

Lappeenrannan-Lahden teknillinen yliopisto LUT  
School of Engineering Science  
Tietotekniikan koulutusohjelma

DIPLOMITYÖ

**Jaakko Ketola**

# **METSIKKÖKUVIDEN TUOTTAMINEN KAUKO- KARTOITUSAINESTOA KÄYTTÄEN**

Työn tarkastajat: Professori Jari Porras  
Tutkijaopettaja Ari Happonen  
Työn ohjaaja: Professori Jari Porras

# TIIVISTELMÄ

Lappeenrannan-Lahden teknillinen yliopisto LUT

School of Engineering Science

Tietotekniikan koulutusohjelma

Jaakko Ketola

## **Metsikkökuvioiden tuottaminen kaukokartoitusaineistoa käyttäen**

Diplomityö 2020

93 sivua, 50 kuvaa ja 9 taulukkoa

Työn tarkastajat: Professori Jari Porras

Tutkijaopettaja Ari Happonen

Hakusanat: Kaukokartoitus, LiDAR, ilmakuva, metsikkökuvio, kuvankäsittely, bilateraalinen suodatus, Mean Shift -suodatus, Gradient Vector Flow, segmentointi, Watershed-muunnos, dynamiikkasuodatus, Active Contour

Metsikkökuvioita hyödynnetään metsän toimenpidesuunnittelussa ja puuston arvioinnissa. Perinteisesti näitä kuvioita on rajattu käsin ilmakuvista. Kuvioita voidaan tuottaa myös automaattisesti erilaisia kaukokartoitusaineistoja käyttäen. Tässä keskitytään väärävärililmakuvien ja laserkeilausaineiston ominaisuuksiin, käsittelyyn ja muuntamiseen muotoon, jossa niitä voidaan hyödyntää metsikkökuvioiden automaattiseen tuottamiseen. Työssä esitetään kehitetty menetelmä aineiston esikäsittelyyn, jossa eri aineistot yhdistetään samassa georeferenssissä olevaksi kuvaksi, josta kohinan suodatuksella ja bilateraalista tai Mean Shift -suodatusta käyttäen tuotetaan segmentointia varten aineisto, josta metsikkökuviot tuotetaan dynamiikkasuodatetulla Watershed-muunnoksella. Lisäksi esitetään menetelmä alueiden yhdistämiseen samankaltaisuuden perusteella. Lopuksi kuvataan menetelmä tuotettujen kuvioiden muuntamiseksi vektorimuotoon, jolle suoritetaan reunaviivojen tarkennus topologian säilyttävällä Active Contour -menetelmällä. Kuvioinnin laadun arviointi ilman maastokäyntejä on haastavaa tuotettujen rajausten oikeellisuuden varmistamiseksi.

# ABSTRACT

Lappeenranta-Lahti University of Technology LUT

School of Engineering Science

Degree Programme in Software Engineering

Jaakko Ketola

## **Forest Stand Delineation Using Remote Sensing Data**

Master's Thesis 2020

93 pages, 50 figures and 9 tables

Examiners:        Professor Jari Porras  
                     Associate Professor Ari Happonen

Keywords:        Remote sensing, LiDAR, aerial image, forest stand, image processing, bi-lateral filtering, Mean Shift filtering, Gradient Vector Flow, segmentation, Watershed-transform, dynamics filtering, Active Contour

Forest stands are used in forest operational planning and forest inventory. Traditionally forest stand delineation has been done manually using aerial images. Stand delineation can also be done automatically using various remote sensing data. Here we describe false color aerial images and aerial laser scanning data and use and transform them into a form that can be used in an automatic stand delineation process. In this work, we present methods that were developed for preprocessing data using noise reduction and bilateral or Mean Shift filtering methods for the segmentation that is using dynamics filtered watershed transform to perform the forest stand delineation. We also present a method to merge the stands by their similarities. In the end, we describe a method to vectorize the segmentation result and a method to correct the vectorized form of the stand delineation using the topology-preserving Active Contour method. The quality assessment of the produced stand delineation without field visits is difficult.

# ALKUSANAT

Kiitokset Arbonaut Oy:lle työn aiheesta ja mahtavista työkaluista paikkatietoaineiston käsittelyyn, joiden kehittämiseksi työ toteutettiin.

Kiitos Nergiz tuesta ja kannustuksesta.

Kiitos Eino parhaista kysymyksistä itse aiheeseen liittyen.

Kiitos Ida seurasta työn kirjoitusvaiheen aikana.

# SISÄLLYSLUETTELO

<b>1</b>	<b>JOHDANTO</b>	<b>5</b>
1.1	Tavoitteet ja rajaukset . . . . .	5
1.2	Työn rakenne . . . . .	6
<b>2</b>	<b>KAUKOKARTOITUS</b>	<b>7</b>
2.1	Laserkeilaus kaukokartoituksessa . . . . .	7
2.1.1	Etäisyyden mittaus . . . . .	8
2.1.2	Laserkeilaus . . . . .	9
2.1.3	Ilmalaserkeilauksen mittauslaitteisto . . . . .	10
2.1.4	Mittausaineisto . . . . .	12
2.1.5	Digitaalinen maastomalli ja latvusmalli . . . . .	13
2.1.6	Laserkeilausaineiston korkeuksien normalisointi . . . . .	16
2.2	Ilmakuvaus kaukokartoituksessa . . . . .	16
2.2.1	Ilmakuvauksen käyttökohteet . . . . .	17
2.2.2	Kvantamistekniikka . . . . .	18
2.2.3	Kuvien erotuskyky . . . . .	20
2.2.4	Orto-oikaisu . . . . .	21
2.2.5	Radiometrinen korjaus . . . . .	22
2.2.6	Kasvillisuusindeksit . . . . .	23
<b>3</b>	<b>KAUKOKARTOITUS- JA PAIKKATIETOAINIESTOT</b>	<b>24</b>
3.1	Koordinaattijärjestelmä Suomessa . . . . .	24
3.2	Laserkeilausaineisto . . . . .	25
3.3	Georeferoitu rasteriaineisto . . . . .	26
3.4	Vektoriaineisto . . . . .	27
<b>4</b>	<b>DIGITAALINEN KUVANKÄSITTELY</b>	<b>29</b>
4.1	Laserkeilausaineiston rasterointi . . . . .	30
4.2	Pistemäiset operaatiot . . . . .	31
4.3	Spatiaalinen suodatus . . . . .	32
4.3.1	Gaussin silotus . . . . .	33

4.3.2	Mediaanisuodatus . . . . .	34
4.3.3	Bilateraalinen suodatus . . . . .	35
4.3.4	Mean Shift -suodatus . . . . .	37
4.3.5	Sobel suodatus . . . . .	39
4.3.6	Laplace ja kuvan terävöitys . . . . .	39
4.3.7	Gradient Vector Flow . . . . .	42
4.4	Segmentointi . . . . .	43
4.4.1	Segmentoinnin määritelmä . . . . .	45
4.4.2	Watershed-muunnos . . . . .	45
4.4.3	Dynamiikkasuodatus ja Watershed . . . . .	50
4.4.4	Active Contour -menetelmä . . . . .	51
4.5	Vektoriesitys ja kuvan vektorisointi . . . . .	54
<b>5</b>	<b>AINEISTOT JA METSIKKÖKUVIOIDEN TUOTTAMINEN</b>	<b>58</b>
5.1	Maanmittauslaitoksen avoimet aineistot . . . . .	58
5.2	Sovellus- ja kehitysympäristö . . . . .	61
5.3	Kuvankäsittelymenetelmien toteuttaminen . . . . .	62
5.4	Alueiden yhdistäminen graafista . . . . .	64
5.5	Menetelmä metsikkökuvioiden tuottamiseen . . . . .	66
5.5.1	Aineiston esikäsittely . . . . .	67
5.5.2	Segmentointi . . . . .	70
5.5.3	Jälkikäsittely . . . . .	74
5.6	Menetelmä metsikkökuvioiden rajojen tarkentamiseen . . . . .	76
5.6.1	Aineiston esikäsittely . . . . .	76
5.6.2	Kuviorajojen tarkentaminen . . . . .	77
<b>6</b>	<b>JOHTOPÄÄTÖKSET</b>	<b>82</b>
	<b>LÄHTEET</b>	<b>85</b>

# LYHENNELUETTOLO

ALS	Aerial Laser Scanning
ASPRS	American Society for Photogrammetry and Remote Sensing
BMS	Blurring Mean Shift
CHM	Canopy Height Model
CIR	Color-Infrared
CW	Continuous Wave
DEM	Digital Elevation Model
DGPS	Differential Global Positioning System
DoG	Difference of Gaussian
DSM	Digital Surface Model
DTM	Digital Terrain Model
ETRS89	European Terrestrial Reference System 1989
EVI	Enhanced Vegetation Index
ESRI	Environmental Systems Research Institute
FIR	Finite Impulse Response
GDAL	The Geospatial Data Abstraction Library
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
GVF	Gradient Vector Flow
IDW	Inverse Distance Weighting
IIR	Infinite Impulse Response

IMU	Inertial Measurement Units
KDE	Kernel Density Estimation
LiDAR	Light Detection and Ranging
LoG	Laplacian of Gaussian
LWG	LAS Working Group
MS	Mean Shift
MSF	Minimal Spanning Forest
MST	Minimal Spanning Tree
NDVI	Normalized Difference Vegetation Index
NIR	Near-Infrared
OGC	Open Geospatial Consortium
OGR	OGR Simple Features Library
PDAL	Point Data Abstraction Library
PRF	Pulse Repetition Frequency
SAR	Synthetic Aperture Radar
TIN	Triangulated Irregular Network
TLS	Terrestrial Laser Scanning
TOF	Time of Flight
VMI	Valtakunnan metsien inventointi
WGS84	World Geodetic System 1984
WKT	Well Known Text



# 1 JOHDANTO

Metsikkökuvioiden muodostus on ollut perinteisesti käsityötä käyttäen ilmakuvia kuvioiden rajaamiseen. Metsikkökuvioita tehdään metsänhoidon tai hakkuiden toimenpidesuunnittelua varten. Tavoitteena kuvioiden rajaamiseen on muodostaa lajistoltaan ja kehitysluokaltaan homogeenisiä alueita, joiden perusteella voidaan suunnitella tarvittavia toimenpiteitä tai arvioida puustoa. (Leppänen et al., 2008)

Menetelmiä vähentää metsikkökuvioinnin käsityötä on kehitetty aikaisemminkin eri hankkeissa automatisoimalla kuviointiprosessia käyttäen vääräväri-ilmakuvia ja laserkeilausaineistoa. Puuston kehitysluokan huomioiminen kuvioinnissa on ainakin pelkkien vääräväri-ilmakuvien perusteella ollut haastavaa, mutta laserkeilausaineistolla on puuston korkeus ja siten puuston kehitysluokkaa selittävä ominaisuus saatu huomioitua kuvioinnissa. (Pekkari-  
nen, 2002; Leppänen et al., 2008)

Metsikkökuvioita käytetään toimenpidesuunnittelun lisäksi puuston inventoinneissa, vaikka inventointia tuotetaan laserkeilausaineiston ja ilmakuva-aineiston perusteella laskentaruudukolle, niin näiden metsävaratietojen yhdistäminen ja yleistäminen suoritetaan jollekin laajemmalle alueelle esimerkiksi metsikkökuvioille tai toimenpidekuviolle. Metsien kuviointi toimenpidesuunnittelun tarpeisiin varten on ollut erityisen haastavaa. (Leppänen et al., 2008)

## 1.1 Tavoitteet ja rajaukset

Tässä työssä on tavoiteltu menetelmän kehittämistä metsikkökuvioiden tuottamiseen kaukokartoitusaineistosta. Tavoitteeseen pyrittiin tutkimalla kaukokartoitusaineistojen käsittelyä, kaukokartoitusaineistojen muuntamista kuvankäsittelymenetelmien käytettäväksi, ja aineiston käsittelyyn ja metsikkökuvioiden tuottamiseen tarvittavia ja soveltuvia kuvankäsittelymenetelmiä. Tavoitteen saavuttamisella on pyritty kehittämään toimiva operatiivinen sovellus ja menetelmäprosessi, jolla voidaan kaukokartoitusaineistoa käyttäen tuottaa käyttökelpoisia metsikkökuvioita.

Tässä työssä käytettävät kaukokartoitusaineistot rajataan vääräväri-ilmakuviin ja laserkeilausaineistoon. Satelliittikuvien hyödyntämistä ei tässä työssä käsitellä eikä muidenkään mahdollisesti lisäinformaatiota tuottavien aineistojen käyttämistä. Aineistojen käsittelymenetelmiä ja kuvankäsittelymenetelmiä ei ole rajoitettu, mutta työssä esitetään ne menetelmät, jotka ovat valittu työssä toteutetun sovelluksen tarpeiden mukaisesti.

## **1.2 Työn rakenne**

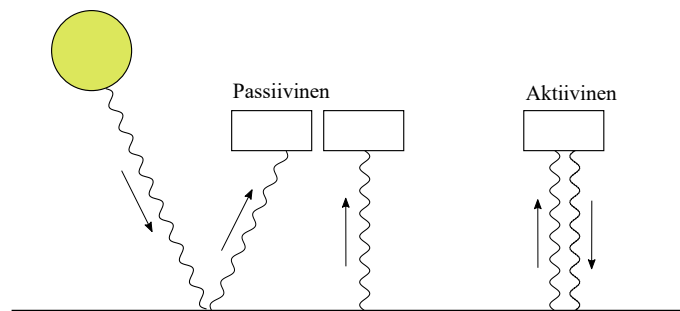
Työn rakenteen on tarkoitus tukea työssä esiteltyjen menetelmien käyttämien aineistojen ominaisuuksia ja niihin liittyviä haasteita sekä sitä prosessia, jolla esitellyt menetelmät hyödyntävät käytettyä kaukokartoitusaineistoa.

Työssä esitellään ensin kaukokartoitusmenetelmät ja näihin liittyvät menetelmät aineistojen keräämiseen, aineistoihin liittyvät ominaisuudet ja mahdolliset virhelähteet. Lisäksi käydään läpi menetelmiä, joilla aineistoja käsitellään, jotta ne olisivat käyttökelpoisia kuvankäsittelymenetelmille ja siten myös menetelmälle metsikkökuvioiden tuottamiseen. Esi- teltäviä kaukokartoitusmenetelmiä ovat ilmalaserkeilaus ja ilmakuvaus. Seuraavassa kappaleessa esitellään paikkatietoaineistoihin yhdistettyä maantieteellistä koordinaatistoa, ja kaukokartoitus- ja paikkatietoaineistoja käytettävien aineistoformaattien suhteen. Näitä kuvataan Suomessa käytetyn koordinaatiston ja työssä hyödynnettyjen avoimien paikkatieto- kirjastojen mahdollisuuksien mukaisesti.

Ennen työssä kehitettyjen menetelmien esittämistä käydään läpi omassa kappaleessaan näiden menetelmien hyödyntämät ja työssä toteutetut kuvankäsittelymenetelmät. Tässä työssä esitetään toteutetuista kuvankäsittelymenetelmistä vain työn lopullisissa menetelmissä käytetyt menetelmät ja näistä työn kannalta kriittiset toteutetut algoritmit. Lopuksi käydään läpi työssä käytetty ohjelmointi- ja sovellusympäristö, sekä kehitetyt ja toteutetut menetelmät metsikkökuvioiden tuottamiseksi muuntamalla kaukokartoitusaineisto kuvankäsittelymenetelmien käytettäväksi ja näiden menetelmät näiden aineistojen muuntamiseksi metsikkökuvioiden rajauksiksi. Kyseisessä kappaleessa pyritään kuvaamaan menetelmien käyttöä kehitettyjen menetelmäprosessien kannalta.

## 2 KAUKOKARTOITUS

Kaukokartoituksella kerätään aineistoa lintuperspektiivistä havainnoimalla maan pinnasta heijastuneen tai säteilleen yhden tai useamman sähkömagneettisen spektrin alueella olevaa sähkömagneettista säteilyä, ja tuotetaan tästä tietoa maapallon maa- ja vesipinnoista käyttämällä (Campbell et al., 2011). Kaukokartoitusjärjestelmät jaetaan kahteen ryhmään toimintaperiaatteensa mukaan. Passiivisiin ja aktiivisiin kaukokartoitusjärjestelmiin. Passiiviset kaukokartoitusjärjestelmät mittaavat olemassa olevaa säteilyä esimerkiksi maasta heijastunutta auringon valoa. Aktiivisten järjestelmien toiminta ei riipu ulkoisista säteilyn lähteistä, vaan ne lähettävät itse säteilyä, jonka heijastumista kohteesta mittaavat. Laserkeilaus on aktiivinen kaukokartoitusjärjestelmä ja ilmakuvauus on passiivinen. Kuvassa 1 on havainnollistettu toimintaperiaatteiden erot. (Mårtensson, 2011; Tempfli et al., 2009)



Kuva 1: Passiivinen kaukokartoitusjärjestelmä mittaa maasta heijastunutta ja lähtenyttä säteilyä. Aktiivinen järjestelmä sisältää itsessään säteilyn lähteen.

### 2.1 Laserkeilaus kaukokartoituksessa

Laserkeilaus on menetelmänä kehitetty jo 1960-luvulla, mutta 1990-luvun paikannusmenetelmien kehittyminen on vaikuttanut laserkeilauksen yleistymiseen, ja siitä on muodostunut 1990-luvulta lähtien kaukokartoituksen tiedonkeruun tärkein lähde (Shan et al., 2018). Ilmalaserkeilauksen eli ALS:n (Aerial Laser Scanning) lisäksi laserkeilausta suoritetaan myös maalaserkeilauksena eli TLS:nä (Terrestrial Laser Scanning), jossa laserkeilainlaitteisto on maassa (Tempfli et al., 2009). Maalaserkeilausta on aloitettu hyödyntämään sotateollisuudessa etäisyyden mittaamiseen ja seurantaan jo 1970-luvulla. Laseriin pohjautuvaa etäisyysmittausta alettiin siviilikäytössä hyödyntämään topografiassa kolmiomittauksissa

(Shan et al., 2018) ja viime vuosina lisääntyvässä määrin myös esimerkiksi metsätaloudessa (Liang et al., 2016). Ilmalaserkeilausta aloitettiin hyödyntämään korkeuden mittaamisessa kuitenkin jo 1960-luvulla (Shan et al., 2018) ja nykyään metsätaloudessa esimerkiksi valtakunnan metsien inventointi (VMI) Suomessa hyödyntää laserkeilausta (Maltamo et al., 2004).

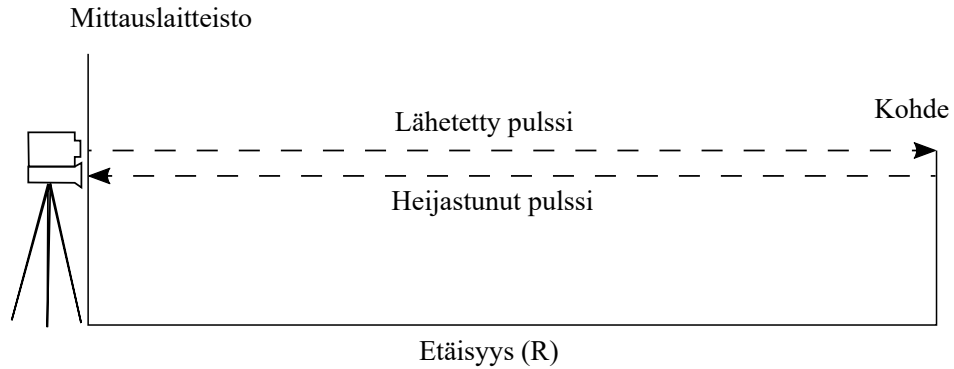
Etenkin ilmasta suoritettavan laserkeilauksen kehittymiselle on ollut tärkeää asennon estimointilaitteiden eli IMU (Inertial Measurement Unit), satelliittipaikannuslaitteiden eli GPS:n (Global Positioning System) tai yleisemmin GNSS:n (Global Navigation Satellite System), sekä näiden yhdistelmien kehittyminen. Ilmasta suoritettavan laserkeilauksen kannalta on tärkeää, että laitteiston sijainti ja asento tiedetään mahdollisimman tarkasti, jotta etäisyysmittauksella saadut arvot on mahdollista sijoittaa käytettyyn maantieteelliseen koordinaatistoon. (Shan et al., 2018)

### 2.1.1 Etäisyyden mittaus

Laserkeilauksessa käytetään etäisyyden mittaamiseen joko valon lentoaikaan eli ToF (Time of Flight) tai signaalin vaihe-eroon perustuvaa menetelmää (Shan et al., 2018). Valon lentoaikaan ja pulssimodulointiin perustuvassa menetelmässä, joka on esitetty kuvassa 2, laitteisto lähettää pulssimuotoista valoa, jolloin etäisyys saadaan laskettua mittaamalla viive paluupulssille. Tässä menetelmässä on tärkeää tunnistaa paluupulssien ajankohdat tarkasti, jotka voidaan tunnistaa paluupulssin signaalin huipuista, etureunasta, suhteellisella tai vakioidulla raja-arvolla (Vosselman et al., 2010). Etäisyys  $R$  lasketaan ToF -menetelmässä kaavalla:

$$R = \frac{ct}{2}, \quad (1)$$

jossa  $c$  on tarkkaan tunnettu valonnopeus, joten etäisyyden  $R$  mittauksen tarkkuus on suoraan verrannollinen ajan  $t$  mittaamisen tarkkuuteen (Shan et al., 2018).

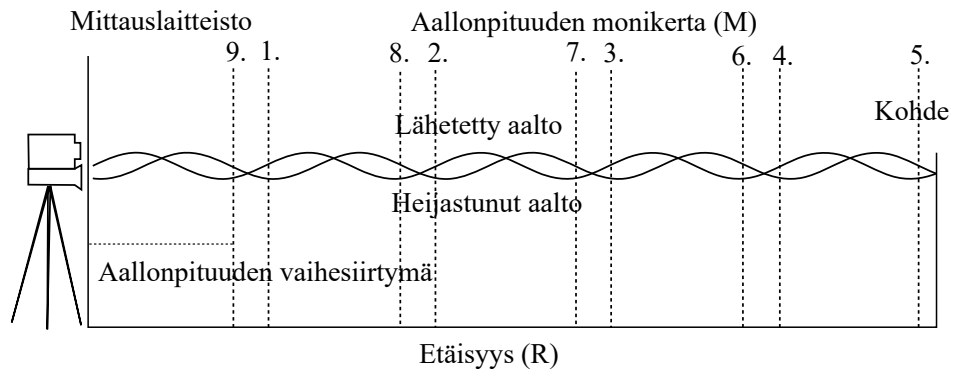


Kuva 2: Etäisyysmittaus pulssimoduloidulla laserkeilauslaitteistolla.

Signaalin vaihe-eroon perustuvassa kuvassa 3 esitetystä jatkuvan aallon eli CW (Continuous Wave) sinimodulointiin perustuvassa menetelmässä laitteisto lähettää pulssimuotoisen valon sijaan sinimuotoista aaltoa, jolloin etäisyys  $R$  voidaan laskea vaihesiirtymästä hyödyntämällä lähetysoimakkuuden siniaaltomuotoista vaihtelua seuraavasti:

$$R = \frac{M\lambda + \Delta\lambda}{2}, \quad (2)$$

jossa  $M$  on aallonpituuden  $\lambda$  lähetysoimakkuuden vaihtelulla johdettu kokonaislukumonikerta ja  $\Delta$  on vaihesiirtymästä laskettu aallonpituuden murto-osakerroin (Shan et al., 2018).



Kuva 3: Etäisyysmittaus sinimoduloidulla jatkuvan aallon sinimoduloidulla laserkeilauslaitteistolla.

### 2.1.2 Laserkeilaus

Laserprofilointi tarkoittaa ilmalaserkeilauksen osalta kaksiulotteista etäisyysmittausta eli etäisyyttä mitataan kiinteään suuntaan. Mittauspisteiden maantieteelliset sijainnit johdetaan laser-profiloinnissa tuntemalla laitteiston maantieteellinen sijainti ja mittauslaitteiston

lähettämän valon suunta. Laserkeilauksella tarkoitetaan vastaavaa mittausta kolmiulotteisesti eli laitteisto lähettää valoa eri suuntiin. Laitteistossa on tällöin opto-mekaaninen osa. (Shan et al., 2018; Wehr et al., 1999)

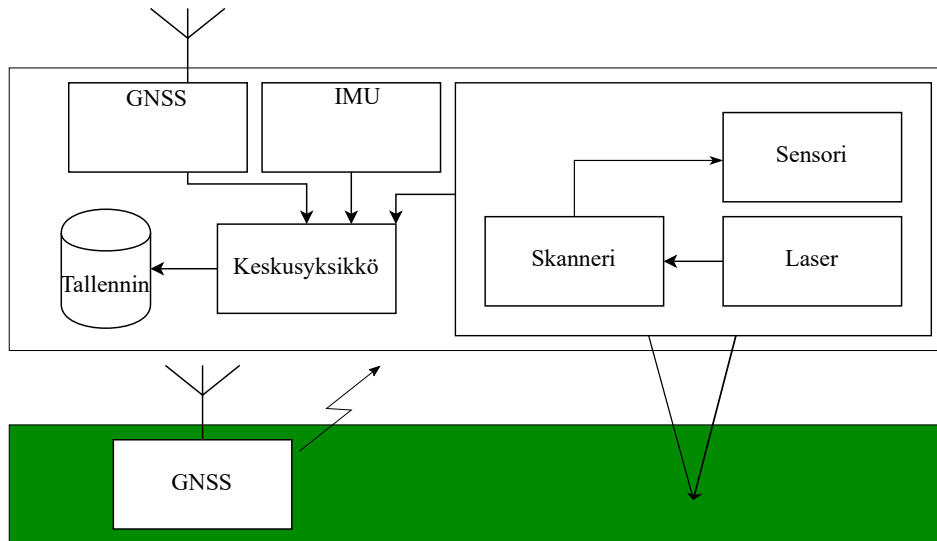
Mittausta varten lentoreitit suunnitellaan tarkasti siten, että lentolinjat menevät osittain liittämättä, ovat mahdollisimman suorina, vakiokorkeudella ja -nopeudella (Shan et al., 2018). Ilmalaserkeilauksessa mittauksen kannalta tärkeitä muuttujia, joilla on vaikutusta mittaus-tarkkuuteen ja mitatun aineiston pistetiheyteen, ovat (Vosselman et al., 2010):

- Lentokorkeus
- Lentonopeus
- Keilauskulma
- Pulssin toistotaajuus / Pulse Repetition Frequency (PRF)
- Laservalon aallonpituus

### **2.1.3 Ilmalaserkeilauksen mittauslaitteisto**

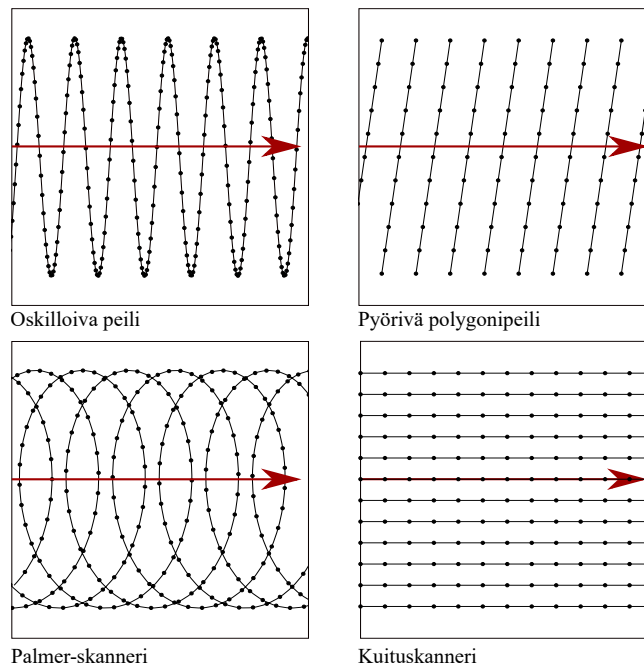
Ilmalaserkeilauksen mittauslaitteisto koostuu keilainyksiköstä, asennon estimointiyksiköstä IMU:sta, paikannuslaitteesta GNSS:stä, keskusyksiköstä ja aineiston tallentimesta. Keilainyksikkö koostuu laser lähteestä, sensorista, ja keilaimesta. Kuvassa 4 on esitetty laitteiston eri osien riippuvuuksia. Näiden lisäksi paikannuslaite hyödyntää maassa olevaa paikannuksen viiteasemasta. (Vosselman et al., 2010)

Mittauslaitteiston keilaimen opto-mekaanisina osina käytetään oskilloivaa peiliä, pyörivää polygonipeiliä, epäkeskosti pyörivää peiliä tai valokuituja. Oskilloivassa peilissä mittauspisteitä tulee suoralta linjalta, joten mittauspisteet muodostavat ns. zigzag-kuvion, kun pyörivällä polygonipeilillä mittauspisteitä tulee samansuuntaisilta suorilta linjoilta. Palmerkeilaimessa peiliä pyöritetään muodostaen ellipsin muotoisia kuvioita mittauspisteille maaston tasossa, jonka etuna on se, että mitattavaa kohdetta saadaan mitattua hieman eri kulumista. Kuitukeilaimen tekniikka on hyvin samankaltainen Palmer-keilaimen kanssa mutta



Kuva 4: Ilmalaserkeilauksessa käytettävän laitteiston keskeiset osat.

optiikkana on valokuitunippu, jolloin lähettävä ja vastaanottava optiikka ovat identtisiä ja pulssin lähetys ja vastaanotto pysyvät synkronoituina ja pulsseja voidaan lähettää suuremalla tiheydellä mittausrésoluution ja suurimman mittausetäisyyden kärsimättä. Kuitukeilainta voidaan käyttää myös jatkuvan aallon mittauseräilymenetelmällä. Kuvassa 5 on esitetty eri tyyppisille keilaimille tyypilliset mittauspisteiden muodostamat kuviot maanpinnalla. (Wehr et al., 1999; Vosselman et al., 2010)

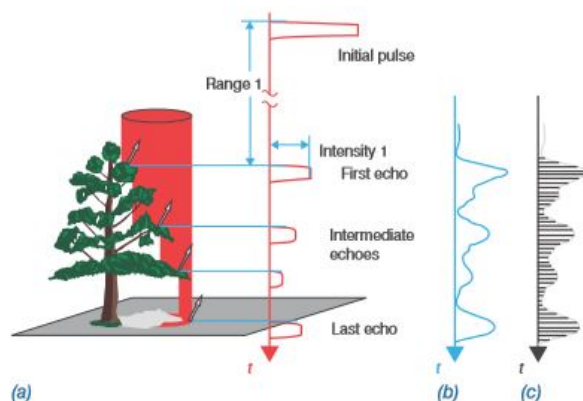


Kuva 5: Eri laisten keilaintyyppien mittauspisteiden sijoittuminen. Punainen nuoli kuvaa lentolinjaa.

Vosselman et al., 2010, mukaan järjestelmän haasteita ovat sijainnin ja asentojen tarkka mittaaminen sekä mittausten keskinäinen synkronointi, koska eri osat tuottavat tietoa eri taajuuksilla, mutta tässä voidaan hyödyntää Kalman-suotimia estimointiyksikön ja paikannuslaitteen tietojen suodatukseen. Sijainnin mittaustarkkuutta voidaan korjata käyttäen paikannuksen viiteasemaa, jonka sijainti tunnetaan täsmällisesti, tästä menetelmästä käytetään nimitystä DGPS eli Differential Global Positioning System ja tällä menetelmällä saavutetaan paikannuksessa alle 10 cm tarkkuuteen, kun IMU:n mittaukset ja viiteaseman mittaukset yhdistetään Kalman-suotimella. Viiteaseman etäisyys lentokoneeseen tulisi olla alle 30 kilometriä (Vosselman et al., 2010). Tarkkuutta voidaan parantaa keräämällä useista eri satelliittipaikannusjärjestelmistä tietoa yhtäaikaaisesti ja yhdistämällä tämä monijärjestelmädata vastaavasti viiteaseman tietoon (Lehtinen et al., 2008; Jahkola et al., 2017).

#### 2.1.4 Mittausaineisto

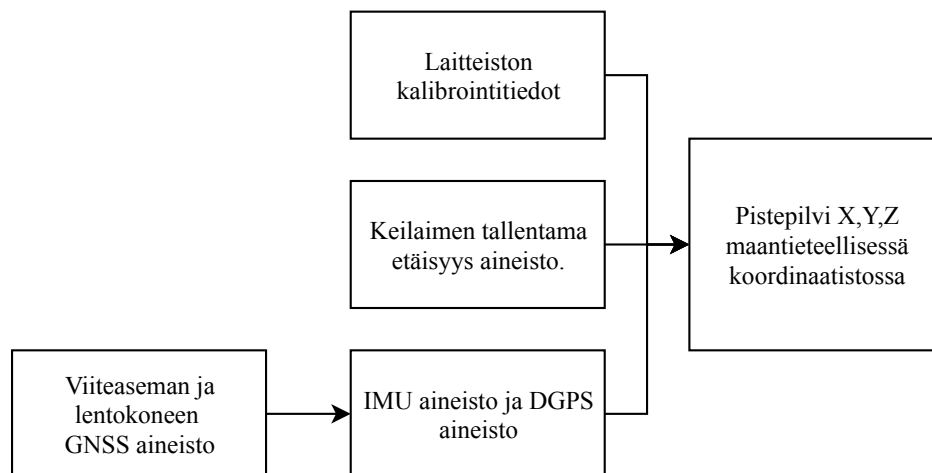
Laserkeilauslaitteisto tallentaa mittauspisteitä käytetyssä maantieteellisessä koordinaatistossa. Kullekin mittauspisteelle tallennetaan tässä vaiheessa ajankohta ja suunta, ja diskreetin paluupulssin aineistossa tallennetaan sijainnin lisäksi kullekin mittaukselle paluupulssin järjestysnumero ja intensiteetti. Täyden jakauman aineistossa ajankohdan ja suunnan lisäksi tallennetaan kullekin mittaukselle diskretoitu intensiteettijakauma (Shan et al., 2018). Kuvassa 6 on kuvattu diskreetin paluupulssin aineistoa ja täyden jakauman aineistoa paluusignaalin kannalta. Nykyaikaiset keilaimet voivat suorittaa jopa 300000 mittausta sekunnissa, jolloin mittausaineistoa voi kertyä 20 gigatavua tunnissa (Vosselman et al., 2010).



Kuva 6: Mittauksen paluusignaalista tallennetaan, joko havaitut paluupulssit tai diskretoitu intensiteettijakauma (Vosselman et al., 2010).



Prosessi pistepilven tuottamiseksi mitatuista aineistoista on esitetty kuvassa 7, jossa satelliittipaikannusaineisto korjataan viiteaseman sijaintitietojen ja -mittausten avulla. Tämä korjattu aineisto yhdistetään lentokoneen asennon estimointiaineistoon ja laserkeilaimella mitattuihin etäisyyksiin ja mittaussuuntiin, jolloin saadaan tuotettua maantieteellisessä koordinaatissa oleva pistejoukko. (Vosselman et al., 2010)



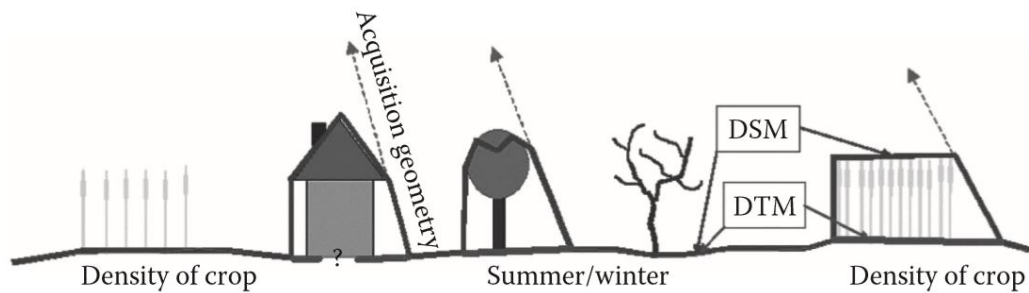
Kuva 7: Laserkeilaus aineistonkäsittelyprosessi.

Uutena ominaisuutena laserkeilauslaitteistoihin on tullut myös monispektri-keilaimia, jotka lähettävät ja vastaanottavat laservaloa useammalla aallonpituudella. Useammasta mitatusta aallonpituudesta on hyötyä mittauspisteiden luokittelussa, koska materiaalit voidaan usein erottaa toisistaan niiden eri aallonpituuksien heijastavuuden perusteella. (B. Chen et al., 2017)

### 2.1.5 Digitaalinen maastomalli ja latvusmalli

Digitaalisen maastomalli eli DTM:n (Digital Terrain Model), josta käytetään myös lyhennettä DEM (Digital Elevation Model), voidaan kuvata seuraavasti: DTM on digitaalisesti tallennettu jatkuva funktio, joka kuvaa kaksiulotteisen tasosijainnin maaston korkeuteen  $z = f(x, y)$ . Maaston korkeus määritetään ilman ja maan rajasta, joka on tyypillisesti sama kuin veden valumapinta (Shan et al., 2018). Perinteisesti maastomallia on tuotettu manuaalisesti ilmakuvista fotogrammetrialla, mutta laserkeilaus ja suuren laskennallisen läpimitan tutka eli SAR (Synthetic Aperture Radar) ovat tulleet menetelminä yleisesti saataville ja tuottavat tarkkaa ja tiheää mittausta (Vosselman et al., 2010).

Toisin kuin DTM digitaalinen pintamalli eli DSM (Digital Surface Model) sisältää myös maanpäälliset kohteet esimerkiksi kasvillisuuden ja rakennelmat. Nämä korkeusmallit ovat käytännössä 2.5-ulotteisia eli yhtä maapistettä  $(x, y)$  vastaa tasan yksi korkeus  $z$ . Joka on huomioitava niin mallin luomisessa kuin käytössä, koska esimerkiksi rakennelmat voivat joissain käyttötapauksissa olla osana maastomallia, kun taas toisissa nämä eivät kuuluisi maastomalliin. Rakennelmat voivat siten aiheuttaa reikiä malliin. Kuvassa 8 on esitetty maastomalli, pintamalli ja rakennelmien aiheuttamat ongelmat. (Shan et al., 2018)

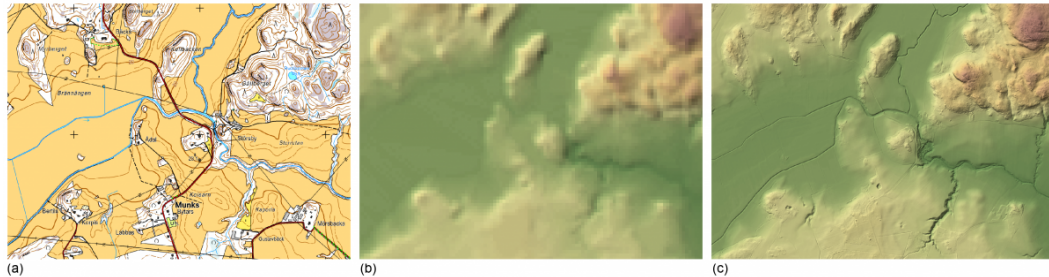


Kuva 8: Digitaalinen maastomalli DTM ja pintamalli DSM. Rakennelmien alta ei voida tuottaa maastomallia ja pintamallissa voi olla virheitä, jos kohde on liian tiheää tai harvaa. (Shan et al., 2018)

Kovilla pinnoilla maastomalli on usein selkeästi määriteltävissä, mutta pinnoilla, joilla on kasvillisuutta voi maastomallin määrittäminen olla epätasemmalla tasolla, jota voi täsmentää käyttämällä tiheämpää laserkeilausaineistoa eli matalampaa lentokorkeutta tai korkeampaa pulssin toistotaajuutta. Kasvillisuuden vuoksi laserkeilausaineistosta johdetussa maastomallissa voi usein olla noin  $\pm 10$  cm virhe. (Shan et al., 2018)

Maastomalli tallennetaan yleensä rasterikuvana eli malli on tällöin suorakulmainen ruudukko tai vektorimuodossa, jolloin malli on kolmioitu epäsäännöllinen verkko eli TIN (Triangulated Irregular Network), joka voi olla kolmioitu esimerkiksi Delaunay-kolmioinnilla. Maastomallia voidaan tuottaa laserkeilausaineistosta usealla eri menetelmällä. Eräitä menetelmiä ovat pistepilven suodatus, morfologinen suodatus, tai asteittainen tihentäminen. Suodatuksen jälkeen rasterimuotoista maastomallia tuottaessa suoritetaan interpolointi esimerkiksi painotamalla maastopisteet käänteisesti etäisyyden mukaan eli IDW (Inverse Distance Weighting), tai käyttäen Kriging- tai splini-interpolointia. (Vosselman et al., 2010)

Suomessa Maanmittauslaitos tuottaa kuvan 9 mukaista rasterimuotoista maastomallia 2 metrin pikselikoolla, joka perustuu ilmalaserkeilaukseen sekä karkeampia maastomalleja 10, 25 ja 200 metrin pikseliko'oilla. (Oksanen, ei julkaisupäivää)



Kuva 9: Topografinen kartta (a), maastomallit 25 m (b) ja 2 m (c) pikseliko'oilla (Oksanen, ei julkaisupäivää).

### **Pistepilven maastopisteiden luokittelumenetelmiä**

Pistepilven suodatuksessa luokitellaan pisteet joko maastoon kuuluviksi tai kuulumattomiksi jollain paikallisella kriteerillä. Yksinkertaisimmillaan kriteeri voi olla pienin arvo jonkin alueen sisällä (Vosselman et al., 2010). Yhtenä keinona luokitteluun on vinouden balansointi-menetelmä, jossa alueellisesti pyritään saamaan tilastollinen käsite jakauman vinous mahdollisimman lähelle arvoa 0 pudottamalla jakauman otoksesta korkeimpia mittauspisteitä pois (Bartels et al., 2006).

Morfologinen suodatus voidaan tehdä käyttäen eroosiota ja dilaatiota. Näitä käyttäen avaaminen eli eroosion dilaatio tuottaa luokittelun DSM:ään ja sulkeminen eli dilaation eroosio tuottaa luokittelun DTM:ään. Strukturoivana elementin koon valinta riippuu pistepilven tiheydestä. Eroosio saa pienimmän korkeuden ja dilaatio saa suurimman korkeuden strukturoivan elementin sisällä. Tällä menetelmällä on saatu parempia tuloksia, kun eroosiossa ja dilaatiossa on sallittu strukturoivan elementin etäisyydestä keskipisteeseen riippuva suurin sallittu erotus korkeudessa. (Vosselman et al., 2010)

Asteittainen tihentäminen perustuu TIN malliin, jossa valitaan ensin alueellisesti matalimmat pisteet alueen ollessa riittävän iso ja aletaan lisäämään malliin pisteitä kunkin kolmion sisälle, mikäli kolmion sivut ovat liian pitkiä eli haluttua tarkkuutta ei ole saavutettu. Lisättävä piste valitaan kolmion  $xy$ -tasolla olevien kandidaattipisteiden joukosta se piste, jolla kandidaattipisteen ja kolmion kulmapisteiden muodostamien suorien ja kolmion tason välisten kulmien summa on pienin. Vaihtoehtoisesti voidaan valita aluksi koko alueen neljä

kulmapistettä ja lisätä kolmiointiin pisteitä siten, että valitaan mukaan aina kunkin kolmion sisällä oleva matalin piste, kunnes haluttu pistetiheys on saavutettu. (Vosselman et al., 2010)

Luokittelussa voidaan hyödyntää myös segmentointiin perustuvia suodatuksia, joista voi olla hyötyä etenkin rakennetulla alueella. Näillä segmentointimenetelmillä pyritään tunnistamaan maastomalliin kuulumattomat kohteet tai alueet pistepilvestä. Segmentoinnilla pyritään tunnistamaan tasaisia alueita, joiden voidaan olettaa olemaan maastoon kuulumattomia rakennettuja kohteita. (Vosselman et al., 2010)

### **2.1.6 Laserkeilausaineiston korkeuksien normalisointi**

Korkeuksien normalisointi tarkoittaa sitä menetelmää, jossa kustakin pistepilven pisteestä vähennetään maastomallin mukainen korkeus. Latvusmallin eli CHM:n (Canopy Height Model) muodostaminen on vastaava operaatio kuin korkeuksien normalisointi. CHM lasketaan DSM:n ja DTM:n erotuksena, ja sitä kutsutaankin joskus normalisoiduksi DSM:ksi. (Shan et al., 2018)

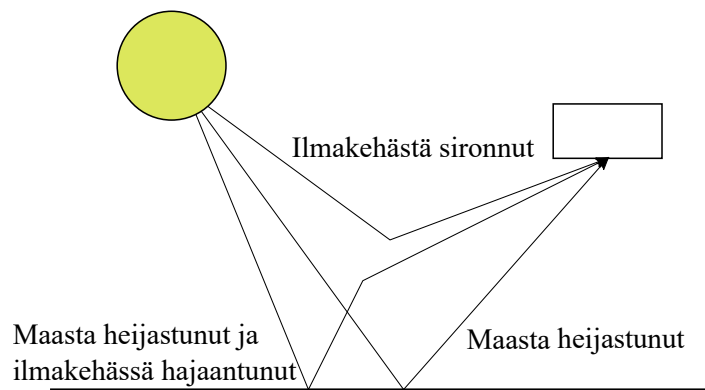
Normalisointi voidaan suorittaa käyttäen suoraan DTM:ää tai edellisessä luvussa mainittuja interpolointimenetelmiä ja maastopisteiksi luokiteltuja pisteitä. Mikäli käytetään rasterimuotoista DTM:ää voidaan maastomallin korkeus interpoloida normalisoitavan pisteen sijaintiin käyttäen jotain splini-interpolointia esimerkiksi bicubic-interpolaatiota. (Shan et al., 2018)

## **2.2 Ilmakuvaus kaukokartoituksessa**

Ilmakuvausta tiettävästi ensimmäistä kertaa yritti Gaspard-Félix Tournachon vuonna 1858 kuumailmapallosta käsin. Tämän jälkeen ilmakuvaus alkoi kehittymään ja ilmakuvia otettiin käyttäen kuumailmapalloja ja leijoja. Lentokoneen keksiminen kiihdytti ilmakuvauksen kehittymistä ja 1. maailmansodan aikaan ilmakuvaus oli jo yleinen toimenpide. Satelliittikuvien kehittyminen alkoi vuonna 1960 ja vuonna 1972 laukaistiin ensimmäinen Landsat-satelliitti (Land Satellite), joka havainnoi maata näkyvän valon ja lähi-infrapuna säteilyn aallonpituuksilla. (Campbell et al., 2011)

Ilmakuvauksessa sensori mittaa säteilyn voimakkuutta  $I$ , joka on maasta heijastuneen auringonvalon säteilyn  $I_S$ , ilmakehästä sironneen säteilyn  $I_O$ , ja maasta heijastuneen ilmakehässä hajaantuneen säteilyn  $I_D$  summa, joka on havainnollistettu kuvassa 10. Kaukokartoituksessa ollaan kiinnostuneita säteilystä  $I_S$ , joka ilmakuvien täysin varjoisilla alueilla tulisi olla 0. Käytännössä varjoissakin havaitaan säteilyä, joka on maasta heijastunutta ilmakehästä sironnutta säteilyä  $I_D$ , koska ilmakehästä sironnut säteily  $I_O$  on yleensä vakio kuvantamishetkellä ja siten suoraan suodatettavissa. (Campbell et al., 2011)

$$I = I_S + I_O + I_D \quad (3)$$



Kuva 10: Sensorin havaitsema säteily on maasta heijastuneen, ilmakehästä sironneen ja maasta heijastuneen ilmakehästä sironneen säteilyn summa.

### 2.2.1 Ilmakuvauksen käyttökohteet

Maantiedossa ilmakehävasta on hyödynnetty jo 1900-luvun alusta kartoitukseen. Kartoituksen lisäksi esimerkiksi maalajien ja kivilajien luokitteluun voidaan hyödyntää etenkin hyperspektrikameroilla tuotettuja ilmakehävasta ja satelliittikuvia, jolloin hyödynnetään tietoa erilaisten aallonpituuksien heijastuvuudesta erilaisista materiaaleista. Ilmakehävasta tehtyjen luokittelujen perusteella voidaan tuottaa maankäyttö tai maanpeite karttoja, joista esimerkiksi maankäyttökarttaa hyödynnetään verotuksessa ja maankäytön suunnittelussa. (Campbell et al., 2011)

Kasvillisuuspeitteestä ja kasvillisuuden latvuksista heijastuva säteily on usein kaukokartoitusmenetelmien ensimmäisenä havaitsema. Kaukokartoitusta voi hyödyntää yleisesti kasvil-

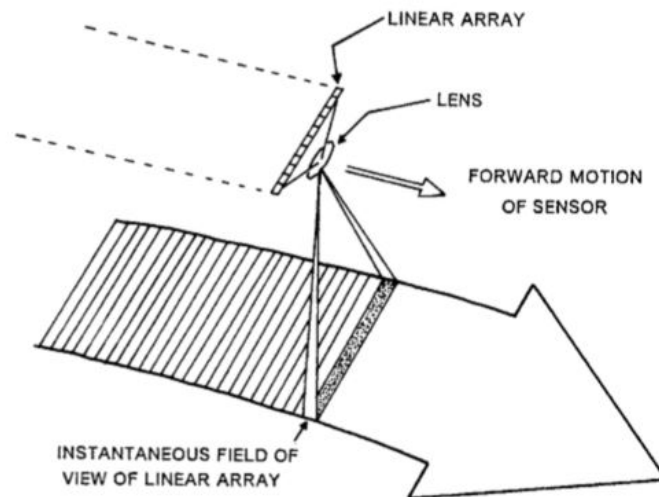
lisuuden, metsien ja viljelmien sadon seurantaan esimerkiksi tautien ja tuholaisien vaikutusten arvioinnissa. Lisäksi ilmakuvia ja etenkin satelliittikuvia hyödynnetään alueellisen ekologisen muutoksen havaitsemiseen. Kasvillisuuden kannalta ilmakuvauksessa aallonpituuksista lähi-infrapuna NIR (Near Infrared) on tärkeä. Valon heijastuminen kasvillisuudesta on voimakasta lähi-infrapuna-aallonpituudella kasvillisuus heijastaa 35 – 60% säteilystä, kun näkyvän punaisen valon aallonpituudella takaisin heijastuu 3 – 10% säteilystä. Tästä ominaisuudesta on johdettu normalisoitu kasvillisuusindeksi NDVI (Normalized Difference Vegetation Index). (Campbell et al., 2011; Tempfli et al., 2009)

### **2.2.2 Kvantamistekniikka**

Filmikamerat ovat olleet käytössä ilmakuvauksessa vuosikymmeniä ja niitä käytetään edelleen eikä digitaaliset kamerat ilmakuvauksessa ole onnistuneet syrjäyttämään näitä yhtä nopeasti kuin kuluttajakameroita. Ilmakuvauksessa käytetään lisäksi hyperspektri- tai monispektrikameroita tai -skannereita, jotka usein toisin kuin tavallinen kamera, toimivat peilien avulla skannaten yksittäisen juovan kerrallaan erottaen valon eri aallonpituudet optisesti. Ilmakuvausta voidaan suorittaa kohtisuoraan alaspäin eli nadiirista tai viistosti, mutta yleisemmin kohtisuoraan alaspäin ja eri lentolinjat ovat usein siten, että otetut kuvat ovat 60%:sti päällekkäin, jolla pyritään varmistamaan laadukas kuva-aineisto kuvantamialueella. Ilmakuvauksessa käytetty laitteisto koostuu laserkeilauslaitteiston tavoin GNSS- ja IMU-yksiköistä, tallennus-, keskusyksiköstä ja kuvantamislaitteistosta. (Tempfli et al., 2009)

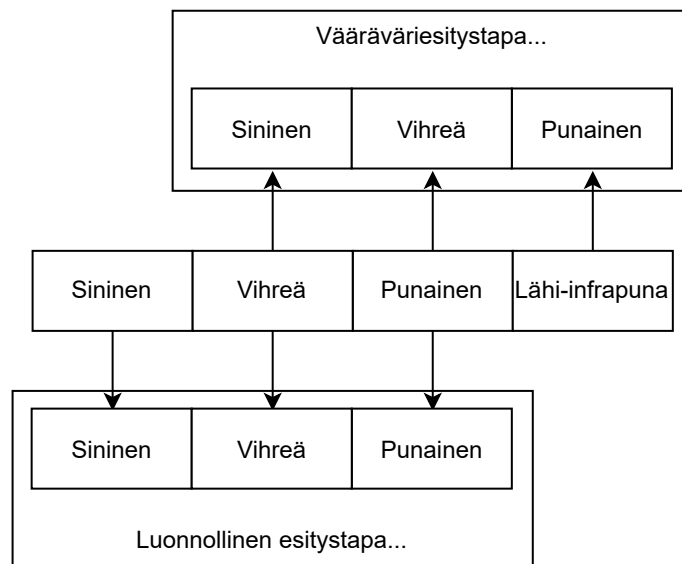
Digitaalisessa ilmakuvauksessa käytetään joko tavallisten kameroiden tapaan kaksiulotteisia (matrix array tai area array) tai yksiulotteisia (linear array) sensoreita. Kaksiulotteiset sensorit vastaavat filmikameran tai kuluttajakameran toimintaa, kun taas yksiulotteisilla havainnoidaan vain yksittäinen juova kerrallaan. Kuvassa 11 on havainnollistettu yksiulotteista sensoria. Monispektrikamerat ovat hyödyntävät yksiulotteista sensoria tai joissain tapauksissa nollaulotteisia sensoreita laserkeilausjärjestelmien tapaan. (Campbell et al., 2011)

Järjestelmissä on yleensä useita sensoreita, joista osa on pankromaattisia eli näkyvän aallonpituuden alueella ja osa näkyvän aallonpituusalueen ulkopuolisella alueella mitaten esi-



Kuva 11: Digitaalisessa ilmakuvaukammerassa käytetty yksiulotteinen sensori (Campbell et al., 2011).

merkiksi lähi-infrapunasäteilyä. Kullakin sensorilla on oma suodattimensa, jolla rajataan sensorille tuleva säteily haluttuun aallonpituuteen. Varsinainen kuva saadaan yhdistämällä sensorien mittaukset interpoloimalla, koska aallonpituudesta riippuen mittaustarkkuus vaihtelee. Kuvassa 12 esitetään yleiset esitystavat ilmakuville eli luonnollinen esitysmalli ja väri-infrapuna esitysmalli, jota kutsutaan usein myös vääräväriesitykseksi. (Campbell et al., 2011)



Kuva 12: Ilmakuvauksessa usein käytetyt väri-infrapunakuvat voidaan visualisoida väärävärikuvina tai luonnollisina kuvina.

Kaksiulotteista sensoria käyttäviä ilmakuvauuskameroita valmistaa ZI ja Microsoft, jotka vastaavat teknologialtaan perinteisiä ilmakuvauksessa edelleen käytettäviä filmikameroita (Tempfli et al., 2009). Esimerkiksi ZI:n DMC kamerassa neljä pankromaattista 7000 x 4000 pikselin kennoa tuottaen lopulta 3824 x 7680 pikselin kuvan ja neljä 3000 x 2000 pikselin monispektikennoa (lähi-infrapuna, punainen, sininen, vihreä) (Campbell et al., 2011). Leican ADS40 oli ensimmäinen digitaalinen ilmakuvauuskamera käyttäen yksiulotteisia sensoreita, jossa on kolme pankromaattista sensoria (eteenpäin, alaspäin ja taaksepäin) ja neljä monispektrisensoria (lähi-infrapuna, punainen, sininen, vihreä) kukin tarkkuudeltaan 12000 pikseliä. (Tempfli et al., 2009)

### 2.2.3 Kuvien erotuskyky

Erotuskyvyllä eli resoluutiolla tarkoitetaan pienintä sensorin havaitseman signaalin fysikaalista yksikköä. Ilmakuvauksessa erotuskyvyllä voidaan tarkoittaa spatiaalista, radiometristä tai spektrin erotuskykyä (Aksakal-Kocaman, 2008). Spatiaalinen erotuskyky tarkoittaa pikselin kokoa kohteessa. Tämä riippuu polttovälistä, sensorin koosta, näytteistysajasta, lentokorkeudesta ja lentonopeudesta (Campbell et al., 2011; Aksakal-Kocaman, 2008), jolloin spatiaalinen erotuskyky pituussuunnassa on:

$$l = v\Delta t, \quad (4)$$

jossa  $l$  on pikselin pituus maan pinnalla,  $v$  on lentonopeus ja  $\Delta t$  on laitteiston näytteistysaika. Esimerkiksi edellä mainitun Leica ADS40:n polttoväli (focal length) on 62.77 mm, näkökenttä (field of view) on  $64^\circ$ , ja näytteistys on pienimmillään 1.2 ms välein (Leica-Geosystems, 2004), jolloin pituussuunnassa spatiaaliseksi erotuskyvyksi saadaan 240 solmun nopeudella kuvattuna tarkimmillaan n. 15 cm. Leveyssuunnassa spatiaalinen erotuskyky on:

$$w = \frac{2H \tan(\frac{fov}{2})}{n}, \quad (5)$$

jossa  $w$  on pikselin leveys maan pinnalla,  $H$  on lentokorkeus,  $fov$  on näkökenttä ja  $n$  on sensorin leveys. Yhtälöstä saadaan johdettua, että lentokorkeudella 1500 m spatiaalinen erotuskyky olisi n. 15 cm ja siten 240 solmun nopeudella pikselit olisivat neliöitä.



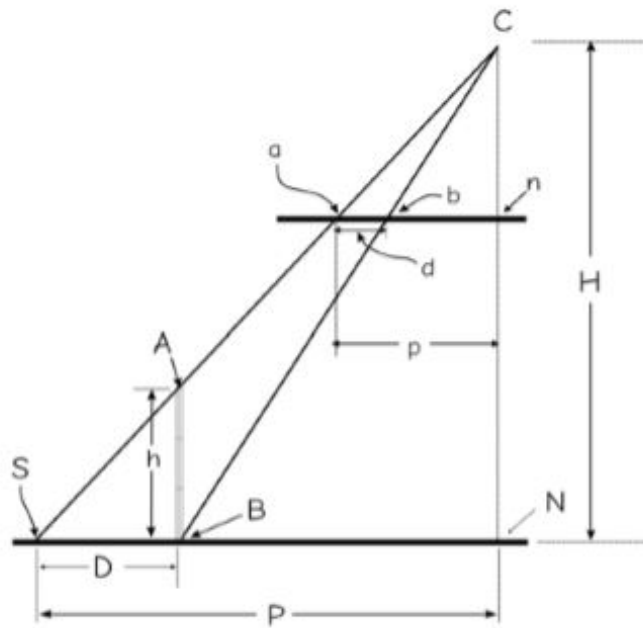
Radiometrinen erotuskyky tarkoittaa sensorin kykyä erottaa signaalin intensiteettivaihteluja, jotka vaihtelevat sensoreittain 8 ja 16 bitin välillä eli 256 erillisestä arvosta 65536 erilliseen arvoon (Aksakal-Kocaman, 2008). Esimerkiksi Leica ADS40:n radiometrinen resoluutio on 12 bittiä eli 4096 eri arvoa (Leica-Geosystems, 2004). Spektrin erotuskyvyllä tarkoitetaan laitteiston kykyä erotella ja mitata eri aallonpituuksia (Aksakal-Kocaman, 2008). Hyperspektrikameroissa voi spektrin erottelukyky olla alle 2nm (Tempfli et al., 2009). Esimerkiksi Leica ADS40:n spektrin erotuskyky eri värikanavittain on: infrapuna 835 – 885nm, punainen 610 – 660nm, vihreä 535 – 585nm, ja sininen 430 – 490nm (Leica-Geosystems, 2004).

#### **2.2.4 Orto-oikaisu**

Orto-oikaisulla pyritään korjaamaan ilmakuvan geometrisia vääristymiä. Prosessissa siirretään kukin pikseli kuvaussuuntaa, maastomallia ja kameraparametreja käyttäen sijaintiin, joka vastaa pikselin todellista maantieteellistä sijaintia. Tuloksena on ortografinen projektiotason, jota kutsutaan usein ortoilmakuvaksi. (Tempfli et al., 2009; Bolstad, 2012)

Vääristymät aiheutuvat kuvauskulmasta ja maaston pinnan korkeusvaihteluista eikä orto-oikaisemattomia kuvia tulisi useimmissa käyttötapauksissa käyttää kaukokartoituksessa. Korjauksessa muunnetaan perspektiiviprojektiot ortografiseksi projektioksi arvioimalla kontrollipisteitä käyttäen tai tuntemalla muutoin pinnan geometria. Kuvassa 13 on esitetty orto-oikaisun periaatetta yksinkertaistettuna. (Bolstad, 2012)

Ilmakuville on yleensä tiedossa kameran tarkka sijainti, kuvaussuunta sekä muut kameraparametrit. Joten korjaus voidaan tehdä käyttäen olemassa olevaa korkeusmallia tai laserkeilausaineistoa. Tällöin kameraparametreista voidaan johtaa perspektiiviprojektion matriisi, jota käyttäen projisoidaan kukin pikseli maastomallia käyttäen oikealle geografiselle sijainnille, jolloin näistä pikselien uudelleenprojektiosta muodostuu orto-oikaistu kuva. (Bolstad, 2012)



Kuva 13: Yksinkertaistus miten kohteiden korkeus  $h$  aiheuttaa siirtymää  $d$  kuvatasolla (Bols-  
 tad, 2012).

### 2.2.5 Radiometrinen korjaus

Radiometristä korjausta tehdään joko sensorien eroavaisuuksien vuoksi, ympäristön eroavaisuuksien vuoksi, tai kuvan parantamiseksi visuaalisista syistä. Sensorien eroavaisuuksien vuoksi korjausta tehdään laitteiston radiometrisellä kalibroinnilla. Ympäristön eroavaisuuksien vaikutusten korjaaminen on haastavampaa ja ympäristötekijöitä on esimerkiksi valaistusolosuhteet ja ilmasta heijastuva valo, jotka vaihtelevat eri kuvauskerroilla. (Tempfli et al., 2009; Honkavaara et al., 2011)

Ajankohdasta ja vuodenajasta riippuvia valaistusolosuhteita voidaan yksinkertaisimmillaan korjata muuntamalla pikseleiden arvot siten, että ne odotetusti vastaisivat tilannetta, jossa aurinko olisi zeniitissä. Tämä muunnos tehdään jakamalla kukin pikseli auringon kuvaushetken kulman sinillä, joka heikentää samalla kuvan dynamiikkaa. Ilmasta heijastuneen säteilyn määrää, jonka oletetaan olevan vakio kuva-alalla, voidaan korjata kuvista vähentämällä koko kuvasta jonkin kuvassa olevan tunnetun kohteen mitatun säteilyn ja oletetun todellisen säteilyn välinen erotus. Käytännössä tällaisia kohteita ovat sellaiset, joiden kuuluisi olla ns. mustia esimerkiksi varjot. (Tempfli et al., 2009)

### 2.2.6 Kasvillisuusindeksit

Kasvillisuusindeksit perustuvat kuvan eri värikanavien eli eri aallonpituusalueiden intensiteetteihin ja tietoon kasvillisuuden heijastavuudesta eri aallonpituusalueilla. Elävän kasvuston osalta tiedetään punaisen valon klorofyllin imevän punaista valoa lehtien solurakenteen heijastavan vihreää valoa ja infrapunasäteilyä. Tästä tiedosta on johdettu suhteet  $G/R$  ja  $NIR/R$ , joista ensiksi mainittua ei pidetä yhtä hyvänä mittarina kuin jälkimmäistä. Normalisoidun kasvillisuusindeksin eli NDVI:n (Normalized Difference Vegetation Index) on ensimmäisenä esittänyt CJ Tucker vuonna 1979. (Campbell et al., 2011)

$$NDVI = \frac{NIR - R}{NIR + R} \quad (6)$$

NDVI on edelleen paljon käytetty kasvillisuuden ja viljelmien seurannassa (Campbell et al., 2011). Mutta sen saturaatioherkkyyden vuoksi siitä on pyritty kehittämään paremmin kasvillisuutta kuvaavia indeksejä esimerkiksi parannettu kasvillisuusindeksi EVI (Enhanced Vegetation Index). EVI:n hyödyntämä sinisen valon aallonpituus on usein alttiimpi ilmakehän vaikutuksille eikä kuvaa biofysisiä ominaisuuksia (Jiang et al., 2008). Jiang et al., 2008, ovat esittäneet indeksin EVI2, joka on johdettu EVI:stä käyttämättä sinisen valon aallonpituutta. EVI2:n vakio 2.4 on saatu sovittamalla indeksi EVI2 vastaamaan indeksiä EVI.

$$EVI = 2.5 \frac{NIR - R}{NIR + 6R - 7.5B + 1} \quad (7)$$

$$EVI2 = 2.5 \frac{NIR - R}{NIR + (6 - \frac{7.5}{2.4})R + 1} \quad (8)$$

## 3 KAUKOKARTOITUS- JA PAIKKATIETOAINIESTOT

Paikkatietojärjestelmät hyödyntävät yleisesti kaukokartoitusaineistoja rasteri- ja vektorimuodoissa. Nykyisissä järjestelmissä on usein myös mahdollista hyödyntää laserkeilausaineistoa pistepilvenä. Yhteistä näissä aineistoissa on, että aineisto on sidottu johonkin geografiseen sijaintiin, joihin aineistot ovat kytkettyjä koordinaattijärjestelmien avulla. (Bolstad, 2012)

### 3.1 Koordinaattijärjestelmä Suomessa

Suomessa koordinaattijärjestelmänä käytetään usein EUREF-FIN -datumia eli parametrijoukkoa, joka on JHS-suosituksena sitova julkisessa hallinnossa. Tämä järjestelmä on ETRS89 (European Terrestrial Reference System 1989) terrestrisen vertausjärjestelmän Suomalainen realisaatio, joka eroaa satelliittipaikannuksessa käytetystä WGS84:stä (World Geodetic System 1984) siten, että se on ellipsisen koordinaattijärjestelmän parametrit ovat optimoituja Suomen alueelle, kun WGS84:ssä parametrit ovat maan massakeskipisteen mukaan globaalisti sovitettuja. Geodeettiset koordinaatit eli pituus- ja leveysasteet eroavat näissä siten hieman toisistaan. Mannerlaattojen liikkeen ja maankohoamisen vuoksi koordinaatit elävät, mutta datumit ovat kuitenkin kiinnitettyjä johonkin tiettyyn ajanhetkeen eli epookkiin. Koordinaattijärjestelmän esittämiseen on useita tapoja, mutta yksi käytetty esitystapa on OGC:n (Open Geospatial Consortium) WKT (Well Known Text). (JHS196, 2016; Shan et al., 2018)

JHS-suosituksen mukaan projektiokoordinaatistona tulisi julkishallinnon käyttää ETRS89-TM35FIN tasokoordinaatistoa, joka on osa EUREF-FIN määrittelyä. Tulosteessa 1 on WKT muotoinen määrittely ETRS89-TM35FIN projektiokoordinaatistosta. Koordinaatistossa yksikkönä on metri ja se on suorakulmainen (JHS197, 2016). Suorakulmainen koordinaatisto helpottaa laajojen paikkatietoaineistojen käsittelyä, koska tasossa etäisyydet ovat helpompia laskea kuin ellipsoidin pinnalla, sekä rasterimuotoisessa aineistossa yksittäinen pikseli on kaikkialla saman kokoinen.

## Tuloste 1: ETRS89-TM35FIN projektiokoordinaatistomääritelmä WKT muodossa

---

```
1 PROJCS["ETRS89-TM35FIN",
2   GEOGCS["GCS_EUREF_FIN",
3     DATUM["ETRS_1989",
4       SPHEROID["GRS_1980",6378137,298.257222101]],
5     PRIMEM["Greenwich",0],
6     UNIT["degree",0.0174532925199433]],
7   PROJECTION["Transverse_Mercator"],
8   PARAMETER["latitude_of_origin",0],
9   PARAMETER["central_meridian",27],
10  PARAMETER["scale_factor",0.9996],
11  PARAMETER["false_easting",500000],
12  PARAMETER["false_northing",0],
13  UNIT["metre",1],
14  AUTHORITY["EPSG","9001"]]]
```

---

---

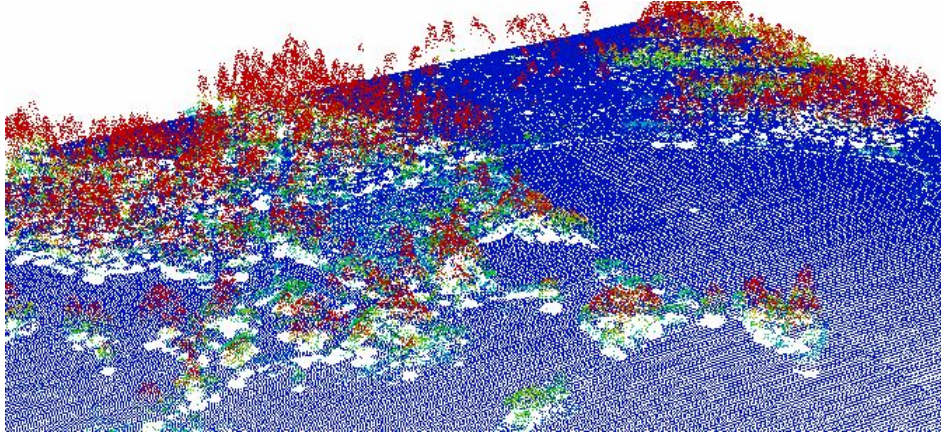
### 3.2 Laserkeilausaineisto

Laserkeilausaineistolle on luotu formaatti LAS, joka on ASPRS:n ylläpitämä (American Society for Photogrammetry and Remote Sensing) ja LWG:n (LAS Working Group) määrittelemä. LWG:ssä formaatin määrittelyyn osallistuu niin LiDAR laitevalmistajat kuin ohjelmistokehittäjätkin. Formaatin ensimmäinen 1.0 versio julkaistiin vuonna 2002 ja formaatin viimeisin versio 1.4 on vuodelta 2011. (Shan et al., 2018)

Formaatissa ennen versiota 1.4 spatiaalinen referenssi eli käytetty koordinaattijärjestelmä oli liitettyä GeoTIFF -formaattissa, mutta versiossa 1.4 koordinaattijärjestelmä määritellään WKT-formaatissa. Käsittelytehokkuuden vuoksi formaatista ei ole mahdollista poistaa olemassa olevia pisteitä, mutta näitä voidaan luokitella kuulumaan luokkaan, jossa näitä pisteitä ei käsitellä. (Shan et al., 2018)

Formaatti määrittelee eri tyyppisiä tietueita pisteille. Perustasolla pisteen tietue sisältää seuraavat tiedot: id, koordinaatit  $(x, y, z)$ , intensiteetti, luokka, pulssin järjestysnumero, pulssien lukumäärä, keilaimen mittauskulma, ja lentosuunta. Muissa tietuetyypeissä pisteelle voidaan lisäksi määrittää väriarvo 3 tai 4 kanavaisena, aikaleima sekä paluusignaalin koko jakauma. (LAS Specification Version 1.4, 2010)

LAS-formaatin lukemiseen ja käsittelyyn on saatavilla avoimen lähdekoodin kirjasto PDAL Point Data Abstraction Library, joka on korvannut libLAS -kirjaston (PDAL Contributors, 2018). Kuvassa 14 on visualisoitu laserkeilausaineiston pistepilveä.



Kuva 14: Laserkeilausaineistoa maanpinnan korkeudelle siirrettynä ja korkeuden mukaan värjättyinä. Kuvassa väritys on sinisestä punaiseen pisteen korkeuden mukaan.

### 3.3 Georeferoitu rasteriaineisto

Kuva voidaan yhdistää maantieteellisiin koordinaatteihin yksinkertaisimmillaan lineaari-muunnoksella kuvakoordinaatistosta, jolloin kuvakoordinaatit  $(i, j)$  muunnetaan maantieteellisiin  $(x, y)$  koordinaatteihin (Tempfli et al., 2009). Yleisesti riittävä muunnos on siis muotoa:

$$\begin{aligned}x &= ai + bj + c \\y &= di + ej + f.\end{aligned}\tag{9}$$

Kuvaa georeferoitaessa kuvasta etsitään kontrollipisteitä eli GCP:itä (Ground Control Points), joiden sijainnit tunnetaan käytetyssä projektiokoordinaatistossa. Tällöin yhtälöparin vakiot voidaan sovittaa pienimmän neliösumman menetelmällä. Georeferoiduissa kuvassa on mukana tieto käytetystä koordinaattijärjestelmästä ja kuvan siirto toiseen geodeettiseen datumiin vaatii enemmän laskentaa ja kuvan uudelleennäytteistämistä. (Tempfli et al., 2009)

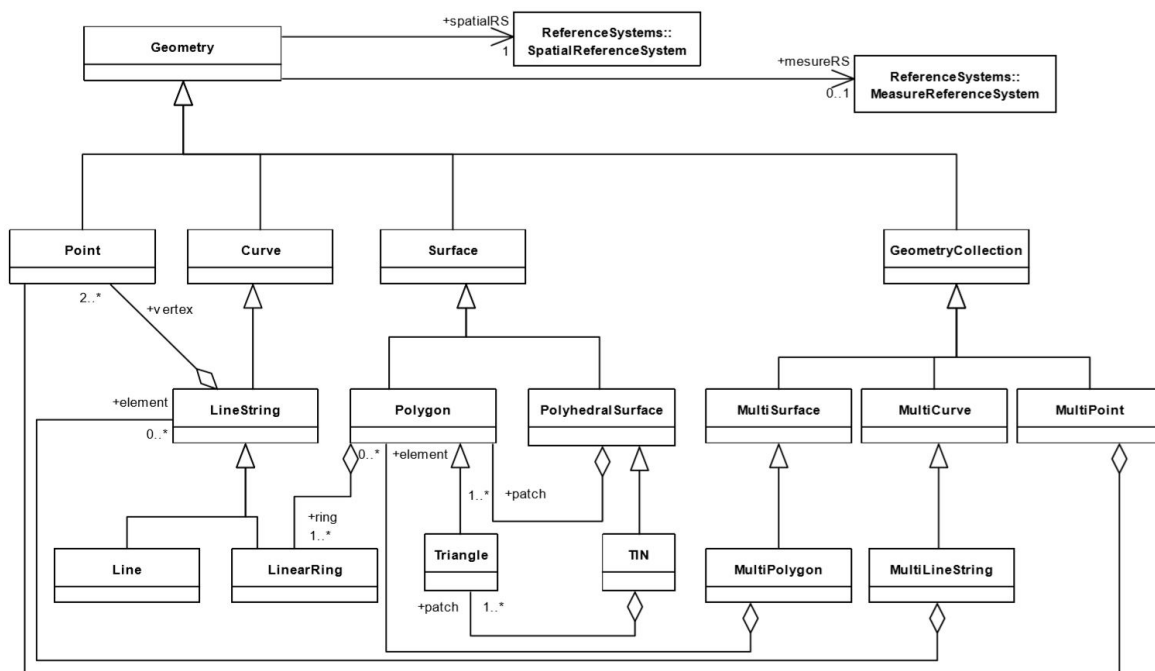
Avoimen lähdekoodin kirjasto GDAL (The Geospatial Data Abstraction Library) pystyy lukemaan ja kirjoittamaan lukuisia eri kuvaformaatteja ja sisältää työkaluja koordinaatis-

tomuunnoksiin. Osa kuvaformaateista tarvitsee erillisiä kaupallisia kirjastoja. Esimerkiksi yleisesti käytetyssä GeoTIFF -formaatissa on kuvaan liitetty tieto käytetystä koordinaattijärjestelmästä ja georeferoinnin lineaarimuunnoksessa tarvittavat parametrit, joilla voidaan laskea kuvatason erillisten pisteiden sijainnit kuvaan liitettyssä koordinaattijärjestelmässä. (GDAL/OGR contributors, 2019)

### 3.4 Vektoriaineisto

Vektoriaineistossa voi geometrinen tarkkuus olla suurempi, ylläpitäminen ja tiedon hakeminen on helpompaa kuin rasterimuotoisella aineistolla. Tietorakenteet ovat kuitenkin monimutkaisempia ja tiedon esitys voi olla vaikeammin ymmärrettävää (Mårtensson, 2011). Vektoriaineistossa yksikertaisimmillaan kohteen geometrian tyyppi on piste, jolla on  $xy$ -tai  $xyz$ -koordinaatit ja viittaus käytettyyn koordinaattijärjestelmään. Lisäksi kohteeseen on usein liitetty muuta tietoa, jotka kutsutaan attribuuttitiedoksi tai temaattiseksi tiedoksi, jota voidaan hyödyntää kohteen visualisoinnissa. Muita geometriatyyppejä ovat viiva ja alue, sekä näiden kaikkien monigeometriset tyypit. (Huisman et al., 2009)

OGC on määritellyt vektorimuotoisen paikkatietoaineiston geometrialle standardoidun mallin OGC Simple Feature Access, jonka luokkahierarkia on esitetty kuvassa 15, geometrialuokkien metodit ja spatiaaliset relaatiot ja operaatiot, jotka tulisi standardin toteuttamiseksi implementoida. Standardi määrittelee myös WKT-formaatin, jolla geometrian voi esittää tekstimuotoisena (Herring, 2011). Vektorimuotoista paikkatietoaineistoa voi käsitellä esimerkiksi avoimen lähdekoodin OGR (OGR Simple Features Library) -kirjastolla, joka on osa GDAL -kirjastoa. OGR kirjasto ei täysin toteuta OGC Simple Feature Access standardia, mutta sillä pystyy kuitenkin lukemaan ja kirjoittamaan useimpia vektorimuotoisia paikkatietoaineistoja. (GDAL/OGR contributors, 2019)



Kuva 15: Vektorimuotoisen paikkatietoaineiston OGC standardoidun geometrian luokkahierarkia (Herring, 2011).



## 4 DIGITAALINEN KUVANKÄSITTELY

Tässä kappaleessa käydään läpi työssä toteutetut kuvankäsittelymenetelmät, joita hyödynnetään menetelmässä metsikkökuvioiden tuottamiseen. Nämä kaikki kuvatut menetelmät toteutettiin osana työtä olemassa olevien toteutusten puutteen vuoksi. Valmiit toteutukset eivät olleet soveltuvia, koska paikkatietoaineistossa voi olla alueellisia aukkoja eli pikseleillä ei ole aina arvoa, kuvankäsittelymenetelmiä hyödynnettiin ketjutettuna Just-In-Time -filosofialla vain lopullisessa prosessissa käsiteltävällä alueella, ja kuvien pikseleiden tietotyypit ovat vaihtelevia.

Kuva voidaan määritellä kaksiulotteiseksi funktioksi  $f(x, y)$ , jossa  $x$  ja  $y$  ovat kuvatason koordinaatteja ja  $f$  kuvan intensiteetti koordinaateissa  $(x, y)$ . Kun  $f$ :n intensiteettiarvot ovat äärellisiä ja diskreettejä suureita, niin kuvaa kutsutaan digitaaliseksi kuvaksi. Digitaalinen kuva koostuu äärellisestä joukosta elementtejä, joista jokaisella on tietty sijainti ja arvo. Näitä elementtejä kutsutaan yleisimmin pikseleiksi. Digitaalinen kuva ei rajoitu vain näkyvän valon aallonpituuden alueelle vaan voi olla mitä tahansa mitattavissa olevaa elektromagneettista säteilyä gammasäteilystä radioaaltoihin. (Gonzalez et al., 2006)

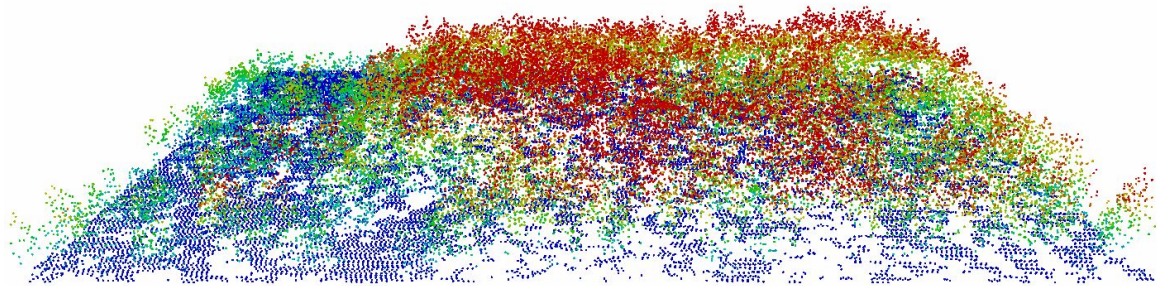
Kuvankäsittelyn ja kuva-analyysin tai tietokonenäön rajan määrittämiselle ei ole yleistä yksimielisyyttä, mutta yhtenä rajana voidaan pitää rajaamalla kuvankäsittely käsittämään prosessit ja menetelmät, joissa syötteenä ja tulosteena on molempina kuva. Tämä on keinotekoinen raja eikä silloin esimerkiksi kuvan keskimääräisen intensiteetin laskeminen olisi kuvankäsittelyä vaan kuva-analyysiä tai tietokonenäköä. Toisaalta tietokonenäön toinen äärilaita on saada tietokone ymmärtämään kuvia ihmisen tavoin eli tällöin ala on tekoälytiedettä. (Gonzalez et al., 2006)

Tietokonenäköä voidaan ajatella käännteiseksi tietokonegrafiikaksi. Tietokonegrafiikassa muodostetaan kuva digitaalisen mallin ja ominaisuuksien mukaan, kun tietokonenäkö pyrkii muodostamaan mallin ja ominaisuudet kuvasta, sekä tulkitsemaan näitä. (Szeliski, 2011)

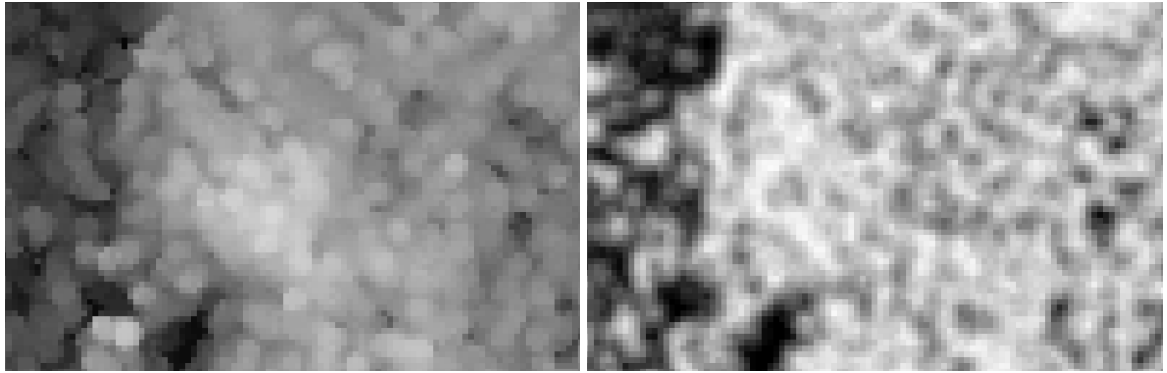
## 4.1 Laserkeilausaineiston rasterointi

Laserkeilausaineistoa rasteroidaan usein latvusmallia varten, jolloin tavoitteena on tuottaa kuva, jossa pikselit vastaavat latvuksen korkeutta maanpinnasta. Yksinkertaisimmillaan menetelmänä on käytetty pikselin alueelle osuvan normalisoidun laserkeilausaineiston mittauspisteen suurinta korkeuden arvoa. Tällöin kuvalle voi jäädä pikseleitä, joilla ei ole mitään arvoa, koska laserkeilausaineissa on usein alueita, jotka ovat niin harvoja, ettei kaikkien pikseleiden alueelle osu yhtään mittauspistettä. Reikien poistamiseksi tai minimoimiseksi on esitetty esimerkiksi menetelmä, jossa ensin rasteroidaan karkealla tasolla tarkeantaen resoluutiota, tai erilaisia keskiarvoistamiseen tai mediaanisuodatukseen perustuvia menetelmiä. Muita menetelmiä rasterointiin on luonnollisimman naapurin interpolaatio tai IDW -menetelmällä painotettuun naapurustoon perustuva interpolaatio. (C. Chen et al., 2017; Mielcarek et al., 2018)

Laserkeilausaineistosta voidaan rasteroida myös muita ominaisuuksia kuin puuston korkeutta kuvaavaa latvusmallia. Kasvillisuuden tiheyttä voidaan arvioida maahan osuneiden mittapisteiden ja maan pinnan yläpuolella olevien mittapisteiden välisestä suhteesta (Bicking, 2009). Kuvassa 17 on rasteroituna latvusmalli ja kasvillisuuden tiheys, jotka ovat tuotettu tässä työssä toteutetulla rasterointimenetelmällä, jossa rasteroidaan kunkin pikselin alueelle ja sen naapurustoon osuvia mittauspisteitä painottamalla IDW -menetelmällä.



Kuva 16: Maanpinnalle normalisoitua laserkeilausaineistoa visualisoituna.



Kuva 17: Kuvassa 16 visualisoidusta pistepilvestä rasteroitu latvusmalli vasemmalla, ja oikealla kasvillisuuden tiheys.

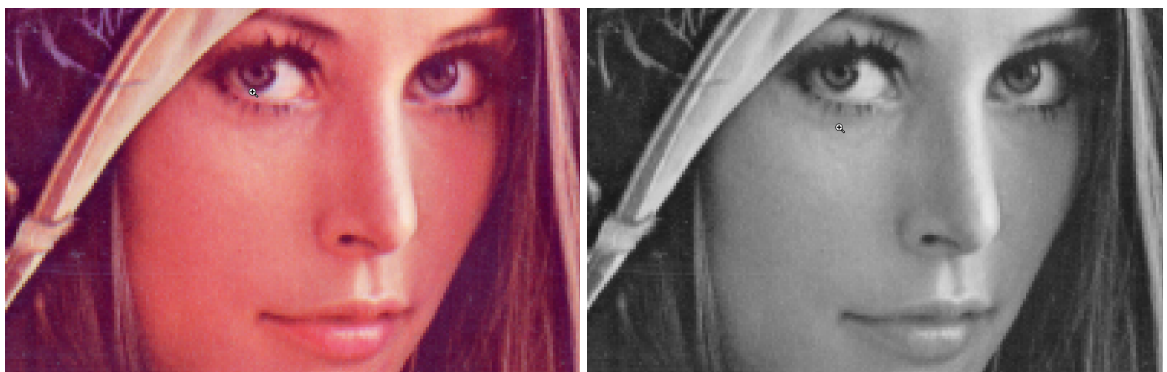
## 4.2 Pistemäiset operaatiot

Kuvankäsittelyn useimmiten yksinkertaisimmat operaatiot ovat pistemäisiä. Nämä kohdistuvat yksittäiseen pikseliin ja esimerkkeinä näistä ovat kirkkauden ja kontrastin säätö. Erilaiset kuvien ja värikanavien yhdistämis-, tilastojen laskentaoperaatiot ja värimallin muunnokset ovat myös pistemäisiä operaatioita. Yksi yleisestikäytetty epälineaarinen pistemäinen operaatio on Gamma-korjaus (Szeliski, 2011)

$$g(x) = |f(x)|^{\frac{1}{\gamma}}. \quad (10)$$

Värikuvan muunnos harmaasävykuvaksi voidaan tehdä seuraavalla pistemäisellä operaatiolla

$$g(x) = 0.3f_r(x) + 0.59f_g(x) + 0.11f_b(x). \quad (11)$$



Kuva 18: Alkuperäinen värikuva ja yhtälön 11 mukaan kuva muunnettu värikuvasta harmaasävykuvaksi.

Kasvillisuusindeksin tuottaminen on myös pistemäinen operaatio, jossa pikselin arvo lasketaan seuraavasti:

$$g(x) = \frac{f_{nir}(x) - f_r(x)}{f_{nir}(x) + f_r(x)}. \quad (12)$$

### 4.3 Spatiaalinen suodatus

Suodatus termi on lainattu signaalinkäsittelyn taajuusavaruuden käsittelystä, jossa voidaan valita tiettyjä taajuuksia esimerkkinä alipäästösuodatus, jolla saadaan silottavaa vaikutusta rajoittamalla korkean taajuuden signaaleja. Spatiaalinen suodatus on yksi kuvankäsittelyn perusteista. Spatiaalinen suodatin koostuu naapurustosta ja operaatiosta, joka suoritetaan kuvan kullekin kuvan pikselille sen naapurustossa. Naapurusto on usein neliömäinen alue käsiteltävän pikselin ympärillä, jota kutsutaan usein ikkunaksi. Suodatuksella lasketaan kullekin pikselille uusi arvo. Jos arvo lasketaan lineaarikombinaationa naapuruston pikselien arvoista, puhutaan lineaarisesta spatiaalisesta suodattimesta, muutoin puhutaan epälineaarista spatiaalisesta suodattimesta. (Gonzalez et al., 2006)

Lineaarisia spatiaalisia suodattimia ovat korrelaatio ja konvoluutio. Korrelaatio mittaa kahden signaalin samankaltaisuutta, kun konvoluutio mittaa toisen signaalin vaikutusta toiseen. Käytännössä erona on suodattimen (kernel) kääntyminen 180 asteella (Gonzalez et al., 2006).

Kuvan  $f$  ja suodattimen  $w$  konvoluutio esitetään seuraavasti:

$$(f * w)(x, y) = \sum_{s=-a}^{s=a} \sum_{t=-b}^{t=b} w(s, t) f(x - s, y - t), \quad (13)$$

jossa suodatin  $w$  on kokoa  $N \times M$ , joten  $a = N/2 + 1$  ja  $b = M/2 + 1$ . Esimerkkinä lineaarisesta spatiaalisesta suodattimesta on aritmeettinen keskiarvo suodatin, jossa suodattimen kaikki elementit saavat arvon  $\frac{1}{NM}$ , jolloin kukin pikseli saa konvoluutiossa arvoksi naapuruston keskiarvon (Gonzalez et al., 2006). Mikäli suodatin on separoituva eli voidaan matriisimuodossa esittää kahden vektorin  $v$  ja  $h$  tulona  $w = vh^T$ , voidaan konvoluutio suorittaa kahden konvoluution summana ja tällöin konvoluution laskennan kompleksisuus pienenee  $N \times M$  laskentaoperaatiosta  $N + M$  laskentaoperaatioon pikseliä kohden. Monia

suodattimia mukaan lukien Gaussin suodatinta voidaan separoida ja joskus voidaan ainakin approksimoida suodatinta separoituvalla suodattimella (Szeliski, 2011).

$$(f * w) = ((f * v) * h^T) \quad (14)$$

Epälineaarisia spatiaalisia suodattimia ovat muut spatiaaliset suodattimet eli niissäkin on vastaavasti määritelty naapurusto ja operaatiot, josta tulos riippuu. Yhtenä esimerkkinä on ääriarvosuotimet min ja max, joilla tulos on joko pienin tai suurin arvo naapurustossa. Spatiaalisia suodattimia on myös rekursiivisina, jolloin suodatus riippuu edellisen suodatuksen tuloksesta. Signaalinkäsittelyssä näitä kutsutaan impulssivasteeltaan äärettömiksi eli IIR (Infinite Impulse Response) suodatin, kun edellä mainittu aritmeettinen keskiarvo suodatin on impulssivasteeltaan äärellinen eli FIR (Finite Impulse Response). (Gonzalez et al., 2006; Szeliski, 2011)

#### 4.3.1 Gaussin silotus

Kuvassa on usein satunnaista kohinaa tai artefakteja, joita on tarve saada poistettua. Kohina voi olla normaalijakautunutta tällöin kuva voidaan määrittellä  $f(x, y) = f_r(x, y) + N(0, \sigma)$ , jossa  $f_r$  on kohinaton kuva ja  $N(0, \sigma)$  normaalijakautunutta nollakeskiarvoista kohinaa keskihajonnalla  $\sigma$ . Kohinaa voidaan vähentää kuvasta keskiarvoistamalla. Gaussin silotus on lineaarinen spatiaalinen suodatus ja käyttää Gaussin suodatinta  $G(x, y)$ , jossa painokertoimet johdetaan normaalijakauman yhtälöstä. (Stockman et al., 2001)

$$G(x, y) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (15)$$

Diskretisoituna suodattimeksi yhtälön vakiokerroin  $\frac{1}{\sqrt{2\pi\sigma}}$  voidaan jättää huomioimatta ja jakaa kukin arvo ytimen kaikkien arvojen summalla. Yhtälöissä 16 on esitetty  $3 \times 3$  Gaussin suodatin, joka on johdettu yksiulotteisten suodatinten  $[1 \ 2 \ 1]$  ja  $[1 \ 2 \ 1]^T$  tulosta, ja  $7 \times 7$  kokoinen Gaussin suodatin on diskretisoitu kokonaislukumuodossa varianssilla  $\sigma^2 = 2$ , jolla on suodatettu kuva 19.

$$G_{3 \times 3} = \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} \quad G_{7 \times 7} = \frac{1}{1098} \begin{bmatrix} 1 & 3 & 7 & 9 & 7 & 3 & 1 \\ 3 & 12 & 26 & 33 & 26 & 12 & 3 \\ 7 & 26 & 55 & 70 & 55 & 26 & 7 \\ 9 & 33 & 70 & 90 & 70 & 33 & 9 \\ 7 & 26 & 55 & 70 & 55 & 26 & 7 \\ 3 & 12 & 26 & 33 & 26 & 12 & 3 \\ 1 & 3 & 7 & 9 & 7 & 3 & 1 \end{bmatrix} \quad (16)$$



Kuva 19: Kuva silotettuna Gaussin  $7 \times 7$  suodattimella ja  $\sigma^2 = 2$ . Kuvasta näkyy Gaussin silotuksen sumentava ja kuvan terävyyttä vähentävä vaikutus.

### 4.3.2 Mediaanisuodatus

Mediaanisuodatus on epälineaarinen spatiaalinen suodatus, jolla pyritään silottamaan kuvasta kohinaa. Keskiarvoistava suodatus Gaussin silotus tai keskiarvosuodatus toimii hyvin homogeenisessa naapurustossa normaalijakautuneen kohinan poistossa, mutta kahden erilaisten alueiden rajoilla reunan piirteet sumentuvat, joiden säilyttämisessä mediaanisuodatus toimii paremmin eli voidaan puhua reunat säilyttävästä suodatuksesta, joka vähentää kuvan terävyyttä vähemmän kuin Gaussin silotus (Stockman et al., 2001). Mediaanisuodatus on hyvä impulssikohinan tai ns. salt-and-pepper -kohinan suodatuksessa (Gonzalez et al., 2006).

Mediaani tarkoittaa suuruusjärjestettyjen arvojen keskimmäistä arvoa. Jos arvoja on parillinen määrä, niin on myös kaksi mediaania, jolloin käytetään näiden keskiarvoa. Mediaanisuuodatus on laskennallisesti lineaarisia spatiaalisia suodatusta laskennallisesti vaativampi vastaavalla naapuruston koolla. Naapuruston arvot järjestetään suuruusjärjestykseen ja valitaan niistä mediaani, jolla korvataan pikselin arvo. Tällöin suodatuksessa kukin pikseli saa naapurustonsa kaltaisen arvon. Kuvassa 20 on käytetty  $7 \times 7$  kokoista naapurustoa mediaanisuuodatuksessa. (Stockman et al., 2001)



Kuva 20: Kuva silotettuna  $7 \times 7$  mediaanisuuodattimella. Kuvasta näkyy mediaanisuuodatuksen voimakkaat pienet yksityiskohdat hävittävä vaikutus säilyttäen kuitenkin reunat.

### 4.3.3 Bilateraallinen suodatus

Bilateraalinen suodatus silottaa kuvia säilyttäen rajat käyttäen epälineaarista yhdistelmää naapuruston arvoista. Aiemmin mainitun Gaussin silotuksen heikkoutena on sen ominaisuus sumentaa rajoja, joiden säilyttäminen on monesti hyödyllistä. Suodatus voidaan tehdä kuvan kaikille värikanaville kerralla esimerkiksi CIE-Lab -väriavaruudessa, jolloin suodatus pyrkii säilyttämään kuvan rajoja ihmissilmän havainnoinnin suhteen johtuen silmän havainnoinnin suhteen CIE-Lab väriavaruuden ortogonaalisuudesta. (Tomasi et al., 1998)

Tomasi et al., 1998, esittävät suodatuksen seuraavasti:

$$\mathbf{h}(\mathbf{x}) = \frac{1}{\mathbf{x}} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \mathbf{f}(\xi) c(\xi, \mathbf{x}) s(\mathbf{f}(\xi), \mathbf{f}(\mathbf{x})) d\xi \quad (17)$$

$$k(\mathbf{x}) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} c(\xi, \mathbf{x}) s(\mathbf{f}(\xi), \mathbf{f}(\mathbf{x})) d\xi, \quad (18)$$

jossa  $c(\xi, \mathbf{x})$  mittaa geometristä etäisyyttä ja  $s(f(\xi), f(\mathbf{x}))$  spektraalista samankaltaisuutta. Etäisyys- ja samankaltaisuusfunktioina on käytetty Gaussin funktioita (Tomasi et al., 1998):

$$c(\xi, \mathbf{x}) = e^{-\frac{1}{2} \left( \frac{\|\xi - \mathbf{x}\|}{\sigma_d} \right)^2} \quad (19)$$

$$s(\xi, \mathbf{x}) = e^{-\frac{1}{2} \left( \frac{\|\mathbf{f}(\xi) - \mathbf{f}(\mathbf{x})\|}{\sigma_r} \right)^2}, \quad (20)$$

jossa  $\sigma_d$  on etäisyyden hajonta ja  $\sigma_r$  spektraalisen samankaltaisuuden hajonta. Geometrinen etäisyysfunktio rajoittaa myös laskennallisesti järkevää ikkunan kokoa suodatusta laskettaessa. Bilateraalista suodatuksen tulosta on vaikeampi analysoida sen epälineaarisuuden takia, mutta yleiskäyttöisenä ja yksinkertaisena menetelmänä se on hyvin käyttökelpoinen reunat säilyttävään kuvan silotukseen (Tomasi et al., 1998). Kuvasta 21 näkee tämän bilateraalisuodatuksen reunat säilyttävän ominaisuuden.



Kuva 21: Kuva silotettuna bilateraalisuodatuksella käyttäen spatiaalista sädettä  $\sigma_c = 5$  ja spektraalista sädettä  $\sigma_s = 15$ . Kuvasta näkyy bilateraalisuodatuksen reunat säilyttävä muutoin voimakkaastikin silottava vaikutus.



#### 4.3.4 Mean Shift -suodatus

Mean Shift (MS) algoritmia käytetään segmentointiin (Szeliski, 2011), klusterointiin (Carreira-Perpiñán, 2015) ja rajat säilyttävään silotukseen (Solomon et al., 2014). Silotuksessa Mean Shift algoritmi voi olla ensimmäiseltä iteraatioltaan identtinen bilateraalisuodatuksen kanssa tai käänteisesti iteratiivinen bilateraalisuodatus johtaa samaan kuin Mean Shift -suodatus (Solomon et al., 2014). Kuvassa 22 on käytetty vastaavia parametreja kuin bilateraalisella suodatuksella 21, mutta tuloksessa näkyy suodatuksen klusteroiva vaikutus.

Itse Mean Shift algoritmi on hyvin yksinkertainen ja se on keksitty useita kertoja, mutta nimitys Mean Shift on vuodelta 1975, jolloin Fukunaga ja Hostetler kuvasivat sumentavan version algoritmista eli BMS (Blurring Mean Shift) ydinestimointiin eli KDE:hen (Kernel Density Estimation) käyttäen Epanechnikovin ydintä gradienttimenetelmään gradientiksi. He esittivät ajatuksen menetelmän käyttämisestä klusterointiin ja kohinan poistoon. Menetelmä on yleistynyt kuvankäsittelyssä ja tietokonenäössä 2000-luvulla Comaniciu et al., 2002, esitellyä menetelmän tehokkuuden kuvan suodatuksessa, segmentoinnissa ja myöhemmin seurannassa. (Carreira-Perpiñán, 2015; Comaniciu et al., 2002)

Mean Shift iteraatiossa lasketaan naapuruston painotettu keskiarvo käyttäen sopivaa ydintä  $K$ , joka voi olla Gaussin ydin. Käyttämällä erillisiä ytimiä geometriseen etäisyyteen ja spektraaliseen etäisyyteen painottamiseksi menetelmän ensimmäinen iteraatio vastaa bilateraalista suodatusta (Solomon et al., 2014). MS ja BMS on esitetty algoritmeissa 1 ja 2.

---

#### Algoritmi 1: Mean shift algoritmi (MS)

---

```
1 input : dataset  $x_1, \dots, x_N$ 
2 output : dataset
3 begin
4   for  $n = 1, \dots, N$ 
5      $x \leftarrow x_n$ 
6     do
7        $x \leftarrow (\sum_{m=1}^N K(x, x_m)x_m) / (\sum_{m=1}^N K(x, x_m))$ 
8     while not converged
9      $z_n \leftarrow x$ 
10  end
11 return  $z_1, \dots, z_N$ 
12 end
```

---

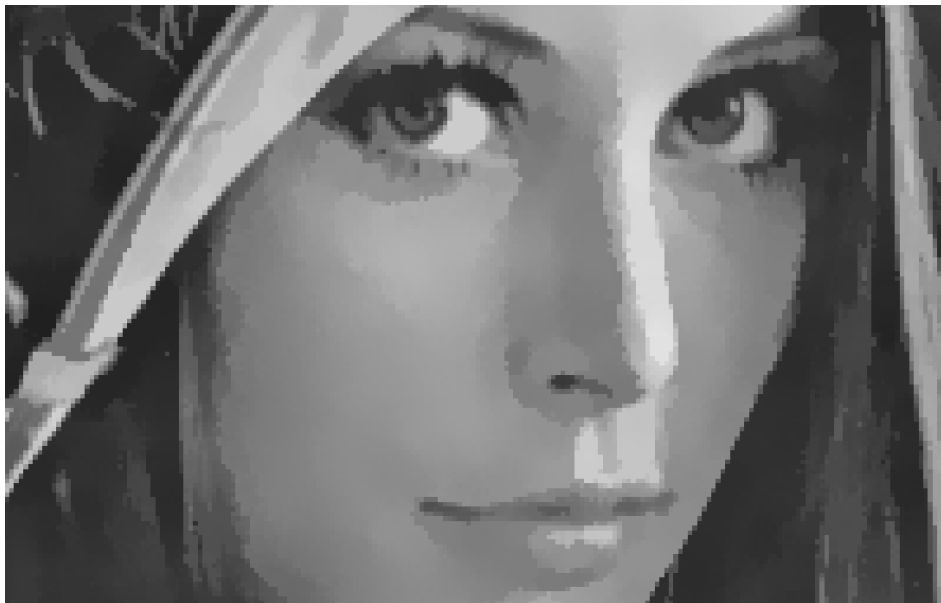
## Algoritmi 2: Sumentava Mean Shift algoritmi (BMS)

---

```
1 input: dataset  $x_1, \dots, x_N$ 
2 output: dataset
3 begin
4   do
5     for  $n = 1, \dots, N$ 
6        $y_m \leftarrow (\sum_{n=1}^N K(x_m, x_n)x_n) / (\sum_{n=1}^N K(x_m, x_n))$ 
7     end
8     for  $m = 1, \dots, N$ 
9        $x_m \leftarrow y_m$ 
10    end
11  while not converged
12  return  $x_1, \dots, x_N$ 
13 end
```

---

Edellä esitetyissä algoritmeissa ydin  $K$  on etäisyyden painotuksen funktio,  $X_i$  on vektori sisältäen pikselin koordinaatit ja väriarvot. Molemmat algoritmit MS ja BMS perustuvat samaan Mean Shift iteraatioon, mutta tuottavat erilaisen tuloksen. BMS on ilman rinnakaistamista laskennallisesti nopeampi, mutta toisaalta MS algoritmissa optimointiprosessit ovat toisistaan riippumattomia. Molempia algoritmeja voi nopeuttaa hyödyntämällä sopivia tietorakenteina, rajoittamalla Mean Shift askeleen laskennan naapurustoon, ja hyväksymällä konvergenssin tunnistamiseen pienen toleranssin. (Comaniciu et al., 2002)



Kuva 22: Mean Shift (MS) menetelmässä käytetty Gaussin ytimiä, spatiaalista sädettä  $\sigma_c = 5$  ja spektraalista sädettä  $\sigma_s = 15$ . Kuvassa näkyy menetelmän klusteroivaa vaikutusta.

### 4.3.5 Sobel suodatus

Tähän mennessä spatiaalisista suotimista on käsitelty kuvan silotukseen käytettäviä suotimia. Reunantunnistus on digitaalisen kuvankäsittelyn perusteita ja siihen on kehitetty useampia menetelmiä. Yhtenä näistä menetelmistä on Sobel suodatus, jossa gradientti on johdettu differenssistä. Differenssi on  $(f(x+h) - f(x))/h$  ja keskeisdifferenssi on  $(f(x+h) - f(x-h))/2h$ , josta Sobelin suodatuksen suodattimet ovat johdettu. Muita differenssiin perustuvia menetelmiä ovat Roberts'n risti tai Prewittin suodatin. Haralick:n gradientti ei perustu differenssiin vaan johdetaan naapurustoon sovitetun pinnan yhtälöistä. Sobelin suodatinta voi hyödyntää kuvan terävöityksessä, reunantunnistukseen, tai sellaisenaan gradienttina. (Bovik, 2009)

Yhtälössä 21 on esitetty Sobel suodatuksen gradientin laskentaan käytettävät suodattimet  $G_x$  ja  $G_y$ . Yhtälössä 22 on esitetty Sobel suodatuksen tulos eli gradientin voimakkuus, jota voi hyödyntää kuvan terävöittämiseen  $g = f + c|\nabla f|$ , jossa  $|\nabla f|$  on kuvan  $f$  gradientin voimakkuus. Kuvassa 23 on esitetty Sobel suodatuksella laskettu gradientin voimakkuus yhtälön 22 mukaisesti.

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad (21)$$

$$M = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2} = \sqrt{(f * G_x)^2 + (f * G_y)^2} \quad (22)$$

### 4.3.6 Laplace ja kuvan terävöitys

Laplacen operaattori on toisen asteen differentiaalioperaattori, joka määritellään  $xy$ -avaruudessa seuraavasti:

$$\Delta f = \nabla^2 f = \nabla \cdot \nabla f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \quad (23)$$



Kuva 23: Harmaasävykuvasta laskettu Sobelin gradientin voimakkuus. Gradientti on reunapiirteiden kohdalla suurempi kuin tasaisemmillä alueilla.

Laplacen operaattoriin perustuva reunantunnistus tuottaa ohuita rajoja eikä siten reunan ohennukselle ole tarvetta. Lisäksi operaattori on isotrooppinen eikä ole reunan suunnasta riippuvainen. Reunantunnistuksessa Laplacen operaattori on herkempi kohinan suhteen kuin ensimmäisen asteen derivaattaan perustuvat reunantunnistusmenetelmät (Bovik, 2009). Diskretisoituna Laplacen operaattori voidaan johtaa differenssistä  $f(x + 1) - f(x)$ , jolloin:

$$\Delta f(x, y) = f(x + 1, y) + f(x - 1, y) + f(x, y + 1) + f(x, y - 1) - 4f(x, y), \quad (24)$$

josta voidaan johtaa matriisimuotoinen suodatin:

$$L = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}. \quad (25)$$

Yleensä kuvassa reunat eroavat toisistaan terävyydeltään ja mittakaavoiltaan sumeista ja asteittaisista teräviin ja jyrkkiin. Tieto reunojen mittakaavasta helpottaa usein näiden tunnistamista. Suodatuksen herkkyys kohinan suhteen on johtanut menetelmään LoG (Laplacian of Gaussian), jossa Laplacen operaattori suoritetaan Gaussin silotuksen jälkeen. Yhtälössä 26 on esitetty Laplacen suodatin, joka on ns. 'sombbrero' tai 'mexican hat'. (Bovik, 2009)

$$LoG(x, y) = \nabla^2 G(x, y) = \frac{x^2 + y^2 - 2\sigma^2}{\sigma^4} e^{-\frac{x^2 + y^2}{2\sigma^2}} \quad (26)$$

LoG suodatinta voidaan approksimoida kahden Gaussin suodattimen erotuksella, jolloin menetelmää kutsutaan DoG:ksi (Difference of Gaussian). Tällöin voidaan hyödyntää myös separoitua konvoluutiota ja suorittaa konvoluutio käyttäen yksiulotteisia suodattimia. (Bovik, 2009)

$$DoG(x, y) = G_{\sigma_1}(x, y) - G_{\sigma_2}(x, y), \quad (27)$$

jossa approksimoidessa LoG suodatinta  $\frac{\sigma_2}{\sigma_1} \approx 1.6$  ja vastaavasti DoG:lla voidaan approksimoida verkkokalvon reunoja korostavaa vaikutusta, jolloin  $\frac{\sigma_2}{\sigma_1} \approx 5$  (Bovik, 2009). Kuva 24 on suodatettu LoG suodatusta approksimoivalla DoG suodatuksella.



Kuva 24: DoG-suodatettu kuva käyttäen LoG suodatusta approksimoivia parametreja  $\sigma_1 = 1.0$  ja  $\sigma_2 = 1.6$ . Kuvassa musta väri on  $-15$  ja valkoinen  $15$ , ja suodatuksella reuna on nollan leikkauskohdassa eli ns. 'zero crossing'.

Kuvan terävöityksellä pyritään korostamaan intensiteetin muutoksia. Kuvan keskiarvoistamisen eli sumentamisen analogia on integroiminen, niin silloin kuvan terävöittämisen analogia on luonnollisesti differentioiminen. Laplacen suodattimen isotrooppisuuden vuoksi se on käyttökelpoinen ja yksinkertaisin suodatin kuvan terävöittämiseen. Kuvankäsittelyssä ja segmentoinnissa kuvanparantamiseksi voi olla hyödyllistä suorittaa terävöityksen lisäksi Gamma-korjaus piirteiden korostamiseksi. Koska Laplacen suodinta voidaan approksimoi-

da Gaussin suodinten erotuksella, niin saadaan terävöitys esitettyä seuraavassa muodossa. (Gonzalez et al., 2006)

$$g = f + (f * DoG), \quad (28)$$

jossa  $g$  on terävöitetty kuvasta  $f$  käyttäen  $DoG$  suodinta. Kuvassa 25 on esitetty harmaasävykuva terävöitettynä.



Kuva 25: Kuvan terävöitys käyttäen DoG suodatusta parametreilla  $\sigma_1 = 1.0$  ja  $\sigma_2 = 1.6$ .

#### 4.3.7 Gradient Vector Flow

Gradient Vector Flow (GVF) on Xu et al., 1998, esittelemä kuvankäsittelymenetelmä Active Contour -menetelmän viivoihin vaikuttavaksi ulkoiseksi voimaksi, jolla tuotetaan kuvasta gradienttien vektorikenttä, joka kattaa laajemman alueen kuin paikallisen gradientin. GVF:n vektorikenttä  $\mathbf{v}(x, y) = [u(x, y) \ v(x, y)]$  minimoi seuraavasti määritellyn energiafunktionaalien

$$\varepsilon = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \mu(u_x^2 + u_y^2 + v_x^2 + v_y^2) + |\nabla f|^2 |\mathbf{v} - \nabla f|^2 dx dy. \quad (29)$$

Kun  $|\nabla f|$  on suuri jälkimmäinen termi dominoi funktionaalia ja energia minimoituu, kun  $\mathbf{v} = \nabla f$ . Parametri  $\mu$  reguloi ensimmäisen termin eli ympäristön gradienttien vaikutusta vektorikenttään, jolloin kohinaisella kuvalla suurempi  $\mu$  johtaa tasaisempaan vektorikenttään.

(Xu et al., 1998)

GVF voidaan johtaa ratkaisemalla seuraavat Eulerin yhtälöt

$$\begin{aligned}\mu \nabla^2 u - (u - f_x)(f_x^2 + f_y^2) &= 0 \\ \mu \nabla^2 v - (v - f_y)(f_x^2 + f_y^2) &= 0,\end{aligned}\tag{30}$$

joissa  $\nabla^2$  on Laplacen operaattori ja iteratiivinen numeerinen approksimaattoriratkaisu tälle yhtälöparille on esitetty seuraavilla yhtälöillä, joissa kuvan on oltava normalisoitu välille  $[0, 1]$  (Xu et al., 1998). Iteraatiota jatketaan, kunnes  $|u^{n+1} - u^n| + |v^{n+1} - v^n|$  on pienempi kuin haluttu tarkkuus.

$$\begin{aligned}u_{i,j}^{n+1} &= (1 - b_{i,j} \Delta t) u_{i,j}^n + r(u_{i+1,j}^n + u_{i,j+1}^n + u_{i,j}^n + u_{i-1,j}^n - 4u_{i,j}^n) + c_{i,j}^1 \Delta t \\ v_{i,j}^{n+1} &= (1 - b_{i,j} \Delta t) v_{i,j}^n + r(v_{i+1,j}^n + v_{i,j+1}^n + v_{i,j}^n + v_{i-1,j}^n - 4v_{i,j}^n) + c_{i,j}^2 \Delta t,\end{aligned}\tag{31}$$

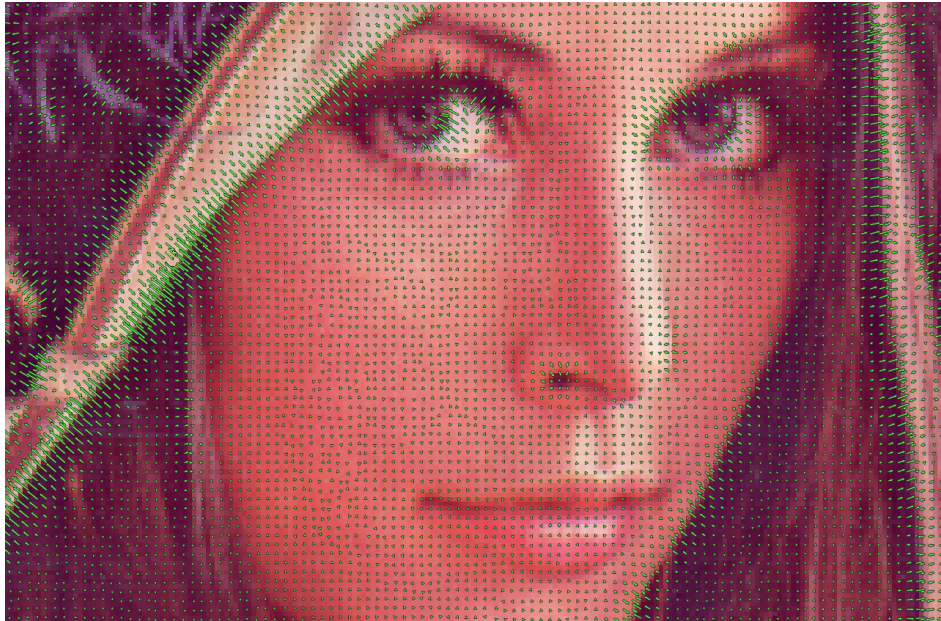
jossa

$$\begin{aligned}b_{i,j} &= ((f_{i+1,j} + f_{i-1,j})/2)^2 + ((f_{i,j+1} + f_{i,j-1})/2)^2 \\ c_{i,j}^1 &= b_{i,j}((f_{i+1,j} + f_{i-1,j})/2) \\ c_{i,j}^2 &= b_{i,j}((f_{i,j+1} + f_{i,j-1})/2) \\ r &= \frac{\mu \Delta t}{\Delta x \Delta y},\end{aligned}\tag{32}$$

jossa yleisesti  $\Delta x = \Delta y = 1$ ,  $\mu$  vakioparametri ja Courant–Friedrichs–Lewy askelpituuden rajoitus  $r \leq 1/4$ , jolloin  $\Delta t$  askel tulee rajoittaa konvergenssin takaamiseksi  $\Delta t \leq \frac{\Delta x \Delta y}{4\mu} = \frac{1}{4\mu}$  (Xu et al., 1998). Kuvassa 26 on menetelmällä tuotettu vektorikenttä.

## 4.4 Segmentointi

Segmentoinnilla tarkoitetaan kuvan jakamista merkityksellisiin ja yhteenkuuluviin osaluokkiin tai kohteisiin. Tilastollisessa data-analyysissä käytetty klusterointi on hyvin lähellä kuvankäsittelyn segmentointia. Segmentoinnin yksityiskohtaisuuden vaatimukset riippuvat käyttötarkoituksesta. Monimutkaisten kuvien segmentointi on yksi kuvankäsittelyn haastavimpia osaluokkia ja lopulta segmentoinnin tarkkuus ratkaisee usein koko analyysiprosessin onnistumisen tai epäonnistumisen. Siksi onkin tärkeää parantaa segmentoinnin onnistumisen todennäköisyyttä. (Gonzalez et al., 2006; Szeliski, 2011)



Kuva 26: GVF-menetelmällä laskettu vektorikenttä parametrilla  $\mu = 0.2$ .

Tietokonenäössä segmentointi on yksi vanhimpia ja laajimmin tutkittuja ongelmia. Aikaisimmat menetelmät perustuvat alueiden pilkkomiseen ja yhdistämiseen ja uudemmat algoritmit pyrkivät optimoimaan jonkin globaalin kriteerin kuten alueiden välisen yhtenevyyden tai alueiden välisten reunojen pituuden tai alueiden epäsamankaltaisuuden. (Szeliski, 2011)

Vaikka menetelmiä on kehitetty pitkään, niin silti ero on edelleen suuri verrattuna ihmisen kykyyn jakaa kuvia alueisiin (Szeliski, 2011). Tässä kappaleessa käsitellään tarkemmin työssä toteutettuja ja hyödynnettyjä segmentointimenetelmiä Watershed-muunnosta ja Active Countour -menetelmää. Lisäksi aikaisemmin esitettyä Mean Shift kuvankäsittelymenetelmää voidaan käyttää segmentoinnissa. Muita yleisesti käytettyjä segmentointi menetelmiä ovat esimerkiksi yksinkertaisimmillaan erilaiset kynnystys eli thresholding -menetelmät tai aluepohjaiset menetelmät. Kynnystystä voidaan suorittaa yhdellä raja-arvolla, tai adaptiivisella raja-arvolla tai moniarvoisina, ja aluepohjaisia menetelmiä ovat alueen kasvatus eli region growing, ja alueen jako ja yhdistys eli region splitting and merging (Szeliski, 2011; Gonzalez et al., 2006).



#### 4.4.1 Segmentoinnin määritelmä

Kuvankäsittelyssä naapurustolla tarkoitetaan pikselin  $(x, y)$  spatiaalista naapurustoa. Segmentoinnissa puhutaan usein 4- ja 8-yhdistetyistä (connected) naapureista. 4-yhdistetyssä naapurustossa pikselin  $(x, y)$  naapureita ovat siten  $(x-1, y)$ ,  $(x+1, y)$ ,  $(x, y-1)$  ja  $(x, y+1)$ . 8-yhdistetyssä naapurustossa on 4-yhdistetyn naapuruston naapureiden lisäksi kulmittaiset pikselit  $(x-1, y+1)$ ,  $(x+1, y-1)$ ,  $(x-1, y-1)$  ja  $(x+1, y+1)$ . (Gonzalez et al., 2006)

Gonzalez et al., 2006, esittää segmentoinnin seuraavasti.  $R$  on alue, joka kattaa koko kuva-alan.  $R$  on jaettu osa-alueisiin  $R_1, R_2, \dots, R_n$  siten, että

$$(a) \bigcup_{i=1}^n R_i = R$$

(b)  $R_i$  on yhdistetty joukko,  $i = 1, 2, \dots, n$ .

(c)  $R_i \cap R_j = \emptyset$  kaikilla  $i$  ja  $j$ , kun  $i \neq j$ .

(d)  $Q(R_i) = \text{TOSI}$ , kun  $i = 1, 2, \dots, n$ .

(e)  $Q(R_i \cup R_j) = \text{EPÄTOSI}$ , kun  $R_i$  ja  $R_j$  ovat vierekkäisiä.

Ehdon (a) mukaan segmentoinnin täytyy olla täydellinen eli kaikki pikselit kuuluvat johonkin alueeseen  $R_i$ . Ehto (b) tarkoittaa, että alueeseen kuuluvat pikselit ovat 4- tai 8-yhdistetty, ja ehto (c) vaatii kaikkien pikseleiden kuuluvan vain yhteen alueeseen. Ehtojen (d) ja (e)  $Q(R_k)$  on looginen predikaatti määriteltynä alueen  $R_k$  pisteille eli (d) tarkoittaa alueen pikseleiden toteuttavan predikaatin ja (e) tarkoittaa, ettei vierekkäisten alueiden pisteet toteuta keskenään predikaattia. Predikaatti voi olla esimerkiksi, että pikseleiden intensiteetit ovat yhtä suuret. (Gonzalez et al., 2006)

#### 4.4.2 Watershed-muunnos

Jos kuvaa visualisoidaan kolmiulotteisesti eli pikselin intensiteetillä tarkoitettaisiin korkeutta, niin Watershed-muunnoksella kuvattaisiin kuvassa olevia harjanteita eli vedenjaka-

ja. Watershed-muunnoksella etsitään ne rajat, joille pudotettu vesipisara ei valuisi johonkin yksittäiseen syvänteeseen vaan harjanteen laella se voisi valua kahteen tai useampaan syvänteeseen. Watershed-muunnoksen käyttö segmentoinnissa perustuu näiden vedenjakajina toimivien harjanteiden paikallistamiseen. Harjanteet voi paikantaa joko valuttamalla vettä ja etsimällä pikseleiden yhteinen valuma-alue tai minimi, tai täyttämällä valuma-alueita vedellä niiden minimeistä eli syvimmistä kohdista tasaisella nopeudella, kunnes alueet kohtaavat. (Gonzalez et al., 2006)

Roerdink et al., 2003, mukaan muunnos ei ole aina yksitulkintainen vaan tulos voi menetelmästä riippuen olla hieman erilainen. Eroa voi muodostua tasankoalueilla, joilla naapuruston intensiteetti on vakio. Etenkin jos harjanne muodostuu tällaiselle tasangolle, joka ei ole paikallinen minimi, niin harjanne ei ole välttämättä yksikäsitteinen. Täyttämiseen perustuvat menetelmät ovat myös herkkiä vuodoille topografisessa pinnassa. Menetelmiä Watershed-muunnokseen on esitetty useita, mutta tässä on kuvattu kaksi eri lähtökohdista muunnoksen ratkaisevaa menetelmää, joista Meyerin tulviminen on alhaalta ylöspäin, ja vesipisaramenetelmä on ylhäältä alaspäin (Roerdink et al., 2003). Nämä toteutettiin työssä, jotta voitiin vertailla eri Watershed-muunnoksia työhön parhaiten sopivan menetelmän valintaan.

### **Meyerin tulviminen (Meyer's flooding)**

Meyerin esittämässä menetelmässä Watershed-muunnos tehdään valitsemalla alkupisteet, joista aluetta aletaan laajentamaan eli menetelmällä on suora yhteys alueen kasvatukseen. Alkupisteiden valinnalla voidaan vähentää ylisegmentointia, kuvasta 27 näkyy tämä menetelmälle ominainen voimakas ylisegmentointi. Yksinkertaisimmillaan alkupisteiksi valitaan kuvasta ne pikselit, joiden arvo on pienempi tai yhtä suuri kuin naapuruston pikseleiden arvot. Ylisegmentoinnin välttämiseksi paikalliset minimi voidaan valita käyttäen kuvan homotooppista muunnosta ja morfologista gradienttia, jolloin ääriarvopisteitä, joissa gradientti on 0, tulee vähemmän tutkittavaksi. (Beucher et al., 1993; Roerdink et al., 2003)

Meyerin esittämä menetelmä hyödyntää tietorakenteina prioriteettijonoja valitsemaan seuraavaksi käsiteltävän pikselin. Prioriteettijonossa jonon kärjessä on aina korkeimman prioriteetin ehdokas yhdistettäväksi alueeseen. Korkein prioriteetti on tässä menetelmässä määritelty kyseisen ehdokkaan kohdalla olevan gradientin pituudeksi. (Beucher et al., 1993)



Kuva 27: Watershed-muunnoksella saadut reunaviivat. Kuvasta näkyy voimakasta ylisegmentointia ja prosessointialueiden välinen raja ennen alueiden uudelleennimiöimistä.

Algoritmissa 3 funktio *LocalMinimas* palauttaa listan kaikista paikallisista minimeistä mukaan lukien kuva-alan reunoilla olevat, ja *Neighbors* palauttaa listan  $(x, y)$  pisteistä, jotka ovat kuva-alueella ja ovat parametrien 4- tai 8-yhdistetyssä naapurustossa.

### Algoritmi 3: Meyer's flooding

---

```

1  input: image  $f$ 
2  output: image
3  begin
4    priority_queue  $seeds$ 
5    image  $\phi$  size of  $f$  reset to 0
6    int  $label \leftarrow 1$ 
7    for  $(x, y) \in LocalMinimas(f)$ 
8      push  $(x, y, label)$  into  $seeds$  with priority  $\nabla f(x, y)$ 
9       $label \leftarrow label + 1$ 
10   end
11   while  $seeds$  is not empty
12      $(x, y, label) \leftarrow pop(seeds)$ 
13     if  $\phi(x, y) = 0$ 
14        $\phi(x, y) \leftarrow label$ 
15       for  $(x_n, y_n) \in Neighbors(x, y)$ 
16         push  $(x_n, y_n, label)$  into  $seeds$  with priority  $\nabla f(x_n, y_n)$ 
17       end
18     end
19   end
20   return  $\phi$ 
21 end

```

---

## Vesipisaramenetelmä Watershed Cuts

Cousty et al., 2009, esittävät Watershed-muunnoksen pienimmän virittävän metsän eli minimal spanning forest (MSF) avulla, jolla tarkoitetaan pienimpien virittävien puiden eli minimal spanning tree (MST) joukkoa. Pienimmät virittävät puut etsitään kuvasta muodostetusta graafista vesipisaran virtaaman mukaan käyttäen graafissa leveyshakua tasangoilla ja syvyys-hakua muualla, kunnes löytävät paikallisen minimin tai yhdistyvät jo toiseen puuhun. Menetelmä ei tarvitse alkupisteitä vaan iteratiivisesti yritetään lähteä etsimään valuma-alueita kaikista pikseleistä. Koska osa pisteistä on jo käsitelty valumien tarkastelussa eikä niitä tarvitse käsitellä enää uudelleen, päästään menetelmällä lineaariseen kompleksisuuteen. (Cousty et al., 2009)

Tässä algoritmi 4 on esitetty ilman graafia toisin kuin Cousty et al., 2009. Graafina  $G = (V, E)$  esitettyinä kuva esitettäisiin solmujen  $V$  ja kaarien  $E$  joukkoina. Solmut vastaavat tällöin pikseleitä ja kaaret yhteyksiksi kahden vierekkäisen pikselin välillä. Tässä esitetyssä muodossa menetelmän käyttämä funktio *Pixels* palauttaa listan kaikkien pikseleiden sijainneista ja algoritmin 5 mukaisesti määriteltyä *Stream* funktiota.

---

### Algoritmi 4: Watershed Cuts

---

```
1  input : image  $f$ 
2  output : image
3  begin
4    image  $\phi$  size of  $f$  reset to 0
5    int  $label \leftarrow 1$ 
6    for  $(x, y) \in Pixels(f)$ 
7      if  $\phi(x, y) = 0$ 
8         $(s, label_s) \leftarrow Stream(f, \phi, (x, y))$ 
9        if  $label_s = 0$  /* stream  $s$  ends to minima */
10         for  $(x_s, y_s) \in s$ 
11            $\phi(x_s, y_s) \leftarrow label$ 
12            $label \leftarrow label + 1$ 
13         end
14       else
15         for  $(x_s, y_s) \in s$ 
16            $\phi(x_s, y_s) \leftarrow label_s$ 
17         end
18       end
19     end
20   end
21   return  $\phi$ 
22 end
```

---

Algoritmissa 5 esitetty *Stream* funktio käyttää tietorakenteina joukkoja, *Neighbors* funktiota kuten Meyerin tulviminen, ja *MinNeighborValue* funktiota, joka palauttaa pienimmän arvon pisteen naapurustossa, jolloin virtaama etenee jyrkimpään suuntaan ja vastaa Meyerin menetelmän suurimman gradientin suuntaan alueen laajentamista.

---

#### Algoritmi 5: Stream

---

```

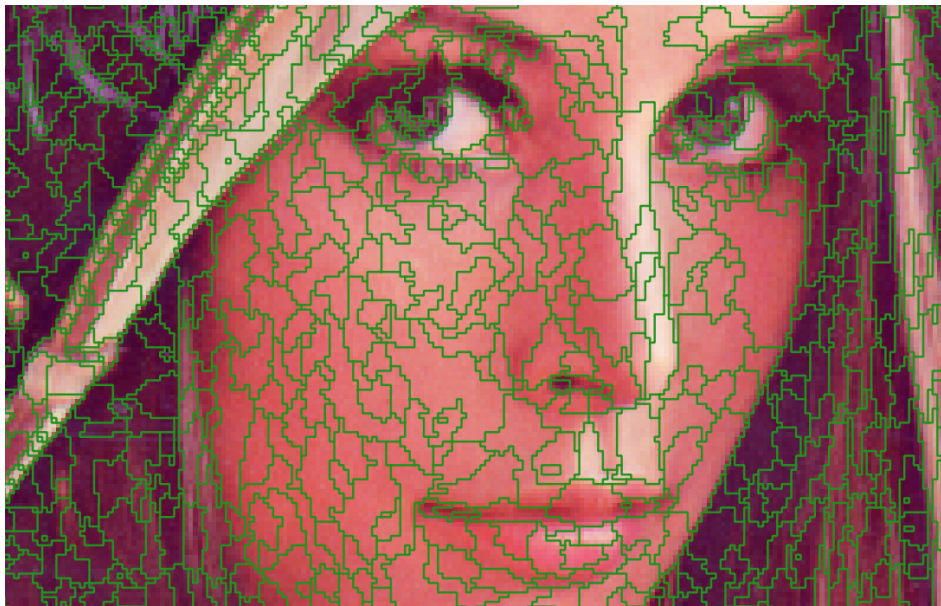
1  input: image  $f$ 
2         image  $\phi$ 
3         point  $(x, y)$ 
4  output: set of points
5         int
6  begin
7    set of points  $s \leftarrow \{(x, y)\}$ 
8    set of points  $s' \leftarrow \{(x, y)\}$  /* set of unexplored bottoms */
9    int  $label \leftarrow 0$ 
10   while  $s'$  is not empty
11      $(x_y, y_y) \leftarrow \text{pop}(s')$ 
12     bool  $breath\_first \leftarrow TRUE$ 
13     while  $breath\_first$ 
14       for  $(x_z, y_z) \in \text{Neighbors}(x_y, y_y)$ 
15         if  $f(x_z, y_z) > \text{MinNeighborValue}(f, (x_y, y_y))$  or  $(x_z, y_z) \in s$ 
16           continue
17         if  $\phi(x_z, y_z) \neq 0$  /* stream joins existing stream */
18           return  $(s, \phi(x_z, y_z))$ 
19         else if  $f(x_z, y_z) < f(x_y, y_y)$  /* this is now the only bottom */
20           push  $(x_z, y_z)$  into  $s$ 
21            $s' \leftarrow \{(x_z, y_z)\}$ 
22            $breath\_first \leftarrow FALSE$ 
23           break
24         else /* continue breath first within plateau */
25           push  $(x_z, y_z)$  into  $s$ 
26           push  $(x_z, y_z)$  into  $s'$ 
27         end
28     end
29   end
30 end
31 return  $(s, 0)$  /* new stream */
32 end

```

---

#### 4.4.3 Dynamiikkasuodatus ja Watershed

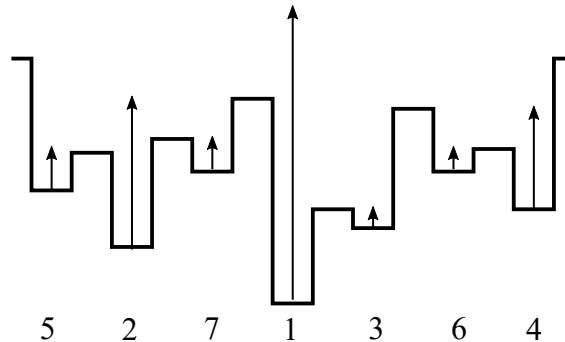
Ylisegmentoinnin välttämiseksi voidaan paikallisia minimeitä suodattaa käyttäen dynamiikkasuodatusta, jolloin ne minimiit, joilla dynamiikka on liian pieni, jätetään pois alkupisteistä (Cousty et al., 2009). Suodatuksessa käytetään Bertrand, 2007, kehittämää dynamiikan mitata, joka kuvaa Watershed-muunnoksen yhteydessä pienintä korkeutta, joka on ylitettävä, päästäkseen jostain paikallisesta minimistä mihin tahansa itseään matalampaan paikalliseen minimiin. Täydellistä Watershed-muunnoksella johdettua segmentointia voidaan ajatella graafina, jossa graafin solmut ovat kaikki paikalliset minimiit, ja kaaret ovat niiden solmujen välillä, jotka jakavat segmentoinnissa yhteisen rajan. Kaarien arvo on se pienin arvo, jolla kyseisen rajan voi ylittää. Kuvassa 28 on esitetty dynamiikalla suodatettu Watershed-muunnos.



Kuva 28: Dynamiikalla suodatettu kuvan 27 Watershed-muunnos. Kuvan dynamiikkasuodatuksessa minimien dynamiikka on vähintään 20.

Dynamiikka lasketaan graafin solmulle siten, että solmut järjestetään suuruusjärjestykseen. Pienimmän solmun arvon dynamiikaksi asetetaan  $\infty$ . Sitten lasketaan kaikille muille solmuille dynamiikka etsimällä se polku graafissa kaikkiin pienemmän arvon solmuihin, joilla polun kaarien arvo on suurimmillaan pienin. Solmun dynamiikka saa arvokseen polun kaarien suurimman arvon ja solmun arvon tämän suurimman arvon ja solmun arvon erotuksen. Kuvassa 29 on yksiulotteisessa tapauksessa esitetty dynamiikan laskenta. (Bertrand, 2007)

Laskennassa voidaan käyttää jotain minimax -polun ongelman ratkaisumenetelmää, joka on käänteinen leveimmän polun ongelmalle. Ongelma voidaan ratkaista pienimmän virittävän puun kautta, ja siitä muodostettua karteesista puuta käyttäen, josta pienin suurin arvo polulla saadaan ratkaistua lineaarisessa ajassa.



Kuva 29: Dynamiikka yksiulotteisessa tapauksessa. Minimit ovat järjestetty suuruusjärjestykseen pienimmästä suurimpaan. Nuolet kuvaavat kunkin paikallisen minimin dynamiikkaa. Pienimmän dynamiikka on  $\infty$ . Esimerkiksi 2. pienimmän minimin dynamiikka on korkein este pienimpään minimiin, vaikka välissä on 7. pienin minimi.

#### 4.4.4 Active Contour -menetelmä

Ensimmäinen Active Contour -menetelmä oli nimetty kehittäjiensä mukaan käärmeiksi (snakes), jossa parametrinenä käyränä oli paloittain polynomialinen käyrä eli splini, jonka energiaa minimoitiin. Energiassa on sisäinen ja ulkoinen komponentti. Sisäiseen energiaan vaikuttaa käyrän jäykkyys ja elastisuus, ja ulkoiseen energiaan vaikuttaa kuva tai kuvasta johdettu arvo. Kaikille Active Contour -menetelmille on kuitenkin yhteistä, että jollain keinolla parametroitua käyrää optimoidaan iteratiivisesti kohti tilaa, jossa käyrän energia minimoituu. Menetelmää voidaan käyttää alueen reunojen etsimiseen tai kuvasekvenssissä kohteen seurantaan. (Szeliski, 2011)

Parametroitu käyrä esitetään yleistettynä seuraavasti:

$$C(s) = (x(s), y(s)), \quad (33)$$

jossa  $s = [0, 1]$ .  $x(s)$  ja  $y(s)$  ovat kuvataason koordinaatit. Kokonaisenergia on tällöin

$$E(C(s)) = \int_0^1 (E_{int}(C(s)) + E_{ext}(C(s))) ds, \quad (34)$$

jossa käyrän sisäinen ja ulkoinen energia ovat

$$E_{int}(C(s)) = \frac{1}{2}(\alpha|C_s(s)|^2 + \beta|C_{ss}(s)|^2) \quad (35)$$

$$E_{ext}(C(s)) = -|\nabla f(C(s))|^2, \quad (36)$$

jossa  $\alpha$  on elastisuuden ja  $\beta$  on jäykkyyden painokerroin. Notaatio  $C_s(s)$  tarkoittaa käyrän  $C$  derivaattaa kohdassa  $s$  muuttujan  $s$  suhteen ja  $C_{ss}(s)$  on toinen derivaatta ja niin edelleen. Jos etsitään reunoja, niin ulkoinen energia on kuvan  $f$  gradientista riippuvainen, jolloin energia minimoituu, kun käyrään vaikuttavat voimat ovat tasapainossa. Tämä voidaan esittää seuraavasti:

$$F_{int}(C(s)) + F_{ext}(C(s)) = \alpha C_{ss}(s) + \beta C_{ssss}(s) - \nabla E_{ext}(C(s)) = 0. \quad (37)$$

Perinteisesti ulkoisena voimana käytetyn kuvan gradientin ongelmana on ollut koverien alueiden tunnistaminen ja siihen on esitetty ratkaisuksi käyttää ulkoisena voimana GVF-menetelmällä johdettua gradienttien vektorikenttää eli  $F_{ext}(C(s)) = GV F(C(s))$ , jossa diffuusioprosessin vuoksi kauempana olevilla gradienteilla on vaikutusta ja siten myös koverat muodot löytyvät paremmin. (Xu et al., 1998)

Alkuperäisen Active Contour -menetelmän käyrät etenevät toisistaan riippumattomina. Tähän on esitetty ratkaisu siten, että asetetaan käyrän päätepisteille vaatimus, että päätepisteessä minimoidaan kokonaisenergia kaikkien siihen pisteeseen yhdistyvien käyrien suhteen. Tällöin muodostuu parametroitujen käyrien verkko, joka säilyttää topologisen eheyden. Tätä on esitetty käytettäväksi esimerkiksi katuverkon geometrioita sovitettaessa ilma-kuviin. (Butenuth et al., 2012)

Käyrien energioiden minimointi voidaan suorittaa yksinkertaisimmillaan iteroimalla voiman suuntaan, interpoloimalla parametroidusta käyrästä uudet käyrää kuvaavat pisteet jokaisella



iteraatiolla ja hyödyntämällä differenssimenetelmää käyrän pisteisiin vaikuttavien voimien laskemiseen. Gradienttimenetelmällä minimointi on herkkä oskillaatiolle ja siten askelpituus  $\delta t$  tulee valita riittävän pieneksi. Oskillaatiota voidaan vähentää pehmentämällä käyriä diffuusioyhtälöllä käyttäen Gaussin silotusta, jolloin voidaan myös johtaa suurin askelpituus kullakin ajanhetkellä, jolla oskillaatio suppenee. Konjugaattigradienttimenetelmällä haasteena on kuvasta riippuen mahdollisuus siihen, että käyrä hyppää jonkin kuvan tärkeän piirteen ylitse. (Ivins et al., 1993)

### Algoritmi 6: Network Active Contour

---

```

1  input : image  $f$ 
2          float  $h, \alpha, \beta, \delta t$ 
3          set of curves  $C$ 
4  output : set of curves
5  begin
6    image  $gvf \leftarrow GVF(f)$ 
7    do
8      set of curves  $C' \leftarrow \emptyset$ 
9      for  $c \in C$ 
10        $c \leftarrow Interpolate(c, h)$  /* resample curves  $s = [0, 1]$  */
11     end
12     for  $c \in C$ 
13       for  $s$  in  $c$  /* apply forces to all points of a curve */
14          $c'(s) \leftarrow c(s) + \delta t(\alpha C_{ssss}(c(s)) + \beta C_{ss}(c(s)) - gvf(c(s)))$ 
15         if  $s = 0$  or  $s = 1$  /* apply forces from neighbors */
16           for  $[c_2, s_2] \in NeighborCurves(C, c, s)$ 
17              $c'(s) \leftarrow c'(s) + \delta t(\alpha C_{ssss}(c_2(s_2)) + \beta C_{ss}(c_2(s_2)))$ 
18           end
19         end
20         push  $c'$  into  $C'$ 
21       end
22     end
23      $C \leftarrow C'$ 
24   while not converged
25   for  $c \in C$ 
26      $c \leftarrow Interpolate(c, h)$  /* resample curves  $s = [0, 1]$  */
27   end
28   return  $C$ 
29 end

```

---

Kuvassa 30 on esitetty käyrän siirtyminen tasapainotilaan algoritmilla 6, joka on johdettu yhdistämällä yksinkertainen gradienttimenetelmään perustuva minimointi Xu et al., 1998, esittämään GVF-menetelmään ja Butenuth et al., 2012, esittämään sisäiseen voimaan

käyrien päätepisteissä käyttäen funktiota *NeighborCurves*, joka palauttaa ne käyrät ja niiden päätepisteet, jotka ovat yhteiset käyrän pisteen  $c(s)$  kanssa. Differenssimenetelmällä toisen asteen derivaatan approksimaatio on  $C_{ss}(s) = C(s-h) - 2C(s) + C(s+h)$  ja neljännen asteen derivaatta on  $C_{ssss}(s) = C(s-2h) - 4C(s-h) + 6C(s) - 4C(s+h) + C(s+2h)$ . Päätepisteissä  $C_{ss}(0) = C_{ss}(1) = 0$ ,  $C_{ssss}(0) = C(0) - 2C(h) + C(2h)$  ja  $C_{ssss}(1) = C(1) - 2C(1-h) + C(1-2h)$ .

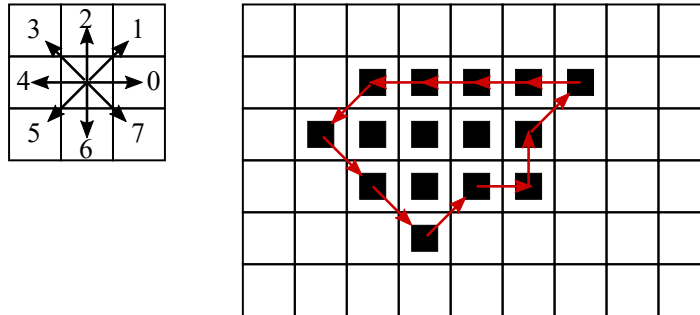


Kuva 30: Parametroidun käyrän siirtyminen Active Contour -menetelmällä. Lähtötilanne punaisella, vihreä on 19 iteraation jälkeen,  $\alpha = 0.3$ , ja  $\beta = 0.2$ .

## 4.5 Vektoriesitys ja kuvan vektorisointi

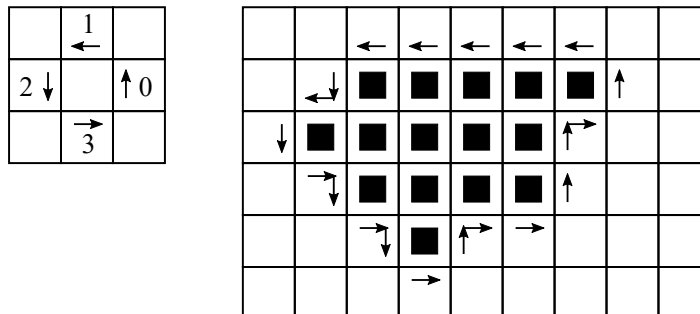
Segmentoinnin tuloksena on kuva, jossa kukin pikseli on nimetty sen segmentin mukaan, mihin se kuuluu. Tällaisen ja muidenkin kuvien vektorisointiin voidaan käyttää jotain reunaviivan seuranta -algoritmia. Segmentoinnin tuloksena tuotettuja alueita voidaan kuvata niiden sisäisten ja ulkoisten piirteiden avulla. Ulkoiset piirteet liittyvät alueen reunaviivan ja sen ominaisuuksiin. Reunaviiva voidaan esittää eri tavoilla ja ketjukoodit ovat yksi tapa esittää reunaviivaa. Reunaviivan seuranta -algoritmeissa etsitään kuvasta alkukohta, josta aletaan seurata reunaviivaa myötä- tai vastapäivään. Reunaviivaa voidaan seurata 4- vai 8-yhdistetyssä naapurustoissa. Ketjukoodiesitys reunaviivalle on numerosarja, jossa kukin numero kuvaa reunan suuntaa edellisestä pisteestä. 4-suuntaisessa ketjukoodissa käytetään

numeroita 0 (oikea), 1 (ylös), 2 (vasen) ja 4 (alas), kun 8-suuntaisessa esityksessä on mukana vastaavasti kulmittaiset suunnat, jolloin koodit ovat 0 (oikea), 1 (yläoikea), 2 (ylös), jne. Kuvassa 31 on esitetty 8-suuntainen ketjukoodi. (Gonzalez et al., 2006)



Kuva 31: Ketjukoodi esitetään suuntina reunapikselistä seuraavaan reunapikseliin. Kuvan ketjukoodi alkupisteestä (3, 2) on 5,7,7,1,0,2,1,4,4,4,4.

Reunaviiva voidaan esittää olevan myös pikselien väleissä, jolloin 4-suuntaisena puhutaan ns. 'crack code' -esityksestä. Tämän esityksen 8-suuntaista variaatiota kutsutaan ns. 'mid crack code' -esitykseksi. Kuvassa 32 on esitetty ketjukoodia vastaava 'crack code'. (Shih, 2010)



Kuva 32: Reunaviiva 'crack code' -esityksenä esitetään siirtyminä reunapikselien väleissä. Kuvan 'crack code' -esitys alkupisteestä (3, 2) on 2,1,2,3,2,3,2,3,0,3,3,0,0,3,0,1,1,1,1,1.

Reunaviiva esitettynä ketjukoodi tai 'crack code' -esityksenä voidaan muuntaa vektoriesitykseksi laskemalla koordinaattien siirtymät esityksestä tuntien ketjukoodin alkupisteen. Vektoriesityksestä voidaan laskea ulkoisia piirteitä kuten yksinkertaisimmillaan alueen kompaktius, joka on reunanviivan pituuden neliö jaettuna alueen pinta-alalla, jolloin vastaavasti alueen suhteellinen pyöreys voidaan esittää:

$$R_c = \frac{4\pi A}{P^2}, \tag{38}$$

jossa  $A$  on alueen pinta-ala ja  $P$  on reunaviivan pituus. Suhteellinen pyöreys  $R_c$  on täydellisellä ympyrällä 1. (Gonzalez et al., 2006)

Vektorimuotoinen geometria on usein segmentoinnin tuloksena yksityiskohtainen ja sitä voi olla tarpeen yksinkertaistaa. Yksinkertaistukseen käytetään usein paikkatietokontekstissa Douglas-Peucker -menetelmää, joka poistaa vektorimuotoisesta geometriasta taitepisteitä, joiden poistaminen vaikuttaa vähemmän alkuperäiseen geometriaan kuin hyväksyty virhetoleranssi. Tämä menetelmä on rekursiivinen, jossa muodostetaan kahden taitepisteen välille jana ja etsitään näiden pisteiden väliltä kauimmainen taitepiste, jos kauimmainen taitepiste on virhetoleranssia lähempänä niin kaikki pisteet poistetaan käsiteltävien taitepisteiden väliltä, jos se on kauempana, niin käsitellään samoin alkuperäisten taitepisteiden ja tämän kauimmaisen pisteen välit. Kuvassa 33 on esitetty alla esitetyn Douglas-Peucker -algoritmin tuottama geometrian yksinkertaistus. (Douglas et al., 2011)

#### Algoritmi 7: Douglas-Peucker

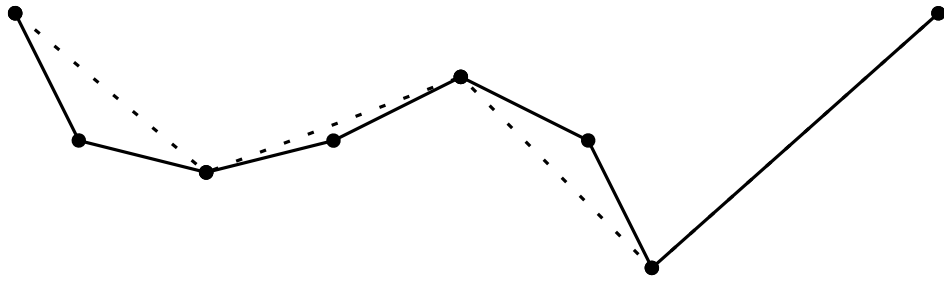
---

```

1  input: list of points  $P$ 
2          float  $\epsilon$ 
3  output: list of points
4  begin
5       $d_{max} \leftarrow 0$ 
6       $index \leftarrow 0$ 
7       $len \leftarrow Length(P)$ 
8      for  $i = 1, \dots, len - 2$ 
9           $d \leftarrow Distance(P[0], P[len - 1], P[i])$ 
10         if  $d_{max} < d$ 
11              $d_{max} \leftarrow d$ 
12              $index \leftarrow i$ 
13         end
14     end
15     if  $d_{max} > \epsilon$ 
16          $left \leftarrow DouglasPeucker([P[0]..P[index]], \epsilon)$ 
17          $right \leftarrow DouglasPeucker([P[index]..P[len - 1]], \epsilon)$ 
18         return  $[RemoveLastPoint(left), right]$ 
19     else
20         return  $[P[0], P[len - 1]]$ 
21     end
22 end

```

---



Kuva 33: Viivan yksinkertaistus Douglas–Peucker menetelmällä. 1 metrin toleranssilla, kun viivan päätepisteiden etäisyys on 14 metriä, alkuperäisestä viivasta poistetaan taitepisteet 2, 4 ja 6.

## 5 AINEISTOT JA METSIKKÖKUVIDEN TUOTTAMINEN

Työn tavoitteena oli kehittää menetelmä metsikkökuvioiden tuottamiseksi kaukokartoitusaineistosta ja tässä kappaleessa kuvataan kehitetyt menetelmät kuvioiden tuottamiseen ja kuviorajojen tarkentamiseen. Näiden kuvioiden tuottamiseen tarvittavat menetelmät ovat käsitelty edeltävissä luvuissa, ja tässä esitetään, miten näitä menetelmiä on hyödynnetty työn tavoitteen saavuttamiseen. Työssä toteutettiin esitettyjen kuvankäsittelymenetelmien lisäksi kaukokartoitusaineiston käsittelyssä tarvittavia menetelmiä.

Työssä käytettävää aineistoa kerätään yksityisestikin metsävaratiedon inventointihankkeissa, mutta vastaavaa aineistoa on nykyään saatavilla valtakunnallisesti Maanmittauslaitoksen julkaisemana avoimena aineistona, tällöin aineiston ajantasaisuus ja tarkkuus riippuvat Maanmittauslaitoksesta, jonka seuraava kartoitusohjelma valtakunnalliseen laserkeilaukseen ja ilmakuvaukseen on tarkoitus suorittaa 2020 ja 2025 välillä. Työssä toteutetut menetelmät ja työkalut eivät kuitenkaan ole rajoittuneet Maanmittauslaitoksen avoimiin aineistoihin.

Käytetyt menetelmät ovat toteutettu käyttäen avoimen lähdekoodin kirjastoja, joita Arbonaut Oy:n ArboSDK -ympäristö voi käyttää. ArboSDK abstraktoi niin rasteri-, kuin vektoriesitykset siten, että se voi hyödyntää näiden avoimen lähdekoodin kirjastojen lisäksi kaupallisia Environmental Systems Research Institute:n eli ESRI:n ArcObjects -kirjastoja. ESRI on pitkään ollut hallitseva toimija paikkatietoteknologioissa ja joidenkin yleisesti käytettyjen paikkatietoformaattien hyödyntäminen voi vaatia näiden kirjastojen käyttämistä.

### 5.1 Maanmittauslaitoksen avoimet aineistot

Maanmittauslaitoksen avoin paikkatietoaineisto on julkaistu Creative Commons Nimeä 4.0 Kansainvälinen -lisenssillä, joka sallii vapaasti jakaa ja muunnella aineistoa, kunhan lähde nimetään ja mahdolliset muutokset aineistoon ovat merkittynä. Tässä työssä käytettyjä Maanmittauslaitoksen avoimia kaukokartoitustuotteita ovat kiinteistörekisterikartta, laser-

keilausaineisto ja Maanmittauslaitoksen ortokuva.

Kiinteistörekisterikartta-aineistossa on kiinteistöjen tunnukset ja maantieteelliset rajat. Työssä esitettävää menetelmää käytetään jollain maantieteellisellä alueella rajattuna usein joidenkin tiettyjen kiinteistöjen alueille. Taulukossa 1 on Maanmittauslaitoksen Kiinteistörekisterikartan tuoteominaisuudet.

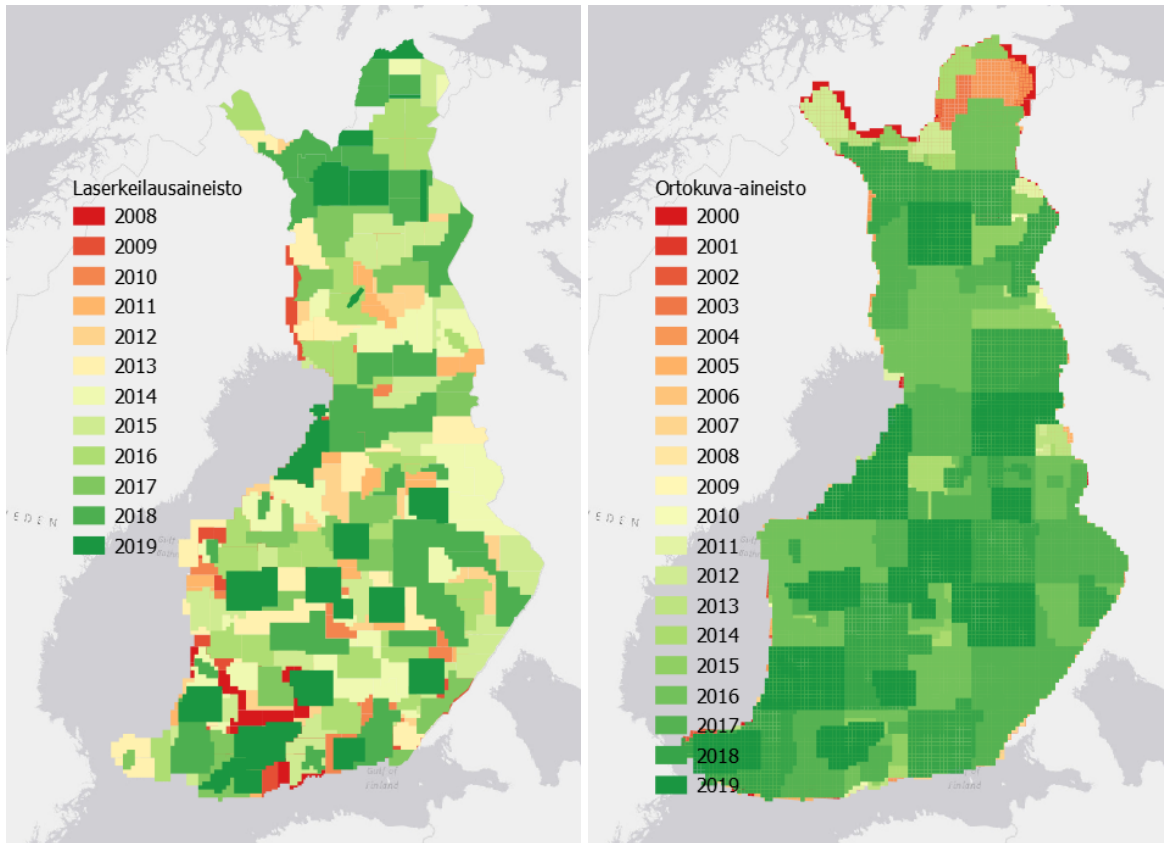
Taulukko 1: Kiinteistörekisterikartta

Kuvaus	Kiinteistöjen ja muiden rekisteriyksiköiden rajat, rajamerkit ja kiinteistötunnukset.
Maantieteellinen kattavuus	Koko Suomi
Esitystapa	Vektori
Tarkkuus	2005 jälkeen haja-asutusalueilla tarkkuus 0.2 m ja aiemmassa aineistossa 0.5-4 m.
Aineistoformaatti	GML, ESRI ShapeFile, MapInfo mif
Ylläpidon tiheys	Jatkuva
Lisätietoja	<a href="https://www.maanmittauslaitos.fi/kartat-ja-paikkatieto/asiantuntevalle-kayttajalle/tuotekuvaukset/kiinteistorekisterikartta">https://www.maanmittauslaitos.fi/kartat-ja-paikkatieto/asiantuntevalle-kayttajalle/tuotekuvaukset/kiinteistorekisterikartta</a>

Laserkeilaus- tai ilmakeilaus-aineistoa voidaan joko mitata/kuvata itse tai hyödyntää avoimia aineistoja. Maanmittauslaitoksen avoimet laserkeilaus- ja ortoilmakuvatuotteet ovat kuvattuna taulukoissa 2 ja 3. Kuva 34 esittää näiden tuotteiden maantieteellisestä kattavuutta ja alueellisesti tuoreinta saatavissa olevaa aineistoa.

Mikäli ei ole käytettävissä valmiiksi orto-oikaistua ilmakeilaus-aineistoa, niin aineistolle täytyy suorittaa orto-oikaisu, jotta se olisi käyttökelpoista metsikkökuvien tuottamiseen. Käsiteltävän alueen rajoittuessa useamman ilmakeilan alueelle, niin myös valmiin ortoilmakuva-aineiston radiometriikkaa voidaan joutua korjaamaan siten, että kuvamosaiikin kuvien välisille reunoille ei jää visuaalisia rajoja.

Vastaavasti käytettävissä olevassa laserkeilausaineistossa korkeudet eivät välttämättä ole valmiiksi muunnettuja korkeuksiksi maanpinnasta vaan ovat korkeuksia koordinaatiston referenssiellipsoidiin. Tällöin laserkeilausaineiston korkeudet pitää muuntaa korkeuksiksi maanpinnasta joko käyttäen jotain olemassa olevaa maastomallia tai implisiittisesti johtamal-



Kuva 34: Maanmittauslaitoksen avoimen laserkeilaus- ja ortoilma kuva-aineistojen maantieteellinen kattavuus ja aineiston mittaus- tai kuvausvuosi.

Taulukko 2: Laserkeilausaineisto

Kuvaus	Laserkeilausaineisto on kolmiulotteinen pistemäinen aineisto, joka kuvaa maanpintaa tai maanpinnalla olevia kohteita. Aineistoa hyödynnetään muun muassa korkeusmallien valmistamiseen ja metsävaratiedon tuottamiseen.
Maantieteellinen kattavuus	Koko Suomi
Esitystapa	Vektori
Aineistoformaatti	LAS
Tarkkuus	0.5 pistettä/m <sup>2</sup>
Ylläpidon tiheys	Tavoitteena koko maan kattavuus 2020 mennessä. Päivitetään Korkeusmalli 2 m -tuotteen yhteydessä eli tarvittaessa.
Lisätietoja	<a href="https://www.maanmittauslaitos.fi/kartat-ja-paikkatieto/asiantunnevalle-kayttajalle/tuotekuvaukset/laserkeilausaineisto">https://www.maanmittauslaitos.fi/kartat-ja-paikkatieto/asiantunnevalle-kayttajalle/tuotekuvaukset/laserkeilausaineisto</a>



Taulukko 3: Maanmittauslaitoksen ortokuva

Kuvaus	Ortokuva-aineisto on orto-oikaistua ilmakehän kuva-aineistoa, jota hyödynnetään kartoituksessa, ympäristön suunnittelussa ja seurannassa. Aineistoa on saatavilla mustavalkoisena, värillisenä tai lähi-infrapunakanavan sisältävinä väärävärivärikuvina.
Maantieteellinen kattavuus	Koko Suomi
Esitystapa	Rasteri 12000x12000 pikseliä
Tarkkuus	0.25 m tai 0.5 m pikselikoko.
Aineistoformaatti	GeoTIFF, ECW (JPEG2000)
Ylläpidon tiheys	Digi 0.5 m väri ja väri-infra tuotteilla alueesta riippuen 3-10 vuotta. Ilmakehän kuvaus toteutetaan kansallisen ilmakehän kuvausohjelman mukaisesti.
Lisätietoja	<a href="https://www.maanmittauslaitos.fi/kartat-ja-paikkatieto/asiantuntevalle-kayttajalle/tuotekuvaukset/ortokuva">https://www.maanmittauslaitos.fi/kartat-ja-paikkatieto/asiantuntevalle-kayttajalle/tuotekuvaukset/ortokuva</a>

la maastomalli laserkeilausaineistosta ja vähentämällä maastomallin mukainen paikallinen maanpinnan korkeus pistepilven pisteiden korkeuksista.

## 5.2 Sovellus- ja kehitysympäristö

Työssä toteutetut aineiston- ja kuvankäsittelymenetelmät toteutettiin käyttäen C++ -ohjelmointikieltä ja kehitysympäristönä käytettiin Microsoft Visual Studio:ta. Toteutettu ohjelmakoodi on kuitenkin käyttöjärjestelmäriippumaton C++14 -standardin mukaista ohjelmakoodia ja käytetyt ulkoiset kirjastotkin ovat saatavilla useille eri käyttöjärjestelmille esimerkiksi Linux:lle. C++14 ohjelmakoodi mahdollistaa funktioiden paluuarvon tyyppin automaattisen päättelyn vähentäen tarvetta templaateille ja vastaavasti C++11:n mukana tulleet lambda lausekkeet hyödyntävät C++14:ssa tietotyyppien automaattista päättelyä.

Käytetyt ulkoiset kirjastot ovat listattuna taulukossa 4. Projektitiedostojen ollessa kuitenkin Visual Studio -projekteja, niin ohjelmakoodin kääntäminen jollekin toiselle käyttöjärjestelmälle vaatisi projektitiedostojen kääntämistä esimerkiksi CMake-tiedostoiksi, joka voitaisiin toteuttaa jollain olemassa olevalla konvertterilla. Paikkatietoaineistojen

esittämisessä on tässä työssä käytetty QGIS-paikkatietosovellusta (entinen Quantum GIS), joka on avoimen lähdekoodin vastine kaupalliselle ESRI ArcGIS Desktop -sovellukselle. Paikkatietosovelluksessa projektiin voidaan lisätä erilaisia paikkatietoaineistoja, visualisoida, käsitellä ja julkaista niitä.

Taulukko 4: Sovellusympäristön ulkoiset kirjastot

GDAL/OGR	GDAL on kirjasto paikkatieto rasterikuvien ja vektoriaineistojen käsittelemiseen esittäen eri rasterikuvaformaattit omassa abstraktissa tietomallissa ja eri vektoriformaatit omassa abstraktissa tietomallissa, josta vastaa OGR -kirjasto osana GDAL -kokonaisuutta. Kirjasto on julkaistu Open Source Geospatial Foundation:n X/MIT tyyllisellä avoimen lähdekoodin lisenssillä. Kirjasto on aktiivisessa kehityksessä ja viimeisin suurempi kirjaston versiopäivitys on 3.0 vuodelta 2019. Lisätietoja: <a href="https://gdal.org/">https://gdal.org/</a>
PDAL	PDAL on kirjasto pistepilven käsittelyyn, lukemiseen ja kirjoittamiseen eri formaateissa. Kirjasto on julkaistu BSD avoimen lähdekoodin lisenssillä ja viimeisin suurempi kirjaston versiopäivitys on 2.0 vuodelta 2019. Lisätietoja: <a href="https://pdal.io/">https://pdal.io/</a>
Boost	Boost kirjastot ovat tarkoitettu mahdollisimman yleiskäyttöisiksi ja siirrettäviksi moniin eri tarkoituksiin. Boostin osia on siirretty ja siirretään osaksi C++:n standardikirjastoa. Kirjasto sisältää niin muistinhallintaan, rinnakkaistamiseen, laskentaan, indeksointiin, tietorakenteisiin, kuin moneen muuhunkin geneerisiä toteutuksia. Kirjasto on julkaistu Boost:n avoimen lähdekoodin lisenssillä, joka on hyvin vastaava kuin BSD tai MIT. Kirjaston viimeisin versio 1.72 on julkaistu vuonna 2019. Lisätietoja: <a href="https://www.boost.org/">https://www.boost.org/</a>

### 5.3 Kuvankäsittelymenetelmien toteuttaminen

Työssä kuvatut menetelmät toteutettiin ArboSDK-ympäristön rajapintoja toteuttaen. Menetelmien valintaan vaikutti aineistojen silmämääräinen arviointi ja joissain tapauksissa eri menetelmien kokeileminen käytännössä. Paikkatietoaineistolle on ominaista, että kaikella aineistolla on jokin maantieteellinen sijainti, ja aineistossa voi olla puutteita tai epätarkkuuksia. Lisäksi ilmakehän aineistossa voi olla ilmakehän ominaisuuksien tai pilvisyyden aiheuttamia aukkoja tai reikiä. Maantieteellisen yhteyden vuoksi menetelmien parametointi toteutettiin käyttäen metrejä yksikköinä pikseleiden sijaan siellä missä se oli mahdollista.

Puutteellisten aineistojen vuoksi kaikissa menetelmissä tuli ottaa huomioon reikien mahdollisuus eli aineistojen 'NODATA'-arvojen käsiteltiin menetelmästä riippuen soveltuvimmalla tavalla. Käytännössä tämä tarkoittaa monessa algoritmossa, että menetelmien tuloksia täytyy normalisoida oikeasti spatiaalisen ikkunan sisällä havaituista arvoista. Esimerkiksi keskiarvoistavassa suodatuksessa tulee normalisoida arvo vain suotimen siltä osalta, jolla kuvassa ei ollut ns. 'NODATA'-arvoja.

ArboSDK-ympäristössä kuvankäsittelymenetelmät toteutettavat joko IRasterOp tai IRasterReduceOp rajapinnat, jotka molemmat perivät IAbstractRasterOp abstraktin luokan. Rasterioperaatiot saavat parametrikseen yhden tai useamman rasterikuvan tai rasterioperaation, jolloin operaatioita voidaan ketjuttaa ja tarvittaessa tallentaa välimuistiin, joka on hyödyllistä, mikäli saman operaation tulosta hyödynnetään useammassa prosessin vaiheessa. Operaatio suoritetaan Just-In-Time -filosofian mukaisesti kokonaisuena ketjuna vasta, kun sitä tarvitaan eli esimerkiksi tallennetaan tulos tai eksplisiittisesti suoritusta kutsutaan esimerkiksi redusoivan operaation tulosten tuottamiseksi.

Kaikissa kuvankäsittelyoperaatioissa käsitellään kuvia prosessointialueittain, ja vain niiltä prosessointialueilta, jotka ovat käsiteltävän aluegeometriarajauksen sisällä. Kuvan jako prosessointialueisiin johtuu useassa tapauksessa käytetystä kuvaformaattista, joissa kuva on usein tallennettuna lohkoissa. Spatiaalisissa suodatuksissa tulee silloin huomioida, ettei operaatio käsittele koko kuva-aineistoa kerralla eli operaatiosta riippuen tulee huomioida osatulosten yhdistäminen. Lisäksi käsiteltävän aluegeometrian rajaaminen tulee suorittaa prosessissa vasta siinä vaiheessa, kun se on tarpeen. Tulosteessa 2 tehdään sama mediaanisuoatus samalle kuvalle, mutta lopputulos eroaa rajaavan alueen (geometry) reunoilla, koska ensimmäisestä kuvasta ensin leikataan alue ja sitten suodatetaan ja jälkimmäisessä kuva suodatetaan ennen leikkaamista. Laskennallisesti näissä ei ole juurikaan eroa, koska ensimmäisessä mediaanisuoatukseen tulee tuon rajaavan alueen ulkopuolelle vain 'NODATA'-arvoja oikeiden kuvan arvojen sijaan.

Operaatioita toteuttaessa tulee määrittellä syötteen ja tulosteen tietotyyppi tai sallitut tietotyypit. Aineistojen ja operaatioiden mahdollisia tietotyyppisiä pikseleille ovat byte, short, unsigned short, int, unsigned int, float ja double. Lisäksi operaation tulee määrittellä, onko

kyse spatiaalisesta operaatiosta ja kuinka suuri on operaation tarvitsema käsittelyikkuna, voidaan operaatio suorittaa värikanavittain separoituna vai tuleeko operaatio suorittaa kuvan kaikille värikanaville kerralla, ja operaation aarisuus eli onko kyseessä binäärinen, unaarinen vai n-aarinen operaatio. Redusoivat operaatiot tarkoittavat operaatiota, joiden tuloksena voi tulla mitä tahansa esimerkiksi redusoiva operaatio voi esimerkiksi palauttaa yksittäisen suurimman intensiteetti arvon koko kuva-aineiston kattamista pikseleistä, tai kuvasta johdetun graafiesityksen.

#### Tuloste 2: Kuvankäsittelyoperaatiot ArboSDK-ympäristössä

---

```
1 sptr<const IRaster> lena = workspace->openRasterDataset(L"lena.tif", RONone);
2
3 auto extractedLena = ExtractByMask(geometry, lena);
4 auto medianExtractedLena = Median(extractedLena, Length::pixels(3));
5 copy(medianExtractedLena, workspace, L"medianExtractedLena.tif", &ptrack);
6
7 auto medianLena = Median(lena, Length::pixels(3));
8 auto extractedMedianLena = ExtractByMask(geometry, medianLena);
9 copy(extractedMedianLena, workspace, L"extractedMedianLena.tif", &ptrack);
```

---

---

## 5.4 Alueiden yhdistäminen graafista

Aiemmin kuvattujen kuvankäsittely- ja kaukokartoitusaineistojen käsittelymenetelmien lisäksi työssä toteutettiin menetelmät graafin luontiin segmentointituloksesta, sisäisten piirteiden laskentaan graafille, samankaltaisten alueiden yhdistämiseksi graafista, ja kuvan uudelleennimiöimiseksi graafin perusteella.

Graafi luodaan segmentointituloksesta tunnistamalla segmentoinnilla nimiöidyt alueet ja niiden naapurit. Samalla lasketaan graafin solmuille pinta-alat ja kaarille sen yhdistävien segmenttien välisten reunaviivojen pituudet. Lisäksi näille solmuille ja kaarille lasketaan segmentoinnissa käytetyn gradienttirasterin arvoista статистиikkaa kuten minimi, keskihajonnat ja keskiarvot. Kaarien osalta статистиikka lasketaan kaikkien rajan molemminpuolisten pikseliparien suuremmista intensiteettiarvoista eli siitä arvosta, joka on suurempi rajan ylittämiseksi.

Sisäisten piirteiden laskennassa lasketaan kunkin segmentin alueella kuvasta tai kuvista piir-

teitä, joista tässä työssä käytettiin samoja tilastollisia arvoja kuten graafin muodostuksessa. Työn yhteydessä toteutettiin myös Haralick-tekstuuripiirteiden sisäisten piirteiden laskin mahdollista myöhempää käyttöä varten. Haralick-piirteet ovat yhteistapahtumamatriisista johdettuja tilastollisia muuttujia eli matriisista, jossa on pikselin ja viereisen pikselin arvojen yhteistapahtuman todennäköisyydet.

Samankaltaiset naapurit yhdistetään graafista iteratiivisella prosessilla, jossa verrataan alueiden sisäisten piirteiden samankaltaisuutta eli tässä tapauksessa parametreilla painotettua euklidista etäisyyttä. Yhdistämistä rajoittavat ulkoiset piirteet eli kohteiden yhteisen rajaviivan pituuden suhde koko mahdollisesti yhdistettävän alueen rajaviivaan, yhdistettävän alueen pyöreys, yhdistettävien kohteiden välinen dynamiikka, ja leikkaavatko samankaltaisuuden laskennassa käytettyjen piirteiden jakaumat toisiaan.

Algoritmissa 8 on esitetty käytetty kuvioiden yhdistämismenetelmä. Algoritmin funktio *MakeGraph* tuottaa nimiöidystä kuvasta graafin, jossa gradienttikuvasta johdetaan alueille eli graafin solmuille tilastolliset arvot ja vastaavat arvot lasketaan sisäisiksi piirteiksi kuvan *img* värikanavista. Solmuja yhdistäville kaarille lasketaan arvot gradientista alueita yhdistävien reunapikseleiden osalta. Graafin kaaret voidaan järjestää funktiolla *SortByDynamics*, jolloin lasketaan kaaren yhdistävien solmujen välinen dynamiikka *AvgDynamics*-funktioilla, jonka mukaan kaarien lista järjestetään. Toisin kuin dynamiikkasuodatuksessa tässä dynamiikkaan käytetään reunalla ylitettävien pikseleiden keskiarvoa minimin asemasta. Pyöreuden laskentaan käytetään funktioita *CombinedArea* ja *CombinedPerimeter*, jotka palauttavat kaaren yhdistävien solmujen yhteisen pinta-alan ja ympärysmittan. Solmujen yhdistämisen ehtona on lisäksi vaatimus, että yhdistettävien solmujen kuvasta laskettujen piirteiden jakaumien täytyy olla limittäin keskihajontojensa puitteissa. Graafin solmut yhdistetään funktiolla *Merge*, joka yhdistää gradientin mukaan suuremman arvon saaneen solmun pienempään yhdistäen näiden solmujen tilastolliset muuttujat ja listat solmun kaarista. Solmujen yhdistäminen ei poista olemassa olevia solmuja vaan merkitsee poistettavan solmun juureksi sen solmun, johon se yhdistetään.

Uudelleennimiöimisessä nimiöidään segmentoinnin tulos muokatun graafin perusteella uudelleen. Graafista yhdistettyjä ja poistettuja solmuja vastaavat pikselit asetetaan vastaamaan

niitä graafin solmuja, joihin ne ovat yhdistettyjä. Lopputuloksena on nimiöity kuva, jossa alueet ovat yhdistetty ja jota voidaan käyttää rasterikuvan vektorisoinnissa.

---

### Algoritmi 8: Alueiden yhdistäminen

---

```

1  input : image f
2          array of numbers similarityWeights
3          number roundnessLimit, similarityLimit
4          image  $\{img, grad\}$ 
5  output : graph
6  begin
7      g  $\leftarrow$  MakeGraph(f, grad, img)
8      merged  $\leftarrow$  TRUE
9      while (merged)
10         merged  $\leftarrow$  FALSE
11         SortByDynamics(g.edges) /* g.edges dynamiikan mukaan nousevasti */
12         for e  $\in$  g.edges
13             if AvgDynamics(e) > dynamicsLimit
14                 break
15             end
16             roundness  $\leftarrow$   $4\pi$ CombinedArea(e)/CombinedPerimeter(e)
17             if roundness  $\geq$ 
18                 Min(roundnessLimit, e.n1.roundness, e.n2.roundness)
19                 if WeightedDistance(e, similarityWeights)  $\leq$  similarityLimit
20                     if SimilarityOverlap(e) = TRUE
21                         Merge(e)
22                         merged  $\leftarrow$  TRUE
23                     end
24                 end
25             end
26         end
27     end
28     return g
29 end

```

---

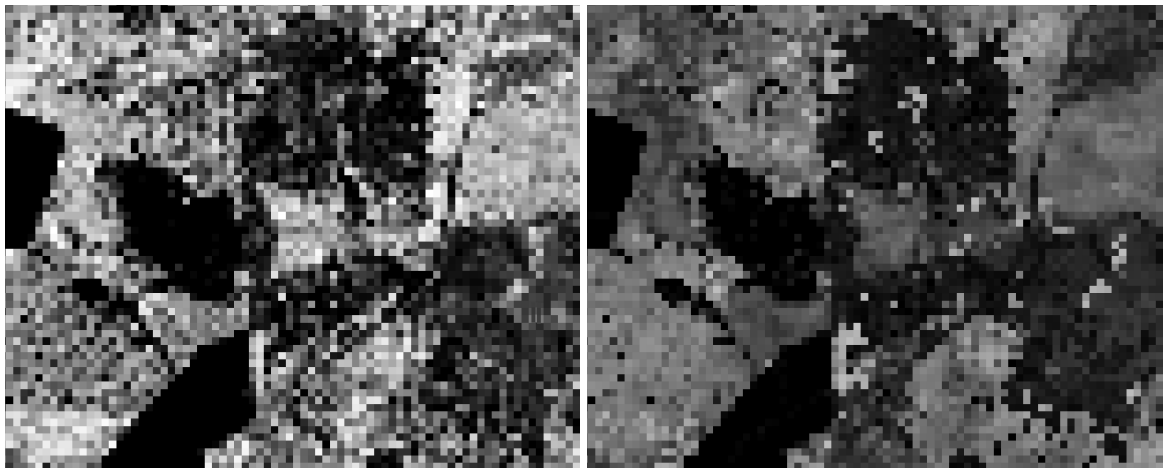
## 5.5 Menetelmä metsikkökuvioiden tuottamiseen

Kehitetty menetelmä voidaan jakaa kolmeen osaan. Menetelmän eri vaiheet ovat aineiston esikäsitely, segmentointi ja segmentoinnin jälkikäsitely. Kaikki menetelmän vaiheet voidaan suorittaa joko peräkkäin ketjutettuna tai erillisinä vaiheina. Erillisinä vaiheina suoritettaessa seuraavien vaiheiden tarvitsemat aineistot tallennetaan tiedostoihin. Ketjutettuna vaiheet suoritetaan kerralla siten, että edeltävien vaiheiden tulokset tallennetaan välimuistiin

vain, jos sitä hyödynnetään useammassa eri operaatioissa.

### 5.5.1 Aineiston esikäsittely

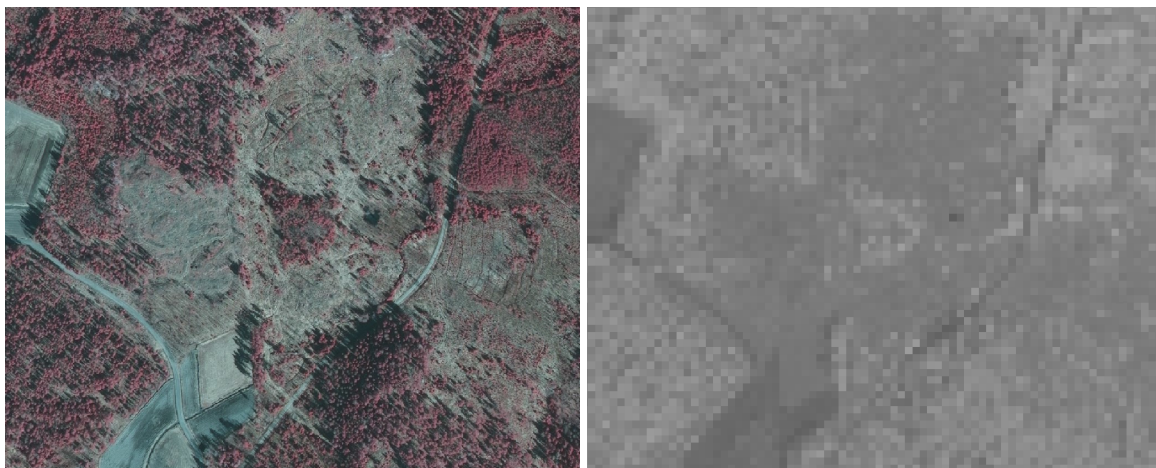
Aineiston esikäsittelyssä tuotetaan ensin laserkeilausaineistosta puuston korkeutta kuvaava latvusmalli ja kasvillisuuden tiheyttä esittävä kuva, jossa pikseleiden arvot ovat laskettu alle 2 metriä maanpinnasta olevien laserkeilauspisteiden suhteesta kaikkiin käsiteltävän pikselin naapurustoon osuneisiin mittauspisteisiin. Koska Maanmittauslaitoksen laserkeilausaineisto on suhteellisen harvaa eli n. 0.5 pistettä/m<sup>2</sup>, niin rasteroinnissa käytettäväksi naapurustoksi valittiin 8 metriä ja pikselikooksi myöskin 8 metriä. Kuvassa 35 on laserkeilausaineistosta rasteroituna kasvillisuuden tiheys ja puuston korkeus.



Kuva 35: Vasemmalla on 8 metrin pikselikoolla, 8 metrin rasterointi-ikkunalla laserkeilausaineistosta rasteroitu kasvillisuuden tiheyttä kuvaava kuva ja oikealla vastaavasti rasteroitu puuston korkeutta esittävä latvusmalli.

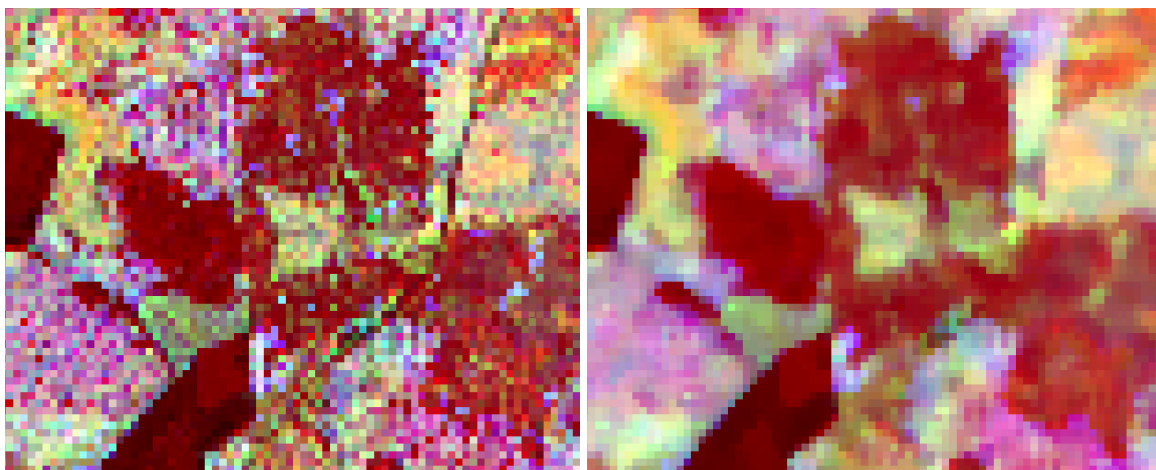
Kasvillisuusindeksin laskentaan voi menetelmässä valita joko EVI2 tai NDVI -indeksin, joka johdetaan alkuperäisen aineiston tarkkuudella ja uudelleennäytteistetään laserkeilausaineistotasojen tarkkuuteen käyttäen kuutiollista interpolointia, joskin tässä tapauksessa kuvan tarkkuutta heikentäessä interpolointimenetelmällä ei ole suurta väliä. Kuvassa 36 on esimerkkinä käytetyn alueen ilmakehäväärivärikuvana ja siitä johdettu laserkeilausaineistosta rasteroitujen kuvien tarkkuuteen ja maantieteelliseen sijaintiin (georeferenssiin) uudelleennäytteistetty kasvillisuusindeksi.

Kasvillisuusindeksi, kasvillisuuden tiheys ja puuston korkeus yhdistetään yhdeksi kuvaksi



Kuva 36: Vasemmalla vääräväri-ilmakuva, jonka lähi-infrapunakanavasta ja punaisesta kanavasta johdettu oikealla oleva NDVI kasvillisuusindeksi.

kukin erillisiksi värikanavikseen, jonka jälkeen aineiston voimakasta kohinaa suodatetaan ensin käyttäen mediaanisuodatusta. Mediaanisuodatuksella pyritään silottamaan aineistosta puiden latvusten aiheuttamaa voimakastakin vaihtelua säilyttäen kuitenkin riittävän tarkat rajaukset metsiköltään erityyppisten alueiden välillä. Kuvassa 37 on aineistojen yhdistelmäkuva ja yhdistelmäkuva mediaanisuodatettuna.

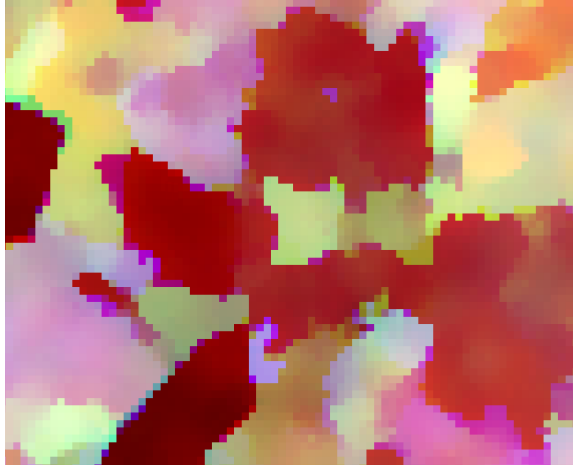


Kuva 37: Kuvissa punainen värikanava on kasvillisuusindeksi, vihreä on kasvillisuuden tiheys, ja sininen on puuston korkeus. Vasemman puoleisessa kuvassa on aineiston yhdistelmäkuva ja oikean puoleisessa sama kuva mediaanisuodatettuna 8 metrin säteellä.

Esikäsittelyn viimeisessä vaiheessa kuvalle suoritetaan Mean Shift -suodatus, jolla saadaan tasoitettua entisestään kohinaa ja säilytettyä reunaviivojen piirteitä. Mean Shift -suodatuksen sijaan tässä voidaan käyttää myös bilateraalista suodatusta. Tässä vaiheessa lasketaan kuvasta eri värikanavien keskiarvot ja keskihajonnat, joiden mukaan värikanavat normalisoidaan



nollakeskiarvoisiksi ja keskihajonnoiltaan toisiaan vastaaviksi. Kuvassa 38 on mediaanisuodatettu kuva Mean Shift suodatettuna. Mean Shift -suodatuksessa on tässä käytetty Gaussin ydintä Mean Shift iteraatioissa naapuruston painotuksessa.

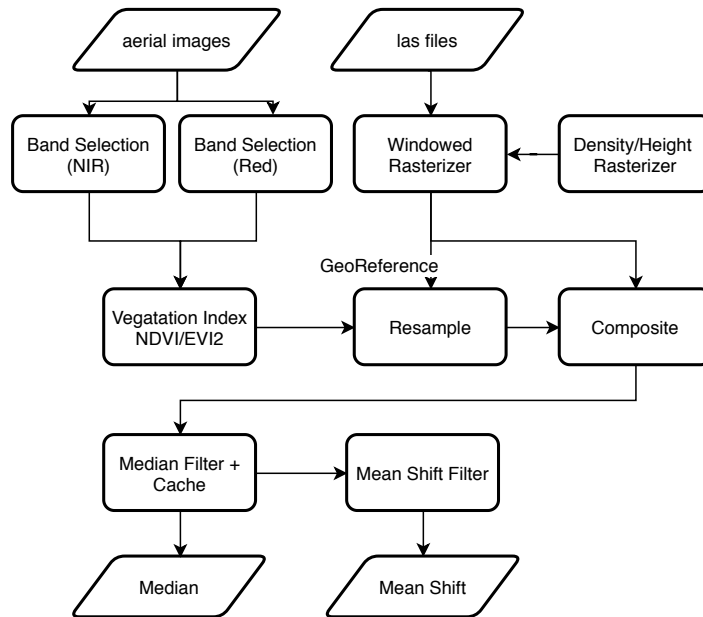


Kuva 38: Mean Shift suodatettu kuvan 37 oikeanpuoleinen kuva käyttäen suodatuksessa 24 metrin spatiaalista sädettä ja värikanavien spektraalisia säteitä 0.2, 0.3 ja 5.0.

Taulukossa 5 on koostettu menetelmän esikäsittelyssä käytetyt esimerkkiparametrit ja kuvassa 39 on kuvattu esikäsittelyvaihe prosessina, jossa lähtöaineistona on maanpinnantasolle korjattu laserkeilausaineisto, ja orto-oikaistu ilmakehän kuva-aineisto sisältäen ainakin punaisen ja lähi-infrapuna kanavan. Seuraavat vaiheet hyödyntävät tämän vaiheen Mean Shift ja mediaanisuodatusten tuloksia.

Taulukko 5: Esikäsittelyn esimerkkiparametrit

Windowed Rasterizer	Pikselikoko: 8.0 metriä.
	Rasterointi-ikkuna: 8.0 metriä.
Median Filter	Säde: 8.0 metriä.
Mean Shift Filter	Spatiaalinen säde: 24 metriä.
	Kasvillisuusindeksin säde: 0.2.
	Kasvillisuuden tiheyden säde: 0.3.
	Puuston korkeuden säde: 5.0 metriä.



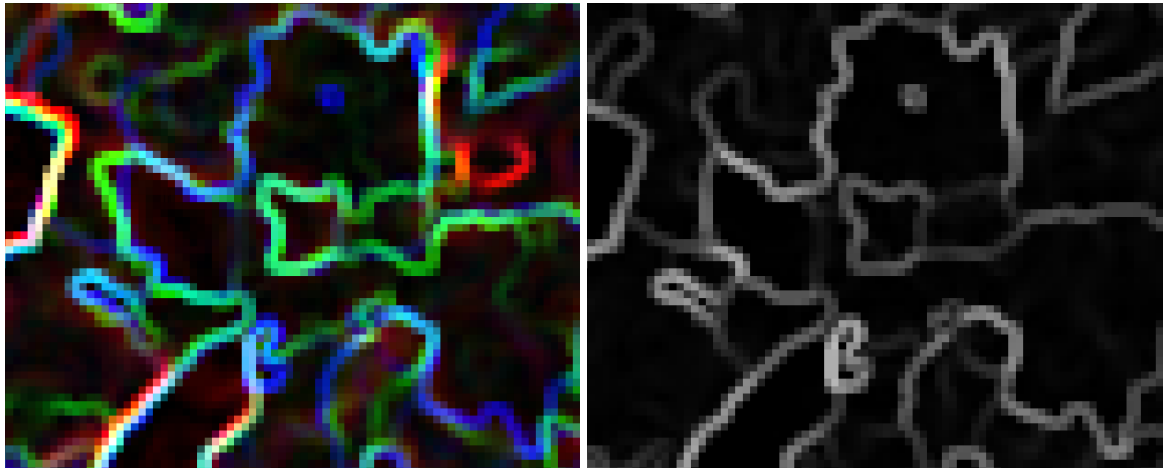
Kuva 39: Esikäsittelyprosessi kaaviona, jonka vaiheita vastaavat esimerkkiparametrit ovat taulukossa 5. Kasvillisuusindeksin uudelleennäytteistystä varten lainataan rasteroidusta aineistosta georeferenssi, jolloin aineistot saadaan vastaamaan maantieteellisesti toisiaan. Prosessin tuloksena muodostuu mediaanisuodatettu yhdistelmäkuva ja tästä Mean Shift -suodatettu kuva.

### 5.5.2 Segmentointi

Segmentointivaiheen tarkoituksena on tuottaa esikäsittelystä aineistosta alustavia met-sikkökuvioita. Segmentoinnissa tässä työssä päädyttiin lopulta käyttämään kuvan gradientin paikallisista minimeistä Watershed-muunnoksella etsittyjä vedenjakajina toimivia harjanteita. Tässä toteutettiin myös vedenjakajien etsiminen valuttamalla ylhäältä alaspäin, mutta silloin prosessointialueiden reunoilla oli ongelmia alueiden yhdistämisessä. Lisäksi selvitettiin simuloituun jäähtymiseen perustuvan ja ns. split-and-merge -segmentointien soveltuvuutta, mutta silmämääräisellä tarkastelulla Watershed-muunnokseen perustuva segmentointi vaikutti soveliaimmalta.

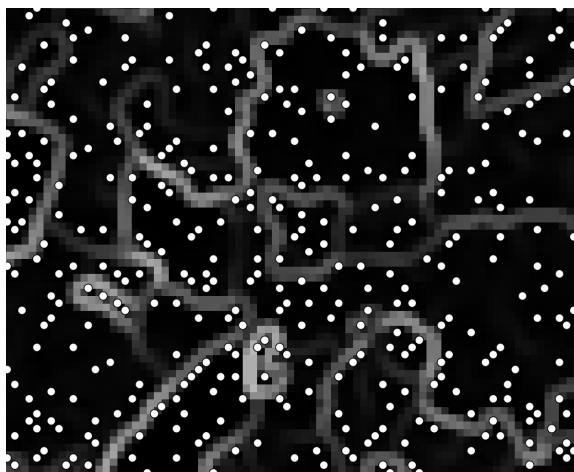
Tämä vaihe hyödyntää esikäsittelyn tuloksia ja se alkaa esikäsittelyn kuvan gradientin laskennalla Sobel suodatuksella. Kuvassa 40 on Mean Shift -suodatuksen tuloksesta Sobel suodatuksella saadut gradienttien voimakkuudet, joista on laskettu gradienttien voimakkuuksien summa painottamalla eri värikanavia parametroidin mukaan ja jakamalla kukin parametri sitä vastaavan mediaanisuodatetun kuvan värikanavan keskihajonnalla, jolloin käytetyt pai-

nokerroinparametrit ovat selkeämmin ymmärrettäviä eri värikanavien ollessa keskenään eri asteikoilla.



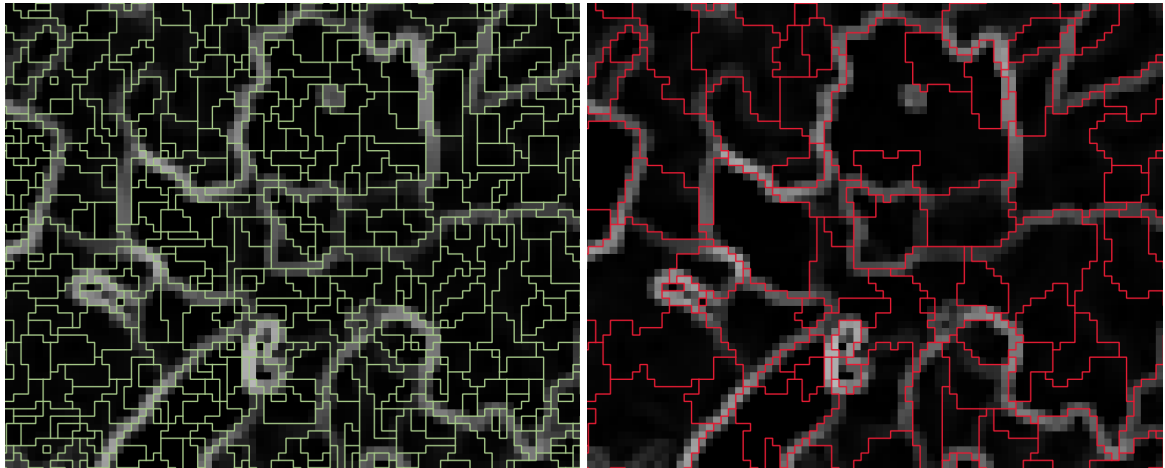
Kuva 40: Mean Shift -suodatetusta kuvasta johdetaan kanavittain gradientin voimakkuus käyttäen Sobel suodatusta, joka on kuvassa vasemmalla ja gradientin voimakkuudesta lasketaan painotettu summa Watershed-muunnosta varten, joka on kuvassa oikealla.

Gradienttien voimakkuuksien summaa käytetään vedenjakajien muodostamiseen. Käytetty Watershed-muunnos tarvitsee lähtöpisteet, jotka etsittiin paikallistamalla kuvasta kaikki paikallisten minimien sijainnit, jotka ovat esitettynä pisteinä kuvassa 41. Tästä voi havaita menetelmän ylisegmentoinnin syyn eli paikallisia minimejä voi olla todella paljon. Alkupisteiden suodatukseen voi käyttää esimerkiksi ääriarvon dynamiikan laskentaa ja paikallisten minimien suodatusta liian pienen dynamiikan perusteella, mutta tässä työssä minimien suodatus dynamiikalla tehtiin valuma-alueiden etsinnän jälkeen iteratiivisesti.



Kuva 41: Käytetty Meyerin tulvimiseen perustuva Watershed-muunnoksen tarvitsemat alkupisteet ovat etsitty gradienttien voimakkuuksien summasta valitsemalla ne pikselit, joiden intensiteetti on pienempi tai yhtä suuri kuin naapuruston pikseleiden intensiteetti.

Segmentointi Watershed-muunnoksella toteutettiin algoritmia 3 käyttäen, josta muodostuu pieniä alueita rajoittuen gradienttikuvan mukaan siten, että kunkin alueen kaikki pikselit ovat intensiteetiltään vähintään yhtä suuria kuin alueen minimi. Käytetty ArboSDK ympäristö prosessoi laajempaa kuvaa tai kuvien mosaiikkia alueittain esimerkiksi 512x512 pikselin lohkoissa, josta takia prosessointialueen reunoille lisätään keinotekoiset paikalliset minimi eli reunoilla tarkastellaan, onko reunan pikseli paikallinen minimi vain prosessointialueella olevan naapuruston mukaan. Kuvassa 42 on esitetty Watershed-muunnoksen lisäksi myös minimien dynamiikan mukaan suodatettu segmentointi, joka yhdistää myös prosessointialueen reunoilla olevat alueet yhtenäisiksi, mikäli reunalle lisätty minimi ei ollut koko kuvan suhteen todellinen paikallinen minimi, koska näiden välinen dynamiikka on vedenjakajana toimivan todellisen reunan puuttuessa 0.



Kuva 42: Kuvassa vasemmalla on Watershed-muunnoksella tuotetut rajaukset vihreällä, jossa näkyy menetelmälle ominainen voimakas ylisegmentointi. Oikealla sama segmentointi dynamiikkasuodatettuna siten, että kunkin alueen dynamiikan on oltava vähintään 20. Tässä dynamiikkasuodatuksessa on käytetty algoritmia 9, joka iteratiivisesti yhdistää kuvioita, joiden dynamiikka on liian pieni.

Dynamiikka suodatus toteutettiin iteratiivisesti algoritmin 9 mukaisesti, jossa *MakeGraph* tuottaa nimiöidystä kuvasta graafin, jossa kullekin solmulle on gradientin voimakkuudesta laskettu pienin arvo. Kaarille arvot ovat laskettu gradientin voimakkuudesta sen mukaan mikä on pienin mahdollinen arvo kaaren yhdistäviä solmuja vastaavien alueiden rajalla eli pienin ylitettävä arvo alueelta toiselle siirtymisessä. Funktio *GetOtherNode* palauttaa kaaren toisen solmun, *Merge* yhdistää kaaren solmut siten, että jäljelle jää se solmu, jonka arvo on pienin ja yhdistäen tähän jäljellejäävään solmuun poistettavan solmun kaaret. Mikäli yhdistetylle solmulle tulee moninkertaisia kaaria, niin näistä valitaan jäljelle se, jonka arvo

on pienempi. Funktio *Relabel* tuottaa lähtöaineistosta uuden kuvan nimiöiden pikselit uudelleen poistettujen solmujen osalta vastaamaan niitä niiden solmujen nimiä, joihin se on lopulta yhdistetty.

---

### Algoritmi 9: Kuvioiden dynamiikkasuodatus

---

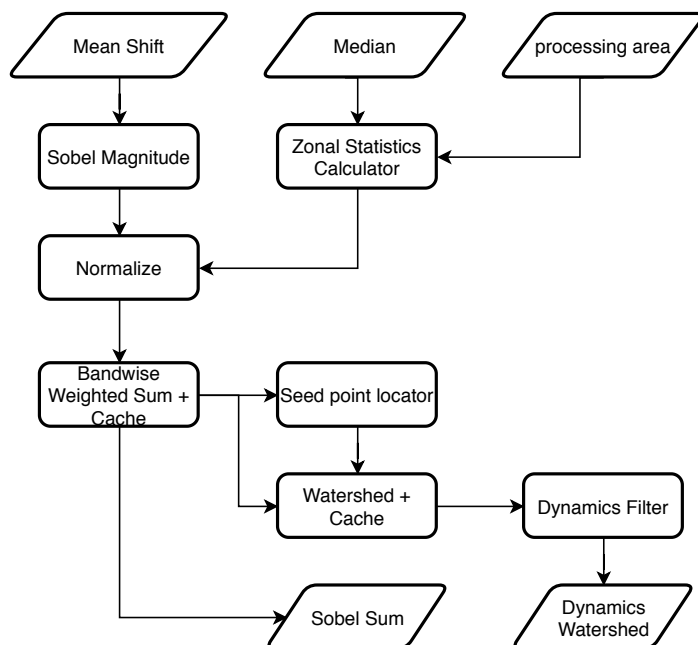
```

1  input:  image f /* labels */
2          number dynamicsLimit
3          image grad /* gradient */
4  output: image
5  begin
6    g ← MakeGraph(f, grad)
7    merged ← TRUE
8    while (merged)
9      merged ← FALSE
10     for n ∈ g.nodes
11       emin ← NULL
12       for e ∈ n.edges
13         if n.value ≥ GetOtherNode(e, n).value and e.value − n.value ≤ dynamicsLimit
14           if emin = NULL or emin.value < e.value
15             emin ← e
16           end
17         end
18       end
19       if emin ≠ NULL
20         Merge(e)
21         merged ← TRUE
22       end
23     end
24   end
25   return Relabel(f, g)
26 end

```

---

Segmentointiprosessin esimerkkiparametointi on esitetty taulukossa 6 ja prosessikaavio on esitettyinä kuvassa 43, jossa lähtöaineistona on esikäsittelyn Mean Shift -suodatuksen tulos, ja parametrien normalisoinnissa käytetty mediaanisuodatuksen tulos yhdistettynä käsittelyalueisiin. Tuloksena segmentointiprosessista tulee kuva, joka on dynamiikkasuodattu Watershed-muunnoksella tuotettu kuva, jossa erilliset alueet ovat nimiöityinä.



Kuva 43: Segmentointiprosessin kaavio, josta tuloksena muodostuu jälkikäsitellylle gradienttien voimakkuuksien summa ja dynamiikalla suodatettu Watershed-muunnoksella tuotettu nimiöity kuva, jossa pikselit ovat nimiöity sen alueen mukaan mihin kukin pikseli kuuluu.

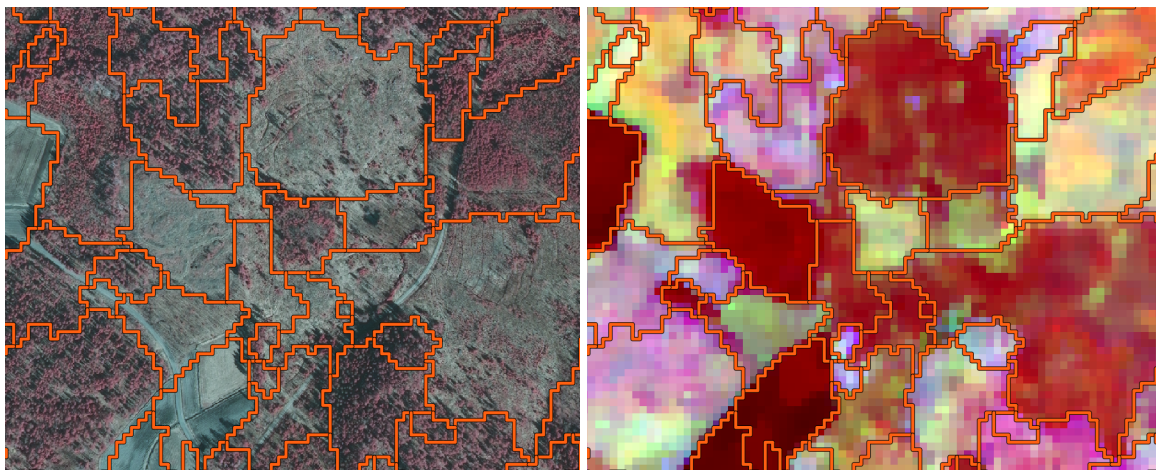
Taulukko 6: Segmentoinnin esimerkkiparametointi

	Kasvillisuusindeksin painokerroin: 0.1.
Bandwise Weighted Sum	Kasvillisuuden tiheyden painokerroin: 0.3. Puuston korkeuden painokerroin: 0.6.
Dynamics Filter	Dynamiikan kynnys: 20.

### 5.5.3 Jälkikäsitely

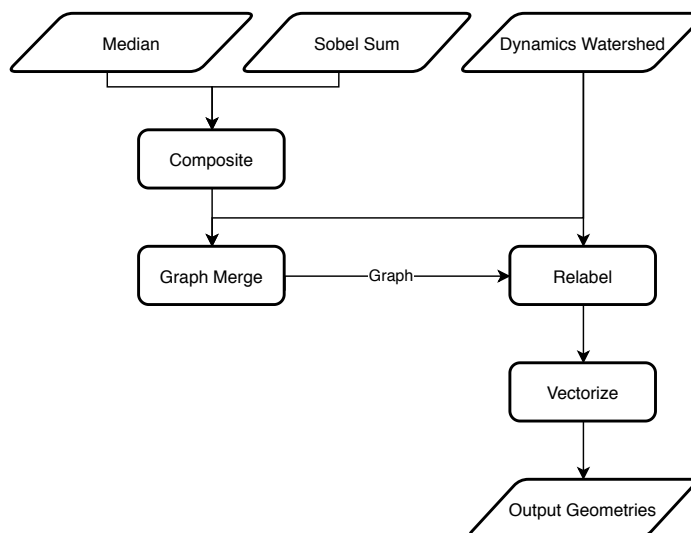
Dynamiikkasuodatetusta Watershed-muunnoksesta johdetusta nimiöidystä kuvasta yhdistellään kuvioita jälkikäsitelyssä algoritmin 8 mukaisesti. Menetelmän tarkoituksena yhdistää pareittain sopivia toistensa naapurialueita huomioiden, ettei yhdistettyjen alueiden muoto joko heikkene tai ylitä kynnysarvoa, ja huomioimalla yhdistettävien alueiden sisäinen samankaltaisuus. Kuvassa 44 on kuvioiden yhdistämisen tulos.

Jälkikäsitelyn esimerkkiparametointi on taulukossa 7 ja prosessi on esitetty kuvassa 45, jossa prosessin syötteenä ovat esikäsitelyn mediaanisuodatettu, segmentoinnin Sobelin gradienttien summa, sekä segmentoinnin dynamiikkasuodatettu Watershed-muunnoksella tuo-



Kuva 44: Segmentoinnin tuloksesta on yhdistetty kuvioita käyttäen algoritmia 8, ja yhdistämisessä käytetyt parametrit ovat taulukossa 7. Yhdistetyt alueet ovat visualisoituna vasemmalla vääräväri-ilmakuvan päälle ja oikealla segmentoinnin esikäsitellyssä tuotetun medianisuodatetun kuvan päälle.

tettu nimiöity kuva. Tuloksena prosessista tulee joukko viivageometrioita, joilla on attribuuttitietona viivan vasemman- ja oikeanpuoleisen alueen nimiö. Näiden attribuuttitietojen perusteella voidaan tuottaa aluemaisia monikulmiokohteita käyttäen jonkin paikkatietoympäristön perustoiminnallisuksia.



Kuva 45: Prosessikaavio segmentoinnin jälkikäsitelystä, jossa segmentoinnin tuloksesta yhdistetty alueita käyttäen algoritmia 8 ja yhdistämisessä käytetyt parametrit ovat koostettuna taulukossa 7.

Taulukko 7: Jälkikäsittelyn esimerkkiparametrointi

---

---

Graph Merge	Kasvillisuusindeksin painokerroin: 0.15.
	Kasvillisuuden tiheyden painokerroin: 0.2.
	Puuston korkeuden painokerroin: 0.4.
	Yhteisen reunaviivan suhteellinen osuus: 10%.
	Yhdistelmän pyöreys vähintään: 0.25.
	Suurin sallittu dynamiikka yhdistettävien välillä: 40.

---

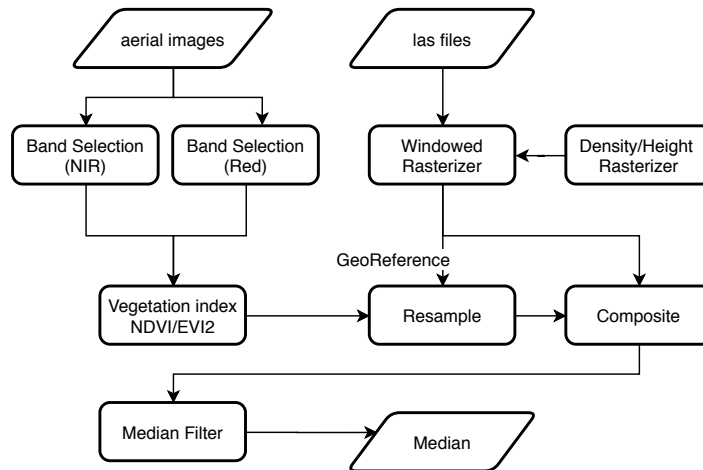
## 5.6 Menetelmä metsikkökuvioiden rajojen tarkentamiseen

Työssä yritettiin parantaa metsikkökuvioiden tuottamisen laatua lisäämällä segmentointiin suoraan alueen ulkoisiin piirteisiin eli alueen reunan muotoon liittyviä rajoitteita. Tällöin segmentoinnin tuloksena tuli herkästi reikiä ja muita epäjohdonmukaisuuksia. Active Contour -menetelmässä reunan muodon rajoitteet toimivat reunaviivan sisäisenä energiana ja reunaviivaan vaikuttavina voimina. Kehitetty menetelmä rajojen tarkentamiseen voidaan jakaa kahteen osaan eli esikäsittelyyn ja rajojen tarkentamiseen. Näissä menetelmissä voidaan hyödyntää korkeampaa resoluutiota kuin aiemmin kuvatussa menetelmässä metsikkökuvioiden tuottamiseksi, koska tässä vaiheessa on käytössä jo kohtuullisen hyvä alkuarvaus kuvioiden rajauksille.

### 5.6.1 Aineiston esikäsittely

Kuviorajojen tarkentamisen esikäsittelyvaihe on suurelta osalta vastaava kuin aiemmin esitetty esikäsittelyvaihe menetelmässä metsikkökuvioiden tuottamiseksi. Pikselikokona on tässä käytetty 2 metriä ja rasterointi-ikkunan kokona 4 metriä, jolloin kuviointia voidaan tarkentaa ei vain reunaviivojen alipikselitarkkuuden vaan myös laskennassa käytettävän aineiston tarkkuuden vuoksi. Kuvassa 46 on esitetty kuviorajojen tarkentamisen esikäsittelyprosessi, jonka erona segmentoinnissa käytettyyn esikäsittelyprosessiin on se, ettei tässä tuoteta Mean Shift -suodatettua kuvaa. Vaikka parametrointi voikin olla eriävä tässä esikäsittelyssä, niin samaa ohjelmakoodia voidaan käyttää jättäen menetelmän esikäsittelyn viimeinen vaihe pois. Tässä käytetyt esimerkkiparametrit ovat esitettynä taulukossa 8.





Kuva 46: Kuviorajojen tarkennuksen esikäsittelyprosessi, jota vastaavat parametroidit ovat esitettyinä taulukossa 8. Tämä prosessi vastaa metsikkökuvioiden tuottamisen esikäsittelyprosessia.

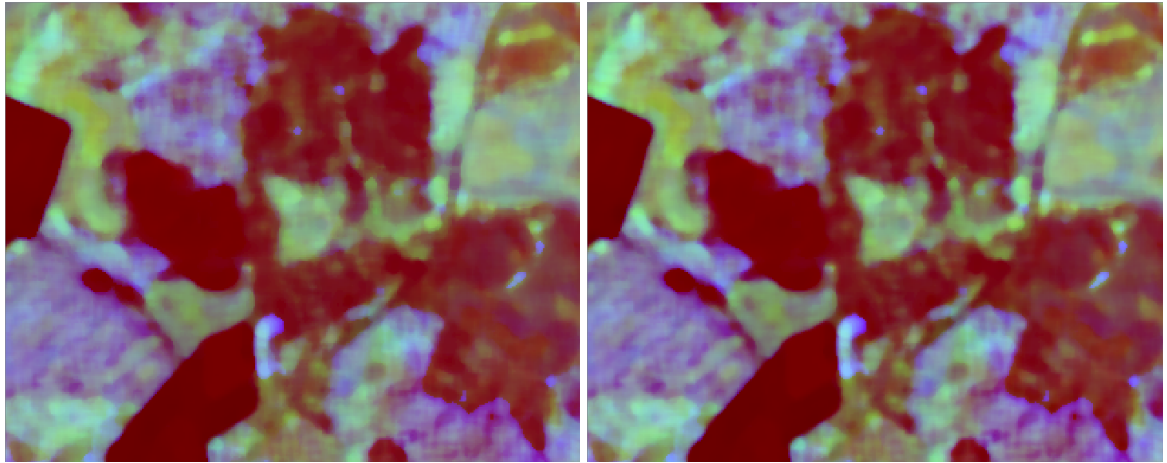
Taulukko 8: Rajojen tarkennuksen esikäsittelyn esimerkkiparametointi

Windowed Rasterizer	Pikselikoko: 2.0 metriä. Rasterointi-ikkuna: 4.0 metriä.
Median Filter	Säde: 4.0 metriä.

### 5.6.2 Kuviorajojen tarkentaminen

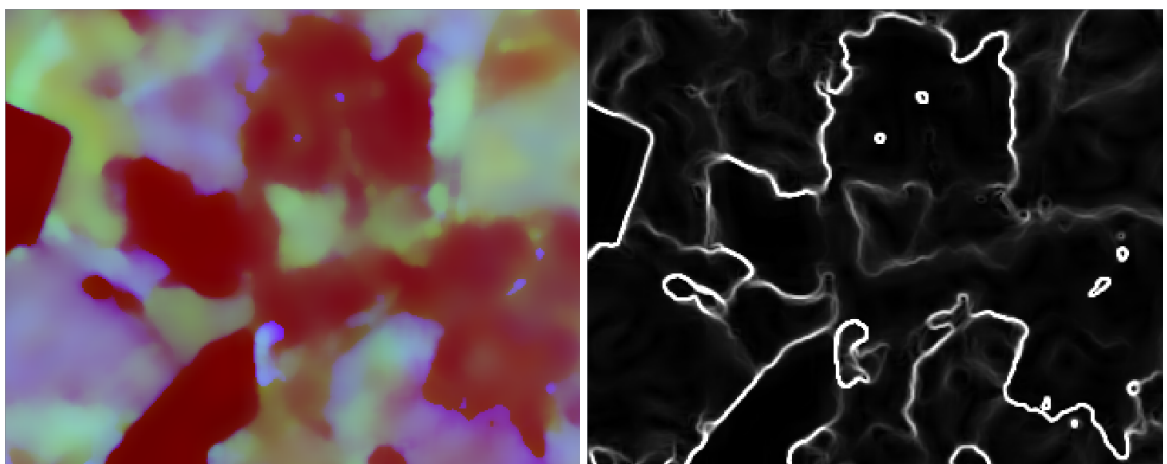
Esitetty menetelmä kuviorajojen tarkentamiseen terävöittää ensin lähtöaineiston korostaakseen alueiden reunoja. Mediaanisuodatetun kuvan terävöityksessä on käytetty toisen asteen derivaattaan perustuvaa menetelmää, jossa kuvaa terävöitetään lisäämällä siihen kuvan toisen asteen derivaattojen summa eli Laplacen suodatettu kuva. Laplacen suodatusta on tässä approksimoitu Gaussin suodatusten erotuksella. Tämä menetelmä mahdollistaa ottamaan huomioon reuna-alueiden oletetun leveyden valitsemalla sopivasti suodatinten käyttämät varianssit, jolloin terävöitys korostaa eniten halutun levyisiä reunoja. Kuvassa 47 on esitetty esikäsittelyn tulos ja sen terävöitys.

Aiemmin esitetystä menetelmästä metsikkökuvioiden tuottamiseksi poiketen tässä käytetään Mean Shift -suodatuksen sijasta bilateraalista suodatusta vastaavilla spektraalisilla säteillä, mutta spatiaalisenä säteenä tässä on käytetty 10 metriä. Sekä spatiaalisen, että spektraalisen etäisyyden painotuksessa on käytetty Gaussin ydintä.



Kuva 47: Vasemmalla mediaanisuodatettu kuva 2 metrin pikselikoolla ja oikealla terävöitetty kuva käyttäen terävöityksessä variansseja 1.7 ja 2.7.

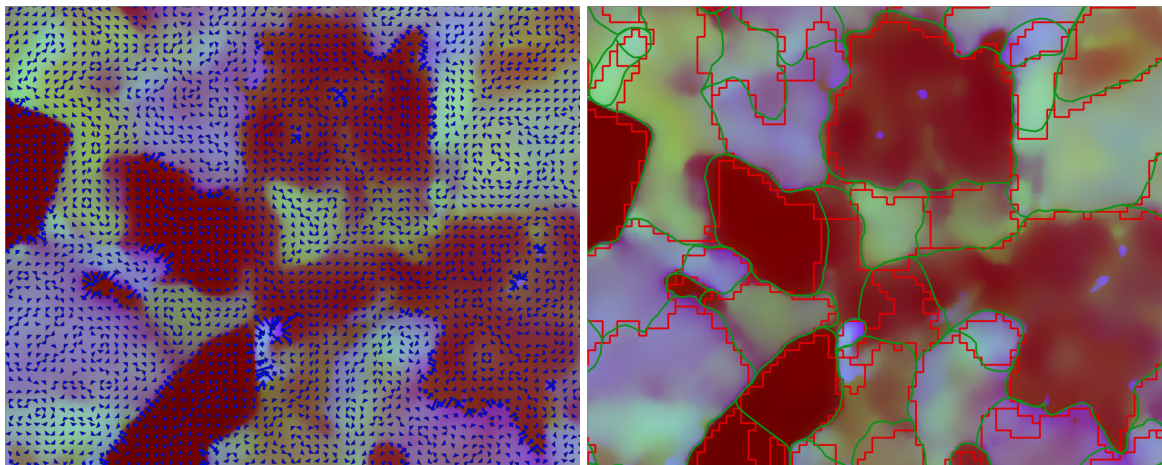
Bilateraalisella suodatuksella saadaan aikaan voimakkaasti silotettu kuva säilyttäen kuitenkin reunapiirteet. Menetelmä on laskennallisesti kevyempi kuin Mean Shift -suodatus, eikä tuota yhtä voimakasta, jopa segmentoivaa, silotusta. Kuvassa 48 on bilateraalisella suodatuksella käsitelty kuva ja siitä on laskettu gradienttien voimakkuuksien painotettu summa käyttäen taulukon 9 mukaisia painotuksen parametreja. Painotuksen parametrit normalisoidaan jakamalla painokertoimet mediaanisuodatetusta kuvasta lasketuilla parametreja vastaavien värikanavien keskihajonnoilla. Gradienttikuva voisi tässä olla hyvä suodattaa Gamma-korjauksella eli niin sanotulla Power Law -muunnoksella, jolla saataisiin kuvan intensiteettien varianssia pienennettyä.



Kuva 48: Vasemmalla on bilateraalisella suodatuksella suodatettu terävöitetty kuva, ja oikealla tästä kuvasta laskettu gradienttien voimakkuuksien summa, jossa gradientti on johdettu Sobel suodatuksella.

Gradienttien voimakkuutta esittävästä kuvasta johdetaan Active Contour -menetelmää varten seuraavaksi gradienttien vektorikenttä. Gradienttikuvaa hyödynnetään tässä, koska metsikkökuvioiden rajat ovat edellisessä vaiheessa tuotettu Watershed-muunnosta käyttäen vastaavasta gradienttikuvasta. Vektorikenttä johdetaan iteratiivisella GVF-menetelmällä, jossa parametroidulla vaikuttaa diffuusion voimakkuuteen eli siihen, kuinka paljon kuhunkin pisteeseen vaikuttaa naapuruston gradientit. Kuvassa 49 vasemmalla on kuvattu gradienttien vektorikenttää suuntanuolilla.

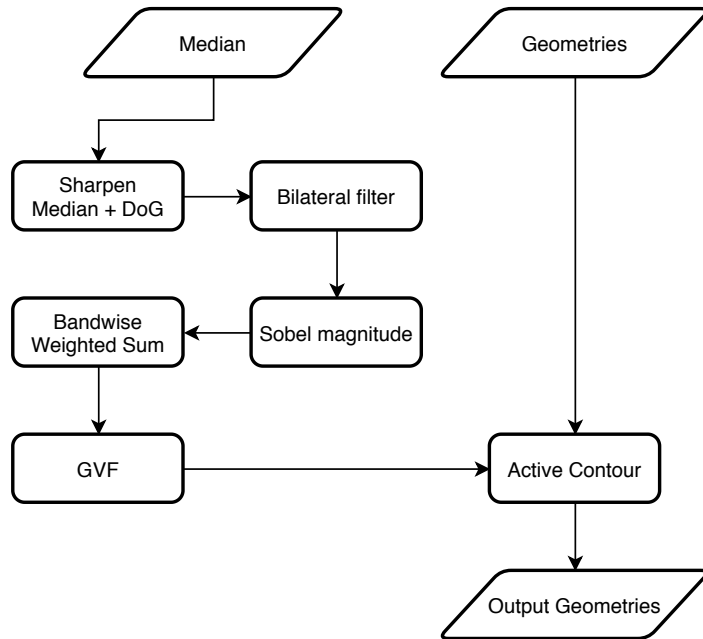
Rajojen korjaus tehdään käyttäen Active Contour -menetelmää, joka on esitetty algoritmissa 6. Rajan alkuarvauksena käytetään aiemmin tuotettuja metsikkökuvioiden rajoja. Näitä rajoja on kahden tyyppisiä rengasmaisia ja rajoja, jotka päättyvät toisten rajojen risteykseen. Rengasmaisella rajan rajaama alue on kokonaisuudessaan toisen alueen sisällä, kun toisten rajojen risteykseen päättyvät rajat muodostavat kuvioverkon. Kuvioverkon rajojen päätepisteisiin kohdistuvat voimat käsitellään siten, että kaikki samaan päätepisteeseen päättyvät rajat vaikuttavat toisiinsa estäen Active Contour -menetelmän siirtämästä näitä alueiden reunaviivojen päätepisteitä irti toisistaan. Tätä reunaviivojen korjausta jatketaan iteratiivisesti, kunnes kaikki reunaviivat ovat konvergoituneet eli näiden kokonaisenergia on lähellä 0. Kuvassa 49 on esitetty oikealla korjatut ja alkuarvauksina käytetyt kuviorajat.



Kuva 49: Vasemmalla gradienttien vektorikenttä tuotettuna GVF-menetelmällä, jota käytetään ulkoisena voimana Active Contour -menetelmälle. Oikealla Active Contour -menetelmällä korjatut kuviorajat vihreällä ja punaisella kuviorajojen alkuarvaukset. Kokonaisuudessaan tämä menetelmä on nimeltään Gradient Vector Flow Active Contour.

Esitetty menetelmä metsikkökuvioiden reunaviivojen tarkennukselle prosessina on kuvassa 50 ja tässä työssä käytetyt prosessin vaiheita vastaavat esimerkkiparametroidut ovat listat-

tuna taulukossa 9. Menetelmällä on haasteita löytää joissain kohdin reunaviivaa, jos alue on liian kapea ja tuotettu vektorikenttä ei osoita missään pisteessä koveran muodon sisälle. Lisäksi menetelmällä on haasteita niissä kohdissa, joissa ei korkeamman resoluution mukaisella aineistolla kuuluisi olla ollenkaan reunaviivaa, joten alueiden yhdistämisessä voisi olla myös hyödyllistä käyttää korkeamman resoluution aineistoa alueiden samankaltaisuuksien laskennassa käytettyjen alueiden sisäisten piirteiden laskentaan.



Kuva 50: Metsikkökuvioiden rajojen korjausmenetelmä prosessina, jossa lähtöaineistosta johdettua vektorikenttää hyödyntäen tarkennetaan kuviorajat Active Contour -menetelmällä.

Taulukko 9: Kuviorajojen tarkentamisen esimerkkiparametointi

Sharpen	Ensimmäinen varianssi ( $\sigma_1$ ): 1.7 (n. 3 metrin hajonta). Toinen varianssi ( $\sigma_2$ ): 2.7.
Bilateral Filter	Spatiaalinen säde: 10 metriä. Kasvillisuusindeksin säde: 0.2. Kasvillisuuden tiheyden säde: 0.3. Puuston korkeuden säde 5.0 metriä.
Bandwise Weighted Sum	Kasvillisuusindeksin painokerroin: 0.1. Kasvillisuuden tiheyden painokerroin 0.3. Puuston korkeuden painokerroin: 0.6.
GVF	Diffuusiokerroin ( $\mu$ ): 0.2.
Active Contour	Viivasegmentin minimipituus: 8 metriä. Elastisuus: 0.2. Jäykkyys: 0.3. Ulkoisen voiman paino: 0.5.

## 6 JOHTOPÄÄTÖKSET

Työn tavoitteena oli menetelmän kehittäminen metsikkökuvioiden tuottamiseen kaukokartoitusaineistosta, jonka toteuttamiseksi toteutettiin operatiivinen sovellus. Työn tavoitteen saavuttamiseksi toteutettiin menetelmät erityyppisten kaukokartoitusaineistojen yhdistämiseksi, joka tehtiin toteuttamalla laserkeilausaineiston rasterointitoiminnallisuus, toiminnallisuus kuva-aineiston uudelleennäytteistykseen ja hyödynnettiin ympäristön mahdollisuutta yhdistää eri rasteriaineistoja yhdeksi eri värikanaville. Työn tavoite aineiston muuntamiseksi metsikkökuvioiksi toteutettiin kuvankäsittelymenetelmiä käyttäen ja segmentoimalla kuva Watershed-muunnoksella ja näitä alueita yhdistämällä. Lisäksi kehitettiin ja toteutettiin menetelmä kuviorajojen tarkentamiseen optimoimalla kuvion reunaviivan piirteiden ja segmentoinnissa käytetyn aineiston ominaisuuksien mukaan Active Contour -menetelmällä.

Työssä kehitetyt menetelmät ovat olleet suurelta osin tuotannollisessa käytössä joidenkin yksityiskohtien ollen erilaisia, mutta yleisesti ottaen toteutetuilla menetelmillä pystytään tuottamaan hyvää ja käyttökelpoista metsikkökuviointia erilaisia kaukokartoitusaineistoja käyttäen. Tässä työssä esiteltyjen aineistojen osalta suuri osa kehitetyistä menetelmistä on myös tuotantokäytössä. Käytännössä kuvioinnin laatu riippuu myös aineistosta, mutta hyvälaatuisella aineistolla kuvionnin tuloskin on hyvä.

Menetelmän tuottamien metsikkökuvioiden laadunvarmistus on haasteellista myös silmämääräisesti, koska ilmakuvasta on hankala tunnistaa puuston korkeutta ja usein myös lajistoa. Tarkempi laadunvarmistus vaatisi todellista tietoa alueen puustosta, jota voidaan saada maastokäynneillä. Yleisesti metsän piirteitä ja tunnuslukuja pyritään estimoimaan metsän inventointimenetelmillä käyttäen kaukokartoitusaineistoja ja maastokäynneillä mitattuja koealoja, jonka vuoksi tätä esitettyä menetelmää voisi olla hyödyllistä käyttää inventointimenetelmillä tuotettuja rasterikuvilla eli käyttämällä jotain inventointimenetelmää segmentoitavan rasterikuvan pikseleiden alueella. Tällöin menetelmän tässä esitetyn menetelmän syötteenä ei olisi ilmakuvat ja laserkeilausaineisto vaan inventointimenetelmiä käyttäen tuotettu rasterikuva, jossa pikselin värikanavat vastaisivat esimerkiksi puulajikohtaisia puuston tilavuuksia.

Tuotettujen metsikkökuvioiden rajausten tarkennus puolestaan on silmämääräisestikin hyvin toimiva menetelmä, jolla voidaan korjata ainakin matalammalla resoluutiolla tuotettua kuviointia vastaamaan todellisia rajauksia. Muttei ole syytä olettaa, etteikö rajojen tarkennus toimisi myös samalla resoluutiolla tuotetulla aineistolla kuviorajojen tarkentamiseen, koska menetelmässä interpoloidaan kuvan reunaan vaikuttavat voimat mahdollistaen alipikselitarkkuuden.

Toteutetut kuvankäsittelymenetelmät toimivat metsikkökuvioiden tuottamisessa kaukokartoitusaineistosta ja näillä menetelmillä pystytään tuottamaan useimmissa tapauksissa riittävän tarkkoja metsikkökuvioita, mutta menetelmät ovat herkkiä aineiston suhteen ja parametrien suhteen. Siksi voisi olla hyödyllistä automatisoida optimaalisten parametrien etsintää kehittämällä mittari, jolla voidaan arvioida tuotettuja kuvioita metsänhoitajan käsin arvioimiin rajauksiin ja pyrkimällä löytämään parametrinti, jolla menetelmä tuottaa mahdollisimman samankaltaista kuviointia.

Haasteena metsikkökuvioiden tuottamisessa on myös eri aineistojen temporaalinen ero eli aineistojen mittaamisen välillä voi olla useampia vuosia ja tällä välillä metsikköön on voitu tehdä useita toimenpiteitä. Siksi voisi olla hyödyllistä käyttää avoimena aineistona Suomen Metsäkeskuksen julkaisemaa metsänkäyttöilmoitusaineistoa, jonka perusteella aineistosta voitaisiin rajata tai esikäsitellä ne alueet, joille on tehty tai ainakin suunniteltu tehtäväksi toimenpiteitä kuvantamisen välillä. Lisäksi voisi olla hyödyllistä käyttää puuston kasvunlaskentamenetelmää tai kasvumallia ainakin laserkeilausaineistosta johdetun puuston korkeuksien korjaamiseksi samalle ajankohdalle ilmakuvan kanssa.

Muita mahdollisia aineistoja menetelmän käytettäväksi ovat satelliittikuvat, joista ainakin Landsat-8, Sentinel-2 ja Radarsat-2 -satelliittien aineistot olisivat kiinnostavia ja todennäköisesti toisivat lisäinformaatiota laadun parantamiseksi menetelmään metsikkökuvioiden tuottamiseksi. Näistä etenkin Sentinel-2:n monispektrikuvat ovat jo todettu hyväksi ja käyttökelpoisiksi, mutta satelliittikuvien spatiaalinen erotuskyky on kuitenkin selvästi ilmakuvia heikompi.

Metsikkökuvioiden tuottamisessa käytettyjen aineistojen ja kuvien kehittämisen lisäksi voisi olla mahdollista kehittää verkkopalvelu, joka tuottaa kiinteistörajojen ja avoimen maas-

totietokannan tiestön puitteissa kuvioinnin ja kuvioille metsävaratietojen inventoinnin reaaliaikaisesti, mutta tämä vaatisi enemmän laskenta-aikaa vaativan aineistojen esikäsittelyn olevan valmiiksi esilaskettua. Tällaisessa verkkopalvelussa voitaisiin hyödyntää standardeja karttapalvelurajapintoja, jolloin se olisi integroitavissa suoraan olemassa oleviin paikkatietosovelluksiin.



# LÄHTEET

- AKSAKAL-KOCAMAN, Sultan, 2008. *Sensor Modeling and Validation for Linear Array Aerial and Satellite Imagery*. Tohtorinväitöskirja. ETH Zurich.
- BARTELS, M.; WEI, Hong; MASON, D.C., 2006. DTM Generation from LIDAR Data using Skewness Balancing. Teoksessa: *DTM Generation from LIDAR Data using Skewness Balancing. 18th International Conference on Pattern Recognition (ICPR'06)*. IEEE.
- BERTRAND, Gilles, 2007. ISMM05 SPECIAL ISSUE: ON THE DYNAMICS. Teoksessa: *ISMM05 SPECIAL ISSUE: ON THE DYNAMICS*.
- BEUCHER, Serge; MEYER, Fernand, 1993. The Morphological Approach to Segmentation: The Watershed Transformation. Teoksessa: vol. Vol. 34, s. 433–481. ISBN 9781315214610. Saatavilla osoitteesta DOI: 10.1201/9781482277234-12.
- BICKING, Barbara, 2009. *Lidar Solutions in ArcGIS: Estimating Forest Density and Height* [[https://www.esri.com/arcgis-blog/products/arcgis-desktop/3d-gis/lidar-solutions-in-arcgis\\_part4-estimating-forest-density-and-height/](https://www.esri.com/arcgis-blog/products/arcgis-desktop/3d-gis/lidar-solutions-in-arcgis_part4-estimating-forest-density-and-height/)]. Luettu: 2.2.2010.
- BOLSTAD, Paul, 2012. *GIS Fundamentals: A First Text on Geographic Information Systems, Fifth Edition*. ISBN 9781506695877.
- BOVIK, Alan C., 2009. *The Essential Guide to Image Processing*. USA: Academic Press, Inc. ISBN 0123744571.
- BUTENUTH, Matthias; HEIPKE, Christian, 2012. Network snakes: Graph-based object delineation with active contour models. *Mach. Vis. Appl.* Vol. 23, s. 91–109. Saatavilla osoitteesta DOI: 10.1007/s00138-010-0294-8.
- CAMPBELL, J.B.; WYNNE, R.H., 2011. *Introduction to Remote Sensing, Fifth Edition*. Guilford Publications. ISBN 9781609181772.
- CARREIRA-PERPIÑÁN, Miguel, 2015. A review of mean-shift algorithms for clustering.
- CHEN, Biwu; SHI, Shuo; GONG, Wei; ZHANG, Qingjun; YANG, Jian; DU, Lin; SUN, Jia; ZHANG, Zhenbing; SONG, Shalei, 2017. Multispectral LiDAR Point Cloud Classification: A Two-Step Approach. *Remote Sensing*. Vol. 9, s. 373. Saatavilla osoitteesta DOI: 10.3390/rs9040373.

- CHEN, Chuanfa; WANG, Yifu; LI, Yanyan; YUE, Tian-Xiang; WANG, Xin, 2017. Robust and Parameter-Free Algorithm for Constructing Pit-Free Canopy Height Models. *ISPRS International Journal of Geo-Information*. Vol. 6. Saatavilla osoitteesta DOI: 10.3390/ijgi6070219.
- COMANICIU, D.; PETER, M., 2002. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. 25, s. 281–288.
- COUSTY, Jean; BERTRAND, Gilles; NAJMAN, Laurent; COUPRIE, Michel, 2009. Watershed Cuts: Minimum Spanning Forests and the Drop of Water Principle. *IEEE transactions on pattern analysis and machine intelligence*. Vol. 31, s. 1362–1374. Saatavilla osoitteesta DOI: 10.1109/TPAMI.2008.173.
- DOUGLAS, David H.; PEUCKER, Thomas K., 2011. Algorithms for the Reduction of the Number of Points Required to Represent a Digitized Line or its Caricature. Teokessa: *Classics in Cartography*. Toim. DODGE, Martin. Chichester, UK: John Wiley & Sons, Ltd, s. 15–28. ISBN 9780470669488. Saatavilla osoitteesta DOI: 10.1002/9780470669488.ch2.
- GDAL/OGR CONTRIBUTORS, 2019. *GDAL/OGR Geospatial Data Abstraction software Library* [<https://gdal.org>].
- GONZALEZ, Rafael C.; WOODS, Richard E., 2006. *Digital Image Processing (3rd Edition)*. USA: Prentice-Hall, Inc. ISBN 013168728X.
- HERRING, John R., 2011. *OpenGIS® Implementation Standard for Geographic information - Simple feature access - Part 1: Common architecture* [[http://portal.opengeospatial.org/files/?artifact\\_id=25355](http://portal.opengeospatial.org/files/?artifact_id=25355)]. Tekninen raportti. Open Geospatial Consortium Inc.
- HONKAVAARA, Eija; MARKELIN, Lauri; NURMINEN, Kimmo, 2011. Digitaalinen ilmakekuvaus ja sen mahdollisuudet. *The Photogrammetric Journal of Finland*. Vol. 22, nro 3, s. 68–81.
- HUISMAN, O.; DE BY, R.A., 2009. *Principles of geographic information systems: an introductory textbook*. Netherlands: International Institute for Geo-Information Science ja Earth Observation. ITC Educational Textbook Series. ISBN 9789061642695.

- IVINS, Jim; PORRILL, John, 1993. Everything you always wanted to know about snakes (but were afraid to ask). *Revised June*.
- JAHKOLA, Oskari; HAPPONEN, Ari; KNUTAS, Antti; IKONEN, Jouni, 2017. What Should Application Developers Understand about Mobile Phone Position Data. Teoksessa: *What Should Application Developers Understand about Mobile Phone Position Data*, s. 171–178. Saatavilla osoitteesta DOI: 10.1145/3134302.3134346.
- JHS196, 2016. *EUREF-FIN -järjestelmän mukaiset koordinaatit Suomessa* [<http://www.jhs-suositukset.fi/web/guest/jhs/recommendations/196>]. Luettu: 26.1.2020.
- JHS197, 2016. *EUREF-FIN -koordinaattijärjestelmät, niihin liittyvät muunnokset ja karttalehtijako* [<http://www.jhs-suositukset.fi/web/guest/jhs/recommendations/197>]. Luettu: 26.1.2020.
- JIANG, Zhangyan; HUETE, Alfredo; DIDAN, K.; MIURA, Tomoaki, 2008. Development of a two-band enhanced vegetation index without a blue band. *Remote Sensing of Environment*. Vol. 112, s. 3833–3845. Saatavilla osoitteesta DOI: 10.1016/j.rse.2008.06.006.
- LAS Specification Version 1.4*, 2010 [[https://www.asprs.org/wp-content/uploads/2010/12/LAS\\_1\\_4\\_r13.pdf](https://www.asprs.org/wp-content/uploads/2010/12/LAS_1_4_r13.pdf)]. Luettu: 27.1.2020.
- LEHTINEN, Mikko; HAPPONEN, Ari; IKONEN, Jouni, 2008. Accuracy and Time to First Fix Using Consumer-grade GPS Receivers. Teoksessa: *Accuracy and Time to First Fix Using Consumer-grade GPS Receivers*, s. 334–340. Saatavilla osoitteesta DOI: 10.1109/SOFTCOM.2008.4669506.
- LEICA-GEOSYSTEMS, 2004. *ADS40 Large-Scale Digital Imagery* [[https://www.dlr.de/os/Portaldata/48/Resources/dokumente/ads40\\_flyer\\_leica.pdf](https://www.dlr.de/os/Portaldata/48/Resources/dokumente/ads40_flyer_leica.pdf)]. Luettu: 24.1.2020.
- LEPPÄNEN, V.; TOKOLA, Timo; MALTAMO, Matti; MEHTÄTALO, Lauri; PUSA, Tuomo; MUSTONEN, J., 2008. Automatic delineation of forest stands from lidar data. Vol. 38.

- LIANG, Xinlian et al., 2016. Terrestrial laser scanning in forest inventories. *ISPRS Journal of Photogrammetry and Remote Sensing*. Vol. 115, s. 63–77. ISSN 0924-2716. Saatavilla osoitteesta DOI: <https://doi.org/10.1016/j.isprsjprs.2016.01.006>. Theme issue 'State-of-the-art in photogrammetry, remote sensing and spatial information science'.
- MALTAMO, Matti; KANGAS, Annika, 2004. *Metsän mittaus ja kartoitus*. Joensuu: Silva Carelica 40. Joensuun Yliopisto. ISBN 9526103009.
- MIELCAREK, Miłosz; STEREŃCZAK, Krzysztof; KHOSRAVIPOUR, Anahita, 2018. Testing and evaluating different LiDAR-derived canopy height model generation methods for tree height estimation. *International Journal of Applied Earth Observation and Geoinformation*. Vol. 71. Saatavilla osoitteesta DOI: [10.1016/j.jag.2018.05.002](https://doi.org/10.1016/j.jag.2018.05.002).
- MÅRTENSSON, Ulrik, 2011. *Introduction to Remote Sensing and Geographical Information Systems*. Tekninen raportti. Department of Physical Geography ja Ecosystems Sciences Lund University.
- OKSANEN, Juha. *Digital Elevation Model* [<https://www.maanmittauslaitos.fi/en/research/interesting-topics/digital-elevation-model>]. Luettu: 29.1.2020.
- PDAL CONTRIBUTORS, 2018. *PDAL Point Data Abstraction Library* [<https://doi.org/10.5281/zenodo.2556738>]. Saatavilla osoitteesta DOI: [10.5281/zenodo.2556738](https://doi.org/10.5281/zenodo.2556738). Luettu: 26.1.2020.
- PEKKARINEN, A., 2002. A method for the segmentation of very high spatial resolution images of forested landscapes. *International Journal of Remote Sensing - INT J REMOTE SENS*. Vol. 23, s. 2817–2836. Saatavilla osoitteesta DOI: [10.1080/01431160110076162](https://doi.org/10.1080/01431160110076162).
- ROERDINK, Jos; MEIJSTER, A., 2003. The Watershed Transform: Definitions, Algorithms and Parallelization Strategies. *Fundam Inf*. Vol. 41.
- SHAN, J.; TOTH, C.K., 2018. *Topographic Laser Ranging and Scanning: Principles and Processing, Second Edition*. Taylor & Francis. ISBN 9781351650427.
- SHIH, Frank Y., 2010. *Image Processing and Pattern Recognition*. John Wiley & Sons, Ltd. ISBN 9780470590416.

- SOLOMON, Justin; CRANE, Keenan; BUTSCHER, Adrian; WOJTAN, Chris, 2014. A General Framework for Bilateral and Mean Shift Filtering.
- STOCKMAN, George; SHAPIRO, Linda G., 2001. *Computer Vision*. 1st. USA: Prentice Hall PTR. ISBN 0130307963.
- SZELISKI, Richard, 2011. *Computer vision algorithms and applications*. London; New York: Springer. ISBN 9781848829343.
- TEMPFLI, Klaus; HUURNEMAN, G; BAKKER, Wim; JANSSEN, L, 2009. *Principles of remote sensing: an introductory textbook*. ITC. ISBN 9789061642701.
- TOMASI, C.; MANDUCHI, R., 1998. Bilateral Filtering for Gray and Color Images. Teok-  
sessa: *Bilateral Filtering for Gray and Color Images. Proceedings of the Sixth International Conference on Computer Vision*. USA: IEEE Computer Society, s. 839–846. ISBN 8173192219.
- WEHR, Aloysius; LOHR, Uwe, 1999. Airborne laser scanning—an introduction and overview. *ISPRS Journal of Photogrammetry and Remote Sensing*. Vol. 54, nro 2-3, s. 68–82.
- VOSSelman, G.; MAAS, Hans-Gerd, 2010. *Airborne and Terrestrial Laser Scanning*. Whittles. ISBN 9781904445876.
- XU, Chenyang; PRINCE, Jerry L., 1998. Snakes, Shapes, and Gradient Vector Flow. *IEEE TRANSACTIONS ON IMAGE PROCESSING*. Vol. 7, nro 3, s. 359–369.