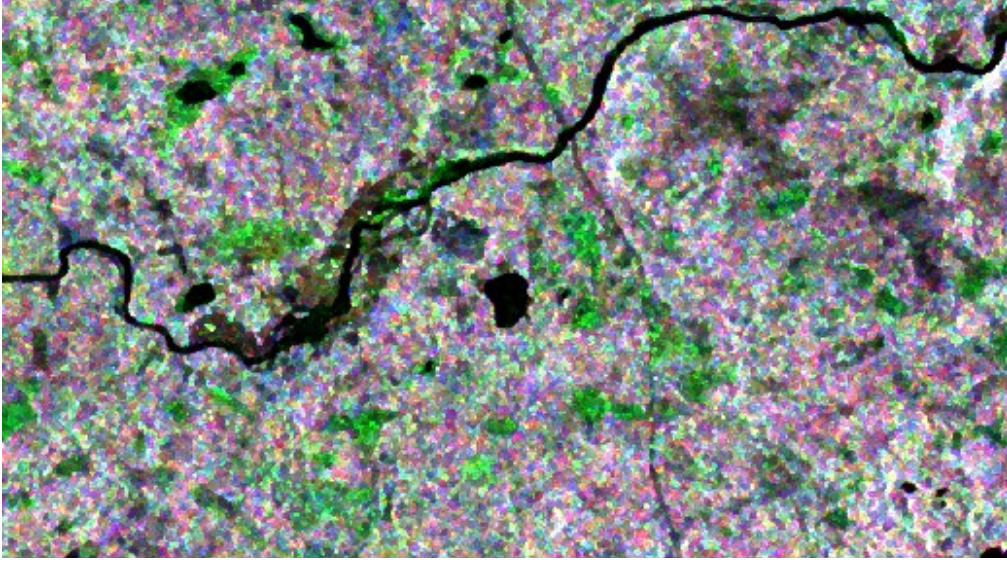
e) Lee 7x7



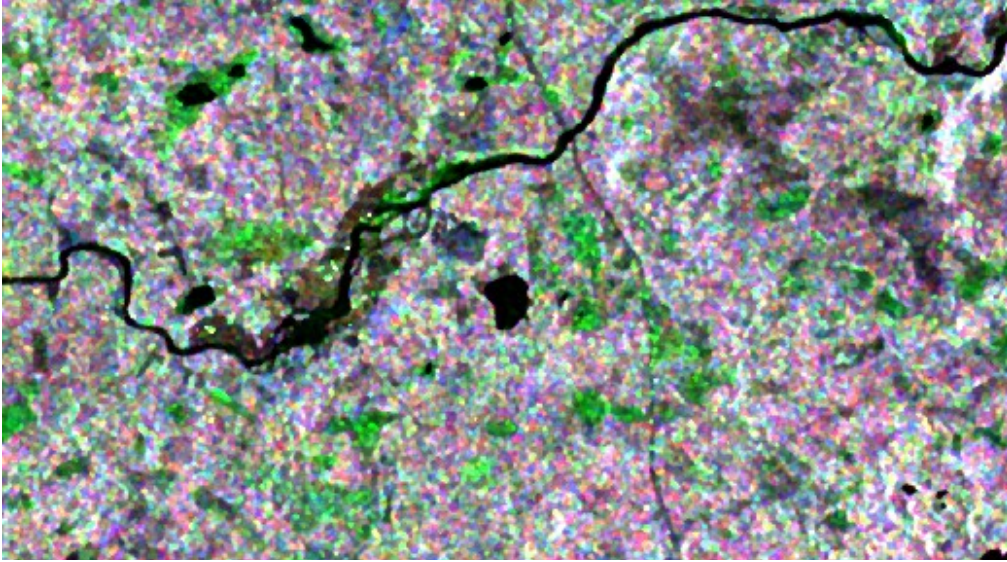
f) Frost 7x7



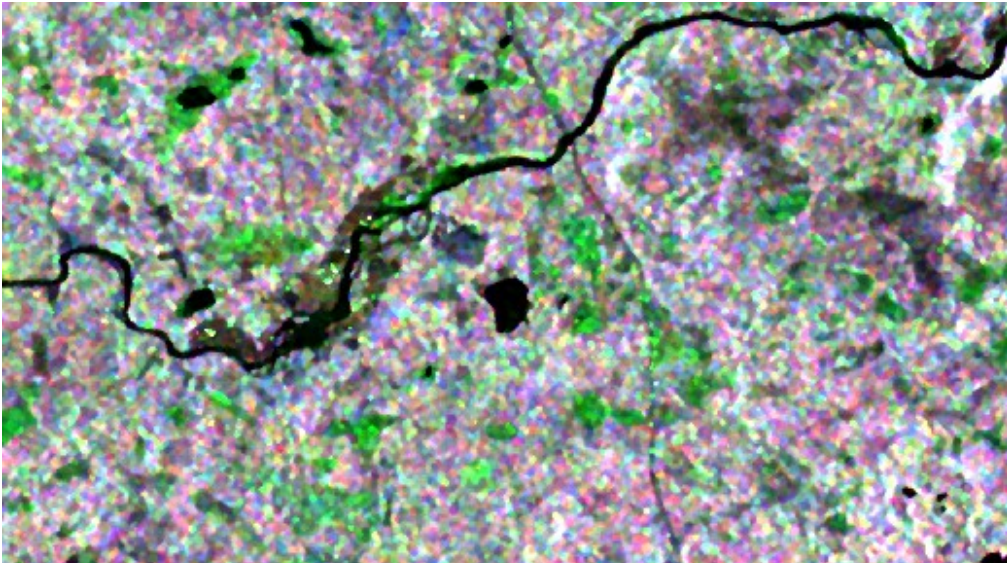
g) Refined Lee x 2



h) Frost 5x5 & Refined Lee



i) Frost 7x7 & Refined Lee



Liite 2. ENL-arvot

Taulukko 3. Valittujen alueiden ENL-arvot ennen suodatusta.

| | Intensity_VH | Intensity_VV |
|-------|--------------|--------------|
| Metsä | 3.8459349 | 4.260151 |
| Järvi | 4.8220716 | 3.7789881 |

Taulukko 4. Metsäalueen ENL-arvot suodatuksen jälkeen.

| | Sigma0_VH | Sigma0_VV |
|-------------------------|-----------|-----------|
| Lee 5x5 | 9.853814 | 15.893731 |
| Lee 7x7 | 13.546052 | 25.268162 |
| Frost 5x5 | 8.795168 | 14.194882 |
| Frost 7x7 | 11.205772 | 20.384937 |
| Refined Lee | 6.316167 | 9.191529 |
| Refined Lee x 2 | 8.501823 | 13.608615 |
| Frost 5x5 & Refined Lee | 10.100193 | 17.152021 |
| Frost 7x7 & Refined Lee | 12.060008 | 22.856209 |

Taulukko 5. Järvialueen ENL-arvot suodatuksen jälkeen.

| | Sigma0_VH | Sigma0_VV |
|-------------------------|-----------|-----------|
| Lee 5x5 | 22.070158 | 11.097392 |
| Lee 7x7 | 38.41742 | 15.4725 |
| Frost 5x5 | 15.387378 | 10.10727 |
| Frost 7x7 | 20.61728 | 13.140149 |
| Refined Lee | 12.087988 | 7.1115828 |
| Refined Lee x 2 | 22.03894 | 9.939627 |
| Frost 5x5 & Refined Lee | 22.493176 | 11.690965 |
| Frost 7x7 & Refined Lee | 28.182772 | 14.176989 |

Liite 3. Random Forest -luokittelun tulokset

Taulukko 6. Luokan ”Metsä” luokittelun tulokset.

| | Accuracy | Precision | True-Positives | False-Positives | True-Negatives | False-Negatives |
|-------------------------|----------|-----------|----------------|-----------------|----------------|-----------------|
| Ei suodatusta | 0,9474 | 0,8579 | 1183 | 196 | 3554 | 67 |
| Lee 5x5 | 0,9834 | 0,9417 | 1244 | 77 | 3673 | 6 |
| | 0,9836 | 0,9418 | 1245 | 77 | 3673 | 5 |
| | 0,9846 | 0,9481 | 1241 | 68 | 3682 | 9 |
| Lee 7x7 | 0,9890 | 0,9643 | 1241 | 46 | 3704 | 9 |
| | 0,9892 | 0,9628 | 1244 | 48 | 3702 | 6 |
| | 0,9912 | 0,9704 | 1244 | 38 | 3712 | 6 |
| Frost 5x5 | 0,9804 | 0,9299 | 1246 | 94 | 3656 | 4 |
| | 0,9808 | 0,9325 | 1244 | 90 | 3660 | 6 |
| | 0,9816 | 0,9360 | 1243 | 85 | 3665 | 7 |
| Frost 7x7 | 0,9874 | 0,9548 | 1246 | 59 | 3691 | 4 |
| | 0,9858 | 0,9517 | 1242 | 63 | 3687 | 8 |
| | 0,9952 | 0,9812 | 1250 | 24 | 3726 | 0 |
| Refined Lee | 0,9762 | 0,9300 | 1223 | 92 | 3658 | 27 |
| | 0,9736 | 0,9100 | 1239 | 121 | 3629 | 11 |
| | 0,9961 | 0,9719 | 1244 | 36 | 3714 | 6 |
| Refined Lee x 2 | 0,9788 | 0,9256 | 1244 | 100 | 3650 | 6 |
| | 0,9804 | 0,9337 | 1240 | 88 | 3662 | 10 |
| | 0,9886 | 0,9592 | 1246 | 53 | 3697 | 4 |
| Frost 5x5 & Refined Lee | 0,9838 | 0,9405 | 1248 | 79 | 3671 | 2 |
| | 0,9900 | 0,9644 | 1246 | 46 | 3704 | 4 |
| | 0,9844 | 0,9453 | 1244 | 72 | 3678 | 6 |
| Frost 7x7 & Refined Lee | 0,9892 | 0,9614 | 1246 | 50 | 3700 | 4 |
| | 0,9902 | 0,9659 | 1245 | 44 | 3706 | 5 |
| | 0,9902 | 0,9719 | 1246 | 36 | 3714 | 4 |

Taulukko 7. Luokan "Pelto" luokittelun tulokset.

| | Accuracy | Precision | True-Positives | False-Positives | True-Negatives | False-Negatives |
|-------------------------|----------|-----------|----------------|-----------------|----------------|-----------------|
| Ei suodatusta | 0,8958 | 0,7995 | 973 | 244 | 3506 | 277 |
| Lee 5x5 | 0,9794 | 0,9783 | 1173 | 26 | 3724 | 77 |
| | 0,9794 | 0,9799 | 1171 | 24 | 3726 | 79 |
| | 0,9642 | 0,9315 | 1156 | 85 | 3665 | 94 |
| Lee 7x7 | 0,9748 | 0,9692 | 1169 | 45 | 3705 | 81 |
| | 0,9752 | 0,9668 | 1160 | 40 | 3710 | 84 |
| | 0,9752 | 0,9600 | 1175 | 49 | 3701 | 75 |
| Frost 5x5 | 0,9626 | 0,9382 | 1138 | 75 | 3675 | 112 |
| | 0,9762 | 0,9788 | 1156 | 25 | 3725 | 94 |
| | 0,9598 | 0,9456 | 1113 | 64 | 3686 | 137 |
| Frost 7x7 | 0,9762 | 0,9601 | 1180 | 49 | 3701 | 70 |
| | 0,9714 | 0,9526 | 1165 | 58 | 3692 | 85 |
| | 0,9912 | 0,9951 | 1212 | 6 | 3744 | 38 |
| Refined Lee | 0,9484 | 0,9066 | 1106 | 114 | 3636 | 144 |
| | 0,9388 | 0,8973 | 1066 | 122 | 3628 | 184 |
| | 0,9784 | 0,9561 | 1197 | 55 | 3695 | 53 |
| Refined Lee x 2 | 0,9558 | 0,9371 | 1103 | 74 | 3676 | 147 |
| | 0,9748 | 0,9622 | 1170 | 46 | 3704 | 80 |
| | 0,9664 | 0,9486 | 1144 | 62 | 3688 | 106 |
| Frost 5x5 & Refined Lee | 0,9706 | 0,9554 | 1157 | 54 | 3696 | 93 |
| | 0,9808 | 0,9769 | 1182 | 28 | 3722 | 68 |
| | 0,9818 | 0,9801 | 1183 | 24 | 3726 | 67 |
| Frost 7x7 & Refined Lee | 0,9798 | 0,9713 | 1184 | 35 | 3715 | 66 |
| | 0,9772 | 0,9588 | 1187 | 51 | 3699 | 63 |
| | 0,9790 | 0,9720 | 1179 | 34 | 3716 | 71 |

Taulukko 8. Luokan "Suo" luokittelun tulokset.

| | Accuracy | Precision | True-Positives | False-Positives | True-Negatives | False-Negatives |
|-------------------------|----------|-----------|----------------|-----------------|----------------|-----------------|
| Ei suodatusta | 0,9214 | 0,8481 | 1044 | 187 | 3563 | 206 |
| Lee 5x5 | 0,9738 | 0,9458 | 1187 | 68 | 3682 | 63 |
| | 0,9720 | 0,9419 | 1183 | 73 | 3677 | 67 |
| | 0,9652 | 0,9283 | 1166 | 90 | 3660 | 84 |
| Lee 7x7 | 0,9768 | 0,9507 | 1196 | 62 | 3688 | 54 |
| | 0,9758 | 0,9462 | 1197 | 68 | 3682 | 53 |
| | 0,9772 | 0,9530 | 1195 | 59 | 3691 | 55 |
| Frost 5x5 | 0,9652 | 0,9297 | 1164 | 88 | 3662 | 86 |
| | 0,9730 | 0,9514 | 1175 | 60 | 3690 | 75 |
| | 0,9798 | 0,9556 | 1205 | 56 | 3694 | 45 |
| Frost 7x7 | 0,9762 | 0,9557 | 1186 | 55 | 3695 | 64 |
| | 0,9722 | 0,9462 | 1178 | 67 | 3683 | 72 |
| | 0,9810 | 0,9661 | 1197 | 42 | 3708 | 53 |
| Refined Lee | 0,9488 | 0,8907 | 1133 | 139 | 3611 | 117 |
| | 0,9502 | 0,8975 | 1130 | 129 | 3651 | 156 |
| | 0,9594 | 0,9205 | 1146 | 99 | 3651 | 104 |
| Refined Lee x 2 | 0,9626 | 0,9182 | 1167 | 104 | 3646 | 83 |
| | 0,9698 | 0,9449 | 1167 | 68 | 3682 | 83 |
| | 0,9774 | 0,9438 | 1209 | 72 | 3678 | 41 |
| Frost 5x5 & Refined Lee | 0,9760 | 0,9491 | 1194 | 64 | 3686 | 56 |
| | 0,9852 | 0,9674 | 1217 | 41 | 3709 | 33 |
| | 0,9794 | 0,9562 | 1202 | 55 | 3695 | 48 |
| Frost 7x7 & Refined Lee | 0,9800 | 0,9585 | 1202 | 52 | 3698 | 48 |
| | 0,9760 | 0,9593 | 1180 | 50 | 3700 | 60 |
| | 0,9858 | 0,9660 | 1222 | 43 | 3707 | 28 |

Taulukko 9. Luokan "Vesi" luokittelun tulokset.

| | Accuracy | Precision | True-Positives | False-Positives | True-Negatives | False-Negatives |
|-------------------------|----------|-----------|----------------|-----------------|----------------|-----------------|
| Ei suodatusta | 0,9278 | 0,8789 | 1031 | 142 | 3608 | 219 |
| Lee 5x5 | 0,9754 | 0,9600 | 1176 | 49 | 3701 | 74 |
| | 0,9734 | 0,9552 | 1172 | 55 | 3695 | 78 |
| | 0,9714 | 0,9631 | 1150 | 44 | 3706 | 100 |
| Lee 7x7 | 0,9830 | 0,9694 | 1203 | 38 | 3712 | 47 |
| | 0,9806 | 0,9660 | 1195 | 42 | 3708 | 55 |
| | 0,9800 | 0,9637 | 1195 | 45 | 3705 | 55 |
| Frost 5x5 | 0,9718 | 0,9640 | 1152 | 43 | 3707 | 98 |
| | 0,9732 | 0,9464 | 1183 | 67 | 3683 | 67 |
| | 0,9724 | 0,9506 | 1173 | 61 | 3689 | 77 |
| Frost 7x7 | 0,9786 | 0,9665 | 1184 | 41 | 3709 | 66 |
| | 0,9742 | 0,9568 | 1174 | 53 | 3697 | 76 |
| | 0,9802 | 0,9535 | 1210 | 59 | 3691 | 40 |
| Refined Lee | 0,9459 | 0,9086 | 1084 | 109 | 3641 | 166 |
| | 0,9490 | 0,9170 | 1094 | 99 | 3651 | 156 |
| | 0,9506 | 0,9101 | 1113 | 110 | 3640 | 137 |
| Refined Lee x 2 | 0,9656 | 0,9462 | 1143 | 65 | 3685 | 107 |
| | 0,9650 | 0,9402 | 1148 | 73 | 3677 | 102 |
| | 0,9736 | 0,9605 | 1166 | 48 | 3702 | 84 |
| Frost 5x5 & Refined Lee | 0,9772 | 0,9781 | 1170 | 34 | 3716 | 80 |
| | 0,9780 | 0,9597 | 1190 | 50 | 3700 | 60 |
| | 0,9784 | 0,9680 | 1181 | 39 | 3711 | 69 |
| Frost 7x7 & Refined Lee | 0,9786 | 0,9643 | 1187 | 44 | 3706 | 63 |
| | 0,9774 | 0,9574 | 1190 | 53 | 3697 | 60 |
| | 0,9792 | 0,9621 | 1193 | 47 | 3703 | 57 |

Taulukko 10.

| | % correct predictions | RMSE | Bias | 1 piirre | 2 piirre | 3 piirre |
|---------------------|-----------------------|---------------|------------------|----------|----------------|----------------|
| Ei suodatusta | 84,62 | 0.51264022471 | -0.06040000000 | Mean_VH | Homogeneity_VH | Jun_VH_db |
| Lee 5x5 | 95,00 | 0.28284271247 | -0.02320000000 | Mean_VH | Jun_VH_db | Homogeneity_VH |
| | 95,42 | 0.28284271247 | -0.02239999999 | Mean_VH | Jun_VH_db | Entropy_VH |
| Lee 7x7 | 94,26 | 0.32093613071 | -0.03299999999 | Mean_VH | Jun_VH_db | Entropy_VH |
| | 96,18 | 0.26570660511 | -0.00940000000 | Mean_VH | Entropy_VH | Jun_VH_db |
| Frost 5x5 | 96,04 | 0.26944387170 | -0.01059999999 | Mean_VH | Entropy_VH | PCA2 |
| | 96,18 | 0.28425340807 | -0.00960000000 | Mean_VH | Entropy_VH | Jun_VH_db |
| Frost 7x7 | 94,00 | 0.32124756808 | -0.03960000000 | Mean_VH | Jun_VH_db | Entropy_VH |
| | 95,16 | 0.29966648127 | -0.01980000000 | Mean_VH | Jun_VH_db | Entropy_VH |
| Frost 7x7 | 94,68 | 0.31144823004 | -0.01980000000 | Mean_VH | PCA4 | Jun_VV_db |
| | 95,92 | 0.27495454169 | -0.02279999999 | Mean_VH | Entropy_VH | Homogeneity_VH |
| Refined Lee | 95,18 | 0.29933259094 | -0.02120000000 | Mean_VH | Entropy_VH | Homogeneity_VH |
| | 97,38 | 0.24939927826 | 5.99999999999E-4 | Mean_VH | Entropy_VH | MAX_VH |
| Refined Lee | 90,92 | 0.44294469180 | -0.03140000000 | Mean_VH | Jun_VH_db | Homogeneity_VH |
| | 90,58 | 0.42918527467 | -0.04299999999 | Mean_VH | Jun_VH_db | PCA2 |
| Refined Lee x 2 | 94,00 | 0.36905284174 | -0.01780000000 | Mean_VH | Jun_VH_db | Homogeneity_VH |
| | 93,14 | 0.34957116585 | -0.03140000000 | Mean_VH | Homogeneity_VH | PCA4 |
| Frost 5x5 & Refined | 94,50 | 0.24812400792 | -0.03020000000 | Mean_VH | Jun_VH_db | Homogeneity_VH |
| | 95,30 | 0.29325756597 | -0.29325756597 | Mean_VH | PCA2 | Jun_Vh_db |
| Lee | 95,38 | 0.29086079144 | -0.03220000000 | Mean_VH | Jun_VV_db | Jun_VH_db |
| | 96,70 | 0.24248711305 | -0.01079999999 | Mean_VH | PCA4 | Jun_VH_db |
| Frost 7x7 & Refined | 96,20 | 0.23771486138 | -0.02380000000 | Mean_VH | Entropy_VH | Jun_VV_db |
| | 96,38 | 0.27422618401 | -0.01600000000 | Mean_VH | Entropy_VH | PCA2 |
| Lee | 96,04 | 0.27820855486 | -0.01459999999 | Mean_VH | Entropy_VH | Jun_VH_db |
| | 96,80 | 0.24779023386 | -0.00740000000 | Mean_VH | Jun_VH_db | PCA2 |

Liite 4. Kohinanpoistoon kulunut aika

Taulukko 11. Kohinanpoistomenetelmien ajallinen vertailu

| | 1. ajo | 2. ajo | 3. ajo |
|---------------------|-----------|-----------|-----------|
| Lee 5x5 | 0.0771129 | 0.0149975 | 0.0360947 |
| Lee 7x7 | 0.0079961 | 0.0089996 | 0.0130107 |
| Frost 5x5 | 0.0129974 | 0.0109968 | 0.0180066 |
| Frost 7x7 | 0.0089967 | 0.0089746 | 0.0140063 |
| Refined Lee | 0.0089941 | 0.0100033 | 0.0130002 |
| Refined Lee x 2 | 0.0200069 | 0.0199926 | 0.0259964 |
| Frost 5x5 & Refined | 0.0180006 | 0.0189993 | 0.0230036 |
| Frost 7x7 & Refined | 0.0179987 | 0.0179932 | 0.0258262 |

Liite 5. SAR-kuvien esikäsittely snappy-moduulilla

```
In [1]: import numpy as np
import matplotlib
import matplotlib.pyplot as plt
import os
```

```
In [2]: import sys
sys.path.append('C:\\Users\\TarjaAjo\\.snap\\snap-python')
```

```
In [3]: import snappy
from snappy import Product
from snappy import ProductIO
from snappy import ProductUtils
from snappy import WKTRReader
from snappy import HashMap
from snappy import GPF
import subprocess
```

```
In [4]: import preprocessingFunctions as preF
import plotFunctions
import time
```

```
In [5]: data_0708 = 'C:/Users/TarjaAjo/Satelliittidata/satellite-images-paula/S1A_IW_GRDH_15
product = ProductIO.readProduct(data_0708)
```

```
In [6]: preF.printInfo(product)

Name: S1A_IW_GRDH_1SDV_20210708T153345_20210708T153410_038688_04908F_1A4C
Width: 25632 px
Height: 16642 px
Band names: Amplitude_VH, Intensity_VH, Amplitude_VV, Intensity_VV
```

```
In [7]: prod_orb = preF.applyOrbitFile(product)
```

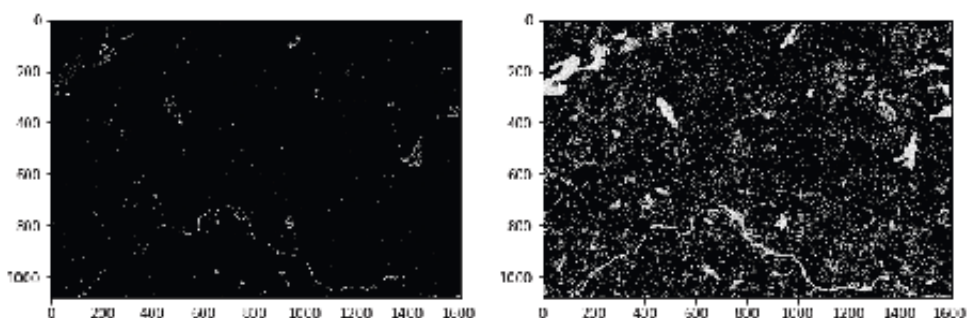
```
In [8]: #Koordinaattien järjestys
# west Long north Lat, east Long north Lat, east Long south Lat, west Long south Lat
wkt = "POLYGON((27.432 65.338, 27.747 65.338, 27.747 65.268, 27.432 65.268, 27.432 65.338))"
prod_sub = preF.subset(wkt, prod_orb)
```

```
In [9]: preF.printInfo(prod_sub)

Name: Subset_S1A_IW_GRDH_1SDV_20210708T153345_20210708T153410_038688_04908F_1A4C_Orb
Width: 1606 px
Height: 1083 px
Band names: Amplitude_VH, Intensity_VH, Amplitude_VV, Intensity_VV
```

```
In [10]: prod_ther = preF.thermalNoiseRemoval(prod_sub)
```

```
In [11]: bands = ['Intensity_VV', 'Intensity_VH']
plotFunctions.plot_bands(prod_ther, bands, 12,12,1,2,0,10000)
```



```
In [12]: prod_calib = preF.calibration(prod_ther)
```

Vertailuun valitut menetelmät:

- Lee:
 - ikkunan koot 5x5 ja 7x7
- Refined Lee (ikkunan koko on vakio 7x7)
- Refined Lee sovellettu kahdesti
- Frost
 - vaimennuskertoimen 2 (pieni vaimennuskertoimen poistaa kohinaa tehokkaammin)
 - ikkunan koot 5x5 ja 7x7
- Refined Leen ja Frostin yhdistelmä
 - ikkunan koot 5x5 ja 7x7

```
In [13]: # Kohinan poistaminen (Speckle filtering) Lee 5x5
start_time = time.time()

prod_lee_55 = preF.speckleFilterLee(prod_calib,5,5)

time_lee_55 = (time.time() - start_time)
```

```
In [14]: # Kohinan poistaminen (Speckle filtering) Lee 7x7

start_time = time.time()

prod_lee_77 = preF.speckleFilterLee(prod_calib,7,7)

time_lee_77 = (time.time() - start_time)
```

```
In [15]: # Kohinan poistaminen (Speckle filtering) Refined Lee

start_time = time.time()

prod_refinedLee = preF.speckleFilterRefinedLee(prod_calib)

time_ref = (time.time() - start_time)
```

```
In [16]: # Kohinan poistaminen (Speckle filtering) Refined Lee x 2

start_time = time.time()
```

```

prod_Ref_1 = preF.speckleFilterRefinedLee(prod_calib)
prod_RefinedLee_2 = preF.speckleFilterRefinedLee(prod_Ref_1)

time_ref_2 = (time.time() - start_time)

```

```

In [17]: # Kohinan poistaminen (Speckle filtering) Frost, ikkunan koko 5x5, vaimennuskerroin
start_time = time.time()

prod_Frost_55 = preF.speckleFilterFrost(prod_calib,2,5,5)

time_frost_55 = (time.time() - start_time)

```

```

In [18]: # Kohinan poistaminen (Speckle filtering) Frost, ikkunan koko 7x7, vaimennuskerroin
start_time = time.time()

prod_Frost_77 = preF.speckleFilterFrost(prod_calib,2,7,7)

time_frost_77 = (time.time() - start_time)

```

```

In [19]: # Kohinan poistaminen (Speckle filtering) Refined Lee & Frost, ikkunan koko 5x5, vai
start_time = time.time()

prod_Frost_and_Ref_55 = preF.speckleFilterFrost(prod_calib,2,5,5)
prod_Frost_and_Ref_55 = preF.speckleFilterRefinedLee(prod_Frost_and_Ref_55)

time_frost_55_and_ref = (time.time() - start_time)

```

```

In [20]: # Kohinan poistaminen (Speckle filtering) Refined Lee & Frost, ikkunan koko 7x7, vai
start_time = time.time()

prod_Frost_and_Ref_77 = preF.speckleFilterFrost(prod_calib,2,5,5)
prod_Frost_and_Ref_77 = preF.speckleFilterRefinedLee(prod_Frost_and_Ref_77)

time_frost_77_and_ref = (time.time() - start_time)

```

```

In [21]: print('Käsittelyajat:')
print('Lee 5x5: ', time_lee_55)
print('Lee 7x7: ', time_lee_77)
print('Frost 5x5: ', time_frost_55)
print('Frost 7x7: ', time_frost_77)
print('Refined Lee: ', time_ref)
print('Refined Lee x 2: ', time_ref_2)
print('Frost 5x5 & Refined Lee: ', time_frost_55_and_ref)
print('Frost 7x7 & Refined Lee: ', time_frost_77_and_ref)

```

```

Käsittelyajat:
Lee 5x5:  0.014997482299804688
Lee 7x7:  0.008988618850708008
Frost 5x5:  0.010996818542480469
Frost 7x7:  0.008974552154541016
Refined Lee:  0.010003328323364258
Refined Lee x 2:  0.019992589950561523
Frost 5x5 & Refined Lee:  0.018999338150024414
Frost 7x7 & Refined Lee:  0.01799321174621582

```

Geometrysten vääristymien poistaminen (Terrain correction)

```
In [22]: prod_Lee_55_ter = preF.terrainCorrection(prod_Lee_55)
prod_Lee_77_ter = preF.terrainCorrection(prod_Lee_77)
prod_RefinedLee_ter = preF.terrainCorrection(prod_RefinedLee)
prod_RefinedLee_2_ter = preF.terrainCorrection(prod_RefinedLee_2)
prod_Frost_55_ter = preF.terrainCorrection(prod_Frost_55)
prod_Frost_77_ter = preF.terrainCorrection(prod_Frost_77)
prod_Frost_and_Ref_55_ter = preF.terrainCorrection(prod_Frost_and_Ref_55)
prod_Frost_and_Ref_77_ter = preF.terrainCorrection(prod_Frost_and_Ref_77)
```

Muunnokset desibeleiksi (Linear to dB)

```
In [23]: prod_Lee_55_dB = preF.linearTodB(prod_Lee_55_ter)
prod_Lee_77_dB = preF.linearTodB(prod_Lee_77_ter)
prod_RefinedLee_dB = preF.linearTodB(prod_RefinedLee_ter)
prod_RefinedLee_2_dB = preF.linearTodB(prod_RefinedLee_2_ter)
prod_Frost_55_dB = preF.linearTodB(prod_Frost_55_ter)
prod_Frost_77_dB = preF.linearTodB(prod_Frost_77_ter)
prod_Frost_and_Ref_55_dB = preF.linearTodB(prod_Frost_and_Ref_55_ter)
prod_Frost_and_Ref_77_dB = preF.linearTodB(prod_Frost_and_Ref_77_ter)
```

```
In [24]: ProductIO.writeProduct(prod_Lee_55_dB, 'july_Lee_55', 'BEAM-DIMAP')
```

```
In [25]: ProductIO.writeProduct(prod_Lee_77_dB, 'july_Lee_77', 'BEAM-DIMAP')
```

```
In [26]: ProductIO.writeProduct(prod_RefinedLee_dB, 'july_RefinedLee', 'BEAM-DIMAP')
```

```
In [27]: ProductIO.writeProduct(prod_RefinedLee_2_dB, 'july_RefinedLee_2', 'BEAM-DIMAP')
```

```
In [28]: ProductIO.writeProduct(prod_Frost_55_dB, 'july_Frost_55', 'BEAM-DIMAP')
```

```
In [29]: ProductIO.writeProduct(prod_Frost_77_dB, 'july_Frost_77', 'BEAM-DIMAP')
```

```
In [30]: ProductIO.writeProduct(prod_Frost_and_Ref_55_dB, 'july_Frost_and_Ref_55', 'BEAM-DIMAP')
```

```
In [31]: ProductIO.writeProduct(prod_Frost_and_Ref_77_dB, 'july_Frost_and_Ref_77', 'BEAM-DIMAP')
```

Liite 6. ENL-arvojen laskeminen

```
In [1]: import numpy as np
import matplotlib
import matplotlib.pyplot as plt
import os
```

```
In [2]: import sys
sys.path.append('C:\\Users\\TarjaAjo\\.snap\\snap-python')
```

```
In [3]: import snappy
from snappy import Product
from snappy import ProductIO
from snappy import ProductUtils
from snappy import WKTReader
from snappy import HashMap
from snappy import GPF
import subprocess
```

```
In [4]: import preprocessingFunctions as pref
import plotFunctions
```

```
In [5]: data_0708 = 'C:/Users/TarjaAjo/Satelliittidata/satellite-images-paula/S1A_IW_GRDH_1S
product = ProductIO.readProduct(data_0708)
```

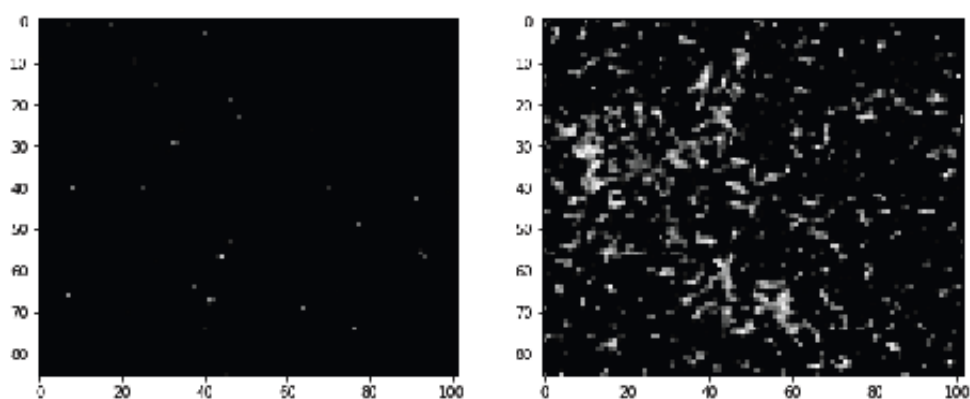
```
In [6]: # Lasketaan ENL kahdelta eri alueelta

wkt_forest = "POLYGON((27.069 65.294, 27.088 65.294, 27.088 65.288, 27.069 65.288, 2
wkt_lake = "POLYGON((27.264 65.297, 27.286 65.297, 27.286 65.290, 27.264 65.290, 27.
```

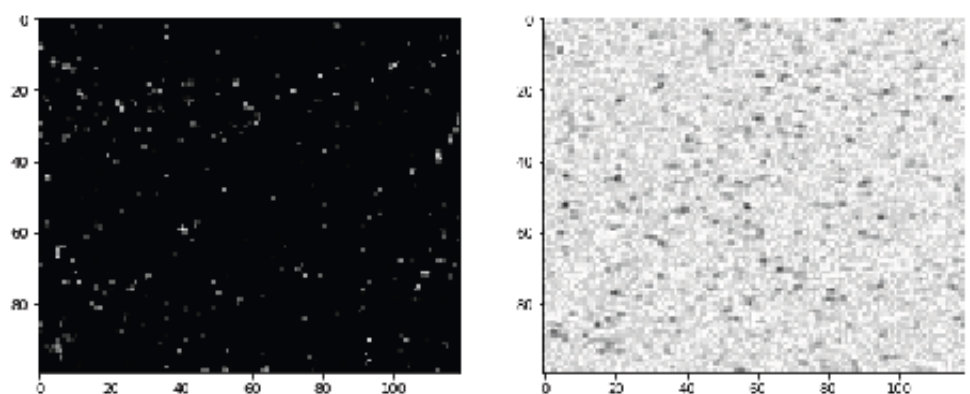
```
In [7]: product = pref.applyOrbitFile(product)
subset_forest = pref.subset(wkt_forest, product)
subset_lake = pref.subset(wkt_lake, product)
```

```
In [8]: prod_thermal_forest = pref.thermalNoiseRemoval(subset_forest)
prod_thermal_lake = pref.thermalNoiseRemoval(subset_lake)
```

```
In [9]: bands = ['Intensity_VV', 'Intensity_VH']
plotFunctions.plot_bands(prod_thermal_forest, bands, 12,12,1,2,0,10000)
```



```
In [10]: bands = ['Intensity_VV', 'Intensity_VH']
plotFunctions.plot_bands(prod_thermal_lake, bands, 12,12,1,2,0,10000)
```



```
In [11]: prod_calib_forest = preF.calibration(prod_thermal_forest)
prod_calib_lake = preF.calibration(prod_thermal_lake)
```

```
In [12]: print(preF.enl(subset_forest, 'Intensity_VH'))
print(preF.enl(subset_forest, 'Intensity_VV'))
print(preF.enl(subset_lake, 'Intensity_VH'))
print(preF.enl(subset_lake, 'Intensity_VV'))
```

```
3.8459349
4.260151
4.8220716
3.7789881
```

```
In [13]: prod_Lee55_forest = preF.speckleFilterLee(prod_calib_forest,5,5)
prod_Lee77_forest = preF.speckleFilterLee(prod_calib_forest,7,7)
prod_Frost55_forest = preF.speckleFilterFrost(prod_calib_forest,2,5,5)
prod_Frost77_forest = preF.speckleFilterFrost(prod_calib_forest,2,7,7)
prod_Ref_forest = preF.speckleFilterRefinedLee(prod_calib_forest)
prod_Ref2_forest = preF.speckleFilterRefinedLee(prod_Ref_forest)
prod_Frost55_Ref_forest = preF.speckleFilterRefinedLee(prod_Frost55_forest)
prod_Frost77_Ref_forest = preF.speckleFilterRefinedLee(prod_Frost77_forest)

print(preF.enl(prod_Lee55_forest, 'Sigma0_VH'), preF.enl(prod_Lee55_forest, 'Sigma0_VV'))
print(preF.enl(prod_Lee77_forest, 'Sigma0_VH'), preF.enl(prod_Lee77_forest, 'Sigma0_VV'))
print(preF.enl(prod_Frost55_forest, 'Sigma0_VH'), preF.enl(prod_Frost55_forest, 'Sigma0_VV'))
```

```

print(preF.enl(prod_Frost77_forest, 'Sigma0_VH'), preF.enl(prod_Frost77_forest, 'Sig
print(preF.enl(prod_Ref_forest, 'Sigma0_VH'), preF.enl(prod_Ref_forest, 'Sigma0_VV')
print(preF.enl(prod_Ref2_forest, 'Sigma0_VH'), preF.enl(prod_Ref2_forest, 'Sigma0_VV')
print(preF.enl(prod_Frost55_Ref_forest, 'Sigma0_VH'), preF.enl(prod_Frost55_Ref_fore
print(preF.enl(prod_Frost77_Ref_forest, 'Sigma0_VH'), preF.enl(prod_Frost77_Ref_fore

```

```

9.853814 15.893731
13.546052 25.268162
8.795168 14.194882
11.205772 20.384937
6.316167 9.191529
8.501823 13.608615
10.100193 17.152021
12.060008 22.856209

```

In [14]:

```

prod_Lee55_lake = preF.speckleFilterLee(prod_calib_lake,5,5)
prod_Lee77_lake = preF.speckleFilterLee(prod_calib_lake,7,7)
prod_Frost55_lake = preF.speckleFilterFrost(prod_calib_lake,2,5,5)
prod_Frost77_lake = preF.speckleFilterFrost(prod_calib_lake,2,7,7)
prod_Ref_lake = preF.speckleFilterRefinedLee(prod_calib_lake)
prod_Ref2_lake = preF.speckleFilterRefinedLee(prod_Ref_lake)
prod_Frost55_Ref_lake = preF.speckleFilterRefinedLee(prod_Frost55_lake)
prod_Frost77_Ref_lake = preF.speckleFilterRefinedLee(prod_Frost77_lake)

print(preF.enl(prod_Lee55_lake, 'Sigma0_VH'), preF.enl(prod_Lee55_lake, 'Sigma0_VV')
print(preF.enl(prod_Lee77_lake, 'Sigma0_VH'), preF.enl(prod_Lee77_lake, 'Sigma0_VV')
print(preF.enl(prod_Frost55_lake, 'Sigma0_VH'), preF.enl(prod_Frost55_lake, 'Sigma0_VV')
print(preF.enl(prod_Frost77_lake, 'Sigma0_VH'), preF.enl(prod_Frost77_lake, 'Sigma0_VV')
print(preF.enl(prod_Ref_lake, 'Sigma0_VH'), preF.enl(prod_Ref_lake, 'Sigma0_VV'))
print(preF.enl(prod_Ref2_lake, 'Sigma0_VH'), preF.enl(prod_Ref2_lake, 'Sigma0_VV'))
print(preF.enl(prod_Frost55_Ref_lake, 'Sigma0_VH'), preF.enl(prod_Frost55_Ref_lake, 'Sigma0_VV'))
print(preF.enl(prod_Frost77_Ref_lake, 'Sigma0_VH'), preF.enl(prod_Frost77_Ref_lake, 'Sigma0_VV'))

```

```

22.070158 11.097392
38.41742 15.4725
15.387378 10.10727
20.61728 13.140149
12.087988 7.1115828
22.03894 9.939627
22.493176 11.690965
28.182772 14.176989

```

Liite 7. Esikäsittelyn funktiot

```

1 import numpy as np
2 import os
3
4 import sys
5 sys.path.append('C:\\Users\\TarjaAjo\\.snap\\snap-python')
6
7 import snappy
8 from snappy import Product
9 from snappy import ProductIO
10 from snappy import ProductUtils
11 from snappy import WKTRReader
12 from snappy import HashMap
13 from snappy import GPF
14 import subprocess
15
16
17 def printInfo(product):
18     width = product.getSceneRasterWidth()
19     height = product.getSceneRasterHeight()
20     name = product.getName()
21     band_names = product.getBandNames()
22
23     print("Name: {}".format(name))
24     print("width: {} px".format(width))
25     print("Height: {} px".format(height))
26     print("Band names: {}".format(", ".join(band_names)))
27
28
29 def applyOrbitFile(product):
30     parameters = HashMap()
31     GPF.getDefaultInstance().getOperatorSpiRegistry().loadOperatorSpis()
32     parameters.put('orbitType', 'Sentinel Precise (Auto Download)')
33     parameters.put('polyDegree', '3')
34     parameters.put('continueOnFail', 'false')
35     prod_orb = GPF.createProduct('Apply-Orbit-File', parameters, product)
36
37     return prod_orb
38
39
40 def subset(wkt, product):
41     SubsetOp = snappy.jpype.get_type('org.esa.snap.core.gpf.common.SubsetOp')
42     geometry = WKTRReader().read(wkt)
43     parameters = HashMap()
44     parameters.put('copyMetadata', True)
45     parameters.put('geoRegion', geometry)
46     prod_sub = snappy.GPF.createProduct('Subset', parameters, product)
47
48     return prod_sub
49
50
51 def thermalNoiseRemoval(product):
52     parameters = HashMap()
53     parameters.put('removeThermalNoise', True)

```

```
50
51 def thermalNoiseRemoval(product):
52     parameters = HashMap()
53     parameters.put('removeThermalNoise', True)
54     parameters.put('selectedPolarisations', 'VW,VH')
55     prod_thermal = GPF.createProduct('ThermalNoiseRemoval', parameters, product)
56
57     return prod_thermal
58
59
60 def calibration(product):
61     parameters = HashMap()
62     parameters.put('outputSigmaBand', True)
63     parameters.put('sourceBands', 'Intensity_VW,Intensity_VH')
64     parameters.put('selectedPolarisations', 'VW,VH')
65     prod_calib = GPF.createProduct('Calibration', parameters, product)
66
67     return prod_calib
68
69
70 # Filttereihin kovakoodattu käsiteltävät aallonpituusvälit
71
72 def speckleFilterLee(product, x, y):
73     parameters = HashMap()
74     parameters.put('filter', 'Lee')
75     parameters.put('filterSizeX', x)
76     parameters.put('filterSizeY', y)
77     parameters.put('sourceBands', 'Sigma0_VW,Sigma0_VH')
78     prod = GPF.createProduct('Speckle-Filter', parameters, product)
79
80     return prod
81
82
83 def speckleFilterRefinedLee(product):
84     parameters = HashMap()
85     parameters.put('filter', 'Refined Lee')
86     parameters.put('sourceBands', 'Sigma0_VW,Sigma0_VH')
87     prod = GPF.createProduct('Speckle-Filter', parameters, product)
88
89     return prod
90
91
92 def speckleFilterFrost(product, d, x, y):
93     parameters = HashMap()
94     parameters.put('filter', 'Frost')
95     parameters.put('dampingFactor', d)
96     parameters.put('filterSizeX', x)
97     parameters.put('filterSizeY', y)
98     parameters.put('sourceBands', 'Sigma0_VW,Sigma0_VH')
99     prod = GPF.createProduct('Speckle-Filter', parameters, product)
100
101     return prod
102
```

```

121
122 def speckleFilterFrost(product, d, x, y):
123     parameters = HashMap()
124     parameters.put('filter', 'Frost')
125     parameters.put('dampingFactor', d)
126     parameters.put('filterSizeX', x)
127     parameters.put('filterSizeY', y)
128     parameters.put('sourceBands', 'Sigma0_VV,Sigma0_VH')
129     prod = GPF.createProduct('Speckle-Filter', parameters, product)
130
131     return prod
132
133
134 def terrainCorrection(product):
135     parameters = HashMap()
136     parameters.put('demName', 'GETASSE30')
137     parameters.put('pixelSpacingInMeter', 10.0)
138     parameters.put('sourceBands', 'Sigma0_VV,Sigma0_VH')
139     prod = GPF.createProduct("Terrain-Correction", parameters, product)
140
141     return prod
142
143
144 def linearToDb(product):
145     parameters = HashMap()
146     parameters.put('sourceBands', 'Sigma0_VV,Sigma0_VH')
147     prod = GPF.createProduct("LinearToFromdB", parameters, product)
148
149     return prod
150
151
152 def enl(product, band):
153     band = product.getBand(band)
154     w = band.getRasterWidth()
155     h = band.getRasterHeight()
156     band_data = np.zeros(w * h, np.float32)
157     data = band.readPixels(0, 0, w, h, band_data)
158     mean = np.mean(data)
159     std = np.std(data)
160     ENL = (mean * mean)/(std * std)
161
162     return ENL
163
164 # Käytettävissä olevat toiminnot ja niiden parametrit:
165
166 # print(subprocess.Popen(['gpt', '-h'], stdout=subprocess.PIPE,
167     universal_newlines=True).communicate()[0])
168
169 # print(subprocess.Popen(['gpt', '-h', 'Speckle-Filter'], stdout=subprocess.PIPE,
170     universal_newlines=True).communicate()[0])
171
172
173

```